HP 9000 Series 800 Computer Systems

HEWLETT PACKARD

Software and Communications Technical Data

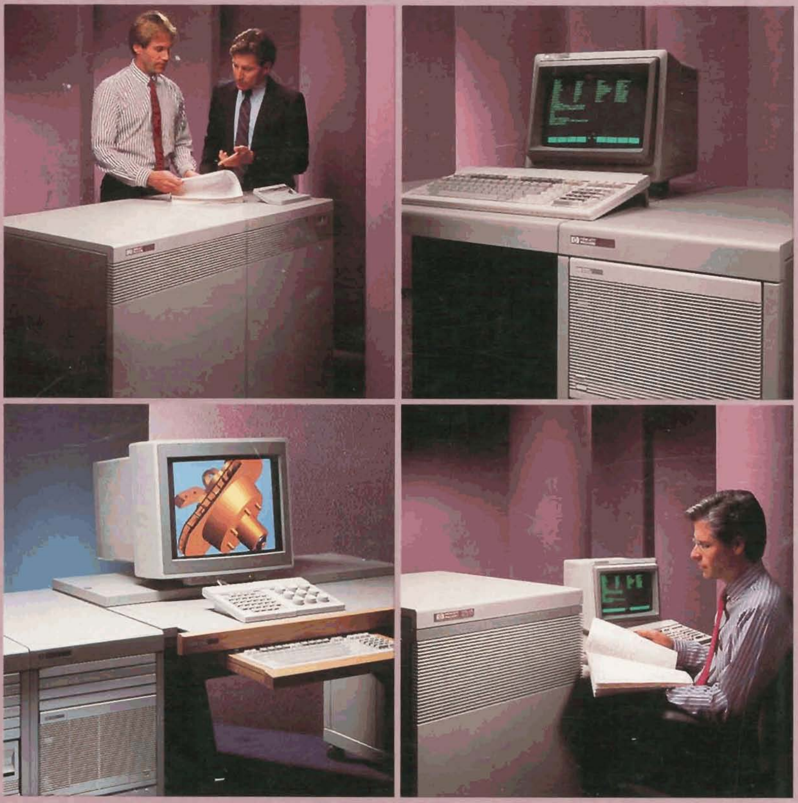


Table of Contents

HP-UX Operating System

HEWLETT PACKARD

For HP 9000 Series 800 Systems

HP-UX is a superset of AT&T's System V Interface Definition (SVID), Issue 2 Volumes 1 & 2, assuring portability of applications developed on other SVID compliant systems. HP-UX thus supports the trend for more and more computer professionals to choose the UNIX® operating system as technology moves towards the goal of universal software compatibility. HP-UX also incorporates features from AT&T UNIX System V, Release 2.0 and from U.C. Berkeley's 4.2 BSD.

Additionally, HP-UX includes a selection of enhancements along with many innovations of its own to extend the capabilities of the UNIX system. Those HP-UX features include: real-time operation, high performance file access, device I/O programming, Native Language Support (NLS), and graphics capabilities.

HP-UX offers three of the most popular languages in the scientific computing world: FORTRAN 77, Pascal, and C. And because all these languages and extensions are linkable at the object level, a programmer is free to mix the computational power of FORTRAN with the structured methodology of Pascal and C to produce optimum software solutions. Programs written in any of these languages can access all HP-UX system calls as well as other libraries.

Hewlett-Packard has included file and record locking and a rich assortment of software engineering tools, such as 2-D and 3-D graphics, symbolic debuggers, and data communications. For complex problems involving several software programs, HP-UX offers the standard ability to connect or "pipeline" separate programs together via a set of simple commands.

HP-UX is an exceptionally powerful, standards-based operating system offering high capacity, high performance, and excellent transportability of applications between systems operating under the UNIX system, including systems from different vendors.

Features

- Compliance with the AT&T System V Interface Definition Issue 2 for compatibility with similarly compliant systems
- Real-time extensions to suit the UNIX operating system for real-time applications as well as time-sharing applications
- Fast and deterministic real-time response to meet the demanding performance needs of real-time applications
- HP 1000 migration through extensive emulation of the RTE EXEC, libraries, file system, and Edit/1000, plus a Migration Analysis Utility
- Choice of 16, 32, 64, or unlimited user license
- Berkeley 4.2 BSD Fast File System for quick access to data and excellent data recovery (omitting features incompatible with System V)
- Over 200 industry standard utilities to assist program development and data manipulation
- C compiler, symbolic debugger, assembler, screen editor, source code control system, 'make' utility, and other tools for support of application development
- Compatibility with HP 9000 Series 200, 300, and 500 systems and the Integral PC
- Native Language Support (Internationalization)
- Device I/O library for programming HP-IB instruments
- Serviceability and system administration enhancements for maximum ease of use and reliability

Support for

- C, FORTRAN 77, and Pascal programming with optimizing compilers
- Starbase Graphics Library software based on evolving ANSI and ISO standards, plus Device-independent Graphics Library (DGL) and Advanced Graphics Package (AGP) software to provide compatibility for existing applications
- HP-GKS Graphics Library which conforms to level 2B of the ANSI/ISO standard
- ALLBASE database management system with network and relational interfaces
- HPtoday for transaction oriented application generation
- AdvanceNet Network Service and LAN software for communication with other HP 9000 Series 800 systems, HP 1000 A-Series systems, and HP 9000 Series 300 and 500 systems
- ARPA/Berkeley Networking Services for multivendor communication
- Localization and use of applications in 16 different native languages

*UNIX is a registered trademark of AT&T in the U.S. and other countries.

Compatibility with Other UNIX Systems

HP-UX is completely compatible with the AT&T System V Interface Definition (SVID) Issue 2. It is thus compatible with other systems that operate under a UNIX system to the extent that the other system complies with the SVID. HP-UX also supports all machine-independent features of AT&T UNIX System V, Release 2.0 and many features of the Berkeley 4.2 BSD version of the UNIX system that are not part of the SVID. These additional supported features, described later in this data sheet, extend the potential compatibility of HP-UX.

Compatibility among HP 9000 Systems

Hewlett-Packard has designated HP-UX as its standard operating system for all HP 9000 systems. The Series 800 systems are the high end of the HP 9000 systems running the HP-UX operating system. The Series 800 is source code compatible with HP 9000 Series 300 and 500 systems running the HP-UX operating system. Most applications from Series 300 or 500 systems will run on the Series 800 by simply recompiling and relinking the programs. However, when non-standard features of the language or operating system have been used, and the compiler or linker cannot resolve the difference, changes will need to be made.

Compatibility with HP 1000 Systems

HP-UX includes a PORT/HP-UX package to help customers move applications from HP 1000 systems operating under RTE-A or RTE-6/VM to the HP-UX environment. In addition to emulation of about 95% of RTE calls on HP-UX, PORT/HP-UX provides a Migration Analysis Utility for analyzing applications to determine which program statements need to be changed and what kinds of changes are needed.

In addition to addressing operating system related migration issues, PORT/HP-UX and the new ALLBASE/HP-UX DBMS provide facilities for migration of IMAGE/1000 and IMAGE/1000-II databases. The DGL and AGP libraries for the Series 800 differ only slightly from the DGL and AGP libraries for HP 1000 systems, which simplifies migration of graphics applications to the HP 9000 Series 800. For more information on HP 1000 applications migration, see the PORT/HP-UX data sheet in this data book.

The Importance of a Standard Operating System

The wide acceptance of the UNIX system can be traced to three features:

1. It is the most important industry standard operating system for multi-user computers.
2. Its excellent software development environment.
3. Its ability to adapt to different applications.

The System V Interface Definition (SVID) specifies the operating system components that are available. The components and their functionality are defined, but no specification is made of their implementation. The definition specifies the application source code interface as well as the runtime behavior seen by an application program. The emphasis is on defining a common computing environment for applications and users, not on the internals of the operating system.

Because the SVID defines a UNIX system, adherence to the SVID guarantees a high degree of compatibility between UNIX systems. This form of standardization simplifies the task of porting applications software between machines to the point where many times all that is required is a recompilation of the applications. Porting is thus a matter of hours instead of the days, weeks, months, or even years required to move applications from one proprietary operating system to another. This represents a move to software that is almost entirely processor independent.

The HP-UX to UNIX System Relationship

The UNIX operating system is a licensed software product. As a licensee of this product, Hewlett-Packard is required to distribute a UNIX System V, Release 2.0 binary license with every HP-UX operating system.

To support source code compatibility with AT&T UNIX System V, Release 2.0, HP-UX includes all hardware-independent System V, Release 2.0 system calls, including support for file and record locking. HP-UX also includes interprocess communication in messages, semaphores, and shared memory calls.

- Messages provide process synchronization and data communication between processes in a much more flexible way than pipes.
- Semaphore calls supply a general process synchronization mechanism. Semaphores are usually used to grant or deny permission to access some resource shared among cooperating programs.
- Shared memory provides the capability for sharing memory among processes.

HP-UX Standard Features

The HP-UX operating system offers a productive environment for those who depend upon computers to solve large, complex problems. Software developers can easily share files and tools without sacrificing system security or reliability.

With HP-UX, programming functionality is increased by several enhancements. Many commands have been added for file manipulation. Tools are also provided to create, modify, compile, assemble, debug, link, and execute programs, as well as to track changes to programs and to tune system operations. FORTRAN 77 and Pascal compilers and the Device-independent Graphics Library (DGL), the Advanced Graphics Package (AGP), Starbase Graphics Library and HP-GKS Graphics Library may be added to enlarge HP-UX program development capabilities.

Virtual Memory Management

HP 9000 Series 800 Systems support a 48-bit address space, which is utilized by HP-UX as multiple 32-bit address spaces that are allocated one per process. The HP Precision Architecture and HP-UX operating system support virtual memory for processes so users can run programs that do not fit into the system's physical memory. In virtual memory, processes can be partially memory-resident and partially disc-resident. HP-UX manages virtual memory automatically and transparently to the user. Processes may have up to 64 megabytes of data, 64 megabytes of text (instructions), 8 megabytes of stack, and multiple shared memory segments up to 64 megabytes each. HP-UX can be configured to support even larger process sizes.

With the Series 800, code may be shared among users on a machine so that only one copy is assigned physical memory space. With demand loading, portions of a program are loaded into memory only when they are accessed.

File System

The HP-UX file system is SVID-compatible. This system is well-suited for a software development environment in which users wish to share programs and data. The hierarchical or "tree-like" structure of the system supports convenient, logical organization of user's files. For each file, permission to read, write, or execute can be assigned on an individual, group, or system-wide basis. This ability to group users, files, and accesses facilitates sharing of resources among project teams.

The HP-UX file system is based on the McKusick/Berkeley implementation of the UNIX file system, which is much faster than a standard AT&T UNIX file system. This file system is better suited for multi-user and real-time applications, which are often dependent on file system throughput.

Interchange of data with other computers is supported by a set of utilities that will selectively convert and copy HP-UX files to Logical Interchange Files (LIFs). The LIF directory format is a vehicle for transporting ASCII files on removable mass storage media between a wide variety of Hewlett-Packard computers. A set of utilities which support access to media formatted on the HP 9000 Series 500 in the SDF file format is also provided.

HP-UX Extensions

Real-Time Functionality

HP-UX includes real-time functionality similar to that of the RTE-A and RTE-6/VM systems that manage real-time operation of HP 1000 Computer Systems. These features equip HP-UX HP 9000 Series 800 systems for computer-integrated manufacturing applications, where the system interfaces with machines, rather than with people as in a time-shared application.

Real-Time Features in HP-UX

- **Real-Time Priorities** can be assigned to processes to give them execution preference over time-shared and lower-priority processes
- **Time-based Scheduling** calls allow processes to manage time with microsecond resolution, and to schedule processes with an accuracy of tens of milliseconds. (System V only supports a 1 second resolution for time and time-based scheduling calls.)
- **Driver Asynchronous** I/O allows a process to read or write from a suitably equipped driver without wait
- **User Control of Buffering** is used by applications that must bypass disc caching for data consistency
- **Reliable Software Signals** support process interaction with asynchronous events, with deterministic results
- **Methods for Controlling Access to Real-Time Privileges,** such as real-time priorities and memory locking.
- **User Control of File System Buffering**
- **Interprocess Communication** is facilitated by sharing large segments of virtual memory among multiple real-time tasks. Semaphores coordinate shared memory, I/O, and other resources. Messages pass short blocks of data between processes. Pipes connect related task operations and signals permit or kill (terminate) task execution.
- **Process Locking** can be used by a program to lock itself, its data, or its shared data into memory. This ensures that critical programs or data are not swapped or paged to disc but remain in memory for best performance.
- **File Locking** makes it possible to give a program exclusive use of a file, thereby assuring its availability.

Real-Time Performance Enhancements in HP-UX

- **Fast, Deterministic Real-Time Response.** HP-UX on the Series 800 has been tuned for fast and deterministic real-time response, especially as measured by process dispatch latency (the time from when a process is ready to run until it is dispatched by the operating system). This measurement has been tuned in HP-UX to be orders of magnitude faster than typical UNIX systems.
- **Disc Queue Prioritization** orders requests in the disc queue according to the real-time priority of the requesting program, speeding completion of real-time processes.
- **The McKusick/Berkeley implementation of the UNIX file system** speeds file system throughput.

FORTRAN 77 Compiler

The FORTRAN 77/HP-UX compiler is a superset of the ANSI X3.9-1978 FORTRAN standard, including MIL-STD-1753 extensions. Further extensions, such as the NAMELIST feature, have been added to track de facto standards. This compiler is designed to facilitate portability of FORTRAN programs from other HP computer systems. It offers a compile-time choice of two levels of program optimization, or no optimization. The result is a modern compiler that supports growth and efficiency for building on a sizable base of existing FORTRAN programs.

Pascal Compiler

HP Pascal is the tool used for design of structured application software. Pascal has gained wide acceptance among software professionals as a block-structured, strong-typed language providing easily read source code. Because it is a simple and concise language, Pascal is recognized as being easy to learn and highly portable. The Pascal/HP-UX compiler conforms to the ANSI/ISO Pascal language standard, but is a superset that is richly enhanced with extensions to support data manipulation, strings, and I/O. HP Pascal/HP-UX offers a compile-time choice of two levels of program optimization or no optimization.

Device I/O Library

The Device I/O Library (DIL), part of HP-UX, provides users access to any device on HP-IB or parallel interfaces. Any program written for HP-UX can use the DIL. The user-programmer can read from, write to, and control via the HP-IB and parallel interfaces to operate instruments or custom peripherals.

Starbase Graphics Library

Starbase/HP-UX is a 2-D and 3-D graphics library. The Starbase Graphics Library provides a high performance interface to Hewlett-Packard graphics input, display, and output peripherals. Starbase supplies procedures to perform basic graphics operations. Output operations include lines, markers, text, and polygons. Control functions, inquiry functions, and echoing are also supported. Starbase supports raster operations, multiple input and output devices, dynamic allocation of devices, and true device independence. The Starbase Graphics Library is based on the ANSI/VD-CGI standard.

DGL/AGP Graphics

The Device Independent Graphics Library (DGL) is a set of graphics tools that supports application program output to, and interaction with, a variety of HP graphics input, display, and output devices.

The Advanced Graphics Package (AGP) builds upon DGL. AGP provides picture segmentation, interactive picking, and 3-D viewing transformations.

DGL and AGP are provided in the DGL/AGP/HP-UX graphics library, along with the Starbase Graphics Library for customers who wish to move existing DGL/AGP graphics applications from an HP 1000 or HP 9000 system to an HP 9000 Series 800 system.

Database Management: ALLBASE

The ALLBASE/HP-UX database management system (DBMS) provides customers with relational and network data access in one DBMS. With HPSQL and HPIMAGE in one DBMS, the user can choose the data model most appropriate to the application on an application-by-application basis. Thus, ALLBASE provides maximum productivity and performance in a single DBMS.

HPtoday

The HPtoday Developer Pack/HP-UX is a software system for computer-assisted development of ALLBASE-related applications. HPtoday applications are developed by filling in blanks on formatted screens instead of coding reams of program instructions. By bridging the gap between conception and the resulting application, HPtoday significantly shortens the time required to develop useful, bug-free applications.

System Networking

HP 9000 Series 800 systems can be connected via available networking services to other HP-UX based HP 9000 systems, to HP 1000 A-Series, and with other computer vendors.

These networking services are provided over the high speed IEEE 802.3 Standard Local Area Network (LAN) link. The hardware and software interface, including the transport facility, are provided in the LAN/9000 Series 800 product. Higher level network services are provided in the NS/9000 Series 800 product.

Together, these products support:

- Network File Transfer (NFT) to other HP-UX based HP 9000 systems, and to HP 1000 A-Series machines.
- Remote File Access (RFA) between HP-UX based HP 9000 systems.
- Network Interprocess Communication (NetIPC) with other HP-UX based HP 9000 Series 800 systems, and with HP 1000 A-Series systems.
- Interprocess Communication (BSD Sockets) with other HP-UX based HP 9000 Series 800 systems.

Data communications are also supported over the LAN using the ARPA/9000 Series 800 product. This product requires the LAN/9000 Series 800 Link to establish connection to the LAN.

The ARPA/9000 Series 800 product permits connection to other HP-UX based HP 9000 Series 300 and 800 systems over an Ethernet LAN only.

The services available via the ARPA/9000 Series 800 product are:

▪ File Transfer Protocol	– ARPA FTP
▪ Simple Mail Transfer Protocol	– ARPA SMTP
▪ Teletype Network	– ARPA TelNet
▪ Remote Copy	– BSD rcp
▪ Remote Login	– BSD rlogin
▪ Remote Shell	– BSD remsh
▪ Remote Execution	– BSD rexec
▪ Remote Who	– BSD rwho
▪ Remote Uptime	– BSD ruptime

Local Area Network Data Communication

HP-UX supports the LAN/9000 Series 800 Link product and the NS/9000 Series 800 Network Services. Together, these two networking products support Remote File Access (RFA) between HP-UX based HP 9000 systems, Network File Transfer (NFT) with HP 1000 A-Series and HP-UX based HP 9000 systems, and Network Interprocess Communication (NetIPC) between HP 9000 Series 800 systems.

HP-UX also supports communication over a Local Area Network (LAN), using ARPA/Berkeley Network Services.* The ARPA/Berkeley Network Services product requires the LAN/9000 Series 800 Link to establish connection to the LAN. (The ARPA/Berkeley Network Services product will be orderable and available by the end of 1986.)

The ARPA/Berkeley Network Services product for HP 9000 Series 800 will support the following services between HP 9000 Series 800 systems communicating over IEEE 802.3 or Ethernet LAN:

1. ARPA FTP (File Transfer Protocol).
2. ARPA SMTP (Simple Mail Transfer Protocol).
3. ARPA Telenet (Teletype Network).
4. Berkeley rcp (remote copy).
5. Berkeley rlogin (remote login).
6. Berkeley rsh (remote shell).
7. Berkeley sendmail.

Asynchronous Data Communication

HP-UX B.1 includes asynchronous multiplexer manager software, which supports CCITT modem communication, BSD job control, block mode communication, and non-blocking I/O. In addition to providing local communication features, the asynchronous multiplexer manager supports HP 9000 Series 800 communication with other UNIX systems via one or more multiplexer channels and hardwired or modem links using the uucp (file transfer) capability of HP-UX. Uucp commands provide file transfer (uucp), remote process execution (uux), and virtual terminal (cu) capabilities.

Native Language Support (NLS)

Localization is the process of adapting a software application or system for use in different countries or local environments. In many cases, a user's native language or data processing requirements may differ dramatically from those in the environment of the software developer.

Native Language Support (NLS) provides the tools for an application designer or programmer to produce applications which are localizable without modifying source code. These tools include software facilities within the operating system and subsystems (libraries and commands), as well as peripheral support. NLS addresses the internal functions of a program, such as sorting, as well as its user interface, including displayed messages, user inputs, special formats (date, currency, etc.).

A major NLS objective is to provide the capabilities or adapting character sets and sequences to local language needs. This takes into account that internal character representation determines the maximum number of distinct characters contained in a set. The default set is the 7-bit ASCII character set; all non-localized programs use the ASCII set. Eight-bit character sets are required for most European languages, including upper and lower case, punctuation, and special symbols. For languages with larger character sets, such as Kanji (the Japanese ideographic character set based on Chinese), character codes greater than 8 bits are required; 2-byte characters are therefore used for support of those languages.

In addition to character support (editing, sorting, shifting, type analysis, etc.), NLS provides for regional and country requirements, such as date and time formatting, decimal number formatting, etc. These local customs and the character handling mentioned above are supported via library routines which are callable by application programmers. In addition, many HP-UX commands have been modified to perform the correct language-dependent functions on user data.

HP-UX supports localization and use of applications in 16 different native languages, as summarized in Table 1. Some of these languages are also supported by keyboard and character set capabilities of printers and plotters, as noted in the comments column.

Table 1. Supported Native Languages

Supported Language	Character Set	Comments
Amer. English Can. French Danish Dutch English (U.K.) Finnish French German Italian Norwegian Portuguese Spanish Swedish	ASCII Roman8 Roman8 Roman8 Roman8 Roman8 Roman8 Roman8 Roman8 Roman8 Roman8 Roman8 Roman8	Supported by HP terminals, printers, and plotters
Greek Turkish	Greek8 Turkish8	Supported by HP terminal
Japanese	Japan15	Katakana is supported by HP printers and plotters. Katakana, Kanji, and Hiragana are supported by a special terminal available from Yokogawa-Hewlett-Packard in Japan

Messages. An Application Message facility provides for storage of application-related messages in tables that are separate from compiled program code. Several sets of messages can be provided, each in a different language. The messages are accessed from programs with very little overhead. Application programs written to communicate in one language can be modified to run in any other language without recompiling or modifying the source.

Flexible Support of Native Languages. Native languages are supported under HP-UX by language data tables. Library routines and messages appropriate to a particular supported language are selected by simply entering the name assigned to that language, prior to initiating a language-switchable application or operation. This approach will make it possible to support new native languages simply by preparing appropriate tables.

Auto Restart After Power Failure

If AC power is lost and restored within 15 minutes and the battery backup system is fully charged, HP-UX is automatically restarted and processing can resume without data loss.

Ordering Information

Each HP 9000 Series 800 system comes standard with an HP-UX operating system. In addition, upgrades to licenses for a greater number of users can be purchased for each. Detailed information for each system and license follows and is summarized in Table 2.

92452A 16-User License HP-UX includes:

1. HP-UX Version B.1, featuring C and the HP-UX Symbolic Debuffer. Also includes Assembler, DIL, Real-Time Package, and PORT/HP-UX. Software Media Option AA0 or AA1 must be ordered.
2. 16-User License.
3. HP 92453K HP-UX Documentation Package, as summarized below.

92453A HP-UX as defined above, but with 32-User license:

Must specify software Media Option AA0 or AA1.

—0A1 Credit for upgrade from 16-user license.
—0B0 Delete 92453K HP-UX Documentation Package.

92454A HP-UX as defined above, but with 64-User license:

Must specify software Media Option AA0 or AA1.

—0A1 Credit for upgrade from 16-user license.
—0A2 Credit for upgrade from 32-user license.
—0B0 Delete 92453K HP-UX Documentation Package.

92455A HP-UX as defined above, but with unlimited user license:

Must specify software Media Option AA0 or AA1.

—0A1 Credit for upgrade from 16-user license.
—0A2 Credit for upgrade from 32-user license.
—0A3 Credit for upgrade from 64-user license.
—0B0 Delete 92453K HP-UX Documentation Package.

Software Media Options

AA0: Software on CS-80 cartridge tape.
AA1: Software on 1600 cpi mag tape.

Table 2. Summary of HP-UX Operating System Licenses HP 9000 Series 800

	Model 825s	**Model 840s**	**Model 850s**
16-user	Included with hardware	Included with hardware	N/A
32-user	92453 #0A1	92453 #0A1	Included with hardware
64-user	92454 #0A1 or 92454 #0A2	92454 #0A1 or 92454 #0A2	92454 #0A2
Unlimited User	N/A	92455 #0A1 or 92455 #0A2 or 92455 #0A3	92455 #0A2 or 92455 #0A3

92453K Documentation Package

- Documentation Guide
- User's Guide
- Text Editors and Processors
- Text Formatters
- Programming Environment
- Device I/O and User Interfacing
- Shells and Miscellaneous Tools
- Advanced Unix Programming
- Asynchronous Serial Communications Programming
- HP-UX Reference Manual
- Real-Time Programming Manual
- HP C Programmer's Guide
- HP C Reference Manual Supplement
- HP C Quick Reference Guide
- HP Symbolic Debugger User's Guide
- HP Symbolic Debugger Quick Reference
- Assembly Language Reference Manual
- Support Tape User's Manual
- HP 9000 Series 800 System Administrator's Manual

92439A/R Native Language I/O and Stick KANJI Font
I/O and font software necessary to support Japanese.

92433A/R Simplex KANJI Font
Enhanced KANJI Font. Requires 92439A/R.

Additional Software

- FORTRAN 77/HP-UX compiler.
- Pascal/HP-UX compiler.
- ALLBASE/HP-UX Relational and IMAGE database management system package with IMAGE/1000 translator.
- DGL/AGP/HP-UX Graphics Libraries with Starbase/HP-UX Graphics Library to support Graphics/1000-II DGL/AGP applications on the HP 9000 Series 800 systems.
- Starbase/HP-UX Graphics Library.
- HP-GKS Graphics Library.
- Development System bundle, including FORTRAN 77 and Pascal compilers and DGL/AGP and Starbase Graphics Libraries.
- HPtoday Developer Pack/HP-UX (includes HPtoday Run-Time Environment).
- HPtoday Run-Time Environment/HP-UX.
- ALLBASE/HP-UX and HPtoday Developer Pack/HP-UX bundle.

Software Support Services*

Hewlett-Packard offers a full range of software support services from personal, on-site assistance to materials-only services. The range of service allows flexibility in choosing the level of HP support to meet changing requirements. Available support services include software update services, notification services, consulting, and on-site or call-in assistance.

Training*

A wide selection of training classes are available, including HP-UX, C, FORTRAN, Pascal, and System Administration.

*Consult your local Hewlett-Packard Sales Office for more details on these services.

SYSTEM CALLS

AT&T

SVID, Issue 2 Base System

_exit
abort**
access
alarm
calloc(malloc)**
chdir
chmod*
chown
clrearerr(ferror)**
close
creat
dup*
execl(exec)
execle(exec)
execlp(exec)
execv(exec)
execve(exec)
execvp(exec)
exit*
fclose**
fcntl*
fdopen(fopen)**
feof(ferror)**
ferror**
fflush(flcose)**
filemo(ferror)**
fopen**
fork
fread**
freopen(fopen)**
fseek**
fstat(stat)
ftell(fseek)**
fwrite(fread)**
getcwd**
getegid(getuid)
geteuid(getuid)

getgid(getuid)
getpgrp(getpid)
getpid
getppid(getpid)
getuid
ioctl*
kill*
link
lockf
lseek
mallinfo(malloc)**
malloc(malloc)**
mallopt(malloc)**
mknod
mount
open*
pause
pclose(popen)**
pipe
popen**
read*
realloc(malloc)**
rewind(fseek)**
setgid(setuid)
setpgrp
setuid
signal*
sleep**
stat
stime
sync
system**
time
times
ulimit
umask
umount
uname*
unlink
ustat*
utime
wait
write*

Kernel Extension

acct
brk*
chroot
ftime

gtty
msctl
msgget
msgrcv(msgop)
nice
plock
profil
ptrace
sbrk(brk)*
semctl
semget
semop
shmat(shmop)
shmctl
shmd(tshmop)
shmget
stty

Non-SVID

brk*
ftime
getty(etty)
sbrk(brk)*
stty

U.C. Berkeley

dup2
fchmod(chmod)
fchown(chown)
fsync
ftruncate(truncate)
getgroups
gethostname
getitimer
getimeofday
mkdir
readv(read)
reboot
rmdir
select*
setgroups
sethostname
setitimer(getitimer)
settimeofday(gettimeofday)
sigblock
sigpause
sigsetmask
sigvector*
swapon[HFS]
truncate
vfork
writev(write)

Hewlett-Packard

getprivgrp
prealloc
rtprio
setprivgrp(getprivgrp)
setresgid(setresguid)
setresguid
sigspace

**Enhanced by Hewlett-Packard*

***SVID System Calls that are implemented as Library Calls*

SUBROUTINES AND LIBRARIES

AT&T

SVID, Issue 2 Base System

_tolower(conv)
_toupper(conv)
abs
asctime(ctime)
atof(srtod)
atoi(strtol)
atol(strtol)
bsearch
calloc(malloc)
clock
crypt
ctime
drand48
encrypt(crypt)
erand48(drand48)
errno(perror)
free(malloc)
frexp
ftw
getenv
getopt
gmtime(ctime)
gsignal(ssignal)
hcreate(hsearch)
hdestroy(hsearch)
hsearch
isalnum(ctype)
isalpha(ctype)
isascii(ctype)
isatt(ttyname)
iscntrl(ctype)
isdigit(ctype)
isgraph(ctype)
islower(ctype)
isprint(ctype)
ispunct(ctype)
isspace(ctype)
isupper(ctype)
isxdigit(ctype)
jrand48(drand48)
lcong48(drand48)
idexp(frexp)
lfind(lsearch)
localtime(ctime)
_longjmp(setjmp)
lrand48(drand48)
lsearch
malloc
memccpy
(memory)
memchr(memory)
memcmp(memory)
memcpy(memory)
memset(memory)
mktmep
modf(frexp)
mrand48(drand48)
nrand48(drand48)
perror
putenv
qsort
rand
realloc(malloc)
seed48(drand48)
setjmp
setkey(crypt)
srand(rand)
srand48(drand48)
ssignal
strcat(string)
strchr(string)
strcmp(string)
strcpy(string)
strcspn(string)

strlen(string)
strncat(string)
strncmp(string)
strncpy(string)
strpbrk(string)
strrchr(string)
strspn(string)
strtod
strtok(string)
strtol
swab
sys_errlist(perror)
sys_nerr(perror)
tdelete(tsearch)
tfind(tsearch)
toascii(conv)
tolower(conv)
toupper(conv)
tsearch
ttyname
twalk(tsearch)
tzet(ctime)

acos(trig)
asin(trig)
atan(trig)
atan2(trig)
ceil(floor)
cos(trig)
cosh(sinh)
erf
erfc(erf)
exp)
fabs(floor)
floor
fmod(floor)
gamma
hypot
j0(bessel)
j1(bessel)
jn(bessel)
log(exp)
log10(exp)

matherr
pow(exp)
sin(trig)
sinh
sqrt(exp)
tan(trig)
tanh(sinh)
y0(bessel)
y1(bessel)
yn(bessel)

ctermid
fgetc(getc)
fgets(gets)
fprintf(printf)
fputc(putc)
fputs(puts)
fscanf(scanf)
getc
getchar(getc)
gets
getw(getc)
printf
putc
putchar(putc)
scanf
setbuf
setvguf(setbuf)
sprintf(printf)

sscanf(scanf)
tempnam(tmpnam)
tmpfile
tmpnam
ungetc
vfprintf(vprintf)
vprintf
vsfprint(vprintf)

assert
calloc(malloc)
free(malloc)
mallinfo(malloc)
malloc
mallopt(malloc)
realloc(malloc)
regcmp
regex(regcmp)

_longjmp(setjmp)
_setjmp(setjmp)
a641
daylight(ctime)
dial
ecvt
edata(end)
end
endgrent(getgrent)
endpwent(getpwent)
endutent(getut)
etext(end)
fcvt(ecvt)
fgetrent(getgrent)
fgetpwent
(getpwent)
ftok(stdipc)
gcvt(ecvt)
getgrent
getgrgrid(getgrent)
getgrnam(getgrent)
getlogin
getpass
getpw
getpwent
getpnam(getpwent)
getpwuid(getpwent)
getutent(getut)
getutid(getut)
getutline(getut)
irand48(drand48)
krand48(drand48)
13tol
164a(a641)
ltol3(l3tol)
monitor
nlist
putpwent
pututline(getut)
setgrent(getgrent)
setpwent(getpwent)
setutent(getut)
timezone(ctime)
ttyslot
tzname(ctime)
undial(dial)
utmpname(getut)
signgam(gamma)
cuserid
curses
logname
sgetl(sputl)
sputl

U.C. Berkeley

closedir(directory)
initgroups
opendir(directory)
readdir(directory)
rewinddir(directory)
seekdir(directory)
telldir(directory)

endfsent(getfsent)
getfsent
getsfile(getfsent)
getfsspec(getfsent)
setfsent(getfsent)
tgetent(termcap)
tgetflag(termcap)
tgetnum(termcap)
tgetstr(termcap)
tgoto(termcap)
tputs(termcap)

Hewlett-Packard

catread
currlangid(langinfo)
datalock
fprintmsg(printmsg)
getmsg
idtolang(langinfo)
langinfo
langtoid(langinfo)
nl_asctime(ctime)
nl_ator(strtod)
nl_ctime(ctime)
nlgcvt(ecvt)
nl_isalnum(nl_ctype)
nl_isalpha(nl_ctype)
nl_isgraph(nl_ctype)
nl_islower(nl_ctype)
nl_isprint(nl_ctype)
nl_ispunct(nl_ctype)
nl_isupper(nl_ctype)
nl_strtod(strtod)
nl_tolower(nl_conv)
nl_toupper(nl_conv)
printmsg
stringmsg(printmsg)
strcmp8(nl_string)
strcmp16(nl_string)
strncmp8(nl_string)
strncmp16(nl_string)

hpib_abort(dil)
hpib_bus_status(dil)
hpib_card_ppoll_resp(dil)
hpib_eoi_ct(dil)
hpib_io(dil)
hpib_pass_ct(dil)
hpib_ppoll(dil)
hpib_ppoll_resp_ct(dil)
hpib_ren_ctl(dil)
hpib_rqst_srvce(dil)
hpib_send_cmnd(dil)
hpib_spoll(dil)
hpib_status_wait(dil)
hpib_wait_on_ppoll(dil)
io_eol_ctl(dil)
io_get_term_reason(dil)
io_in_interrupt(dil)
io_reset(dil)
io_speed_ctl(dil)
io_timeout_ctl(dil)
io_width_ctl(dil)
getfstype(getfsent)
catgetmsg
catgets
nl_tools_16
nl_strcmp(string)
nl_strncmp (string)
nl_printf
nl_sprintf(printf)
nl_fprintf
nl_scanf
nl_fscanf(scanf)
nl_sscanf
nl_init
catopen

UTILITIES AND COMMANDS

AT&T
SVID, Issue 2 Base System

accept
acctcms
acctcom
acctconl(acctcon)
acctcon2(acctcom)
acctdisk(acct)
acctdusg(acct)
acctmerge
accton(acct)
acctprc1(acctprc)
acctprc2(acctprc)
acctwtmp(acct)
adb
admin
ar
as
asa
at
awk
banner
basename
batch(at)
bc
bcheckrc(brc)
bdiff
bfs
brc
bs
cal
calendar
cancel(lp)
cat
cb
cc
cd
cdc
cflow
chargefee(acctsh)
checkmm(mm)
chgrp(chown)
chmod
chown
chroot
ckpacct(acctsh)
clri
cmp
col
comb
comm
cp
cpio
cpp
cpset
cron
crontab
crypt
csplit
cu
cut
cxref
date

dc
dd
delta
devnm
df
diff
diff3
diffmk
dircmp
dirname(basename)
isable(enable)
diskusg
dodisk(acctsh)
du
echo
ed
edit
egrep(grep)
enable
env
ex
expr
f77
factor
false(true)
fgrep(grep)
file
find
fwtemp
get
getopt
getty
grep
grpck(pwck)
hashcheck(spell)
hashmake(spell)
help
hp
hyphen
id*
init
install
ipcrm
ipcs
join
kill
killall
lastlogin(acctsh)
ld
lex
line
link
lint
ln(cp)
login
logname
lorder
lp
lpadmin
lpmove(lpsched)
lpsched

lpshut(lpsched)
lpstat
ls
m4
machid
mail
mailx
make
makekey
man
mesg
mkdir
mknod
mm
monacct(acctsh)
mv(cp)
mvdir
ncheck
neqn
newform
newgrp
news
nice
nl
nm
nm3
nohup
nuladm(acctsh)
od
osdd(mm)
pack
passwd
paste
pcat(pack)
pg
powerfail(brc)
pr
prctmp(acctsh)
prdaily(acctsh)
primes(factor)
prof
prs
prtacct(acctsh)
ps
pwck
pwd
ratfor
rc(brc)
red(ed)
reject(accept)
rm
rmail(mail)
rmdel
rmdir(rm)
rsh(sh)
runacct(acctsh)
sact
sccsdiff
sdiff
sed
setmnt

sh
shutacct(acctsh)
shutdown
size
sleep
sort
spell
spellin(spell)
split
startup(acctsh)
strip
stty
su
sum
tabs
tail
tar
tbl
tee
telinit(init)
test
tic
time
touch
tput
tr
true
tsort
tty
turnacct(acctsh)
unmask
uname
unget
uniq
units
unlink(link)
unpack(pack)
uucico
uuclean
uucp
uulog(uucp)
uuname(uucp)
uupick(uuto)
uustat
uusub
uuto
uux
uuxqt
val
vedit(vi)
vi
view(vi)
wait
wall
wc
what
who
whodo
write
wtmpfix(fwtmp)
xargs
yacc

U.C. Berkeley

catman
chsh
clear
csh
ctags
deroff
diffh(diff)
dmesg
expand
fold
from
fsck[HFS]*
fsdb[HFS]*
groups
head
hostname
iostat
l(ls)
last
lastb(last)
leave
ll(ls)
lock
lsf(ls)
lsr(ls)
lsx(ls)
mkfs[HFS]*
mkstr
more
mount[HFS]*
mt
newfs[HFS]
page(more)
prmail
ptx
reset(tset)
rev
savecore
strings
swapon[HFS]
sync
tset
tunefs[HFS]
ul
umount(mount)[HFS]
unexpand(expand)
uusnap
vmstat
whereis
which
whoami

Hewlett-Packard

adjust
fc
findmsg
findstr
fixman
fsclean
gencat
getprivgrp
getx25
insertmsg
lifcp
lifinit
lifls
lifrename
lifrm
lsdev
mkdev
mklp
opx25
pc(hp)
prealloc
rmnl
rtprio
setprivgrp
ssp
uuls
vis
xd(od)

Enhanced by Hewlett-Packard

UTILITIES AND COMMANDS

AT&T

SVID, Issue 2 Base System

BASIC UTILITIES EXTENSION

ar
awk
banner
basename
cal
calendar
cat
cd
chmod
cmp
col
comm
cp
cpio
cut
date
df
diff
dirname(basename)
du
echo
ed
expr
false(true)
file
find
grep
kill
line
ln(cp)
ls
mail
mkdir
mv(cp)
nl
nohup
pack
paste
pcat(pack)
pg
pr
ps
pwd
red(ed)
rm
rmail(mail)
rmdir(rm)
rsh(sh)
sed
sh
sleep
sort
spell
split
sum
tail
tee
test
touch
tr
true
umask
uname
uniq
unpack(pack)
wait
wc

ADVANCED UTILITIES EXTENSION

at
batch(at)
cancel(lp)
chgrp(chown)
chown
cron
crontab
csplit
dd
dircmp
egrep(grep)
ex
fgrep(grep)
id
join
logname
lp
lpstat
mailx
mesg
newgrp
news
od
passwd
stty
su
tabs
tar
tty
vi
wall
who
write

ADMINISTERED SYSTEMS EXTENSION

acctcms
acctcom
acctcon1(acctcon)
acctcon2(acctcon)
acctdisk(acct)
acctmerg
accton(acct)
acctprc1(acctprc)
acctprc2(acctprc)
acctwtmp(acct)
chargefee(acctsh)
ckpacct(acctsh)
clri
devnm
diskusg
dodisk(acctsh)
fusers
fwtmp
grpck(pwck)
init
ipcrm
ipcs
killall
lastlogin(acctsh)
link
mknod
monacct(acctsh)
mvdir
ncheck
nice
prctmp(acctsh)
prdaily(acctsh)
prtacct(acctsh)
pwck
runacct
setmnt
shutacct(acctsh)
startup(acctsh)
sync
turnacct(acctsh)
unlink(link)
vipw
whodo
wtmpfix(fwtmp)

SOFTWARE DEVELOPMENT EXTENSION

admin
as
cc
cflow
chroot
cpp
cxref
delta
env
get
ld
lex
lint
lorder
m4
make
nm
prof
prs
rmdel
sact
size
strip
time
tsort
unget
val
what
xargs
yacc

NON-SVID

accept
acctdusg(acct)
adb
asa
bc
bcheckrc(brc)
bdiff
bfs
brc
bs
cb
cdc
chroot+
comb
cpset
crypt
dc
diff3
disable(enable)
edit
enable
f77
factor
getopt
getty
hashcheck(spell)
hashmake(spell)
help
hp
install
login
lpadmin
lpmove(lpsched)
lpsched
lpshut(lpsched)
machid
makekey
man
newform
nulladm(acctsh)
powerfail(brc)
primes(factor)
ratfor
rc(brc)
reject(accept)
runacct(acctsh)
sccsdiff
sdiff
shutdown
slp
spellin(spell)
tcio
telinit(init)
units
uucico
uuclean
uusub
uuxqt
vc
vedit(vi)
view(vi)

U.C. Berkeley

catman
chsh
clear
csh
ctags
deroff
diffh(diff)
dmesg
dump
expand
fold
from
fsck
fsdb
groups
head
hostname
l(ls)
last
lastb(last)
lastcomm
leave
ll(ls)
lock
lsf(ls)
lsr(ls)
lsx(ls)
mklost+found
mkfs
mkstr
more
mount
mt
newfs
page(more)
prmail
ptx
reboot
reset(tset)
restore
rev
savecore
script
strings
swapon
tset
tunefs
ul
umount(mount)
unexpand(expand)
uptime
users
uusnap
vmstat
whereis
which
whoami

PUBLIC DOMAIN

cdb
fdb(cdb)
pathalias
pdb(cdb)

UTILITIES AND COMMANDS

HEWLETT-PACKARD

adjust
clrsvc
dumpmsg(findmsg)
fc(f77)
findmsg
findstr
fixman
fsclean
gencat
getprivgrp
getx25
insertmsg
inv(vis)
iostat
lifcp
lifinit
lifls
lifrename
lifrm
mediainit
mklp
nlio
nlioenv
nlioinit
opx25
pc
prealloc
rmnl
rtprio
sdfchgrp(sdfchown)
sdfchmod
sdfchown
sdfcp
sdfdf
sdffind
sdffsck
sdffsdb
sdfll(sdfls)
sdfln(sdfcp)
sdfls
sdfmkdir
sdfmv(sdfcp)
sdfrm
sdfrmdir(sdfrm)
setprivgrp
ssp
syncer
untic
uuls
vis
wdedit
wdutil
xd(od)

X Window System and User Interface Toolbox

For HP 9000 Series 800 Computer Systems

A Windowing System for Everybody

The X Window System and User Interface Toolbox provide the HP 9000 Series 800 and 300 computers with a full function windowing environment and a complete set of development tools.

Through the X Window System, the user can easily access multiple applications simulataneously on a single display by dividing the display into several overlapping text and graphics windows. The windows can be manipulated in various ways, including moving, sizing, hiding, and iconizing. For example, an application that outputs text could be running in a "terminal emulator" window at the same time that a graphics application is drawing in a graphics type window. (See Figure 1.)

Hewlett-Packard has complemented the industry standard X Window System (X) with a powerful set of tools that ease the creation of sophisticated user interfaces for the application programmer. The tools, called XrLIB, allow programmers to easily utilize tools like scroll bars, buttons, text editors, pop-up menus, panels, message boxes and more to create easy-to-use application programs.

