



Software Product Description

**PRODUCT NAME: Developers' Toolkit for Tru64 UNIX
Version 5.0A**

SPD 44.36.22

DESCRIPTION

The *Developers' Toolkit for Compaq Tru64™ UNIX* provides a basic application development tool set for the Tru64 UNIX Operating System. This product combines the previous Developers' Toolkit and C Developers' Extensions products into a single solution.

The Developers' Toolkit for Tru64 UNIX is a prerequisite for all *Compaq Tru64 UNIX* development tools, languages, and environments. The Developers' Toolkit contains the following components:

- *Compaq C* for Tru64 UNIX, an ANSI conformant C Compiler
- Debuggers (Ladebug, dbx)
- The ATOM API
- Program analysis tools (profiling and performance analysis)
- Visual Threads (thread-related profiling and debugging)
- Porting Assistant
- Reordering tools (cord, om)

The C Compiler

The Developers' Toolkit includes the Compaq C for Tru64 UNIX compiler, a standards-compliant, multi-dialect, full-featured, and highly optimizing

implementation of the C language specifically developed to exploit the 64-bit Alpha architecture. It is invoked by the `cc` command.

The compiler provides options to compile the source code in conformance with standards including "ANSI C" (ANSI X3.159-1989[1992], or equivalently ISO/IEC 9899:1990[1994]), and conforms to the XPG4-UNIX standard with the `c89` command. It also provides IEEE floating-point support conformant with IEEE Standard 754 (equivalent to IEC 60559:1989).

The compiler supports these popular C dialects:

- Relaxed ANSI mode: Compiles according to the ANSI C standard but allows some common programming practices disallowed by the standard. This is the default.
- Strict ANSI mode: Strictly enforces the ANSI C standard (with a separate control for the namespace used in standard headers).
- Common mode: Supports the K&R programming style, with certain ANSI extensions where the K&R behavior is undefined or ambiguous.
- Microsoft compatibility mode: Interprets source programs according to certain language rules followed by the C compiler provided by Microsoft with Visual C++.

- VAX™ C compatibility mode: Compiles according to the ANSI C standard, but also supports a number of VAX C extensions that are incompatible with that standard and change the language semantics.

The compiler's diagnostic facility is highly customizable with #pragmas and command-line options. Compiler options are available to generate listing files with source, symbol table, and machine code, and to generate complete debug and traceback information in object modules for use with the Ladebug and dbx debuggers.

The compiler provides the following data types:

- 8-, 16-, 32-, and 64-bit signed and unsigned integer types
- IEEE 32-bit float, 64-bit double, and 128-bit long double floating-point types
- 64-bit pointer type (support for 32-bit pointers is also provided to facilitate porting and reduce memory consumption)

It contains extended support for systems programming, parallel programming, and mathematical computing with features including:

- Support for the OpenMP parallel programming API
- Fine-grained control over optimization behavior with #pragmas and command-line options
- User-defined assembly language sequences using #pragmas and intrinsic functions that expand and optimize these instruction sequences in line
- #pragmas and command-line options for controlling data layout
- IEEE floating-point (with optional handling of exceptional values like NaN, INF, and denormals)

- Built-in functions to generate efficient code sequences for processor interlocked memory access
- Shared library support
- Program profiling in conjunction with pixie, prof, gprof and hiprof
- Run-time feedback optimization
- Inter-language calls with other Tru64 UNIX languages
- Integration with the DEC FUSE graphical software development environment (see SPD 44.71)

Compaq C is also available on the Compaq OpenVMS™ operating system.

The Compaq Ladebug Debugger for Tru64 UNIX

Compaq Ladebug Debugger is a symbolic source code debugger that helps programmers locate run-time programming errors (bugs) in their code. It includes a customizable Graphical User Interface (GUI).

Features

- Supports multiple process debugging
 - Attaches/detaches to one or more running processes
 - Loads/unloads one or more programs into the debugger
 - Debugs across fork/exec
 - Allows process execution in the background
- Debugs multithreaded DECthreads applications
- Attaches/detaches to a running process and debugs a program

- Catches/ignores unaligned access
- Displays a source listing of a program
- Supports line editing (simple emacs binding)
- Uses Ladebug within emacs
- Debugs core files
- Sets breakpoints to stop program execution when specified sections of program code are executed
- Sets watchpoints to stop program execution when a specified area of memory or specified program variable is read or written
- Adds conditions to breakpoints and watchpoints so that program execution will only stop at the specified break or watch event when the condition is true
- Steps into a routine
- Steps through the execution of a program one line at a time
- Examines the stack of currently active functions
- Examines and changes program variable and data structure values
- Disassembles and examines machine code and examines machine register values
- Debugs programs with shared libraries
- Customizes debugging environment by using scripts and aliases for commands and command sequences
- Examines the contents of memory in various formats
- Displays variables in different scopes
- Debugs the kernel

Ladebug provides a choice of command-line or graphical user interface. The command-

line interface is a dbx-like command syntax. Ladebug provides extensive debugging support for the following languages: C, C++, FORTRAN-77, and FORTRAN-90. Ladebug provides limited support for Compaq COBOL and DEC Ada.

