



CA's On-going Commitment to the Eclipse Foundation and Mainframe Platform

John Dueckman, Product Management, Business Service Optimization
August 2006

Background

As a computing platform, the z/OS platform is incredibly rich in its ability to handle virtually any task assigned to it. From providing unbeatable scalability to up-time availability, the time-proven stability of the architecture has proven its worth to sites around the world for many years.

For the developer, time has also provided a very rich selection of tools, utilities, and languages. For many shops, the ultimate development platform that served as an integrated development environment (IDE) has been the ability to run IBM's Time Sharing Option (TSO) with the Interactive System Productivity Facility/Program Development Facility (ISPF/PDF). This provided all third-party vendors, as well as every site, with a stable, documented, standard interface to z/OS components that became known and accepted as the manner in which developers worked on the platform.

As more shops began to integrate distributed platforms, however, it became evident that a new IDE was needed. Since development was now happening on both mainframe and distributed systems, developers have been forced to:

- Only work on mainframe systems using IDE's specific to that environment. In other words, the current staff of experienced z/OS developers is restricted in their exposure to distributed IDE's and typically solely develops on the mainframe.
- Only work on distributed systems using IDE's specific to that environment. New staff, typically untrained on z/OS platforms, work in "new" IDE's usually of their choosing since standards and disciplines for distributed platforms are commonly still evolving.
- Work on both mainframe and distributed systems, but switch between IDE's as different components are being addressed. Cross trained staff must switch between what is typically the familiar (ISPF/PDF) to the unfamiliar as they work on different platforms at different times.

The two dominant IDE's available for distributed development at this time are .Net from Microsoft and Eclipse from the Eclipse Foundation. Of these two platforms, Eclipse has the most to offer in providing a single IDE that encompasses both distributed and mainframe technologies.

What is Eclipse?

“Eclipse is an open platform for tool integration built by an open community of tool providers. Operating under an open source paradigm, with a common public license that provides royalty free source code and world wide redistribution rights, the eclipse platform provides tool developers with ultimate flexibility and control over their software technology.”¹

In December 2004, CA joined the Eclipse Foundation as a Strategic Developer and took on a leadership role in the Enterprise Management tools projects. One of our first interfaces made available using the Eclipse framework has been a plug-in for our CA Harvest Change Manager (CA Harvest) change and configuration management solution. As members of this community and as an interested party in the revitalization of the z/OS platform, the question becomes whether or not the Eclipse framework can be extended to help provide the next generation of zSeries developers with this new, exciting single IDE.

At a pragmatic level and for the z/OS developer, a “base” platform needs to be provided with a “standard” interface protocol that can transcend across traditionally client-server platforms and into the z/OS operating system. This is the role that WebSphere and Eclipse can play in conjunction with each other. In essence, WebSphere becomes the new “TSO” and Eclipse is the new “ISPF/PDF”. The advantage of this approach is that, by sharing an IDE that is also used for distributed platforms, a developer can leverage advantages from both models to develop leading-edge solutions without having to switch between interfaces in order to access different tools. Since Eclipse is an open platform, the single interface has the ability to interact simultaneously with a variety of plug-ins that vendors can make available.

Consider the following on Eclipse adoption rates across industries:

- **At least 44% of IT organizations are now aware that they are using Eclipse.**
At least somewhere in the hierarchy, the latest survey data tells us that somebody knows that Eclipse is in use even if that person isn't the most senior decision-maker. Discounting the organizations that are not candidates to use Eclipse because of their platform or technology choices, the adoption is 50% of candidate organizations.
- **The true penetration rate is probably more than 50%.**
This estimate is based on what past interviews have revealed about hidden Eclipse usage that doesn't show up directly in the survey, and it is confirmed by the admitted unofficial usage and the differential responses by job type.
- **As many as 65% to 75% of Java development teams may be using Eclipse.**
This estimate includes those using Eclipse-based commercial tools. In many cases, they may be using Eclipse in conjunction with other tools and even other Java IDEs for specific projects or purposes. This estimate is based on interviews, as well as what we know from previous surveys about the typical demographic mix of the survey pool, allowing us to estimate the proportion of respondents who are pure Java shops, pure Microsoft shops, or have a mixed environment (including platforms that are neither Java nor .Net).

What has become evident is that Eclipse is already in use in shops around the world, whether a traditionally z/OS site is aware of it or not, and a large and growing number of developers are already acquainted and comfortable with its protocols.

¹ <http://eclipse.org/org/index.html>

² Source: Forester Research, “Eclipse Adoption Rates: Emerging Into The Light”, Carl Zetie, July 1 2005

z/OS Development

The Eclipse Foundation already provides a free beta COBOL plug-in for the Eclipse platform developed by Fujitsu that can be used for COBOL development. As a starting point, this provides the z/OS developer with a commonly-used language-intelligent editor that exceeds the ability of using “dumb” line editors or basic editors such as Notepad.

