



By Curt Schwaderer

CompactPCI &
AdvancedTCA

Eclipse Foundation extends open source to software development tools

Operating systems have taken a miraculous journey over the past 30 years, from custom, internally developed kernels through commercial embedded real-time operating systems to the open source Linux world. The open source paradigm of Linux has found its way into the software development tools arena with the emergence of the Eclipse Foundation Eclipse development environment. This tool framework is written entirely in Java with the ability to plug in any number of application-specific development environments. This paradigm is also gaining significant momentum. The user and developer membership consists of more than 80 companies with far more in the general user community. A recent announcement also introduced a Spotlight on Eclipse supplement jointly sponsored by SD Times and the Eclipse Foundation. With all this activity and momentum around this open source effort, it could be well worth your time to take a look at the organization and technology.

Impressive growth of Eclipse

Eclipse initially went open source in November, 2001. Since that time, more than 600 commercially available plugins have been developed. If you browse SourceForge, you will notice more than 400 projects currently using Eclipse in one way or another. Over the past year alone, a staggering 30 million downloads of the Eclipse framework have taken place on the Eclipse Foundation servers. This number does not include the other 40 download sites around the world.

You might expect significant Linux involvement in Eclipse. You would be right. Linux providers such as Red Hat, SuSE, and TimeSys ship Eclipse C/C++ and Java development tools with their products as an alternative to the standard Linux GCC development environment.

What is Eclipse?

Eclipse is a Java based universal tools framework that has the goal of becoming the foundation for the tool building community. If the goal is realized, Eclipse will

become the industry standard framework within which all tools will be built to integrate and interoperate with each other.

Currently, the Eclipse Foundation has six announced projects defined within the Eclipse development community. These projects are roughly categorized into:

- Core technology advancement of the Eclipse framework
- Technology research
- Web platforms
- Test and performance
- Business intelligence and reporting facilitation through the Eclipse tools

A Project Management Committee oversees the requirements, design, and implementation of each project. The Committee then organizes each project into the project lead, development team, and project management function.

Once the project proposal is accepted, the development life cycle goes into iterative validation and implementation phases with checkpoint reviews coming after validation and release reviews after implementation.

Who is the Eclipse Foundation?

The Eclipse Foundation founding members consist of Borland, IBM, MERANT, QNX Software Systems, Rational Software, Red Hat, SuSE, TogetherSoft, and WebGain. Since its original formation, the Eclipse Foundation has grown to more than 80 member companies.

What is the mission of the Eclipse Foundation?

In their own words, "The Eclipse Foundation is a nonprofit corporation formed to advance the creation, evolution, promotion, and support of the Eclipse Platform and to cultivate both an open source community and an ecosystem of complementary products, capabilities, and services."

Eclipse became a not-for-profit organization in January, 2004. Mike Milinkovich, executive director of the Eclipse

Foundation, came aboard in May, 2004. Mike and his team focus on growing the Eclipse open source community and membership. The member-funded organization supports a staff of seven. The primary functions of the Eclipse team include:

- Supporting Eclipse projects such as handling paperwork, documentation, and IT infrastructure to support development
- Maintain Eclipse website and downloads
- Increase awareness of the Eclipse environment and its uses in various projects
- Value-add services via website
- Intellectual property due diligence

You can imagine the IT challenges involved with such a geographically distributed development team. Mike mentioned substantial work in this area to facilitate Eclipse development projects.

Funding from the Eclipse Foundation comes from *strategic user* and *strategic developer* memberships. Strategic users pay an annual fee. Strategic developers also pay an annual fee along with providing a minimum of eight developers and a commitment to lead at least one Eclipse project.

What can Eclipse do?

Eclipse provides a common foundation for creation of a robust, integrated development environment including the building, debugging, and software configuration management for software development projects. Eclipse is implemented in Java, therefore any platform with the proper Java Runtime Environment (JRE) is able to host the Eclipse development environment.

Mike also cited interesting future additions to the Eclipse framework. An Eclipse communications framework is being developed. This is a peer-to-peer communications framework to make distributed applications development easier within the Eclipse environment. He also

mentioned an *Eclipse trust framework* being led by Social Physics and *parallel development tools* project being led by Greg Watson at Los Alamos National Labs.

