

Eclipse RCP Part V

Automotive Financial Services Insurance Life Science & Healthcare Public Sector
Telecommunications & Media Travel & Logistics Utilities Automotive Financial
Services Insurance Life Science & Healthcare Public Sector Telecommunications
& Media Travel & Logistics Utilities Automotive Financial Services Insurance
Life Science & Healthcare Public Sector Telecommunications & Media Travel
& Logistics Utilities Automotive Financial Services Insurance Life Science
Healthcare Public Sector Telecommunications & Media Travel & Logistics
Utilities Automotive Financial Services Life Science & Healthcare Public
Sector Telecommunications & Media Travel & Logistics Utilities Automotive
Financial Services Insurance Life Science & Healthcare Public Sector
Telecommunications & Media Travel & Logistics Utilities Automotive
Financial Services Insurance Life Science & Healthcare Telecommunications
& Media Travel & Logistics Utilities Automotive Financial Services Insurance
Life Science & Healthcare Public Sector Telecommunications & Media Travel &
Logistics Utilities Automotive Financial Services Insurance Life Science &
Healthcare Public Sector Telecommunications & Media Travel & Logistics Utilities
Automotive Financial Services Insurance Life Science & Healthcare Public Sector

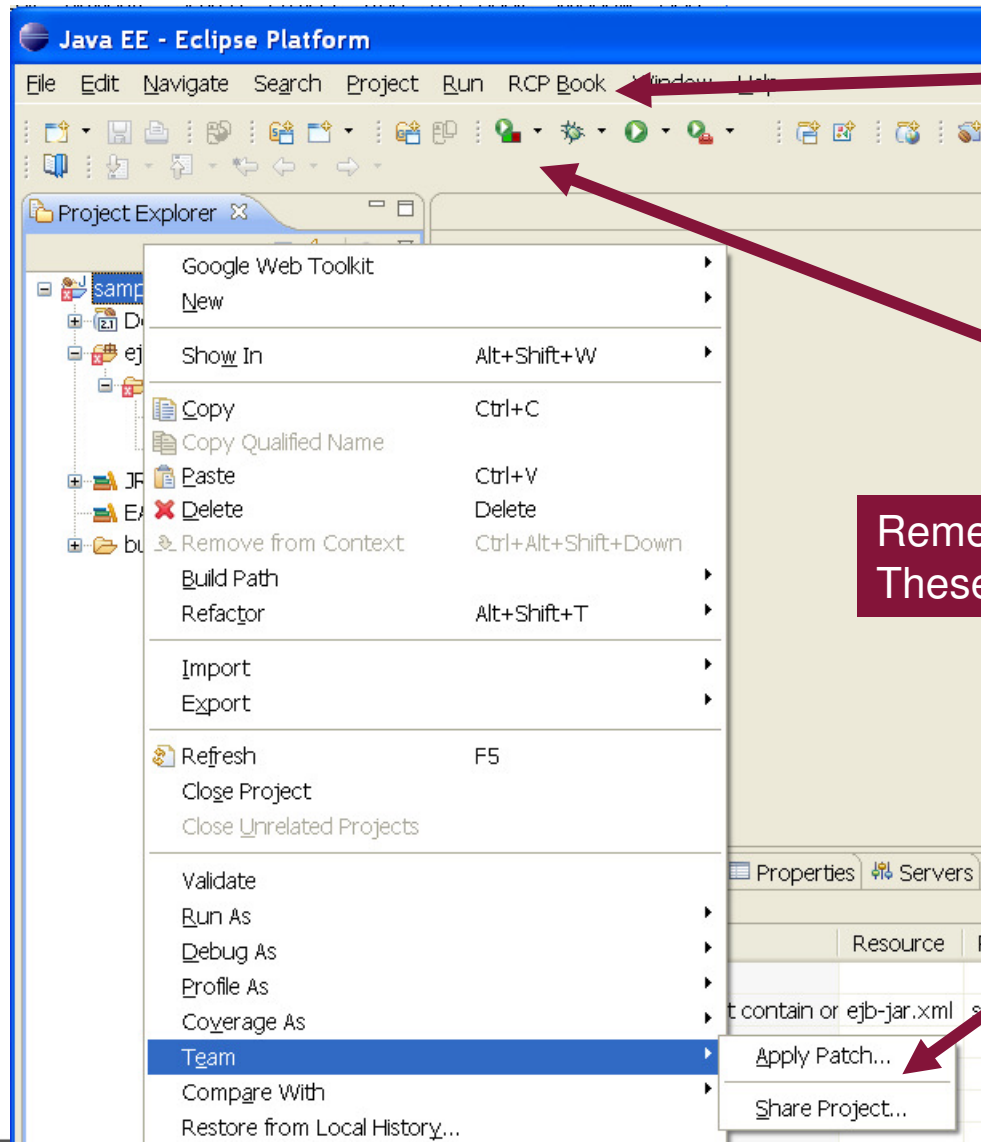


.consulting .solutions .partnership



- Understanding the following extension points
 - Commands
 - Handlers
 - Menu

Menu contribution



sample plug-in

coverage plug-in

Remember:
These plug-ins are still not loaded!

