What is Eclipse?

Eclipse Helios
Eclipse is a Java IDE

- Language-aware editors, views, ...
- Refactoring support
- Integrated unit testing and debugging
- Incremental compilation and build
- Team development support
Eclipse is an IDE Framework

- Eclipse + JDT = Java IDE
  - First class framework for Java, language aware editor, incremental build, integrated debugging, ...

- Eclipse + CDT = C/C++ IDE
  - First class framework for C/C++, language aware editor, refactoring, search

- Eclipse + PDT = PHP IDE

- Eclipse + JDT + CDT + PDT = Java, C/C++, PHP IDE
  - Ruby, TCL, JavaScript, ...
Eclipse is a Tools Framework

- Plug-ins make Eclipse whatever you need it to be
- Platform of frameworks and exemplary tools
- Tools extend the platform using bundles/plug-ins
  - Business Intelligence and Reporting Tools, Web Tools, Data Tools, Eclipse Modeling Framework, ...
Eclipse is a Application Framework

- Remove the IDE elements; you’re left with a general-purpose application framework
  - Linux, Windows, Mac OSX, UNIX, embedded
  - Rich widget set, graphics
  - Native-OS integration (drag and drop, OLE/XPCOM integration)
- A platform for rich clients
Eclipse is Runtimes!

- Remove the UI elements and you're left with a general-purpose component model
  - Equinox, an implementation of the OSGi R4 core framework specification
  - Headless applications, servers, embedded, ...
- Dozens of runtime technologies at eclipse.org
  - Eclipse Communication Framework, EclipseLink, eRCP, Gemini, Gyrex, Higgins, Jetty, Rich Ajax Framework, Riena, SeMantic Information Logistics Architecture, Swordfish, Virgo, ...
Top Level Projects

- Business Intelligence and Reporting Tools (BIRT)
- Data Tools Platform
- Device Software Development Platform
- Eclipse
- Modeling

- RT
- SOA
- Technology
- Tools
- Web Tools Platform
- Test & Performance Tools Platform
Eclipse is a Community
Eclipse is an Ecosystem

- Eclipse is focused on nurturing the ecosystem to complement, and enhance the Eclipse Platform
  - Hundreds of organizations build Eclipse plug-ins
  - Companies collaborate on Eclipse projects
  - Companies compete with products based on Eclipse technology
- Many members, including major Java, Enterprise, Linux, and Embedded vendors
  - http://www.eclipse.org/membership/
Eclipse is a Foundation

- The Eclipse Foundation
  - Manages and directs the ongoing development
  - Provides IT infrastructure required by development teams
- Independent not-for-profit Foundation formed in 2004
- Does not directly employ developers for Eclipse projects
finally, geesh...
Why a Simultaneous Release?

- Help spur commercial adoption of Eclipse technology
- Consumers use many projects not just the Platform
- Inter-dependency between projects
  - Eclipse project teams are independent, but the project code is inter-dependent
- Alignment of version compatibility
- Remove latency between project releases
Before the Simultaneous Release

*Hypothetical and ridiculously simplified
What is Helios?

- Simultaneous release of 39 Eclipse projects
- A transparent and predictable development cycle
  - Allow ecosystem members to start their own integration, cross-project, and cross-product testing efforts earlier
  - Developers helping developers serve the whole Eclipse community
Not Project Unification

- Each project remains a separate open source project
- Projects remain independent
  - Own project leadership
  - Own committers
  - Own project plan
- Projects coordinate their plans and schedules with other participating projects
Seven Years in a Row

- Eclipse 3.0: June 28, 2004
  - TPTP
  - CDT

- Eclipse 3.1: June 28, 2005
  - WTP
  - BIRT
  - EMF
  - VE
  - CDT

- Callisto: June 30, 2006
  - 10 Projects

- Europa: June 29, 2007
  - 21 Projects
  - 17 million LOC

- Ganymede: June 25, 2008
  - 23 Projects
  - 18 million LOC

- Galileo: June 24, 2009
  - 33 Projects
  - 24 million LOC

- Helios: June 23, 2010
  - 39 Projects
  - 33 million LOC
More Pragmatically... (1/2)

- Coordination
  - Projects have a better sense of downstream use and requirements
  - Staged, predictable release schedule
  - Frequent milestone releases

