



Spying into the Heart of Eclipse

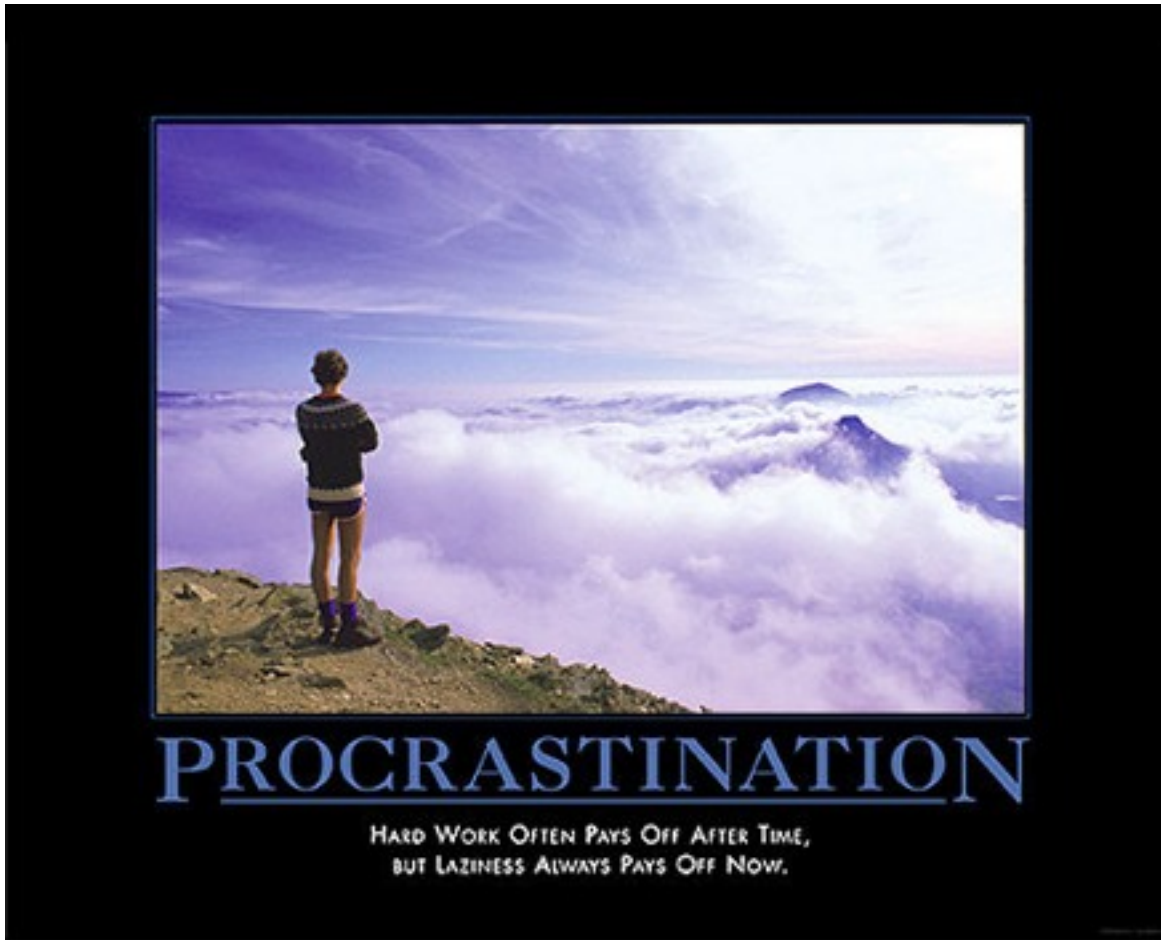
Interesting observations from an Eclipse committer

Chris Aniszczyk <zx@us.ibm.com>
PDE Lead
Eclipse Committer

October 14, 2009

Wednesday, October 14, 2009

Procrastination...



© 1999 DESPAIR, INC.

| Spying into the Heart of Eclipse | © 2008 by Chris Aniszczyk; made available under Creative Commons Att. Nc Nd 2.5 license

Agenda



-  **Background**
-  **Plug-in Spy**
-  **OSGi Console**
-  **Core Tools**
-  **Conclusion**
-  **Q&A**

Background



- This talk represents insights from a long-time Eclipse committer...
- Eclipse has a high learning curve... why?
- Common developer questions:
 - “How can I find the source for this cool feature?”
 - “How can I contribute to Eclipse? Why is this so hard?”
 - “I don’t understand what is going on inside of the Eclipse runtime...”

