



Eclipse Rich Client Platform
Discover the Anatomy of an
Application Based On Eclipse

Wayne Beaton
Evangelist, Eclipse Foundation

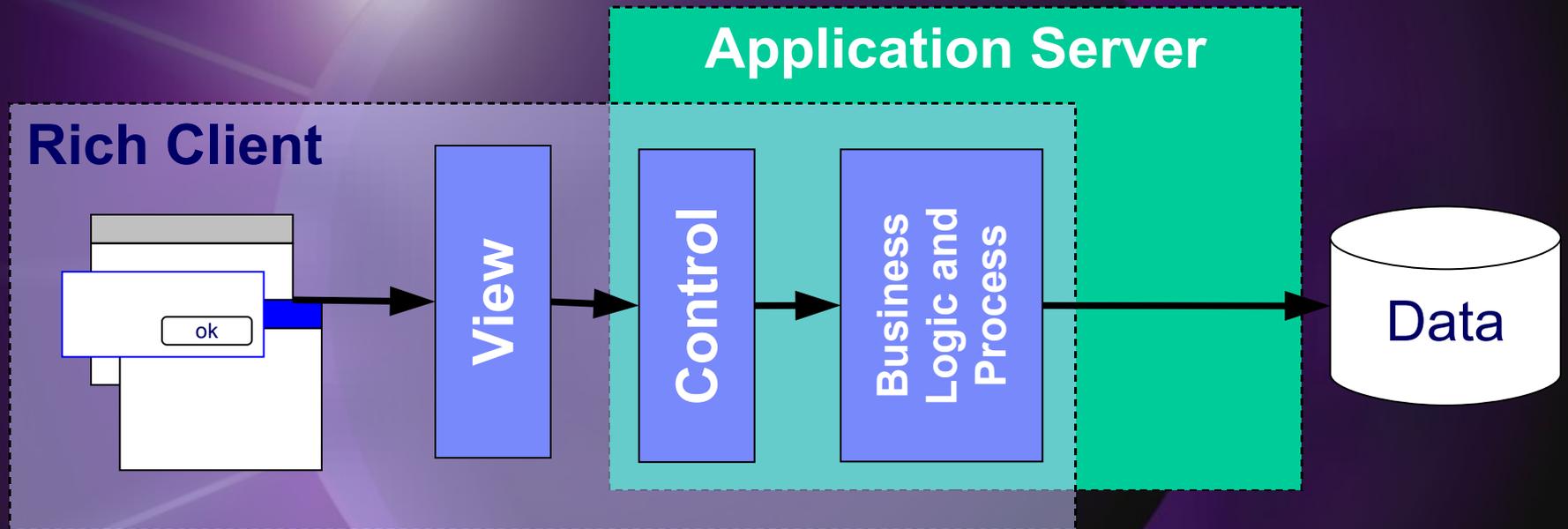
Agenda

- What is the Eclipse Rich Client Platform?
- Overview of the application
- Code walkthrough
 - Application
 - Models
 - Views
- Summary



What is the Eclipse Rich Client Platform?

What is a Rich Client?