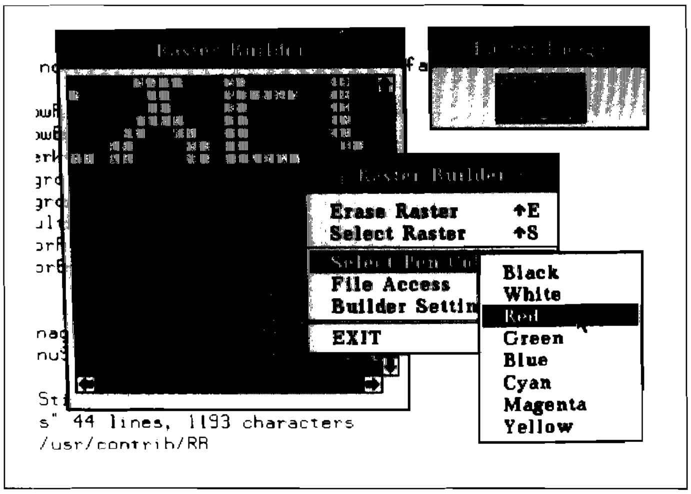
Together, X and the XrLIB provide a comprehensive user and development environment for the HP-UX Series 800 and 300 computer families.

Emerging Standard for Multi-vendor Computing Environments

HP's commitment to standards makes the X Window System a logical choice. X is currently supported as a future standard by almost all of the engineering workstation industry and in the future promises to achieve even broader support in areas such as commercial applications.

The X Window System, developed at the Massachusetts Institute of Technology, remains in the public domain, thus allowing the entire industry to contribute to the evolution of the system. This assures that X will remain a cooperative system and further enhance the engineering workstation marketplace by providing an accessible, stable, standard foundation upon which to build application programs. This ensures the viability of any software investment made with X.

Hewlett-Packard has also shown its commitment to the X standard with the XrLIB user interface tools. HP is participating in the further development of such tools for the X system. And in keeping with the public domain nature of X, HP has contributed the XrLIB tools to the public domain.



```
s" 44 lines, 1193 characters
/usr/contrib/RR
```

Figure 1. The X Window System

Because of the broad range of vendor support, an unprecedented amount of software will be available for X. Software vendors will be able to write the user interface, input and graphics code and have it work the same on nearly any engineering workstation.

Windows for a New Engineering Environment

With the growth in popularity of networked clusters of workstations and an every-increasing range of hardware available for workstations, it is important that the windowing system used be compatible. The X Window System was developed with an entirely new windowing architecture that makes it fully compatible with networks of workstations. Obviously programs can be run on a single workstation and displayed on that same workstation. But, programs can also be run on one workstation and displayed on another, even if the workstations come from different vendors. Or, computationally intensive applications can be run on a mini or mainframe computer without ever changing the user interface — from the same workstation where you do all of your other computing chores. Network compatibility, coupled with the broad range of vendor support, make these things possible today with the X Window System. Figure 2 illustrates network compatibility.

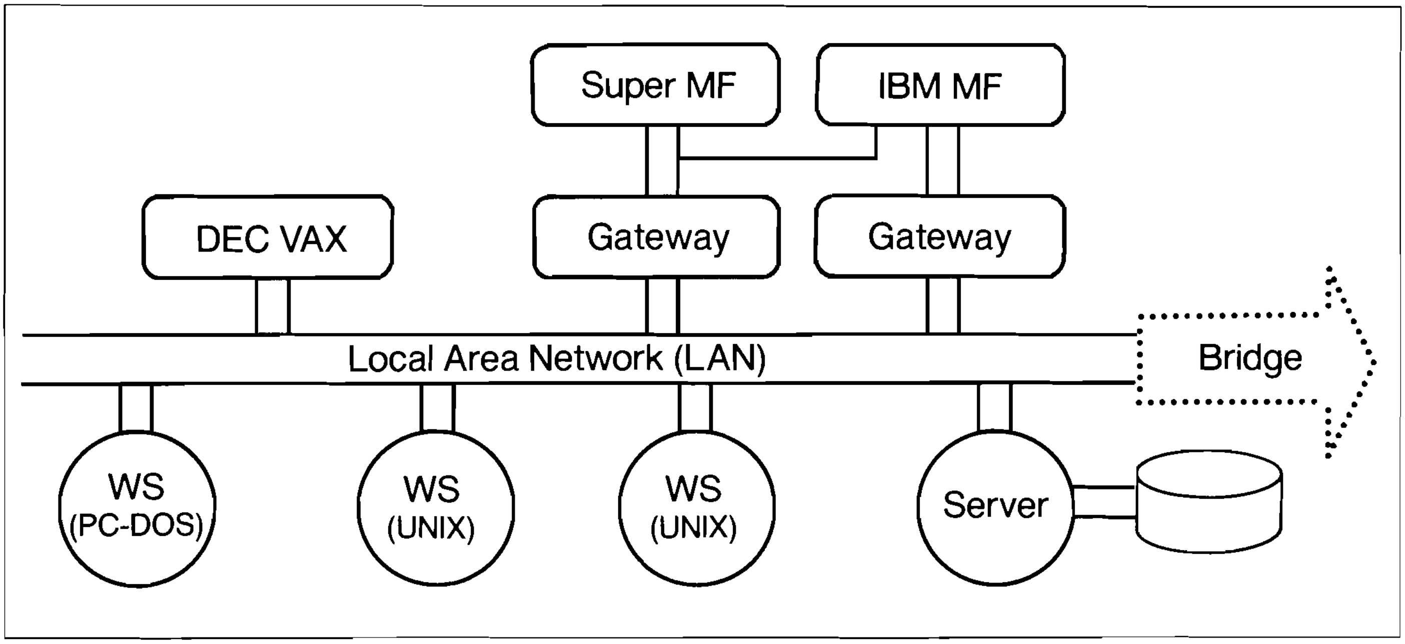


Figure 2. Network Compatibility Model

The new architecture used by X offers another advantage, hardware independence. Often when a computer maker introduces a new display, all of the software written to take advantage of that display has to be altered to support it. Not with X. In the X architecture, all display manipulation is done by a single program (called a "server"). This program receives requests from all of the application programs, (such as drawing lines, placing characters, displaying graphics, etc.), which use the display and carries them out. Thus, entire networks of workstations with diverse display hardware can use the same executable application code. As new display hardware is added, only a new server that understands X is needed for the new hardware to use the same application code. Figure 3 illustrates how X insulates the application from the hardware.

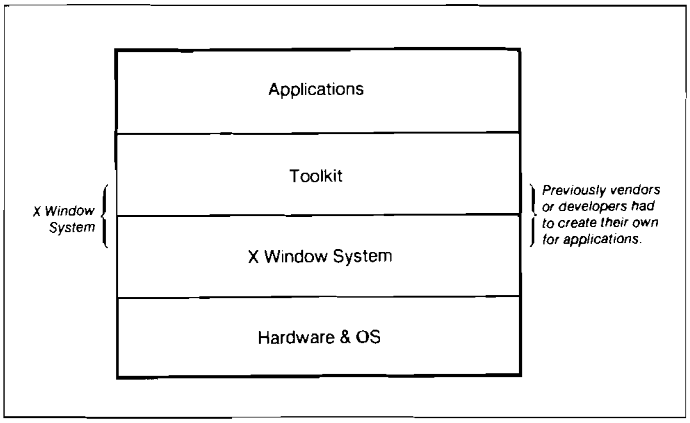

Applications
Toolkit
X Window System
Hardware & OS
X Window System
Previously vendors or developers had to create their own for applications.

Figure 3. The X Architecture

User Interface Toolbox

Hewlett-Packard has added a set of high-level development tools, called XrLIB, to the X Window System. These tools allow programmers to easily and quickly develop sophisticated user interfaces for their application. This provides more time to experiment with different user interface models, and reduces time to market.

A library of routines, XrLIB provides a number of tools like scroll bars, push buttons, text editors, etc., that the programmer can group together into logical groupings called "panels." A full cascading, pop-up menu system is also provided. With these tools, the application writer simply constructs the user interface that he/she desires from the tools. Some examples of toolbox uses follow.

User Interface Toolbox Features

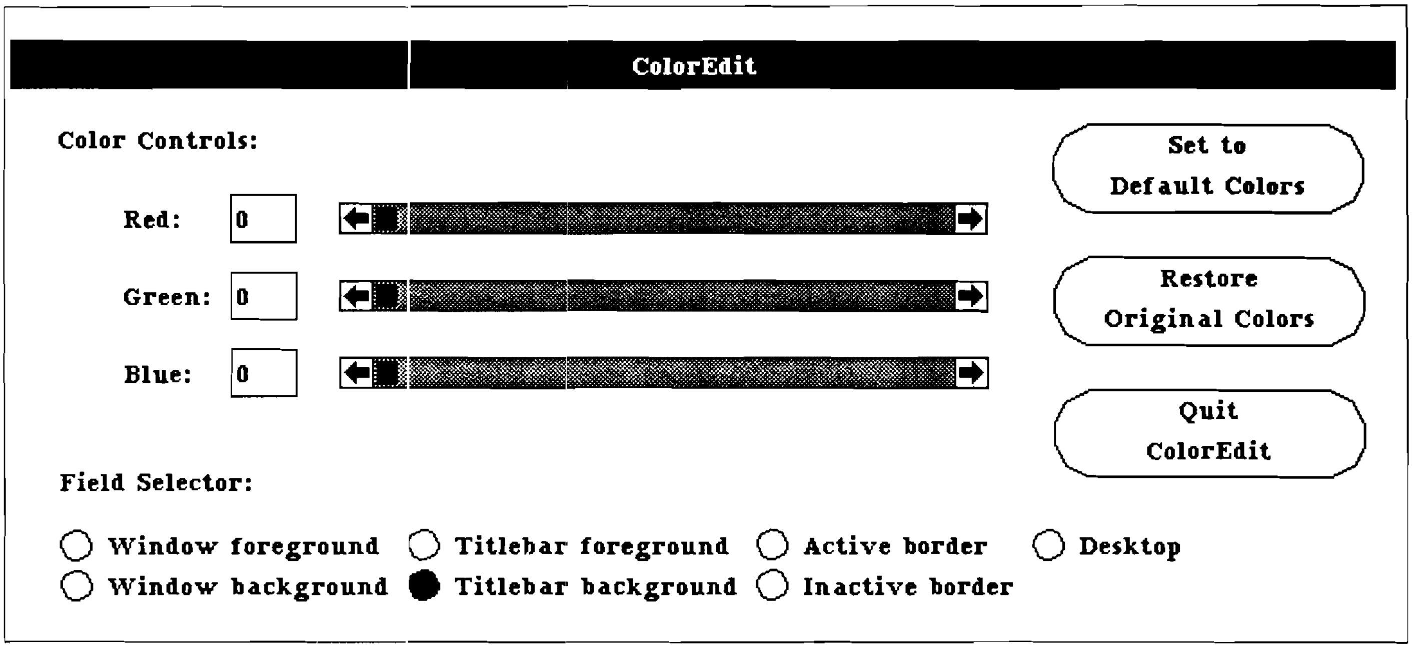
Field editors

Field editors are "gadgets" that allow the user to easily set options and cause action in the application. The type of field editor used to set a particular option is dependent on the option. For example, when setting text attributes, the user may wish to set both bold and underlining. In that case, the field editor will have to allow either or both of the options to be set. This type of field editor is called a "check box editor." Only one font can be selected at a time. The font selection boxes must be mutually exclusive. This type of field editor is called a "radio button." In all, there are 11 predefined field editors that come with the HP User Interface Toolbox:

Titlebar
Scrollbar
Checkbox
Radio button
Pushbutton
Text edit
Raster
Static raster
Static text
Raster select
Page

Figure 4 illustrates a field editor.

UNIX® is a registered trademark of AT&T in the U.S. and other countries.



Red: 0
Green: 0
Blue: 0

○ Window foreground ○ Titlebar foreground ○ Active border ○ Desktop
○ Window background ● Titlebar background ○ Inactive border

Figure 4. Field Editor

Menus

A powerful pop-up menu allows for the easy creation of cascading, pop-up menus that an application can use. Figure 5 shows an example of a pop-up menu.

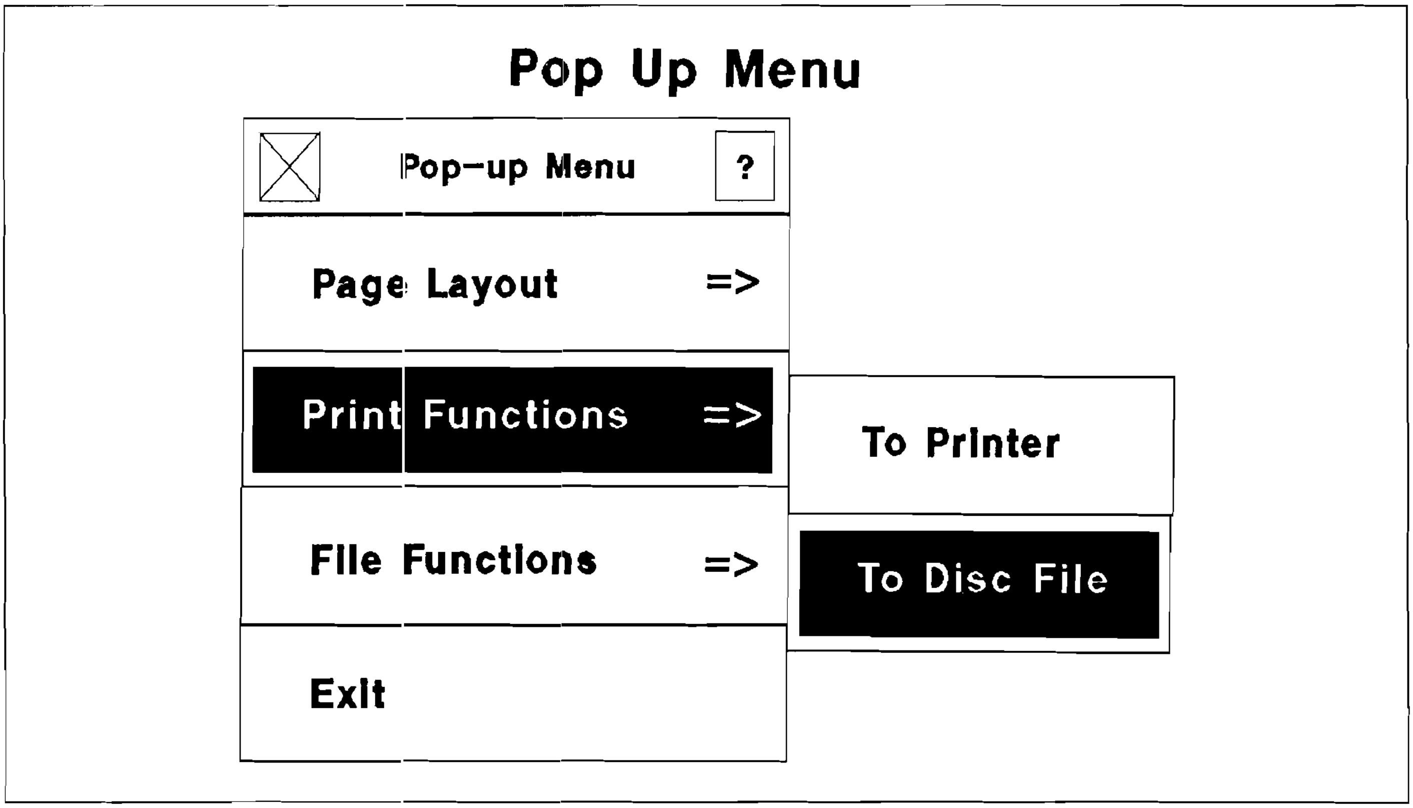
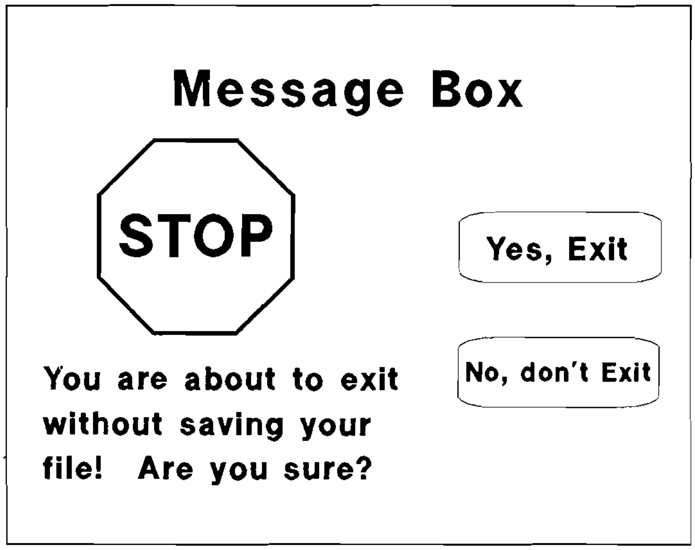


Figure 5. Menu Example

Message Boxes

The message box is a tool used to inform the user of some consequence of an action that was requested, or to keep the user informed about actions that are occurring. The classic example is exiting a program before saving a file. A message box like the one in the following illustration would be used to inform the user of the result.

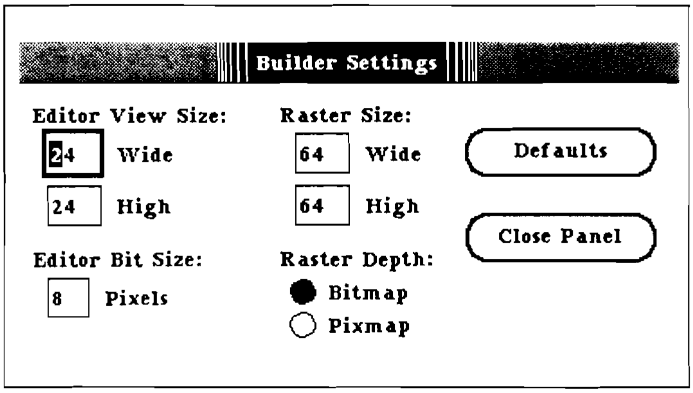


You are about to exit without saving your file! Are you sure?

Figure 6. Message Box Example

Panels

A panel is a collection of field editors displayed in a window. The collection of field editors illustrated here is a panel. The collection forms a single functional unit that allows the user to easily choose all of the options for the application.



Editor View Size: 24 Wide, 24 High

Editor Bit Size: 8 Pixels

Raster Size: 64 Wide, 64 High

Raster Depth: ● Bitmap ○ Pixmap

Figure 7. Panel Example

X Window System Commands

bitmap	the only official way to create bitmaps for use with X applications
hpterm	hp term0 terminal emulator
keycomp	the means by which a keyboard map may be built to customize keyboard actions
pikapix	useful tool to determine what colors should be used by a window
resize	a companion to xterm that alters the environment of a shell to conform to a resized window (does not work with hpterm)
uwm	ultrix window manager (best of the window managers provided)
xclock	displays a digital or analog clock in a window
xfd	displays fonts
xhost	means of restricting or opening access to a system's display
xinit	brings up the X system in a useful state
xload	graphically displays the system load average
xnwm	an alternative window manager
xrefresh	repaints all the windows displayed
xset	sets various preference options for the display
xsetroot	a means of changing the color or tile of the root window
xshell	a way of quickly issuing commands with one keystroke
xterm	vt100/tek 4010 terminal emulator
xuwd	makes a window with information from a file created by xwd
xwd	dumps the image of a window to a specially formatted file
xwininfo	debugging tool by which information about a window can be displayed
xwm	another alternative window manager

X Library Calls

XAddHost	adds a host to the list of hosts able to access a display
XAppendVertex	appends vertices to the output buffer
XAutoRepeatOff	turns off keyboard auto-repeat
XAutoRepeatOn	turns on keyboard auto-repeat
XBeep	activates the beeper
XBitmapBitsPut	performs a function in a region of the window using a pixmap defined by a bitmap and a pair of source pixels defining foreground and background pixel values
XChangeBackground	changes the background tile of a window
XChangeBorder	changes the border tile of a window
XChangeWindow	changes the size of a window
XCharBitmap	requests a bitmap from a specified character of a specified font
XCharWidths	determines the width of each character in a string
XCircWindowDown	lowers the highest mapped child window that is obscuring the view of another child window
XCircWindowUp	raises the lowest mapped child of a window until no part of it is obscured by other child windows
XClear	clears a window and repaints it with the background
XClearIconWindow	clears the icon window for a window
XClearVertexFlag	clears the flag marking if it is safe to append vertices
XClipClipped	sets clipping mode to take child windows into account
XClipDrawThrough	sets clipping mode to ignore child windows
XCloseDisplay	closes a connection with a display server
XCloseFont	closes off any use of a font
XCompressEvents	ignore all but the last mouse movement event
XCondWarpMouse	moves the mouse to a specified area only if the mouse is over a visible portion of a given window
XConfigureWindow	changes the size and location of a window
XCopyArea	copies an area of a window from one place to another within the window
XCreate	creates a window and provides automatic and manual window placement functions
XCreateAssocTable	creates an XAssocTable
XCreateCursor	creates a cursor
XCreateTerm	creates a window, provides window placement facilities, and provides facilities commonly used for text placement
XCreateTransparencies	takes an array of window information and creates several transparent, unmapped windows with one call to the window system
XCreateTransparency	creates an unmapped, transparent subwindow of a specified parent window
XCreateWindow	creates an unmapped, opaque subwindow of a specified parent window
XCreateWindowBatch	takes an array of window information and creates several opaque, unmapped windows with one call to the window system
XDefineCursor	assigns a cursor to a window for use
XDeleteAssoc	delete an item in an XAssocTable
XDestroyAssocTable	destroys an XAssocTable
XDestroySubwindows	destroys all subwindows of a given window
XDestroyWindow	destroys a given window and all of its subwindows
XDraw	draws an arbitrary polygon or curve using a specified function
XDrawDashed	draws a polygon or curve with a dashed line
XDrawFilled	draws a polygon or curve and fills it with a given pixel value
XDrawPatterned	draws a polygon or curve with a specified pattern
XDrawTiled	draws a polygon or curve and fills it with a given tile
XExpandEvents	make all mouse events available
XFetchBuffer	retrieves an arbitrary string of bytes from a particular cutbuffer
XFetchBytes	retrieves an arbitrary string of bytes from a cutbuffer
XFetchName	gets the name of a window
XFlush	flush out all output events buffered

XFocusKeyboard	attach the keyboard to a particular window
XFontWidths	determines the width of each character in a font
XFreeBitmap	frees all storage associated with a bitmap
XFreeColors	frees allocated colors
XFreeCursor	frees all storage associated with a cursor
XFreeFont	tells the server that a font is no longer needed
XFreePixmap	frees all storage associated with a pixmap
XGeometry	performs the XParseGeometry function and aids in window placement using information from the XParseGeometry call
XGetColor	looks up a named color in a color data base to determine its RGB components
XGetColorCells	allocates color cells
XGetDefault	gets default options for a program from the file, .XDefaults
XGetFont	asks the server to load a font
XGetHardxwareColor	defines and gets a color from the display server
XGetHosts	returns a list of the hosts currently allowed to access a display
XGetResizeHint	gets a resize hint for a window
XGrabButton	grab control of a mouse button
XGrabMouse	grab control of the mouse
XGrabServer	grab control of the server
XInterruptLocator	converts absolute coordinates to window relative coordinates
XLine	draws a line in a window
XLockToggle	forces key pressed events NOT to be sent with the shift lock key
XLockUpDown	forces key pressed events to be sent with the shift lock key
XLookUpAssoc	look up an item in an XAssocTable
XLookUpMapping	match a keyboard event to a predefined character string
XLowerWindow	lowers a window so that no sibling windows are covered by it
XMakeAssoc	insert an item into an XAssocTable
XMakePattern	makes a pattern for drawing purposes
XMakePixmap	returns a pixmap constructed from a bitmap
XMakeTile	returns a pixmap suitable for use as a tiling argument
XMapSubwindows	maps all subwindows of a given window
XMapWindow	maps a window and raises the window and all of its subwindows to the top of the stack of windows
XMaskEvent	looks for a particular event in the input queue
XMouseControl	defines how the mouse moves
XMoveArea	moves an area of a window from one place to another within the window
XMoveWindow	moves and raises a window
XNextEvent	flush the output buffer and get the next input event for processing
XOpenDisplay	opens a connection to a display server
XOpenFont	combines several font functions to make a font available for use
XParseColor	takes an alphanumeric representation of a color and returns the numerical RGB values for that color
XParseGeometry	parses a standard geometry parameter of a client
XPeekEvent	flush the output buffer and examine the first element of the input queue without removing it
XPending	flush the output buffer and return a count of unprocessed input events
XPixFill	fills a given region with a given clipping mask
XPixmapBitsPutXY	copies client supplied bits into a window according to a specified display function in the format specified by the name of a procedure in XY format
XPixmapBitsPutZ	copies client supplied bits into a window according to a specified display function in the format specified by the name of a procedure in Z format
XPixmapGetXY	returns a pixmap in XY format into a specified area of memory
XPixmapGetZ	returns a pixmap in Z format into a specified area of memory
XPixmapPut	performs a display function on a specified area of a pixmap to a specified area of the screen
XPixmapSave	creates a pixmap from the given portion of the window
XPixSet	sets a given area in all planes to a given pixel with no clipping mask

XPutBackEvent	push an input event back on the head of the input queue
XQueryBrushShape	returns closest brush shape actually supported by the hardware
XQueryColor	returns the RGB color values for a given pixel
XQueryColors	returns the RGB color values for given pixels
XQueryCursorShape	finds out what size cursors are supported by a particular display
XQueryFont	gets various facts about a font
XQueryMouse	determines the current mouse coordinates
XQueryMouseButtons	returns the coordinates of the mouse and the state of the mouse buttons
XQueryTileShape	returns the closest shape actually supported by the display hardware for tiling
XQueryTree	returns a list of children of a window, its parent, and the number of children it has
XQueryWidth	returns the width in pixels of a string in a given font
XQueryWindow	gets various facts about a window
XRaiseWindow	raises a window so that no sibling window covers it
XReadBitmapFile	reads a file produced by the Bitmap(1) utility
XRebindCode	change the binding of the keyboard keycodes for use by XLookupMapping
XRemoveHost	removes a host to the list of hosts able to access a display
XRotateBuffers	rotates the cut buffers
XSelectInput	defines which vents a window is interested in
XSetDisplay	sets the current display connection
XSetIcon Window	establishes an icon window for a window
XSetResizeHint	sets a hint about resizing a window that can be retrieved by a window manager
XStoreBitmap	creates a bitmap
XStoreBuffer	stores an arbitrary string of bytes in a particular cutbuffer
XStoreBytes	stores an arbitrary string of bytes in a cutbuffer
XStoreColor	changes the color allocated to a pixel
XStoreColors	changes the colors allocated to an array of pixels
XStoreCursor	stores a cursor in the window system
XStoreName	assigns a name to a window
XStorePixmapXY	creates a pixmap in XY format of a specified size and returns an identifier for it
XStorePixmapZ	creates a pixmap in Z format of a specified size and returns an identifier for it
XStringWidth	returns the width of a string in pixels using supplied font information
XSync	flush output buffer and wait for the results of each action flushed
XText	draws text to the display
XTextMask	draws text to the display using the font characters as masks
XTextMaskPad	draws text to the display using the font characters as masks and using given padding between characters
XTextPad	draws text to the display with given padding between characters
XTileAbsolute	changes tile mode of a window to absolute mode
XTileFill	fills a given area with a given tile using a given clipping mask
XTileRelative	changes tile mode of a window to relative mode
XTileSet	fills a given area with a tile with no clipping set
XUndefineCursor	disassociates a cursor from a window
XUngrabButton	release control of a mouse button
XUngrabMouse	release control of the mouse
XUngrabServer	release control of the server
XUnmapSubwindows	unmaps all subwindows of a given window
XUnmapTransparent	unmaps a given transparent window
XUnmapWindow	unmaps a given window
XUpdateMouse	gets the latest mouse coordinates and flushes the input queue of all pending mouse events
XWarpMouse	moves the mouse to a specified position in a specified window
XWindowEvent	looks for specific events from specific windows in the input queue

XrLIB Library Functions

XrAddPt	add two points together
XrCheckBox	create and manipulate a set of check boxes
XrCopyPt	copy one point into another
XrCopyRect	copy one rectangle into another
XrEditor	manipulate a window's editors
XrEditorGroup	define and manipulate a grouping of editors
XrEmptyRect	find out if a rectangle is empty
XrEqualPt	compare two points for equality
XrEqualRect	compare two rectangles for equality
XrGetWindowEvent	find which event a defined value maps to
XrInit	initialize xrlib
XrInput	get input and set input processing
XrInsetRect	shrink or expand a rectangle
XrMapButton	map a button event to a defined value
XrMenu	create and interact with a cascading menu tree
XrMessageBox	create and interact with a pop-up message
XrOffsetPt	offset a point
XrOffsetRect	offset a rectangle
XrPageEdit	create and edit a block of text
XrPanel	create and interact with a collection of field editors
XrPt2Rect	find the smallest rectangle which encloses two points
XrPtInRect	find out if a point is within a rectangle
XrPushButton	create and manipulate a set of push buttons
XrRadioButton	create and manipulate a set of radio buttons
XrRasterEdit	create and modify a raster image
XrRasterSelect	create and manipulate a set of raster images
XrResource	access the resource manager
XrScrollBar	create and manipulate a vertical or horizontal scroll bar
XrSectRect	find the intersection of two rectangles
XrSetPt	set the values of a point
XrSetPtRect	set the values of a rectangle from two points
XrSetRect	set the values of a rectangle
XrStaticRaster	create and display a raster image
XrStaticText	create and display a block of text strings
XrStringWidth	calculate the pixel length of a text string
XrSubPt	subtract two points
XrTextEdit	create and manipulate a single line of text
XrTitleBar	create and manipulate a title bar
XrUnionRect	find the union of two rectangles
XrVersion	return the version string for XrLib

Compatibility

The X Window System product conforms to the X Version 10 Release 4 Specification.

System Requirements

Hardware

- HP 9000 Model 825
- 8 Mbytes RAM

Software

- HP-UX 1.1 or later

Optional

- 98194 Model 825 LAN Link if system is to be networked with other systems.

Ordering Information

Software

- 92524A X Window System for Model 825
  - Opt. AA0 software on 1/4-inch tape cartridge
  - Opt. AA1 software on 1/2-inch, 1600 cpi tape
- 92524R Right-to-reproduce the X Window System. Includes manual set and right-to-reproduce certificate.
  - Opt. 0B0 Delete manuals.

See the latest HP 9000 price guide for complete ordering information.

Documentation

For further technical information, consult the following references:

98672-90001	Programming with the X Window System
82320-90002	X Window User Guide

PORT/HP-UX

HEWLETT PACKARD

For migration from HP 1000 RTE to HP Precision Architecture HP-UX

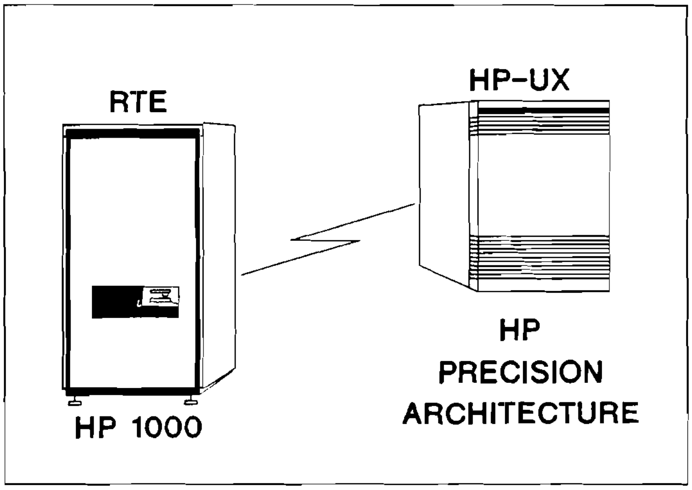
PORT/HP-UX is a collection of utilities for migrating applications from HP 1000 RTE-6/VM and RTE-A to HP Precision Architecture with HP-UX. PORT/HP-UX utilities minimize the porting effort for high level applications and data. Key elements of PORT/HP-UX include a Migration Analysis Utility, file and data transport utilities, and RTE intrinsics emulation routines on HP-UX. These utilities are complemented by highly compatible high level languages and translators to ease the migration of subsystem dependent code.

Features

- **Migration and Coexistence Documentation**
- **Migration Analysis Utility**
- **RTE Emulation Software**
- **RTE Interactive Environment on HP-UX**
- **RTE to HP-UX Data Transfer Utilities**
- **IMAGE Migration Utilities**
- **FORTRAN Conversion Utilities**
- **Compatible Subsystems:**
  - FORTRAN
  - Pascal
  - C
  - IMAGE/ALLBASE
  - Network Services
  - AGP/DGL Graphics
  - FORMS

The Migration Process

The objectives of PORT/HP-UX migration are to maximize performance in the new operating environment and to minimize the migration effort. PORT/HP-UX supports source level migration to fulfill those objectives. There are three steps in the migration process: (1) Evaluation, (2) Transport, and (3) Conversion.



Evaluation/Preparation

Using the Migration Evaluation Kit Manuals as a guide, applications currently being developed can be designed for maximum portability to HP Precision Architecture systems. The Migration Analysis Utility scans source code and identifies RTE system and subsystem dependencies and notes whether or not they are emulated with PORT/HP-UX. This information, along with error messages from the HP Precision Architecture compiler, will indicate the areas in the application which will require modification.

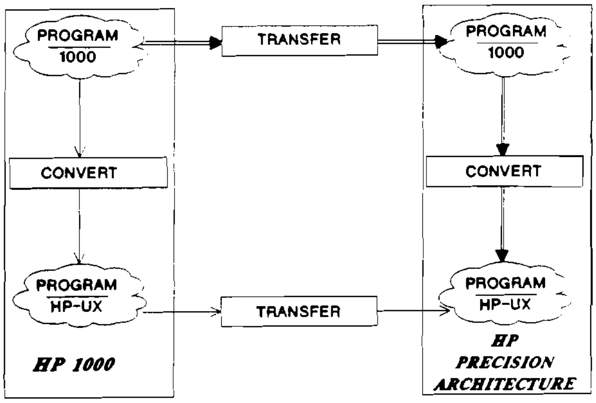
Application Transport

Programs (source code) and data (ASCII or binary) are transported to the HP Precision Architecture system via tape. PORT/HP-UX utilities correct for operating system differences to accomplish the application transport.

Conversion

Programs are compiled on the HP-UX system, which allows non-emulated code to run in native mode. Most RTE file system, EXEC, and system library routines are emulated. RTE dependencies not emulated must be rewritten by the user, using equivalent HP-UX intrinsics.

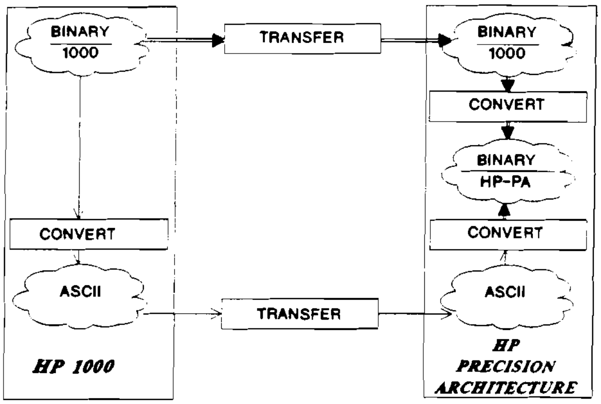
PROGRAM CONVERSION PROCESS



Syntactic and semantic differences in HP 1000 FORTRAN programs are converted to the HP FORTRAN equivalents either programmatically or by the user. A FORTRAN conversion utility translates those differences or they are identified by the compiler. The Language Migration Guides also help identify language differences and provide conversion details.

ASCII data needs no conversion. Binary data is converted using instructions in the PORT/HP-UX Migration User's Guide and associated utilities. Data in floating point format must be converted to IEEE format via a user program using the floating point conversion subroutines provided in PORT/HP-UX.

DATA CONVERSION PROCESS



PORT/HP-UX and Migration Tools

Programming for Portability

The ease with which software is ported from one architecture to another depends on the emphasis placed on portability during the design and implementation of the application. Programming for Portability is a set of documented guidelines for ensuring portability from RTE-6/VM and RTE-A applications to HP Precision Architecture HP-UX. Language, Operating System, and Subsystem considerations are described in Programming for Portability.

Migration Analysis Utility (MAU)

The Migration Analysis Utility is a utility that scans source code for HP 1000 system and subsystem dependencies. It generates a report pointing out the lines that contain non-emulated, partially emulated, and fully emulated RTE calls. The MAU scans programs written in C, Pascal, and FORTRAN and flags RTE system library, FMP, EXEC, IMAGE, DS, and Graphics call dependencies on the HP 1000 or HP 9000 Series 800. The MAU and the HP Precision Architecture compilers can be used to estimate and plan the migration effort.

RTE System Library Emulation

Over 80% of RTE System Library calls are emulated in HP-UX. Non-emulated calls include those used for session accounting and programmatic spooling, which are not used frequently in RTE. The 80% that are emulated typically account for approximately 98% of user's calls to system library routines. Recoding for some non-emulated calls is documented in the PORT/HP-UX User's Guide.

RTE EXEC Emulation

Most RTE EXEC calls are emulated under PORT/HP-UX. The emulated EXEC calls include calls for I/O operations, to terminate or suspend programs, to load program segments into memory, to schedule other programs, to perform class I/O, and to time schedule programs. PORT/HP-UX supports EXEC calls to HPIB, GPIO, RS232, 9-track tape, and line printers. Discs are supported through the file system; programs which make EXEC calls to disc will have to be modified to use HP-UX directly.

RTE File System Emulation

PORT/HP-UX contains an emulated RTE file system that includes both the FMGR and CI file systems. Over 85% of the RTE file system calls are emulated under PORT/HP-UX.

The System Library, EXEC, and File System emulation included in PORT/HP-UX provide a very comprehensive RTE environment and support an estimated 95%–100% of the occurrences of RTE calls in typical HP 1000 applications.

Assembly Language Programs

Due to the architectural dependencies of assembly language, programs written in HP 1000 assembler must be rewritten to port to HP Precision Architecture systems. Often, raw application speed can be maintained by rewriting assembly-language routines in an optimized high level language, such as C.

Interactive RTE Services

Entering the command "rtesh" at the HP-UX prompt will put the user in a "CI-like" environment for controlling RTE emulated programs and supporting the creation and maintenance of an emulated RTE file system. The following RTE commands are emulated under rtesh:

AT	BR	CL	CO	CR
DL	CRDIR	EX	GO	IO
LI	MO	OF	OWNER	PR
PROT	PU	RN	RP	RU
SS	WD	WH	XQ	??

In addition, an EDIT/1000-like screen editor is provided, which supports most EDIT/1000 features.

Database Migration Utilities

Although the HPIMAGE interface of ALLBASE is quite similar to IMAGE/1000, architectural differences make 100% compatibility impossible. Migration utilities can be used to port application programs and databases. IMAGE/1000 applications can be recompiled and executed on HP-UX with a run-time translator intercepting IMAGE/1000 calls and making the appropriate call to HPIMAGE. Only 1% of the calls are architecturally dependent and will have to be changed or removed for translator compatibility. The MAU flags those incompatible calls.

Two tools are used in moving the database from RTE to HP-UX. The Database Migration Unload utility unloads and simultaneously converts IMAGE/1000 databases into HPIMAGE format. A Rootfile Decompiler utility converts IMAGE/1000 root files into HPIMAGE schema files. A manual conversion is necessary to convert IMAGE/1000-I rootfiles. The database migration utilities come standard with ALLBASE.

Subsystem Compatibility

FORTRAN

FORTRAN on HP Precision Architecture systems is compatible with HP 1000 FORTRAN. Both are based on the ANSI177 standard. Ninety percent of the HP 1000 FORTRAN feature set is compatible with FORTRAN on HP Precision Architecture. A FORTRAN conversion utility on HP-UX is run on source programs prior to compilation on the HP Precision Architecture system. This utility increases feature set compatibility to 98%.

Pascal

HP Pascal is a superset of HP Standard Pascal. HP Standard Pascal is based on the ANSI standard. Pascal/1000 is based on HP Standard Pascal and includes some extensions. These extensions are flagged by the Pascal/1000 compiler.

C

Corporate Computer Systems (CCS) provides a C compiler for the HP 1000 that is 99% feature set compatible with HP C/HP-UX on HP Precision Architecture. Both HP C/HP-UX and HP 1000/C conform with the emerging ANSI standard.

Graphics

Graphics support for HP Precision Architecture HP-UX systems includes AGP/DGL, which differs slightly from AGP/DGL on HP 1000 systems. The differences relate primarily to error handling and the use of device file names in HP-UX versus LU parameters in RTE. The MAU flags differences. PORT/HP-UX documentation provides instructions for recoding. The AGP/DGL documentation further identifies areas of differences and offers instructions for porting.

Networking

Networking on HP Precision Architecture/HP-UX systems is based on Network Services software and LAN/Link hardware. HP Precision Architecture HP-UX systems can communicate via the LAN with HP 1000 A-Series systems and HP 9000 Series 200/300/500 systems running Network Services Software. HP 1000 A-Series can thus serve as gateways between HP 1000 E/F-Series systems running DS/1000-IV and HP Precision Architecture systems running Network Services Software.

HP 1000 A-Series applications using Network Services (NS) calls can be ported to HP Precision Architecture systems since the HP Precision Architecture networking products are also based on Network Services. A- and E/F-Series applications using DS/1000-IV can be evaluated using the MAU to flag calls that don't map directly to NS/1000 and HP Precision Architecture Network Services. DS/1000-IV to NS/1000 migration documentation explains how to recode to achieve DS/1000-IV functionality on NS/1000.

FORMS

An emulated implementation of FORMS/1000 is provided on HP Precision Architecture. F1000/HP-UX is 100% compatible with FORMS/1000, so no changes to Forms calls are required when moving an application from an HP 1000 system to an HP Precision Architecture system.

Migration Consulting

Migration consulting for your specific application is available through HP's Applications Engineering Organization. Contact your local HP Sales Office for details.

Migration Documentation

92561-90001	Programming for Portability Guide
92561-90002	PORT/HP-UX Migration Analysis Utility Manual
92561-90003	PORT/HP-UX Migration User's Guide
92561-90004	PORT/HP-UX Reference Manual
92430-90003	HP FORTRAN 77/HP-UX Migration Guide
92431-90004	HP Pascal Migration Guide
36217-90008	Migrating to ALLBASE/HP-UX
92561-90009	NS/9000 Series 800 Migration Guide
92012-90001	F1000/HP-UX Reference Manual

Ordering Information

PORT/HP-UX consists of utilities which run under both RTE and HP-UX. For that reason, these utilities are distributed separately, as follows:

The Migration Evaluation Kit, product number 92561A, contains the utilities that run on the HP 1000. The Migration Analysis Utility, database migration utilities, and some transport utilities are part of the Migration Evaluation Kit. The Migration Evaluation Kit also includes a full set of migration documentation, as listed above.

RTE emulation, the floating point conversion utility, some transport utilities, and F1000/HP-UX are included in HP-UX on HP Precision Architecture.

The FORTRAN conversion utility is included with HP FORTRAN on HP Precision Architecture.

The database translation utility is included in ALLBASE/HP-UX.

Contact your local Hewlett-Packard Sales Office for further information.

HP FORTRAN 77/HP-UX

HEWLETT PACKARD

For HP 9000 Series 800 Systems

Product Number HP 92430

FORTRAN 77, the most recent ANSI FORTRAN standard, incorporates a number of improvements and extensions. Among the more important are an IF-THEN-ELSE control statement, a CHARACTER data type, and generalized Input/Output facilities. HP FORTRAN 77 is a superset of the ANSI FORTRAN 77 standard and includes MIL-STD-1753 extensions and other frequently offered extensions such as NAMELIST and ENCODE/DECODE. It provides a well-defined language standard for software portability and offers extensions that enhance the flexibility and power of the FORTRAN 77 language.

HP FORTRAN 77/HP-UX is an implementation of HP FORTRAN 77 for the HP 9000 Series 800 systems. It runs under HP-UX and provides access to HP-UX subsystems and other HP-UX languages. It is highly compatible with FORTRAN 77/1000, FORTRAN 77/3000, and FORTRAN 77/9000 Series 200, 300 and 500.

Key Features

Program Structures

- SUBROUTINE and FUNCTION subprograms permit creation of modular programs.
- Multiple ENTRY points to a subprogram permit controlled sharing of data.
- Structured programming statements: IF-THEN-ELSE, Block DO, and DO WHILE, increase programmer productivity.

