



Media Contact:

David Orme

847-605-1311

daveo@asc-iseries.com

Visual Editor Project Q&A

- Q:** What companies are participating on the Visual Editor Project team?
- A:** Advanced Systems Concepts, Canoo, IBM, Instantiations, and Red Hat are the core contributors.
- Q:** Who is contributing what code?
- A:** The Visual Editor project team has accepted the contribution of IBM's Visual Editor for Java code base as the "seed" on which everything else will be built. Advanced Systems Concepts, Instantiations, and Red Hat are adding SWT support to this initial contribution.
- Q:** Eclipse is an IDE for other languages in addition to Java. For example, C and C++ are also supported through the C Developer Tooling project (CDT). Will the Visual Editor Project be able to edit user interfaces for other languages in addition to Java?
- A:** Nobody has stepped forward to contribute this support. However, the reason that the Visual Editor Project team chose to accept IBM's source code as the initial contribution is because of all the choices that were available, it is best suited to supporting multiple languages.
- Q:** What other code bases were considered?
- A:** Advanced Systems Concepts has a GUI editor for Eclipse's Standard Widget Toolkit (SWT) called SWTworkbench. Instantiations has SWT Designer, another GUI editor for SWT. However, neither of these code bases was designed to support multiple target languages. In discussions among all three parties, we mutually decided that IBM's code base was the best fit toward meeting our project's objectives.
- Q:** Why would companies that are involved in creating GUI builder technology decide to commoditize their own markets by releasing competitive technology as open source?
- A:** While each company involved has its own reasons for doing this, collectively we believe that the opportunity to have a standard graphical editor platform on which we all can build *far outweighs* the immediate cost to us of delivering this open-source project.
- Q:** What kind of opportunity are you talking about?
- A:** Eclipse has a rich heritage of creating projects that dramatically reduce the market barriers-to-entry for problems that have traditionally been difficult to solve. For example, the Java Development Tooling project is not just a first-class Java programmer's editor. It also exposes the internals of Eclipse's Java compiler so that other tools that need to directly manipulate Java source code can do this much more easily than if they had to create and maintain their own a Java compiler. In

fact, the Visual editor project inherits its own Java source code parser from the Java Development Tooling project's compiler. In a similar manner, people will be able to create new, sophisticated tools for manipulating graphical user interfaces by taking advantage of the open APIs that will be exposed by the Visual Editor project.

Q: Why are you going to support SWT in addition to Swing?

A: There have been hundreds of messages posted to the Eclipse newsgroups requesting (or looking for) a GUI builder for SWT. In short, developers have been clamoring for it; we intend to deliver it.

Q: Doesn't SWT just run on Windows?

A: No. SWT applications run on all of the most popular operating systems including Windows XP, Windows 2000, Windows 98, Windows ME, Red Hat Linux (x86/Motif and x86/GTK), SuSE Linux (x86/Motif and x86/GTK), Solaris 8 (SPARC/Motif), QNX (x86/Photon), AIX (PPC/Motif), HP-UX (HP9000/Motif), and Mac OSX (Mac/Carbon).

Q: Do you believe that SWT is superior to Swing?

A: We believe that SWT and Swing are complementary technologies. Depending on a project's objectives, either could be a good choice. The Visual Editor Project will enable customers to choose the best technology available to meet their needs.

Q: I heard that SWT and Swing are going to be interoperable? Is this project going to enable that to happen?

A: While the Visual Editor Project applauds the SWT and Swing interoperability goals in Eclipse 3.0, we are not the ones who are delivering this feature. However, we agree that this is an important feature to support going forward.

Q: Does the Visual Editor support Java code that is produced by other IDEs such as JBuilder or SunOne (Forte)?

A: The current Visual Editor will recognize VisualAge for Java code patterns, as one would expect. We have started adding JBuilder and SunOne code pattern support as well. This effort is not complete, but most of it is there. Customer demand will determine how highly this feature is prioritized along with all other requested features.

Q: In addition to SWT and Swing, what other UI technologies are you considering supporting?

A: Nobody has yet volunteered to add support for anything but Swing or SWT. However, the architecture that we are creating is designed to accommodate any programming language and any platform. For example, it will be possible to create a GUI builder that operates in real time on small, "embedded" devices, yet displays the user interface that is being designed on the developer's workstation.

Q: I would like to participate in this project, how can I?

A: The Visual Editor project team is making final test passes on the Swing-based GUI builder implementation. The fastest and easiest way to learn what we're doing is to join us in the testing. Go to <http://www.eclipse.org/vep> and follow the “Visual Editor Test Pass” link. The VE Project team has written some nice tutorials to make this easy and fun.

Q: What is your schedule for delivering this?

A: Swing support is available today. We expect to deliver SWT support during the first half of 2004.

Contact Information:

David Orme
Visual Editor Project Lead
Advanced Systems Concepts, Inc.
1300 Woodfield Road, Suite 310
Schaumburg, IL 60173

Phone: (847) 605-1311
Email: daveo@asc-iserics.com

Eclipse web site:
<http://www.eclipse.org>

Visual Editor Project web site:
<http://www.eclipse.org/vep>