Newsletter Archive

Eclipse Source

Issue Date: 12 April,2005

Eclipse Evolves into Its Own Universe

http://www.eclipsenews.com/f3/index.cfm?fuseaction=newsletters.getArticle&webID=1003&newsID=5024&issueID=5142&articleID=50708

by Todd Williams
contributor

April 12, 2005 —

Based on the download statistics, it's obvious that millions of developers worldwide are now using Eclipse as their primary Java IDE. But what most of them don't realize is that Eclipse is neither an IDE nor limited to only Java. According to Eclipse.org's Web site, "Eclipse is an open platform for tool integration built by an open community of tool providers." Perhaps surprisingly, there's no mention of Java or an IDE. Now that Eclipse 3.1 is being readied for release, let's pause for a moment to take a look at what Eclipse really is and what it's likely to become in the future.

Just What Is Eclipse?

At its core, Eclipse is a set of open-source projects that is managed under the standard open-source rules of engagement: Meritocracy, transparency, and open participation. However, it's the operating model of the foundation as a whole that makes Eclipse truly unique. The Eclipse Foundation is a non-profit corporation currently composed of more than 90 member organizations. The membership contributes the vast majority of the foundation's funding and development resources for the open-source projects. This marriage of extensive corporate sponsorship for collaborative open-source development is a novel approach that lets each member organization contribute only in its areas of expertise and specific interest, while benefiting enormously from the cumulative effort. Given this ability to leverage their individual efforts, the size of the foundation, and its continued expansion, it's now easy to see why so many companies feel safe building their strategic products on Eclipse technology.

Interestingly, the foundation's membership roster includes many companies that compete fiercely in the commercial software tools market. However, they realize the collective benefit that is gained by cooperating to create compatible infrastructure and extensible frameworks, upon which they build additional capabilities to compete at a higher level. Competition drives product innovation, which results in increased user satisfaction,

which brings more users into the community, which creates the demand for more Eclipse-based products. Since Eclipse lowers the barriers and costs for entry into the tools segment, it encourages creative entrepreneurs to offer competitive products and services at greatly reduced rates. These market forces will likely give rise to new, secondary, or derivative products and services creating even greater opportunities. And we're already seeing the beginning of this, as the commoditization and value-creation cycle has engendered an Eclipse product ecosystem, the likes of which has never been seen before. As a result, Eclipse has become the cross-platform common runtime environment at the base of a new and fast growing segment of the software tools industry.

But that's just the beginning. Until Eclipse 3.0 arrived last year, most application developers had to design their own architecture and then build it into an application framework themselves. The costs of this approach are considerable expense, time, debug effort, support, and aggravation expended on solving a problem that is peripheral to building the business functionality of the application. In Eclipse 3.0, the development team completely separated the base runtime platform, known as the rich-client platform (RCP), from the development tools infrastructure. The result is an extensible, cross-platform application framework that is suitable for building almost any kind of desktop or rich-client application. Already, we're beginning to see the first wave of commercial RCP-based applications in a wide variety of industry verticals, with more announcements almost weekly. In addition, many organizations are standardizing on the RCP as the architecture for their in-house applications, thus enabling them to run as a manageable, integrated suites rather than as disassociated silos of functionality.

Where Is Eclipse Heading?

But Eclipse is continuously evolving and will continue to grow both vertically, further into the software tools space, and horizontally, into completely new market segments. Interestingly, the growth into new industry verticals will be for the same reasons that Eclipse was formed in the first place. Although Eclipse was initially formed to build an integration platform for software tool providers, the separate availability of the RCP changes everything. Rather than being a platform exclusively for tool providers, Eclipse has become a general-purpose platform that has simply been leveraged initially in the software tools arena. With this seminal change, Eclipse will begin to draw participants from other verticals that want to cooperate in the same manner that the current group of tool providers has. In the near future, I expect to see interest in building infrastructure for productivity applications, modeling, security, and process workflow among others. Now that Eclipse is completely open and inclusive across the entire software industry, its membership and growth will explode in the coming years.

Another vehicle of Eclipse's future growth will likely come from completely outside the software industry. Consortia â€” from such diverse industries as healthcare, insurance, automotive, and finance â€” regularly set software platform and interoperability standards. However, without a portable, cross-platform implementation of the standards, consortium members must independently construct their own, solely based on the industry specifications. This tremendous duplication of work is both expensive and error-prone. Collaborating to build a common set of specification-compliant infrastructure would universally reduce costs while insuring interoperability. But what competitors require before they can cooperate is a level playing field that benefits all of them equally. When they begin to research their options, they will find that Eclipse's maturity, extensibility, and royalty-free redistribution model is very attractive as the base of their own collaborative development efforts.

As I said earlier, Eclipse isn't a Java IDE. However, the Java IDE included as part of Eclipse will be the key to one of its largest growth opportunities. The number of software developers worldwide continues to rise rapidly, and as it does, the market for development tools increases. However, in some markets, traditional software development tools simply aren't affordable or budget cuts have forced companies to look to non-traditional sources for their tools. The Java IDE is Eclipse's "hook" into the developer community because it is both excellent and free. But after developers choose Eclipse, many naturally become curious about the platform itself and begin extending it with their own plug-ins. As the IDE continues its rapid market penetration, there will be a commensurate increase in competent plug-in developers, such that at some point, it will be as easy to hire an

Eclipse plug-in developer as a basic Java developer. This rapid increase in skilled Eclipse plug-in developers will allow corporations to easily acquire the skills they need to accelerate Eclipse's adoption as the base infrastructure for both in-house applications and commercial products.

Eclipse is constantly expanding, evolving, and surprising us all â€“ so much so that I don't think anyone could've envisioned where it has gone in its first few years of existence. And while I've tried to do a reasonable job predicting what is next for Eclipse, there's only one thing for certain: The future is arriving every day, and no one really knows what it holds. Software visionary Alan Kay once said, "The best way to predict the future is to invent it." I agree, and that's what I believe we're doing right now at the Eclipse Foundation.