Data Manipulation

- INTEGER, REAL, LOGICAL and COMPLEX data types and operations provide efficient numeric computation.
- DOUBLE PRECISION and DOUBLE COMPLEX data types and operations provide extended precision numeric computation.
- CHARACTER data types and operations provide convenient string management operations.
- Bit manipulation functions provide shift, extract, and test operations.

Input/Output

- READ, WRITE, OPEN, CLOSE, and INQUIRE provide control and management of sequential or direct access, formatted or unformatted files.
- NAMELIST, list directed formatting, and standard devices, provide simplified I/O programming.
- ENCODE/DECODE and internal files provide in-memory FORMAT conversion.
- I/O specifiers ERR=, END=, and IOSTAT= allow flexible I/O error handling.

Source Program Form

- INCLUDE statement allows insertion of program text (e.g., COMMON declarations) from another file.
- $IF-$THEN-$ELSE directives provide conditional compilation.
- Names may have more than six characters to allow creation of meaningful names.
- End of line comments provide convenient placement of program commentary.
- Up to 99 continuation lines.

Compiler Features

- The compiler permits specification of three levels of optimization:
  - Default optimization, necessary for symbolic debugging.
  - Level 1 optimization, produces smaller and faster code.
  - Level 2 optimization, produces the smallest and fastest code.
- Symbolic Debug/HP-UX support.
- Statement functions are implemented as macros.
- Compiler directives provide:
  - control of range checking and parameter checking.
  - control of conformance-to-standard checking.
  - selection of alternate semantics for DO loops, INTEGERs and LOGICALs, and data initialization.
  - control of program listing content.
  - PRAGMAS for improved optimization control.
  - $ALIAS, which facilitates inter-language calls.
  - NLS support.
  - Alignment options to handle specialized data alignment needs.

Compatibility

HP FORTRAN 77/HP-UX is completely compatible with HP FORTRAN 77/XL, HP FORTRAN 77/V, and HP FORTRAN 77/9000-500. It is highly compatible with FORTRAN 77/1000, and FORTRAN 77/9000-300.

- Reference Manual Supplement (92430-90001) describes conversion to HP FORTRAN 77/HP-UX issues.
- A conversion utility, supplied with PORT/HP-UX, translates FORTRAN 77/1000 features which have HP FORTRAN 77/HP-UX equivalents.
- Language extensions, provided by HP FORTRAN 77/HP-UX, support most FORTRAN 77/1000 features which do not have HP FORTRAN 77/HP-UX equivalents.

System Environment

HP FORTRAN 77/HP-UX is supported on HP-UX based HP 9000 Series 800 systems.

Ordering Information

Software

Product Number	Product Description
92430A	HP FORTRAN 77/HP-UX Right-to-Use Product.
92430R	HP FORTRAN 77/HP-UX Right-to-Copy product.

Documentation

For further technical information consult the following manuals:

Part Number	Document Description
5957-4685	HP FORTRAN 77 Reference Manual
92430-90001	Reference Manual Supplement
5957-4686	HP FORTRAN 77 Programmer's Guide
92430-90002	Programmer's Guide Supplement
5957-4687	HP FORTRAN 77 Quick Reference Guide
92430-90003	HP FORTRAN 77/HP-UX Migration Guide

Customer Training

Two training courses are available: a self-paced course with self-study guide and videotape and a five day classroom course.

Course Number	Course Name
22961C	HP FORTRAN 77 Self-paced Training
22959C	Programming in FORTRAN 77

HP Pascal/HP-UX

For HP 9000 Series 800 Systems

Product Number HP 92431

HP Pascal/HP-UX (hereafter referred to as HP Pascal) is Hewlett-Packard's implementation of the Pascal language for the HP 9000 Series 800 systems. It allows several extensions to the ANSI/IEEE and ISO standards for Pascal, including those which provide it with system programming capabilities. HP Pascal will be particularly useful for development of large systems and subsystems, and those applications which require system level programming.

Pascal offers a very rich and powerful set of data types and control flow structures. Along with control constructs, Pascal's basic building blocks of structure, namely procedures and functions, permit a top-down approach to program development. The modular, self documenting character of Pascal's programs provide code which is easy to maintain and to enhance. As a further aid to program development, the language offers strong type checking and range checking capabilities. A significant portion of the debugging responsibility now rests with the compiler and the language.

HP Pascal offers systems programming extensions to the above features. These extensions have permitted Pascal to be used extensively within Hewlett-Packard for systems level applications.

Key Features

Program Structures

- Powerful control structures, such as IF-THEN-ELSE, CASE, WHILE-DO, and REPEAT-UNTIL, eliminate the need for GOTOs and provide structure to the programs, improving their readability.
- PROCEDURE, FUNCTION and MODULE subprogram statements allow top-down, modular development of programs.

Data Manipulation

- REAL, LONGREAL, and INTEGER types provide efficient numeric computation.
- CHARACTER and STRING types, when coupled with pre-defined string handling functions, allow powerful string management operations.
- ARRAY, RECORD, SET, and FILE types facilitate the handling of data in structured, user-defined format.
- ENUMERATED and SUBRANGE types provide further customization of standard data types.
- CONFORMANT ARRAY parameter type provides flexibility in parameter passing.

Input/Output

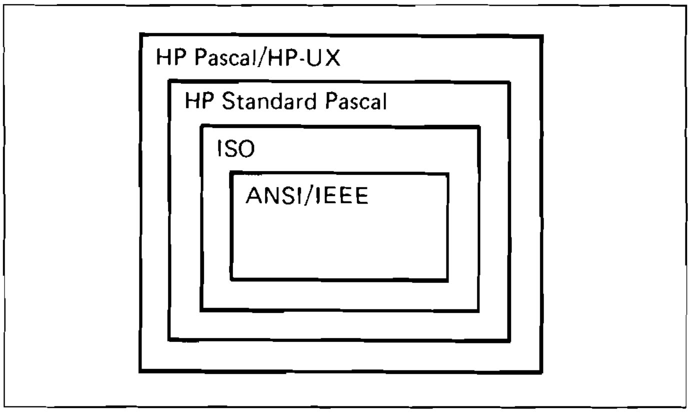
- I/O is with files which are of two types: text (sequential only), or structured files (can be sequential or direct access).
- Convenient I/O functions including READ, WRITE, OPEN, and CLOSE file functions, and other essential file management capabilities.
- Implicit data conversion for text files during input converts from ASCII to appropriate file variable format. The reverse is true for output.

Development Environment

- The full complement of utilities available with HP-UX can be used with HP Pascal. These utilities include full screen editing, version control via SCCS (Source Code Control System), and symbolic debugging via the HP-UX Symbolic Debugger.

Language Standards

HP Pascal is a superset of the ANSI/IEEE 770X3.97-1983 and ISO 7185-1983 standards for Pascal. The systems programming extensions also make it a superset of the HP Standard for Pascal. The following figure illustrates HP Pascal's data set.



HP Standard Extensions to ANSI

The following are important extensions to ANSI/IEEE and ISO:

- STRING data type and associated functions
- Structured constants, constant expressions
- Direct access and other I/O extensions
- MODULES and separate compilation
- LONGREAL data type

Systems Programming Extensions

HP Pascal includes extensions which permit a very high degree of flexibility for data manipulation. Some of the salient features are:

- Control over Pascal's type checking
- ANYPTR type and pointer manipulation routines
- Exception handling
- Procedure and Function variables
- Extended addressing
- Flexible procedure calling mechanisms

Compiler Features

- Compiler options facilitate control of:
  - source code, its conformance to ANSI or HP standards, and its conditional compilation.
  - content of program listings.
  - object code generated, and range checking.
  - data layouts and their manipulation.
- The HP 9000 Series 800 architecture permits special optimizing strategies. The following levels of optimizing are possible:
  - No optimization (default), which is necessary during symbolic debugging.
  - Level 1 optimization, done over small code sections, reduces code space and execution time.
  - Level 2 optimization, done over an entire procedure, results in the smallest and fastest code.

Subsystem Access

- Provides access and interaction mechanism for HP-UX file system, and all other HP-UX languages. Users can build their own "intrinsics," and enjoy the flexibility that is provided by the intrinsic mechanism.
- Provides access to data bases (via HPSQL and HPIMAGE), and all other HP-UX subsystems. Relevant data sheets provide more information.

Compatibility

- HP Pascal is a superset of the ANSI/IEEE, ISO and Hewlett-Packard standards for the Pascal language.
- HP Pascal offers a high degree of compatibility with the Pascal on the HP 1000 and HP 9000 Series 200/300/500 Systems.
- Migration documentation outlines issues regarding migration of Pascal programs on the HP 1000 to HP Pascal on the HP 9000 Series 800 Systems.

System Environment

HP Pascal is supported on the HP 9000 Series 800 Systems.

Ordering Information

Software

Product Number	Product Description
92431A	HP Pascal/HP-UX Right-to-Use product
92431R	HP Pascal/HP-UX Right-to-Copy product

Documentation

Part Number	Document Description
31502-90001	HP Pascal Reference Manual
31502-90002	HP Pascal Programmer's Guide
92431-90004	HP Pascal Migration Guide

Customer Training

Course Number	Course Description
31502AB	Advanced course that introduces Pascal programmers to HP Pascal extensions, particularly the system programming features, and to programming on HP-UX.

HP C/HP-UX

HEWLETT PACKARD

For HP 9000 Series 800 Systems

Bundled with HP-UX

C is a general-purpose programming language which features modern control flow and data structures, a rich set of operators, and economy of expression. It combines the convenience and portability of a high level language with the flexibility and efficiency of assembly language. C is the system programming language for UNIX* systems and a vehicle for transporting software from other UNIX systems.

HP C/HP-UX is an implementation of C for the HP 9000 Series 800 systems. It runs under HP-UX and provides access to HP-UX subsystems and other HP-UX languages. It is highly compatible with C/9000, and VAX† C and with CCS/C 1000 and CCS/C 3000 from Corporate Computer Systems.

Key Features

Program Structures

- Functions are recursive and can return scalar, structure, or union values, or void.
- Block structured scope of variables is provided.
- Public and private functions and data is provided.
- Structured programming statements: `if-else`, `for`, `while`, `switch`, are available.

Data Manipulation

- `int`, `float`, `double`, and `char` provide integer, floating point, and character data types and operations.
- `short`, `long`, `signed`, and `unsigned` provide control of size and interpretation of numeric data items.
- `auto`, `static`, `extern`, and `register` permit control of allocation of data items.
- Arrays, structures, and unions provide composite data types.
- Pointers provide effective manipulation of addresses.

Input/Output

- `fscanf`, `fprintf`, `fopen`, `fclose`, and their variants provide control and management of sequential, formatted files.
- Standard input and output files simplify programming.
- Standard error file is available for error messages.
- In-memory format conversion is available.

Source Program Form

A macro pre-processor provides a variety of source program management functions.

- `#define` statement allows definition of macros.
- `#include` statement allows inclusion of program text (e.g., common declarations) from another file.
- `#if-#else-#endif` directives provide conditional compilation.
- Names may have more than eight characters to allow creation of meaningful names.

Compiler Features

- The compiler permits specification of three levels of optimization:
  - No optimization (default), is necessary for symbolic debugging.
  - Level 1 optimization, produces smaller and faster code.
  - Level 2 optimization, produces the smallest and fastest code.
- Symbolic Debug/HP-UX support.

Compatibility

HP C/HP-UX is highly compatible with CCS/C 1000, CCS/C 3000‡, C/9000, and VAX C.

- Documentation included with product describes compatibility issues.
- Compatibility with ANSI C will be provided soon after the standard is defined.

System Environment

HP C/HP-UX is supported on HP-UX based HP 9000 Series 800 systems.

*UNIX is a registered trademark of AT&T in the U.S. and other countries.
†VAX is a trademark of Digital Equipment Corporation (DEC).
‡Trademark of Corporate Computer Systems, Inc. (N.J.)

Ordering Information

Software

HP C/HP-UX is bundled with the HP-UX operating system.

Documentation

For further technical information, consult the following manuals:

Part Number	Document Description
92434-90001	HP C Reference Manual
92434-90002	HP C Programmer's Guide
92434-90003	HP C Quick Reference Guide
92434-90004	HP C Reference Manual Supplement

Customer Training

A five day classroom course is available:

Course Number	Course Name
35130A	The C Programming Language

HP Assembler/HP-UX

HEWLETT PACKARD

For HP 9000 Series 800 Systems

Bundled with HP-UX

The HP Precision Architecture Assembler provides access to all architectural features of HP Precision Architecture computer systems.

HP Precision Architecture Assembler/HP-UX is an implementation of the Assembler for HP 9000 Series 800 systems. It runs under HP-UX and provides access to HP-UX subsystems and other HP-UX languages.

Key Features

- Symbolic statement labels.
- Mnemonic instruction codes.
- Pseudo-instruction codes for frequently used instruction sequences.
- Pseudo-operations for data initialization.
- Directives to control allocation of code and data.
- Directives to facilitate use of the standard call sequence.
- Execution options to control listing content.

System Environment

Assembler/HP-UX is supported on HP-UX based HP 9000 Series 800 systems.

Ordering Information

Software

Assembler/HP-UX is bundled with the HP-UX operating system on the Series 800.

Documentation

For further technical information consult the following manuals:

Part Number	Document Description
92432-90001	Assembly Language Reference Manual

Customer Training

A five day classroom course will be available.

HP Symbolic Debugger/HP-UX

HEWLETT PACKARD

For HP 9000 Series 800 Systems

Bundled with HP-UX

The HP-UX Symbolic Debugger is a powerful, full-featured symbolic debugger to help you find and correct errors in your programs. It runs under the HP-UX operating system on Hewlett-Packard's HP 9000 Series 800 systems. It supports interactive source level debugging for Hewlett-Packard's HP-UX languages of C, Pascal and FORTRAN 77. The HP-UX Symbolic Debugger is based on the popular industry debugger, **cdb**, and has been extended to include additional features to provide greater functionality and user friendliness. For example, a window-oriented interface, on-line help, and macro facility have been added.

The HP-UX Symbolic Debugger allows you to examine the program state in which an error or some other condition, such as a signal, occurs. You may take corrective action and resume execution, or abort the program.

When first entered, the debugger displays the first executable statement, unless there is a core file present, in which case the debugger will display the signal that caused the program to dump core and the location of the error. Once inside the debugger, a wide variety of commands are available for use.

- An optional **command list** associated with a breakpoint supports conditional breaking and allows other actions to be taken automatically when the breakpoint is reached.
- A **break** facility halts execution of your program at specified locations, allowing you to examine the state of your program.
- A **trace** facility lets you trace specified program flow by displaying the stack of a program.
- A **step** facility lets you single step through the program a source statement or machine instruction at a time.
- An **assertion** facility supports the execution of a list of commands before every source statement, for location of elusive bugs.
- **Data viewing commands** allow display of program data in various formats.
- A **record and playback** feature is provided to save debugger commands in a file and play them back for later use, allowing easy recreation of the state of a program.
- HP-UX Symbolic Debugger is **window-oriented** and will support multiple panes, allowing viewing of source statements and assembly instructions simultaneously. The window-like interface is implemented through the use of memory-locking. Terminals that do not support memory-locking may be used with the product via a line-oriented interface.
- A **macro** facility is supported to allow renaming of commands or creating a command consisting of several commands.
- A host of additional features beyond those just described, including an **on-line help and documentation facility**, make HP-UX Symbolic Debugger a friendly and powerful programmer's productivity tool.

Key Features

- Language sensitive expression evaluator for HP C, HP Pascal and HP FORTRAN 77 programming languages.
- Unrestricted number of breakpoints that can be set, deleted, suspended and activated.
- Single-stepping through a program at the source statement or the equivalent assembly instruction level.
- Assertions for tracing elusive bugs.
- Control of user program exceptions (signal handling) within the debugger.
- Display and modification of program variables.
- Regaining control of free running user program initiated by the debugger.
- Multi-level stack tracing.
- Extensive interrogation of the debugger state, including breakpoints, files, globals, procedures and their variables.
- Record and playback of debugger commands to help recreate debugging sessions.
- Access to HP-UX operating system via running a shell program within the debugger.
- Macro facility allowing users to define their own names for command strings.
- Debugging of core files, allowing post-mortem debugging.
- Friendly user interface including online help, support of multiple skill levels, and a window-like interface.
- Ability to adopt and debug a free running program.

Functional Description

Source level symbolic debugging. While in the debug process, the HP-UX Symbolic Debugger recognizes the names, types and locations of all the variables and routines for the HP-UX languages of C, FORTRAN 77 and Pascal. Programs can be debugged without the need for load maps, symbol table dumps or mixed listings.

Assembly level debugging. The debugger will allow you to display high-level source code in assembly language. It can single-step at the instruction level and supports display and modification of registers. Assembly level debugging is also the default condition for programs with no compiler-generated symbolic debug information.

Interactive debugging process. The user interacts with the program as it runs and can examine or alter variable values while the program runs without having to insert statements into the code.

Separate Process. HP-UX Symbolic Debugger runs as a process separate from the user program. No significant code or data space is lost and no extra statements are added in order to debug. The program being debugged runs exactly the same as it would normally. There is no need to restructure the program to debug it.

Free running programs. HP-UX Symbolic Debugger supports an option to adopt and debug a program which began execution outside the control of the debugger.

Unrestricted number of breakpoints. HP-UX Symbolic Debugger supports an unrestricted number of breakpoints to monitor program execution; for example, to halt if a variable reaches a specified value. Any number of possible paths can be trapped and values can be quickly tracked through the program to determine error conditions.

Supports command files and message logging. Debugging sessions may occur where users can submit debug commands in a file, and have results logged in another file. This automates the debugging process, so users don't have to wait for bugs whose symptoms may take a prolonged period of time to occur.

Compatibility

- The debugger is Hewlett-Packard's version of the **cdb** debugger, which runs on the HP 9000 Series 200, Series 300 and Series 500 product lines. HP-UX Symbolic Debugger is highly compatible with **cdb**. However, to support greater functionality and user friendliness, some modifications and extensions to the **cdb** debugger have been implemented. For example, the positioning of the command arguments has been altered slightly, and the window-like interface and macro capabilities have been added. To support additional languages beyond C, the separator for variable names has been changed from a period (.) to a colon (:), since a period is an operator used for accessing the fields of a record in the HP-UX languages of C and Pascal. Compatibility information is contained in the "HP-UX Symbolic Debugger Users Guide."
- HP-UX Symbolic Debugger does not support source-level debugging of optimized code.

System Environment

HP-UX Symbolic Debugger is supported on the HP-UX based HP 9000 Series 800 systems.

Ordering Information

Software

The HP Symbolic Debugger is bundled with the HP-UX operating system.

Documentation

For further technical information consult the following manuals:

Part Number	Document Description
92435-90001	HP-UX Symbolic Debugger Users Guide
92435-90002	HP-UX Symbolic Debugger Quick Reference Guide

Customer Training

Bundled with the advanced languages classroom training course.

Starbase Graphics Library

Starbase is a high-performance 2-D and 3-D graphics library for the HP 9000 family. Starbase is an extension of the evolving ANSI Virtual Device Computer Graphics Interface (VD-CGI) standard. The intent of this standard is to provide a common interface to graphics devices for the industry.

The Starbase implementation is a layered approach allowing the user to be insulated from device hardware due to device independence. Any industry-standard graphics libraries added to the system will be above Starbase, allowing those libraries to also be device-independent. Starbase provides application programs and standard graphics libraries a high-performance interface to HP graphics hardware and other selected graphics peripherals. The library is accessible by FORTRAN, C, Pascal, LISP, or Ada allowing the application developer to choose the language most suitable for his application. In addition to being based on the ANSI VD-CGI standard, Starbase has many extensions which provide the user increased flexibility, power, and performance.

Starbase provides procedures to perform basic graphics operations such as lines, markers, text and polygons, to 3-D transformations, to output primitives generating high-quality text, filled polygons, and providing synchronous and asynchronous input. In addition, Starbase has been enhanced beyond VD-CGI to support such things as polygon shading with multiple light sources, hidden-surface removal, splines and spline surfaces, and depth cueing.

Starbase Graphics Library Commands

append_text — Append a string of characters to something already written.

arc, ellipse, partial_arc, partial_ellipse — Define a circular/elliptical region to be filled and/or edged.

await_event — Wait for an event to occur, then return the class of that event.

await_retrace (&) — Wait for vertical retrace to start on raster scanning devices.

backface_control (@) — Control the action of back-facing polygons in modelling mode.

background_color, background_color_index — Set the background color.

bank_switch (&) — Specify which bank of frame-buffer planes to enable (for machines with more than 8 planes).

bitmap_print, dcbitmap_print — Copy a bitmap image to a hard-copy graphics printer.

bitmap_to_file, dcbitmap_to_file — Store an image onto a file.

block_move (&), dcblock_move (&) — Frame-buffer-to-frame-buffer block copy procedure.

block_read (&), dcblock_read (&) — Frame-buffer-to-main-memory block copy procedure.

block_write (&), dcblock_write (&) — Main-memory-to-frame-buffer block copy procedure.

buffer_mode — Set the buffering mode for output primitives; i.e., enabled or disabled.

character_expansion_factor — Specify character cell height-to-width ratio.

character_height, dccharacter_height — Specify character height.

character_slant — Specify character slant.

character_width, dccharacter_width — Specify character width.

clear_control — Select the portion of display surface to be cleared when subsequent calls to clear_view_surface are made.

clear_view_surface — Set portion of display surface to background color (portion specified by clear_control).

clip_depth — Define the front and back clipping planes.

clip_indicator — Enable/disable clipping to specified boundary.

clip_rectangle — Define the current clip rectangle boundary.

concat_matrix — Multiply two matrices and return the resulting matrix.

concat_transformation2d, concat_transformation3d — Concatenate a matrix with current transformation matrix.

curve_resolution — Specify the resolution for arc or spline interpolation.

dbuffer_switch (&) — Switch buffers double-buffered image transition.

dc_to_vdc — Transform a DC point into a VDC point.

default_knots, u_knot_vector, v_knot_vector — Specify the knot vectors for spline curves and surfaces.

define_color_table (#) — Set one or more entries in the device color map.

define_raster_echo (&) — Define the bit pattern for a raster echo.

define_trimming_curve (@) — Define a spline-trimming curve.

depth_cue (@) — Enable/disable depth-cueing of output primitives.

depth_indicator — Enable/disable clipping to front and/or back clipping planes.

designate_character_set — Associate a G-set with a character set.

disable_events — Disable queuing of events from specified graphic input device.

display_enable (&) — Select which planes of a raster device are to be displayed.

double_buffer (&) — Enable double buffering for smooth image transition.

draw2d, draw3d, dcdraw — Draw line from current position to specified position.

drawing_mode (#) — Select pixel replacement rule for output primitives.

echo_type (#), dcecho_type (#) — Select output device's type of echo.

echo_update, dcecho_update — Specify the position of the output device's echo.

enable_events — Enable the queuing of events from the specified input device.

file_print — Send an image contained in a file to a graphics printer.

file_to_bitmap, file_to_dcbitmap — Retrieve an image which has been stored on file.

fill_color, fill_color_index — Set the fill color for polygons.

fill_dither (#) — Select number of colors searched for and placed in dither cell.

flush_matrices — Pop all matrices from the matrix stack.

gclose — Close the specified graphics device.

gerr_procedure, gerr_defaults, gerr_print_control, gerr_message — Control the handling of a graphics error.

gescape — Device-dependent input and output control.

gopen (with INDEV or OUTINDEV specified) — Open a Starbase graphics device.

gopen — Open a Starbase graphics device.

hidden_surface (@) — Enable/disable back-facing polygon cull and Z-buffer hidden-surface removal.

initiate_request — Start a request process without waiting for the result.

inquire_color_table — Return one or more current color map entries.

inquire_fb_configuration — Return information about the device's frame buffer.

inquire_file — Inquire about a file on which a bitmap image was stored.

inquire_gerror — Return information on the most recent graphics error.

inquire_hit, set_hit_mode — Enable/disable the hit-detection mechanism.

inquire_id — Return device driver information.

inquire_input_capabilities — Inquire the capabilities of the input device.

inquire_request_status — Inquire status of a request to an input device.

inquire_sizes — Return device-dependent sizes.

inquire_text_extent — Return the coordinates of the text extent rectangle.

interior_style — Select fill type and boundary visibility for polygons.

intra_character_space — Specify the spacing between character cells.

light_ambient (@) — Specify color and brightness of ambient light.

light_model (@) — Modify aspects of positional light sources.

light_source (@) — Define the color and position of light sources.

light_switch (@) — Turn on/off light sources.

line_color, line_color_index — Select the color for line primitives.

line_repeat_length (&) — Set the size of line type patterns for line primitives.

line_type (#) — Select a line type for line primitives.

make_picture_current — Flush all internal output buffers.

mapping_mode — Select isotropic or anisotropic mapping.

marker_color, marker_color_index — Select the color for polymarker primitives.

marker_orientation — Specify the orientation of markers.

marker_size, dcmarker_size — Specify the size of markers.

marker_type — Select a type of marker for polymarker primitives.

move2d, move3d, dcmove — Update the current pen position.

partial_polygon2d, partial_polygon3d, dcpartial_polygon — Define a subpolygonal region.

perimeter_color, perimeter_color_index — Select the color for polygon perimeters.

perimeter_repeat_length (&) — Set size of line type patterns for polygon perimeters.

perimeter_type — Select a line type for polygon perimeters.

polygon2d, polygon3d, dcpolygon — Define a polygonal region to be filled and/or edged.

polyline2d, polyline3d, dcpolyline — Move/draw between specified points.

polymarker2d, polymarker3d, dcpolymarker — Draw markers centered at specified points.

pop_matrix, pop_matrix2d, pop_matrix3d — Pop and optionally return transformation matrix on TOS.

push_matrix2d, push_matrix3d — Push a matrix onto the top of the matrix stack.

push_vdc_matrix — Push the VDC-to-DC transformation matrix on the top of the matrix stack.

read_choice_event — Read a choice event from the top of the event queue.

read_locator_event — Read a locator event from the top of the event queue.

rectangle2d, dcrectangle — Define a rectangular region to be filled and/or edged.

replace_matrix2d, replace_matrix3d — Replace the top of the matrix stack matrix.

request_choice — Request choice input.

request_locator — Request locator input.

sample_choice — Immediately return current choice state.

sample_locator — Immediately return current locator state.

set_locator — Set the locator location.

set_p1_p2 — Set physical device limits.

set_pick_window, set_pick_depth, inq_pick_window, inq_pick_depth — Define the pick aperture for the hit-detection mechanism.

set_signals — Disable/enable the signal function of the specified device.

shade_mode (@) — Enable or disable light source polygon shading.

shade_range (@) — Specify the range of color map indices from which to select colors.

spline_curve2d, spline_curve3d, spline_surface — Draw a free-form curve or surface.

surface_model (@) — Define specular-reflection characteristics for a surface.

text2d, text3d, dctext — Output a string of characters.

text_alignment — Select the type of alignment of text lines.

text_color, text_color_index — Select the color for text primitives.

text_font_index — Select the character font for text primitives.

text_line_path — Define the relative position between successive lines of text.

text_line_space — Set the spacing between lines for text primitives.

text_orientation2d, text_orientation3d — Specify text orientation.

text_path — Select the direction of text characters.

text_precision (@) — Select how text will be drawn. (With stroke text only)

text_switching_mode — Select designation and invocation mode of character sets.

track (&) — Asynchronously echo an input device's locator position on output device.

track_off (&) — Stop asynchronous tracking.

transform_point — Translate a point between Modelling, World, and Virtual Device Coordinates.

transform_points — Transform points using the matrix on the top of the matrix stack.

vdc_extent — Define a scaling transformation for p1-p2 region.

vdc_justification — Specify VDC extent placement within the viewport.

vdc_to_dc — Transform a VDC point into a DC point.

vdc_to_wc — Transform a VDC point into a WC point.

vertex_format (#) — Specify the format of coordinate-list vertices.

view_camera — Define a 3D viewing transformation using a camera model.

view_matrix — Define a viewing transformation matrix.

view_port — Define an area on the view surface for subsequent view_window, view_volume, and view_camera to map into.

view_volume — Define a 3D viewing transformation matrix using a volume/viewport model.

view_window — Define a 2D viewing transformation matrix using a window/viewport model.

viewpoint (@) — Define the eye position in world coordinates.

wc_to_vdc — Transform a WC point into a VDC point.

write_enable (&) — Select which planes of a frame buffer device are modifiable.

zbuffer_switch (@) — Enable a section of the display surface to do Z-buffer hidden-surface removal.

Note:

@ = SRX with transform engine only.

# = Conditionally available on terminals (HPTERM driver).

& = Not available on terminals (HPTERM driver).

Display List

The Display List is an optional Starbase subsystem allowing graphics data and commands to be stored in an application-independent manner to be later re-executed for display. The Starbase Display List subsystem provides the ability to create, modify and manipulate a hierarchical network of graphical "segments." A segment is a unit of graphical data containing primitive, attribute, control information and application-dependent data.

The ability to create and modify a flexible hierarchical segment network is a powerful feature allowing graphics applications to more easily create and interact with their application models. One common use of these capabilities is to create a segment network that models geometric relationships of a physical model, for example, a robot arm. Modification of modeling transformation in the segment network provides the ability to move the various components of the physical model, while modification of the viewing transformation and lighting/shading information changes the viewer's perception of the physical model.

While there is not an accepted standard for the 3-D hierarchical display list, several graphics standards include display lists. The ANSI GKS standard defines a 2-D linear display list. The Programmer's Hierarchical Interactive Graphics Standard (PHIGS), currently in development by an ANSI committee, will define a 3-D hierarchical display list. HP's Display List is modeled after the PHIGS standard.

Display List Commands

application_data — Insert application data into the currently open segment.

call_segment — Insert a call to a segment into currently open segment.

change_segment_references — Change call_segment elements to reference a different segment.

close_segment — Close currently open segment.

copy_segment — Copy contents of specified segment into currently open segment.

delete_eles — Delete elements from currently open segment.

delete_segment, delete_segment_and_references — Delete a segment or delete a segment and all references to it.

display_element — Display current element on specified graphics display device.

display_segment — Display a segment network on specified graphics display device.

dl_label — Insert a display list label into currently open segment.

inq_application_data, inq_application_data_size — Return information about application data in the current element of the currently open segment.

inq_calling_segment_list, inq_calling_segment_count — Return information about segments calling a given segment.

inq_ele_ptr, inq_ele_ptr_at_bound — Return information about element pointer position within currently open segment.

inq_ele_type — Return type of current element in open segment.

inq_num_refs — Return number of references to specified segment.

inq_open_segment — Return information about the currently open segment.

inq_pick_path, inq_pick_path_depth — Return the path to a picked primitive.

inq_segment_count, inq_segment_list, inq_segment_exists — Return information about segments in specified display list.

inq_traversal_depth — Return maximum depth of display list traversal.

open segment — Open a segment in display list of the specified device.

pick_from_segment — Initiate a pick from a segment.

refresh_element — Display current element on associated graphics display device.

refresh_segment — Display specified segment network on associated graphics display device.

rename_segment, rename_segment_and_references — Rename segment or segment and all references to segment.

replace_ele — Switch display list into replace mode.

set_disp_traversal_control — Insert display traversal control element into open segment.

set_ele_ptr, set_ele_ptr_relative, set_ele_ptr_relative_to_label, set_ele_ptr_at_end — Set element pointer in currently open segment.

set_pick_mode — Set picking mode used by pick_from_segment.

set_pick_traversal_control — Insert pick traversal control element into open segment.

set_traversal_depth — Set maximum depth of display list traversal.

traversal_lock — Control display locking during display list traversal.

HP-GKS Graphics Library

HP-GKS is a 2-D graphics library to use with HP-UX on the HP 9000 family. It is strictly compatible with the ANSI and ISO Graphical Kernel System (GKS) standard, Level 2b. Many third parties and other computer vendors offer implementations of the GKS standard; the contribution of HP-GKS is high performance and high quality, attributes not common to all implementations.

HP-GKS provides the same graphics functionality across different HP 9000 graphics hardware subsystems. Applications based on GKS for another vendor's system will be ported very easily to HP-GKS. HP-GKS makes use of Starbase device drivers so it will support the same input/output devices as the Starbase Graphics Library. HP-GKS will also run in a window.

HP-GKS Features

- **Output Primitives:** Polyline, polymarker, text, cell array (pixel array) and fill area. Individual attributes for text are: font, color, character, size, alignment and path; for fill area: color and interior style; for polyline: color and linetype; for polymarker: type, size and color.

- **Segment Primitives:** To create, copy, associate and delete segments. Segments can have a transformation, be inserted into other segments with different transformations, and be kept in a Workstation's Independent Segment Storage area, the WISS. Metafile input and output is defined.
- **Attribute Binding:** Attributes can be individually specified —for each primitive or bundled together, and tables can be defined for general use.
- **Input Operations:** Six classes of input are defined — locator, stroke, string, choice, valuator, and pick. Level 2b implements these for request mode of input (as opposed to sample/event which allows for asynchronous input).
- **Extensive Inquiries:** About current state and device capabilities. This allows the application to take full advantage of hardware features.
- **Dynamic Workstation Configurability:** A GKS "workstation" is a group of one or more graphical input devices and one output device. Dynamic configurability allows this combination to be defined at run time rather then when the application is developed.

HP-GKS Functions

gactm — accumulate transformation matrix (level: 1a)
gacwk — activate workstation (level: ma)
gasgwk — associate segment with workstation (level: 2a)
gca — cell array (level: 0a)
gclks — close gks (level: ma)
gclrwk — clear workstation (level: ma)
gclsg — close segment (level: 1a)
gclwk — close workstation (level: ma)
gcrsg — create segment (level: 1a)
gcsgwk — copy segment to workstation (level: 2a)
gdawk — deactivate workstation (level: ma)
gdsg — delete segment (level: 1a)
gdsgwk — delete segment from workstation (level: 1a)
geclks — emergency close gks (level: 0a)
gerhnd — error handling (level: 0a)
gerlog — error logging (level: 0a)
gesc — escape (level: ma)
gevtm — evaluate transformation matrix (level: 1a)
gfa — fill area (level: ma)
ggdp — generalized drawing primitive (level: ma)
ggtitm — get item type from gksm (level: 0a)
giitm — interpret item (level: 0a)
ginch — initialize choice (level: mb)
ginlc — initialize locator (level: mb)
ginpk — initialize pick (level: 1b)
ginsg — insert segment (level: 2a)
ginsk — initialize stroke (level: mb)
ginst — initialize string (level: mb)
ginsts — initialize string (FORTRAN 77 subset) (level: mb)
ginvl — initialize valuator (level: mb)
gmsg — message (level: 1a)
gmsgs — message (FORTRAN 77 subset) (level: 1a)
gopks — open gks (level: ma)
gopwk — open workstation (level: ma)
gpl — polyline (level: ma)
gpm — polymarker (level: ma)
gprec — pack data record (level: 0a)
gprecs — pack data record (FORTRAN 77 subset) (level: 0a)
gqacwk — inquire set member of active workstations (level: 1a)
gqasf — inquire aspect source flags (level: 0a)
gqaswk — inquire set member of associated workstations (level: 1a)
gqcf — inquire colour facilities (level: ma)
gqchb — inquire character base vector (level: 0a)
gqchh — inquire character height (level: ma)
gqchs — inquire choice device state (level: mb)
gqchsp — inquire character spacing (level: 0a)
gqchup — inquire character up vector (level: ma)
gqchw — inquire character width (level: 0a)
gqchxp — inquire character expansion factor (level: 0a)
gqclip — inquire clipping indicator (level: ma)
gqcntn — inquire current normalization transformation number (level: ma)
gqcr — inquire colour representation (level: ma)
gqdch — inquire default choice device data (level: mb)
gqdds — inquire default deferral state values (level: 1a)
gqdlc — inquire default locator device data (level: mb)
gqdpk — inquire default pick device data (level: 1b)
gqdsga — inquire dynamic modification of segment attributes (level: 1a)
gqdsk — inquire default stroke device data (level: mb)
gqdsp — inquire display space size (level: 0a)
gqdst — inquire default string device data (level: mb)
gqdvl — inquire default valuator device data (level: mb)
gqdwka — inquire dynamic modification of workstation attributes (level: 1a)
gqeci — inquire list element of colour indices (level: ma)
gqefai — inquire list element of fill area indices (level: 1a)
gqegdp — inquire list element of available generalized drawing primitives (level: ma)
gqentn — inquire list element of normalization transformation numbers (level: 0a)
gqepai — inquire list element of pattern indices (level: 1a)
gqepli — inquire list element of polyline indices (level: 1a)
gqepmi — inquire list element of polymarker indices (level: 1a)
gqetxi — inquire list element of text indices (level: 1a)
gqewk — inquire list element of available workstation types (level: 0a)
gqfaci — inquire fill area colour index (level: ma)
gqfaf — inquire fill area facilities (level: ma)
gqfai — inquire fill area index (level: 0a)
ggfais — inquire fill area interior style (level: ma)
gqfar — inquire fill area representation (level: 1a)
gqfasi — inquire fill area style index (level: 0a)
gqgdp — inquire generalized drawing primitive (level: 0a)
gqlcs — inquire locator device state (level: mb)
gqli — inquire number of available logical input devices (level: mb)

gqln — inquire linetype (level: ma)
gqlvks — inquire level of gks (level: ma)
gqlwk — inquire maximum length of workstation state tables (level: 0a)
gqlwsc — inquire linewidth scale factor (level: 0a)
gqmk — inquire markertype (level: ma)
gqmksc — inquire marker size scale factor (level: 0a)
gqmntn — inquire maximum normalization transformation number (level: 0a)
gqnt — inquire normalization transformation (level: ma)
gqops — inquire operating state value (level: 0a)
gqopsg — inquire name of open segment (level: 1a)
gqopwk — inquire set member of open workstations (level: 0a)
gqpa — inquire pattern size (level: 0a)
gqpaf — inquire pattern facilities (level: 0a)
gqpar — inquire pattern representation (level: 1a)
gqparf — inquire pattern reference point (level: 0a)
gqpcr — inquire predefined colour representation (level: 0a)
gqpfar — inquire predefined fill area representation (level: 0a)
gqpkid — inquire pick identifier (level:1b)
gqpks — inquire pick device state (level:b)
gqplci — inquire polyline colour index (level: ma)
gqplf — inquire polyline facilities (level: ma)
gqpli — inquire polyline index (level: 0a)
gqplr — inquire polyline representation (level: 1a)
gqpmci — inquire polymarker colour index (level: ma)
gqpmf — inquire polymarker facilities (level: ma)
gqpmi — inquire polymarker index (level: 0a)
gqpmr — inquire polymarker representation (level: 1a)
gqppar — inquire predefined pattern representation (level: 0a)
gqpplr — inquire predefined polyline representation (level: 0a)
gqppmr — inquire predefined polymarker representation (level: 0a)
gqptxr — inquire predefined text representation (level: 0a)
gqpx — inquire pixel (level: 0a)
gqpxa — inquire pixel array (level: 0a)
gqpxad — inquire pixel array dimensions (level: 0a)
gqsga — inquire segment attributes (level: 1a)
gqsgp — inquire number of segment priorities supported (level: 1a)
gqsgus — inquire set member of segment names in use (level: 1a)
gqsgwk — inquire set member of segment names on workstation (level: 1a)
gqsks — inquire stroke device state (level: mb)
gqsts — inquire string device state (level: mb)
gqstss — inqurie string device state (FORTRAN 77 subset) (level: mb)
gqtxal — inquire text alignment (level: ma)
gqtxci — inquire text colour index (level: ma)
gqtxf — inquire text facilities (level: ma)
gqtxfp — inquire text font and precision (level: 0a)
gqtxi — inquire text index (level: 0a)
gqtxp — inquire text path (level: 0a)
gqtxr — inquire text representation (level: 1a)
gqtxx — inquire text extent (level: ma)
gqtxxs — inquire text extent (FORTRAN 77 subset) (level: ma)
gqvls — inquire valuator device state (level: mb)
gqwkc — inquire workstation connection and type (level: ma)
gqwkca — inquire workstation category (level: 0a)
gqwkcl — inquire workstation classification (level: 0a)
gqwkdu — inquire workstation deferral and update states (level: 0a)
gqwkdm — inquire workstation maximum numbers (level: 1a)
gqwks — inquire workstation state (level: 0a)
gqwkt — inquire workstation transformation (level: ma)
grditm — read item from gksm (level: 0a)
grensg — rename segment (level: 1a)
grqch — request choice (level: mb)
grqlc — request locator (level: mb)
grqpk — request pick (level:1b)
grqsk — request stroke (level: mb)
grqst — rquest string (level: mb)
grqsts — request string (FORTRAN 77 subset) (level: mb)
grqvl — request valuator (level: mb)
grsgwk — redraw all segments on workstation (level: 1a)
gsasf — set aspect source flags (level: 0a)
gschh — set character height (level: ma)
gschm — set choice mode (level: mb)
gschsp — set character spacing (level: 0a)
gschup — set character up vector (level: ma)
gschxp —set character expansion factor (level: 0a)
gsclip — set clipping indicator (level: ma)
gscr — set colour representation (level: ma)
gsds — set deferral state (level: 1a)
gsdtec — set detectability (level:1b)
gselnt —select normalization transformation (level: ma)
gsfaci — set fill area colour index (level: ma)
gsfai — set fill area index (level: 0a)
gsfais — set fill area interior style (level: ma)
gsfar — set fill area representation (level: 1a)
gsfasi — set fill area style index (level: 0a)
gshlit — set highlighting (level: 1a)
gslcm — set locator mode (level: mb)
gsln — set linetype (level: ma)
gslwsc — set linewidth scale factory (level: 0a)
gsmk — set markertype (level: ma)
gsmksc — set marker size scale factor (level: 0a)
gspa — set pattern size (level: 0a)
gspar — set pattern representation (level: 1a)
gsparf — set pattern reference point (level: 0a)
gspkid — set pick identifier (level:1b)
gspkm — set pick mode (level:1b)
gsplci — set polyline colour index (level: ma)
gspli — set polyline index (level: 0a)
gsplr — set polyline representation (level: 1a)
gspmci — set polymarker colour index (level: ma)

gspmi — set polymarker index (level: 0a)
gspmr — set polymarker representation (level: 1a)
gssgp — set segment priority (level: 1a)
gssgt — set segment transformation (level: 1a)
gsskm — set stroke mode (level: mb)
gsstm — set string mode (level: mb)
gstxal — set text alignment (level: ma)
gstxci — set text colour index (level: ma)
gstxfp — set text font and precision (level: 0a)
gstxi — set text index (level: 0a)
gstxp — set text path (level: 0a)
gstxr — set text representation (level: 1a)
gsvis — set visibility (level: 1a)
gsvlm — set valuator mode (level: mb)
gsvp — set viewport (level: ma)
gsvpip — set viewport input priority (level: mb)
gswkvp — set workstation viewport (level: ma)
gswkwn — set workstation window (level: ma)
gswn — set window (level: ma)
gtx — text (level: ma)
gtxs — text (FORTRAN 77 subset) (level: ma)
gurec — unpack data record (level: 0a)
gurecs — unpack data record (FORTRAN 77 subset) (level: 0a)
guwk — update workstation (level: ma)
gwitm — write item to gksm (level: 0a)

DGL/AGP

The DGL/AGP System is a set of tools for display and design graphics. This system consists of two sets of procedure libraries: DGL (Device—independent Graphics Library) and AGP (Advanced Graphics Package). Both can be called from application programs.

DGL is the foundation of DGL/AGP. It provides fundamental graphics functionality and device support. AGP builds upon DGL, providing similar functionality plus enhanced capabilities. For this reason, you must have DGL to operate AGP.

DGL Features

- Device-independent graphics input and output
- Graphics workstation capability for flexible operator environment
- Output primitives: Move, Draw, Polyline, Polygon, Hardware Text, and Alphanumeric Text
- Color modeling capability using either Red/Green/Blue (RGB) or Hue/Saturation/Luminosity (HSL) models

AGP Features

AGP has all the DGL features just mentioned, plus:

- Run-time device independent to allow concurrent operation of multiple workstations, without reprogramming or relinking an application program
- Two- or three-dimensional viewing transformations with parallel or perspective projections
- Picture segment capability for interactive manipulation of graphics images
- Additional pick input function to allow operator interaction with picture segments
- Full clipping capability
- Additional software-generated graphics text with six different fonts.