Additionally, Integrated Service Vendors (ISVs) are able to offer their own branded versions of Eclipse. One such branded version is the IBM WebSphere Studio family of IDE’s. One of the strategies that IBM is pursuing in helping to modernize the experience with z/OS is to leverage their WebSphere offering into “traditional” MVS development activities. To that end, they have a variety of solutions including the WebSphere Developer for z/Series.

“WebSphere Developer for zSeries V6.0 includes significant enhancements built on a comprehensive set of capabilities that help make traditional mainframe development, Web development, and integrated mixed workload or composite development faster and more efficient.”³

CA believes IBM’s strategy is the right one; Eclipse continues to prove it is capable of handling multiple workloads across a variety of platforms and is making ongoing inroads as the IDE of choice amongst various sites. As more vendors begin to provide more plug-ins for their tools and other development communities contribute other tools to the Eclipse Foundation, the framework will continue to grow in importance and relevance across all platforms.

CA Endeavor® Change Manager Evolution

As the next generation of developer begins to make more and more use of the Eclipse framework, CA has anticipated the need to ensure CA Endeavor Change Manager (CA Endeavor) is ready for the transition. The new Eclipse Plug-in for CA Endeavor demonstrates CA’s ongoing commitment to both the Eclipse Foundation and the mainframe platform.

Given the growing trend of developing cross platform applications with business logic remaining on the mainframe, the CA Endeavor Eclipse plug-in allows the developer to make use of Eclipse to execute CA Endeavor actions such as retrieve and add/update, as well as browsing results from the actions CA Endeavor executed.