Both a standard and an international implementation of Ladebug ship with this version of the Developers' Toolkit. They are two distinct kits, with the standard implementation being the default. See the product documentation for details.

The Ladebug GUI supports all Ladebug command-line functionality. The GUI main window includes the basic debugging and convenience features, optional windows, various pop-up menus and dialog boxes, and a command-entry prompt.

Key Features of the Ladebug GUI

- A complete debugger command set available through the communications view
- Logical and easy access to the most common debugging functions
- Convenient call stack maneuvering via a call stack menu
- Convenient display of program execution through the source, communication, breakpoint, instruction, register, local variable, and monitor displays
- Paned windows that minimize the use of screen area
- Intelligent word selection
- Source navigation through the source browser
- A customizable push-button panel to accommodate individual debugging styles
- Data views that can be expanded to show their contents or to follow pointers

- An I/O window for application terminal I/O
- Integration with the DEC FUSE graphical software development environment (see SPD 44.71)
- Support for the common desktop environment (CDE) and its own icon in the Developers' Toolkit group

Over time, Ladebug will replace the dbx debugger.

The dbx Debugger

The dbx debugger is a portable, symbolic source code debugger for customers who desire a debugger with a consistent, industry-standard, command-line user interface. The dbx symbolic debugger will be retired in a future release of Tru64 UNIX by the Compaq Ladebug debugger, a powerful replacement for dbx.

Features

- Supports the C, FORTRAN, and Pascal languages
- Debugs multithreaded applications via kernel threads, does NOT support DECthreads
- Attaches/detaches to a running process and debugs a program (same or multiple executables)
- Performs remote kernel debugging
- Displays a source listing of a program
- Sets breakpoints and tracepoints to stop program execution when certain conditions are met (for example, a program variable value changes)
- Steps through the execution of program one line at a time
- Examines the stack of currently active functions

- Examines and changes program variable and data structure values
- Disassembles and examines machine code and examines machine register values
- Debugs programs with shared libraries
- Customizes the debugging environment by using scripts and aliases for commands and command sequences
- Directly examines memory in various formats
- Displays values in different scopes
- Displays variables
- Debugs the kernel
- Debugs across fork/exec

Program Analysis Tools

The ATOM API can be used to create simple or sophisticated tools. ATOM uses the target application program, an instrumentation file, and an analysis file to create a new executable, that when run collects analysis data for a wide variety of purposes.

Tools are created by writing two sets of C routines: an instrumentation file and an analysis file. In the instrumentation file, the ATOM API is used to specify where calls to analysis routines should be inserted in an application. A single call to an ATOM routine creates a new executable. The analysis routines are called, at run time, by instructions ATOM inserted in the new executable. These routines run in the background of the application and analyze it for attributes such as quality and performance. While ATOM greatly expedites tool creation, it offers vast flexibility in designing the scope of the tool. Compaq has created several tools with ATOM, including hiprof, pixie, and Third Degree.

All these tools support shared libraries and threaded programs and are licensed with the Developers' Toolkit.

The Third Degree tool profiles heap memory for programs written in C and C++. It helps developers identify three memory bugs: memory leaks, reading un-initialized memory, and accessing an invalid memory address.

Pixie is a basic block profiler that allows instruction-level execution counts to be obtained. Developers can then feed this profiling data to the compiler and the code reordering tools.

Hiprof provides procedure call information on the program that is similar to but more powerful than the gprof (cc -pg) profiler. It shows the amount of CPU time utilized by each primary procedure and the subordinate procedures they call.

Graphical Program Analysis Tools

The Graphical Program Analysis Tools (GPA) help application developers examine the run-time behavior of their applications on Tru64 UNIX.