- Collaboration
  - Project teams work together
  - Reduced functional redundancy

<table>
<thead>
<tr>
<th></th>
<th>Staging</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>Release</th>
<th>Public Access</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+0</td>
<td>+1</td>
<td>+2</td>
<td>+3</td>
<td>EPP</td>
<td></td>
<td>Server Freeze</td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>Sep 21</td>
<td>Sep 26</td>
<td>Oct 3</td>
<td>n/a</td>
<td>Oct 5</td>
<td>n/a</td>
<td>Oct 9</td>
<td></td>
</tr>
<tr>
<td>M2+</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>n/a</td>
<td>Oct 9</td>
</tr>
<tr>
<td>M3</td>
<td>Nov 2</td>
<td>Nov 7</td>
<td>Nov 14</td>
<td>Nov 16</td>
<td>Nov 19</td>
<td>?</td>
<td>Nov 20</td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>Dec 14</td>
<td>Dec 17</td>
<td>Jan 7</td>
<td>Jan 8</td>
<td>Jan 9</td>
<td>Jan 10</td>
<td>Jan 11-Jan 16</td>
<td></td>
</tr>
<tr>
<td>M5</td>
<td>Feb 8</td>
<td>Feb 11</td>
<td>Feb 18</td>
<td>Feb 20</td>
<td>Feb 22</td>
<td>Feb 25</td>
<td>Feb 26</td>
<td></td>
</tr>
<tr>
<td>M6 - API Freeze</td>
<td>Mar 28</td>
<td>Mar 31</td>
<td>Apr 7</td>
<td>Apr 9</td>
<td>Apr 11</td>
<td>Apr 14</td>
<td>Apr 15*</td>
<td></td>
</tr>
<tr>
<td>M7</td>
<td>May 2</td>
<td>May 5</td>
<td>May 6</td>
<td>May 7</td>
<td>May 9</td>
<td>May 12</td>
<td>May 13*</td>
<td></td>
</tr>
<tr>
<td>RC1</td>
<td>May 16</td>
<td>May 19</td>
<td>May 20</td>
<td>May 21</td>
<td>May 23</td>
<td>May 26</td>
<td>May 27*</td>
<td></td>
</tr>
<tr>
<td>RC2</td>
<td>May 23</td>
<td>May 26</td>
<td>May 27</td>
<td>May 28</td>
<td>May 30</td>
<td>Jun 2</td>
<td>Jun 3*</td>
<td></td>
</tr>
<tr>
<td>RC3</td>
<td>May 30</td>
<td>Jun 2</td>
<td>Jun 3</td>
<td>Jun 4</td>
<td>Jun 6</td>
<td>Jun 9</td>
<td>Jun 10*</td>
<td></td>
</tr>
<tr>
<td>RC4</td>
<td>Jun 6</td>
<td>Jun 9</td>
<td>Jun 10</td>
<td>Jun 11</td>
<td>Jun 13</td>
<td>Jun 16</td>
<td>Jun 17*</td>
<td></td>
</tr>
<tr>
<td>GJymede</td>
<td>Jun 13</td>
<td>Jun 16</td>
<td>Jun 17</td>
<td>June 18*</td>
<td>June 23</td>
<td>June 24</td>
<td>June 25</td>
<td></td>
</tr>
</tbody>
</table>
More Pragmatically... (2/2)

- Build Aggregation
  - All components collected together
  - Easily installed via a single repository

- Communication
  - All this goodness comes from getting the project teams to talk to each other
**Official Helios Stats**

- 39 project teams
- 33+ million LOC
- 44 companies providing committers
- 490 committers
  - 382 associated with a company
  - 108 individual committer
39 Projects, You Say...

- 39 declared participation
- 54 sets of documentation were sent to the EMO for the release review;
- 58 projects submitted a short description of their Helios contributions;
- 62 IP Logs were submitted; and
- 71 Eclipse “Projects” participated in release.
Confused?

- You probably should be...
- Definition of “project” at Eclipse is pretty flexible
  - We'll tighten this up next year...

![Diagram of project hierarchy]

- Top Level
- Project
- Sub Project

Level 1: “Top Level Project”
Level 2: “Project”
Level 3: “Subproject”
What's the Version Number?