Composition of Components



Eclipse SDK/Java IDE

Generic IDE Components



Eclipse Rich Client Platform





Overview of the Application

Eclipse Organizer

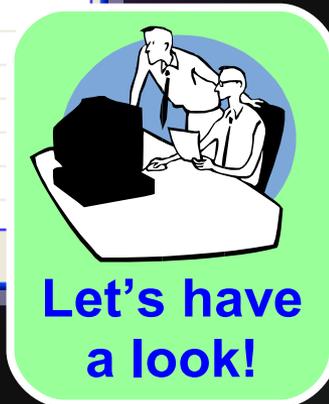
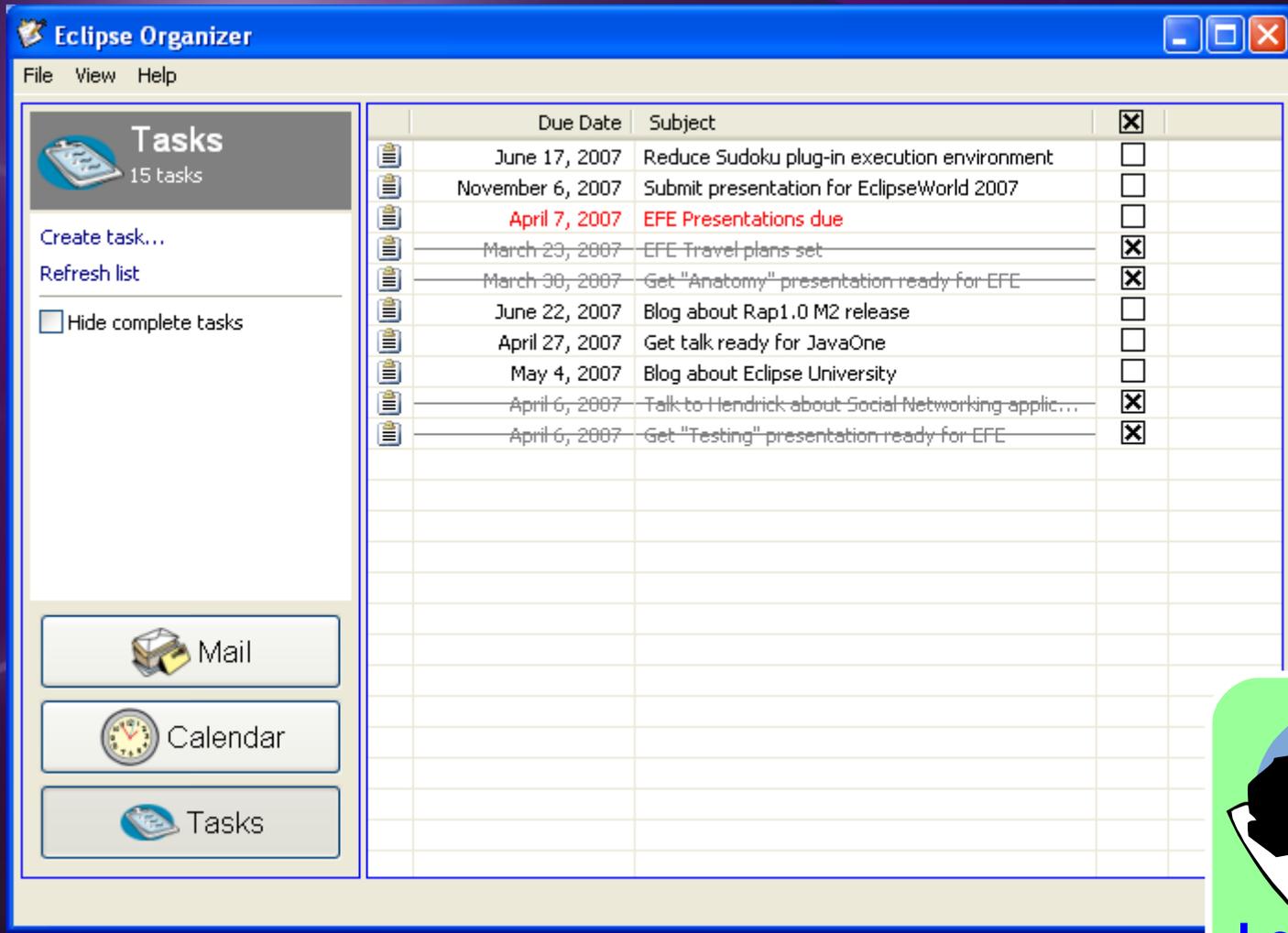
- E-Mail, Task List, Calendar
- Composition of components
- Custom branding
- Extensible
 - Back ends
 - Features
 - Drop handlers
 - ...

*WARNING:
The sample application is
under development!*



<http://www.eclipse.org/evangelism/samples/organizer/install.php>

Organizer as an Eclipse RCP Application



Organizer as Eclipse Plug-ins

The screenshot displays the Eclipse IDE interface with several plug-in windows. The 'Update the...' window is the central focus, showing a task titled 'Update the Evangelism WebSite' with a due date of 'Tuesday, October 31, 2006' and a priority of 'Normal'. Below the task details is a text area with the placeholder text 'Add information about the RCP Webinar.' To the left, the 'Package Explorer' shows a project structure with folders like 'EclipseBanking', 'Pictures', and 'Wayne's Tasks'. The 'Image Preview' window at the bottom left shows a photo of a person in a body of water. The 'Tasks' view at the bottom right is a table with columns for 'Due Date', 'Subject', and a checkbox. It lists two tasks: 'November 8, 2006 Update the Evangelism WebSite' and 'Get a better picture for the website'. A calendar pop-up for November 2006 is overlaid on the 'Tasks' view, with the 8th of the month highlighted.

Java - Update the... - Eclipse SDK

File Edit Navigate Search Project Run Window Help

Package Explorer Hierarchy

Update the... Subject: Update the Evangelism WebSite
Due date: Tuesday, October 31, 2006 Priority: Normal
Reminder date: Completed

Add information about the RCP Webinar.

Outline An outline is not available.

Image Preview

Problems Javadoc Declaration Tasks

Due Date	Subject	
November 8, 2006	Update the Evangelism WebSite	<input type="checkbox"/>
	Get a better picture for the website	<input type="checkbox"/>

