Rich Client GUI's with RCP & RAP

Alexey Aristov
WeigleWilczek GmbH
aristov@weiglewilczek.com
What is Rich Client?

“A fat client or rich client is a computer (client) in client-server architecture networks which typically provides rich functionality independently of the central server. Originally known as just a 'client' or 'thick client', the name is derived from thin client, describing a computer which heavily depends on a server's applications.” (Wikipedia)
Advantages of Rich Clients:

- Reduces server load
- Offline operation
- Better performance
- Access to local resources
- Better OS integration
Available Rich Client Technologies:

- 4GL Environments
  - PowerBuilder
  - Borland Delphi
  - Visual Basic
  - <others>
- Microsoft .NET
- Java Based
  - SWING
  - Eclipse RCP
  - NetBeans RCP
- Adobe AIR
Eclipse RCP

• Technology behind Eclipse IDE
• A Platform for building Client applications with Rich functionality
• History
  • Hackers in the eclipse community started building non-IDE apps on 2.1
  • Since Eclipse 3.0 IDE aspects were separated from the Workbench
Advantages of Eclipse RCP

- Very modular
- Well defined programming model
- Familiar look and feel (uses native widgets)
- A lot of high level services and concepts (not only UI)
  - OSGi
  - Background Jobs
  - Data Binding
  - Commands
Build Rich Client GUI's with RCP & RAP

Example Application (Eclipse Trader)
RCP By Numbers

• Size of minimal RCP application (on the disk): 15Mb
• VM size of minimal RCP application: ~ 22 Mb
• Time before your application code runs: ~ 1.2 Sec.
• Number of plug-ins in the RCP binary: 23
What is inside?

<table>
<thead>
<tr>
<th>Help (optional)</th>
<th>Update (optional)</th>
<th>Text (optional)</th>
<th>IDE Text</th>
<th>Compare</th>
<th>Debug</th>
<th>Search</th>
<th>Team/CVS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IDE personality

**Generic Workbench**

- JFace
- SWT

**Resources**

- Runtime
- OSGi

[www.weiglewilczek.com](http://www.weiglewilczek.com)
Key Concepts

• Plug-in / Bundle
  The basic component in Eclipse with dependencies to other plug-ins / bundles.

• Extension Points and Extentions
  The declarative wiring between the Eclipse framework and custom code

• Feature
  A group of plug-ins with a version number

• Product
  A deployable application with branding, defined in terms of plug-ins or features
Programming Model

• Create your own plugins
• Make contributions to predefined extension points
• Extends workbench
• It is possible to define your own extension points
  • If you want others to extend your software
  • Makes it even more modular
  • Possible to establish own "plugin eco-system" inside organization
Build Rich Client GUI's with RCP & RAP

Demo (RCP)
Thin Client Technologies

- Plain traditional WEB applications
  - JSP
  - Servlets

- New wave – AJAX
  - More or less static content
  - JavaScript code runs inside browser and communicates with server
  - Browser becomes a terminal
Problem?! 

Does it mean that we will have to write client code twice? 
- in case of usual rich client technologies **YES**
Solution!

If you develop your rich client with **Eclipse RCP**

You may **reuse** most of your code for the thin client
Rich Ajax Platform (RAP)

• Is based on the same programming model as RCP
• Reuses a lot of code from RCP
• Simply uses another rendering engine
  • instead of SWT
  • it uses qooxdoo for client-side presentation
RCP Applications in a Browser

- In most cases your RCP application will run in a browser out of the box.
- Only few areas are not supported
  - i.e. custom drawing of widgets
Demo
(RAP)
More Information:

- [http://www.eclipse-training.com](http://www.eclipse-training.com)
- [http://www.eclipse.org/rcp](http://www.eclipse.org/rcp)
- [http://www.eclipse.org/rap](http://www.eclipse.org/rap)