

ORACLE SOLARIS STUDIO INTEGRATED DEVELOPMENT ENVIRONMENT

KEY FEATURES

- Feature-rich, language-aware code editor for C, C++, and Fortran
- Premier support for C++ templates
- Leverages powerful Oracle Solaris Studio memory, thread, and performance analysis features
- Remote development capabilities that enable the creation of Oracle Solaris or Linux applications from Microsoft Windows, Mac, or other UNIX desktops
- Support for Oracle Database application development: Pro*C, OCI/OCCI

BENEFITS

- Increases developer productivity
- Streamlines development process
- Reduces time to market

The Oracle Solaris Studio IDE is tailored for C, C++, and Fortran developers, improving developer productivity and reducing time to market.

Introduction

The Oracle Solaris Studio IDE delivers a premier environment for the development of serial and parallel C, C++, and Fortran enterprise applications. It is optimized to help maximize the return on hardware and engineering skills, enabling development teams to focus on innovating and differentiating where it matters.

Productive development teams contribute a tremendous amount of value and are great assets for organizations. Great development tools are instrumental in boosting developer productivity, allowing developers to leave the routine work behind and focus on the work that is going to give your applications a leading edge. The Oracle Solaris Studio IDE greatly increases development efficiency, enabling the creation of high-performance, high-quality, robust, and easily maintainable enterprise applications. In addition, the advanced development platform enables developers to further enhance their skillset by simplifying advanced memory, threading, and performance analysis.

Next-Generation IDE

The Oracle Solaris Studio IDE is built on the award-winning NetBeans platform. The Oracle Solaris Studio IDE extends this platform by leveraging advanced analysis and debugging capabilities from the Oracle Solaris Studio tool suite.

The NetBeans platform has attracted over a million active users across the globe. The high level of community involvement enables Oracle Solaris Studio IDE users to benefit from exhaustive community testing on thousands of projects of different styles and nature, myriads of extensible Oracle-supported and community contributed plug-ins, and the ability to extend the IDE with Java, PHP, and many other languages.

The rich C, C++, and Fortran language-aware editor is part of NetBeans and it evolves very rapidly, driven by feedback from thousands of native developers. It provides a comprehensive and complete list of features, enabling efficient coding and reverse engineering. In addition, the IDE leverages powerful Oracle Solaris Studio features, allowing users to utilize these technologies within the IDE. These include the following:

- Industry-leading debugger (dbx debugger) that simplifies the debugging of complex applications
- Performance analysis capabilities that help identify hotspots and bottlenecks in serial and parallel applications

- Thread analysis tooling that identifies hard-to-pinpoint race and deadlock conditions in multithreaded code
- Static and dynamic code analysis that highlights common coding errors, including memory leaks and usage information