November 2006

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

Today None

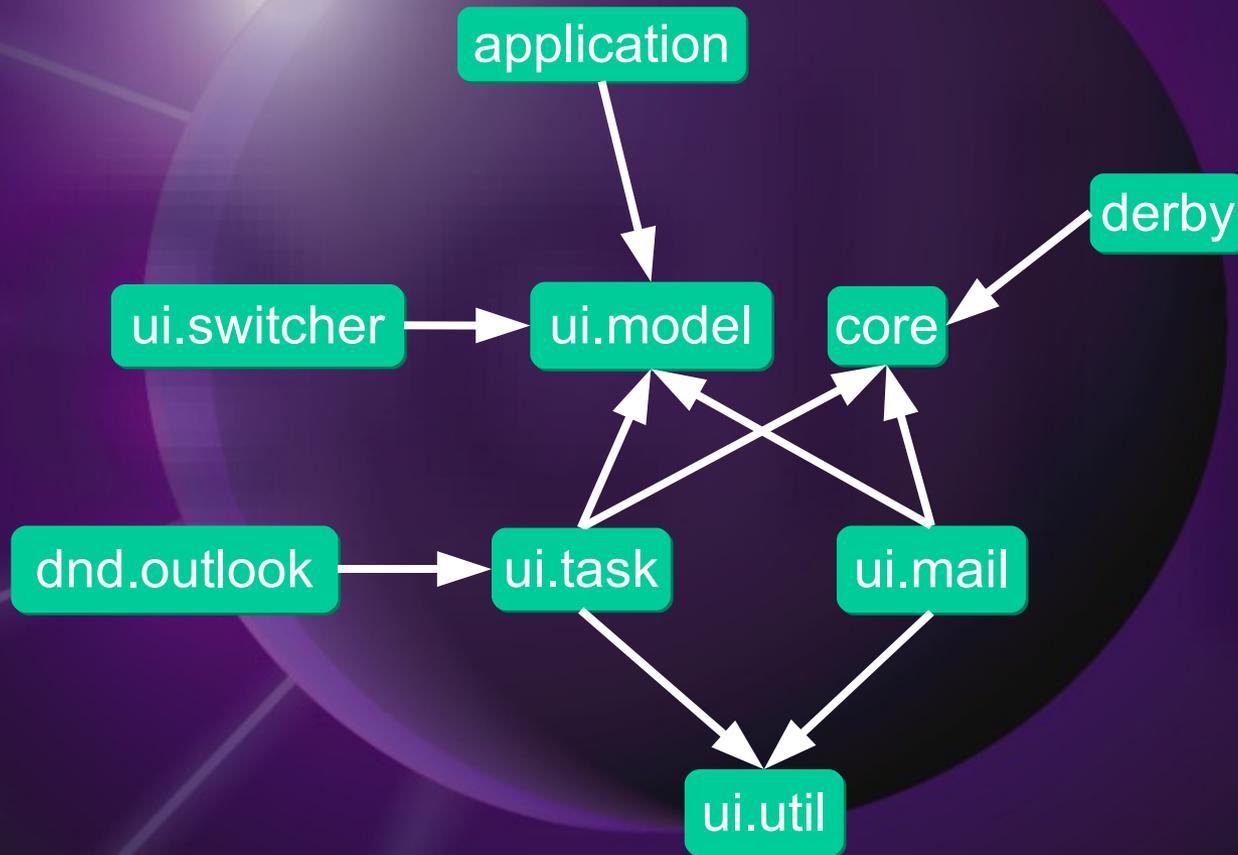
Projects

- +  org.eclipse.nebula.widgets.cdatetime
- +  org.eclipse.samples.organizer.application
- +  org.eclipse.samples.organizer.core
- +  org.eclipse.samples.organizer.derby
- +  org.eclipse.samples.organizer.dnd.outlook
- +  org.eclipse.samples.organizer.ui.mail
- +  org.eclipse.samples.organizer.ui.model
- +  org.eclipse.samples.organizer.ui.switcher
- +  org.eclipse.samples.organizer.ui.task
- +  org.eclipse.samples.organizer.ui.util

