On the shoulders of giants

Harnessing the Power of Eclipse Enterprise Ecosystem

Kirill Grouchnikov
What is he going to talk about?

To show you how Eclipse makes you more productive
Why should I care?

Less code to write, test and support
And then you can retire
What is he going to talk about?

Amdocs Smart Client Designer
Provides familiar experience
Immersed in Eclipse
Immersed in Eclipse
Immersed in Eclipse
public void textId_PostTextChange(UifEvent event) {
    t(null);
}

Multiple markers at this line
- Using unpublished class com.amdocs.uif.widgets.LabelInfra
- Using unpublished class com.amdocs.uif.widgets.ControlInfra

Problems:
- Application Code Unpublished Usage (2 items)
  - Using unpublished class com.amdocs.uif.widgets.ControlInfra
  - Using unpublished class com.amdocs.uif.widgets.LabelInfra
Immersed in Eclipse
Immersed in Eclipse
For your users

Provides familiar experience
For your developers

Less code to write, test and support
Especially for enterprise solutions
Eclipse – rich foundation
</acronyms>
<table>
<thead>
<tr>
<th>Eclipse Projects</th>
<th>Name &amp; Project Information Page</th>
<th>Ph</th>
<th>Mm</th>
<th>Nu</th>
<th>Ma</th>
<th>WU</th>
<th>IP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Business Intelligence and Reporting Tools (BIRT)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Data Tools Platform</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Connectivity</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DTP Incubator</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enablement</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model Base</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SQL Dev Tools</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Device Software Development Platform</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Device Debugging</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERCP - Embedded Rich Client Platform</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mobile Tools for Java</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Native Application Builder</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Real-Time Software Components</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tools for mobile Linux</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Eclipse Project</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>e4 Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JDT - Java development tools</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDE - Plugin Development Environment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eclipse Platform</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eclipse Project Incubator</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Galileo Simultaneous Release</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Eclipse Modeling Project</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modeling Abstraction Project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Generative Modeling Technologies (GMT)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graphical Modeling Framework</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model Development Tools (MDT)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Eclipse Technology Project**

- Equinox
- GEF3D
- Glimmer
- Higgins

**The Eclipse on Linux Project**

- Open Financial Market Platform
- Open Healthcare Framework
- Open Requirements Management Framework
- Open System Engineering Environment
- Eclipse Packaging Project

**Pluggin**

- Eclipse Process Framework Project
- Phocaos Project
- The Eclipse Spaces Project
- Subversion - SVN Team Provider
- Summer of Code

**Supplemental Widgets for SWT (Nebula)**

- SWTBot
- Tigertango

**Tool Services Framework**

- Voice Tools Project

**Tools Project**

- AJDT - AspectJ Development Tools Project
- AspectJ

**Buckminster Component Assembly**

- C/C++ Development Tools (CTI)
- C++Eclipse IDE
- GEF - Graphical Editor Framework
- Hibachi
- Mylyn

**Eclipse Ohb Project**

- Parallel Tools Platform (PTP)
- PHP Development Tools
- VE - Visual Editor

**Test and Performance Tools Platform Project**

- Monitoring Tools
- Testing Tools
- TPTP Platform
- Tracing & Profiling Tools

**Eclipse Web Tools Platform Project**

- AJAX Tools Framework
- WTP Common Tools
- Dali Java Persistence Tools
- Datasets
- WTP EJB Tools
- WTP Java EE Tools
- JavaServer Faces
- WebTools Releng
- Server Tools
- Source Editing
- Web Services Tools
- WebTools Website
- WTP Incubator

Community health - example

http://www.eclipse.org/modeling/emf/project-info/team.php

- Open Canarias
- RedHat
- Zeligsoft
- Puzzle ITC
- Geensys
- Embarcadero
- Okidoo
- IBM
- Oracle
- ES-Computersysteme
- Elver
- FernUniversitaet
- University of Brock
Our building blocks

• EMF – persistence
• JEM / VE – UI builder
• GEF – editors
• JDT – Java source trees
Putting the pixels on canvas

XML
   ↓ EMF
Java model
   ↓ JEM
Swing frame
   ↓ GEF
Canvas
EMF as the persistence layer
EMF as the persistence layer

Starting with the schema definition
EMF as the persistence layer

Generating the model classes
EMF as the persistence layer

xml -> java model
xml ↔ java
notifications
validations
JEM proxies

Java model

Offscreen Swing JFrame

Creation and updates via JEM proxies

GEF graphical edit parts

image changes
GEF - everything is an *EditPart*
All together now

XML
↓ EMF

Java model
↓ JEM

Swing frame
↓ GEF

Canvas
Java-based warnings

Java class change
  \arrow{down}
    core

Builder invocation
  \arrow{down}
    JDT

Java syntax trees
  \arrow{down}

Marker creation
  \arrow{down}
    core

UI update
Flagging API usage violations

```java
/**
 * @param event The event object.
 */

public void textId_PostTextChange(UifEvent event) {
    ((LabelInfra)labelHoles).addAlert(null);
}
```

```
org.eclipse.jdt.core.dom.ASTParser
public void createASTs(ICompilationUnit[] compilationUnits,
        String[] bindingKeys,
        ASTRequestor requestor,
        IProgressMonitor monitor)
```
All together now

Java class change
  ↓ core
Builder invocation
  ↓ JDT
Java syntax trees
  ↓ Marker creation
  ↓ core
UI update
And those are just the big pieces
Asynchronous
Background
Queuing
Cancellable
Progress
big pieces
+
small pieces
the Eclipse way
The only rule of Eclipse way
don’t fight it
provide familiar experience to your users
Our experience

• Reuse as much plumbing as possible
• Reuse as many UI parts as possible
• Use very few synchronous UI operations
• Reuse as many UI flows as possible
• Decompose into plugins
• Build for extensibility
• Don’t depend on internal code
</dudeTalking>
Q & A

http://www.eclipse.org/documentation/

kirillg@amdocs.com

www.pushing-pixels.org
Thank you