Agenda

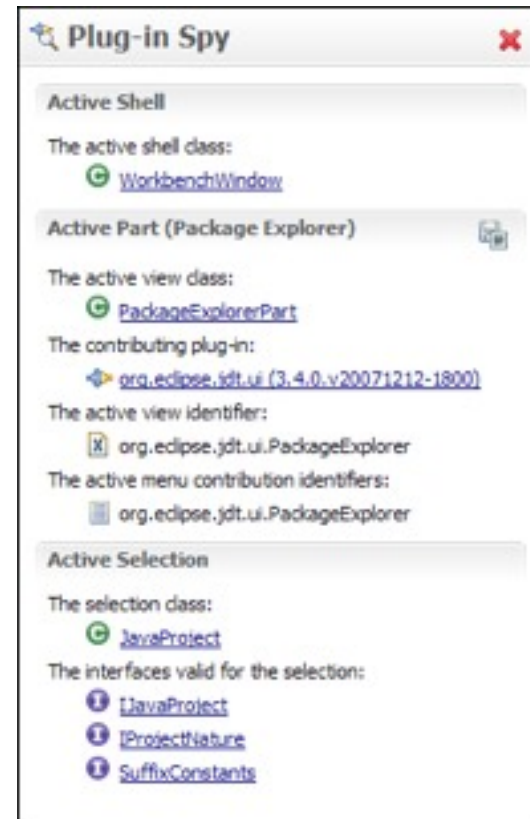


-  **Background**
-  **Plug-in Spy**
-  **OSGi Console**
-  **Core Tools**
-  **Conclusion**
-  **Q&A**

Plug-in Spy



- Coming to Eclipse 3.4!
- ALT+SHIFT+F1
- Allows you to introspect what you're looking at...
- Hyperlinking
- Shows contributing plug-ins









Demo



- **Demo 1** (“How can I find the source for this cool feature?”)
 - Error Log View
- **Demo 2** (“How can I contribute to Eclipse? Why is this so hard?”)
 - Patching Eclipse... CVS Selection Deficiency...

Agenda



-  **Background**
-  **Plug-in Spy**
-  **OSGi Console**
-  **Core Tools**
-  **Conclusion**
-  **Q&A**

Who remembers Quake?

- Useful commands like saving screenshots and cheat codes!
- iddqd or idkfa ☺



OSGi Console



- Kind of like that quake console! 😊
- Common commands
 - Status
 - Start/stop
 - Install/uninstall
 - diag
- Custom Commands

```
hello [OSGi Framework] C:\Program Files\Java\jre1.5.0_08\bin\javaw.exe (Oct 21, 2008)
osgi> install file:///C:/workspaces/test/hello2
Bundle id is 2

osgi> start 2
Hello World 2!!

osgi> ss

Framework is launched.

id      State      Bundle
0       ACTIVE    system.bundle_3.3.0.v20060919
1       ACTIVE    hello_1.0.0
2       ACTIVE    hello2_1.0.0

osgi> stop 2
Goodbye World 2!!

osgi> uninstall 2

osgi> ss

Framework is launched.

id      State      Bundle
0       ACTIVE    system.bundle_3.3.0.v20060919
1       ACTIVE    hello_1.0.0

osgi>
```

Common Commands



| Command | Description |
|--------------------------------|--|
| start | Starts a bundle given an ID or symbolic name |
| stop | Stops a bundle given an ID or symbolic name |
| install | Adds a bundle given a URL for the current instance |
| uninstall | Removes a bundle given a URL for the current instance |
| update | Updates a bundle given a URL for the current instance |
| active | Lists all active bundles in the current instance |
| headers | List the headers for a bundle given an ID or symbolic name |
| ss | Lists a short status of all the bundles registered in the current instance |
| services <filter> | Lists services given the proper filter |
| diag | Runs diagnostics on a bundle given an ID or symbolic name |







Demo



- **Demo 1**
 - Common commands
- **Demo 2**
 - uname
 - Custom commands*
- *<http://www-128.ibm.com/developerworks/opensource/library/os-ecl-osgiconsole/>*