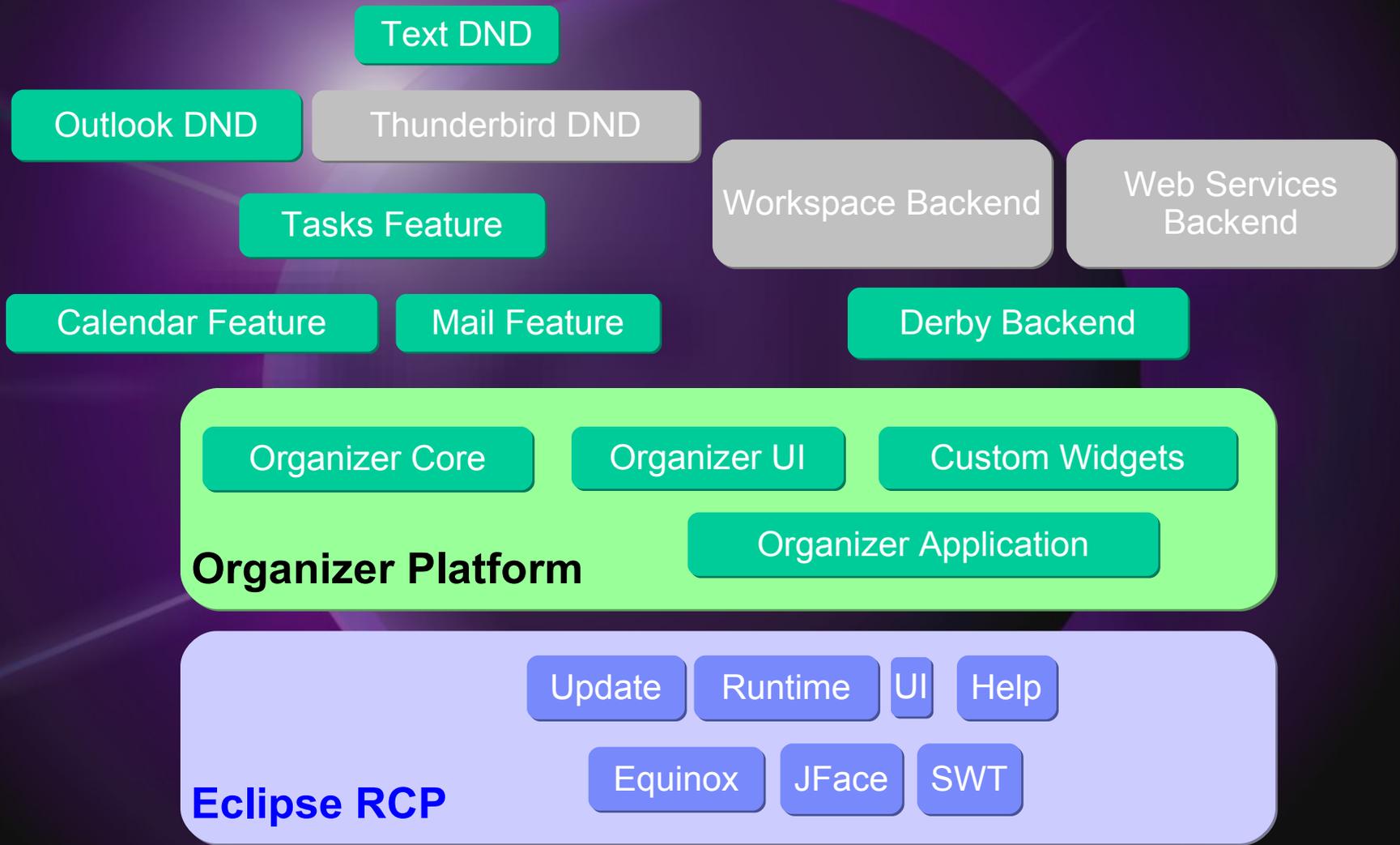
Components



Dependency Graph



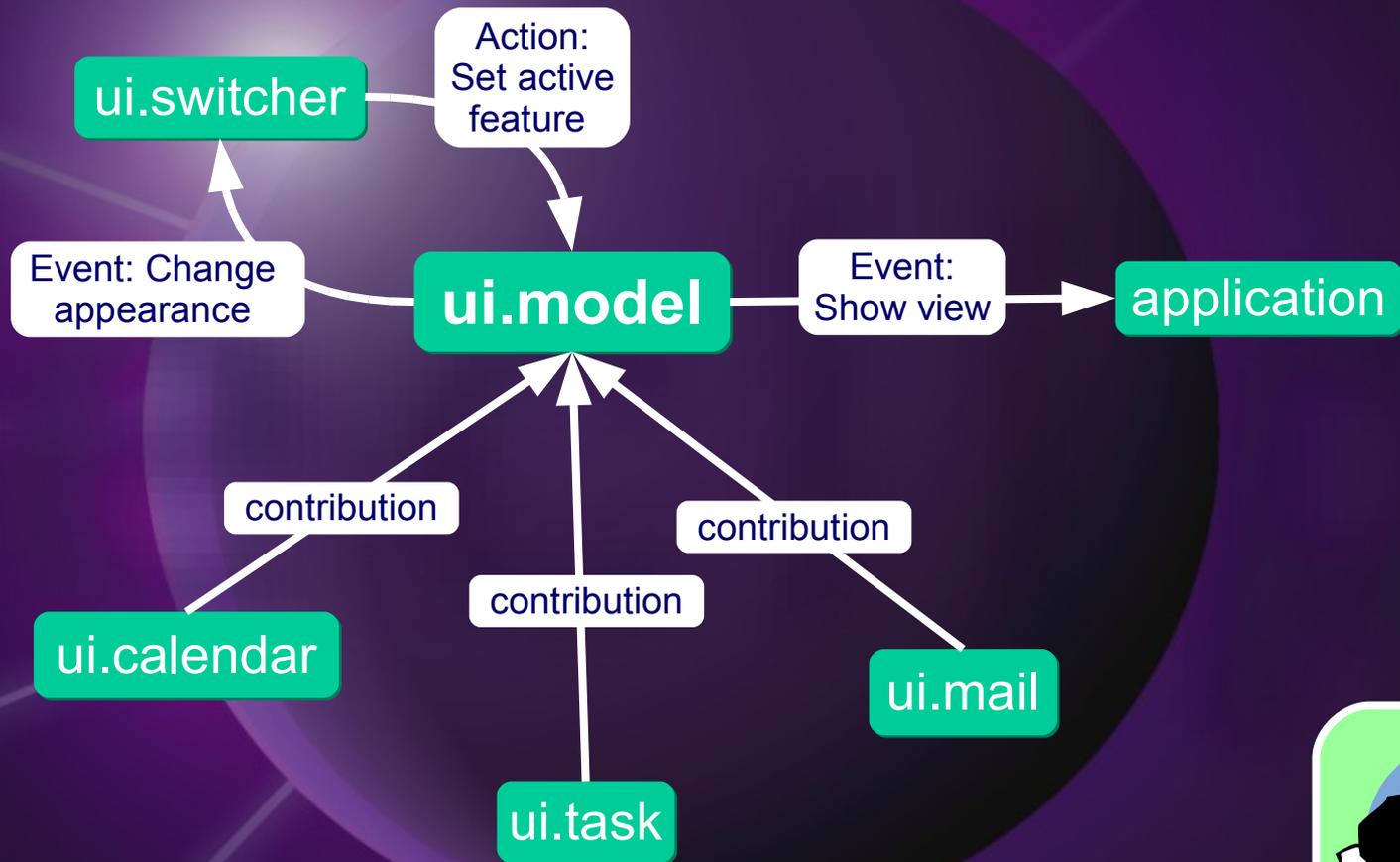
Platform



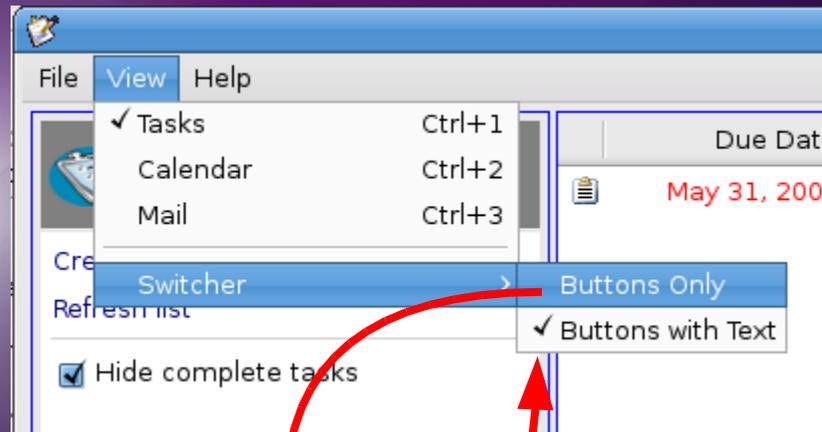
Models Component

- Object Model
- Pluggable support for multiple back ends
- Change notification using observer pattern

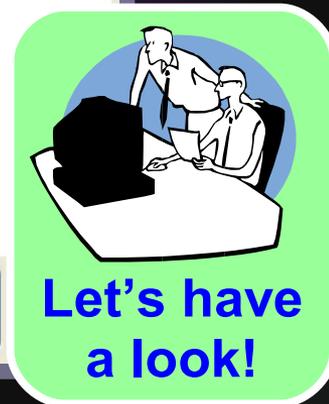