A brief look at Eclipse

The high-level Eclipse development architecture is shown in Figure 1. Eclipse exposes the development environment to the user through a workbench GUI environment. The workbench allows the developer to create and manage multiple projects. One or many projects defined in the workbench can be accessed and worked on simultaneously. The workbench includes a Software Widget Toolkit (SWT), which provides the tools for building graphical user interfaces and a JFace component. The JFace component facilitates the development of user interfaces and works with the SWT to allow the plug-in tool to easily provide a familiar GUI for the developer using the tool.

Users create a workspace consisting of one or many projects. The workspace is linked to a particular plug-in for the Eclipse environment, which then provides facilities such as software development, build, and debugging for the workspace. For example, one of the plug-ins available for Eclipse is a Java development environment. The specific workspace for a particular plug-in is called a perspective. The Eclipse environment allows the developer to select which perspective a particular workspace belongs to. There are perspectives for creating Java applications and C/C++ applications among others.

The Eclipse architecture provides a help framework that allows the plug-in tools developer to provide help menus for various features or tasks the tool implements.

The platform runtime consists of a JRE and associated operating system running on a hardware platform like a personal computer. Finally, the Team component provides hooks for integration with a variety of software configuration management tools such as CVS.

The plug-in tools conform to the Eclipse plug-in tools interface. Plug-in developers then write an application specific plug-in

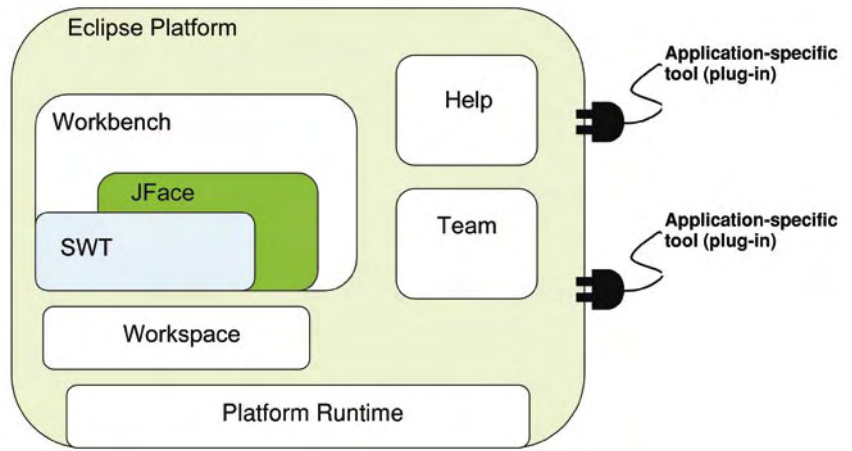


Figure 1

framework and user interface while the plug-in provides the actual execution of developing, building, and debugging for a specific application.

Eclipse: Not just for Java anymore

There is a common misconception that Eclipse is strictly a development tool for Java. While this capability is an available part of the environment, and Eclipse is implemented in Java, a large and far ranging group of projects use Eclipse.

Eclipse is strongly engaged in the embedded marketplace. PalmOS and Nokia use Eclipse integrated development environments as the open application development environment for their PDA and cell phone products. QNX, TimeSys, and Wind River all ship C/C++ development plug-ins for Eclipse. The Rational product line is built on Eclipse. Borland's Together Control Center and modeling tools, SAP Netweaver Studio, and IBM's Web Server Business products are also based on Eclipse.

One unique testimony to the flexibility of the Eclipse framework comes from a company called IP Fabrics. IP Fabrics has used the Eclipse framework as the development tool environment for their packet processing language runtime software environment for highly parallel network processing environments.

The myriad of available products and plug-ins based on Eclipse prove that Eclipse can be successfully incorporated

ocracy, those who show interest and aptitude are rewarded with contributor status. There are no formal channels for contributing. You can begin by downloading the environment. Get involved with news groups and try things out with the Eclipse environment. If you are really ambitious, create a plug-in. If you notice problems, create bug reports and suggest fixes. Mike describes a well-formed and detailed bug report as a *small nugget of gold*. So, if you are interested in open source software and would like to get involved, the Eclipse Foundation is pleased to accommodate.

Summary

Eclipse is a large and growing community dedicated to the development of a universal tools framework upon which any *development environment* can be built. The term development environment is not just limited to C/C++ and Java projects.

Additional information on the Eclipse Foundation and Eclipse projects can be found at www.eclipse.org. This website also provides downloads for Eclipse and its associated plug-in development environments.

For further information, contact Curt by e-mail at cshwaderer@opensystems-publishing.com.