Graphics Workstation

A graphics workstation is a collection of one or more graphics devices that perform an an assigned function under the control of DGL or AGP. A workstation may perform the following functions: display, locate, valuate, read keyboard input and handle alphanumeric messages. The picking function is available in AGP only.

A Workstation Program (WSP) is a program, configured by the user, which serves as the interface to AGP application programs in controlling graphics devices.

Picture Segmentation and Picking

AGP contains an internal data base called Segment Display Area (SDA). When AGP output primitives are transformed into display images, users may specify groups of output primitives (called segments) to be stored in the SDA. A single image may contain many segments. With AGP, segments may be easily referenced for highlighting or for permanently or temporarily deleting a displayed segment.

AGP supports one level of segmentation which precludes nesting of picture segments. Pick identifiers, however, can be assigned to output primitives within a segment to provide a second level of picture element selection.

HP ALLBASE/HP-UX

HEWLETT PACKARD

For HP 9000 Series 800 Systems

Making It Easy to Have It All

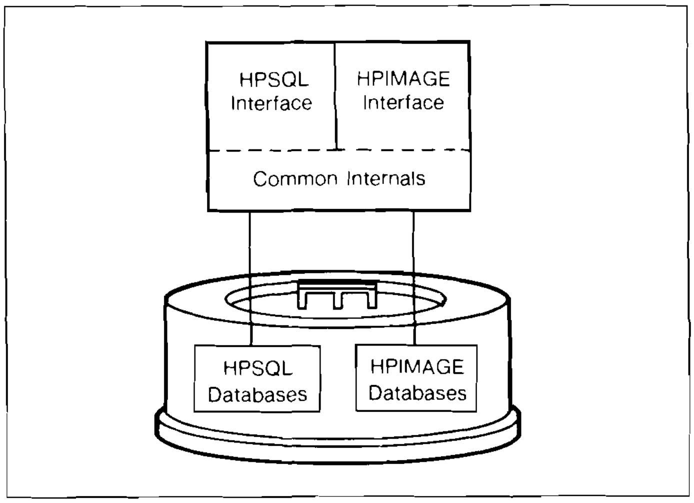
HP ALLBASE is the leading edge database management system (DBMS) which lets you choose the appropriate data model on an application by application basis. With ALLBASE's integrated relational and network data models, you are not forced to fit all of your applications to a single data model, relational or network. Since you are now provided with a choice, the decision of which model to use depends solely on which model provides the optimal foundation for each application.

ALLBASE offers a comprehensive set of functionality whether using HPSQL, the relational model interface, or HPIMAGE, the network model interface. There are many additional features tailored to the specific needs of each interface which may be used with the databases created using that interface. A future release of ALLBASE will allow HPIMAGE databases to be accessed via HPSQL.

ALLBASE is built on a solid foundation of common internals which have been designed specifically to exploit the performance of HP Precision Architecture.

Key Features

- Coexisting relational model interface and network model interface allow the optimal data model to be chosen on an application by application basis.
- Data may be accessed directly from a terminal or via application programs written in common programming languages.
- Security is maintained by allowing specification of appropriate levels of access privileges to individual users or groups of users.
- Concurrent access allows multiple users to access data simultaneously.
- Data independence allows changes to be made to the database structure without requiring that applications be modified.
- User controlled transactions ensure that data is always in a consistent state.
- Automatic locking ensures data integrity in a multi-user environment by preventing access to data while it is being updated.
- Automatic (rollback) recovery ensures logical data integrity in the event of a soft crash.
- The rollback capability may also be invoked in a program to allow erroneous data, usually generated in an on-line situation, to be purged before the transaction is completed.



- Rollforward recovery ensures logical and physical data integrity in the event of a hard crash.
- Fast data access is supported through B-tree indices (HPSQL and HPIMAGE) and hashing (HPIMAGE only).

HPSQL Specific Features

- Relational data model allows the user to specify only what data is required without specifying how to retrieve it.
- HPSQL is based on the emerging SQL ANSI standard for data definition and manipulation.
- Views may be defined which allow a group of users to view parts of one or more tables as a single, virtual table customized to their particular needs.
- Interactive SQL (ISQL) allows the database to be queried from a terminal using the powerful SQL relational language.
- Language preprocessors allow the same statements which are used for a direct query via a terminal to be embedded in FORTRAN, C, or Pascal application programs.
- A sophisticated query optimizer relieves the programmer of the details of query planning without sacrificing performance.
- Dynamic restructuring allows the data structure, table capacities, and security to be changed without unloading and loading the database.
- Null data values allow use of fields which are relevant to some, but not all, records in a table.

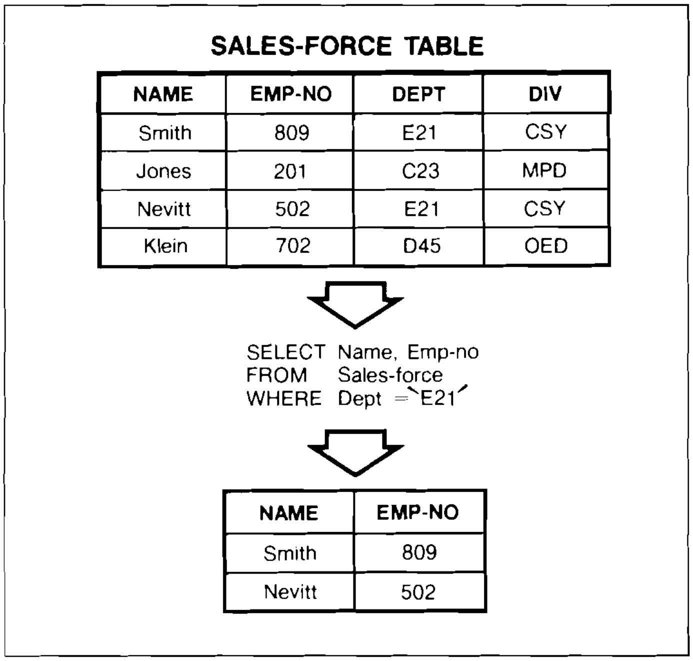
HPIMAGE: Specific Features

- Network data structure allows fast access to complex relationships among data.
- Multiple level relationships allow hierarchical data structures to be easily modeled.
- IQUERY provides the programmer and database administrator with a powerful query facility which includes the ability to report data combined from different data sets.
- IMAGE/1000 Translator facilitates a smooth migration of IMAGE/1000 applications and databases to HPIMAGE.
- High level intrinsics allow applications written in FORTRAN, C, or Pascal to easily access the database.
- Dynamic capacity expansion and security modification allow data set capacities to be expanded and security designations to be modified without unloading and loading the database.
- Generic search allows records to be located by specifying only a portion of the value in a field.
- Referential integrity may be maintained through definition of parent/child relationships.

Functional Description

HPSQL: A complete relational model interface

HPSQL is a functionally complete, relational model interface which features a full implementation of the emerging SQL relational language. SQL is a non-procedural language whose powerful commands operate on entire sets of data at a time, rather than using the procedural, record-at-a-time approach of network and hierarchical model databases.



SALES-FORCE TABLE

NAME	EMP-NO	DEPT	DIV
Smith	809	E21	CSY
Jones	201	C23	MPD
Nevitt	502	E21	CSY
Klein	702	D45	OED

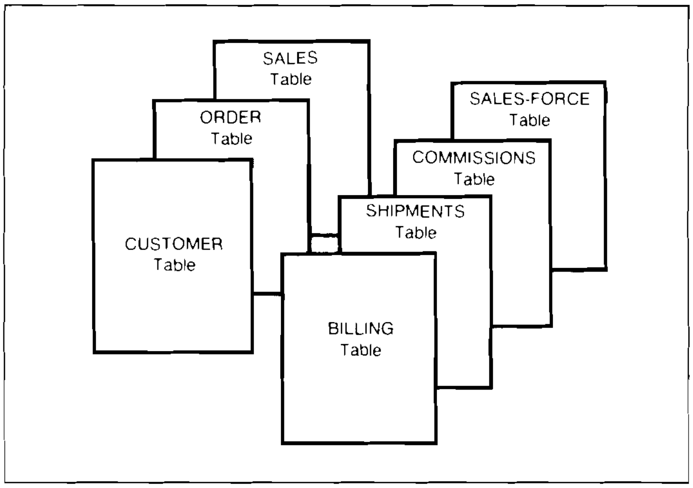
```
SELECT  Name, Emp-no
FROM    Sales-force
WHERE   Dept ='E21'
```

NAME	EMP-NO
Smith	809
Nevitt	502

The Set-at-a-Time Query Approach of HPSQL

The select, project, and join operators of relational theory are supported. These are implemented through the SQL language with the SELECT command for reading data. HPSQL also supports the other three data manipulation language (DML) commands of the SQL language: INSERT, for inserting groups of records; UPDATE, for updating (modifying) groups of records; and DELETE, for deleting groups of records.

HPSQL does not require that explicit relationships between different tables (data sets) be defined. Relationships are determined, at the time a query is performed, by matching values between a column common to two or more tables. Since data from any number of tables which share a common column may be related as needed, an HPSQL database consists of a group of tables which the user decides he may have a need to relate.



Simplified HPSQL Relational Structure

A sophisticated query optimizer determines the optimal data access strategy based on such factors as the presence of indices and the relative sizes of the tables being accessed. This frees the programmer from detailed query planning without sacrificing performance.

Language preprocessors provide easy, high level access to the database via common programming languages. This approach significantly improves performance by allowing query optimization to be performed when the application is compiled instead of at run time.

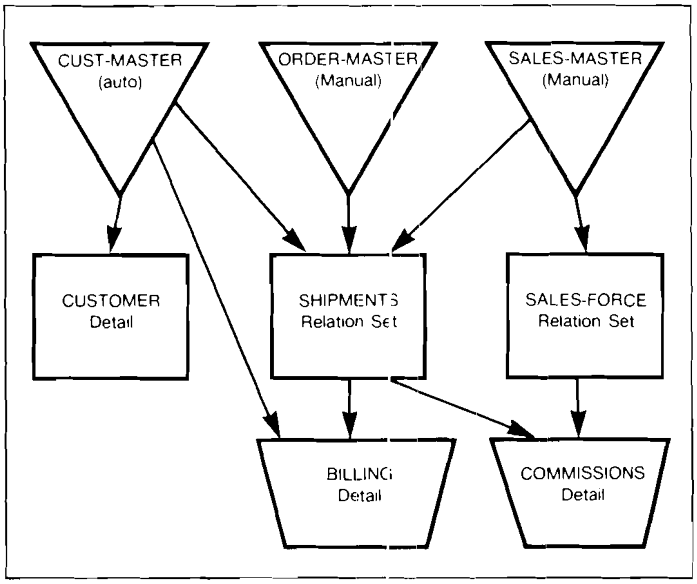
Interactive SQL (ISQL) is the comprehensive interactive interface to HPSQL. There are two distinct sets of functionality which may be used via ISQL. The first is the data definition language (DDL) which allows the database administrator (DBA) to control all aspects of database creation and modification.

The second set of ISQL accessible functionality is a complete implementation of the SQL data manipulation language which allows the programmer or frequent user to query the database from a terminal using the non-procedural SELECT, UPDATE, INSERT, and DELETE commands.

HPIMAGE: A complete network model interface

HPIMAGE is a functionally complete, network model interface which provides functionality beyond that of IMAGE/1000, TurboIMAGE, and IMAGE/9000. With HPIMAGE, relationships between data sets are predefined as part of the database structure (called the schema). These predetermined relationships provide application programs with the fastest data access ALLBASE has to offer. HPIMAGE allows multiple levels of parent/child relationships to be defined. A parent/child relationship, for example, exists between an order data set (parent) and a shipments data set (child) since a valid shipment can only be made against an existing order.

A rich set of relationships may be defined using HPIMAGE: a parent set may have multiple children; a child set may have multiple parents; and a set may be both a parent to some sets and a child to others (multiple levels). With these relationships, HPIMAGE can directly model the information flow of your business.



Simplified HPIMAGE Network Structure

HPIMAGE implements its data manipulation language with a full set of reading, inserting, updating, and deleting capabilities which may be exercised from an application program via intrinsics (library routines).

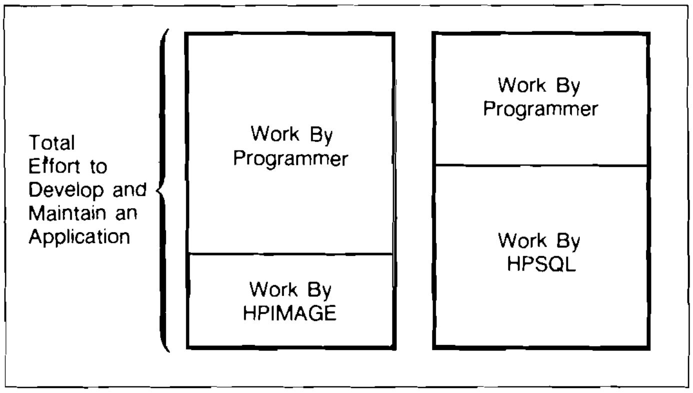
A comprehensive set of data definition language commands and utilities for defining and maintaining the database are provided by HPIMAGE.

HPIMAGE also includes IQUERY, which is a powerful query facility for the programmer and database administrator (DBA).

When to use HPSQL — When to use HPIMAGE

ALLBASE includes a relational model interface and a network model interface because both data models have their place in meeting your business needs.

Applications may be developed in less time with HPSQL than with HPIMAGE. With HPSQL, the programmer must specify only what data is required, not how to access it. This significantly reduces the amount of query preplanning required during application development and maintenance. The non-procedural nature of SQL also results in greater data independence and less application code. The high degree of data independence often allows the database to be restructured without requiring that applications be modified. The reduction in code also improves the productivity of maintenance programmers by allowing application code to be quickly understood and modified when changing business needs require that applications be updated.



Programmer Productivity Advantage of HPSQL

The procedural HPIMAGE interface allows the programmer to specify exactly how the data is to be accessed. While this requires more query preplanning, it allows the sophisticated programmer the granularity of control necessary to obtain optimum performance to support throughput on response-time critical applications.

A future release of ALLBASE will allow HPIMAGE databases to be accessed via HPSQL, although not vice versa. Thus, databases which you will eventually want to access via both interfaces should be created using HPIMAGE.

ALLBASE: Ensuring Data Integrity

Views

HPSQL allows views to be created. A view is a table derived by defining a "filter" over one or more tables to let users or programs view only certain data in the tables. Views improve security by allowing users to access only data for which they have a need to know. Since the view is not actually a physical table, use of views does not result in redundant data. When data in a table is updated, all views which use that data are automatically updated.

Security

Security is maintained by allowing specification of appropriate levels of access privileges to individual users or groups of users.

HPSQL allows read (SELECT) access and write access privileges to be assigned at the table level. Read or write access restriction at a finer granularity than the table level, such as at the column level, may be obtained by defining a view of the table which omits the sensitive information. Modification (UPDATE) authority may be granted at the column level without requiring that a view be specified. Write access may be assigned to allow any combination of the following capabilities: row modification (UPDATE), row insertion (INSERT), and row deletion (DELETE). The DBA assigns access privileges by grouping users with common access needs into authorization groups. The particular read and write authorities are then assigned by the DBA to each of these groups. Users with unique access requirements may also be granted privileges directly.

HPIMAGE allows read and write access to be assigned at the data set and item level. The DBA groups all users into security classes. These classes are analogous to the authorization groups of HPSQL. The DBA specifies in the database schema which security classes have read access, which have write access, and which have both read and write access to each data item and data set. When a user opens the database, the password he gives identifies the security class to which he belongs and therefore his read and write access privileges throughout the HPIMAGE database.

Logical Transaction Defined

A logical transaction is a series of database modifications of which either all or none must be performed to leave the database in a consistent state. The particular grouping of modifications defined by the user to be a transaction will vary depending on the particular application. An example of a transaction is the sign-out of a tool from a tool crib: both the removal of the tool from the crib on-hand inventory records and the addition of the tool to the list of tools assigned to a particular department is required to keep the tool from being either double counted or lost in the system. This concept of a logical transaction is essential to ensuring that data integrity is maintained when multiple users are concurrently accessing the database or in the event of a system failure.

Concurrency

ALLBASE preserves data integrity when multiple users are accessing a database through a comprehensive locking scheme based on the transaction concept. When a user begins a transaction, a lock is automatically granted for each page read or modified by the transaction (a page is a unit of data storage which contains 4,096 characters). This ensures that no one else may update the data on those pages while the user is reading or updating them. If data is only being read, then other users are not prevented from reading it simultaneously; they just can't update it. If data is being updated, however, then data integrity is ensured by preventing the other users from reading or updating the data. When the user's transaction is completed, all acquired locks are automatically released.

Tables (data sets) may also be locked explicitly. This feature is provided to allow the programmer greater flexibility in applications where it is advantageous to lock large portions of the database. Since explicit locks reduce concurrency, they are not recommended for general use.

ALLBASE also contains sophisticated deadlock detection and correction capabilities. When a lock conflict is detected, ALLBASE will rollback one of the transactions, thereby preserving data integrity, and notify the application program that it has done so. This notification allows the application program to determine how to continue processing.

Recovery

ALLBASE ensures that the logical and physical integrity of the database is protected in the event of a program abort, system failure, or destruction of the media on which the database resides.

Rollback recovery is an automatically activated recovery feature which ensures that the database is always in a logically consistent state. ALLBASE logs before-images and after-images for each write transaction to a log file on disc. In the event of a system failure or program abort, ALLBASE uses this log file to automatically back out any partially completed transactions.

The rollback capability may also be invoked in an HPSQL or an HPIMAGE program. This is a particularly valuable feature in an on-line application as it allows a user who has entered incorrect information to nullify the transaction before its completion.

Rollforward recovery protects the physical and logical integrity of the database against media failure. In the event of a hardware or software failure, the transactions from the log file are reapplied to a backup copy of the data to bring it up to the current state.

A simultaneously updated copy of the log files used for rollback or rollforward recovery may be kept on another disc to provide additional protection in the case of a disc failure.

Referential Integrity

The predefined relationships of HPIMAGE allow referential integrity to be maintained. Once a parent/child relationship has been established between HPIMAGE data sets, referential integrity is maintained by preventing the deletion of a parent record without first deleting all corresponding child records and by preventing the addition of a child record unless a corresponding parent record exists. For example, an HPIMAGE database could be easily defined which would prevent deletion of an order record (parent) until all of the shipments against that order (children) have been deleted and which also would prevent any shipment records from being inserted unless they are against a valid order.

ALLBASE: Interactive Data Access

ALLBASE includes two powerful query facilities which aid the application programmer and the database administrator in the creation and support of HPSQL and HPIMAGE databases. Interactive SQL (ISQL), for use with HPSQL databases, and IQUERY, for use with HPIMAGE databases, may be executed interactively (from a terminal) or in a batch mode. Output may be directed to a terminal or to any other output device (i.e., line printer, disc, tape). Frequently used ISQL and IQUERY procedures may be stored on disc for future use.

ISQL and IQUERY both honor the ALLBASE security system. When accessing the database, the user may only exercise the read/write access privileges which have been allowed him in the database definition.

ISQL for HPSQL

ISQL is a database support tool integrated into HPSQL which features a powerful, non-procedural query facility. ISQL allows SQL queries to be executed directly from a terminal, giving programmers and frequent users the ability to use this powerful language for ad hoc queries. Since the syntax of the SQL query is the same whether executed using ISQL or embedded in an application program, the programmer can cut application development time by using ISQL to test database queries before embedding them in applications. ISQL also includes a powerful bulk load facility which is particularly useful when creating test databases.

IQUERY for HPIMAGE

IQUERY is a database support tool included with ALLBASE for use with HPIMAGE. IQUERY features a powerful, command driven interface designed for application programmers and database administrators. IQUERY provides extensive locating, reporting, and updating commands. These commands provide flexible data access on an ongoing basis and are particularly valuable when testing new database applications and structures.

ALLBASE: Access of Data via Programs

Both HPSQL and HPIMAGE databases may be easily accessed from application programs. A list of the high level languages from which each of the databases may be accessed is included in the *Other Specifications* section at the end of this data sheet.

Preprocessors for HPSQL

HPSQL uses language preprocessors to provide easy, high level access to the database via common programming languages. To access an HPSQL database from an application, SQL queries are embedded directly in the source code. Before compiling the application, it is preprocessed (precompiled). Preprocessing replaces the SQL queries with language specific calls and performs the query optimization.

The preprocessor approach improves performance by allowing query optimization to be performed when the application is compiled instead of at run time. At run time, HPSQL will detect if a change in the database structure has invalidated the access strategy for any of the queries and will automatically optimize those queries for the new structure.

Intrinsics for HPIMAGE

HPIMAGE databases are accessed from application programs via intrinsics, which are calls to library routines. Intrinsics provide a straightforward method of database access. This direct access approach is optimal in HPIMAGE since the data access paths are specified in the application program. Intrinsics also ensure maximum compatibility with IMAGE/1000 and IMAGE/9000.

ALLBASE: Creating and Maintaining Databases

Database Creation

As part of the database design process, the DBA must decide how many databases should be included in each ALLBASE DBEnvironment. A DBEnvironment may contain one or more databases. Multiple databases which will be accessed via a single program should be placed in the same DBEnvironment. The DBEnvironment is also the level at which the data is backed up; therefore unrelated databases should be placed in separate DBEnvironments.

After the DBA has designed the database structure on paper, he may easily create the database. The first step in creating a database is the same for HPSQL and HPIMAGE. A DBEnvironment must be configured for the database unless it will be included with other databases in an existing DBEnvironment. The remaining step in creating an HPSQL database is to create the tables (CREATE TABLE command), indices (CREATE INDEX command), and views (CREATE VIEW command) which comprise the database.

The DBA who is creating an HPIMAGE database creates the *schema* for the database after configuring a new DBEnvironment (if required). A schema is a text file which describes the structure of the database. The final step when creating an HPIMAGE database is to run the schema processor which creates the actual database structure from the schema file.

Database Restructuring

ALLBASE provides a full set of database restructuring capabilities. HPSQL and HPIMAGE support dynamic restructuring for commonly required structural changes. Dynamic restructuring allows users to continue to access data, except for the affected areas, during restructuring.

HPSQL provides dynamic restructuring for the following cases: expanding table capacities; altering security designations; adding columns; and adding or deleting indices, views, and tables.

HPIMAGE provides dynamic restructuring capabilities for increasing data set capacities and for changing security designations.

Database Compatibility

HPIMAGE (ALLBASE) and IMAGE/1000

IMAGE/1000 applications can be run on HP 9000 Series 800 systems in many cases with no source code changes, via the IMAGE/1000 Translator in conjunction with the HPIMAGE interface of ALLBASE. The IMAGE/1000 Translator translates, at run-time, the IMAGE/1000 calls from the application into the appropriate HPIMAGE calls. This translation is performed transparently to the application with little overhead. A data migration utility, DBMUN, simplifies data migration by unloading the IMAGE/1000 database for subsequent loading into an HPIMAGE database using the standard HPIMAGE LOAD utility. A third migration aid, DBDECODE, allows an HPIMAGE database schema to be generated from an IMAGE/1000-II (P/N 92081A) root file.

The IMAGE/1000 Translator is included with ALLBASE/HP-UX. DBMUN and DBDECODE are part of the Migration Evaluation Kit (P/N 92561A).

HPIMAGE (ALLBASE) and IMAGE/9000

HPIMAGE provides a superset of functionality over that provided with IMAGE/9000 using similar, but not fully compatible, intrinsic calls. To move IMAGE/9000 applications from a Series 500 to HPIMAGE on HP 9000 Series 800 systems, the IMAGE/9000 calls in the application must be changed to HPIMAGE calls.

ALLBASE/HP-UX and ALLBASE/XL

ALLBASE is the common Hewlett-Packard DBMS across all HP Precision Architecture systems. ALLBASE/XL is the ALLBASE implementation on the HP Precision Architecture-based 900 Series of HP 3000 computer systems. The ALLBASE/HP-UX and ALLBASE/XL data and intrinsic interfaces are fully compatible.

ALLBASE/HP-UX and HPtoday/HP-UX

HPtoday/HP-UX is a high productivity application development environment which can be used to quickly develop menu-driven applications based on HPSQL (ALLBASE/HP-UX) databases. A bundle of ALLBASE/HP-UX and the HPtoday/HP-UX development environment is also available. The HPtoday/HP-UX data sheet provides more information on the capabilities of the HPtoday products.

System Environment

ALLBASE/HP-UX is supported on the HP 9000 Series 800 systems.

Ordering Information

The right to use ALLBASE on the HP 9000 Series 800 systems is available according to the terms and conditions of the Hewlett-Packard Software Purchase Agreement.

Software

Product Number	Product Description
92460A	ALLBASE/HP-UX Right-to-Use on Model 825.
92460R	ALLBASE/HP-UX Right-to-Copy on Model 825 with sublicense.
92459A	Bundle of ALLBASE/HP-UX Right-to-Use and HPtoday Developer Pack/HP-UX Right-to-Use for Model 825.
36217A	ALLBASE/HP-UX Right-to-Use on Model 840.
36217R	ALLBASE/HP-UX Right-to-Copy on Model 840 with sublicense.
92442A	Bundle of ALLBASE/HP-UX Right-to-Use and HPtoday Developer Pack/HP-UX Right-to-Use for Model 840.
92469A	ALLBASE/HP-UX Right-to-Use on Model 850.
92469R	ALLBASE/HP-UX Right-to-Copy on Model 850 with sublicense.
92468A	Bundle of ALLBASE/HP-UX Right-to-Use and HPtoday Developer Pack/HP-UX Right-to-Use for Model 850.

Documentation

Part Number	Document Description
36217-90001	SQL Reference Manual
36217-90002	HPIMAGE Reference Manual
36217-90003	IQUERY Reference Manual
36217-90004	ISQL Reference Manual
36217-90005	HPSQL Database Administration Guide
36217-90013	HPSQL FORTRAN Application Programming Guide
36217-90014	HPSQL C Application Programming Guide
36217-90007	HPSQL Pascal Application Programming Guide
36217-90008	Migrating to ALLBASE
36217-90009	HPSQL Message Manual
36217-90010	HPIMAGE Message Manual
36217-90012	ALLBASE Quick Reference Guide

Customer Training

Twelve classes, each one tailored to the needs of a specific group of ALLBASE users, are available through your local Hewlett-Packard Sales Office. Each course includes instruction and labs.

Course Number	Course Name
36217AJ	ALLBASE/HP-UX Interactive User's Course
36217AK	ALLBASE/HP-UX Database Administrator's Course
36217AL	ALLBASE/HP-UX Application Programmer's Course
36217AM	ALLBASE/HP-UX Database Designer's Course
36217AB	HPSQL/HP-UX Interactive User's Course
36217AC	HPSQL/HP-UX Database Administrator's Course
36217AI	HPSQL/HP-UX Application Programmer's Course
36217AD	HPSQL/HP-UX Database Designer's Course
36217AE	HPIMAGE/HP-UX Interactive User's Course
36217AH	HPIMAGE/HP-UX Database Administrator's Course
36217AG	HPIMAGE/HP-UX Application Programmer's Course
36217AF	HPIMAGE/HP-UX Database Designer's Course

Contact your local Hewlett-Packard Sales Office for course outlines and schedules.

Other ALLBASE Specifications

	HPSQL	HPIMAGE
Data Types	Decimal (packed) IEEE Standard, Double Precision Floating Point (8 byte) Integer (2 & 4 byte) Fixed Length (<3996 bytes) Character Variable Length (<3996 bytes) Character	Decimal (packed & zoned) Floating point (4 & 8 byte) Integer (2 & 4 byte) Fixed Length (<3996 bytes) Character
Languages	FORTRAN C Pascal	FORTRAN C Pascal
Passwords/Security Groups (Classes) per Database	Unlimited Groups	63 Classes
Maximum Database Parameters		
Tables (sets) per database	$2^{31}-1$	199
Records per table (set)	$2^{31}-1$	$2^{31}-1$
Record Length	3996 bytes	3996 bytes
Columns (items) per table (set)	255	255
Field (item) length	3996 bytes	3996 bytes
Sub-items per item	n/a	255
Children per Parent	n/a	31
Indices per Table	$2^{31}-1$	n/a
Search Items per Set	n/a	31
Columns per Index	15	n/a
Sort Items per Path	n/a	8
Concurrent transactions per database	240	240

HP Visor/HP-UX

HEWLETT PACKARD

For HP 9000 Series 800 Systems

End User Interface to HPSQL Databases

HP Visor/HP-UX is an easy-to-use, terminal-based interface to ALLBASE/HP-UX (HPSQL) databases. Visor's flexible selection of access and reporting modules enables end users or database administrators to perform queries and to generate reports without programmer involvement. Visor also provides powerful programmer facilities to improve productivity when working with HPSQL databases.

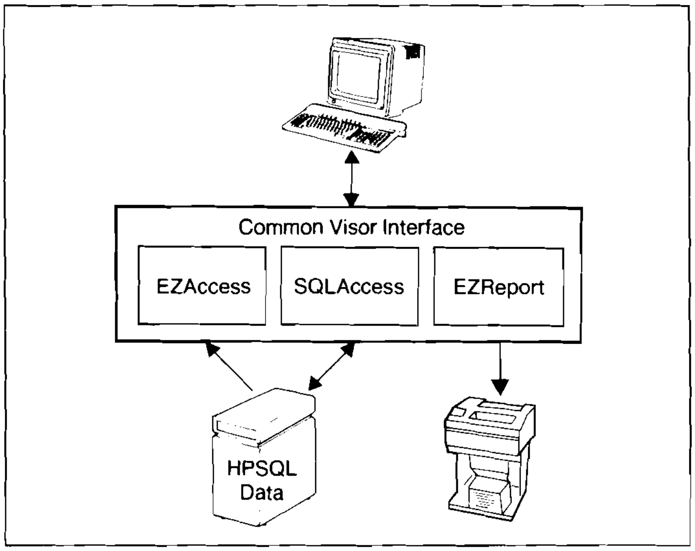
The Visor interface is forms-based, so that the user only needs to mark boxes and press function keys to step through tasks such as performing ad hoc queries or generating customized reports. Additional capabilities are provided for experienced users to directly enter Visor or SQL commands.

Overview of Modules

The three modules integrated with the Visor interface are called EZAccess, SQLAccess and EZReport. EZAccess enables novice and infrequent users of the database to easily generate queries. SQLAccess enables users familiar with HPSQL to use a screen-based editor to formulate industry standard SQL queries and perform a variety of other database operations. EZReport formats and displays query results from EZAccess or SQLAccess, and allows users to customize report formats and calculate additional statistics using the data retrieved by the query.

Key Visor Features

- A common, forms-based interface is presented to the user independent of the module in use so that related tasks are performed in a consistent manner.
- Context-sensitive help, available at any time with a single keystroke, allows quick question resolution.
- Two access methods are available: menu-driven EZAccess for novice or infrequent database users and SQLAccess for users familiar with both HPSQL and the database structure.
- Queries, reports and sequences of commands can be saved for future use, thus streamlining frequently performed tasks.
- Data security is maintained since Visor relies on the HPSQL system catalog to determine each user's authority to access data.
- Default mode of operation shields novice users from complex functions and smoothly steps them through tasks that cross module boundaries.



- Experienced users can run Visor in the mode that allows them to easily select advanced features.
- Batch mode operation is available to automate time consuming tasks.
- Data supported in all languages with eight bit character sets: USASCII, ROMAN8 and KANA8.

EZAccess: Query Tool for Novice and Infrequent Database Users

With EZAccess, novice users and infrequent users of the database can quickly and easily learn to formulate queries and, in conjunction with EZReport, generate customized reports. No prior knowledge of either the database structure or a data manipulation language is required.

EZAccess steps users through the process of specifying the desired database, tables, columns, search values, and sort columns; the user only needs to choose among options. Visor's context-sensitive help is available at all times with a single keystroke.

If the user would like to learn SQL, the SQLAccess module can be invoked and the current query, generated by EZAccess, will be displayed as a modifiable SQL command.

EZAccess Features

- Novice and infrequent database users are stepped through the process of creating a query so the expertise of a programmer is not required.
- Query results can be displayed immediately, and then iteratively refined until the desired query has been obtained.
- New, calculated data columns can be defined.
- EZAccess makes "common sense" default assumptions about the fields on which tables should be joined.
- EZAccess familiarizes users with SQL by generating an HPSQL query that the user can view and modify if desired.

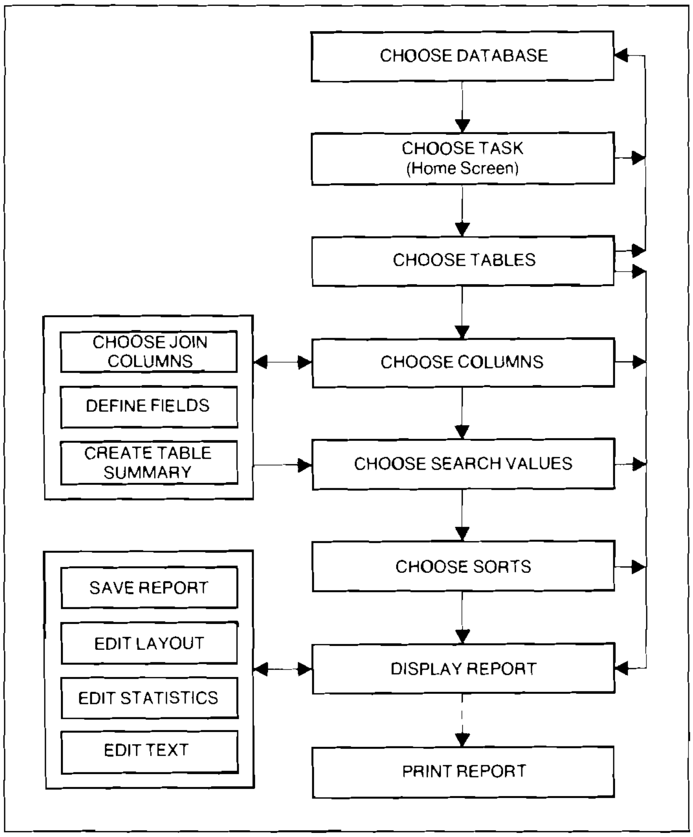
SQLAccess: Screen-based Editor for SQL Commands

Users familiar with HPSQL can use SQL commands to retrieve and modify data, create new database objects and maintain the database from within Visor. Commands created using the SQLAccess screen-based editor can be executed immediately, or named and saved for future use or modification.

In SQLAccess, SQL commands that perform a variety of functions beyond the capabilities of EZAccess can be created by programmers and saved for use by database end users. Commands can contain variables. Visor will prompt the user to input values when the command is executed.

SQLAccess Features

- Screen-based editor for SQL commands boosts programmer productivity with an easy way to design and test commands.
- Commands can contain variables, allowing programmers to design UPDATE, INSERT and complex SELECT commands for end users.
- New tables and views can be created without leaving the Visor environment.



Activity flow for ad hoc query, using EZAccess and EZReport. To progress along the solid vertical path (top to bottom) the user simply marks choices in boxes on the screen and then presses the function key marked "Next Step." At any point, the "Help" function key is available to provide context-sensitive help.

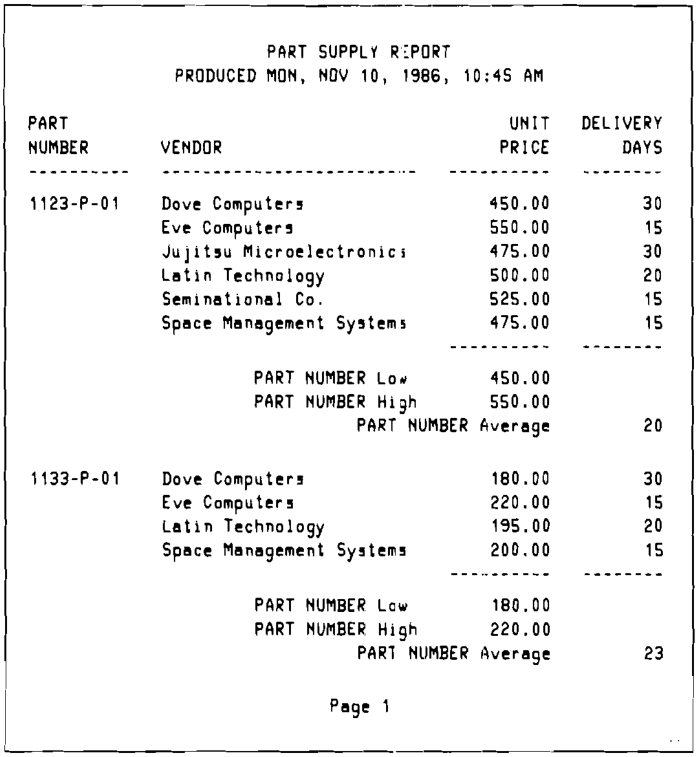
EZReport: The Easy Way to Create Customized Reports

Once data has been retrieved by EZAccess or SQLAccess, EZReport displays the data on the screen using a default format. From that point, the report may be customized in a variety of ways: the page layout may be changed so that columns appear in a different order or alignment, multi-line headings and footers can be added or changed, or a variety of statistics (average, total, high, low, count) can be calculated and displayed at the end of the report or at specified sort breaks within the report. All of these choices can be accomplished with a few keystrokes, as the user only has to selectively edit the default specifications provided on full screen displays.

The reports resulting from successive modifications are displayed only on the screen until the user tells EZReport to print a hard copy or save the output to a text file. With EZReport, programmers can create and save complex report specifications for end users, and end users can easily generate their own ad hoc reports.

EZReport Features

- Novice database end users can design and print their own customized reports.
- Users can check to see that the correct data has been retrieved before taking the time to customize the report design.
- EZReport provides a default report design for novice users while experienced users can easily change the layout and add headings and footers.
- Statistical functions can be calculated for any column and displayed at the end of the report or at selected sort breaks.
- The most current data is retrieved from the database each time a saved report is run.
- Reports can be printed at a system printer or local printer, or into a text file for inclusion in a document.



```
                   PART SUPPLY REPORT
           PRODUCED MON, NOV 10, 1986, 10:45 AM

PART                                      UNIT   DELIVERY
NUMBER     VENDOR                        PRICE       DAYS
---------- ------------------------ ----------   --------
1123-P-01  Dove Computers               450.00         30
           Eve Computers                550.00         15
           Jujitsu Microelectronics     475.00         30
           Latin Technology             500.00         20
           Seminational Co.             525.00         15
           Space Management Systems     475.00         15
                                    ----------   --------

                   PART NUMBER Low      450.00
                   PART NUMBER High     550.00
                             PART NUMBER Average       20

1133-P-01  Dove Computers               180.00         30
           Eve Computers                220.00         15
           Latin Technology             195.00         20
           Space Management Systems     200.00         15
                                    ----------   --------

                   PART NUMBER Low      180.00
                   PART NUMBER High     220.00
                             PART NUMBER Average       23

                        Page 1
```

Sample report created using EZReport

Macro-like Facility to Save Work

To enhance the productivity of all users, the Visor interface maintains four types of items: PROFILE, SQLCOMMANDS, REPORT and SCRIPT.

The PROFILE allows users to customize the Visor interface to meet their individual needs. It includes specifications for the access method (EZAccess or SQLAccess), the numeric and string null value display characters, the destination and page size for print commands, and the option of whether or not to confirm before performing tasks that alter the contents of the database.

An SQLCOMMAND is a query or other database command expressed in SQL (Structured Query Language). The command can be manually entered using the SQLAccess module or generated automatically by use of the EZAccess module.

A REPORT is the design of a report. It contains an SQLCOMMAND to retrieve the desired data, along with specifications for the page layout, additional text, and statistical calculations to be performed on the data. Each time a saved REPORT is used, the most recent data is retrieved from the database.

A SCRIPT contains a series of Visor commands to be executed in consecutive order. To minimize duplication of effort, the commands in the SCRIPT can include references to save items, such as SQLCOMMANDs, REPORTs, and other SCRIPTs. SCRIPTs are particularly useful for automating routine functions, such as generating month-end reports and printing them at several different printers.

Visor enables the user to iteratively refine SQLCOMMANDs, REPORTs and SCRIPTs, view the results immediately, and then save the items for future use or modification. While only one PROFILE per user can be active at a time, numerous SQLCOMMANDs, SCRIPTs, and REPORTs can be named and saved for private use or to be shared with other users.

Specifications

HP Visor (maximum)

Lines per script	24
Variables per script	10
Number of saved items †	unlimited

EZAccess (maximum)

Table columns per query	15
Computed columns per query	4
Tables per join or query	3
Columns sorted per query	9

SQLAccess (maximum)

Table columns per query	15
Computed columns per query	10
Tables per join or query	10
Lines per command	24
Variables per command	10
Columns sorted per query	9

EZReport (maximum)

Width of column (chars)	240
Width of page or report	240
Lines per page ‡	999
Length of report (chars)	2.56M

† Bounded by the size of the database (DBEnvironment).
‡ Continuous output capability also provided.

System Environment

HP Visor/HP-UX runs on HP 9000 Series 800 systems and requires ALLBASE/HP-UX. Visor/HP-UX operates only on HPSQL-accessible databases. Visor/300 and Visor/XL SQLCOMMANDs, REPORTs and SCRIPTs are compatible with Visor/HP-UX.

Consult your sales representative for information about the availability of Visor on HP 3000 systems.

Current supported display terminals include the 239X family and the HP 150 (Touchscreen Personal Computer). The HP Vectra Personal Computer is supported via AdvanceLink (terminal emulation mode).

Ordering Information

Software

The right to use Visor on the HP 9000 Series 800 systems is available according to the terms and conditions of the Hewlett-Packard Software Purchase Agreement.

Product Number	Product Description
92533A	HP Visor/HP-UX Right-to-Use on Model 825
92533R	HP Visor/HP-UX Right-to-Copy on Model 825
92534A	HP Visor/HP-UX Right-to-Use on Model 840
92534R	HP Visor/HP-UX Right-to-Copy on Model 840
92535A	HP Visor/HP-UX Right-to-Use on Model 850
92535R	HP Visor/HP-UX Right-to-Copy on Model 850

Documentation/Training

Part Number	Document Name
92534-6001	HP Visor/HP-UX Tutorial

This document primarily provides self-paced training for new Visor users. A reference information section is included. In addition, most of the information needed by users who have completed the tutorial is always at their fingertips with HP Visor/HP-UX's on-line help facility.

HPtoday/HP-UX

HEWLETT PACKARD

For HP 9000 Series 800 Systems

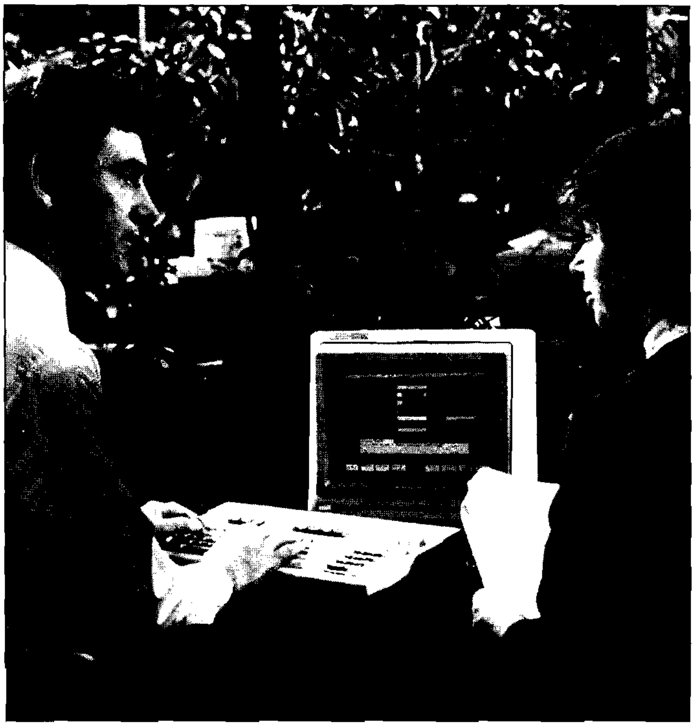
Product Numbers HP 92440A, 92441A

The Programmer's 4GL

HPtoday is an advanced Fourth-Generation Language (4GL) for developing data- or transaction-processing applications. For the professional software developer, HPtoday significantly improves productivity, compared with conventional methods. And HPtoday's simplified application maintenance can be expected to provide even more significant productivity gains.