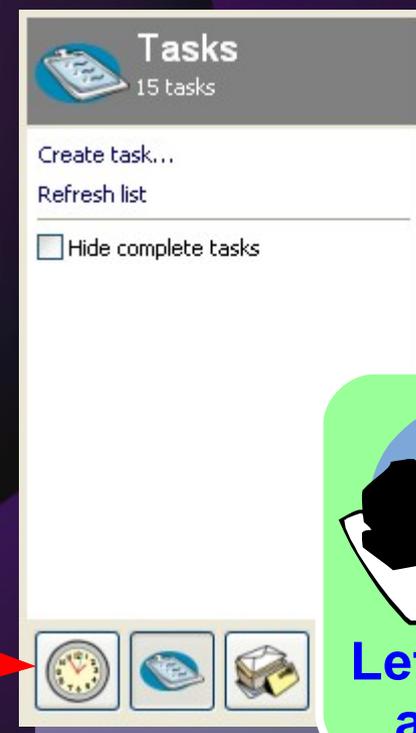
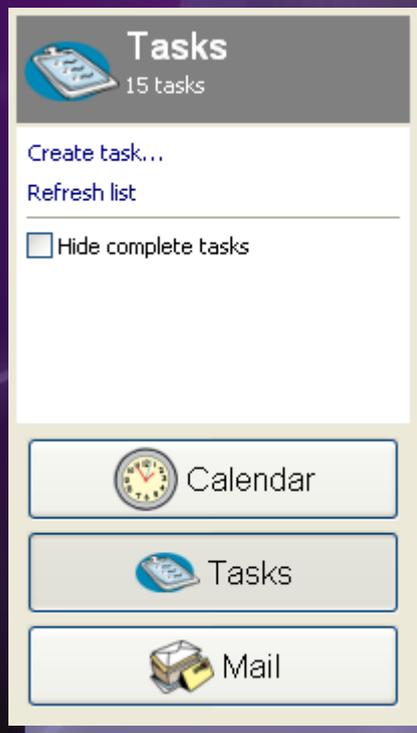
UI Model



Switcher UI Component

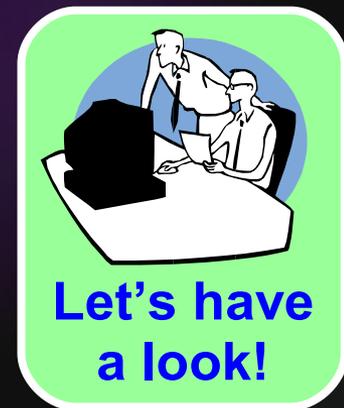


ui.model



Task UI Component

- **TasksView**
 - Dynamic Table
 - Extensible Drag and Drop
- **ImportTasksWizard**
- **TaskEditor**