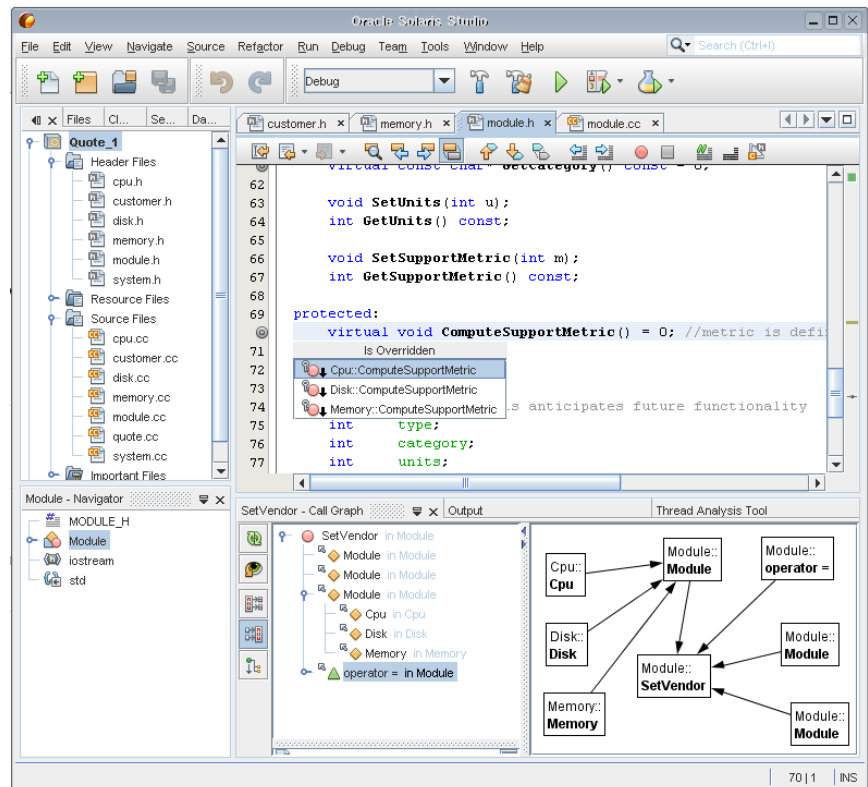


Figure 1: Oracle Solaris Studio IDE

Rich, Language-Aware C, C++, and Fortran Code Editor

The rich language-aware editor supports all the expected features available in modern IDEs, including code folding, syntax highlighting, semantic highlighting, code completion, declaration navigation, definition navigation, code snippets, and much more. It also supports a variety of ways to look for certain information including a quick search and the ability to go to a particular file, type, or symbol. The editor also provides many capabilities that are useful for reverse engineering, for example:

- **Find Usages:** Identifies all reference to a variable, function, class, or method
- **Call Graph:** Visualizes the call hierarchy
- **Type Hierarchy:** Shows the parent/child hierarchy of classes
- **Include Hierarchy:** Helps users understand include dependencies
- **Macro View:** Displays preprocessed source code and helps read code that uses meta-programming

In addition, the editor provides premier support for complicated C++ code by adding special annotations that help developers understand whether a method overrides a parent method or is overridden in a child class, a class has children, a template has specializations, and a type specializes a template. In all of these cases, it is possible to navigate to the related entities. The editor is aware of overridden methods and template specializations, offering smart navigation and code completion.

Advanced Analysis for Improved Application Quality

Oracle Solaris Studio provides advanced analysis tools to help developers create scalable, reliable, high-quality applications. The Oracle Solaris Studio IDE leverages a subset of these powerful technologies.

The Oracle Solaris Studio IDE facilitates visual debugging by exposing all the power of the Oracle Solaris Studio dbx debugger. Beyond the debugger, there is a Profile-on-Run action that leverages advanced performance analysis capabilities to collect information about the application during runtime with impressive precision and detail and with negligible overhead. It provides an intuitive visualization of CPU, memory, and thread synchronization usage, allowing users to quickly identify hotspots and bottlenecks and to easily navigate to problematic areas in the code and fix issues.

For more complex issues, there are two additional types of runtime analysis that are reported in real time: memory access errors and data races and/or deadlocks. The memory access errors feature identifies memory leaks, uninitialized memory reads, buffer overruns, and many other typical but dangerous and hard-to-find programmatic errors. The data races and deadlocks feature identifies concurrent access to data in memory that has occurred without proper synchronization as well as deadlocks. It reports both actual and potential issues.

The IDE also prevents issues while developers are coding. The static analysis feature displays errors in the editor in real time, as a developer is actively coding. This feature dramatically increases productivity by discovering bugs early in the development process.