Agenda



-  **Background**
-  **Plug-in Spy**
-  **OSGi Console**
-  **Core Tools**
-  **Conclusion**
-  **Q&A**

Core Tools



- Basic tools to help understand Eclipse runtime
- http://wiki.eclipse.org/Core_Tools

The screenshot shows two windows from the Eclipse IDE. The top window, titled 'Activated Plugins', displays a table of installed plugins with columns for Plugin, Classes, Alloc, Used, Startup time, and Order. The bottom window, titled 'Loaded Classes', displays a table of loaded classes with columns for Class, Order, Memory, and Plug-in.

| Plug-in | Classes | Alloc | Used | Startup time | Order |
|----------------------------------|---------|-------|------|--------------|-------|
| org.eclipse.ui.net | 2 | 0 | 0 | 16 | 35 |
| org.eclipse.core.resources* | 233 | 0 | 0 | 906 | 14 |
| org.eclipse.core.runtime* | 21 | 0 | 0 | 2359 | 4 |
| org.eclipse.core.tools | 33 | 0 | 0 | 16 | 46 |
| org.eclipse.equinox.common* | 50 | 0 | 0 | 94 | 2 |
| org.eclipse.help* | 4 | 0 | 0 | 16 | 20 |
| org.eclipse.jdt.apt.core | 9 | 0 | 0 | 62 | 45 |
| org.eclipse.jdt.ui* | 433 | 0 | 0 | 969 | 24 |
| org.eclipse.jdt.core.refactoring | 27 | 0 | 0 | 62 | 47 |
| org.eclipse.jdt.ui.refactoring | 3 | 0 | 0 | 16 | 48 |
| org.eclipse.osgi* | 0 | 0 | 0 | 0 | 1 |
| org.eclipse.pde.core | 237 | 0 | 0 | 172 | 39 |
| org.eclipse.pde.ui* | 65 | 0 | 0 | 94 | 31 |
| org.eclipse.debug.core* | 66 | 0 | 0 | 46 | 32 |
| org.eclipse.team.core* | 61 | 0 | 0 | 47 | 30 |
| org.eclipse.team.cvs.core | 114 | 0 | 0 | 360 | 37 |
| org.eclipse.team.cvs.ui | 27 | 0 | 0 | 297 | 43 |
| org.eclipse.team.ui* | 18 | 0 | 0 | 15 | 33 |
| org.eclipse.ui.ide* | 127 | 0 | 0 | 16 | 17 |
| org.eclipse.ui.views* | 5 | 0 | 0 | 0 | 23 |
| Total: 1 | 65 | 0 | 0 | 94 | |

| Class | Order | Memory | Plug-in |
|---|-------|--------|--------------------|
| org.eclipse.pde.internal.ui.PDEConstants* | 1 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.PDEPlugin* | 2 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.launcher.LaunchConfigurationListener* | 3 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.launcher.PDELogFileProvider* | 4 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.views.imports.ExternalPluginLibrariesFilter* | 5 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.views.plugins.PluginsView | 6 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.views.plugins.PluginsView\$I3v4Filter | 7 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.views.plugins.PluginsView\$I3v4SpaceFilter | 8 | 0 | org.eclipse.pde.ui |
| org.eclipse.pde.internal.ui.views.plugins.PluginsView\$I3v4DisabledFilter | 9 | 0 | org.eclipse.pde.ui |

