BEA Uses RCP Developer™ to Improve Quality and to Save Time & Money

BEA relies heavily on the WindowTester™ component of the RCP Developer™ software from Instantiations to automate testing of GUI elements

Overview

BEA Systems and Instantiations have had a unique relationship while working in the Eclipse ecosystem. BEA is a strategic member of the Eclipse Foundation, has a representative on the Eclipse Board, and is an active participant in the Eclipse Web Tools project. Instantiations is a long time Eclipse member company, has a representative on the Eclipse Foundation Board, and has had a number of committers on Eclipse projects. The work delivered by Instantiations in the Eclipse Pollinate project provided early proof-of-concept technology for BEA's Eclipse-based tooling.

BEA Workshop makes developing Java applications easier by allowing Eclipse developers to quickly create, debug and test SOA components, Web Services, Web applications, BEA WebLogic Portal applications, and enable Service-Oriented Architecture (SOA) solutions. BEA Workshop is based on the Eclipse Open Source Integrated Development Environment (IDE). Eclipse includes a Rich Client Platform (RCP) layer that makes it easy to develop applications with extensive GUI capabilities.

BEA also utilizes other key Eclipse-based tools to develop and test their own applications. For example, BEA replies heavily on the WindowTester component of the RCP Developer software from Instantiations to automate the testing of GUI elements. RCP Developer is a software development product that accelerates the creation of Eclipse RCP applications by providing tools for constructing and testing graphical user interfaces, composing Help documentation and packaging rich client applications for deployment. We will explore how BEA is using the WindowTester component of RCP Developer in the remainder of this case study.

> "In terms of our cost of investment in the WindowTester tool, we believe the cost of the licensing is greatly worth the enhancement and efficiency gained so far."
>
> *—Bill Roth,*
> *VP of BEA Workshop Unit*

Development Challenge

Three years ago, Workshop's automated IDE-based testing was from a home-grown testing infrastructure. BEA is shifting its product release schedule so releases will be delivered in 1/6th the time previously required. According to Steve Tocco, BEA Workshop Director of Quality Assurance, "Our internal IDE testing had inadequate code coverage numbers with 1/20th the number of tests our current automated test suite contains. We determined that the amount of intermittent failures, the inadequate automated coverage, as well as the cost to implement tests was prohibitive in getting products to market quickly." The BEA Workshop group needed a better solution for automated testing of their application GUIs. They chose Instantiations RCP Developer and its innovative WindowTester functionality to meet this need.

Solution: Using WindowTester for Automated GUI Testing

Testing History

In addition to the in-house testing infrastructure used by BEA Workshop group, they previously used and evaluated other commercial off-the-shelf products. While some of the tools provided a quick-click and record of Swing tests, they would not address the needs raised as Workshop moved to the Eclipse infrastructure in 2004. They researched options and decided to initially create their own test suite using the open source Abbot SWT GUI testing framework. For the product release that shipped in July 06, they used the Abbot structure for testing.

However, the team found that creating and running tests still did not meet their needs. The group decided that developing and maintaining their own test harness was not their core competency and they did not want to devote extensive resources to creating a test infrastructure. Tocco states, "To meet a rapid release schedule, the test suite must be optimized as we can't afford false negatives or instabilities drawing precious resources away from our deliverables. Even more, tests need to be authored rapidly as we try to find optimizations in our schedules while not compromising quality for our customers. We weren't getting that in any form before. Our team felt we weren't agile enough to meet the new release timelines—we simply don't have time now to rely on manual tests alone."

Moving to WindowTester for Automated GUI Testing

The BEA Workshop team began looking at tools that could automate the testing of GUI elements of their Eclipse-based applications. They chose RCP Developer from Instantiations and are converting their test suite to use its WindowTester component. They selected a commercial product even though there are open source alternatives available because of the tools' ease of use as well as Instantiations' support, responsiveness, and professional documentation. According to Bill Roth, VP of BEA Workshop Unit, "In terms of our cost of investment in the WindowTester tool, we believe the cost of the licensing is greatly worth the enhancement and efficiency gained so far."