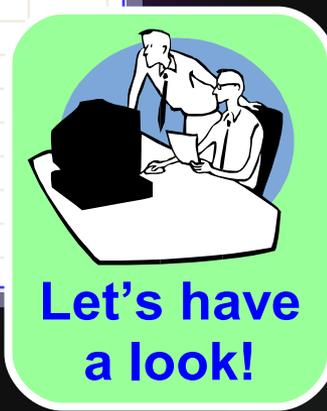
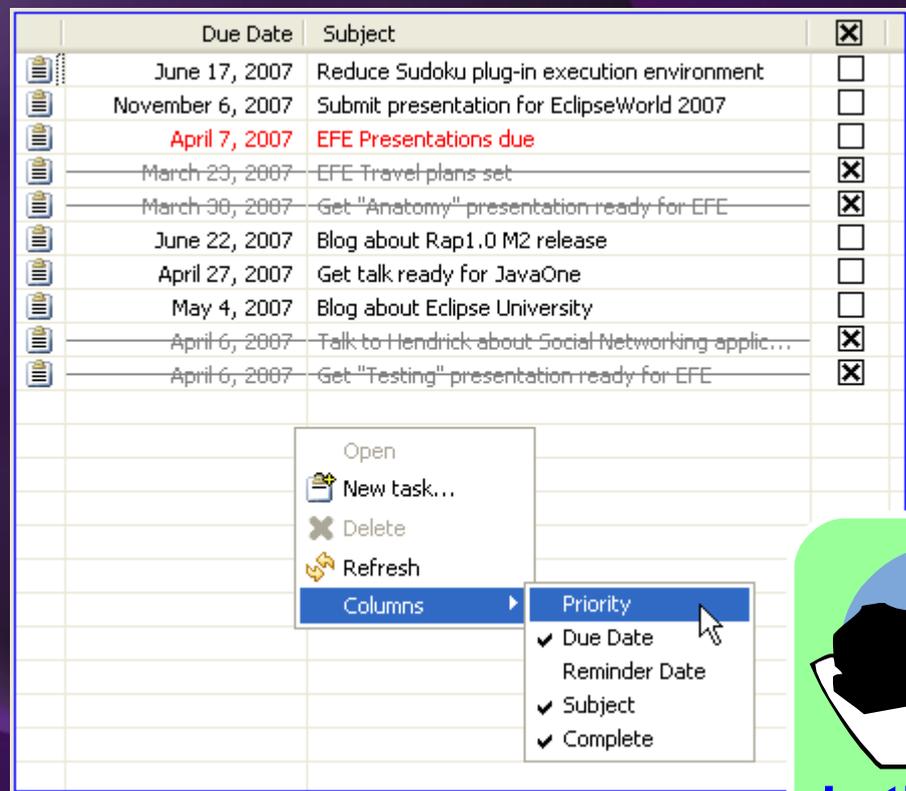
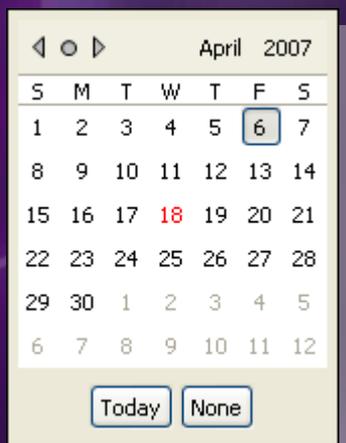
Save State

```
IWorkbenchConfigurer#setSaveAndRestore (boolean)
```

```
@Override
public void saveState(IMemento memento) {
    // Save the column state.
    IMemento columnsMemento = memento.createChild("columns");
    for (TableViewerColumnDefinition column : viewer.getColumnDefinitions()) {
        IMemento columnMemento = columnsMemento.createChild("column");
        columnMemento.putString("id", column.getId());
        columnMemento.putInteger("width", column.getWidth());
        columnMemento.putString("visible", String.valueOf(column.isVisible()));
    }
    // Save the sort state.
    IMemento sortMemento = memento.createChild("sort");
    sortMemento.putString("column", viewer.getSortColumnId());
    sortMemento.putString("direction", viewer.getSortDirection() == SWT.UP ? "UP" : "DOWN");
}
```