Demo

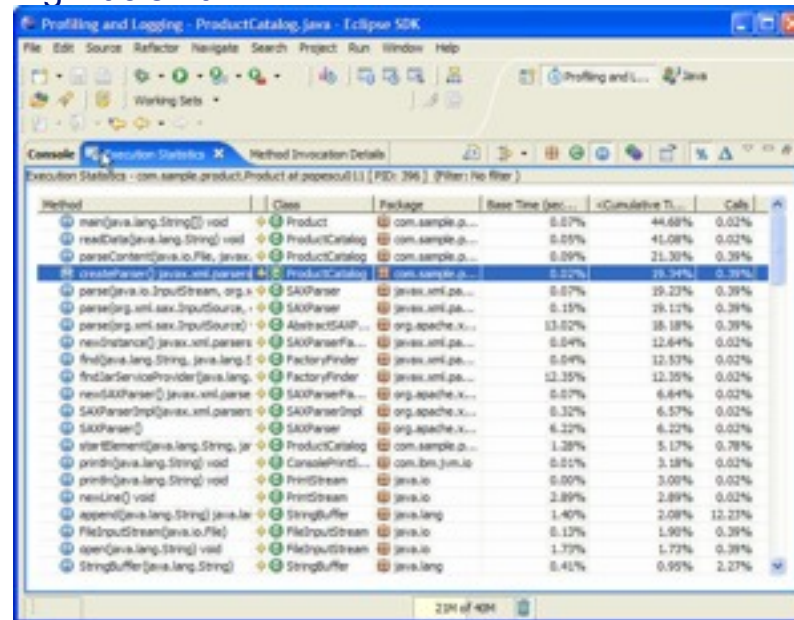


- **Demo 1**
 - Runtime Spy Perspective (PDE Plug-ins View Example)
 - “I don’t understand what is going on inside of the Eclipse runtime...”

- **Demo 2**
 - Metadata Spy Perspective

- **Demo 3**
 - Preferences Spy

- Note: There exists more advanced performance tooling in Eclipse...
- Test and Performance Tools Platform (TPTP)
 - <http://www.eclipse.org/tptp>
 - <http://www.eclipse.org/articles/Article-TPTP-Profiling-Tool/tptpProfilingArticle.html>



Profiling and Logging - ProductCatalog.java - Eclipse SDK

Execution Statistics - com.sample.product.Product at [product1] [PID: 296] [Filter: No filter]

| Method | Class | Package | Base Time (sec...) | %Cumulative Ti... | Calls |
|--------------------------------------|------------------|-----------------|--------------------|-------------------|--------|
| main(java.lang.String[]) void | Product | com.sample.p... | 0.07% | 44.68% | 0.02% |
| readData(java.lang.String) void | ProductCatalog | com.sample.p... | 0.05% | 41.08% | 0.02% |
| parseContent(java.io.File, java.x... | ProductCatalog | com.sample.p... | 0.09% | 21.30% | 0.39% |
| createItems() java.xml.parsers... | ProductCatalog | com.sample.p... | 0.02% | 16.34% | 0.39% |
| parse(java.io.InputStream, org.x... | SAXParser | java.xml.pa... | 0.07% | 19.22% | 0.39% |
| parse(org.xml.sax.InputSource, ...) | SAXParser | java.xml.pa... | 0.15% | 19.12% | 0.39% |
| parse(org.xml.sax.InputSource) | AbstractSAXP... | org.apache.x... | 13.02% | 38.18% | 0.39% |
| newInstance() java.xml.parsers... | SAXParserFa... | java.xml.pa... | 0.04% | 12.44% | 0.02% |
| find(java.lang.String, java.lang... | FactoryFinder | java.xml.pa... | 0.04% | 12.53% | 0.02% |
| findServiceProviders(java.lang... | FactoryFinder | java.xml.pa... | 12.35% | 12.35% | 0.02% |
| newInstance() java.xml.parsers... | SAXParserFa... | org.apache.x... | 0.07% | 6.44% | 0.02% |
| SAXParserImpl() java.xml.parsers... | SAXParserImpl | org.apache.x... | 0.32% | 6.57% | 0.02% |
| SAXParser() | SAXParser | org.apache.x... | 6.32% | 6.22% | 0.02% |
| visitElement(java.lang.String, ja... | ProductCatalog | com.sample.p... | 1.28% | 5.17% | 0.78% |
| println(java.lang.String) void | ConsolePrint... | com.ibm.jvm.io | 0.01% | 3.18% | 0.02% |
| println(java.lang.String) void | PrintStream | java.io | 0.00% | 3.00% | 0.02% |
| newLine() void | PrintStream | java.io | 2.89% | 2.89% | 0.02% |
| append(java.lang.String) java.la... | StringBuffer | java.lang | 1.40% | 2.08% | 12.22% |
| FileOutputStream(java.io.File) | FileOutputStream | java.io | 0.12% | 1.90% | 0.39% |
| open(java.lang.String) void | FileOutputStream | java.io | 1.73% | 1.73% | 0.39% |
| StringBuffer(java.lang.String) | StringBuffer | java.lang | 0.41% | 0.95% | 2.27% |

Agenda



-  **Background**
-  **Plug-in Spy**
-  **OSGi Console**
-  **Core Tools**
-  **Conclusion**
-  **Q&A**







Conclusion



- There's a lot to learn from the runtime...
- Understanding internals can help with debugging problems like...
 - Performance
 - Startup Issues
 - Classloading Exceptions
- Spying into Eclipse is fun and easy!
- Now that you know more... think about contributing back!