HPtoday gives you the ability to specify what an application is to do, without concern for the detailed steps required to do it. Instead of writing reams of source code, with the HPtoday Developer Package you fill in the blanks on formatted screens. This specification process maintains your control over application development, while HPtoday does most of the work. For end-users, the HPtoday Run-Time Environment provides everything required to run developed applications.

HPtoday supports alternative data-bases. You can specify the embedded multi-key indexed file system, or you can utilize the integrated HPSQL interface to access relational databases under ALLBASE/HP-UX. HPtoday also offers a powerful external interface to applications written in other languages.



Key Features

- Computer-assisted development of applications, significantly increasing programmer's productivity.
- Fully integrated development environment that requires no additional editors, compilers, or debugging utilities.
- Allows easy, interactive development of applications in consultation with end users (prototyping).
- Dictionary for definition of all field specifications, messages, records and files, simplifies development and significantly reduces maintenance time and effort.
- Screen Painter, where the process of design simultaneously programs application screens and menus — automatically.
- Report Writer for interactive development of reports, with full control over layout and content.
- Comprehensive logic commands for procedural logic, plus decision tables for complex conditional expressions.
- External program interface, with full parameter passing, allows access to existing packages or other real-time or computationally-intensive code written in conventional languages.
- Application tailoring supports multiple versions of applications: changes to common areas of parent application are automatically applied to versions.
- Security locks can selectively protect part or all of application from unauthorized change.
- Background tasking is supported.
- Run-Time Environment supports delivery of completed applications, and provides System Administrator functions including end-user security control.
- Full access to ALLBASE/HP-UX (HPSQL) databases.
- Embedded multi-key indexed file subsystem for standalone applications.
- Transaction-based automatic record locking prevents data corruption resulting from concurrent updating.
- Transaction logging allows transaction roll-back under program control and allows roll-forward recovery in the event of system failure.

Functional Description

There are two versions of HPtoday — the HPtoday Developer and the HPtoday Run-Time Environment. (See the figure below.)

The Developer Package

The HPtoday Developer has a complete set of facilities for defining, testing and maintaining applications integrated into a single cohesive package. Developer facilities include the Dictionary, development of Screens and Reports, Logic Commands, and Application Testing, together with all the facilities of the Run-Time Environment (as discussed in the following paragraphs).

Dictionary

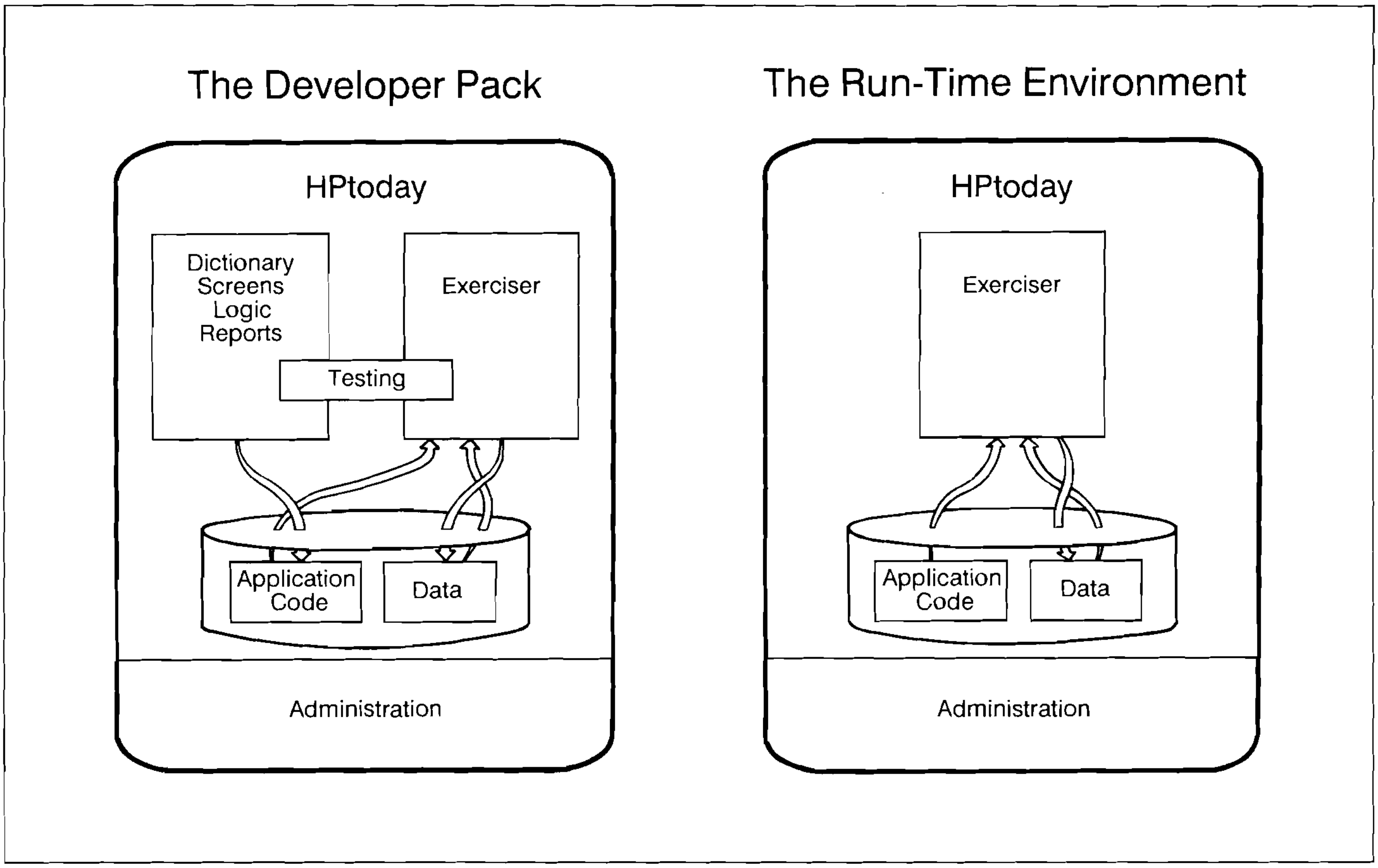
The Dictionary allows central definition of all field specifications, record layouts, files, SQL tables and select lists, messages, help screens, validation tables and ranges, variables, constants, and scratch-pad fields. While the available default values make the definition process very quick and easy, considerable detail can be specified if desired. Changes made to dictionary specifications can be generated throughout an application, and file reformatting is supported.

Screens

Programmers use the Screen Painter to create menu screens and data input/output screens, exactly as they will appear in the application. HPtoday provides all the logic required to drive the designs. Screens can be designed and programmed in less time than it takes to design them using conventional methods. Screen modifications take even less time and effort.

While HPtoday ensures a simple and consistent user interface, the appearance of an application remains in the control of the designer. Programmers can define data screens with overlay windows and/or an automatic scroll area. They can also control the field tabbing sequence, field validation, data movement and any special logic to be performed for each field. Text literals, line drawing characters, and a variety of field types can be created, moved, copied or deleted using function keys and cursor control keys. Screen field definitions can be called up from the Dictionary, or created within the Screen Painter.

HPtoday provides for a high quality user interface. HPWindows is supported on bit mapped monitors; color terminals are also fully supported (see "System Requirements" at the end of this datasheet). Selection of display highlighting/color is under system administration control.



Two Versions of HPtoday

Reports
Report layouts can be designed on screen, with your terminal as a moveable window onto the report. Information for a report can be drawn from many files, linked and sorted as specified on separate formatted screens. There is a selection of report line types and control breaks, and the programmer may even nominate when special stationery is to be loaded into the printer.

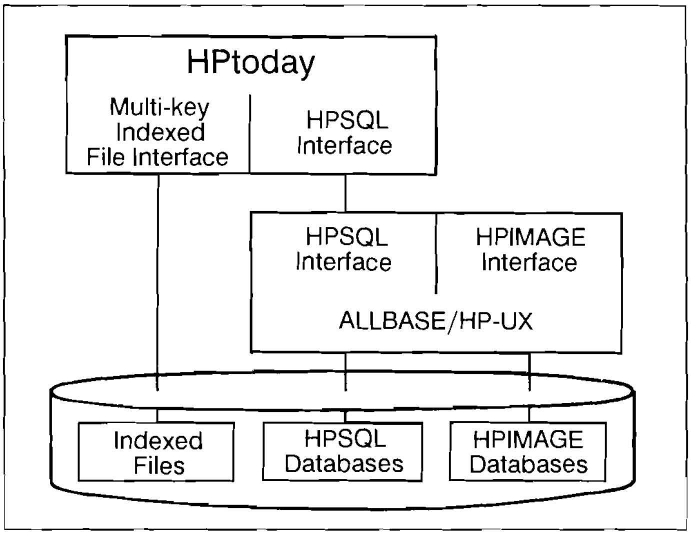
Logic Commands
A comprehensive set of logic commands provides full procedural control of applications. The commands are used in short logic blocks called Processes and Functions, analogous to programs and sub-routines respectively. Formatted input screens interactively indicate the parameters and syntax required for each command. In addition to the facilities expected of any programming language, HPtoday provides decision tables for clear and concise definition of complex conditional structures.

When used with ALLBASE, HPtoday also supports programmatic access of the relational database for processing or reporting.

The Run-Time Environment

The HPtoday Run-Time Environment includes everything the end-user needs to exercise (but not modify) developed HPtoday applications. It is priced to facilitate bundling with completed application(s). Applications are essentially sets of parameters, screen layouts, and report layouts. The Run-Time Environment has all required library routines and underlying logic for running the applications. It also includes a set of administrator facilities.

System Administration
An Administrator application included with HPtoday protects the user from the complexities of the operating system. Routine system operations like file backup and print scheduler management can be controlled with the menu-driven operator interface. It allows setting system-wide parameters such as date and currency formats, numeric separators, etc. It also has facilities for controlling user access to HPtoday or any portion of an application.



Data Storage

The HPtoday Developer and Run-Time Environment (from Release A.01) both support alternative data file management systems.

For a stand-alone application, or where particular programmatic control or tuning of the data base is required, HPtoday includes a multi-key indexed file subsystem. This system provides transparent transaction-based record locking to preserve data integrity, and also allows programmatic locking of files. Inherent transaction logging allows programmatic roll-back if a complex transaction fails, and optionally (under end-user control) allows roll-forward recovery in the event of system failure.

HPtoday also provides an HPSQL interface, allowing HPtoday applications to operate with true relational databases created and maintained under HP ALLBASE. The power of the relational DBMS further simplifies application development, so that tailor-made user interfaces to relational databases now become practical. (See diagram.)

An application can be developed using either data file subsystem, and later be modified with minimum effort to grow to the other as needs change.

Using HPtoday

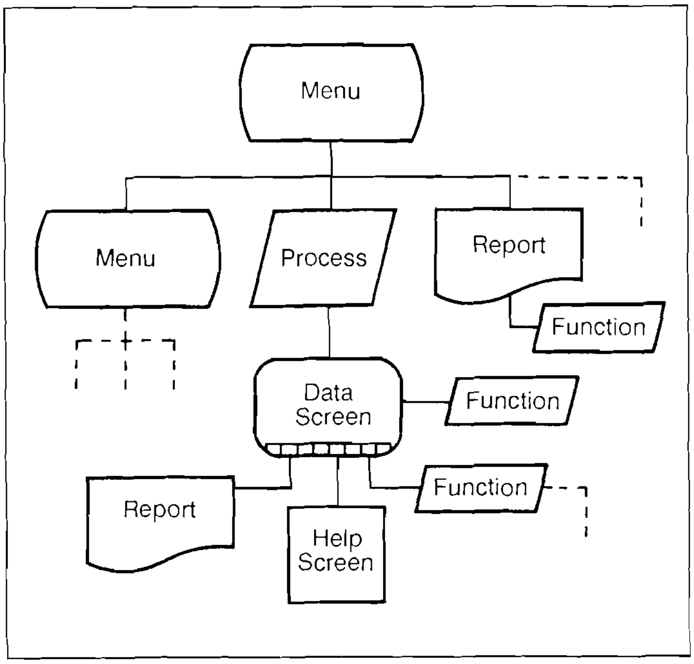
Application Design

As always, design begins with analysis of end-user needs, but instead of designing on paper, the programmer can use the HPtoday Screen Painter and Report Writer to design interactively — and in the process, see built — the complete external interface. This external 'bodywork' may include all menu screens linked to data input and inquiry screens, fundamental field definitions, function keys and report layouts. Help screens can also be included, to document how each screen will work.

The end-user (client) can review this framework to ensure its suitability. Changes to the design can be made interactively until the user is satisfied. Then the programmer can add more detailed dictionary specifications, logic commands, report definitions, etc. This quickly converts the framework into a completed application.

HPtoday also allows a more complete prototyping method, where the end-user and the designer-programmer form a close-knit team. A core sub-set of the application, including logic, can be prototyped reiteratively, with the user providing real-world testing and even external documentation (developer documentation is provided within the application by HPtoday). When this core is satisfactory, the application can be extended in a similar method to include all desired features.

The flexibility and productivity of HPtoday allow developers to formulate the development strategy most appropriate to each situation.



A Typical Application Structure (Simplified)

Application Structure

The component elements of an application include menus, data screens, reports, and logic blocks, supported by data specifications, help screens and messages. Any element can be linked directly to a menu selection, function key, or logic block. Functions can also be linked to fields, on data screens or reports, for execution before and/or after the field is entered.

The concept of linked elements is central to HPtoday, and allows the designer considerable flexibility. For instance, in interactive applications the screens typically control most of the logic. A menu tree may link the screens, or an application may be designed to move freely from one screen to the next. For background (batch) applications, the Process logic usually controls and links the various elements.

Application Development

The HPtoday Developer is designed for easy learning and intuitive use. Menus suggest a rational approach to application development; formatted input screens provide appropriate defaults based on previous inputs; and function keys lead quickly to a likely next task.

Yet HPtoday is not prescriptive of the development approach. Application elements may be specified as the need arises.

Development accuracy is supported by the element checking which HPtoday performs at development time, together with the variety of element listings and debugging facilities available.

Application Documentation

Besides a remark command within logic, HPtoday Developer provides for a short description of elements within an application. The description can be printed with a full listing of the element, or as a summary without details. Documentation is automatically time-stamped. HPtoday can also provide catalogues of application elements, by element type or for a complete application.

As part of the user interface, both in the Developer and within completed applications, HPtoday maintains several message types and multiple, context-sensitive help screens to provide on-line user documentation.

Application Testing

The programmer can test an application within the development environment. Components of an application can be tested in isolation, or the application may be run as if by the end-user. The option of step-tracing is always available.

Application Maintenance

Maintenance and customization of applications are simplified by the Screen Painter, Report Writer, Dictionary facilities and the facility for creating and maintaining versions of applications.

When field lengths or record layouts need to change, HPtoday will adjust all logic and screens at the touch of a key. When several versions of an application are required, such as when tailoring to suit multiple clients, HPtoday keeps track of the differences. Any changes made to the common portion of the parent application are automatically reflected in all versions.

External Interface

Independent programs or routines written in conventional languages such as Cobol, Pascal and C can be linked into HPtoday applications. Parameters and data can be passed both ways. HPtoday can therefore be used for all data handling and user-interface portions of technical applications that have heavy computation or real-time components that might be better handled with other languages. The external interface can also be used to initiate other off-the-shelf Unix system applications, shell scripts, etc. HPtoday can thus be used as a delivery vehicle for a total solution incorporating several applications under one consistent user interface.

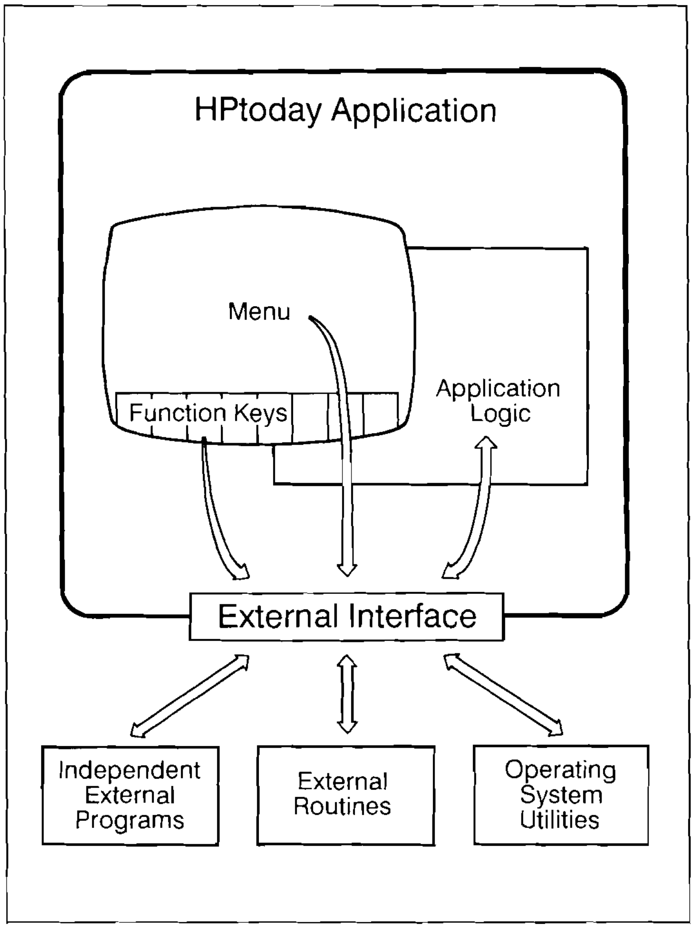
Training and Documentation

HPtoday is a powerful and sophisticated application development package designed for professional analysts and programmers. As with any good tool it can be used more productively by first investing some time and effort in training. However, the cost and disruption of training is minimized by the use of self-paced training.

The HPtoday Developer Pack includes a Developer Self-paced Training Guide, which leads a programmer through the entire process of developing a new application.

The Developer Pack also includes an Installation Guide, a Developer Administration Manual, and a comprehensive Developer Reference Manual. A newsletter of tips and techniques is also available. The HPtoday Run-Time Environment is provided complete with a Run-Time Administration Manual, and Installation Guide.

To assist with end-user training, new users can work in HPtoday's training mode, where they can learn to use a completed application without risk of modifying stored data.



Flexibility of the External Interface

Other HPtoday Specifications

Dictionary Elements	Field Specification, Range, Table, Variable, Calculated Item, Numeric Constant, Alphanumeric Constant, Scratch-pad Field, Record, File, SQL Table, Select List, Help Screen, Message, Application Title. Dictionary facilities include File/Table Creation and Deletion, and File Reformat.
Screen Development	Screen Description, Screen Painter, Screen Field Specification, Function Keys.
Screen Painter Functions	Text Field, Input Field, Output Field, System Item, Line Drawing, Move Field(s), Copy Field(s), Delete Field(s), Column Align, Define Block, Move Block, Copy Block, Delete Block, Menu Action, Renumber Fields, Auto Number Fields.
Screen Field Validation	Character Type, Required Field, Minimum Length, Justification, Blank When Zero, Pad Character, Table Lookup, Range Check, Before/After Function, Auto Move to/from any HPtoday data element.
Report Specifications	Sort, Link Files, Select Records, Control Break, Total, Before/After Print Function, Report Painter, Page Headings, Column Headings, Detail Lines, Extra Lines, Sub Heading Lines, Cross Add, Justification, Field Mask, Underline.
Report Painter Functions	Text Field, Output Field, Zero Suppression, Floating Currency Symbol, Negative indicator (hyphen before, hyphen after, CR after, parentheses), Move Field, Copy Field, Delete Field.
Logic Commands	BACKGRND, CALC, CHECK, CLEAR, DATE, DECISION, DEFINE, DISPLAY, ENTER, EXIT, EXTERNAL, FILE, IF, IFLOOP, LENGTH, LINK, LINKLOOP, MATH, MATHLOOP, MESSAGE, MODE, MOVE, MOVELOOP, NOTE, OFF, ON, PRINT, PROCEED, REPORT, SCREEN, SCROLL, SELECT, SERIES, SHOW, SQL, TIE, TOP, TRANSACT, UPDATE, VALIDATE, VISIT, WINDOW, ZIP.
Supported SQL Commands (summary)	ADD, ALTER, BEGIN, CHECKPOINT, COMMIT, CREATE, DELETE, DROP, EXECUTE, GRANT, INSERT, LOCK, NOTE, REMOVE, RESET, REVOKE, ROLLBACK, SAVEPOINT, SELECT, SQLEXPLAIN, TERMINATE, TRANSFER, UPDATE.
Developer Utilities	Generate Application Items, Generate All, Copy Application Items, Print Application Items, Print All Application, Delete Application Items.
Administrator Functions	Operator Functions, Application Definition, Version Definition, User Definition and Security, Terminal Display Characteristics, System Specifications, Utilities, Data File Logging, File Backup/Restore, Data File Recovery, System Status Inquiry, Process Termination, Printer/Report Status Inquiry, Menu Item Security, User Security, Application Load/Unload.

Limits

Elements per Application	Processes	4000
	Functions	4000
	SQL Logic Blocks	4000
	Menu and Data Screens	4000
	Reports	4000
	Messages	4000
	Help Screens	4000
	Function Key Sets	4000
	Decision Tables	4000
	Record Layouts	4000
	Field Specifications	4000
	Work Areas	999
	File Definitions/SQL Tables/Select Lists	255
	Ranges	255
	Validation Tables	255
	Variables	255
	Alphanumeric Constants	255
	Numeric Constants	255
	Calculated Items	255
	Scratch-Pad Fields	99
Other Ranges	Developer Names, User Names, and User Groups	254
	Record Layouts per File	12
	Fields per Record Layout	255
	Key Fields per Record Layout	19
	Characters per Record Field	999
	Fields per Screen	99
	Characters per Screen Field	80
	Lines per Report Definition	1024
	Fields per Report	255
	Characters per Report Field	255

System Requirements

HPtoday runs under HP-UX on the HP 9000 Series 800, 500, and 300 computer systems. A recommended minimum configuration for a Developer version includes three megabytes of RAM, 55 megabytes of disc storage, one terminal and one printer.*

The HPtoday Run-Time Environment will run on a two megabyte system, with performance dependent on the application and available disc storage.*

For the HP 9000 Series 500, HP-UX version 5.1 or later, and in all other cases, HP-UX version 5.15 or later, is required.

*DMA and high-speed HP-IB are highly recommended on all configurations.

Peripherals

The following peripherals are supported:

Terminals: (including color models)
HP2622 to HP2628
HP239x
HP150 (I & II)
Vectra PC (with terminal emulation software)
Any configuration running HPWindows/9000
Any HP terminal adhering to the TERM0 standard.

Printers: Any printer supported by the HP-UX scheduler lp.

Disc and Tape Drives: Any device supported by HP-UX, other than the auto-change facility of HP 35401A.

Ordering Information

HPtoday Developer Pack

Part Number	Description
HP 98112A	For HP 9000 Series 300
Opt. 022	Media: ¼-inch CS/80 cartridge tape
Opt. 045	Media: 3½-inch disc, double-sided
HP 98113A	For HP 9000 Series 500
Opt. 022	Media: ¼-inch CS/80 cartridge tape
	For HP 9000 Series 800
Opt. AA0	Media: ¼-inch CS/80 cartridge tape
Opt. AA1	Media: ½-inch magnetic tape (1600 bpi)
HP 35305A	Model 825
	Model 840
HP 92440A	Right-to-Use
Opt. 0C8	Upgrade from Model 825 to Model 840
HP 92440R	Right-to-Copy with sublicense
	Model 850
HP 35307A	Right-to-Use
Opt. 0C8	Upgrade from Model 825 to Model 850
Opt. 0C9	Upgrade from Model 840 to Model 850
HP 35307R	Right-to-Copy with sublicense

NOTE: Media option must be specified.

HPtoday Run-Time Environment

Part Number	Description
HP 98118A	For HP 9000 Series 300
Opt. 022	Media: ¼-inch CS/80 cartridge tape
Opt. 045	Media: 3½-inch disc, double-sided
HP 98119A	For HP 9000 Series 500
Opt 022	Media: ¼-inch CS/80 cartridge tape
	For HP 9000 Series 800
Opt. AA0	Media: ¼-inch CS/80 cartridge tape
Opt. AA1	Media: ½-inch magnetic tape (1600 bpi)
HP 35306A	Model 825
HP 92441A	Model 840
HP 35308A	Model 850

NOTE: Media option must be specified

Bundles

Part Number	Description
	For HP 9000 Series 800 HPtoday Developer Right-to-Use product with ALLBASE/HP-UX Right-to-Use product
Opt. AA0	Media: ¼-inch CS/80 cartridge tape
Opt. AA1	Media: ½-inch magnetic tape (1600 bpi)
HP 92459A	Model 825
HP 92442A	Model 840
HP 92468A	Model 850

NOTE: Media option must be specified

HPtoday Manuals

Note: HPtoday products listed at left include one set of appropriate manuals. Additional copies may be ordered with the following part numbers.

Part Number	Document Description
HP 92440K	HPtoday Developer Manual Set includes: HPtoday Developer Administration Manual HPtoday Developer Reference Manual HPtoday Developer Self-paced Training Guide
92440-64001	HPtoday Developer Administration Manual
92440-65005	HPtoday Developer Reference Manual
92440-64003	HPtoday Developer Self-Paced Training Guide
92441-64001	HPtoday Run-Time Administration Manual

For Further Information

The HPtoday Product Evaluation Guide (part number 5954-6324) provides a more detailed overview of HPtoday. Ask your HP Sales Representative for a copy.

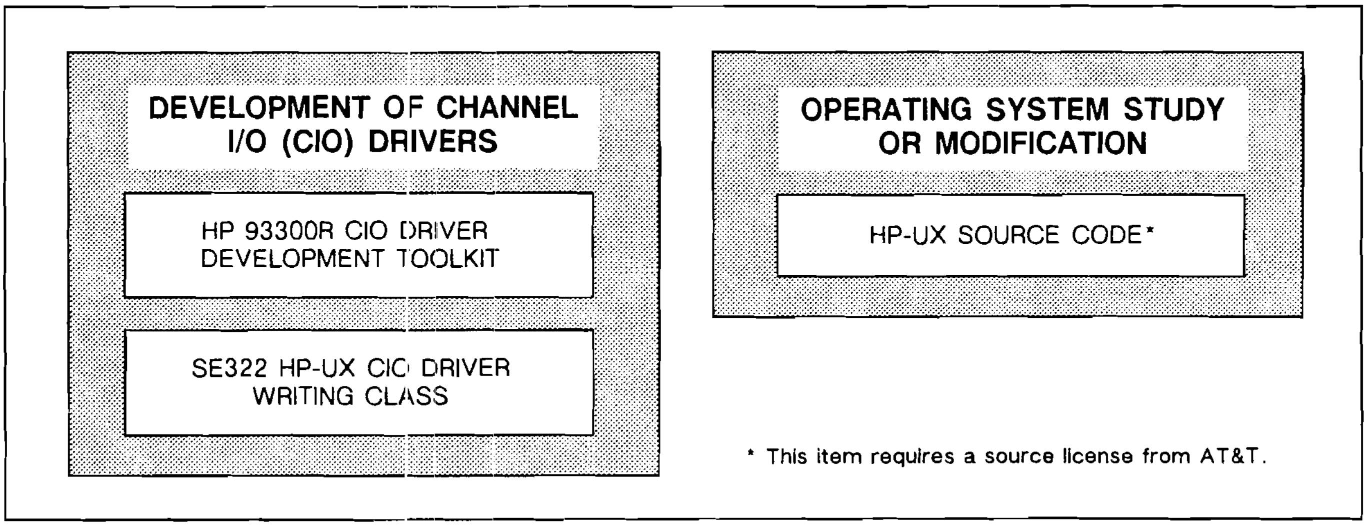
The following documents should be consulted for additional information on HP 9000 systems and HP-UX. Ask your Hewlett-Packard sales representative for further references.

Part Number	Document Description
5954-7066	HP-UX Operating System
5953-9572	HP 9000 Series 300 Hardware Technical Data
98561-90020	HP 9000 Series 300 Configuration Reference Manual
09050-90050	HP 9000 Series 500 Configuration and Order Guide
5954-9901	HP 9000 Series 800 Computer Systems Hardware Technical Data
5954-9902	HP 9000 Series 800 Computer Systems Software and Communications Technical Data

Open System Tools

HEWLETT PACKARD

For HP 9000 Series 800 Systems Models 825S and 840S



DEVELOPMENT OF CHANNEL I/O (CIO) DRIVERS

HP 93300R CIO DRIVER DEVELOPMENT TOOLKIT

SE322 HP-UX CIO DRIVER WRITING CLASS

OPERATING SYSTEM STUDY OR MODIFICATION

HP-UX SOURCE CODE*

* This item requires a source license from AT&T.

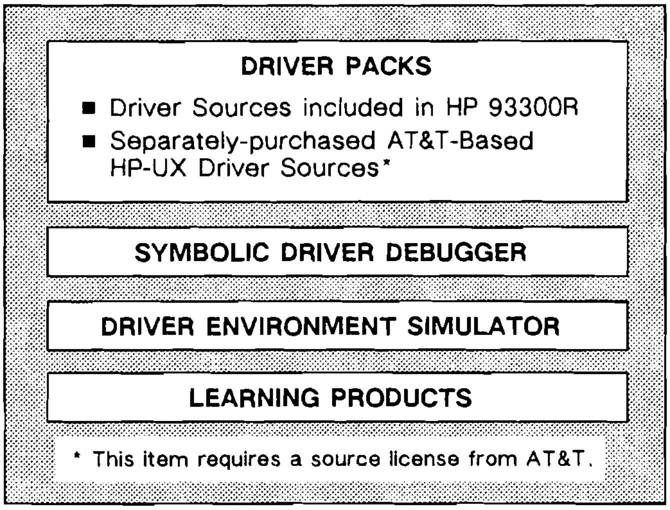
The application versatility of HP 9000 Series 800 (HP 9000/800) Model 825S and 840S computer systems can be augmented significantly with the open system tools for customization that are summarized in the diagram above. Users who want to develop Channel I/O (CIO) HP-UX drivers are supported by both the HP 93300R CIO Driver Development Toolkit and the SE322 HP-UX CIO Driver Writing Class*. Sophisticated users can obtain the HP-UX source code for study or modification of the HP-UX operating system itself.

HP 93300R CIO Driver Development Toolkit

The HP 93300R CIO Driver Development Toolkit provides an extensive set of driver development aids, as shown in Figure 1 and discussed in the following paragraphs.

Driver Packs

Driver packs are sets of CIO driver sources that are functional under HP-UX, along with "skeleton" (or fill-in-the-blanks) drivers. Users can adapt the source code of these drivers to meet the special requirements of their application, thereby getting off to a fast start in the development process.



DRIVER PACKS

- Driver Sources included in HP 93300R
- Separately-purchased AT&T-Based HP-UX Driver Sources*

SYMBOLIC DRIVER DEBUGGER

DRIVER ENVIRONMENT SIMULATOR

LEARNING PRODUCTS

* This item requires a source license from AT&T.

Figure 1. Elements of the HP 93300R CIO Driver Development Toolkit

Drivers Included in the HP 93300R CIO Driver Development Toolkit consist of HP-only code and therefore do not require an AT&T source license. The following drivers are included in the HP 93300R product.:

- An HP-IB Instrument Logical Device Manager.
- An HP-IB Instrument Device Adapter Manager.
- An Async FIFO Interface Logical Device Manager.

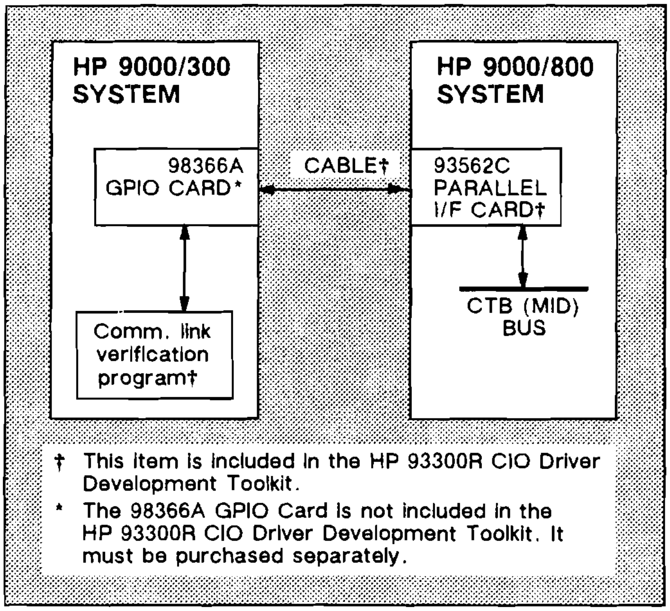
**HP-UX drivers are developed in a high level language (C), which facilitates writing and debugging.*

- An Async FIFO Interface Device Adapter Manager.
- A "skeleton" Logical Device Manager.
- A "skeleton" Device Manager/Device Adapter Manager.

AT&T-Based HP-UX Drivers Source Code. A second set of drivers that can provide a starting point for users' development efforts is available as a separate HP-UX Drivers Sources product. This product is separate because these drivers include AT&T-developed source code and can be purchased only if an AT&T Source License is purchased. These drivers include:

- The HP-IB interfaced Mag Tape Logical Device Manager.
- The HP-IB interfaced Printer Logical Device Manager.
- The HP-IB Disc Monolithic Manager.
- The Multiplexer Monolithic Manager.

Although it is available as a separate product, the HP-UX Drivers Source Code is also included in the HP-UX Source Code Product and would not have to be purchased separately if the HP-UX Source Product is purchased.



† This item is included in the HP 93300R CIO Driver Development Toolkit.

* The 98366A GPIO Card is not included in the HP 93300R CIO Driver Development Toolkit. It must be purchased separately.

Figure 2. HP 9000/300 ↔ HP 9000/800 Connection for CIO Driver Debugging

Symbolic Driver Debugger

After a driver source has been adapted from among the drivers in the Driver Packs and compiled, the Symbolic Driver Debugger supports debugging of the resulting object code. It can also be used to debug modified kernel code. The Symbolic Driver Debugger is used on an HP 9000 Series 300 (HP 9000/300) system to interactively debug modified kernel-level driver code executing on an HP 9000/800 System (see Figure 2 for connections). The Symbolic Driver Debugger includes:

- The Driver Debugger program, which runs on an HP 9000/300 computer and supports symbolic debugging of modified kernel-level code, using XDB commands.
- An I/O driver that supports communication between the HP 9000/300 system and the target HP 9000/800 Model 825S/840S system during the debugging process.
- A HP 93562C Parallel Interface Card and cable for connection between a separately-purchased HP 98622A GPIO card in the HP 9000/300 computer and the target system's CTB (Mid) bus.
- A Linkverify program that runs in the HP 9000/300 to verify HP 9000/300 ↔ HP 9000/800 communication link functions.
- Utility programs for merging kernel global debug into the target's kernel, creation of a kernel global debug information file from sources (for people who change the kernel), and provision of help files for the user.

Driver Environment Simulator

The Driver Environment Simulator simulates the environment of a CIO driver within HP-UX on the HP 9000/800 target system. It thus supports debugging and testing of a driver without requiring that a complete HP-UX system be dedicated only to that activity. All that remains to be done to fully simulate proper functioning of an interface card and driver is to construct input data files to match a specific order of driver input requests. Sample input files for existing CIO cards matched with the simulator's command files and appropriate documentation are provided to help the user with simulations.

Learning Products

The following learning products are provided to support and document customer's use of the HP 93300R CIO Driver Development Toolkit:

1. CIO Driver Writing Manual.
2. CIO Driver Packs User's Guide.
3. CIO Driver Reference Manual.
4. CIO Driver Environment Simulator Manual.
5. CIO Symbolic Driver Debugger (DDB) User's Guide.

SE322 HP-UX CIO Driver Writing Class

The SE322 HP-UX CIO Driver Writing class trains students to use the Symbolic Driver Debugger and Driver Environment Simulator to design, write, and debug I/O drivers. For this class, students must:

1. Have a practical user-level knowledge of the UNIX™ operating system, including commands, such as ls, cd, pwd, and grep and must know how to compile and run programs using the C compiler.
2. Be familiar with general concepts of the UNIX operating system kernel, shell, file system, and I/O drivers.
3. Know basic HP-UX system calls — open, close, read, write, ioctl, fcntl, and select, and know how to use those calls in programs.
4. Know the C programming language at an intermediate to advanced level, including arrays, structures, pointers, bitwise operators, relational operators, logic operators, increment and decrement operators, include files, conditional defines, and multiple object file linking.
5. Know how to generate an HP-UX operating system using the program Uxgen*.
6. Be familiar with the XDB debugger.

The class shows the students how the kernel and I/O system interact by reviewing a skeleton Logical Device Manager (LDM) and completing a skeleton LDM driver. The class also explains how the driver interfaces with the hardware by reviewing the skeleton Device Adapter Manager (DAM). The students will also complete a DAM driver.

Source Code Products

Some technically sophisticated users may need to obtain a copy of the source code for HP-UX or its I/O drivers to:

1. Gain a deeper understanding of how the operating system or driver works, with the potential of providing their own support of the software.
2. Customize the operating system or drivers to meet specific needs.
3. Meet archival requirements of the U.S. Government or other official entities.

Customer Qualification

To preserve Hewlett-Packard's excellent relationship with our customers and to assure customer satisfaction, we are qualifying customers who want to purchase source code. A key prerequisite is the purchase of a source license for AT&T System V, Release 2, Version 2, or a later release. HP sales representatives can provide further information on customer qualification for purchase of HP-UX source code.

HP-UX Source Code

The Source Code Itself. HP-UX source code is provided as an electronic copy on either CS/80 cartridge tape or 1600 cpi magnetic tape, consisting of:

- The kernel. This includes the driver sources, which can also be purchased separately from the HP-UX source code product (see listing under the HP-UX Drivers Source Code heading).
- The HP-UX Commands.
- The HP-UX Libraries.
- The HP-UX Diagnostics.

Learning Products. The following printed learning products will be furnished with the HP-UX Source Code product:

1. HP-UX Theory of Operations Manual.
2. HP-UX Technical Specifications (Two Volumes).
3. HP-UX Source Code Product User's Manual.

AT&T-Based HP-UX Drivers Source Code

Customers can also obtain only the HP-UX drivers source code, which is included in the HP-UX source code, if that is all they need for their customization. The included drivers are listed on page 2.

Support of Source Code

Hewlett-Packard provides no support of source code products or executables made from source code products. This is solely the responsibility of the customer.

Ordering Information

Contact your Hewlett-Packard Sales Representative to order any of the Open System Tools described in this data sheet.

* *Training in HP-UX system generation is provided in the HP 51482A HP-UX System Administration for the HP 9000 Series 800 System Course.*

™ UNIX is a registered trademark of AT&T.

LAN/9000 Series 800 Link

For HP 9000 Series 800 Computers

Product Numbers HP 91786A, 91788A, 98194A

The LAN/9000 Series 800 Link provides all the necessary hardware to interface between an HP 9000 Series 800 computer and an IEEE 802.3 Carrier-Sense Multiple Access with Collision Detection (CSMA/CD) Local Area Network. Also included in the link product is networking software corresponding to layers 1 through 5 of the Open Systems Interconnect (OSI) Reference Model and node management software. Users can choose to write their own software to access NetIPC or Berkeley Sockets or choose one of the higher level networking software products provided by Hewlett-Packard.

The LAN/9000 Series 800 Link, together with Network Services/9000 enable an HP 9000 Series 800 computer to communicate not only with other HP 9000 Series 800 computers but also with the HP 1000 A-Series, HP 9000 Series 300, HP 9000 Series 500 and Digital Equipment Corporation VAX™ (running VMS™) families of computers over a single coaxial cable.

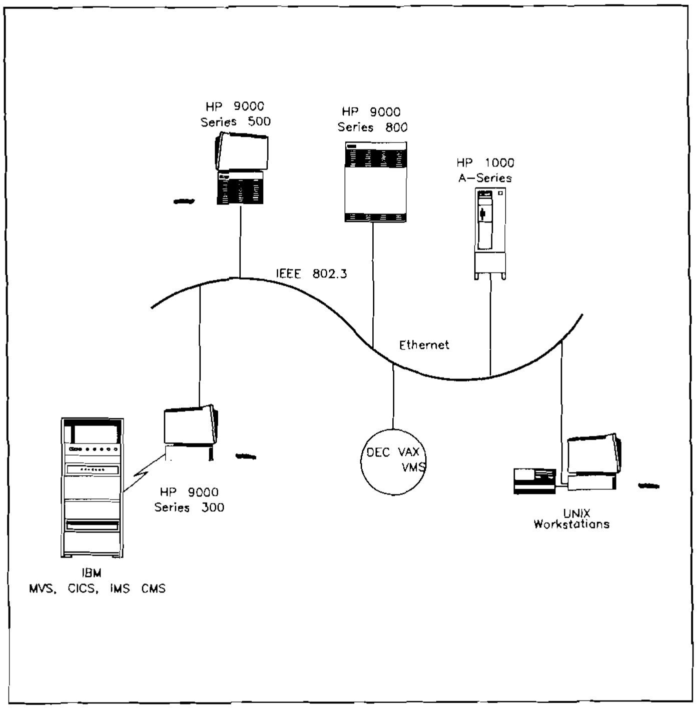
For those users needing multi-vendor communications, the LAN/9000 Series 800 Link can also be used with ARPA Services/800. ARPA Services/800 provides defacto networking software as defined by the Department of Defense Advanced Research Project Agency (ARPA) and the Berkeley Software Distribution (BSD) UNIX® 4.2 system.

For existing networks that use Ethernet hardware and need to transfer Ethernet-type packets instead of 802.3, both NS/9000 and LAN/9000 Link can be configured to operate accordingly. ARPA Services/800 will always transfer Ethernet-type packets.

Key Features

- A complete link connection to the local area network coaxial cable, which includes hardware and transport software to ensure a reliable connection.
- Operational compatibility with IEEE 802.3 and Ethernet Rev. 1.

VAX and VMS are trademarks of Digital Equipment Corporation.
UNIX is a registered trademark of AT&T in the U.S. and other countries.

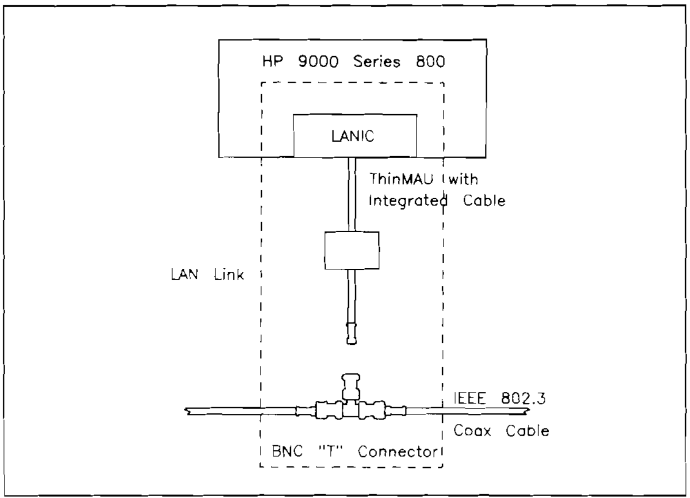


Series 800 in a Multi-vendor Environment

- 10 Megabits/second burst transfer rate.
- Carrier-Sense Multiple Access with Collision Detection (CSMA/CD) protocol controls network access without centralized control. All nodes have equal access.
- Any node may be connected or disconnected while the network is active.
- Microprocessor-driven interface controller that minimizes Series 800 overhead associated with communications line handling.
- Network transport software based on defacto industry-standard Department of Defense Advanced Research Projects Agency (DARPA) protocols, corresponding to the transport and network layer functions.
- Integrated node management software provides on-line network configuration and logging. An example of on-line network configuration would be adding another node without having to shut down the network.