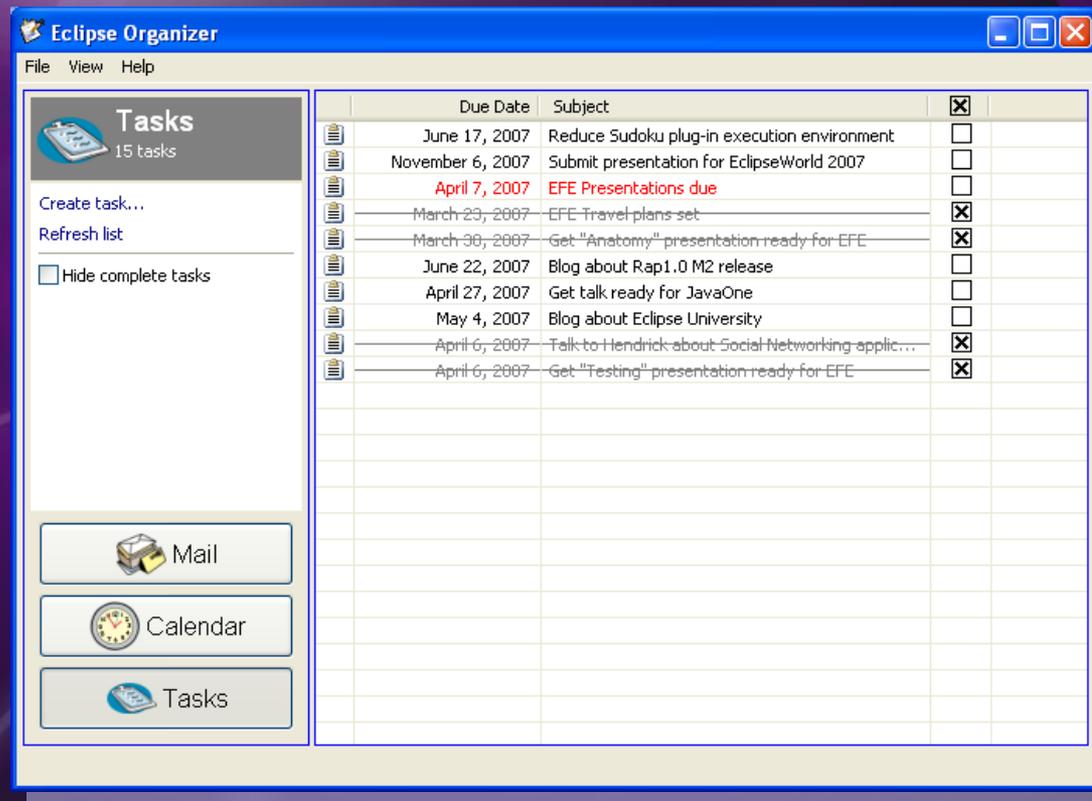
UI Utils Component

- “Super” TableViewer
- Date Picker



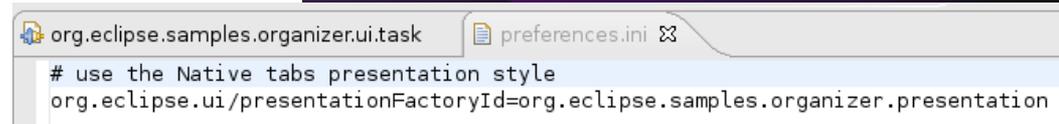
Custom Presentation (1/2)

- Tunes the appearance of workbench
- Very plain: no decoration around view stacks



Custom Presentation (2/2)

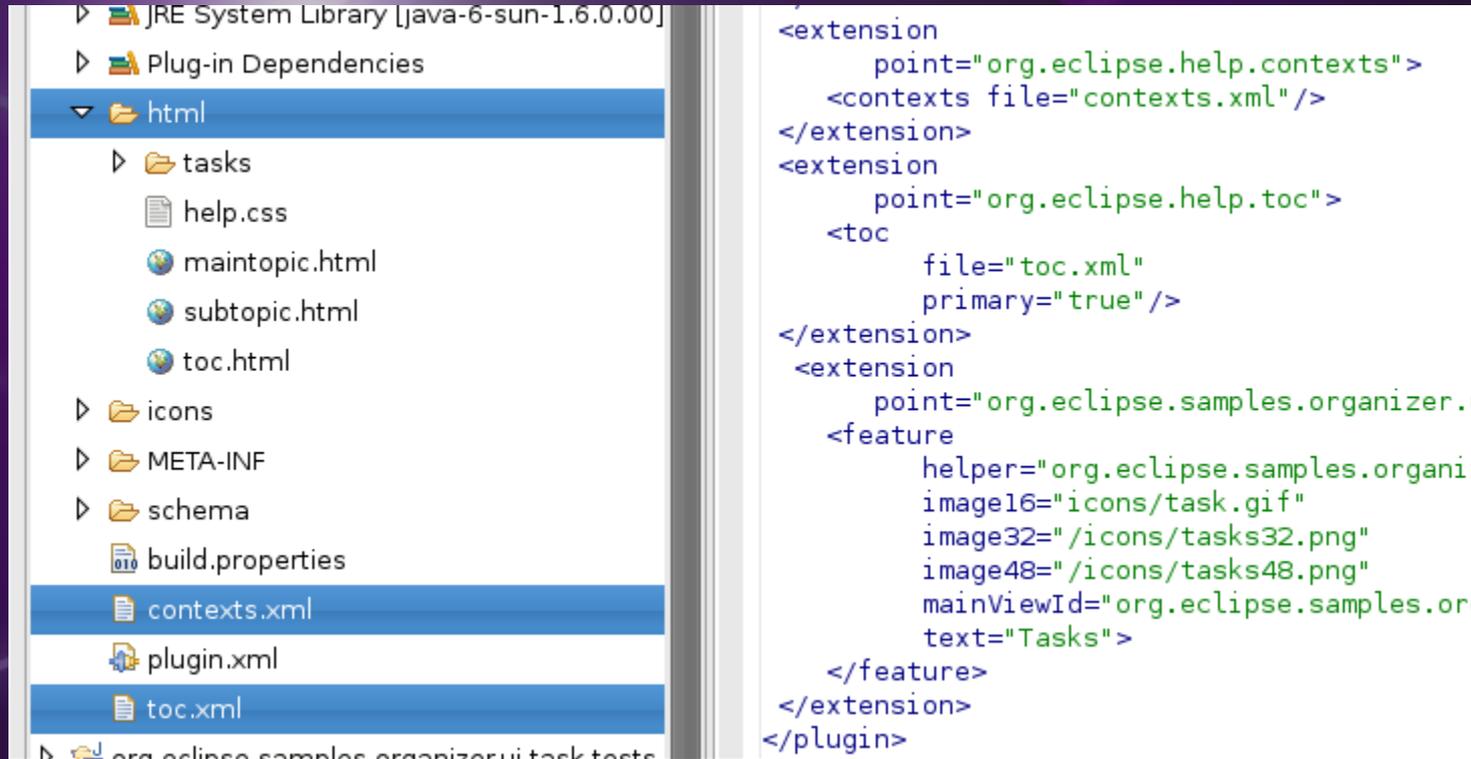
```
<extension
  id="product"
  point="org.eclipse.core.runtime.products">
  <product
    application="org.eclipse.samples.organizer.application.application"
    name="Eclipse Organizer">
    <property
      name="appName"
      value="Eclipse Organizer">
    </property>
    <property
      name="aboutImage"
      value="icons/about.gif">
    </property>
    <property
      name="aboutText"
      value="The Eclipse Organizer&#x0A;&#x0A;Copyright &amp;copy; 2007, The Eclipse F
    </property>
    <property
      name="windowImages"
      value="icons/task_16.gif,icons/task_32.gif">
    </property>
    <property
      name="preferenceCustomization"
      value="preferences.ini"/>
    </product>
  </extension>
  <extension
    point="org.eclipse.ui.presentationFactories">
    <factory
      class="org.eclipse.samples.organizer.presentation.OrganizerPresentationFactory"
      name="Organizer Presentation"
      id="org.eclipse.samples.organizer.presentation"/>
    </extension>
  </plugin>
```