The Graphical Program Analysis Tools can locate problems in the code, explain the problems using diagnostic messages, and even suggest necessary changes through reports. For example, using these tools, developers can:

- Instrument an application, run the application, and look at the memory usage data in one step or separate steps
- Find poorly tested areas in the code
- Locate and correct performance bottlenecks
- Find and fix memory leaks and memory errors
- Find and fix problems with writing past the ends of memory blocks
- Get information about all processes, including child processes, running on local or remote systems
- Invoke utilities and commands to monitor and manage the remote system

- Record all the process information to a file for later review

The Graphical Program Analysis Tools Are:

- **Process Viewer:** Graphically displays performance information about processes and child processes running on specified (local or remote) Tru64 UNIX systems.
- **Memory Profiler:** Graphically displays how an application uses memory over the course of its execution, for example, showing how much is allocated for various purposes and how frequently the application is allocating and deallocating memory. This tool helps developers understand if an application uses memory inefficiently, for example, fragmented memory allocations.
- **Performance Profiler:** Gathers, analyzes, and displays, in graphical form, run-time statistics about an application, such as CPU usage by function or line and test coverage.
- **Heap Analyzer:** Finds and displays in graphical form memory errors and memory leaks in an application.
- **Man Page Browser:** Used for searching, navigating, and printing reference pages (manpages) in a graphical, scrollable hypertext window.

In addition, online help is provided in a hypertext window for each tool.

Developers can use the Process Viewer, Memory Profiler, Performance Profiler, and Heap Analyzer standalone, or use them from a Windows NT PC as part of the Compaq Enterprise Toolkit - UNIX Edition for Tru64 UNIX (SPD 70.03.xx).

For software development, the Graphical Program Analysis Tools require a typical Alpha workstation or server running Common Desktop Environment (CDE) or Motif interface environment software. Developers can run tools directly from the CDE desktop interface or invoke a tool

directly from the command line (for example, pview, mview, dxprof, or dxheap).

The Graphical Program Analysis Tools provide full language support for Compaq C++, Compaq C, and Compaq Fortran (both f77 and Fortran 90).

Visual Threads

Visual Threads lets developers debug and analyze multithreaded applications. It can be used to automatically diagnose common problems associated with multithreading, including deadlock, protection of shared data, and thread usage errors. It can also be used to monitor the thread-related performance of the application, helping to identify bottlenecks or locking granularity problems. It is a unique debugging tool because it can be used to identify problem areas even if an application does not show any specific problem symptoms.

Visual Threads provides the following:

- **Event Collection:** Collects detailed information about the significant thread state changes that occur during the running of a multithreaded application.
- **Event Display:** At any time, developers can view the dynamic display of events as they occur, with options to filter for specific types of events.
- **Automatic Rule-Based Analysis:** Detects violation conditions by evaluating the events against a set of enabled rules. Several predefined rules look for data protection errors, deadlock conditions, and programming errors
- **Rule Customization:** Lets developers use templates to define their own rules to specify criteria for violation conditions, and to specify rules to monitor or identify performance problems.
- **Customized Actions:** Suspends execution of the application when it detects violation conditions. Developers can choose from several options at this point, including invoking the debugger in the appropriate context for the application.
- **Playback:** Records events to a trace file so developers can play back and analyze them later. This is especially useful for analyzing a trace file iteratively, perhaps using different rules

each time as more information is discovered.

- **Statistics:** Provides easy access to object-level statistics and current state information, including use of resources (for example, the mutexes held by a particular thread). Developers can use this information to look for performance problems and to fine tune the application.
- **Performance Analysis:** Provides automatic analysis of lock contention and lock granularity, statistical hotspots, and processor utilization.
- **Profiling:** Profiles lock activity for each lock-related statistic, such as the number of locks and number of contended locks. Graphs are displayed showing the program locations for each type of lock activity statistic.
- **Visualization:** Graphically and continuously displays the state of the program threads and the frequency of thread-related events. Thread snapshots provide historical views of program state. Graphs for each object (threads, mutexes, condition variables, and so on) are provided for each collected statistic.

The Porting Assistant

The Porting Assistant is an integrated graphical toolset designed to reduce the time and cost of porting applications to Tru64 UNIX. It helps when porting applications written in C, C++, and Fortran from other UNIX platforms, such as SunOS, HP-UX, IBM/AIX, and ULTRIX, and from non-UNIX platforms, such as OpenVMS. The Compaq Porting Assistant is a Tru64 UNIX/Motif based toolset.

The Porting Assistant guides the developer through the porting process by suggesting a systematic, iterative porting approach. First, the Porting Assistant analyzes source files and locates changes that are potentially needed for the application to run on Tru64 UNIX. Then, through extensive, graphical hyperlinked information, it helps programmers understand what changes are needed and why they need to make them. Finally, it aids in making each change, either

by using the integrated editor or through its global replace capability.

