# HP C/HP-UX Release Notes for HP-UX 10.20

## HP 9000 Computers

### HP C/HP-UX Version A.10.32.20

HEWLETT®
PACKARD

5967-0005

August 1998

# Legal Notice

*Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.* Hewlett-Packard shall not be liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Corporate Offices:

*Hewlett-Packard Co.*
*3000 Hanover St.*
*Palo Alto, CA 94304*

Use, duplication or disclosure by the U.S. Government Department of Defense is subject to restrictions as set forth in paragraph (b)(3)(ii) of the Rights in Technical Data and Software clause in FAR 52.227-7013.

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Use of this manual and flexible disc(s), compact disc(s), or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs in their present form or with alterations, is expressly prohibited.

A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

PostScript is a trademark of Adobe Systems Incorporated.

UNIX is a registered trademark of The Open Group.

# Contents

# Preface

This document summarizes the changes to HP C/HP-UX in versions A.10.32 and A.10.32.20 running on HP-UX 10.20.

Please note that product software releases and operating system releases often occur independently of each other. In general, a product software release applies to the immediately preceding or concurrent system release and to all subsequent system releases until the software is revised.

Also, the product software release number (shown above) indicates the version level of the software product at the time that the release notes were issued. Since some product updates do not require documentation changes, it is possible that there may not be a one-to-one correspondence between the version number on your software and the version number of the release notes.

# 1 New and Changed Features

This chapter summarizes the new and changed features in versions A.10.32.20 and A.10.32 of HP C/HP-UX.

Version A.10.32.20 of HP C/HP-UX, running on HP-UX 10.20, includes performance improvements and defect fixes made since release A.10.32.

Version A.10.32, also running on HP-UX 10.20, includes enhancements to the compiler, optimizer, and online and print documentation made since release A.10.30.

You can see nearly all HP-UX documentation at the HP 9000 Documentation web site at http://docs.hp.com. Programmer documentation is under the category *Development Tools & Distributed Computing*.

# New Debugger Available for A.10.32.20

Release A.10.32.20 of HP C/HP-UX provides performance improvements and defect repairs. In addition, the new debugger HP Wildebeest (WDB) is available free of charge. HP WDB is HP’s new strategic source-level debugger. The first release, HP WDB 0.75, is a fully HP-supported version of the GDB debugger, first developed by the Free Software Foundation. HP WDB 0.75 is bundled with the HP C/HP-UX and HP aC++ compiler kits on HP-UX 11.0, and you can download a 10.20, 10.30, or 11.0 version for no charge. HP provides support only for customers who have purchased an HP C/HP-UX or HP aC++ compiler with support.

You can download the HP WDB debugger from the web at http://www.hp.com/go/languages by selecting HP Debuggers then HP WDB. Follow the instructions to download the product.

# New Features Introduced in A.10.32

This section describes major changes implemented by HP C version A.10.32.

## New Optimization Options

New or changed optimization options to enhance performance are:

- **+Odataprefetch** — to generate data prefetch instructions for data structures referenced within innermost loops.
- **+Oentrysched** — changing to make save and restore operations more efficient.
- **+Ofailsafe** — to enable fail-safe optimization by default.When a compilation fails at the current optimization level, +Ofailsafe will automatically restart the compilation at a lower level of optimization.
- **+Ofltacc** — to provide better performance for PA-RISC 2.0 targets.
- **+Oglobal_ptrs_unique** — to tell the optimizer whether there are unique global pointer variable names, and what the names are.
- **+Oinline_budget** — to perform aggressive inlining.
- **+Oloop_unroll** — to enable loop unrolling.
- **+Optrs_ansi** — to assume that an **int *p** points to an int field of a struct or union, and that **char *** points to any type of object.
- **+Optrs_to_globals** — to tell the optimizer whether global variables are modified through pointers.
- **+Optrs_strongly_typed** — to assume that all pointers are type safe.
- **+Ostatic_prediction** — to enable static branch prediction.
- **+Owhole_program_mode** — to hold global variables in registers longer and delete inlined or cloned global procedures.
- **+ESfic option** — to replace millicode calls with inline code for fast indirect calls.

## Additional Optimization Changes

New or changed optimization options to enhance performance are:

- Enhancement to optimization levels 2, 3, and 4 to enhance performance of programs including macro definitions of **`ABS`**, **`MIN`**, and **`MAX`**.
- New **`+DA`** designations for PA-RISC 2.0 model and processor numbers to generate code for the PA-RISC 2.0 systems. Also a **`+DAportable`** option to generate code compatible across PA-RISC 2.0 and 1.1 workstations and servers. Default architecture object code generation is now determined automatically on all systems as that of the machine on which you compile.
- New **`+DS`** designations for PA-RISC 2.0 model and processor numbers to perform instruction scheduling tuned for PA-RISC 2.0 systems. Default instruction scheduling is now determined automatically on all systems as that of the machine on which you compile.