Hardware Components

The following diagram shows the three major hardware components of the LAN/9000 Series 800 Link: Local Area Network Interface Controller (LANIC), card connector cable and ThinMAU.



Local Area Network Interface Controller

Local Area Network Interface Controller is a microprocessor-based communication controller that plugs into the Series 800 backplane. It handles buffering, IEEE 802.2 and 802.3 protocols, error checking, and keeps track of network statistics. When addressed by another node on the network, the LANIC receives frames of information and checks for accuracy of the data before passing the frames to the host. Before transmission, an addressed frame is sent from the host to the LANIC, which adds error checking information. The LANIC then tests to see if the cable is busy and, if not, transmits the frame.

Features

- Frame size of 1500 bytes
- Operational compatibility with IEEE 802.3 and Ethernet Rev. 1
- On board microprocessor
- 32K bytes of on board RAM allowing buffering for both transmit and receive packets
- Capable of receiving multiple back-to-back packets
- Provides for multicast, broadcast, and individual addressing
- Collection of link statistics (collided packets, bad packets, etc.)
- Power-on self-test
- Environmental: Class B
- EMC: will pass FCC, VDE Level A

IEEE 802.3 and Ethernet Card Connector Cables

Many times the differences between IEEE 802.3 and Ethernet create confusion. Ethernet LANs are very similar to IEEE 802.3 LANs. Since they utilize the same coaxial cable medium, Ethernet nodes may co-exist on the same LAN segment with IEEE 802.3 nodes. The most significant differences are in the data packet format and the electrical grounding of the hardware. Both the ThinMAU (28641A) and MAU (30241A) can transmit either IEEE 802.3 or Ethernet type packets.

When connecting the LANIC to either the ThinMAU (HP 28641A) or HP MAU (HP 30241A), the IEEE 802.3 Card Connector Cable must be used. This cable is part of the standard ThinMAU product. Some users may have MAUs/transceivers already installed at a node location. It is very important to identify whether the MAU/transceiver is an IEEE 802.3 or Ethernet version. The hardware for IEEE 802.3 and Ethernet nodes reference different electrical grounds. Thus, all the hardware of a particular node must conform to one standard or the other. For new nodes, conformance to IEEE 802.3 is recommended. However, if Ethernet hardware is already installed at a node location, an optional Ethernet card connector cable is available to make this connection.

Medium Attachment Unit

The Medium Attachment Unit (MAU) provides the physical and electrical connection by connecting the Attachment Unit Interface (AUI) cabling to the network coaxial cable. It receives signals from and sends signals to the coax cable, and also detects collisions resulting from two nodes starting to transmit simultaneously. The MAU also provides electrical isolation from the coaxial cable and performs several other functions to ensure network reliability, e.g., if a MAU fails by continuously transmitting, a circuit will detect the failure and shut down the MAU.

The LAN/9000 Link comes standard with a ThinMAU and integrated 1 meter AUI cable and BNC "T" Connector. These hardware components make the connection to the IEEE 802.3 Type 10BASE2 ThinLAN coaxial cable.

By ordering the appropriate option, customers can receive the ThickLAN Medium Attachment Unit with a coaxial tap. These components make the connection to the IEEE 802.3 Type 10BASE5 "thick" coaxial cable.

Functional Specifications

General Characteristics

Coaxial Cable Alternatives

Cable Type	ThinLAN (Standard with LAN/9000 Series 800 Link)	ThickLAN (backbone, optional)
IEEE cable specification	Type 10BASE2	Type 10BASE5
Maximum segment length	185 meters	500 meters
Maximum number of nodes per segment	30	100
Maximum distance between nodes	0.5 meter	2.5 meters
Maximum AUI cable length	1 meter	48 meters

Transmission Characteristics

Transmission Mode: Baseband Digital

Access Method: Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

Impedance: 50 Ohms

Environmental Characteristics

Temperature:
Non-operating: −40°C to +75°C
Operating: 0°C to +70°C

Humidity: 5% to 95% relative humidity

Electrical Specification

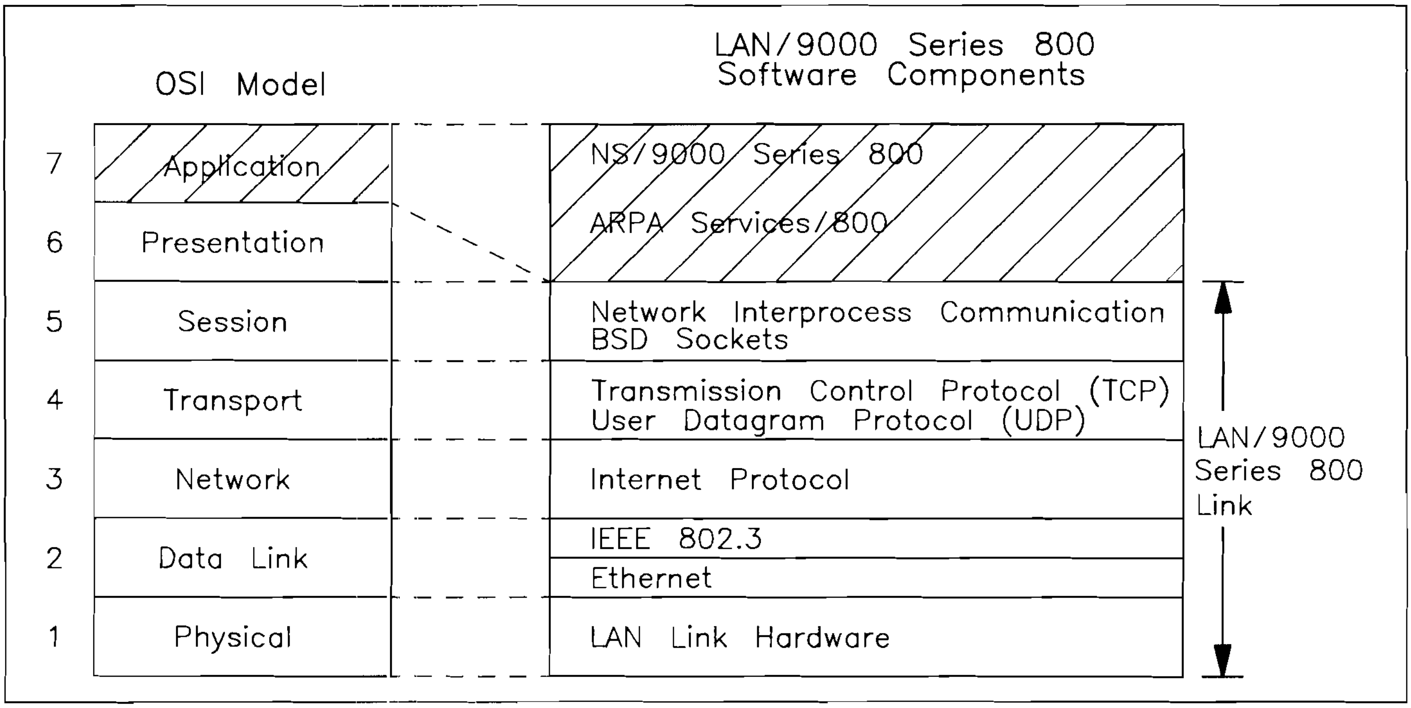
The Maximum Power Consumption for the interface is: 5 Volts; 15 Watts

The interface also powers the ThinMAU which requires: 12 Volts; 4.3 Watts Typical

Software Components

The LAN/9000 Series 800 Link includes software corresponding to Layers 1 through 5 of the Open Systems Interconnect (OSI) Reference Model, (see figure which follows). Node management software is also included.

Layers 1 and 2 of the OSI model consist of IEEE 802.2 and 802.3 protocols; CSMA/CD gives every node on the coaxial cable equal access to the network; a sending node listens on the network to ensure that no other node is transmitting before it attempts transmission. If, while transmitting, the sending node detects a collision, the sending node initiates a jam signal and waits for a random period of time before retransmitting. Transmission consists of sending addressed frames of data on the coaxial cable at a signaling rate of 10 megabits per second. The IEEE 802.3 and Ethernet type of service are Type 1 which is unacknowledged datagrams.



	OSI Model	LAN/9000 Series 800 Software Components	
7	Application	NS/9000 Series 800 ARPA Services/800	
6	Presentation		
5	Session	Network Interprocess Communication BSD Sockets	LAN/9000 Series 800 Link
4	Transport	Transmission Control Protocol (TCP) User Datagram Protocol (UDP)	
3	Network	Internet Protocol	
2	Data Link	IEEE 802.3 Ethernet	
1	Physical	LAN Link Hardware	

LAN/9000 Series 800

The Networking Layer, corresponding to OSI Layer 3, is based on the ARPA Internet Protocol (IP). IP provides fragmentation and reassembly of data as well as internal addressing. The Transport Layer, corresponding to OSI Layer 4, is based on the ARPA Transmission Control Protocol (TCP). TCP provides end-to-end reliable connection oriented services with flow control and multiplexing. TCP has mechanisms for detecting duplicate, lost, or out of sequence packets. Layer 4 also provides User Datagram Protocol (UDP) supported by BSD IPC. UDP provides an unacknowledged datagram service.

Network Interprocess Communication (NetIPC) and BSD Sockets are interfaces between Layer 7, and the product's transport protocols. NetIPC will enable programmers to establish peer-to-peer communication between processes running on HP 9000 Series 800 computers or HP 1000 A-Series computers. NetIPC is a Hewlett-Packard proprietary programmatic interface which currently resides on HP's 1000, 9000, and 3000 computer families. Currently, LAN/9000 Series 800 supports Series 800 to Series 800 and Series 800 to HP 1000 A-Series IPC communications. BSD Sockets also enables customers to establish peer-to-peer communications between processes running on HP 9000 Series 800's, as well as between processes running on HP 9000 Series 300, DEC systems running BSD 4.2 and other UNIX system workstations.

The node manager software uses the commands of the nodal management module to establish computer-to-network connections and to maintain the network. Network maintenance includes initialization and configuring; establishing network security; and using various diagnostic tools to assure proper network operation. Some of the diagnostics are loopback verification at Level 2, 3, and 5, nodal statistics, tracing, logging and security files.

Network Capacity and Performance

Although the signaling rate of the line may be 10 megabits per second, the throughput achieved at a Series 800 node will be lower. This is primarily due to the overhead of the software providing networking services and the user's application programs. Among the factors affecting user throughput is the type of software being used, the main memory and speed of each processor (and its peripherals) involved in the transfer, and the load on each system from non-network applications.

Installation Policy

Hewlett-Packard will provide software installation of the Local Area Network Link product for customers with Account Management Support (AMS). For customers not covered by AMS, Hewlett-Packard software installation is available on a time and material basis.

Customer Installation Responsibility

Installation of the LAN/9000 is the responsibility of the customer. Prior to installation of the LAN/9000, the customer should perform a full Series 800 backup. At that point, the customer will install and verify the operation of the LAN interface controller and perform a system update to add the product software modules to the system, then verify that the correct number and version of the software module has been installed.

The customer is responsible for installation of the coaxial cable, including terminators, T-connectors, taps, MAU's, and routing of the AUI cable to each MAU and to AUI cables on the Series 800. The customer should then connect the LANIC to the AUI cable and verify that the product properly accesses the AUI cable, and ensure that all safety grounds in the power system that are served by the coaxial cable are interconnected. The customer should then verify that proper access is made to the AUI from the product. At this point, installation of the LAN/9000 Series Link product is complete.

System Environment

LAN/9000 Series 800 is supported on the HP 9000 Series 800 computer systems. The minimum memory size required is 8 megabytes of main memory. This minimal configuration includes memory requirements for the HP-UX operating system. One LAN/9000 Series 800 Link product is supported per system.

The MAUs which come with the LAN Link can be used with any coaxial cable which fully complies with IEEE 802.3 specification for 0.4 inch diameter baseband coaxial cable. Use of Hewlett-Packard coaxial cable is recommended, since it contains relative distance markings to allow easy installation, maintenance and troubleshooting of the product.

Ordering Information

Hardware and Software

Order 98194A for Model 840, 91786A for Model 825, and 91788A for Model 850.

Each LAN/9000 Series 800 includes:

27125-60001	Printed Circuit Assembly
27125-63003	IEEE 802.3 Card Connector Cable
28641-60001	ThinMAU
28641-90001	ThinMAU Manual
1250-0781	Coax Adapter
1252-1154	T-Connector
	Lower Level Software on 1/4-inch Linus Tape

Specify Software Media Option

AA1: Replaces 1/4-inch Linus tape media with 1600 bpi, 9 track tape media (98194-13501)

Delete Options

740: Replaces ThinMAU assembly with MAU assembly (30241-60101), coaxial tap (0362-0819), and 6 meter Attachment Unit Interface (AUI) Cable (92254A)

811: Substitutes Ethernet Rev. 1 based card connector cable (98194-63004) for standard cable. The 28641A ThinMAU is also deleted. Due to grounding difference between the two types of hardware, it is important to distinguish the type of media access hardware being used at the node. If it is the 30241A MAU, or compatible 802.3 MAU then the 802.3 card connector cable will be necessary and therefore the standard product should be ordered. If the media access hardware conforms to Ethernet Rev. 1 then Option 811 substitutes an Ethernet card connector cable.

841: Deletes ThinMAU, Coaxial Cable Adapter and BNC "T" Connector

Documentation

Included with LAN/9000 Series 800 Link:

98194-90001	Local Area Network Interface Controller (LANIC) Reference Manual
98195-61000	NS/9000 Series 800 User Programmer Reference Manual
50980-60004	NS-ARPA Series 800 Node Manager Reference Manual
98195-61002	NS/9000 Series 800 Manual Reference Pages
50980-60000	ARPA Services Series 800 User's Guide
50980-60001	ARPA Services Series 800 Node Manager's Guide
50980-60002	Services Overview and Documentation Map
50980-60003	ARPA Services Series 800 Manual Reference Pages

Related Documents

5957-4624	Making the LAN Connection
5955-7680	LAN Cable and Accessories Manual
98195K	NS/9000 Series 800 Manual Kit. Includes customized binders, spines and documentation as described below:
98195-61000	NS/9000 Series 800 User Programmer Manual
50980-60004	NS-ARPA Series 800 Node Manager Reference Manual
98195-61002	NS/9000 Series 800 Manual Reference Pages
98195-61003	NS Cross-System NFT Reference Manual
98195-61004	NS Cross-System Network Reference Manual

Support Products

98194A+S00	Software Material Subscription (SMS) for LAN/9000 Series 800 Model 840
98194A+W00	Extended SMS (EMS) for LAN/9000 Series 800 Model 840
91786A+S00	SMS for LAN/9000 Series 800 Model 825
91786A+W00	Extended SMS (EMS) for Series 800 Model 825
91788A+S00	SMS for LAN/9000 Series 800 Model 850
91788A+W00	Extended SMS (EMS) for Series 800 Model 850

Response Center Support and Account Management Support customers must also order the Data Communications Category Support C, if it has not already been purchased.

Customers with hardware support agreements must add the appropriate level of coverage (SMMC or BMMC) for this Link product to their support agreement.

Coaxial Cable and LAN Accessories

LAN/9000 Series 800 Link provides all the components of a connection to the coaxial cable of an IEEE 802.3 Type 10BASE2 (workstation, "thin") local area network or, optionally, to a Type 10BASE5 (backbone, "thick") coaxial cable. A complete line of local area network products, including coaxial cable, installation tools, and connector products is available from Hewlett-Packard; refer to the current Computer Users Catalog. For cable planning information, refer to the LAN Cable and Accessories Installation Manual (P/N 5955-7680), available from your HP Sales Representative.

NS/9000 Series 800

HEWLETT PACKARD

For HP 9000 Series 800 Computers

Product Numbers HP 98195A/R, 91787A/R, 91789A/R

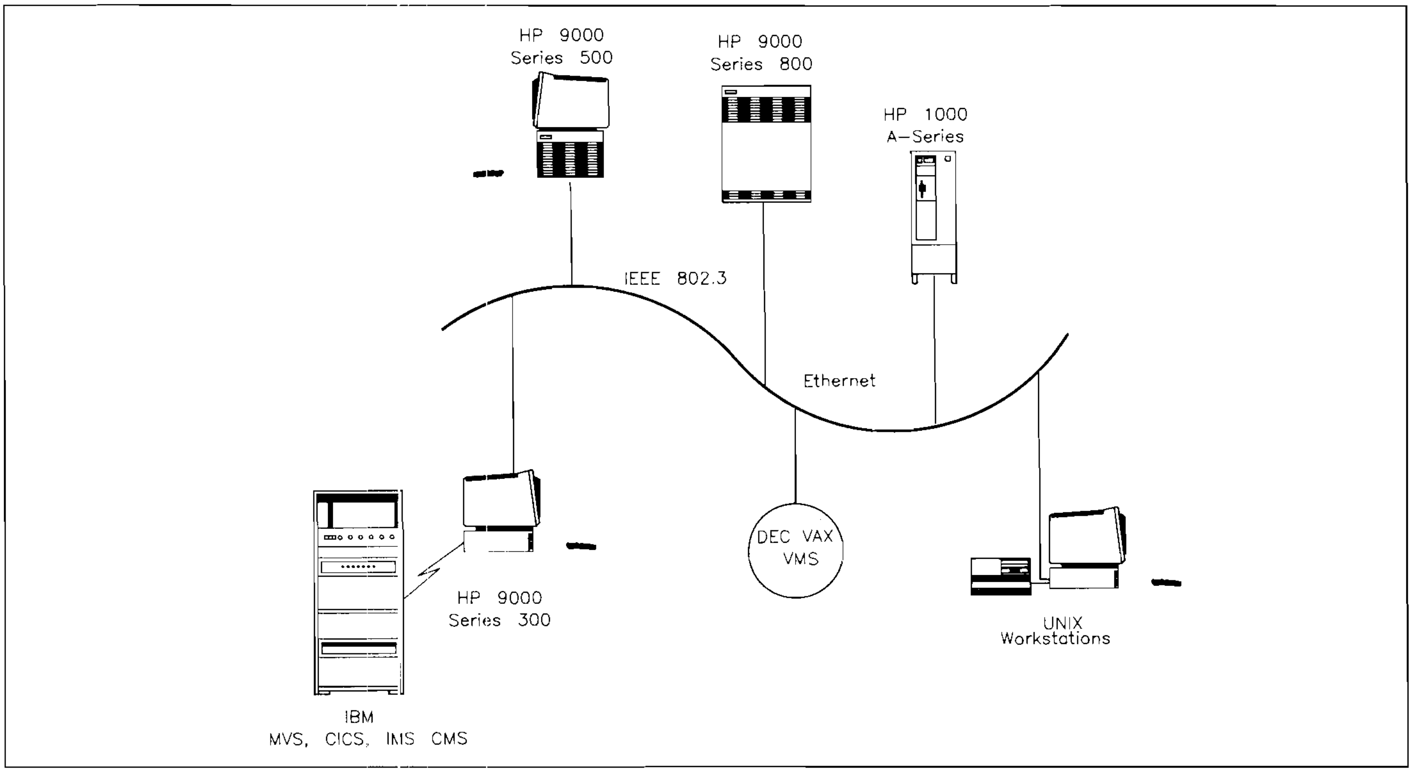


Figure 1. Series 800 in a Multi-vendor Environment

Network Services/9000 Series 800 (NS/9000 Series 800) provides engineering and manufacturing environments with the ability to communicate in a multi-vendor environment. NS/9000 Series 800 is based on HP's AdvanceNet data communications architecture and has been designed in a layer approach following the International Standards Organization's Open Systems Interconnect (ISO/OSI) Model.

NS/9000 Series 800, working with LAN/9000 Series 800 Link, supports communication between the HP 9000 Series 800, 500, and 300, the HP 1000 A-Series and DEC VAX™/VMS™ over a local area network. Users can easily access remote systems while error checking is performed automatically (see Figure 1).

Key Features

- NS/9000 Series 800 supports two network services over the high speed IEEE 802.3 Local Area Network Link (LAN/9000 Series 800 Link):
  - Network File Transfer (NFT) - enables the user to transfer files across a Local Area Network (LAN) between HP 9000 Series 800, 500, and 300, HP 1000 A-Series, and DEC VAX/VMS computers.
  - Remote File Access (RFA) - allows the user to access directories, data files and programs on other HP-UX* systems on the network by using the standard HP-UX commands and system calls.

VAX and VMS are trademarks of Digital Equipment Corporation.

* HP-UX is Hewlett-Packard's implementation of AT&T's System V UNIX® operating system; UNIX® is a registered trademark of AT&T in the U.S. and other countries.

- Resource sharing between users in an HP-UX environment: Engineers can share discs and printers between HP 9000 Series 800 and other HP 9000 HP-UX systems.
- Industry standard based protocols: NS/9000 Series 800 runs on the LAN/9000 Series 800 Link product which supports IEEE 802.3 as well as transport and internet protocol layers based on the Advanced Research Projects Agency (ARPA) Transport Control Protocol and Internet Protocol (TCP/IP). (NS/9000 Series 800 has not been certified as ARPA compatible.)

Functional Description

NS/9000 Series 800 functionality combines the NS Common Services transports with the capabilities of the LAN transports and the IEEE 802.3 LAN/9000 Series 800 Link. These facilities are described in detail in Figure 2 which shows the relationship between the NS Common Services, transports and like components of the product.

User Level Services

Network File Transfer

Network File Transfer (NFT) copies files between any two NS nodes on a network. The HP NFT protocol is common to all HP Network Services implementations. Files can be copied interactively or programmatically. Network File Transfer includes features that allow users to:

Copy Remote Files: Using NFT on a local Series 800 system, a user can copy files from a local node to a remote node, from a remote node to a local node, and between remote nodes.

Translate File Attributes: File attributes are translated transparently and on demand when files are copied between different systems. This means that when an HP-UX file is copied to an HP 1000 A-Series computer, the file attributes will be translated into RTE-A file format.

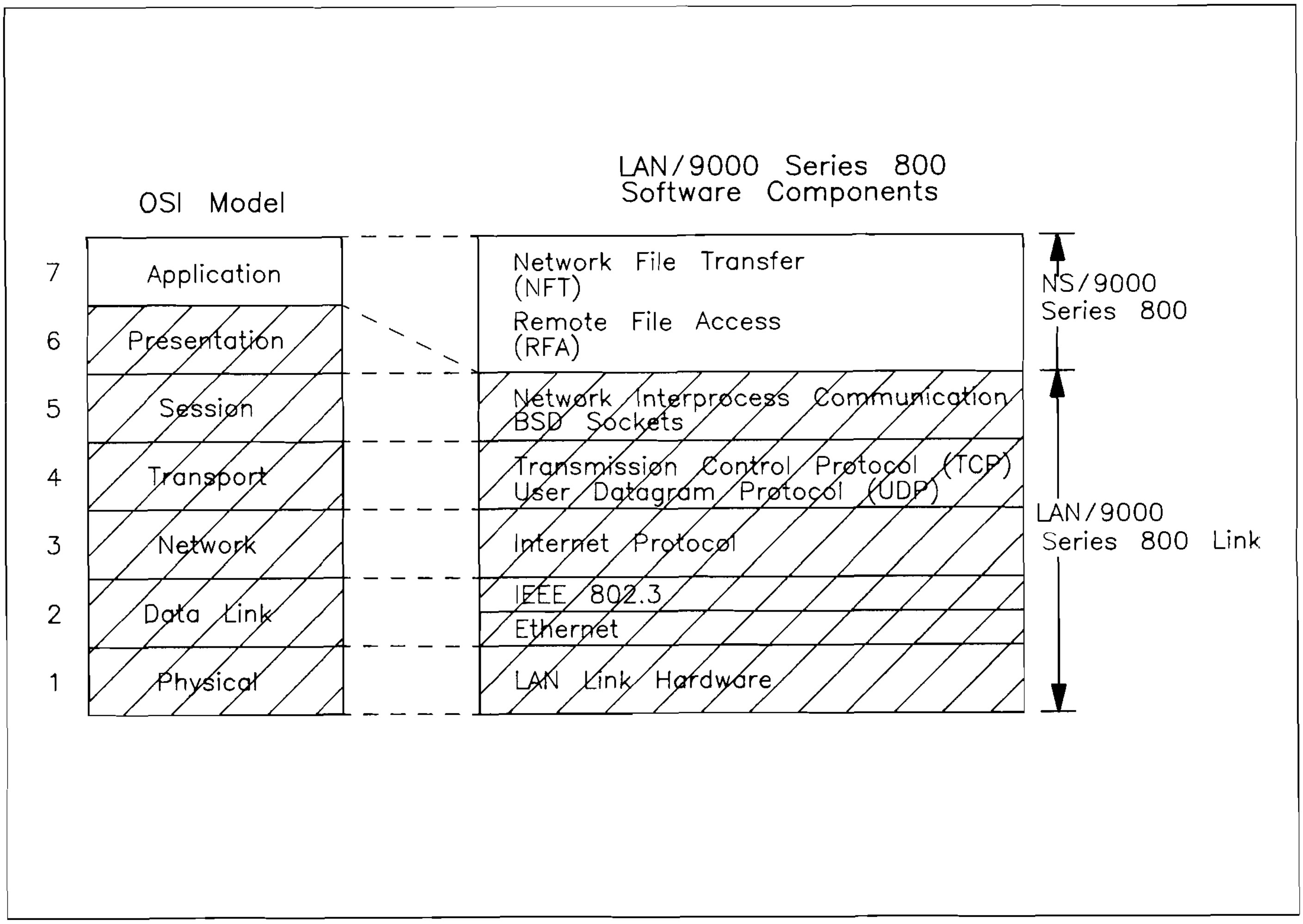


Figure 2. NS and LAN/9000 Series 800 Architecture

Access Remote Accounts: Files under any account can be accessed if the user provides the correct logon and password.

The **dscopy** command is used to copy files from system to system over the network. This command is similar to the HP-UX **cp** command, which is used for local file copies. With **dscopy**, however, you must specify a system name and the appropriate login information before a remote file can be copied. For example, to copy a file from a system named *pc_design* to a system named *pc_fab*, you could execute the command (command is typed on only one line by user):

```
dscopy pc_design#alpha:beta#source_file
\pc_fab#delta:gama#dest_file
```

The source and destination files are *source_file* and *dest_file*, respectively. The login for *pc_design* is *alpha:beta*, and the login for *pc_fab* is *delta:gama*.

Remote File Access

Remote file access allows users to logically extend the local HP-UX file system to include remote file systems. Thus, users can use HP-UX commands and system calls to access remote directions and files as if they are local.

To use remote file access, a network special file is created to logically link a particular remote file system to the local file system. For example, */net/admin* on the local node connects node admin's file system to the local node. For security purposes, the user must establish his login with the remote system using the netunam command before he can access the remote system. For example, to login to a remote system named *pc_design* with the username *alpha* and the password *beta*, one would execute the command:

```
netunam /net/pc_design alpha:beta
```

Another example, to change your working directory to a directory named */users/hardware* on the remote system *pc_fab*, you would execute the command:

```
cd /net/pc_fab/users/hardware
```

Network Capacity and Performance

The maximum number of separate conversations with other nodes that a single processor can support and the peak user data rates achievable on each link are the complex functions of many interrelated variables. Among these variables are the speed of each processor that is a party to the conversation (and its peripherals), and the load on each system from non-network applications. Because of the number and complexity of these factors, it is difficult to make accurate generalizations about capacity and performance.

Network Link Requirements

Use of NS/9000 Series 800 requires the Local Area Network Link (LAN/9000 Series 800). The LAN/9000 Series 800 Link alone enables Series 800 computers to communicate with other Series 800 computers using NetIPC and BSD Sockets, with HP 9000 Series 300 using BSD Sockets, or with HP 1000 A-Series computers using NetIPC. Service products available to run on the LAN/9000 Series 800 Link are NS/9000 Series 800 (as described in this data sheet), and ARPA Services/800 (see data sheet for more information).

LAN/9000 Series 800 Link includes an interface board (LANIC) and a card connector cable Medium Attachment Unit (MAU), Attachment Unit Interface (AUI), and software, to connect an HP 9000 Series 800 to a ThinLAN coaxial cable. (Consult the LAN/9000 Series 800 Link data sheet for more details.) The user may specify ThickLAN, MAU and AUI optionally at time of order. The distance between processors is a function of the length and type of the LAN cable. Up to 30 computers can be supported per ThinLAN segment which has a maximum cable length of 185 meters, and up to 100 computers can be supported on a ThickLAN segment which has a maximum cable length of 500 meters. Three segment combinations can be connected with repeaters: Thin-Thin, Thin-Thick, Thick-Thick. The AUI included with the LAN/9000 Series 800 Link product is 1 meter long, with the optional ThickLAN AUI cable at a maximum length of 48 meters. Both types of LAN cables have a cable speed of 10 megabits per second.

Communication with other Series 800 computers over the LAN requires that the remote system be equipped with a LAN/9000 Series 800 Link and either NS/9000 Series 800 or ARPA Services/800 software. Specific requirements for intersystems communication with LAN/9000 Series 800 Link and NS/9000 Series 800 between Series 800 and non-Series 800 processors are described below.

HP 9000 Series 800 NS and LAN Links

Network System	LAN Link	Software
HP 9000 Series 800 Model 840	LAN/9000 Series 800 Model 840 (98194A)	NS/9000 Series 800 Model 840 (98195A/R)
HP 9000 Series 800 Model 825	LAN/9000 Series 800 Model 825 (91786A)	NS/9000 Series 800 Model 825 (91787A/R)
HP 9000 Series 800 Model 850	LAN/9000 Series 800 Model 850 (91788A)	NS/9000 Series 800 Model 850 (91789A/R)
HP 9000 Series 300	LAN/9000 Series 300 (98643A)	NS/9000 Services Series 300 (50952B)
HP 9000 Series 500	LAN/9000 Series 500 (27125B)	NS/9000 Series 500 (50954A/R)
HP 1000 A-Series	LAN/1000 Link (12076A)	NS/1000 (91790A/R)
DEC VAX* VMS		NS for DEC VAX (50950A)

* NS/9000 Series 800 <-> DEC VAX requires specific software and/or hardware on the DEC VAX computer. Consult the "NS for the VAX" data sheet for more information.

Compatibility

Compatibility with other active HP network products follows:

NS/9000 Series 800 software is compatible with:

- NS-ARPA Services/300 (50952B)
  NFT and RFA Services
- NS/500 (50954A/R)
  NFT and RFA Services
- NS/1000 (91790A/R)
  NFT Service
- NS for the DEC VAX Computer (50950A)
  NFT Service

System Environment

Series 800 Model 825 and 840 computers running NS/9000 Series 800 must be running the HP-UX operating system version A.01.10. Model 850 computers must run HP-UX version A.01.20. A minimal configuration for NS/9000 Series 800 functional operation is 8 MBytes of memory, which includes the memory requirements for the HP-UX operating system, and at least one disc drive.

Hewlett-Packard System Engineers and Data Communications Specialists are available to consult in network design and can assist in designing an effective network. Consult your Sales Representative for more details.

Customer Installation/Responsibility

The customer must assume the following responsibilities with the purchase of NS/9000 Series 800:

1. LAN/9000 Series 800 Link product must be installed prior to the NS/9000 Series 800 software. The customer is responsible for network configuration and installation of the NS/9000 Series 800.
2. It is highly recommended that one person in the customer's organization must be designated as the Network Manager. This person will assume responsibility for configuration and generation of the customer's systems and will function as the focal point for Hewlett-Packard's support of the network.
3. Hewlett-Packard strongly recommends that the customer purchase Account Management Support (AMS) or Response Center Support (RCS) for NS/9000 Series 800 and related hardware and software support products.

Ordering Information

Software

Product Number	Product Description
98195A	HP Network Services/9000 Series 800 Model 840.
98195R	HP Network Services/9000 Series 800 Model 840. Provides Right-to-Copy and sublicense of NS for the HP 9000 Series 800.
91787A	NS/9000 Series 800 Model 825.
91787R	Right-to-Copy
91789A	NS/9000 Series 800 Model 850.
91789R	Right-to-Copy

Options

Customers who order NS/9000 Series 800 will receive software products on ¼-inch Linus tape media. Those customers requiring 1600 bpi, 9-track tape should order option AA1 as noted below:

Option AA1	Replaces ¼-inch Linus tape media with 1600 bpi, 9 track tape media.

Documentation

The following documentation is provided with the NS/9000 Series 800 product. This material describes the functionality and versatility of NS/9000 Series 800.

98195-61003	NS Cross System NFT Reference Manual
98195-61004	NS Cross-System Network Reference Manual

Use of the NS/9000 Series 800 requires LAN/9000 Series 800 Link. The following NS/9000 Series 800 related documents are provided with the LAN/9000 Series 800 Link product.

98195-61000	NS/9000 Series 800 User/Programmers Reference Manual
50980-60004	NS-ARPA Series 800 Node Managers Reference Manual
98195-61002	NS/9000 Series 800 Reference Pages

In addition, a manual kit is orderable which includes all of the above referenced manuals.

98195K NS/9000 Series 800 Manual Kit

- 98195-61000
- 50980-60004
- 98195-61002
- 98195-61003
- 98195-61004

Option

0B0: Deletes 98195-61003 and 98195-61004

Support Products

For all new customers and customers without an existing contract, the available support services include Account Management Support (AMS), Software Materials Subscription (SMS), and Response Center Support (RCS) which includes all materials (SMS) and telephone assistance through the Response Centers.

For customers choosing AMS or RCS on their operating system, NS/9000 Series 800 is supported by purchasing Datacomm C category support. Only one datacomm category support need be purchased for the "highest" datacomm category required on a system (A = lowest, C = highest).

Support Product Numbers

98195A+S00	(Model 840) Software Materials Subscription (SMS)
91787A+S00	(Model 825) Software Materials Subscription (SMS)
91789A+S00	(Model 850) Software Materials Subscription (SMS)
98195A+W00	(Model 840) Extended SMS (EMS)
91787A+W00	(Model 825) Extended SMS (EMS)
91789A+W00	(Model 850) Extended SMS (EMS)
99087N+C00	Datacomm Category C Support
99087N+V00	Extended Datacomm Category Support
99087N+Q00	Manual Update Service (MUS)

ARPA Services/800

For HP 9000 Series 800 Computers

Product Numbers HP 50980A/R, 50981A/R, 50982A/R

Series 800 Models 825, 840, and 850 HP-UX† computers can communicate in a multi-vendor environment using the networking services defined by the Department of Defense Advanced Research Project Agency (ARPA) and the Berkeley Software Distribution (BSD) UNIX® 4.2 system. (See Figure 1.) ARPA Services/800 offers the following features:

- ARPA services are de facto networking standards in the scientific and engineering communities. They define protocols for electronic mail, file transfer and terminal access over local and wide area networks.
- BSD UNIX 4.2 implements a de facto networking standard for the UNIX community. It defines protocols for file transfer, terminal access, remote command execution, electronic mail, and interprocess communication over local and wide area networks.

The ARPA Services/800 software requires the LAN/9000 Series 800 Link hardware product, P/N 91786A (for Model 825), P/N 98194A (for Model 840) and P/N 91788A (for Model 850).

Note: HP's Network Services (NS) are available on the Series 800 providing HP-to-HP communications. NS, ARPA, and Berkeley services can run simultaneously on the Series 800.

ARPA Services

ARPA Services/800 runs on a Series 800 Model 825, 840 or 850 connected to an Ethernet LAN and supports the following ARPA services:

Service	Feature
File Transfer Protocol (FTP), (MIL-STD‡ 1780)	General file utility for performing operations on remote files and directories such as transferring, deleting, renaming and displaying.
TELNET (based on MIL-STD 1782)	Virtual terminal capability for accessing remote systems as a terminal.
Simple Mail Transfer Protocol (SMTP), (MIL-STD 1781)	Enhancement of the UNIX mail facility with support for LANs.

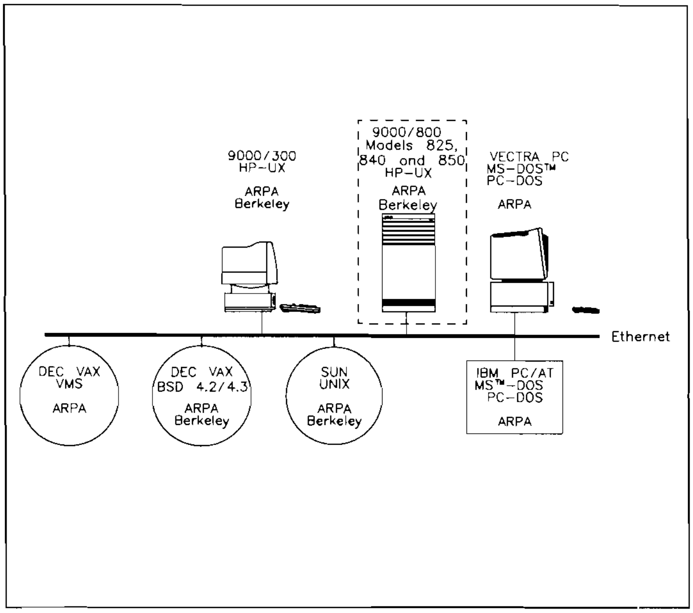


Figure 1. Series 800 in a Multi-vendor Environment

* The networking services that are available for each computer are specified in each box. Computers on the network must have like services to communicate.
MS™-DOS is a trademark of Microsoft Corporation.

The protocols underlying the ARPA services closely adhere to the following standards:

- Internet Protocol (MIL-STD 1777)
- Transmission Control Protocol (MIL-STD 1778)
- User Datagram Protocol (RFC-768)
- Internet Control Message Protocol (RFC-792)
- Address Resolution Protocol (RFC-826)

The protocols are part of the 98194A LAN/9000 Series 800 product.

† HP-UX is Hewlett-Parkard's implementation of AT&T's System V UNIX operating system.
UNIX is a registered trademark of AT&T in the U.S. and other countries.

‡ MIL-STD is an industry-wide term used to abbreviate Military Standard.

BSD 4.2 Services and Sockets

Series 800 HP-UX machines can communicate with other UNIX computers. The Berkeley portion of ARPA Services/800 provides the following capabilities:

Service	Feature
Remote Copy (`rcp`)	Transfers data and program files among computers on the network
Remote Login (`rlogin`)	Virtual terminal capability for accessing remote systems as a terminal
Remote Who (`rwho`)	Displays users logged into systems on the network
Remote Uptime (`ruptime`)	Displays information about systems running on the network
Remote Shell (`remsh`)	Runs a program or shell on a remote computer and receives output
Sendmail	Routes mail (integrated into HP-UX mail)
Berkeley Sockets	Interprocess communications for creating distributed application programs (included with the LAN/9000 Series 800 Link)

Diagnostics

Several diagnostic features are included with the LAN/9000 Link. These features allow monitoring the network, gathering statistical data and diagnosing problems in both multi-vendor and HP-to-HP configurations.

Diagnostic	Feature
`ping`	Verifies connections to remote ARPA/Berkeley systems
`netstat`	Berkeley-derived command that displays statistical information on network activity
`nettrace` `netlog`	Sophisticated trouble-shooting tools for applications programmers and network administrators to trace activity and report errors
`landad`	Menu-driven diagnostic facility for trouble-shooting LAN card/link level

Summary of ARPA Services/800 Capabilities

Capability	ARPA	Berkeley 4.2
File Transfer	File Transfer Protocol (FTP)	remote copy (`rcp`)
Terminal Access	TELNET	remote login (`rlogin`)
Electronic Mail	Simple Mail Transfer Protocol (SMTP)	sendmail (uses SMTP)
Remote Command Execution	—	remote shell (`remsh`)
Interprocess Communication	—	sockets (included with LAN/9000 Link)

The ARPA and Berkeley capabilities can be put in the context of the Open Systems Interconnection (OSI) reference model defined by the International Standards Organization (ISO) as shown in Figure 2:

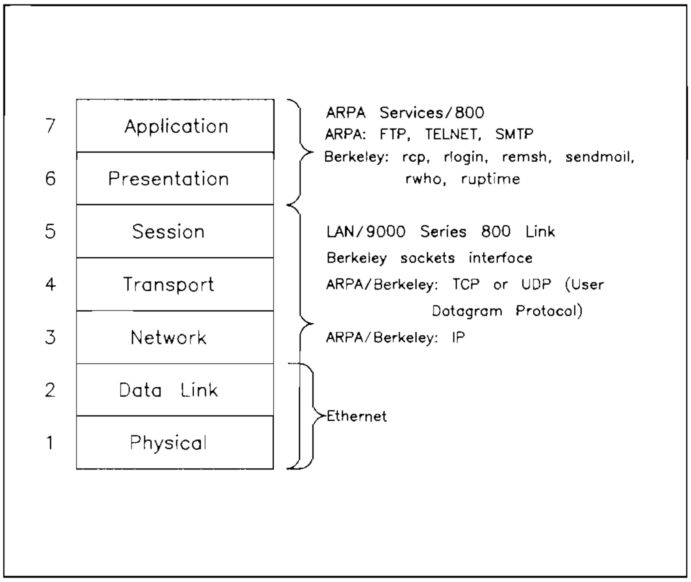


Figure 2. ARPA Services/800 and the OSI Model

Functional Specifications

ARPA - File Transfer Protocol

File Transfer Protocol (FTP) is a family of commands for performing file and directory operations over the network. You can get or put stream files on a remote UNIX system or non-UNIX system machine, using either ASCII or binary transfers. You can append, rename and delete files, list, change, make and remove directories, check status, toggle switches and ask for help.

FTP supports the following list of commands:

```
append    form      mget      quote        tenex
ascii     get       mkdir     recv         type
bell      glob      mls       remotehelp   user
binary    hash      mode      rename       verbose
bye       help      open      rmdir        ?
cd        lcd       prompt    send         !
close     ls        put       sendport
delete    mdelete   pwd       status
dir       mdir      quit      struct
```

Here is a simple example of how you might use FTP:

Command	Results
`ftp rhost`	Establish a connection to the computer named rhost. Prompt for login name and password.
`pwd`	Print the name of the current working directory on the remote machine.
`ls`	Print an abbreviated listing of the contents of the current working directory on the remote machine.
`put file1 rfile1`	Store a local file on the remote machine.
`get rfile2 file2`	Retrieve a remote file and store it on the local machine.
`delete rfile`	Delete a file on the remote machine.
`bye`	Terminate the FTP session with the remote server and exit FTP.

ARPA — Teletype Network Protocol

Teletype Network Protocol (TELNET) lets you use your local workstation as a terminal to another computer on the network. The remote computer can be running an operating system other than the UNIX system. To sign on to a remote host named *rnode*, just type `telnet rnode`. The remote node will prompt you for your login user name and password.

TELNET has both a command mode and an input mode. You can recognize command mode by its `telnet` prompt. Command mode is useful for opening and closing sessions, changing parameters, checking status and getting help. Most of the time you will use input mode. In this mode, you have terminal access to the remote node. You can run programs, edit text, or list directories, as well as execute a `telnet` command.

TELNET does not emulate a particular terminal type. The host computer that you are logged onto must support the terminal you are using.

ARPA/Berkeley — Simple Mail Transfer Protocol (SMTP) and Sendmail

Users can communicate from Series 800 HP-UX systems to other UNIX or non-UNIX systems using either the `mailx` or `mail` commands. Series 800 HP-UX systems running ARPA Services/800 can send mail over RS-232 lines and Ethernet LANs.