team plug-in

Background

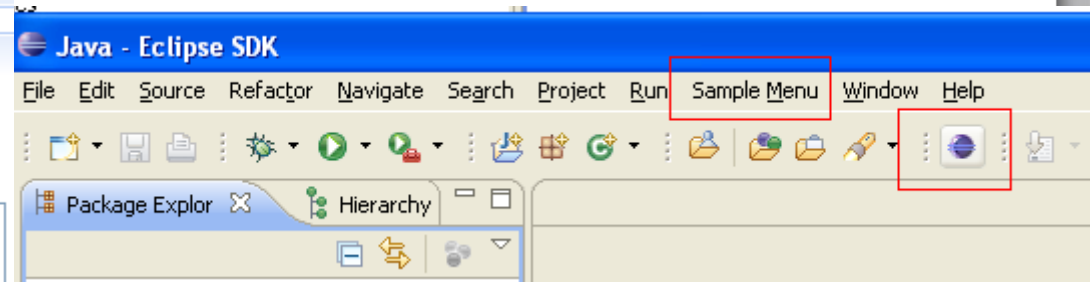
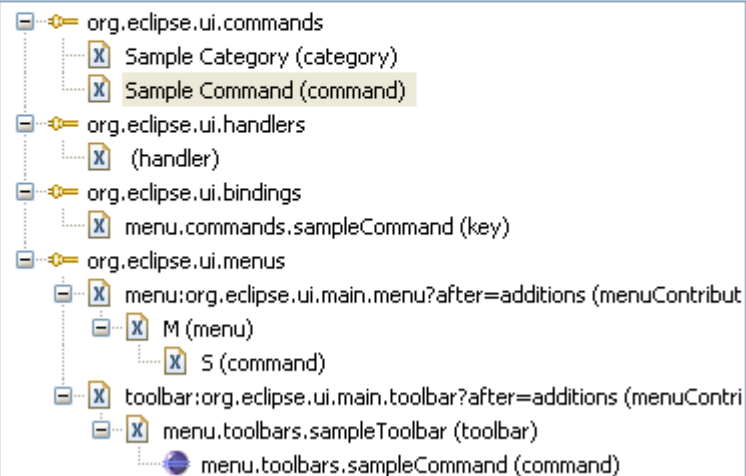
- It's possible for a plug-in to extend the gui

Extensions

All Extensions

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

type filter text



Simple Model

View



```
<extension
  point="org.eclipse.ui.menus">
  <menuContribution
    locationURI="menu:org.eclipse.ui.main.menu">
    <menu
      id="de.emcon.example.commands.menu.main.file"
      label="File">
      <command
        commandId="de.emcon.example.commands.exitcmd"
        icon="icons/exit.gif"
        label="Exit"
        style="push"
        tooltip="Schließt die Anwendung">
      </command>
    </menu>
  </menuContribution>
</extension>
```

Behavior

view plain copy to clipboard print ?

```
public class ExitHandler extends AbstractHandler implements IHandler
{
    @Override
    public Object execute(ExecutionEvent event) throws ExecutionException
    {
        HandlerUtil.getActiveWorkbenchWindow(event).close();
        return null;
    }
}
```

```
<extension
  point="org.eclipse.ui.commands">
  <command
    defaultHandler="de.emcon.example.commands.handler.ExitHandler"
    id="de.emcon.example.commands.exitcmd"
    name="Exit">
  </command>
</extension>
```

Building menu

org.eclipse.ui.menu

...

commandId



org.eclipse.ui.command

...

defaultHandler

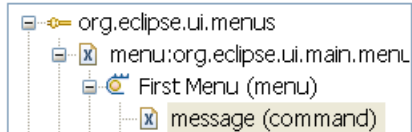


Class implements IHandler

All Extensions

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

type filter text



Add...