## Changes to Debugging of Optimized Code

Support for debugging of optimized code at optimization levels 1 and 2 (in conjunction with HP/DDE) now includes:

- 3 location tracking mappings between source lines and object code locations. Single-stepping and breakpoint setting supported for 3 models of location mappings.
- Breakpoints and single-stepping based on Source Statement Order and Statement Completion location mapping models.
- Assembly level debugging incorporating the new Statement Completion model and enhancements to the statement fragment model.
- Tracking updates of user variables in assignment statements using new statement completion UI commands.
- Notification when variable values are unavailable, or the accuracy of the data value is questionable due to optimization.
- Viewing locals, globals, and parameters within procedures, for variables which are active and have not been eliminated due to optimization.

- Notification of out of order updates to user variables using the Source Order model.
- Identical program behavior between programs compiled with and without debug.
- Tracebacks with line-number annotations.
- Setting breakpoints and single-stepping at the source statement level.
- Viewing and modifying global variables and parameters at procedure entry.

When **-g** is used with optimization levels 1 and 2, the **-y** (static analysis) option is not allowed.

Refer to the discussion on advanced-style (not basic-style) debugging of optimized code in the HP/DDE debugger online help for more information on debugging optimized code.

## Miscellaneous Changes to HP C/HP-UX

- HP C/HP-UX now includes a **+help** option to invoke online help for the HP C compiler and linker.
- Note that when code is ported from a PC to a workstation, a carriage return in the PC code is converted to a newline on the workstation. Control M is ignored by **cpp**.
- The compiler is now large file aware and will give you the following message if your source file is larger than 2 GB: `Can't open` *filename*.

**NOTE** At the next major release of the HP C/ANSI C Developer's Bundle, B3899AA, B3898AA, B3901AA, and B3900AA, Hewlett-Packard intends to change the C language default from K&R C to ANSI C. On versions A.10.32.20 and A.10.32 of HP C, K&R C is the default and ANSI C must be selected at compile time with **-Aa** or **-Ae**. When ANSI C becomes the default (on the HP-UX 11.0 version of HP C/HP-UX), there will be an option to specify K&R C. The compiler release notes for that release will provide more information.

**NOTE** At the next major operating system release the C compiler that is bundled with the operating system and is used to regenerate the kernel will change from **cc** to **cc_bundled**.

# 2 Installation Information

Read this entire document, and any other Release Notes or READMEs you may have, before you begin an installation.

Be aware that, if you install all the packages, they occupy approximately 122 megabytes of disk space.

The HP C/ANSI C Developer’s Bundle for HP-UX contains the following packages:

- C-ANSI-C—HP C/ANSI C Compiler (8.5 Mb)
- C-Dev-Tools—C Language Development Tools (1.1 Mb)
- C-Analysis-Tools —C Language Analysis Tools (330 Kb)
- Auxiliary-Opt—Auxiliary Optimizer for HP Languages (7.5 Mb)
- DDE—Distributed Debugging Environment (23 Mb)
- DebugPrg—Debugging Support Tools (146 Kb)
- XDB—HP XDB Debugger (1.1 Mb)
- HPPAK—HP Programmer’s Analysis Kit (4 Mb)
- BLINKLINK—HP Incremental Linking Facility (1.2 Mb)
- X11MotifDevKit—X11 Motif Developer Kit (21.7 Mb)
- VUEHelpDevKit—HP Vue Online Help Developer Kit (2.6 Mb)
- CDEDevKit—CDE Developer Kit (7.8 Mb)
- AudioDevKit—HP Audio Developer Kit (501 Kb)
- ImagingDevKit—HP-UX Developer’s Toolkit - Imaging (2.1 Mb)
- TechPrintSvcDK— (674 Kb)
- GraphicsPEX5DK—Graphics Developer Kit (Series 700 only) (11.3 Mb)
- GraphicsSBaseDK— Graphics Developer Kit (Series 700 only) (14.2 Mb)
- OpenGLDevKit—OpenGL Developer Kit (5.6 Mb)

- OpenGLDebugger—Open GL Debugger (2.1 Mb)
- DigitalVideoDK—Video Developer Kit (6.2 Mb)

After loading your HP-UX 10.20 operating system, you can install your HP C/ANSI C Developer's Bundle. To install your software, run the SD-UX **`swinstall`** command (see *swinstall*(1M)). It will invoke a user interface that will lead you through the installation.

For more information about installation procedures and related issues, refer to *Managing HP-UX Software with SD-UX* and other README, installation, and upgrade documentation provided or described in your HP-UX operating system package.