ARPA Services/800 provides two alternatives for specifying mail addresses. Historically HP-UX has required users to specify the complete path from the local node to the recipient's node (a relative form of addressing). These names have looked like:

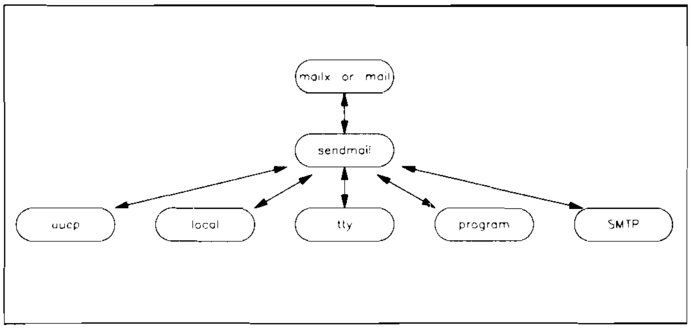
```
node1!node2!node3!user
```

Another alternative is domain style addressing (an absolute form). `Sendmail` and SMTP allow for this form of addressing. You simply specify the name of the user and the domain/subdomain name of the company, government agency or educational institution where the recipient is located. An example*:

```
joe_smith@HP.COM
```

Users do not interact directly with the sendmail program or the SMTP protocol. Sendmail acts as a unified "post office" to which all mail can be submitted. SMTP is an ARPA protocol for sending mail over wide area and local area networks. The relationship of `mailx` and `mail` to `sendmail` and SMTP is shown in the figure below.

The Mail System



* *The degree to which the user can address mail in this simplified manner depends on the availability and extent of the name servers on the network.*

Berkeley — Remote File Copy

Remote UNIX file copy, `rcp`, lets you copy files from one UNIX node to another UNIX node over the network. Both the source and destination nodes can be remote, or one can be local and the other remote. You must have a login (user name and password) and permission for remote command execution on each node.

In the simplest case, you can copy a file on a local node to a file on another node. For example, to copy a file named *lfile* in the current working directory on the local node to a file named *rfile* in your login directory on the remote node *rnode*, type `rcp lfile rnode:rfile`

You can also copy several files or an entire subtree within a directory from one node to a directory on another node.

Berkeley — Remote Login

Remote UNIX login, `rlogin`, gives you terminal access to a remote UNIX node on the network. `Rlogin` has several options for connecting to other systems. You can sign on to the remote host simply by typing `rlogin rnode`. This method assumes that your user name is the same on both the local and remote systems. You can specify a different user name on the remote system with the `-S` option to `rlogin`. You may or may not need to specify a password, depending on how your network is configured.

`Rlogin` also supports `control-S` for stopping the flow of output to the terminal and `control-Q` for starting the flow of output to the terminal.

Berkeley — Remote Shell

Remote shell or `remsh` lets you execute a command on a remote UNIX host. For example, `remsh rnode cc test.c` compiles the C program *test.c* on the remote node named *rnode*. `Remsh` normally terminates when the remote command terminates. For interactive commands such as `more` or `vi`, you must use `rlogin` instead of `remsh`. `Remsh` is the same service as Berkeley's `rsh`. The name `rsh` conflicts with another HP-UX command.

`Remsh`, which runs on the local system, copies its standard input to the standard input of the command that will be executed on the remote system. `Remsh` also copies the standard output and standard error from the command running on the remote system to its own standard output and standard error. For example, consider `cat f1 f2 : remsh rnode sort`. The output from `cat f1 f2` becomes the standard input to `remsh`. `Remsh` sends this input to the sort program running on *rnode*. `Remsh` copies the sorted output to its standard output on the local system. If the sort produces any error messages, `remsh` copies errors to its standard error on the local system.

Berkeley — Sockets and Libraries

Berkeley sockets and libraries are standard tools for interprocess communication on UNIX systems. Application developers use these tools to create networked solutions. Since Berkeley sockets and libraries are widely available, applications based on them can be ported to Series 800 HP-UX systems.

Two types of sockets, *stream* and *datagram*, are supported. Stream sockets are appropriate for reliably transferring large volumes of data. A connection is set up, data is transferred and each packet is checked at the receiving end to verify accurate transmission. Stream sockets use the Transmission Control Protocol (TCP).

Datagram sockets are appropriate for short, fast data transfers without error checking. For example, a program may query some status information from another program and wait for a reply, requiring no follow up. Application developers often prefer datagram sockets because they are faster and easier to use. Datagram sockets use the User Datagram Protocol (UDP).

The Berkeley libraries give the applications developer a way to look up important data about the network. For example, the user can look up the Internet addresses, port numbers and protocol numbers. Berkeley utilities such as `rexec` are also available. The Berkeley libraries and utilities are usually used with socket programming. For detailed information, please refer to the *ARPA Services/800 User's Guide,* (P/N 50980-90000).

Connectivity to Other Vendors' Products

HP has certified operation of ARPA Services and Berkeley Services and sockets by testing communication with products from HP and other vendors. ARPA or Berkeley Services use the Ethernet protocol. The following configurations are certified:

Services	Computer	Operating System	Network Software Package
ARPA	DEC VAX 7xx	VMS 4.0 or later	Wollongong's WIN/VX Rel. 2.2/2.3
ARPA (FTP and TELNET only)	IBM PC-AT	MS-DOS or PC-DOS 2.xx or 3.xx	Network Research Corporation FUSION Rel. 3.1.13 (FNS-PC-TCP)
ARPA (FTP and TELNET only)	HP Vectra	MS-DOS or PC-DOS 2.xx or 3.xx	Network Research Corporation FUSION Rel. 3.1.13 (FNS-PC-TCP)
ARPA/BSD 4.2	SUN (68010)	SUN Release 3.0/3.2	SUN networking
ARPA/BSD 4.2	SUN (68020)	SUN Release 3.0/3.2	SUN networking
ARPA/BSD 4.2	DEC VAX 7xx	BSD UNIX 4.2/4.3	BSD 4.2/4.3 networking
ARPA/BSD 4.2	HP 9000/300	HP-UX 5.2	NS-ARPA Services Rev. 5.171 or 5.4

HP has an ongoing program to certify ARPA Services/800 with additional products. If you would like to use ARPA Services/800 to communicate with products that are not listed here, please contact your local HP Sales Office. Testing may have occurred or may be planned in the near future.

Configuration Information

Computer

- HP 9000 Series 800 Model 825, 840 or 850

Operating System

HP-UX Version A.01.10 for Model 825 and 840. HP-UX Version A.01.20 for Model 850.

Memory and Data Storage

- Eight megabytes RAM
- Two (2) HP 7914 132 Mbyte discs, or
- One 7933/5H 404 Mbyte disc

Ordering Information

Software

- 50981A ARPA Services/800 Model 825
- 50981R Right-to-Copy ARPA Services/800 Model 825
- 50980A ARPA Services/800 Model 840
- 50980R Right-to-Copy ARPA Services/800 Model 840
- 50982A ARPA Services/800 Model 850
- 50982R Right-to-Copy ARPA Services/800 Model 850

Media Options:

- AA0 ¼-inch tape
- AA1 ½-inch 9-track tape

Hardware

- 91786A LAN/9000 Model 825
- 98194A LAN/9000 Model 840
- 91788A LAN/9000 Model 850

Software Support Services

- 50981A+S00 – Software Materials Subscription (SMS)
- 50981A+W00 – Extended Software Materials Subscription (EMS)
- 50980A+S00 – Software Materials Subscription
- 50980A+W00 – Extended Software Materials Subscription
- 50982A+S00 – Software Materials Subscription
- 50982A+W00 – Extended Software Materials Subscription
- 99087N+C00 – Datacomm C Category Support
- 99087N+V00 – Extended Category Support
- 99087N+Q00 – Manual Update Services

Documentation

- 50980-60000 – ARPA Services/800 User's Guide
- 50980-60004 – NS-ARPA/800 Node Manager's Guide
- 50980-60002 – Services Overview and Documentation Map
- 50980-60003 – ARPA Services/9000 Series 800 Manual Reference Pages
- 50980K – ARPA Services Manual Kit

HP-UX Gateway/SNA 3270

HEWLETT PACKARD

For HP 9000 Series 800 Computer Systems

Product Numbers HP 36911A/R, 36918A/R, 36919A/R

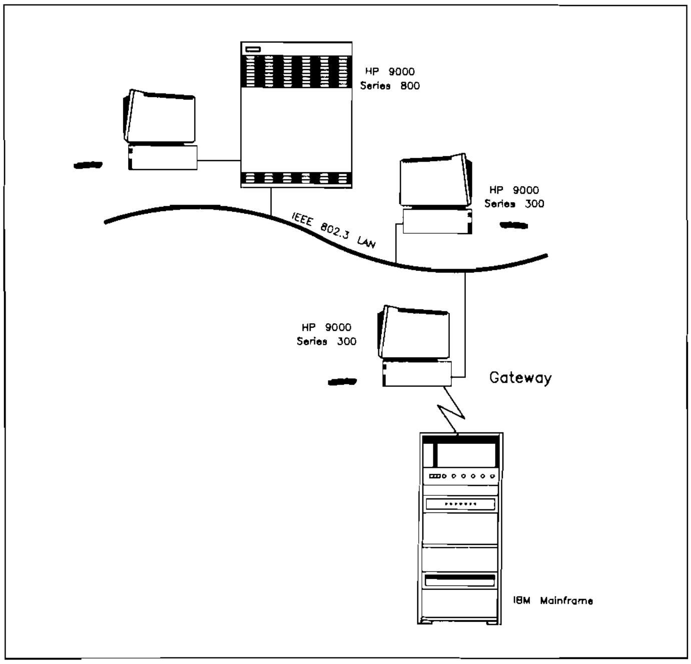
HP-UX Gateway/SNA 3270 allows interactive communications between an HP 9000 Series 800 and an IBM* System/370-compatible mainframe using SNA 3270 protocols. The Series 800 communicates to IBM through a Series 300 processor that functions as a gateway. HP terminals, monitors, and printers on the Series 800 emulate the functions of IBM 3278 terminals and 3287 printers. Gateway/SNA 3270 goes beyond emulation to include file transfer and multiple session capabilities.

Gateway/SNA 3270 requires LAN-based access to a Series 300 running HP-UX Gateway/SNALink (36593A) which provides the major features of an IBM 3274 cluster control unit and the lower four levels of SNA. Physical Unit Type 2 (PU2), Logical Unit Types 1, 2, and 3 (LU1, LU2, LU3) are emulated.

When used with the Series 300's Gateway/SNALink, the Series 800's Gateway/SNA 3270 allows access to 3270 applications on the mainframe such as TSO, CMS, IMS, and CICS.

Key Features

- Allows HP 9000 Series 800 workstation monitors, terminals, and printers to emulate the major features of IBM 3278 terminals and 3287 printers. Terminal access can be through the standard HP-UX supported options.
- PC3270 file transfer enables interactive file transfers on the HP 9000 to IBM systems running TSO, CICS, or CMS.
- Escape to UNIX** shell provides terminal users with the ability to access other HP-UX programs while maintaining the 3270 session in the background.
- Efficient screen handling routines provide a very responsive and low overhead user interface.
- Powerful commands to monitor and control the operation of the product.



Functional Specifications

- When used with HP-UX Gateway/SNALink on a Series 300, the Series 800's Gateway/SNA 3270 emulates a 3274 Model 51C cluster controller and up to 50 active 3278 Model 2 displays and/or 3287 Model 1 printers on the network per Gateway/SNALink.
- The terminals and printers supported by the HP 9000 Series 300 running HP-UX are also supported by HP-UX Gateway/SNA 3270.
- Emulates the PC3270 file transfer capability that communicates with the IBM supported file transfer application on the host.
- Local screen print, screen logging to disk, 3287 printer output redirection to printer, file, or user program.
- Message Encoding: Supports NRZ (Non Return to Zero). Does not support NRZI (Non return to Zero Inverted).

Features Not Supported

- For the 3278 Model 2 display: Programmed symbols, alt cursor, test key, keyclick, extended attributes, encryption.
- For the 3287 Model 1 printer: PA1, PA2, cancel print

* IBM is a registered trademark of International Business Machines Corporation. TSO, IMS, CICS, CMS, MVS, VM, ACF/VTAM, ACF/NCP, and System/370 are products of IBM.

** UNIX is a registered trademark of AT&T in the U.S. and other countries.

Product Requirements

An HP 9000 Series 800 and the latest version of the HP-UX operating system.

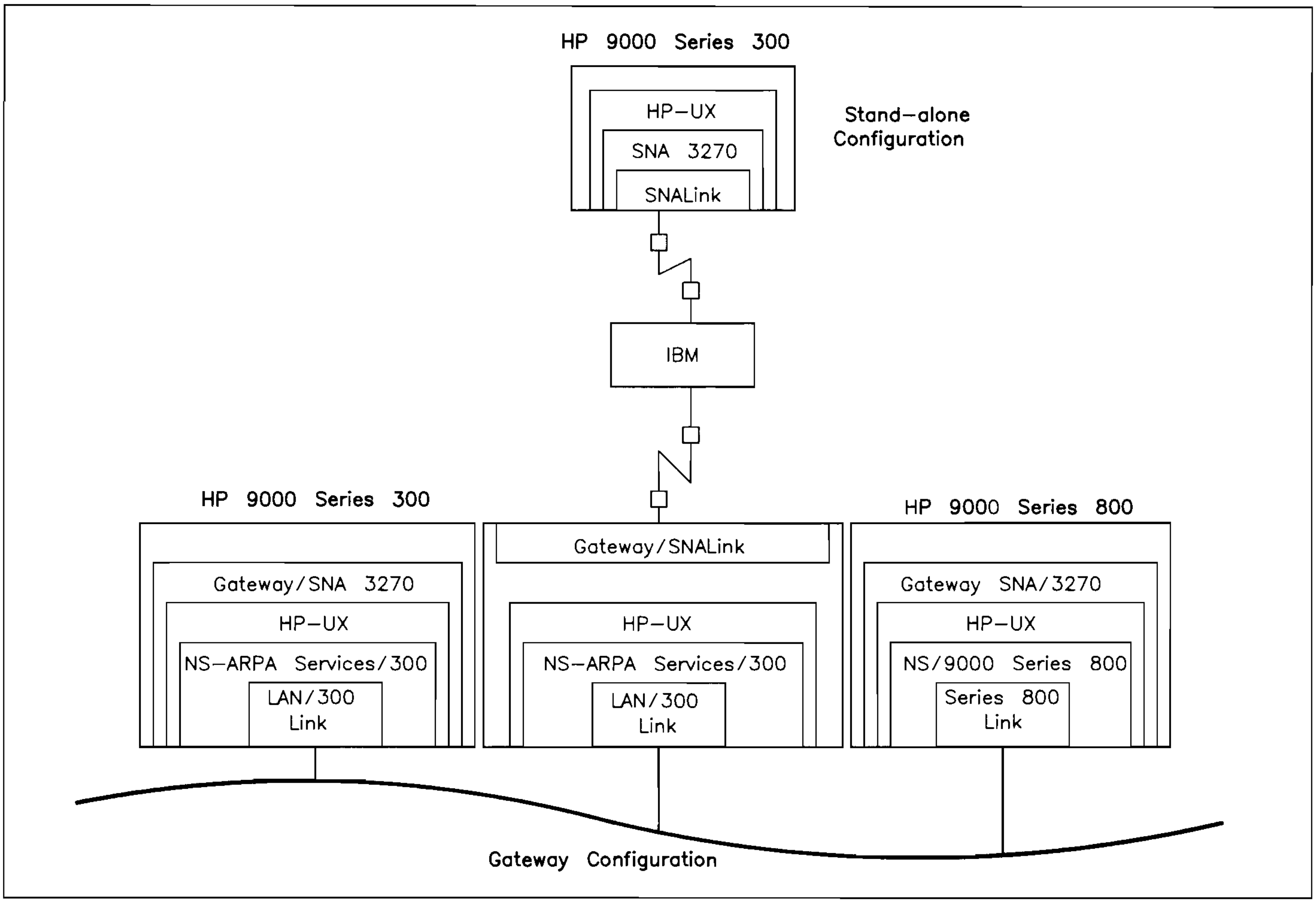
Access over a LAN to 36593A HP-UX Gateway/SNALink on a Series 300 which, in turn, is connected to the IBM system by a modem.

HP Network Services/9000 (NS/9000) Series 800 and LAN/9000 Series 800 Link are required to connect the processor to the LAN.

To ensure good performance, HP recommends 250 kilobytes of memory in addition to what would be required without the SNA product. This will accommodate the first active session. Incremental 3270 or 3287 sessions will require approximately 120K bytes each. One half megabyte of disc space is recommended to accommodate the Gateway/SNA 3270 product.

Customer Installation Responsibility

The product is customer installable. User installation aids such as an automated installation procedure and an IBM host generation guide are provided to simplify the process. Additional assistance can be provided by an HP Systems Engineer on a time-and-materials basis. An additional support service, Network Configuration Checkout (NCC), is available from HP to aid in the timely resolution of network problems. This is required as part of the NS/9000 installation process and has been found valuable in ensuring support for HP-to-IBM communications and identifying unsupportable configurations. For additional information, contact your Hewlett-Packard representative.



Ordering Information

Product Number	Product Description
36911A	HP-UX Gateway/SNA 3270 for Model 840, Right-to-Use license and first copy of software. Provided on ¼-inch cartridge or 1600 BPI magetic tape.
36911R	HP-UX Gateway/SNA 3270 Right-to-Copy and sublicense for an additional copy. Requires prior purchase of 36911A.
36918A	HP-UX Gateway/SNA 3270 for Model 825.
36918R	Right-to-Copy.
36919A	HP-UX Gateway/SNA 3270 for Model 850.
36919R	Right-to-Copy.

YOU MUST ORDER A MEDIA OPTION. Requires access to 36593A HP-UX Gateway/SNALink. Also requires NS/9000 for the Series 800 and LAN/9000 Series 800 Link.

Media Options:

Opt. AA0	¼-inch cartridge tape
Opt. AA1	1600 bits per inch magnetic tape

Documentation

Document Number	Document Description
36590-61000	HP-UX/SNA 3270 and Gateway/SNA 3270 Reference Manual. The following documentation is included to help IBM host site personnel to configure ("gen") the mainframe for HP-to-IBM communications:
5958-8542	HP SNA Products: Manager's Guide
5958-8543	HP SNA Products: ACF/NCP and ACF/VTAM Guide
5958-8545	HP SNA Products: IMS Guide
5958-8546	HP SNA Products: CICS Guide

Support Products

36911A+S00	Software Materials Subscription (SMS) for Model 840.
36911A+W00	Extended Software Materials Subscription (EMS)
36918A+S00	SMS for Model 825.
36918A+W00	Extended SMS
36919A+S00	SMS for Model 850.
36919A+W00	Extended SMS.

Response Center Support and Account Management Support customers must also order the Data Communications Category Support C, if it has not already been purchased.

HP-UX Gateway/SNALink

HEWLETT PACKARD

For HP 9000 Series 300 Computer Systems

Product Number HP 36593A

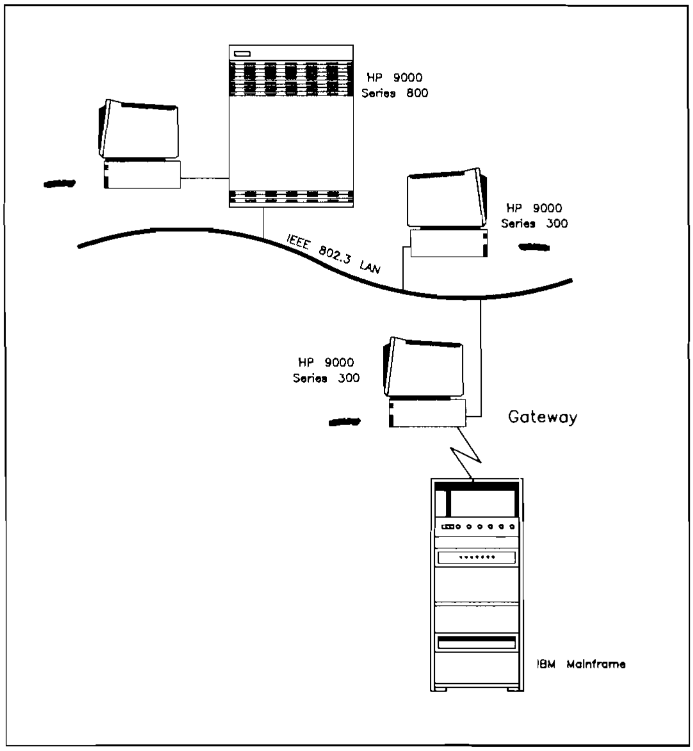
HP-UX Gateway/SNA 3270 provides communications between an HP 9000 Series 300 and an IBM* System/370-compatible mainframe using SNA 3270 protocols. Gateway/SNALink manages the SDLC line to the IBM system and emulates the major features of an IBM 3274 cluster control unit using the lower four levels of SNA. Physical Unit Type 2 (PU2), Logical Unit Types 1, 2, and 3 (LU1, LU2, LU3) protocols are emulated.

Gateway/SNALink supports, and is used in conjunction with, an HP-UX SNA emulation service product such as HP-UX Gateway/SNA 3270. When used with the gateway services on Series 300 or 800 nodes on a LAN, Gateway/SNALink allows HP terminals, monitors, and printers on the LAN to emulate the functions of IBM 3278 terminals and 3287 printers. Important extensions to these IBM devices such as file transfer and multiple sessions are also provided by Gateway/SNA 3270 and supported by Gateway/SNALink.

Gateway/SNALink and Gateway/SNA 3270 allow access to 3270 applications on the mainframe such as TSO, CMS, IMS, and CICS.

Key Features

- Allows the HP 9000 to emulate the major features of a 3274 cluster controller using SNA (PU2, LU1, LU2, LU3) protocols.
- Supports up to 50 active Logical Session Units (LUs) as terminals and printers concurrently over the LAN.
- Supports HP 9000 Series 300 and Series 800 systems running HP-UX and communicating with the gateway processor over a LAN.
- Non-dedicated. Depending on the customer's specific application requirements, HP-UX Gateway/SNALink easily operates on a Series 300 while running other HP-UX applications.
- Supports SDLC line speeds up to 19,200 baud over switched or leased lines.
- Powerful commands to monitor and control the operation of the product.
- Microprocessor-based SNA interface card offloads communications line activity resulting in zero CPU overhead on the HP-UX processor when data is not being transmitted or received.



Functional Specifications

- When used with HP-UX Gateway/SNA 3270, the pair of products emulate a 3274 Model 51C cluster controller and up to 50 active 3278 Model 2 displays and/or 3287 Model 1 printers.
- All terminals and printers supported by the HP 9000 Series 300 running HP-UX are supported by HP-UX Gateway/SNALink.
- Allows communication with a 370-compatible mainframe running VM, MVS/XA, or MVS/SP operating systems and ACF/VTAM through an IBM 3705 or 3725 communications controller running ACF/NCP.
- Emulates the PC3270 file transfer capability that communicates with the IBM supported file transfer application on the host.
- Up to four HP-UX Gateway/SNALink products may be supported by a single Series 300 system. This allows links to multiple hosts.

* IBM is a registered trademark of International Business Machines Corporation. TSO, IMS, CICS, CMS, MVS, VM, ACF/VTAM, ACF/NCP, and System/370 are products of IBM.

- Supports synchronous modem speeds to 19,200 baud. Although HP fully supports the following modems, other configurations will also work. HP-UX Gateway/SNALink requires modems which support the following handshake signals: DTR, DSR, RTS, CTS, CD, RI. Modems must also receive clocks for the SDLC interface card and ensure ground isolation between the communicating systems. Supported modems include: Bell 201C, Bell 208A & 208B, Bell Dataphone II 2024A, 2048A, & 2096A.
- Message Encoding: Supports NRZ (Non Return to Zero). Does not support NRZI (Non return to Zero Inverted).

Features Not Supported

- For the 3278 Model 2 display: Programmed symbols, alt cursor, test key, keyclick, extended attributes, encryption.
- For the 3287 Model 1 printer: PA1, PA2, cancel print.

Product Requirements

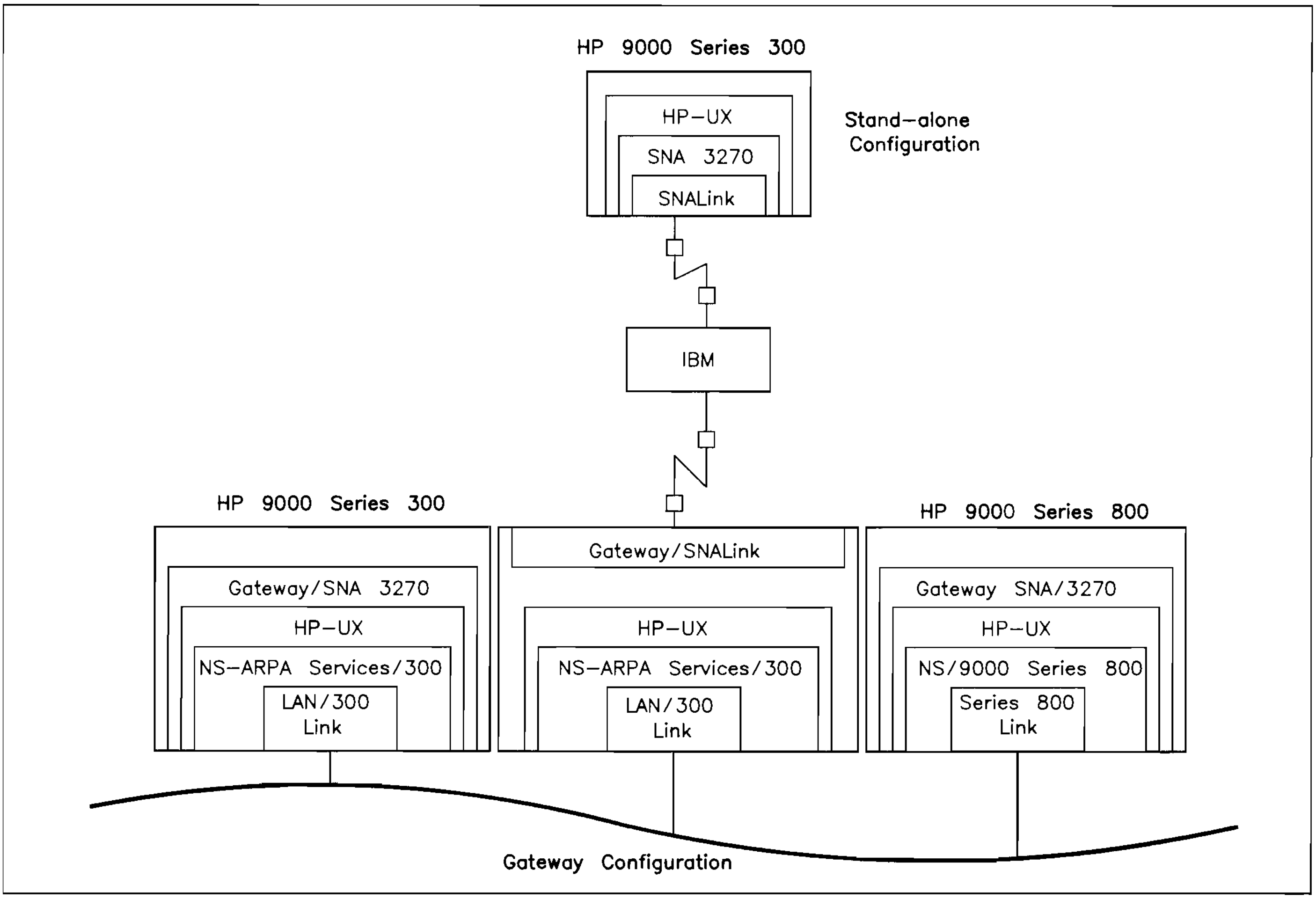
An HP 9000 Series 300 with the latest version of the HP-UX operating system.

Requires the latest version of Network Services 9000 (NS 9000) for the Series 300 and LAN/300 Link.

Supports, and is used in conjunction with HP-UX Gateway/SNA 3270 or other supported HP-UX "gateway" service product. To ensure good performance, HP recommends the gateway processor have at least 455 kilobytes of locked memory in addition to what would be required without the SNALink. Addition of Gateway/SNA 3270 to the gateway processor or other nodes on the LAN will require 250K bytes on each node with the Gateway/SNA 3270 emulation product. This will accommodate the first terminal or printer session on that node. Incremental sessions require 120K bytes per session on their respective processors. Additional Gateway/SNALink products require 340K bytes of memory on the gateway processor per link.

Customer Installation Responsibility

The product is customer installable. User installation aids such as an automated installation procedure and an IBM host generation guide are provided to simplify the process. Additional assistance can be provided by an HP Systems Engineer on a time-and-materials basis. An additional Network Configuration Checkout service (NCC) is offered by HP to assist in troubleshooting network problems in a timely manner. It is also used to ensure HP-to-IBM support and to identify unsupportable configurations. For additional information, ask your Hewlett-Packard representative for details.



Ordering Information

Product Number	Product Description
36593A	HP-UX Gateway/SNALink right-to-use license and software provided on ¼-inch cartridge (CS-80) tape. Includes interface card and cable. Requires modem. Supports HP-UX Gateway/SNA services such as HP-UX Gateway/SNA 3270 on LAN-based Series 300s and Series 800s for multi-user and multi-system communications.

Documentation

Document Number	Document Description
36592-61000	HP-UX SNALink and Gateway/SNALink Manager's Reference Manual.
36592-61001	SNA/9000 Manual Reference Pages. (Included. Sometimes referred to as "MAN Pages".)
36593K	Additional Reference Manual(s) and MAN Pages.

Support Products

36593A+S00	Software Materials Subscription
36593A+W00	Extended Software Materials Subscription

Response Center Support and Account Management Support customers must also order the Data Communications Category Support C, if it has not already been purchased. Customers with hardware support agreements must add the appropriate level of coverage (SMCC or BMMC) for this link product to their support agreement.

Network Services for the DEC VAX Computer

For Communication with HP Computer Systems

Product Number HP 50950A

Network File Transfer

This product implements HP's Network Services (NS) on the Digital Equipment Corporation (DEC) VAX™/VMS™ family of computers. It permits Network File Transfer (NFT) between VAX/VMS and HP 1000, HP 3000, and HP 9000 Series 300, Series 500, and Series 800 Model 840 computers.

For information on NFT support of other Series 800 computers to DEC VAX computers contact your local HP Sales Office.

Three objectives of file transfer between HP and DEC are:

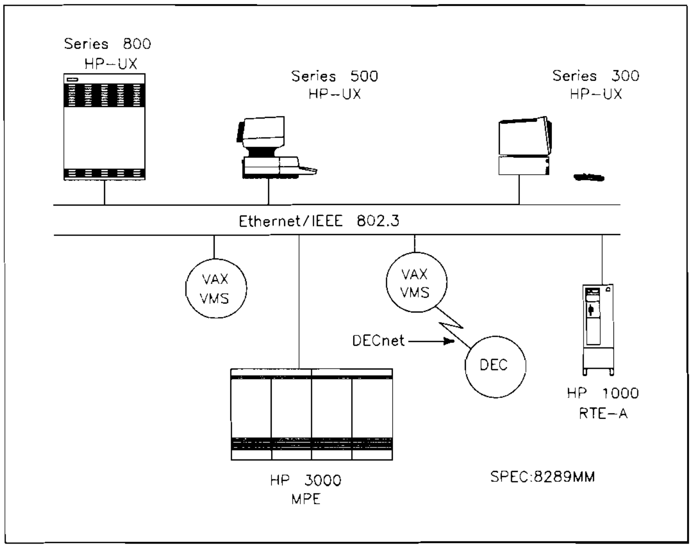
- Co-existence in a friendly manner with other VAX programs, including DECnet software
- Transparent and reliable file transfer
- Co-existence with DECnet systems on the same Ethernet/IEEE 802.3 cable.

NS for the DEC VAX Computer Features and Benefits

Feature	Benefit
Transfers to/from remote DECnet nodes	Increased productivity
Adheres to VAX/VMS Digital Command Language (DCL) grammar	Easy to learn and use
High-speed ASCII/Binary file transfer throughputs	Saves time
Access to NS by application programmers	Unassisted process execution
Provides record level file manipulation	Eliminates file formatting utilities
Runs concurrently with DECnet through DEC controllers and operates on VAXclusters™	Leverages installed DEC hardware

NFT is performed by executing the `dscopy` command. `dscopy` transfers files between file systems under the control of your local computer. A local file system is defined as the file system on the computer you are logged into and remote file systems are located on all the other computers on the network. The syntax of the `dscopy` command follows:

```
$ dscopy [command qualifier] -
source_file_specification -
destination_file_ specification
```



NS for the DEC VAX in a Local Area Network

Command qualifiers are:

`/print`	display `dscopy` results
`/interactive`	maintains connection for multiple transfers
`/replace`	overwrite contents of target file
`/search`	used to delineate records in source file
`/reclen`	sets variable or fixed record size
`/fixed`	source transfers uniform records
`/binary`	destination node receives data verbatim
`/delimiter`	define an end-of-record character
`/filepassword`	permits use of lockwords on the HP 3000

Source = node sending the transferred file
Destination = node receiving the transferred file

Expansion of source or destination file specification:

```
hpnode#username:passwd#file_pathname
```

Sample Commands

The following example shows an HP-UX user transferring a local file *myfile.c* to the VAX/VMS computer at node *VAX1.* The destination node has a user name of *tom* with a password of *thumb;thatfile.c* created in the home directory of *tom.*

```
$ dscopy myfile.c -
VAX1#tom:thumb#thatfile.c
```

DEC, DECnet, VAX, VMS, and VAXcluster are trademarks of Digital Equipment Corporation.

A VAX computer user sends a file *tfile.dat* to the HP 3000 at node *hpnod3K* with a login of *pery.act* and password *mypass*. The source file becomes *tfile* on *hpnod3K*.

```
$ dscopy tfile.dat -
''hpnod3K#pery/mypass.act#tfile''
```

A VAX computer user sends a file *tfile.dat* to the HP 1000 at node *hpnod1K* with a login of *user6* and password of *kappa*.

```
$ dscopy tfile.dat -
''hpnod1K#user6/kappa#tfile''
```

NS supports the transfer of sequential files with the following record formats:

- stream_lf
- stream_cr
- stream_other
- variable
- fixed
- variable fixed control
- user defined control

User Interface

The following commands are additions to the Digital Command Language:

`dscopy`	file transfers between HP AdvanceNet and DEC VAX computers
`npowerup`	configure the node on the network, or display the configuration of the node
`nusers`	limit or monitor the remote users of the node
`nlinkloop`	test the connectivity to one other node on the LAN, on nodes producing IEEE 802.3 packets
`nreadstat`	reads the controller registers of the Micom-Interlan controller
`nclearstat`	clears the controller statistics registers of the Micom-Interlan controller

Configuration Information

NS requires one of the following controllers on the DEC VAX or MicroVAX:

- Ethernet/IEEE 802.3 to UNIBUS Communications Controller (DELUA™)
- DEC Ethernet UNIBUS Network Adaptor (DEUNA™)
- DEC Ethernet Q-bus Network Adaptor (DEQNA™)
- Micom-Interlan NI1010A Unibus Communications Controller

The DEQNA runs on Micro VAX systems while the other controllers run on VAX computers from the 725 Series through the 8800 with UNIBUS backplanes. Currently, HP does not support NS with the DEBNT for VAXBI™ VAXes. The DEC controllers did not support IEEE 802.3 communications prior to VAX/VMS 4.4, so HP 1000, HP 9000 Series 800, and HP 3000 to VAX communications with versions before VMS 4.4 require the Micom-Interlan Controller. NS for the VAX will support DEC VAX computers directly connected to the Local Network Interconnect (DELNI™).

One of these hardware controllers and its accompanying software driver allows VAX/VMS systems to transfer files to and from the HP systems which support NS IEEE 802.3 communications. DEC supports thinwire connections with the DEST-AA transceiver.

Required Hardware on the DEC Computer

Configuration with DEC Components

DELUA-M Controller

- Cabinet Kit
- H4005 Transceiver
- BNE3 [H, K, L, M] - XX cable

or

- DEST-AA Thinwire Adaptor
- BC16M-XX Cable
- DECnet Driver
  - Media/Doc.Q [C,D,E,K,M] DO4-HG
  - License Q [C,D,E,K,M] DO4-UZ

DEQNA-M Controller

- H4000 or 4005 Transceiver
- BNE [3,4] [A,B,C,D,H,K,L,M] - XX cable

or

- DEST-AA Thinwire Adaptor
- BC16M-XX Cable
- DECNet Driver
  - Media/Doc.QZ DO4-H [3,5]
  - License QZDO4-UZ

DEUNA-M Controller

- Cabinet Kit
- H4000 or H4005 Transceiver
- BNE3 [A,B,C,D,H,K,L,M]-XX Cable

or

- DEST-AA Thinwire Adaptor
- BC16M-XX Cable
- DECnet Driver
  - Media/Doc.Q [C,D,E,K,M] DO4-HG
  - License Q [C,D,E,K,M] DO4-UZ

Configuration with Micom-Interlan Components

- NI1010A Unibus Ethernet Communication Controller
- NT100 or NT10 Ethernet Transceiver
- NA1040-XXX Transceiver Cable
- AC-NM1010A Flat Cable
- UM-NI1010A User's Manual
- SDK-NS2030 VMS Driver

DELUA, DELNI, DEUNA, UNIBUS, VAXBI, DEBNT, and DEQNA are trademarks of Digital Equipment Corporation.

Controller Comparison

Characteristics	DELUA	DEUNA	DEQNA	Micom-Interlan
DECnet	Yes	Yes	Yes	No
BUS	Unibus	Unibus	Q-bus	Unibus
Ethernet*	VMS 4.3/NS 1.0	VMS 4.0/NS 1.0	VMS 4.0/NS 1.1	VMS 4.0/NS 1.0
IEEE 802.3*	VMS 4.4/NS 1.1	VMS 4.4/NS 1.1	VMS 4.4/NS 1.1	VMS 4.0/NS 1.0
Power Draw (amps)	8	15	3.5	7.5
Space (hex slots)	1	2	1	1
Controller Sharing†	Yes	Yes	Yes	No
Link Level Diagnostics*	VMS 4.4/NS 1.1	VMS 4.4/NS 1.1	VMS 4.4/NS 1.1	VMS 4.0/NS 1.0
VAX Model	725-8800	725-8800	MICROVAX	725-8800

** Minimum VMS/NS versions*

† Multiple networking software products may access the controller simultaneously.

Targeted Machines

- MicroVAX II
- VAX 11/725 - 8800
- VAXclusters

Supported VAX Systems

- MicroVAX II
- 11/730, 750, 780, 785
- 8200
- 8500
- 8600
- 8650
- VAXcluster™

Please consult an HP Sales Representative for current information regarding VAX models that may require field testing.

Operating System Requirements

Computer	Minimum Operating System
VAX	VMS 4.0
MicroVAX I, II	MicroVMS 4.0
HP 1000	RTE 4.1
HP 3000	UBMIT*
HP 9000 Series 300	HP-UX 5.17
HP 9000 Series 500	HP-UX 5.1
HP 9000 Model 840	HP-UX 1.0

Ordering Information

Software

50950A NS for the DEC VAX computer with an option number includes media, *NS for the DEC VAX User's Manual,* Part Number 50950-90000 and *NS/NFT Cross-system Manual,* Part Number 5954-8563. You must specify one option number with each order.

50950A NS for the DEC VAX Computer

Option	Media	System
044	5-1/4 inch floppy (RX50)	MicroVAX
200	TK50	MicroVAX
300	9-track	11/725-85X0
400	9-track	8600-88X0

50950R Right-to-Reproduce NS for the DEC VAX Computer

Option	System
044	MicroVAX
200	MicroVAX
300	11/725-85X0
400	8600-88X0

Support Services

50950A+S00	SMS for NS for the DEC VAX computer
50950A+W00	Extended SMS for NS for the DEC VAX computer
99087M+C00	Datacomm Category C Support
99087M+V00	Extended Datacomm Category C Support
99087M+Q00	Manual Update Service for Datacomm Category C Support
99064M+H51	RCS for non-HP operating systems (DEC)
99064M+V51	ASC for non-HP operating systems (DEC) - with media
99064M+V00	ASC for non-HP operating systems (DEC) - without media
99064M+S51	SMS for non-HP operating systems (DEC)
99064M+W00	Extended SMS for non-HP operating systems (DEC) - without media
99064M+W51	Extended SMS for non-HP operating systems (DEC) - with media

**HP 3000 may require MPE V/E extended microcode.*

HP-UX Asynchronous Data Communications: cu, uucp, uux

HEWLETT PACKARD

For HP 9000 Series 200, 300, 500, and 800 Computer Systems

cu, uucp, and **uux** for HP 9000 Series 200, 300, 500 and 800 HP-UX systems are included in the HP-UX* operating systems.† Following are detailed descriptions of each capability.

cu

The **cu** command is used for communicating with other HP-UX or UNIX® systems. **cu** can be used over direct or modem lines and at speeds ranging from 100 to 19200 bits per second.

The options and features for **cu** are:

- Date rate – Speeds are determined by the device files available. Speeds up to 19200 bits/second are available, but speeds above 9600 bits/second are not recommended.
- Echo – local echo enabled or disabled
- Handshaking – ENQ/ACK enabled or disabled
- Parity – odd, even, or no parity
- Simple buffering
- ASCII file transfer

As an example of the use of **cu**, suppose that you want to communicate with a remote HP-UX system through a direct connection. The remote system requires that communications be at 9600 bits per second. If the direct connection is through a line named *tty 12,* the command to establish the connection could be:

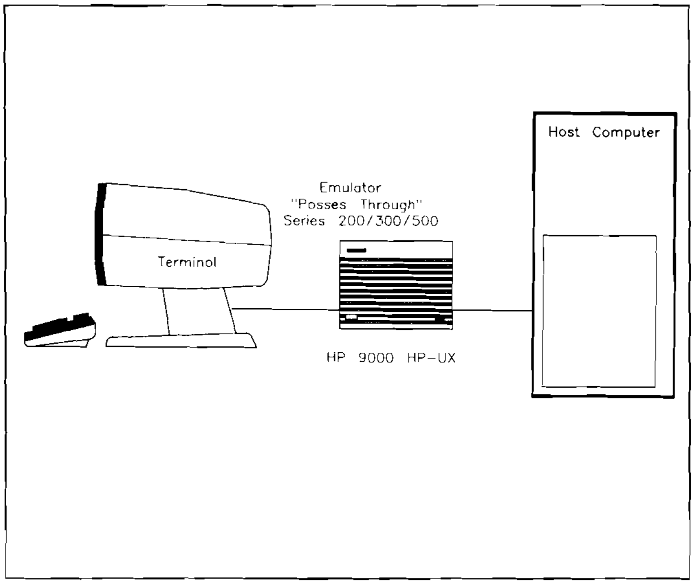
```
cu - 1tty12 dir
```

cu does not perform error checking.

Also included in **cu** are:

- Umodem
- Kermit, by Columbia University – Kermit provides reliable file transfers, re-transferring a file if necessary.

Two Methods of Using cu with Host Computer



uucp

The **uucp** command is used to copy files between HP-UX systems, or between HP-UX and UNIX systems. **uucp** is required to access the remote mail capabilities of UNIX.

uucp features include:

- Forwarding of mail to other systems
- Specifying a protocol
- Adding a prefix to a phone number field
- Reporting the status of a job
- Rejuvenating a job
- X.25 support
- f-protocol

When you copy a file between systems, you need to specify the system name and file name for both the source and the destination of the transfer. **uucp** then automatically checks which line should be used to connect to the remote system, establishes the connection, and transfers the specified file.

Files are transferred in 64-byte packets. Error checking is done after each packet is received. If an error is indicated on a received packet, that packet is re-transmitted up to five times. If that packet cannot be received after these attempts, the transmission is aborted‡.

*HP-UX is Hewlett-Packard's enhanced version of AT&T's System V UNIX Operating System.

†cu, uucp, and uux are not included in Series 300 HP-UX Application Execution Environment.

‡This the "slow" protocol used for unreliable connections (like phone lines). A second protocol can be selected when using reliable connections (like an X.25 connection or a hard-wired connection). This second protocol is faster; data is transferred in 256 byte packets and all packets are transferred without waiting for confirmation. A check-sum is transmitted at the end.

As an example, if you wish to copy a file named */inventory/price_updates* on your system to a file named */ledger/price_changes* on a remote system named *design_2*, you could execute the command:

```
uucp /inventory/price_updates
design_2!/ledger/price_changes
```

Note: This command must be typed on one line to execute properly.

uucp then automatically checks for a connection to the remote system *design_2*, establishes the connection, transfers the file, and then takes down the connection.

Before you can use **uucp**, several files must be added to your HP-UX system to make connections possible. After these files have been added (usually by the system manager), you only need to specify the system and file names when using **uucp**.