- 'Helios' is the version name
  - eclipse-<package>-<version>-<platform>
  - e.g. eclipse-java-helios-linux-gtk
- 'Helios' will be followed by the 'Helios Fall' and 'Helios Winter' releases
- Why no number?
  - What would you call Eclipse SDK 3.6 + Web Tools 3.2 + EMF 2.5 + CDT 6.0 + ... ?
Participating Projects Must... (1/4)

- Declare participation by M4
- Complete project plans by M4
- Subscribe to cross-project mailing list
- Document use of internal APIs
- Support Translations
  - All strings externalized
  - Participate in Babel
  - Use Eclipse message bundles
- Use ICU4J, where appropriate
Participating Projects Must... (2/4)

- 4-part version numbers
- OSGi bundle format
- List execution environment
- Sign bundles
- Use JARred bundles
- Re-use and share common third party jars
- Provide optimized p2 repositories
Participating Projects Must... (3/4)

- Provide basic capability/activity definitions
- Branding
  - 'About' dialog icon with hover text that displays the provider name
  - Descriptive provider-name (for features)
  - Bundle-Vendor header (for plug-ins)
  - Contribute to the welcome page
- Use standard forms of license documents
- Work together with other projects in any combination of any install
Participating Projects Must... (4/4)

- Have a mature, stable build process
- Have some unit tests that can verify at least basic functionality of a build or distribution
- Have a written ramp down policy by M6
  - Describe when the project plans to be feature complete, have API frozen, ...
- API Policy Defined and Documented.
  - Includes how 'APIs' are distinguished from non-API and 'provisional' API, if any
Participating Projects Should... (1/2)

- Engage Community
- Follow the User Interface Guidelines
  - Participate in a User Interface Best Practices Working Group UI walkthrough
- Test Localization using the Babel Pseudo Translation Test
- Design and test for enabling all languages
  - Bidi, unicode characters, etc.
- Design and test for accessibility compliance
Participating Projects Should... (2/2)

- Define and document their retention policy
- Have measurable performance criteria
  - Devote at least one milestone to performance and scalability improvements
- Should provide some summary metrics
  - Number of bundles, number of committers, lines of code, number of bugs opened and fixed
Helios Projects (1/7)

- Acceleo
- Accessibility Tools Framework (ACTF)
- ATL – Atlas Transformation Language
- BPMN Modeler
- Buckminster
- Business Intelligence & Reporting Tools (BIRT)
Helios Projects (2/7)

- C/C++ Development Tooling (CDT)
- CDO Model Repository
- Dali Java Persistence Tools
- Data Tools Platform (DTP)
- Dynamic Languages Toolkit (DLTK)
- Eclipse Communication Framework (ECF) Eclipse Git Team Provider (eGit)
- Eclipse Modeling Framework (EMF)
- Eclipse Packaging Project (EPP)
Helios Projects (3/7)

- Eclipse Platform
- Eclipse Web Tools Platform Project (WTP)
- EclipseLink Project
- EMF Compare
- EMF Teneo Model Relational Mapping
- Equinox
- Extended Editing Framework (EEF)
- Fortran Development Tools (Photran)
- GEF – Graphical Editor Framework
Helios Projects (4/7)

- Graphical Modeling Framework (GMF)
- Java Emitter Templates (JET2)
- Java implementation of Git (jGit)
- Java Workflow Tooling (JWT)
- Java Script Development Tools (JSDT)
- JavaServer Faces (JSF)
- JDT – Java Development Tools
- Linux Tools
- Marketplace Client (MPC)
Helios Projects (5/7)

- MDT: OCL, UML2, XSD
- Memory Analyzer (MAT)
- Mint
- Mobile Tools for Java (DSDP MTJ)
- Model Discovery (MoDisco)
- Model to Text (M2T): M2T, Xpand
- Modeling Amalgamation
- Modeling Team Framework
- Modeling EMFT Ecore Tools MWE
Helios Projects (6/7)

- Mylyn
- Net4J Signalling
- Platform Parallel Tools Platform (PTP)
- PHP Development Tools (PDT)
- Rich Ajax Platform (RAP)
- Jetty
- SCA Tools
- Sequoyah
- SOA Tools (STP)
Helios Projects (7/7)