**NOTE**

Most files related to the HP C compiler are installed in the directories `/opt/ansic` and `/opt/langtools`. The installation scripts add `/opt/ansic/bin` and `/opt/langtools/bin` to the login file `/etc/PATH`. They also add `/opt/ansic/share/man/%L:/opt/ansic/share/man` and `/opt/langtools/share/man/%L:/opt/langtools/share/man` to the login file `/etc/MANPATH`. (`%L` is replaced by the value of the `LC_MESSAGES` environment variable when the **`man`** command is executed. It determines the language used for manpage searches. If `LC_MESSAGES` is not set, `%L` defaults to null. See *environ*(5).)

**NOTE**

The HP C/ANSI C compiler installation package provides the capability to create and remove transition links from HP-UX release 9.*x* locations to HP-UX 10.*x* locations. The HP C/ANSI C product installs the ISU transition link table specification files on the system.

The Software Distribution update tool **`tlinstall`** will use these files to install transition links from HP-UX 9.*x* file and directory names to the corresponding HP-UX 10.*x* file and directory names. To remove these transition links, use the update tool **`tlremove`**. For more detail, read the update tools manpages. These tools are installed in `/opt/upgrade/bin`.

# 3 Documentation Overview

This chapter summarizes the C compiler documentation and various other documents that may be of interest to the C programmer. Some of this documentation is online, and some may be ordered in printed versions, as described below.

The HP C documentation consists of:

- *HP C/HP-UX Release Notes* (this document)
- *HP C/HP-UX Programmer's Guide*
- *HP C/HP-UX Reference Manual*
- *HP-UX Floating Point Guide*
- *HP C/HP-UX Online Help*

The following sections summarize these and other related documents.

# C Compiler Documentation

## HP C/HP-UX Release Notes

The *HP C/HP-UX Release Notes*, which you are reading now, provide release-specific information such as new feature summaries, installation instructions, and known bugs. In addition, they contain this documentation overview to help you orient yourself regarding available documentation. The release notes for version A.10.32 are online in the text file `/opt/ansic/newconfig/RelNotes/ansic.10.32`.

## Printed Documentation

You can order printed versions of a wide variety of Hewlett-Packard documents. Use the **`man manuals`** command for details on which documents are available for ordering as well as ordering information. Listed below are some of the documents most closely related to use of the C Compiler.

- *HP C/HP-UX Reference Manua*l (92453-90024 E0696)

  Provides reference material for HP C as implemented on HP 9000 systems. This document is based on the ANSI C standard 9899-1990, and documents implementations and extensions unique to HP C on HP-UX. It does not replicate the ANSI C standard and you are referred to the standard for any fine points not covered.

- *HP C/HP-UX Programmer's Guide* (92434-90002 E0696)

  Contains a detailed discussion about selected C topics for HP 9000 systems. Included are a discussion of alignment modes, some comparisons between HP C and other languages, and information on optimization.

- *HP-UX Floating-Point Guide* (B3906-90004 E0796)

  Describes how floating-point arithmetic is implemented on HP 9000 systems and discusses how floating-point behavior affects the programmer. Provides information about the C and Fortran math libraries.

## Online Help for HP C and the Linker

The HP C compiler online help and the HP-UX linker and libraries online help include examples, tutorial information, error message discussions, and reference information.

To use the online help, on a system where the HP C/HP-UX compiler is installed enter the following commands:

```
cc +help
```

```
ld +help
```

and in a few moments, the online help window appears on your display. (Note that this requires that your `$DISPLAY` variable is set correctly. If you do not know how to do this, contact your system administrator.)

You can also click on the ? icon on the HP CDE front panel on a system where HP C is installed and select *HP C Online Reference* or *HP-UX Linker and Libraries Online User Guide*.

### Troubleshooting Information

The **`+help`** option may not work on systems running HP CDE. If it does not work, ensure the environment variable `DTHELPSEARCHPATH` is set. (It may not be set if you rlogin to a system, for example.) If it is not set, use the following command to set it:

```
eval $(dtsearchpath)
```

Ensure the `LANG` environment variable is set, typically `LANG=C`.

As a workaround, you can view the C or linker and libraries online help using the ? icon on the HP CDE front panel or by using one of the following commands:

```
/usr/dt/bin/dthelpview -helpVolume linker
```

```
/usr/dt/bin/dthelpview -helpVolume c
```

or

```
/usr/dt/bin/dthelpview -helpVolume\
/opt/langtools/lib/help/C/linker.hv
```

### Character-Mode Help Browser

If you do not have a graphics display or prefer to work with a character-based help interface, enter the command:

```
charhelp
```

and you will get a usage description for this command:

```
charhelp: Usage: charhelp {cc|CC|aCC|f77|ld|dde|
-helpVolume file}
```

For help with the HP C compiler, for example, enter:

```
/opt/langtools/bin/charhelp cc
```

or for help with the linker or libraries, for example, enter:

```
/opt/langtools/bin/charhelp ld
```

and follow the menus for further direction. For more information, see the man page for *charhelp*(1) (`/opt/langtools/share/man/man1.Z` must be in your `$MANPATH` environment variable).