Remove

Extension Element Details

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

commandId*: Browse...

id:

mnemonic:

icon: Browse...

“org.eclipse.ui.file.print”

“org.eclipse.ui.file.revert”

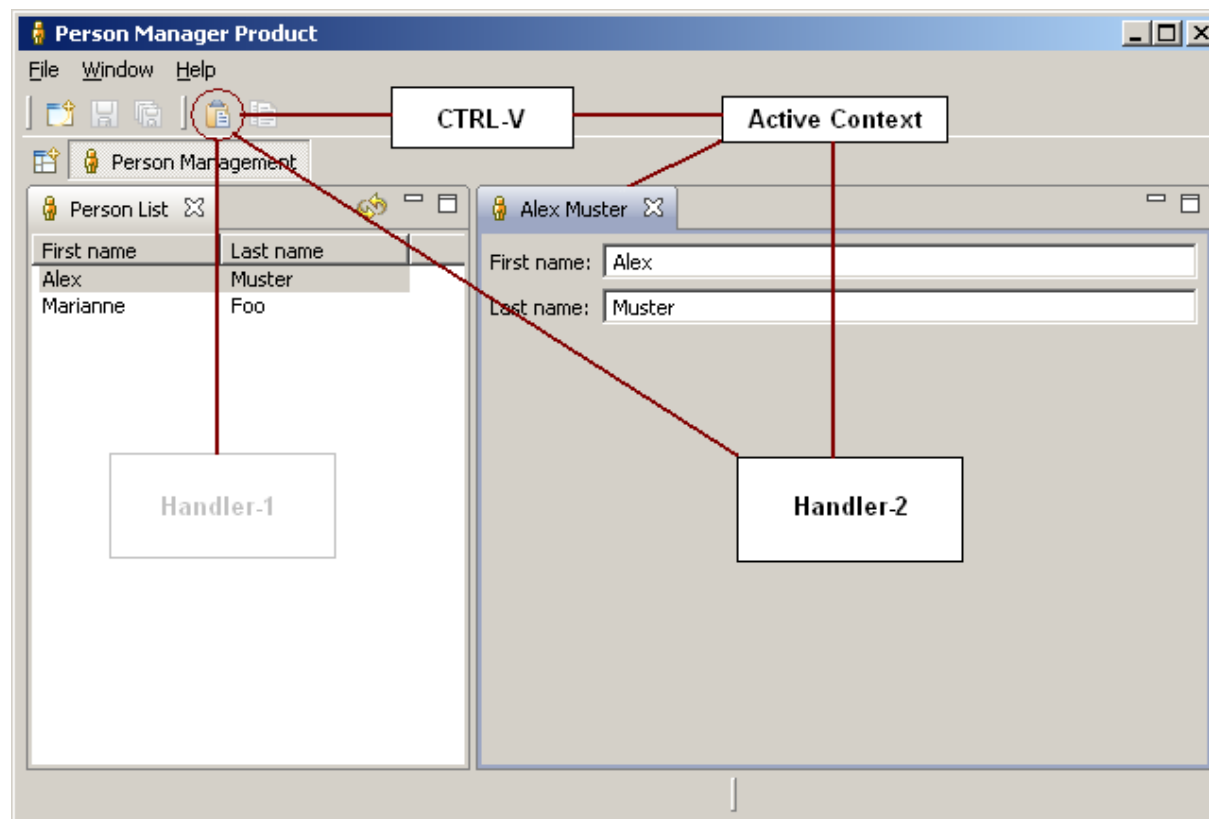
“org.eclipse.ui.file.properties”

→ defaultHandler="org.eclipse.PropertyDialogHandler"