How Testing Has Changed after Implementing WindowTester

When evaluating RCP Developer and WindowTester, the BEA Workshop for WebLogic team found that it took about two months to achieve a 40% line coverage in the product from a code coverage perspective. According to Tocco, "Since we started using WindowTester, tests that took two to three weeks to write previously can now be done in two to three days. It is much faster to get a test developed and running. In addition, we have a higher stability rate than before with a extraordinary pass rates in our automated regression tests. This is fantastic. It lets us spend our energy building the product, not chasing test issues." The BEA Workshop for WebLogic team also found that WindowTester can handle a heavy test load. The team currently runs over 300 tests. Some of the tests are quite broad, with multi-step lengthy testing scenarios. This is a 20 times increase over what was in the release two years ago for IDE automation.

"Since we started using WindowTester, tests that took 2–3 weeks to write previously can now be done in 2–3 days."

—Steve Tocco,
BEA Workshop
Director of Quality Assurance

Future: Full Migration to WindowTester

The BEA Workshop for WebLogic team plans a complete migration away from the old test harness for the IDE in six months. The team intends to use the WindowTester component of RCP Developer as the sole tool for automated IDE testing tool in Eclipse by the end of 2006.

Summary

Moving to Instantiations RCP Developer and its WindowTester functionality has allowed the BEA Workshop for WebLogic group to drastically cut the time it takes to generate new GUI tests. The move to WindowTester has saved the group both time and money allowing them to focus on developing their product rather than creating and maintaining a test infrastructure.

About BEA

BEA Systems *is a company founded in 1995 that specializes in enterprise infrastructure software, and has 77 offices in 37 countries. BEA Systems, Inc. is a world leader in enterprise infrastructure software, delivering powerful standards-based platforms for building enterprise applications and managing Service-Oriented Architectures even in heterogeneous IT environments. Customers depend on BEA Tuxedo®, WebLogic®, and AquaLogic™ product lines to reduce IT complexity, leverage existing resources, and speed the delivery of new services. BEA also provides support for Blended strategies that combine Open Source and commercial software to best suit the needs of business and IT. With over 15,000 customers including the majority of the Fortune Global 500, BEA provides the technology, solutions and services to help companies achieve a state of Business LiquidITy™ where enterprise assets are freed up to deliver maximum business value.*

www.bea.com

About Instantiations

Instantiations, Inc. *provides leading-edge software products, services and technologies for Eclipse, Java and Smalltalk. Instantiations offers professional development environments and software products that integrate seamlessly with the latest development platforms. Instantiations is a member of the Eclipse Foundation and offers a line of products for Eclipse, Rational Application Developer, IBM WebSphere Studio and MyEclipse. Based in Portland, Ore., Instantiations was founded in 1997 by a team of internationally recognized pioneers in the field of component software technology.*

www.instantiations. com

instantiations

This Case Study is available online at
www.instantiations.com/rcpdeveloper/resources/casestudy-bea.pdf

About Eclipse

Eclipse is an open source community *whose projects are focused on providing an extensible development platform and application frameworks for building software. Eclipse is an open source community whose projects are focused on providing a vendor-neutral open development platform and application frameworks for building software.* **The Eclipse Foundation** *is a not-for-profit 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.*

www.eclipse.org

What are RCP Developer and WindowTester?

The RCP Developer product from Instantiations *is an Eclipse-based software development product that accelerates the creation of Eclipse Rich Client Platform (RCP) applications by providing tools for constructing and testing graphical user interfaces, composing Help documentation and packaging rich client applications for deployment. RCP Developer contains four core functionality components. SWT Designer™ empowers developers to quickly create the views, editors, perspective, preference pages and other user interface elements that comprise Eclipse RCP applications. Help Composer™ streamlines the creation of documentation that is fully compatible with the Eclipse Help system. RCP Packager™ simplifies the build and deployment process for Eclipse RCP applications by assembling application elements together into a single package for easy deployment.*

The WindowTester™ component of RCP Developer™ *lets software developers easily create and execute unit tests for Eclipse SWT graphical user interfaces (GUIs) they build. WindowTester captures and records actions such as keyboard clicks and mouse movements, then uses this information to generate JUnit based system level GUI tests. WindowTester automates the execution of GUI test suites and includes intelligent parsing and unique identification of user interface elements. Since WindowTester generates tests in pure Java, generated tests are easy to understand, modify and maintain.*