## Related Documentation

Nearly all HP-UX documentation is available on the HP 9000 Documentation web site at http://docs.hp.com. Programmer documentation is under the category *Development Tools & Distributed Computing.*

- *HP PA-RISC Compiler Optimization Technology White Paper*

  Describes the benefits of using optimization. For printed copies of this document, contact your local HP sales office, the Customer Information Center at (800) 752-0900, or HP DIRECT at (800) 637-7740 in the U.S. The white paper is also available online in the PostScript file:

  ```
  /opt/langtools/newconfig/white_papers/optimize.ps
  ```

- *HP-UX Linker and Libraries Online User Guide* (B2355-90653)

  Replaces the manual *Programming on HP-UX*. To access the *HP Linker and Libraries Online User Guid*e, use the **`ld +help`** command. You can also order a hardcopy manual by calling 1-800-227-8164 in the U.S.

- *HP Assembler Reference Manual* (92432-90009)

  Describes the use of the Precision Architecture RISC (PA-RISC) Assembler on HP 9000 computers. Describes PA-RISC Assembler directives, pseudo-operations, and how to run the Assembler on HP-UX.

- *HP-UX Reference Manual*

  The reference manual pages, or man pages, are available online (use the command `man man` for more information), and are also available on the CD-ROM.

- HP-UX Software Transition Kit (STK)

  Enables the application developer to easily transition software from HP-UX 10.*x* to either the 32-bit or the 64-bit version of HP-UX 11.0. The kit is available free of charge on the HP-UX 11.0 Application Release CD-ROM, or via the World-Wide Web at the following URL:

  ```
  http://www.software.hp.com/STK/index.html
  ```

- WDB Debugger documentation

  WDB is the HP-supported implementation of the GDB debugger. Refer to the web page **`http://www.hp.com/go/languages`**, HP Debuggers for more information.

# 4 Problem Descriptions and Fixes

For a list of HP C problems and their fixes, see the *Software Status Bulletin* and the *System Release Bulletin* using the reference number 92453-01A.

Users with support contracts may get these bulletins and patch information from the HP SupportLine database on the World Wide Web. The URL is:

`http://us-support.external.hp.com/`

Note since HP-UX 10.10 is the last supported OS for PA-RISC 1.0 architecture machines, the 10.20 compilers no longer support the compiling of code for PA-RISC 1.0.

## Nonterminating Parallel Processes

A program compiled with **`+Oparallel`** for parallel execution uses shared memory segments and additional processes when it executes. Except in rare circumstances, the program will terminate the processes and free the shared memory whether it completes normally or terminates because of a signal.

However, if enough parallel programs do not terminate their additional processes over a period of time, you may get messages indicating that the operating system refuses to start another process when you attempt to run any program. **`fork()`** returns the error value [EAGAIN] (resource temporarily unavailable), which in this case means that the system's limits on the number of system-wide or individual-user processes would be exceeded.

If this should happen, you can use the **`ps -f`** command to search for the processes that didn't terminate and the **`kill`** command to terminate them (see *ps*(1) and *kill*(1)). Note that the **`ps -f`** command displays the same name for the processes as it does for their associated main programs.

Likewise, if enough parallel programs happen not to free their shared memory segments over a period of time, programs may start failing because of insufficient memory, either shared or normal. Terminating the additional processes will almost certainly free the shared memory segments as well. If not, you can use the **`ipcs -m`** command to search for shared memory segments and the **`ipcrm`** command to remove them (see *ipcs*(1) and *ipcrm*(1)). Note that the **`ipcs -m`** command lists the same owner and group for the shared memory as the associated parallel program had and that most systems normally have active shared memory segments owned by `root` and `daemon`.

# Memory Allocation Routine alloca()

The implementation of this **`alloca()`** routine can be very system dependent and its use is not encouraged.

**`alloca()`** is a memory allocation routine similar to malloc() (see *malloc*(3C)). The syntax is:

```
void *alloca(size_t size);
```

alloca() allocates space from the stack of the caller for a block of at least size bytes, but does not initialize the space. The space is automatically freed when the calling routine exits. This routine is implemented as an inlined function by the HP C compiler product.

To use this function, you may use the <alloca.h> header file or you may specify your own prototype and use the +Olibcalls option.