In analyzing software code, the Porting Assistant locates:

- 32-bit dependencies
- Conditional code that might also need Tru64 UNIX branching
- References to files that do not exist on Tru64 UNIX
- Calls to library functions that do not match Tru64 UNIX definitions or have different semantics on Tru64 UNIX
- Platform-dependent signal handling code
- Function/subroutine calls in Fortran that are inconsistent with their definitions
- Makefile actions that do not exist on Tru64 UNIX
- Important command options and arguments that differ between platforms

To help developers understand each porting problem, the Porting Assistant graphical hyperlinked information:

- Provides porting tips that complement Compaq's extensive porting guides
- Explains the relevance of each of the analysis steps
- Explains the semantic differences in functions on different vendor's platforms
- Provides detailed help on many individual diagnostics
- Provides hyperlinked access to relevant reference pages

When making code changes, the Porting Assistant:

- Provides developers with a choice of vi, emacs, or a Motif editor

- Steps developers through the problem areas one at a time, bringing each into the editor to be examined and changed
- Provides a global search/replace facility to make changes across files
- Filters out unwanted diagnostics
- Suggests what shareable libraries and archives to use
- Reruns analyses at users' request after changes are made
- Provides support for mapping Fortran compile command lines in a makefile to the equivalent Tru64 UNIX command lines
- Provides the addition of a keyword lookup function for the DXML Library, which includes commonly used Fortran math functions
- Provides support for mapping LINPACK and EISPACK routines to LAPACK routines

The Porting Assistant requires a minimum of 64 MB of memory and 10 MB of disk space.

The Porting Assistant supports CDE and has its own icon in the Developers' Toolkit group.

Reordering Tools

The reordering tools rearrange procedures in an executable file to facilitate better cache mapping, reducing the instruction cache miss rates. The tools are as follows:

- `cord` - Rearranges procedures in an executable to facilitate better cache mapping
- `om` - A post-link optimizer that can optimize executables and reorder them

HARDWARE REQUIREMENTS

Please refer to the Tru64 UNIX Operating System V5.0A Software Product Description (SPD 70.70) for specific hardware supported.

SOFTWARE REQUIREMENTS

Tru64 UNIX Operating System Version 5.0A.

GROWTH CONSIDERATIONS

The minimum hardware and software requirements for any future version of this product might be different from the requirements for the current version.

DISTRIBUTION MEDIA

This product is distributed on the Tru64 UNIX Operating System CD-ROMs.

ORDERING INFORMATION

Developers' Toolkit for Tru64 UNIX:

Software Licenses:
Traditional Licenses: QL-MT5A*-AA
Concurrent Use License: QL-MT5AM-3B

Software Media/Documentation:
QA-MT4AA-H8
Software Documentation (hardcopy):
QA-MT5AE-GZ
Software Product Services: QT-MT5A*-**

* Denotes variant fields. For additional information on available licenses, services, and media, refer to the appropriate price book.

SOFTWARE LICENSING

This software is furnished only under a license and is available on a traditional or Concurrent Use basis. For more information about Compaq's licensing terms and

conditions, contact your local Compaq office.

SOFTWARE PRODUCT SERVICES

A variety of service options are available from Compaq. For more information, contact your local Compaq office.

SOFTWARE WARRANTY

This software is provided by Compaq with a 90-day conformance warranty in accordance with the Compaq warranty terms applicable to the license purchase.

The above information is valid at time of release. Please contact your local Compaq office for the most up-to-date information.

© 2000 Compaq Computer Corporation

COMPAQ, the Compaq logo, DEC, and VAX Registered in the U.S. Patent and Trademark Office. OpenVMS and Tru64 are trademarks of Compaq Information Technologies Group, L.P.

Microsoft and Windows NT are trademarks of Microsoft Corporation. Motif, UNIX, and The Open Group are trademarks of The Open Group.

All other product names mentioned herein may be trademarks or registered trademarks of their respective companies.

Confidential computer software. Valid license from Compaq required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Compaq shall not be liable for technical or editorial errors or omissions contained herein.

The information in this publication is subject to change without notice and is provided "as is" without warranty of any kind. The entire risk arising out of the use of this information remains with recipient. In no event shall

Compaq be liable for any direct, consequential, incidental, special, punitive, or other damages whatsoever (including without limitation, damages for loss of business profits, business interruption, or loss of business information), even if Compaq has been advised of the possibility of such damages. The foregoing shall apply regardless of the negligence or other fault of either party and regardless of whether such liability sounds in contract, negligence, tort, or any other theory of legal liability, and notwithstanding any failure of essential purpose of any limited remedy.

The limited warranties for Compaq products are exclusively set forth in the documentation accompanying such products. Nothing herein should be construed as constituting a further or additional warranty.