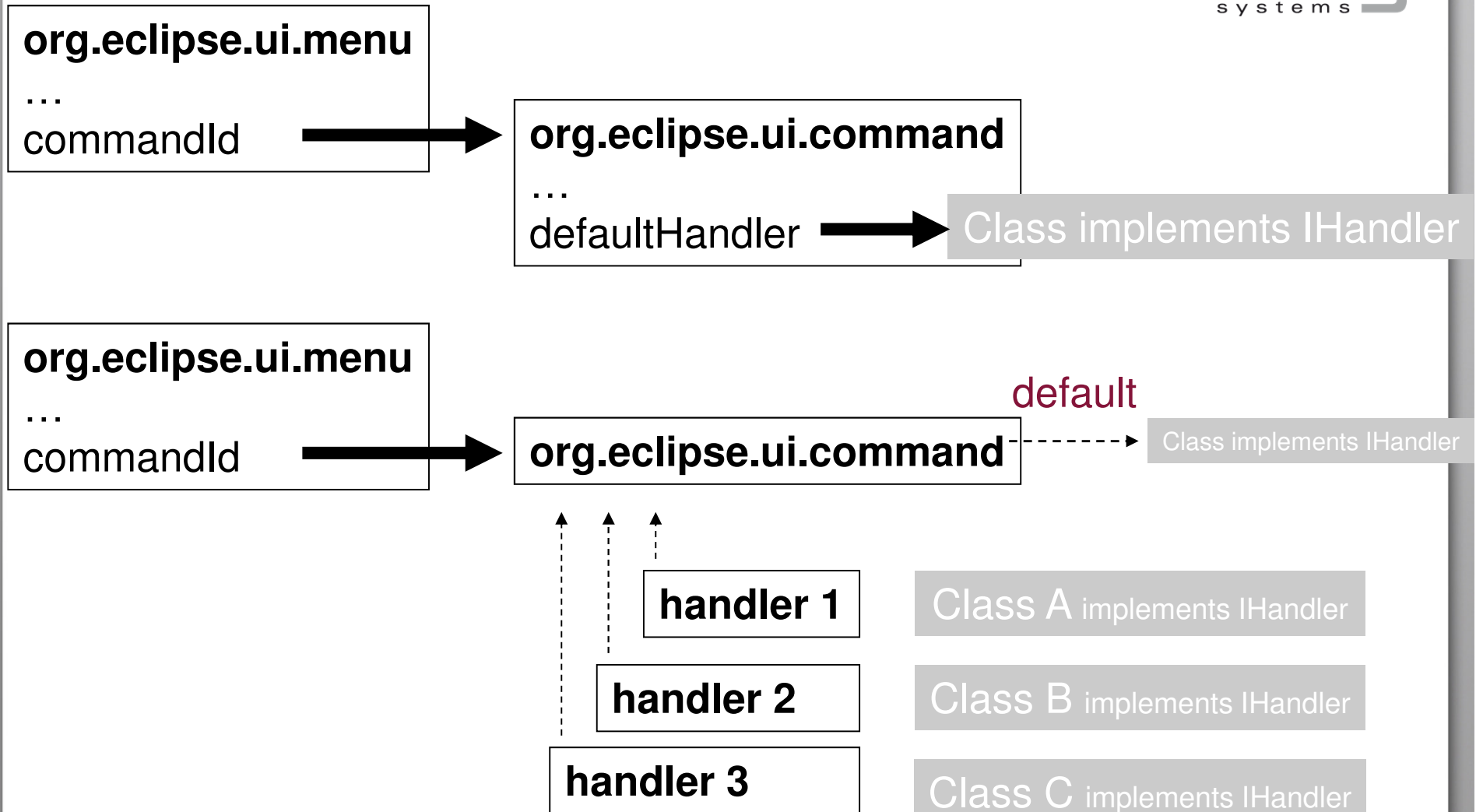
“org.eclipse.ui.edit.undo”

“my.app.whatever”

“my.app.doit”



Handlers




```
<handler
  class="com.eclipse_tips.commads.SomeCommandHandler1"
  commandId="com.eclipse-tips.commands.someCommand">
  <activeWhen>
    <with
      variable="selection">
        <iterate
          operator="or">
            <instanceof
              value="org.eclipse.core.resources.IFile">
            </instanceof>
          </iterate>
        </with>
      </activeWhen>
    </handler>
```

<activeWhen>

```
<handler
  class="com.eclipse_tips.commads.SomeCommandHandler2"
  commandId="com.eclipse-tips.commands.someCommand">
  <activeWhen>
    <with
      variable="selection">
        <iterate
          operator="or">
            <instanceof
              value="org.eclipse.core.resources
            </instanceof>
          </iterate>
        </with>
      </activeWhen>
    </handler>
```

How is the "specificness" of a handler is defined? It depends on the conditions that you give in the activeWhen expression. The order is defined in the ISources class. You can go thru the complete set there, to get a glimpse of the most commonly used ones, the order from least specific to most specific is like this:

- Active Context (activeContexts)
- Active Actions Sets (activeActionSets)
- Active Shell (activeShell)
- Active Workbench Window (activeWorkbenchWindow)
- Active Editor Id (activeEditorId)
- Active Part Id (activePartId)
- Current Selection (selection)