If a system is able to respond to requests from other systems, it can transfer files with **uucp**. After an originating system has completed its file transfers, it checks to see if the responding system has any pending files to be transferred in the other direction. The responding system, however, has no control over how often a requesting system checks for pending file transfers.

uux

The **uux** command is used to execute commands on remote HP-UX or UNIX systems. Every HP-UX system contains a file that specifies which routines can be executed by users on remote systems, so you can protect your system against unauthorized remote execution of commands.

One possible use of **uux** would be to invoke a shellscript named *status_check* on a remote system. If a system named *design_2* has been configured to let you execute this command, you could start this process by entering:

```
uux design_2!status_check
```

If you attempt to execute a command on a remote system and you are not allowed to execute that command, you are notified through the remote mail facility.

Modems

Hewlett-Packard presently supports the HP 92205A Hayes Smartmodem 1200 and several other modems on Series 200, 300 and 500 HP-UX. Please check the appropriate configuration guide for updated lists of qualified modems.

The supplied **dialit** code includes examples of autodial software for the above modems. **uucp** does not support the RS-366 Bell 801 autodial standard.

Configuration Information

Workstation

- Models 217, 220, 236, 236C, 237, 310, 320, 520, 530, 540, 550, and 560 are supported.

RAM

- 2 Mbytes

Interfaces

Series 200/300

- 98626A RS-232C (Serial) Interface
- 98628A Datacomm Interface
- 98642A 4-channel Multiplexer (for HP-UX 5.0 and later)
- 98644A RS-232C (Serial) Interface (with HP-UX 2.1 or later)
- Model 217 built-in serial interface (with HP-UX 2.2L and later)

Note: All applications require a cable with a DTE (male) connector.

Series 500

- 27128A Asynchronous Serial Interface [most applications require a cable with a DTE (male) connector]
- 27140A Asynchronous 6-channel Multiplexer (for direct connections and connections to a modem on all six channels)
- 27130B Asynchronous 8-channel Multiplexer (cannot be used to connect to a modem)

Series 800

- 27140A Asynchronous 6-channel Multiplexer (for direct connections and connections to a modem on all six channels)

Ordering Information

Software

Series 200/300

- **cu, uucp,** and **uux** software is included with the Series 200 and 300 HP-UX Operating System.

Series 500

- **cu, uucp,** and **uux** software is included with the Series 500 HP-UX Operating System.

Series 800

- **cu, uucp,** and **uux** software is included with the Series 800 HP-UX operating system.

Support Services

Support is included with HP-UX operating system support services.

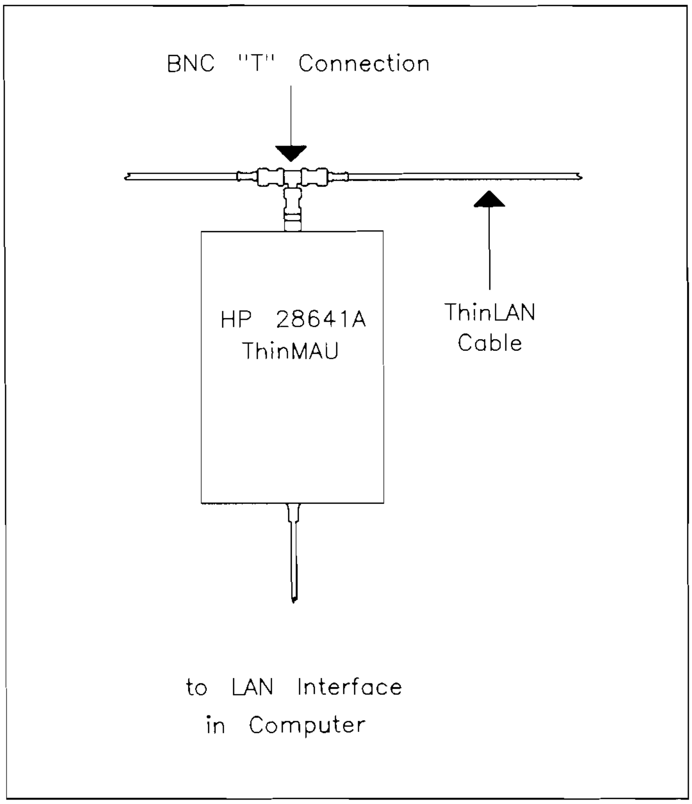
ThinMAU

HEWLETT PACKARD

For HP's ThinLAN Local Area Network

Product Number HP 28641A

The ThinMAU is specifically designed for connection to the 10 Mbps IEEE 802.3 Type 10BASE2 "ThinLAN" coaxial cable. It includes an integrated, 1M Attachment Unit Interface (AUI) cable for connection to the computer's LAN interface controller card. Access to the coaxial cable is through a BNC "T" Connector. Like the HP 30241A MAU for "thick" coax, the ThinMAU sends and receives data, detects collisions on the network, and protects the network's reliability by monitoring malfunctions between itself and the computer. The figure at right illustrates the ThinMAU and "T" Connector.



Features

- Operationally compatible with IEEE 802.3 Type 10BASE2 standard protocol
- Compact, convenient design
- Integrated 1M AUI cable
- Jabber and heartbeat functions minimize the impact of controller, AUI, and ThinMAU malfunctions on the rest of the network
- User installable

Functional Description

The HP 28641A ThinMAU provides a physical and electrically isolated connection between HP LAN interface products and the ThinLAN coax. For connection to IEEE 802.3 Type 10BASE5 "thick" coax, refer to the 30241A MAU. Using the ThinMAU with non-HP equipment and with extension AUI cables are unsupported configurations.

Collision Detection, Jabber, and Heartbeat

Collisions occur when two or more nodes transmit simultaneously on the LAN. The ThinMAU provides collision detection by detecting signals on the coax other than those provided by its transmitter, and signaling the LAN controller.

The jabber circuitry simply turns off the transmitter if the computer tries to send data for an abnormal amount of time, preventing a faulty LAN node from tying up the network indefinitely.

The heartbeat function is a signal sent from the MAU to the LAN interface following each successful transmission. The signal exercises most of the collision detection circuitry when generated and functions to inform the controller that the collision detection path is operating correctly.

Compatibility

The ThinMAU can be used with the following LAN interfaces:

- 98643A - I/F for HP 9000/Series 200, 300
- 27125B - I/F for HP 9000/Series 500
- 12076A - I/F for HP 1000 A-Series
- 30242A - I/F for HP 3000 Computers
- 36921A - I/F for HP 3000/Series 900
- 91786A - I/F for HP 9000/Series 800 Model 825
- 98194A - I/F for HP 9000/Series 800 Model 840
- 91788A - I/F for HP 9000/Series 800 Model 850

Note:

- 98643A/27125B/98194A/91786A/91788A includes ThinMAU
- 98643A/27125B #241/98194A/91786A/91788A #841 deletes ThinMAU for separate ordering
- 12076A/30242A/36921A #241 deletes 30241A MAU for separate ordering of ThinMAU
- 27125B/12076A #001, which replaces IEEE 802.3 card cable with Ethernet card cable is not supported with use of ThinMAU
- 98643A does not have a card edge cable

Functional Specifications

Environmental Characteristics

Meets Environmental Class B Specifications

Operating temperature: 0° to 55°C

Humidity: 5% to 95% relative at 40°C non-condensing

Physical Characteristics

Dimensions: 100 mm (4.5 in.) by 152 mm (6 in.) by 25 mm (1 in.)

Approximate Weight: 454 grams (16 oz.)

Electrical Characteristics

Power Requirement: 8.0 - 13.5 V

Power Consumption: 4.3 Watts (typical); 5.1 Watts (maximum)

Electrical Isolation: 500V AC RMS from coax to AUI cable

Ordering Information

The 28641A ThinMAU includes:

28641-60001	ThinMAU Assembly
28641-90001	Installation Manual
1250-0781	BNC "T" Connector
1252-1154	Plastic Boot

The ThinMAU, LAN cabling, and installation tools are available through Direct Marketing Division. The following documentation on hardware and installation for the entire network, including the ThinMAU, is also available through Direct Marketing Division:

5955-7680	Cable and Accessories Installation Manual
5955-7681	LAN Link Troubleshooting Manual

Please refer to the HP Computer User's Catalog for ordering information.

Medium Attachment Unit

For HP 1000, HP 3000 and HP 9000 Computer Systems

Product Number HP 30241A

The Medium Attachment Unit (MAU) and the included coaxial cable tap together provide a physical and electrical connection to a 10 Mbps IEEE 802.3 Type 10BASE5 "thick" Local Area Network coaxial cable. The MAU transmits and receives data, detects collisions on the network, and improves the network's reliability by monitoring malfunctions between itself and the computer. The figure at right illustrates the MAU and tap.

Features

- Operationally compatible with IEEE 802.3 Type 10BASE5 standard protocol
- Easy tap installation and removal without disturbing network uptime
- Jabber and heartbeat functions minimize the impact of controller, AUI, and MAU malfunctions on the rest of the network

Functional Description

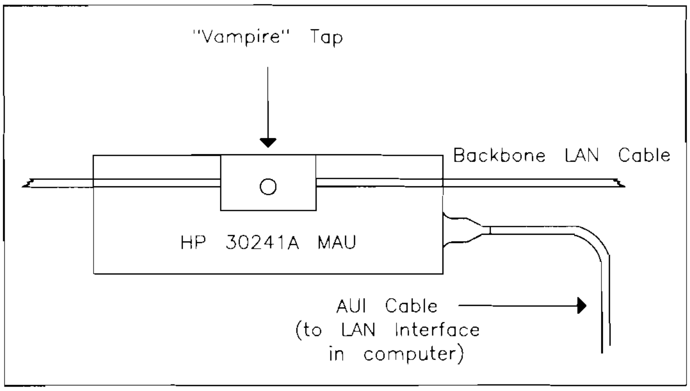
The HP 30241A MAU provides HP computer system LAN interface products with an electrically isolated connection to the thick "backbone" coaxial cable specified in the 802.3 Type 10BASE5 standard. For connection to the Type 10BASE2 ThinLAN cable, refer to the HP 28641A ThinMAU. Using the MAU with non-HP LAN interfaces is an unsupported configuration.

Transmit and Receive Functions

A few special functions are performed by the transmitter and receiver. The receiver amplifies and reshapes the signal before passing it to the Attachment Unit Interface (AUI) cable driver. Both perform a "squelch" function, which turns the cable drivers off when no data is being sent over the coax or from the computer. Finally, the receiver monitors data sent by the transmitter, performing a local loop-back function.

Collision Detection, Heartbeat, and Jabber

When two or more nodes on the LAN transmit simultaneously, a collision occurs. The MAU detects collisions by measuring a rise in the DC voltage on the coax cable, and signaling the LAN controller.



The jabber circuitry turns the MAU transmitter off if the computer attempts to transmit for an abnormal amount of time. This ensures that a faulty LAN node will not tie up the network indefinitely.

The heartbeat function is a signal sent from the MAU to the LAN interface following each successful transmission. The signal exercises most of the collision detect circuitry when generated, and functions to inform the controller that the collision detection path is operating correctly.

Compatibility

The MAU can be used with the following LAN interfaces:

12076A	I/F for HP 1000 A-Series
30242A	I/F for HP 3000 Computers
27125B	I/F for HP 9000/Series 500
98643A	I/F for HP 9000/Series 200, 300
36921A	I/F for HP 3000/Series 900
91786A	I/F for HP 9000 Series 800 Model 825
98194A	I/F for HP 9000 Series 800 Model 840
91788A	I/F for HP 9000 Series 800 Model 850

Note:

- 12076A/30242A/36921A includes MAU
- 12076A/30242A/36921A #241 deletes MAU for separate ordering
- 27125B/98643A #241/98194A #841 deletes 28641A ThinMAU for separate ordering of MAU
- 27125B/12076A #001, which replaces IEEE 802.3 card cable with Ethernet card cable is not supported with use of MAU

Functional Specifications

Environmental Characteristics

Meets Environmental Class B Specifications

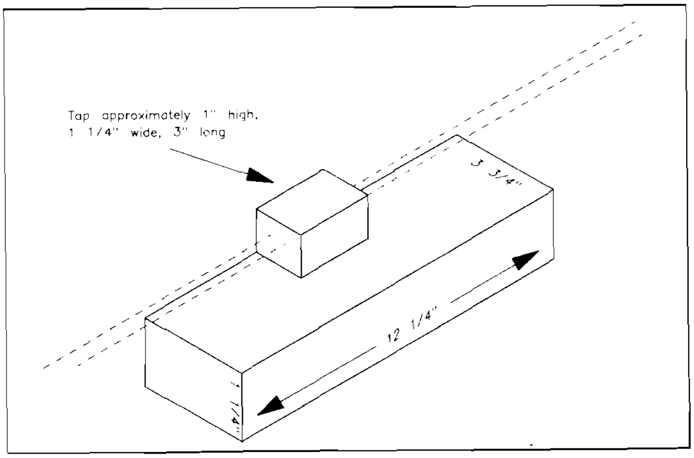
Storage Temperature: -10° to 65°C

Operating Temperature: 0°to 55°C

Humidity: 5% to 95% relative at 40°C non-condensing

Note: Not suitable for installation in ducts, plenums, or other environmental air spaces according to the National Electrical Code Article 300-22 b&c or equivalent.

Physical Characteristics



MAU Dimensions: 310mm (12.25in) long by 95mm (3.75in) wide by 30mm (1.25in) deep

Top Dimensions: 75mm (3in) by 30mm (1.25in) by 25mm (1in)

Electrical Characteristics

Power Requirement: 9.0 - 15.75 V

Power Consumption: 4.6 Watts (typical); 6 Watts (absolute max)

Electrical Isolation: 500V AC RMS from coax to AUI cable

Ordering Information

The HP 30241A includes:
30241-60101 MAU Assembly
0362-0819 Coax Cable Tap

The MAU, LAN cabling, and installation tools are available through HP's Direct Marketing Division. The following documentation on hardware and installation for the entire network, including the MAU, is also available through Direct Marketing Division:

5955-7680	Cable and Accessories Installation Manual
5955-7681	LAN Link Troubleshooting Manual

Please refer to the HP Computer User's Catalog for ordering information.

Repeater Kit

HEWLETT PACKARD

For HP's ThickLAN Local Area Networks

Product Numbers HP 92223A and 92223C

The HP 92223A Repeater Kit allows HP's Local Area Networking (LAN) customers to extend their ThickLAN (IEEE 802.3 type 10BASE5) coaxial segments. The repeater greatly expands the area that can be covered by a single, fully interconnected ThickLAN backbone. (See diagram at right.)

Features

- Transparent increase in size of network
- One product contains the entire solution (Repeater, AUIs, and MAUs)
- Maximum topology is two repeaters and three coaxial segments between any ends of the extended network
- Provides Auto-segmentation to prevent disruption of the network by an unterminated coaxial segment
- Up to 50 meter separation between repeater MAU and repeater via extended repeater AUI cables (two 5 meter repeater AUI cables provided)

Functional Description

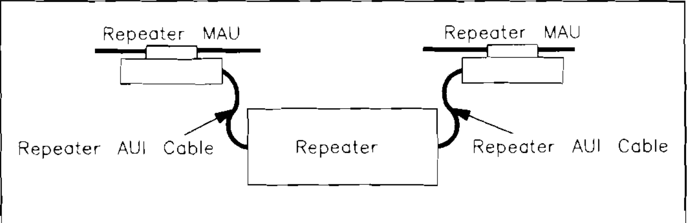
The HP 92223A Repeater Kit provides the means of extending local area networks beyond their limited single segment length (500 meters for backbone segments).

Repeater, Repeater MAU, and Repeater AUI Cable

The main function of the repeater is to transmit LAN signals from one cable segment to the other while maintaining synchronization across the network. If collision is detected, the repeater sends out a jam sequence to both network segments, causing the transmitting node to retransmit its message. Also, if the repeater detects more than 64 consecutive collisions from a segment, it "segments" that side, not passing any of its data to the other side until a packet of data is transmitted, without a collision to the "segmented" side.

The Repeater Media Access Unit (MAU) transmits signals onto the coaxial cable, receives signals from the cable, and detects any message collisions that may occur. It provides electrical isolation and high impedance to the coaxial cable. The Repeater Access Unit Interface (AUI) cable provides the actual physical connection of the repeater MAU and the repeater.

ThickLAN Backbone Extension with Repeater Kit



ThickLAN Backbone Extension with Repeater Kit

92223C 20 Meter Repeater FEP AUI

The 20 meter repeater AUI allows for increased separation of up to 50 meters between the repeater MAU and the repeater.

Compatibility

- The Repeater Kit allows the interconnection of ThickLAN (IEEE 802.3 type 10BASE5) coaxial segments.
- Compatible with Digital Equipment Corporation repeaters (DEREP-AA & DERREP-RA).
- The repeater MAUs and the repeater AUIs provided in 92223A and 92223C must be used with this product. The repeater cannot be connected to a network segment via the standard 802.3 AUI cables or with HP 30241A MAUs. (However, the repeater will operate on a network with nodes connected via the 802.3 AUI cables or the 30241A.)

Network Configurations

(See Cable and Accessories Installation Manual P/N 5955-7680 for details.)

Synopsis:

- A repeater can be connected to a repeater MAU with a maximum of 50 meters of repeater AUI cable.
- The maximum topology is two repeaters and three coaxial segments between any two ends of the extended network.
- The maximum number of nodes on a backbone segment is 100 (including repeater MAUs).

Functional Specifications

Environmental Characteristics

Operating temperature: 5° to 55°C

Humidity: 10% to 90% with a maximum wet bulb temperature of 28°C (82°F) and a minimum dewpoint of 2°C (36°F)

Physical Characteristics

Repeater Unit

Length:	45.72 cm (18 in.)
Width:	30.48 cm (12 in.)
Height:	10.16 cm (4 in.)
Weight:	7.26 kg (16 lbs.)

Repeater MAU

Length:	30.20 cm (11.9 in.)
Width:	9.5 cm (3.74 in.)
Height:	9.0 cm (3.54 in.)
Weight:	1.36 kg (3 lbs.)

Repeater AUI Cable

Length:	5 m (16.40 ft.)

Two 5 meter repeater AUI cables are provided with the 92223A product.

Electrical Characteristics

Repeater Unit requires:
115V AC @ 1.0A (50/60 Hz) or
230V AC @ 0.5A (50/60 Hz)

Ordering Information

The 92223A Repeater Kit includes:

- Repeater
- (2) Repeater MAU
- (2) Repeater AUI (5 meters FEP)
- (4) Barrel Insulating Boot
- Repeater Installation Manual
- Power Cord (Type depends on country)

The 92223C 20 meter Repeater FEP AUI:

- 20 Meter FEP AUI

These products can all be ordered directly from the Direct Marketing Division (DMK) through HP Sales offices by using DMK's toll free number 1-800-538-8787. In California, call (408) 738-4133.

Note: The Repeater Kit (P/N 92223A) is intended to be customer installable. Therefore, a customer who would like to have the repeater kit installed by HP must contract this installation service from their local sales office or project center.

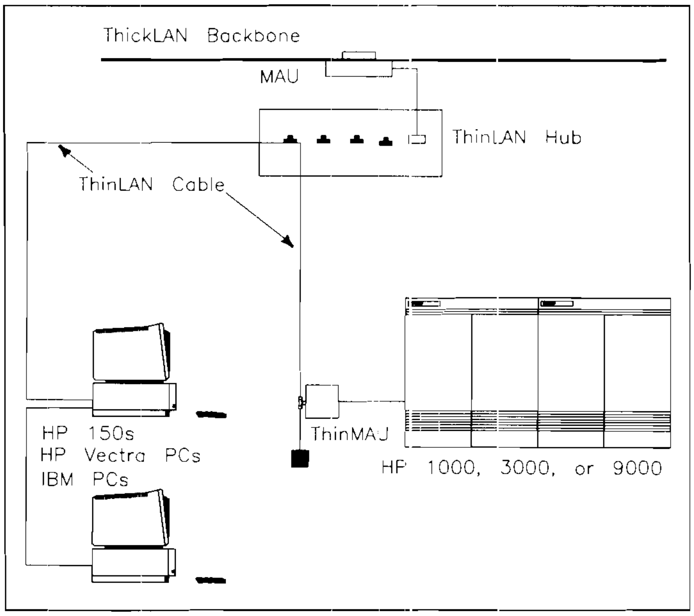
ThinLAN Hub

HEWLETT PACKARD

For IEEE 802.3 and Ethernet Local Area Networks

Product Number HP 28645A

The HP 28645A ThinLAN Hub is a 10Mbit multiport repeater for IEEE 802.3 and Ethernet local area networks. The ThinLAN Hub provides the primary means of interconnecting ThinLAN (IEEE 802.3 type 10BASE2) segments to a ThickLAN (IEEE 802.3 type 10BASE5) backbone. In addition, it can operate stand-alone to extend a ThinLAN network beyond the single segment limit of 185 meters and 30 nodes.



Features

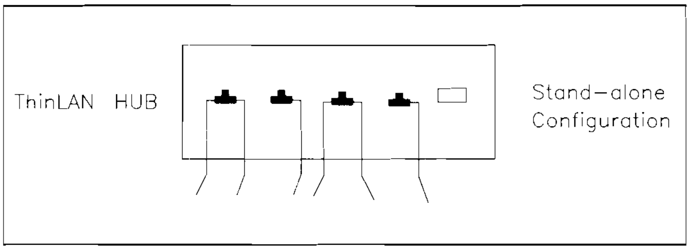
- One product contains entire solution (ThinLAN Hub, 30241A MAU, AUI cable, mounting bracket)
- Provides four ThinLAN ports (supporting eight ThinLAN cable runs) and one AUI Port
- Supports star wiring configurations
- Customer installable
- Provides auto-segmentation to automatically detect a failed ThinLAN segment and disconnect it from the rest of the network
- Transparent to network nodes and software
- Uses standard IEEE 802.3 AUI cable and 30241A MAU for ThickLAN connection
- Provides internal transceivers for the ThinLAN segments

Functional Description

The HP 28645A ThinLAN Hub has four ThinLAN ports and one AUI port. Each ThinLAN port supports a 185 meter ThinLAN segment and the AUI port supports a connection to a 500 meter ThickLAN backbone cable. A ThinLAN Hub port may be located anywhere along a ThinLAN segment. This allows a single port to support two cable runs as long as the total length is less than 185 meters. Therefore, the ThinLAN Hub can act as the concentrator for an eight-cable configuration.

The auto-segmentation feature of the ThinLAN Hub provides greater network reliability. Auto-segmentation automatically detects a failed segment and disconnects it from the rest of the network. The failed ThinLAN segment can be analyzed and serviced offline. Once the problem is corrected, the segment will automatically reconnect to the network. This feature provides greater network reliability and improved support.

In a standalone configuration, the ThinLAN Hub can support up to 116 workstations or systems.



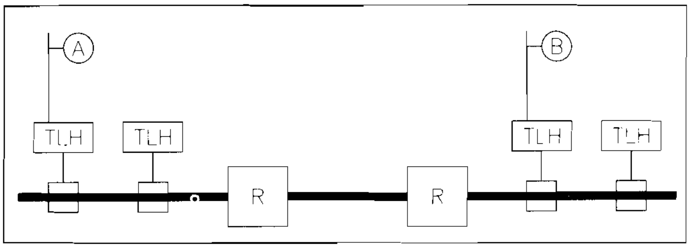
Local ThinLAN workgroups can then be easily connected to a ThickLAN backbone for interdepartmental communication. The ThinLAN Hub provides an economical solution for today's needs and a growth path for tomorrow.

Compatibility

- The ThinLAN Hub is compatible with IEEE 802.3 and Ethernet local area networks.
- The ThinLAN Hub is compatible with standard IEEE 802.3 AUI cable and 30241A MAU for ThickLAN connection.

Network Configurations

- The maximum supported topology is two repeaters (92223A) and two ThinLAN Hubs between any two nodes.



- A ThinLAN Hub can be connected to a 30241A MAU with a maximum of 50 meters of AUI cable.
- The maximum number of nodes on a backbone segment is 100 (including the hub) and 30 on a thin segment (including the hub).

Functional Specifications

Environmental Characteristics

Operating temperature: −10° to 55°C

Relative Humidity: 10% to 90% @ 40°C (104°F) non-condensing

Physical Characteristics

Dimensions: 325mm (12.8 in) × 325mm (12.8 in) × 104mm (4.1 in)

Weight: 4.27kg

Electrical Characteristics

Power Consumption:

Voltage (AC):	115 Volts (nominal)	230 Volts (nominal)
Current:	1 Amp (max)	0.5 Amps (max)
Power:	55 Watts (max)	55 Watts (max)
Frequency:	48 - 66 Hz	48 - 66 Hz

Ordering Information

HP 28645A ThinLAN Hub includes:

28645-60001	ThinLAN Hub Assembly
30241-60101	MAU Assembly
0362-0819	MAU Coax Tap
1250-0781	BNC "T" Connectors (4)
28645-63003	Five Meter AUI Cable
5001-5602	Wall Mounting Bracket
28645-90001	Installation Manual

28645A Options:

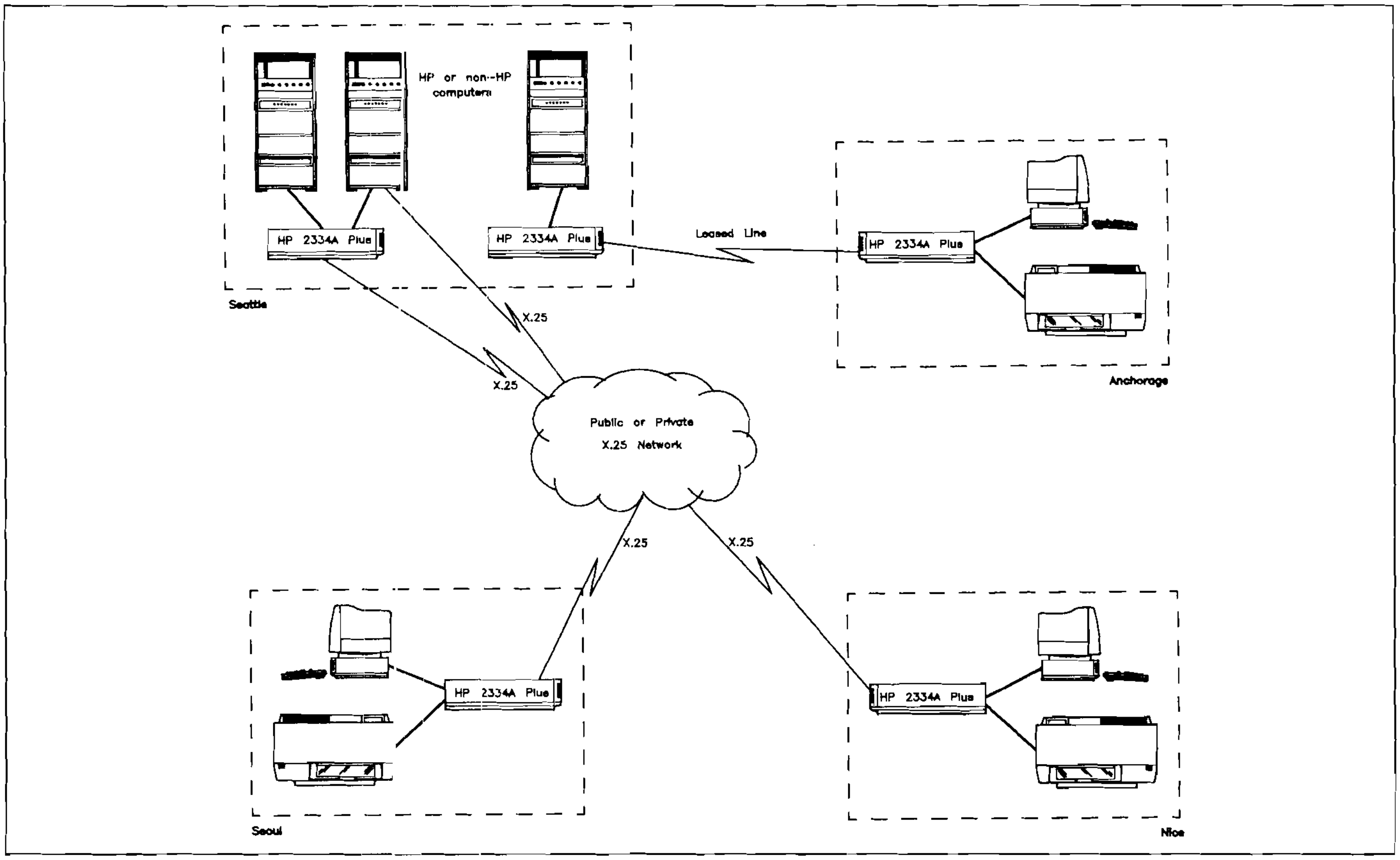
241: Delete MAU, Tap, AUI cable and Mounting Bracket
900: UK Power Cord
901: Australian Power Cord
902: European Power Cord
903: US/CAN 125V Power Cord
904: US/CAN 250V Power Cord
906: Swiss Power Cord
912: Danish Power Cord

Note: The ThinLAN Hub (P/N 28645A) is intended to be customer installable. Therefore a customer who would like to have the ThinLAN Hub installed by HP must contract this installation service from their local sales office or project center.

HEWLETT PACKARD

X.25 Multiplexer

For Connection to X.25 Wide Area Networks Product Number HP 2334A Plus



The HP 2334A Plus – X.25 Multiplexer is designed to connect up to sixteen remote workstations (terminals, personal computers, printers, plotters) to a host computer. Two types of configuration are supported for each HP 2334A Plus – X.25 Statistical Multiplexer and X.25 Cluster Controller.

Features and Benefits

- **Reduce communication cost**
  - Remote connection of up to 16 RS-232-C workstations; expandable from 4 to 16 devices in groups of 4.
  - Can work in statistical multiplexer or in cluster controller configuration over a leased line or an X.25 Packet Switching Network.
  - Implementation of X.25, X.3, X.28, X.29, X.121 November 1980 CCITT recommendations. The HP 2334A Plus is certified on most X.25 Public Packet Switching Networks.
  - Host computer port contention.
- **Transparent user access (in statistical multiplexer mode)**
  - Support of HP applications like HPWORD, HPDESK, AdvanceMail, etc.
  - New design results in high performance for remote terminals and printers.
  - Multiple computer access.
- **Friendly configuration**
  - Default configuration for statistical multiplexers using a leased line.
  - Menu-driven configuration.
  - Online modifiable parameters.

Functional Specifications

Asynchronous Ports Specifications

The HP 2334A Plus uses a 4-port interface that supports full-duplex, asynchronous RS-232-C, CCITT V.24/V.28 point-to-point connections at 110, 150, 300, 1200, 2400, 4800, 9600 or 19200 bits per second (bps).

The 4-port interface has modem control capability with seven signals (TX, RS, signal ground, RTS, CTS, DCD, DTR).

Up to four interfaces can be installed in the same HP 2334A Plus, allowing up to sixteen connections. These connections can be made to workstations or hosts that are local or remote.

Auto-parity (odd, even) and auto-speed (up to 19200 bps) are available on each HP 2334A Plus port.

Binary transfer with no flow control can be done in blocks of maximum 128 bytes. For all HP applications (e.g., Advancelink, HPWord, AdvanceMail) using binary transfer, more details are given in the HP 2334A Plus Configuration Guide.

X-ON/X-OFF or ENQ/ACK flow control and HP block mode handshake methods can be chosen.

Synchronous Port Specifications

The HP 2334A has one RS-232-C, CCITT V.24/V.28 full-duplex, synchronous composite interface to connect it to a remote site via an:

- Analog leased line
- Digital leased line
- Dial-up line
- X.25 Packet Switching Network (Public or Private)

Hardwired connection for the synchronous composite link is not supported; synchronous, full-duplex, short-haul modems or a modem eliminator should be used.

The synchronous interface is supported at any speed between 1200 bps and 19200 bps with external clocking. Speeds of 4800, 9600 or 19200 bps can be selected with internal clocking (modems supported and recommended are described later in this datasheet).

X.3, X.28, X.29 Specifications

The HP 2334A Plus follows the CCITT X.3/X.28/X.29 recommendations (November 1980) which allow it to act as a private Packet Assember/Disassembler (PAD). The standard 18 X.3 parameters are supported and additional HP-defined local parameters are available for enhanced functionality with HP devices. These local HP parameters are:

- Parity
- Local block mode control
- Compatibility with ATP/ADCC (cluster only)
- Block mode terminator (forward packet)
- Data compaction
- Error message to host
- Break at application level
- Information message
- Asynchronous modem signal flexible control
- Auto-speed and auto-parity
- Block mode buffer size
- Asynchronous modem signal timer
- HP hand-check support
- HP 2334A Plus byte count
- Define remote profile

X.25 Specifications

The HP 2334A Plus has an X.25 interface which is fully compatible with the November 1980 version of the CCITT X.25 recommendation :

Level 1: Physical layer

- X.21 bis, RS-232-C, CCITT V.24/V.28 (up to 19200 bps).

Level 2: Data link layer

- LAP-B protocol.
- Modulo eight sequence number.
- Window size (1-7).
- Operates as DCE or as DTE.

Level 3: Network layer

- Switched Virtual Circuit (SVC) or Permanent Virtual Circuit (PVC).
- Up to seventeen virtual circuits simultaneously.
- Window size (1-7).
- Packet size (128 bytes).
- Supports D, M and Q bits.

X.25 and other supported facilities:

- Window size negotiation.
- Incoming calls barred.
- Outgoing calls barred.
- One way outgoing SVC.
- One way incoming SVC.
- Closed user group.
- Closed user group outgoing.
- Closed user group incoming.
- Closed group incoming barred.
- Closed group outgoing barred.
- Bilateral closed user group.
- Bilateral closed outgoing.
- Reverse charging request.
- Reverse charging acceptance.
- Packet size negotiation.
- Throughput class negotiation.

Additional Functionalities

- Automatic connection to pre-configured computer port.
- Symbolic host computer addressing, up to sixteen names.
- Up to three groups of pool ports can be defined.
- Test port for configuration and diagnostics (password protected) accessible from any terminal connected on the HP 2334A Plus.
- Single port reset command and HP 2334A Plus reset command available.
- User-defined welcome message with maximum twenty characters and user-defined PAD message header with ten characters.
- Default configuration for statistical multiplexer configuration.
- Local User Group definition.
- Host computer port contention.
- Data compaction.

Supported and Recommended Products

(Refer to point "Supported Computer Interfaces" for more details on cluster controller configuration).

Supported Workstations

Terminals

HP 2382A
HP 2392A
HP 2393A
HP 2394A
HP 2397A
HP 2621B
HP 2622A
HP 2623A
HP 2624B
HP 2625A
HP 2626A
HP 2626W
HP 2627A
HP 2628A
HP 2645A
HP 2648A
HP 3081A

PC's

HP 150A
HP 150B
HP 150II
HP 110
HP 110 Plus
HP Vectra PC

Printers*

HP 2563A
HP 2564A
HP 2565A
HP 2566A
HP 2602A
HP 2603A
HP 2631B
HP 2686A
HP 2686A Plus
HP 2932A
HP 2933A
HP 2934A
HP 2687A

Plotters

HP 7440A
HP 7450A
HP 7470A
HP 7475A

* The ATP Term Type 26 is available with MIT UB delta 1 and can be used with the HP 2334A Plus. The term type 18 is available for the ADCC. The maximum combined throughput for the printers attached to one HP 2334A Plus is 600 lines per minute.

Supported Applications

PC Applications

Advancelink (150)
Advancelink 2392
AdvanceMail (150)
AdvanceMail (Vectra)

HP 1000 Applications

Forms 1000	4.1	QDM	4.0

HP 3000 Applications

HPDESK	A.03.00	DICTIONARY	A.02.00
HPDRAW	A.03.00	DSG/3000	A.04.00
HPEASYCHART	A.03.00	HPFA	A.02.02
IDS	A.01.00	IFS	A.02.01
IMF	A.51.00	HPLIST	A.00.02
HPMENU	A.01.00	MM	A.07.06
HPMNT	A.00.01	PM	A.03.01
RAPID	A.01.09	HPSLATE	A.04.03
HPTELEX	A.02.00	TDP	A.03.02
HPTOOLSET	A.01.03	VISICALC	A.00.02
VPLUS	B.03.25	HPWORD	A.05.04
HPACCESS	In progress		

HP 9000 Applications

uucp with HP-UX 5.0 or later on Series 300/500
uucp with HP-UX 1.0 or later on Series 800

Supported Computer Interfaces

- HP 1000: MUX, PSI
- HP 3000: ATP, ADCC, INP
- HP 9000: MUX

The HP 2334A Plus in cluster controller configuration is supported when connected to an HP 3000 computer used with MPE-VE, MPE-VR or a later release and the X.25 link/3000 (HP 32187A). For products supported in cluster controller configuration please refer to the datasheet of Product Number HP 32187A.

The HP 2334A Plus in cluster controller configuration is supported when connected to an HP 1000 computer used with RTE-A (A-Series) or RTE-6/VM (E/F-Series). The X.25 interface on the HP 1000 is supported with LAP-B interface card (HP 12075A/12250A) and HP 1000 X.25 software (HP 91751A/R). For products supported in cluster controller configuration please refer to the datasheet of Product Number 91751A.

The HP 9000 computer is only supported in statistical multiplexer configuration with HP 27128A, HP 27140A, HP 98626A, HP 98628A, HP 98644A interface.

Non-HP computers may be connected to the HP 2334A Plus in cluster controller configuration via an X.25 interface supporting X.29 which complies with the 1980 CCITT recommendations.

Verified modems

ASYNC modems	SYNC modems
HP 37212A	Codex 2620, 2640, 2680
HP 35141A	Racal Milgo Alpha 96
HP 92205J	Bell 2096A
Racal Milgo MPS 1222	

Certified X.25 Packet Switching Networks

North America

Canada:	Datapac†
U.S.:	Infonet, Telenet, Tymnet, DDN

Europe

Austria	Datex-P
Belgium	DCS†
Denmark	Datapak
Finland	Datapak
France	Transpac
Germany F.R.	Datex-P
Ireland	Eirpac
Italy	Itapac†
Luxemburg	Luxpac*
Netherlands	Datanet1†
Norway	Datapak
Spain	Iberpac
Sweden	Datapak
Switzerland	Telepac
United Kingdom	PSS

Other Countries

Australia	Austpac
Brasil	Renpac
Hong Kong	Datapak†, Intelpak
Israel	Isranet
Japan	Venus-P
Malaysia	Maypac
Singapore	Telepac†
South Africa	Saponet-P*
South Korea	Datacom-net
Taiwan	Pacnet

Verified X.25 Switching Equipment

- M/A-Com CP9000 series II
- Dynapac model 8 (U.S. only)

Customer Installation and Configuration

The customer is responsible for the installation of the HP 2334A Plus. The HP 2334A Plus Reference Manual and Configuration Guide are needed to install and configure the HP 2334A Plus (see references under "Documentation").

The initial configuration may be set up by a Hewlett-Packard Customer Engineer or System Engineer by ordering the HP 2334A Plus Option 100. Before installing the HP 2334A Plus, the customer should obtain, install and verify the correct operation of any communication line, X.25 PSN access or any other equipment and facilities necessary to interface to the HP 2334A Plus.

Hewlett-Packard is liable only for the correct execution of the loop-back and self-diagnostic tests. All hardware and software connections to the communication line, the X.25 PSN and non-HP computers are the customer's responsibility.

The add-on HP 40261A four-port interface or any upgrade kit may be installed by the customer or by any Hewlett-Packard Customer Engineer on a time and materials basis.

Environmental Characteristics

Temperature (free space):	
Operating:	0° to +55°C
Non-operating:	−40° to + 75°C
Relative Humidity (non-condensing):	
Operating:	5% to 95% at 40°C
Non-operating:	90% at 65°C to 95% at 40°C
Maximum Altitude:	
Operating:	4,600 meters (14,700 ft)
Non-operating:	15,300 meters (49,000 ft)
Vibration:	Up to 0.38 mm ptp, 5-55-5 Hz, 3 axis for 15 minutes
Dwell at resonance:	10 minutes
Shock:	
Bench handling:	102 mm bench drop
Transport handling:	762 mm drop 30 g for 11ms

* The ATP Term Type 26 is available with MIT UB delta 1 and can be used with the HP 2334A Plus. The term type 18 is available for the ADCC. The maximum combined throughput for the printers attached to one HP 2334A Plus is 600 lines per minute.

† The product has received a temporary approval.

Physical Characteristics

Size: 135 mm H (140 mm with feet) x 425 mm W x 540 mm L [5.25 in. H (5.45 in. with feet) x 16.75 in. W x 21.25 in. L].

Net Weight:	13 Kg (29 lbs).
Shipping Weight:	17 Kg (38 lbs).

Electrical Specifications

Input Voltage:	86 to 127 volts; With Opt. 015, 195 to 253 volts.
Input Frequency:	47.5 to 66 Hz.
Power Consumption:	115 watts typical.

Approvals

RFI (Radio Frequency Interference):

- VDE 0871 level A and FCC class A
- Configurations including peripherals with high RFI levels may not be supported or may require on-site verification in some countries.

Safety:

- UL 478, UL 114 for EDP and office equipment
- CSA C22.2-154 for EDP equipment
- Compliance with international standard IEC380, IEC435

Data Communication Approvals:

- Complies with CCITT requirements
- Datacomm certification approval in Australia, Belgium, UK, Germany, Japan and Scandinavia.

Some datacomm regulations may restrict the use of all possible HP 2334A Plus connections. Check with your local datacomm regulation agency.

Ordering Information

Product	Description
HP 2334A	HP 2334A Plus - X.25 multiplexer
Opt 123	4 modem connect ports
Opt 015	220 V
Opt 100	Initial configuration
HP 40261A	Additional 4 modem connect ports
Opt 001	Upgrade 4 direct connect ports to modem connect ports HP 2334A Plus.
HP 40220A	Cable between HP 2334A Plus and ADCC printer ports
HP 40221A[1]	Cable between HP 2334A Plus and ATP ports or ADCC terminal ports.
HP 40223B	Upgrade kit from HP 2333A to HP 2334A Plus
Opt 001[2]	Power supply change
Opt 020	RS-232-C interface card for HP 2333A installed with Data Link
Opt 122	Upgrade 4 direct connect ports to 4 modem connect ports HP 2334A Plus
HP 40224B	Upgrade kit from HP 2334A to HP 2334A Plus
Opt 023	Upgrade 4 direct connect ports to 4 modem connect ports HP 2334A Plus.
Opt 123	Upgrade 4 modem connect ports to 4 modem connect ports HP 2334A Plus

Documentation

Part Number	Description
02334-90013	HP 2334A Plus Reference Manual
02334-90018	HP 2334A Plus Configuration Guide
5958-3402	X.25, The PSN Connection
5957-4635	Networking with X.25
5953-5984	X.25 Wide Area Networking Products
5954-7468	HP 3000 Networking Products

[1] Cable between HP 2334A Plus and ATP printer ports using term type 18 is HP 40220A.

[2] Order must include serial number of HP 2333A power supply, under "special instructions."

WARRANTY INFORMATION

In the U.S.A. and Canada, your Series 800 product is warranted by Hewlett-Packard against defects in materials and workmanship.* The standard warranty for a Series 800 product applies for ninety (90) days from date of shipment. Services provided under this warranty are identical to those provided in Hewlett-Packard's "Standard System Maintenance Service." Contact your local HP Sales and Service Office for the specific warranty terms applicable to your Series 800 product.

Hewlett-Packard will, at its option, repair or replace equipment which proves to be defective during the warranty period. Repairs necessitated by misuse of the equipment, or by use of hardware, software, or interfacing not provided by Hewlett-Packard are not covered by this warranty. HP warrants that the software and firmware designated by HP for use with a CPU will execute its programming instructions when properly installed on that CPU. HP does not warrant the operations of the CPU, software or firmware to be uninterrupted or error free.

NO OTHER WARRANTIES ARE EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. HEWLETT-PACKARD IS NOT LIABLE FOR CONSEQUENTIAL DAMAGES.

*In other countries, contact your local Hewlett-Packard Sales and Service Office to determine warranty terms.