- ▶ OrganizerPartPresentation.java
- ▼ OrganizerPresentationFactory.java
 - ▶ OrganizerPresentationFactory
 - createEditorPresentation(Composite, IStackPresentationSite)
 - createStandaloneViewPresentation(Composite, IStackPresentationSite, boolean)
 - createViewPresentation(Composite, IStackPresentationSite)



Help and Context-Sensitive Help (1/2)



The screenshot displays the Eclipse IDE interface. On the left, the Project Explorer shows a project structure with the following items:

- JRE System Library [java-6-sun-1.6.0.00]
- Plug-in Dependencies
- html
 - tasks
 - help.css
 - maintopic.html
 - subtopic.html
 - toc.html
 - icons
 - META-INF
 - schema
 - build.properties
 - contexts.xml
 - plugin.xml
 - toc.xml

On the right, the XML Editor shows the configuration for the plugin, with the following code:

```
<extension
  point="org.eclipse.help.contexts">
  <contexts file="contexts.xml"/>
</extension>
<extension
  point="org.eclipse.help.toc">
  <toc
    file="toc.xml"
    primary="true"/>
</extension>
<extension
  point="org.eclipse.samples.organizer..."
  <feature
    helper="org.eclipse.samples.organizer..."
    image16="icons/task.gif"
    image32="/icons/tasks32.png"
    image48="/icons/tasks48.png"
    mainViewId="org.eclipse.samples.org...
    text="Tasks">
  </feature>
</extension>
</plugin>
```

Help and Context-Sensitive Help (2/2)

```
private void createDetailsField(Composite composite) {
    final Text detailsField = new Text(composite, SWT.MULTI | SWT.WRAP | SWT.V_SCROLL | SWT.BORDER);
    detailsField.setText(task.getDetails() == null ? "" : task.getDetails());
    GridData gridData = new GridData();
    gridData.horizontalAlignment = SWT.FILL;
    gridData.grabExcessHorizontalSpace = true;
    gridData.horizontalSpan = 3;
    gridData.widthHint = 300;
    gridData.verticalAlignment = SWT.FILL;
    gridData.grabExcessVerticalSpace = true;
    gridData.heightHint = 300;
    detailsField.setLayoutData(gridData);
    detailsField.addModifyListener(new ModifyListener() {
        public void modifyText(ModifyEvent e) {
            task.setDetails(detailsField.getText());
        }
    });
    addDropListener(detailsField);
    PlatformUI.getWorkbench().getHelpSystem()
        .setHelp(detailsField, "org.eclipse.samples.organizer.ui.task.detailField");
}
```

Summary

- Application plug-in
- Model plug-in
- View plug-in
- Custom extension points
 - Back ends and Back end Factories
 - Drop handlers
- More information about the various technologies, see:
 - <http://www.eclipse.org/articles>
 - <http://www.eclipse.org/resources>

The Future

- This application is still under development
- Save and restore state of all views
- Revamp the object model with OSGi Services
- Key bindings
- Package for update sites
- Support for dragging from MacMail, Thunderbird, etc.
- Web Services based backend
- Improve ICU4J support
- More complete help contents
- Custom ICO for launcher
- Dig into Outlook to get dragged task details?

Recommended Reading

- Eclipse Rich Client Platform
 - By Jeff McAffer and Jean-Michel Lemieux
 - Addison-Wesley Professional
 - ISBN: 0321334612
- Official Eclipse 3.0 FAQs
 - by John Arthorne, Chris Laffra
 - Addison-Wesley Professional
 - ISBN: 0321268385
- Contributing to Eclipse: Principles, Patterns, and Plugins
 - By Erich Gamma, Kent Beck
 - Addison-Wesley Professional
 - ISBN: 0321205758