Agenda



-  **Background**
-  **Plug-in Spy**
-  **OSGi Console**
-  **Core Tools**
-  **Conclusion**
-  **Q&A**



Thank you!

Backup Slides



- In case audience is completely confused OSGi...

What is OSGi?



- OSGi is an independent, non-profit corporation working to define open specifications for the delivery of services to networked small-memory devices.
- OSGi Service Platform, Release 4.1 is the latest version of the specification document for a service delivery framework.
- The OSGi Service Platform specifications describe a software deployment and configuration management architecture.
- On the web at <http://www.osgi.org/>.



OSGi Goals



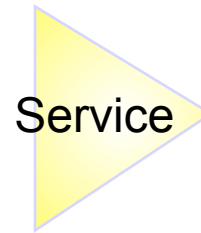
- Provide a consistent, component-based application programming model.
- De-couple a component's interface from its implementation.
- Support multiple implementations of a component's interface.
- Development of components independent of the implementation of other components.



What is a OSGi component?

- An OSGi component is an implementor of a Java interface.
- OSGi refers to the Java interfaces implemented by a component as services or service interfaces.
- Applications are built from collaborating components.
- Collaborating components are coupled only via service interfaces.

What is an OSGi Service?



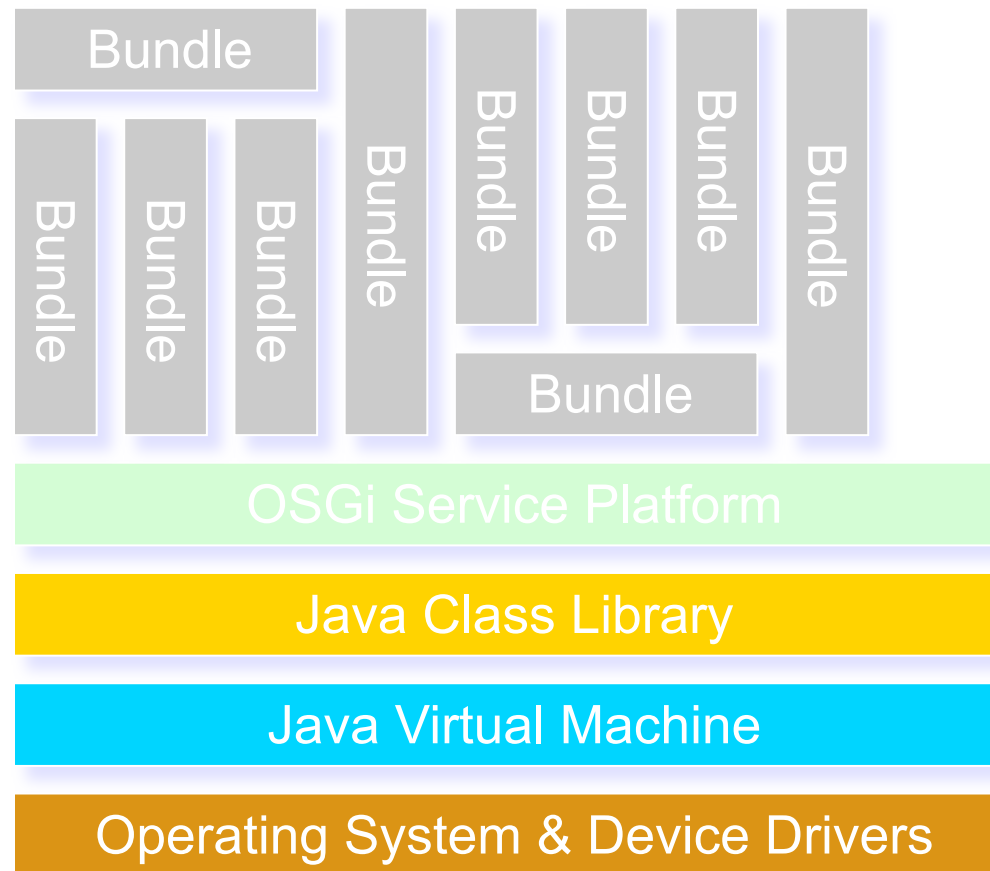
- A service is a self-contained component accessible via a service interface.
- Services are registered with the OSGi framework, making them available to others. This is sometimes called publishing a service.
- Components can acquire services by querying the OSGi framework.
- Components must release services back to the OSGi framework when they are finished with them.