Link to predefined commands

- You can also link your menu items to the predefined commandID's like *org.eclipse.ui.help.aboutAction*, *org.eclipse.ui.file.exit*
- *Use the Browse Button to find the commands (type *about)*

All Extensions

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

type filter text

- org.eclipse.ui.menus
 - menu:org.eclipse.ui.main.menu (menu)
 - First Menu (menu)
 - Exit (command)
 - About (command)
 - Second Menu (menu)
 - Help (command)
 - toolbar:org.eclipse.ui.main.menu (m- org.eclipse.ui.commands
 - name (command)

Buttons: Add..., Remove, Up, Down

Extension Element Details

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

commandId*: Browse...

id:

mnemonic:

icon: Browse...

disabledIcon: Browse...

hoverIcon: Browse...

label:

tooltip:

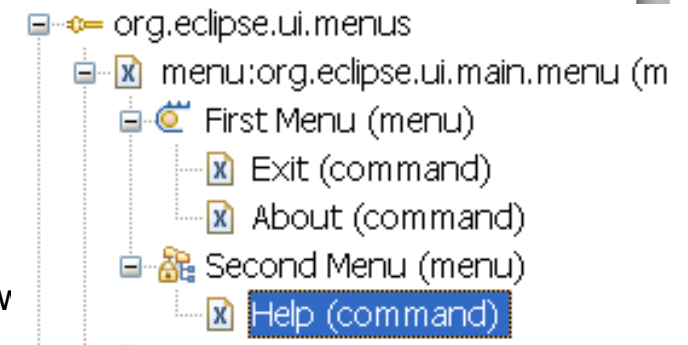
helpContextId:

style: ▼

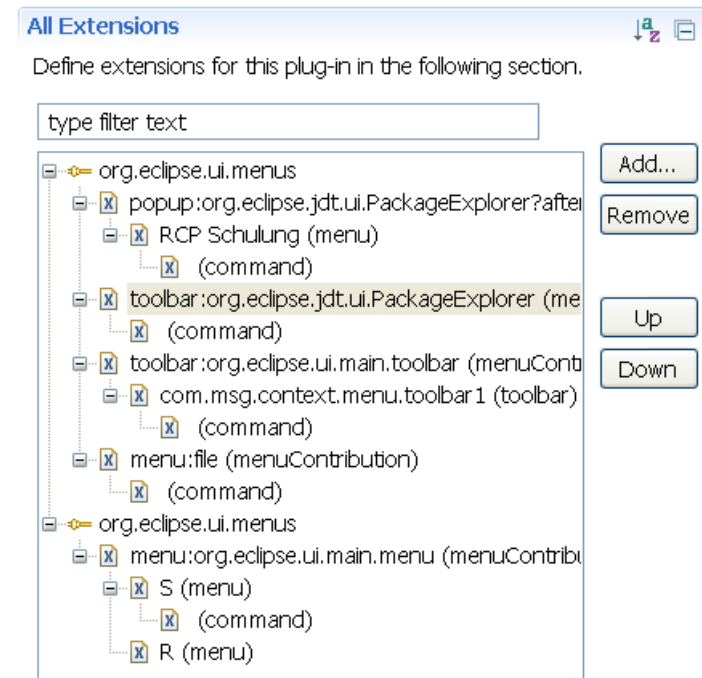
mode: ▼

```
MessageBox message = new MessageBox(shell, SWT.YES|SWT.NO);  
message.setMessage("hello world");  
int result = message.open();
```

- Create an empty rcp application com.msg.training.commands
- Create an empty plug-in project com.msg.training.commands.ui
- Add the some menu items
 - Extension: org.eclipse.ui.menu
 - Location URI: menu:org.eclipse.ui.main.menu
- Create the following menu items:
 - File → New, Close, Import
 - Window → Open new window, Open Perspective, Shov
 - “MyActions” → Show Hello World, Debug Print



- Create an empty plug-in (in a new workspace)
- Add the following menu contributions
 - menu:org.eclipse.ui.main.menu
 - popup:org.eclipse.jdt.ui.PackageExplorer
 - toolbar:org.eclipse.jdt.ui.PackageExplorer
 - toolbar:org.eclipse.ui.main.toolbar
 - menu:file



```
import org.eclipse.core.commands.AbstractHandler;  
and not  
import org.eclipse.ui.commands.AbstractHandler;
```

Vielen Dank für Ihre Aufmerksamkeit



.consulting .solutions .partnership

