

# Eclipse Focus: CDT 3.0

## Q & A with Doug Schaefer, QNX Software Systems, and Chris Recoskie, Texas Instruments

By Don Dingee and Chad Lumsden

*ECD recently had the opportunity to pose a few questions to Doug Schaefer, the new Eclipse C Development Tools (CDT) Project Leader and Senior Software Developer at QNX Software Systems. Doug worked with a committer on the Eclipse CDT team, Chris Recoskie of Texas Instruments (TI), in answering some of our questions. In these responses, we gain some insights into the latest version of CDT 3.0 and how it is being used by software developers.*

**ECD:** What advantages does CDT 3.0 offer for embedded developers in coding, optimizing, profiling, and debugging real-time code?

**Schaefer:** We've implemented a number of usability improvements in CDT 3.0, with significant effort being placed on the CDT core's parser-based features, the managed build system, and the CDT debug views. The core features a faster parser and a complete abstract syntax tree, together called the DOM. This feature enables faster and more accurate refactoring, content assist, and searches. All of these improvements help embedded designers better understand their code and improve productivity.

We've also made a number of enhancements to improve the usability and flexibility of the managed build system. You can now specify pre- and post-build steps, as well as custom steps per file for special file types. You can also specify environment variables and build macros that will be used for the build.

Debug enhancements include user-definable register groups for better organization of registers on the target while debugging. We've also added a new modules view that allows you to inspect the components of the application under debug and to set breakpoints on the contents of those components. Plus, an enhanced memory view improves usability when you're inspecting memory on the target.

**ECD:** Since the release of CDT 3.0 in August, who has adopted it?

*(Editor's Note: The Eclipse Foundation press release on CDT 3.0 dated August 22, 2005 mentions QNX, IBM, Intel, MontaVista, Novell SUSE, PalmSource, Tensilica, Texas Instruments, and TimeSys.)*

**Schaefer:** A number of the contributors to CDT 3.0 are working to adopt it into their products. QNX Software Systems, for example, will integrate CDT 3.0 into the next release of its QNX Momentics development suite. Texas Instruments, meanwhile, is adopting CDT 3.0.x for its upcoming Code Composer Essentials 2.0 for the MSP430 microcontroller. TI is also adopting CDT 3.0.x and beyond for a product roadmap that will address some of its other

silicon offerings. Intel is also among the companies integrating the CDT into their products.

**ECD:** What types of features does CDT 3.0 have to improve software development for multicore processors?

**Schaefer:** The flexibility and extensibility offered by the CDT offers interesting possibilities for developers working on multicore systems in both symmetric and asymmetric configurations. In Asymmetric MultiProcessing (AMP) systems that involve multiple operating systems and tool chains, a developer can target all of the operating systems from the same code base, using the same IDE. In the upcoming years, we'll likely see multicore AMP systems that run Linux on one core and a real-time operating system on another core. The CDT can integrate with both tool chains and allows for debugging of both cores at the same time.

**ECD:** What types of extensions are developers adding to CDT 3.0 and the managed build system? Can you give a couple of examples?

**Schaefer:** A number of vendors are integrating their tool chains into the managed build system. These include Intel and Texas Instruments, who both serve as committers for the managed build system. QNX is also planning an integration in an upcoming QNX Momentics release.

TI is using the new capabilities in the managed build system to provide support for its proprietary code generation tools. This gives its embedded customers the ability to reap the benefits of TI's optimizing compilers – such as global and inter-procedural optimizations for both code size and speed – within an Eclipse environment. TI is also taking advantage of the new capability to customize the New Project wizard to provide a customized project creation workflow for users – this will let users quickly configure their project to properly build and debug for their specific target silicon, embedded operating system, and so on.

There is a lot of interest in the managed build system on the mailing lists and bugzilla, so we anticipate that there are many others, including users, who are working on integrations.

Thanks to the contributions of TI, Intel, Symbian, and others, the managed build system in CDT 3.0 has reached a new level of extensibility for ISVs that we can all reap the benefits of.

## **ECD: What has surprised you in ways developers are using CDT 3.0?**

**Schaefer:** Certainly the biggest surprise is the great work by the Photran team out of the University of Illinois Urbana Champaign (UIUC) and the Los Alamos National Laboratory, which are taking the CDT and customizing it to work for Fortran. There is still a lot of interest in Fortran for use for complex scientific applications. Right now, the Photran team takes the CDT and, applying a fairly small patch, gives it the Fortran flavor. We look forward to working with this team to make the patch unnecessary and to open the door to further language integrations.

When you look through the list of participants on the CDT newsgroup, mailing list, and bugzilla, you quickly see the diversity of projects that people are building with the CDT. We have folks developing systems from bare-iron embedded systems to large enterprise systems that target mainframes and Linux servers through game consoles. This diversity is what makes the CDT such a rich community and ensures that the CDT remains flexible, extensible, and applicable to a wide array of developers.

We were pleasantly surprised to find out that a developer working on Mozilla has tried the CDT on the Mozilla source base and had a measure of success with it. Mozilla is one of the largest and most complex open source projects, and is the benchmark that the CDT developers use to measure performance and scalability

of the CDT. Work continues to ensure the CDT can be used on Mozilla and similarly large-scale projects.

It's also interesting to note that TI uses CDT plus GDB to debug its own GDB ports that support TI silicon.

## **ECD: What's next on the plate for the CDT committers? Is there a CDT 3.1 or 4.0 in the works? Are additional committers stepping up?**

**Schaefer:** CDT will proceed with a six-month release cycle, with additional performance and usability improvements in the works. At October's CDT developer's conference we will discuss future plans for the CDT. All of the contributors to the CDT and others will get together to go through detailed designs and come up with a vision that will take the CDT through the next couple of releases.

We also have additional corporate vendors that are very interested in contributing to the CDT. These topics are on the agenda for the conference in October.

*(Editor's Note: This issue of ECD went to press before the conference – please look on [www.embedded-computing.com](http://www.embedded-computing.com) for the latest developments with CDT.)*

We are definitely excited by the diverse interest in the CDT and will continue to promote contributions that help advance the Eclipse and CDT towards being the defacto-standard IDE for all types of development.

*Doug Schaefer is project lead for Eclipse CDT and a senior software developer at QNX Software Systems. Previously, he worked for ObjecTime, Rational, and IBM, where he specialized in developing code-generation and software modeling tools. He now brings his technical leadership and experience to the team that develops the QNX Momentics IDE, a development environment based on Eclipse and the CDT.*



To learn more, contact Doug at:

**QNX Software Systems**  
175 Terence Matthews Crescent • Ottawa, Ontario  
Canada, K2M 1W8 • Tel: 613-591-0931  
Email: [DSchaefer@qnx.com](mailto:DSchaefer@qnx.com) • Website: [www.qnx.com](http://www.qnx.com)

*Chris Recoskie has been with Texas Instruments for the past five and a half years specializing in software development tools. Chris has contributed to the design and development of both the Code Composer Studio and the Eclipse-based Code Composer Essentials integrated development environments. Chris is currently a committer on the build system of Eclipse CDT.*



To learn more, contact Chris at:

**Texas Instruments**  
150 John Street, Suite 801 • Toronto, ON  
Canada, M5V 3E3 • Tel: 416-340-2098  
Email: [crecoskie@ti.com](mailto:crecoskie@ti.com) • Website: [www.ti.com](http://www.ti.com)