Some OSGi Services...



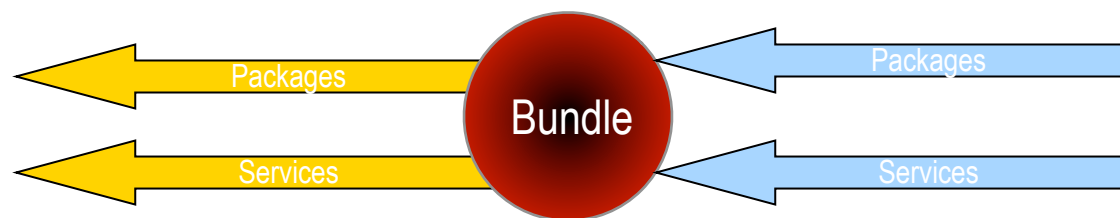
- **Configuration Admin Service** - enables an operator to set the configuration information of deployed bundles.
- **Device Access** - supports automatic detection of attached and detached hardware devices and can automatically download and start appropriate device drivers.
- **HTTP Service** - provides an embedded HTTP server that is capable of serving HTML and servlets.
- **Log Service** - provides a general purpose message logger for the OSGi environment.
- **Package Admin Service** - enables a management bundle to provide the policies for package sharing.
- **Permission Admin Service** - enables a management bundle to administer bundle's permissions and provides defaults for all bundles.
- **Preferences Service** - provides a persistent data store for bundles.
- **User Admin Service** - provides minimum authentication functionality.
- **Start Level Service** - enables a management agent to control the relative starting and stopping order of bundles.
- **URL Handlers Service Support** - enables you to dynamically register multiple URL and Content handler services, which shield bundles from Java limitations.

OSGi Software Architecture



What is a bundle?

- A bundle is often a JAR file.
- A bundle can also be a JXE file.
- A bundle can contain services.
- A bundle can contain Java packages.
- A bundle can export services and Java packages.
- A bundle can import services and Java packages.
- A bundle can be activated by a BundleActivator.
- A bundle always contains an OSGi-specific manifest file
 - MANIFEST.MF



The OSGi Bundle Manifest



- Each bundle must contain a manifest file.
- The bundle's manifest file contains data that the framework needs to correctly install and activate the bundle.
- The headers used in a manifest file must be defined by the OSGi™ specification.
- The MANIFEST.MF is located in the META-INF directory of your bundle project.

OSGi Manifest Headers



- **Bundle-Name**
Provides a human readable name for a bundle.
- **Bundle-SymbolicName**
Identifies a bundle. The Bundle Symbolic Name and Bundle Version allow for a bundle to be uniquely identified in the Framework.
- **Bundle-ClassPath**
A comma separated list of JAR file path names (inside the bundle) that should be searched for classes and resources.
- **Bundle-Activator**
The name of the class that is used to start and stop the bundle.
- **Bundle-Localization**
Specifies where the bundle is located.
- **Require-Bundle**
Specifies the specific bundles that provides packages you use in your bundle. If you do not specify the bundle which provides the packages you need, you may get a noClassDefFound exception when the bundle loads.
- **Import-Package**
Specifies the names of any package that you want your bundle to import from the runtime. If you do not specify the package your bundle needs in this header, you may get a noClassDefFound exception when the bundle loads.
- **Export-Package**
Specifies the name of any package that you want your bundle to export to the runtime. If you do not specify the packages needed by other bundles in this header, the dependent bundles may not resolve.

Eclipse and OSGi: A happy marriage!



- Eclipse is a universal platform for integrating components (plug-ins)
- Open, extensible architecture
- Applications are delivered as **plug-ins**
 - Plug-in - smallest unit of Eclipse function
 - Extension point - named entity for collecting “contributions”
 - Extension - a contribution
- **OSGi Fits!**
 - **Plug-in ↔ Bundle!**