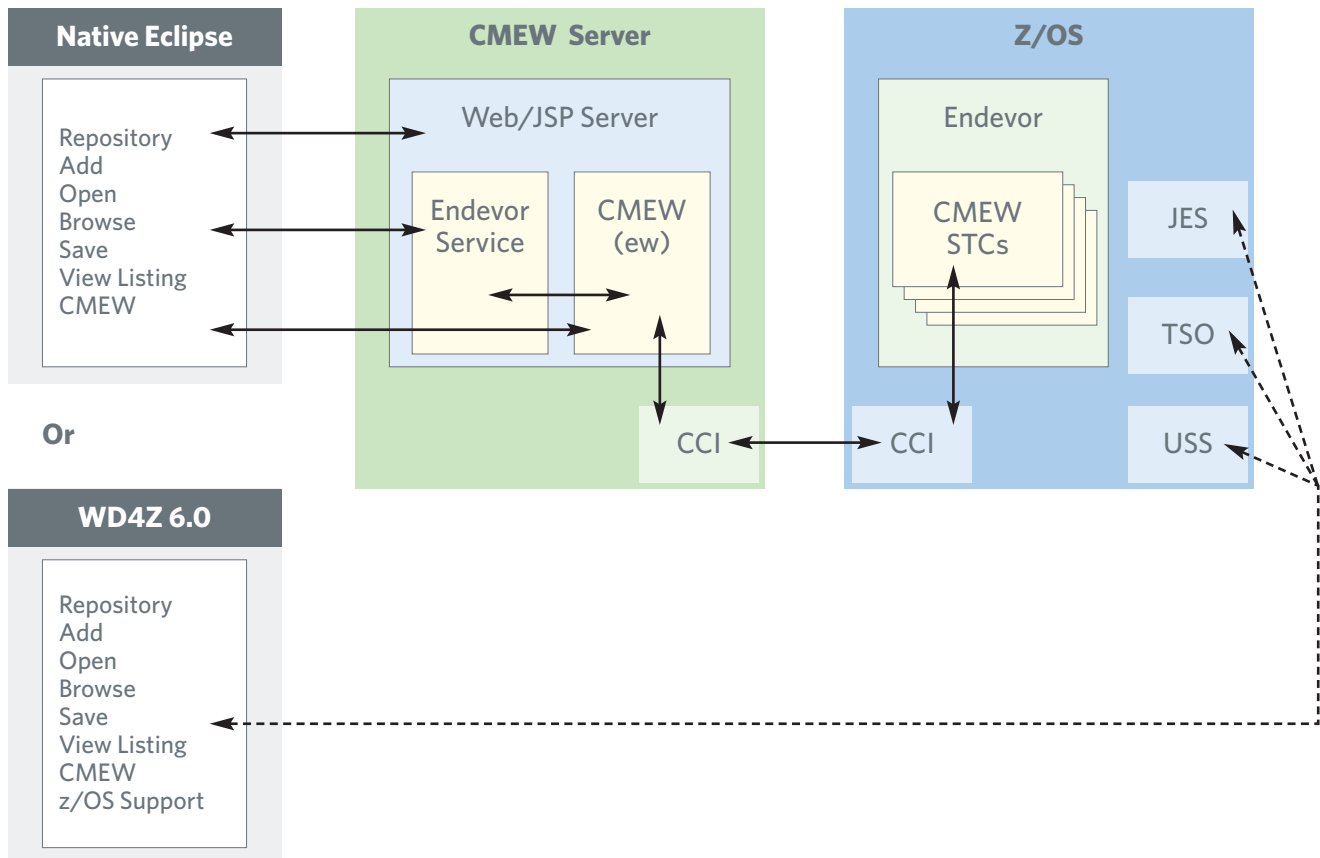


Figure 1. High-level Architecture of Eclipse Plug-in

³<http://www-306.ibm.com/software/awdtools/devzseries>

The CA Endeavor Eclipse Plug-in makes use of CA Change Manager Enterprise Workbench (CA CM Enterprise Workbench) acting as an CA Endeavor Service. Since this service is making use of the CA Endeavor standard and published API, all CA Endeavor actions are secured per whatever definitions have been determined at the executing site. This approach provides CA with the ability to ensure ongoing exploitation of the IDE as new functionality is made available in the CA Endeavor API. Using either "Native Eclipse" or IBM's Websphere Developer for zSeries (WD4Z), the CA Endeavor Service is invoked which in turn provides communication to z/OS through CA's Common Communication Interface (CCI) component.

The plug-in allows a variety of perspectives and view of elements in CA Endeavor while at the same time allowing the developer to switch between the different perspectives they may have available; from plug-ins available from the Eclipse Foundation to those made available by third party vendors such as IBM.

CA's CA CM Enterprise Workbench itself may also be invoked from the plug-in in order to ensure the IDE is as complete as possible in its scope.

The Future

An increasing number of CA mainframe products are designing and planning Eclipse plug-ins; CA Endeavor is merely one of the first.

CA's Business Service Optimization solutions are dedicated to providing a single mechanism and process for capturing and prioritizing business demands. We strive to deliver critical insight into and control of the assets, processes, people, and projects supporting those services.

As a strategic technical platform, the Eclipse Framework allows us to continue to provide solutions that transcend across disparate development models.

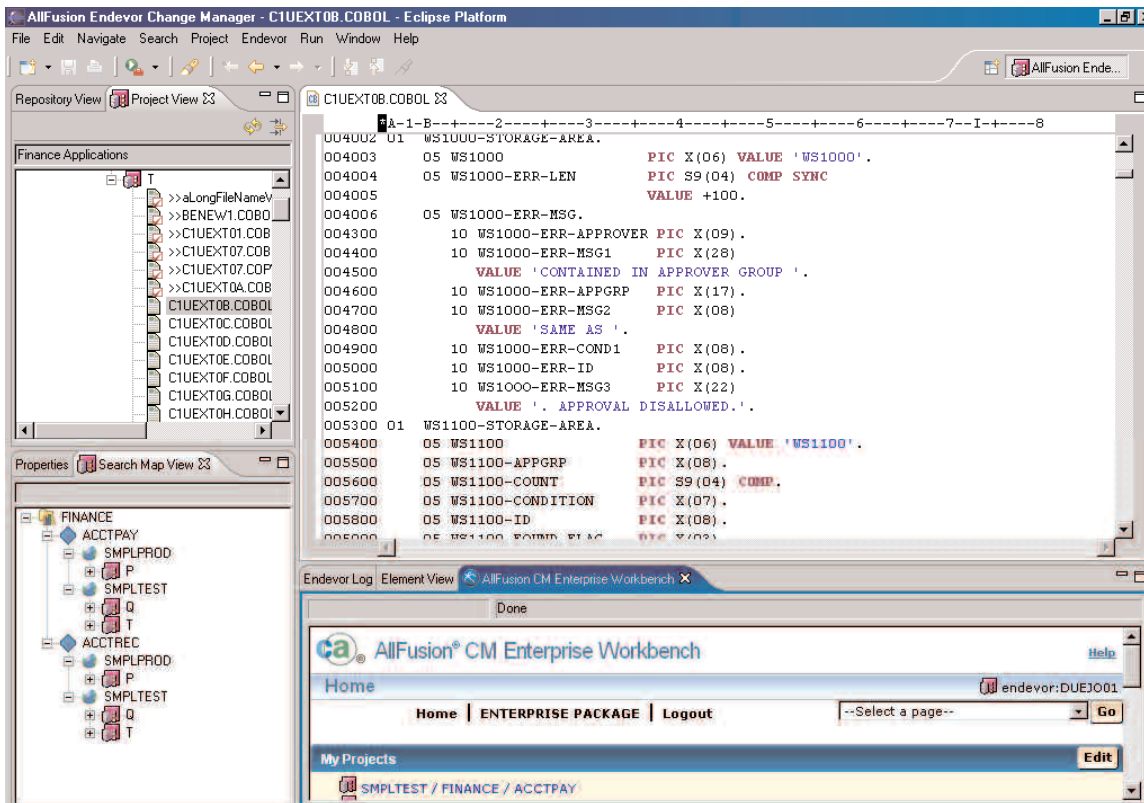


Figure 2. The CA Change Manager Enterprise Workbench is invoked from the Eclipse Plug-in, ensuring the IDE is complete



**Quality
Endorsed
Company**
ISO 9001
Lic. 2443