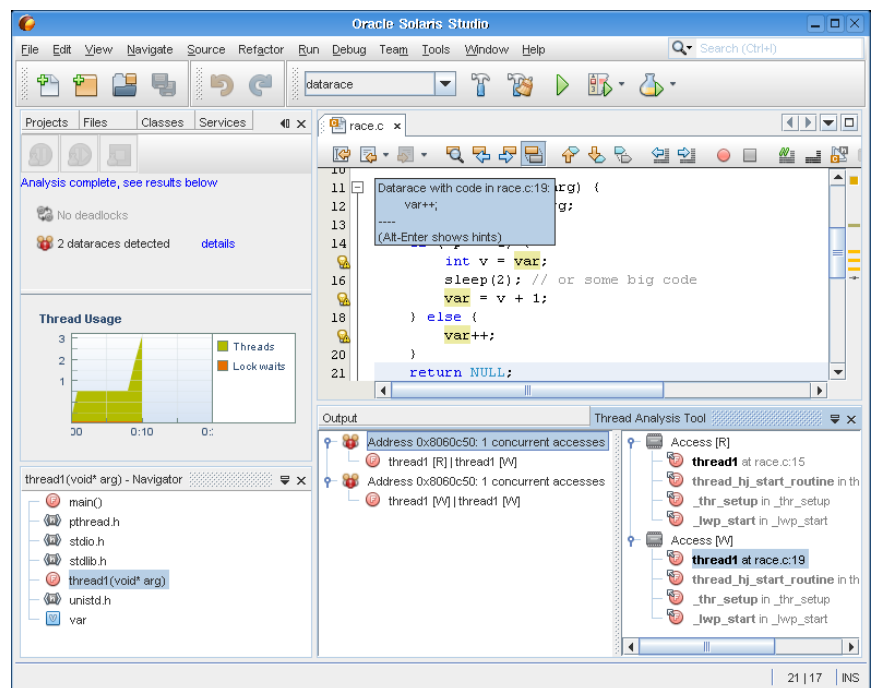


Figure 2: Advanced thread analysis—data race and deadlock detection

Remotely Develop Oracle Solaris or Linux Applications

The Oracle Solaris Studio IDE supports remote development, enabling the creation of Oracle Solaris or Linux applications from Microsoft Windows, Apple Mac, or other UNIX operating systems. This allows users to run the IDE locally on their desktop and remotely build, run, debug, and analyze applications.

After a highly automated initial setup, the remote development process is fully transparent for the user. Developers are able to create a desktop distribution for Microsoft Windows, Apple Mac, or other UNIX desktops, which provides these remote capabilities in a client-server model. The Oracle Solaris Studio IDE provides the following unique capabilities:

- **Remote file system browser:** Explore a remote file system tree or subtree.
- **Open projects from remote host:** Remote projects that have been left open in the IDE when exiting or that have been recently accessed can be opened in the same way as local projects. The user does not have to remember where the projects reside. For other projects, it is possible to access the projects through a remote file system browser and simply open them as if they were local.
- **Create remote project:** Easily and quickly create a remote project.
- **Remote toolbar:** Enabled by default in the desktop distribution, this toolbar provides users with a quick and easy way to select a default development host, connect/disconnect the default host, create/open a remote project, and open a remote file.

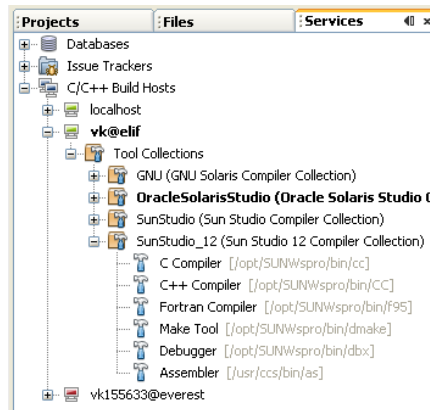


Figure 3: Remote development

The remote development capabilities help developers better utilize desktop resources and more efficiently create server-side enterprise applications. The IDE utilizes only desktop resources and is typically more responsive compared with remote desktop technologies, such as VNC. This lower use of server resources enables more productive use of development servers. In this model, tools, such as the compilers, debuggers, and advanced analysis tools, are invoked on the server using an SSH protocol, and remote files are accessed using a secure SSH/SFTP protocol or an alternate file-sharing mechanism specified by the user.

Complete Support for Large Enterprise Applications

The Oracle Solaris Studio IDE provides excellent support for large enterprise applications. It has been specifically optimized in the following areas:

- **Project creation:** Large, mature projects with sophisticated build systems are traditionally difficult to put under any IDE control. The simple drag-n-drop feature makes it easy to populate a project, configure editor support, and build the application.
- **Editor support:** Code navigation and analysis is typically challenging for large projects and the IDE is engineered for these cases. To ensure it deals well with a large number of files, the parser is heavily multithreaded and uses as many CPU cores as possible.
- **Version control:** All large projects are kept under some version control and support for it is a key part of a developer's productivity when using the IDE. The IDE provides first-class support for popular version control systems such as, CVS, Subversion, and Mercurial. Plug-ins for Git and ClearCase support are available in the NetBeans Update Center.

Contact Us

For more information about Oracle Solaris Studio, visit oracle.com/goto/solarisstudio or call +1.800.ORACLE1 to speak to an Oracle representative.



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2005, 2010, 2011 Oracle and/or its affiliates. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark licensed through X/Open Company, Ltd. 0611

Hardware and Software, Engineered to Work Together