- Subversive
- Swordfish
- Target Management (DSDP TM)
- Test and Performance Tools Platform Project (TPTP)
- Web Tools
- Xtext
New Projects to the Release Train

- eGIT/ jGIT
- EEF
- Photran
- JSDT
- Linux Tools
- Marketplace Client (MPC)
- MoDisco
- Modeling Amalgamation
- Modeling Team Framework
- PTP
- Jetty
Helios Highlights

Eclipse Helios
New and Noteworthy: Platform

- New platforms: PowerPC 64-bit, Windows 7, and Ubuntu 10.04
- Opening a file from the command line
- Linux users can now use WebKitGTK+ instead of Mozilla as the Browser control's native renderer
- UNIX file permissions
- Progress shown in platform task bar
New and Noteworthy: PDE

- OSGi console interacts with the running framework
- Feature based launch
- Support software installation while running and debugging
- Import from CVS
  - Import projects from CVS corresponding to bundles in the target platform
Marketplace Client

- Find open source and commercial Eclipse-related offerings
- Searchable
- Multiple Markets
- ~1 click installation, update, and removal
Git Team Provider

- Git is a distributed SCM
  - Every developer has a full copy of all history of every revision of the code
- JGit is an EDL (BSD) licensed, lightweight, pure Java library implementing Git
- EGit is an Eclipse Team provider for Git
- (aside) Git repositories for projects at eclipse.org
Linux Tools

- Autotools, Valgrind, Oprofile, Gprof, GCov integration
- C/C++ call graph utility
- RPM .spec editor with rpmlint integration
Extended Editing Framework (EEF)

- Helps users create editors, views and wizards for EMF models
- Runtime and a development environment to generate advanced edition UI for EMF model
JavaScript Development Tools (1/2)

- Improved namespace support
- The JavaScript Debug Interface (JSDI)
  - An API that consumers can implement to hook into the use of the debug core and UI bundle
- Rhino Debug Support, Console
- Step Into Script Load
JavaScript Development Tools (2/2)

- Breakpoint Participants
- Line, Function, Script Load Breakpoints
- Breakpoint Detail Panes, Properties
- Suspend when any script loads
- Universal Launching Support
- Watch Expressions
Rich Ajax Platform (1/2)

- RAP implements a subset of the SWT API
  - Target RCP and RAP from a single code base
  - Extended in Helios to cover close to 100% of SWT
- Improved and customizable Look and Feel
  - Rounded borders, gradients, and animations
- EMF provides RAP support for Helios
- Riena's ridgets run on RAP
- Single sourced version of the Memory Analyzer
Rich Ajax Platform (2/2)
**BIRT**

- Polar/Radar chart, pie chart rotation, palette hashing, sort locale and strength
- Bidirectional enhancements (e.g. enhanced BIDI support in BIRT emitters)
- Performance improvements
  - Improved PDF image inclusion performance
  - Improved performance of large datasets
- Usability improvements
  - Multiple resource files, accessibility improvements
Getting Help,
Getting Involved

Eclipse Helios
Getting Involved

- Test new releases
- Create bug reports, feature requests, patches
  - https://bugs.eclipse.org/bugs/
- Ask and answer questions in the forums
  - http://www.eclipse.org/forums
- Blog/Podcast/... about Eclipse
Forums and Mailing Lists

- http://www.eclipse.org/mail/
  - Communication among project committers
- http://www.eclipse.org/forums/
  - news://news.eclipse.org
  - Communication between and among developers and users
- When in doubt, start with the forums
Forums

- Bugzilla ID and password required
- Each project has at least one newsgroup
- If you're not sure, post to eclipse.newcomer
  - Somebody will point you in the right direction
- Well thought-out questions get answered
  - Provide enough information, do your homework
  - Be polite
- Give back: ask and answer questions
Mailing Lists

- Used for communication amongst the project committers
  - Committers are the people who can actually commit code into Eclipse repositories
- Everybody can sign up for a mailing list
  - Feel free to “listen in on the conversation” and contribute
- “How to use” questions should be directed to the project forum
IRC

- `[alias]@irc.freenode.net/#eclipse`
- IRC via Eclipse Communication Framework
  - `http://www.eclipse.org/ecf`
**Summary**

The best way to get involved with the Eclipse community is to get involved with the Eclipse community.