

Tru64 UNIX

Migrating from Tru64 UNIX to HP-UX Shells

May 2005

This white paper contains information about features or functionality that are available in Tru64 UNIX shells, but that are not available, or are implemented differently, on HP-UX. The information includes descriptions of the differences and, in many cases, recommended changes you may need to make to accommodate these differences. Because we assume that you are migrating from Tru64 UNIX to HP-UX, this paper does not discuss functionality specific to HP-UX.

Migrating from Tru64 UNIX to HP-UX Shells

executive summary

When migrating from Tru64 UNIX to HP-UX, you will notice several differences in the functionality available in selected shells, and in the values that shell arguments or variables expect. This white paper describes the differences and, in many cases, recommends changes you may need to make to accommodate these differences. The paper discusses the following shells:

- `/usr/bin/sh` — This is the Bourne shell on Tru64 UNIX, and is commonly used within scripts. On HP-UX, `/bin/sh` is the POSIX shell.
- `/usr/bin/csh` — This is the C shell on both platforms.
- `/usr/bin/ksh` — This is the Korn 88 shell on both platforms. For Tru64 UNIX, the Korn shell is also identical to the POSIX shell (`/usr/bin/posix/sh`). On HP-UX, there are some differences between the Korn and POSIX shells.

Although other shells are available on both platforms, this paper does not discuss them. A majority of Tru64 UNIX customers use one or more of the previously listed shells for scripts or for interactive work, so the highest priority is to provide information that will help those customers migrate from these shells to the HP-UX versions.

This paper does not cover any clusters related support within the Tru64 UNIX shells because TruCluster is not currently supported on HP-UX. When that support is added, this paper will be updated.

This paper is one of a series of papers in the Tru64 UNIX application migration series.

general differences for all shells

There are a number of functional differences between Tru64 UNIX and HP-UX that you need to consider when you migrate from the shells on one platform to the shells on the other. Some differences affect the content of common shell start-up scripts, while others may be noticeable when executing private shell scripts or when using the shells interactively. This section describes general differences that are not specific to any one type of shell.

Tru64 UNIX specific environment variables

The following environment variables are commonly used on Tru64 UNIX, but are not available on HP-UX:

USER

Identifies the current user. It is common for shell scripts to use this variable. For example, a `.login` or `.cshrc` shell script might contain a line similar to the following:

```
set mail=/usr/spool/mail/$USER
```

On Tru64 UNIX, the `login` command sets `USER` and `LOGNAME` to the same value. On HP-UX, `login` only sets `LOGNAME`, which also identifies the current user. You must change references from `USER` to `LOGNAME`. Using the previous example, you change the command to the following:

```
set mail=/usr/spool/mail/$LOGNAME
```

Note that seemingly unrelated commands might give unexpected results if you have not updated shell scripts or configuration files for use on HP-UX. For example, the following command does not return the expected output because the `.cshrc` file references the Tru64 UNIX environment variable `$USER`:

```
% which grep
USER: Undefined variable.
```

The problem might not be obvious from the error message. When you change the `$USER` references to `$LOGNAME`, the `which` command returns the expected results as follows:

```
% which grep
/usr/bin/grep
```

LOCPATH

Designates the location of locale-specific information (see `LOCPATH(5)` for more information). If you currently use `LOCPATH`, you can either include a full, non-standard path name in your value of `LANG` or other `LC_*` environment variables, or pass the full path name to the `setlocale()` calls.

path differences

A variety of shell environment variables are available for defining the location of information within either Tru64 UNIX or HP-UX. Among such environment variables are `PATH`, `MANPATH`, and `NLSPATH`.

The two platforms have different default values for each of these environment variables, and many users redefine or add to the values within

start-up shell scripts like `.login`, `.cshrc`, and `.profile`. In some cases, user-defined path additions will work without changes in HP-UX shells, but in others they will not.

For example, path changes might not be retained when you use the `su` command. On Tru64 UNIX, using `su` does not affect the value of `PATH`; whatever you defined in start-up scripts or interactively is retained. For example, suppose user `foo` defines `PATH` in the following manner:

```
/home/foo/bin:/usr/local/bin:/usr/bin:/tools/bin:/contrib/bin:/usr/bin/X11
```

This says to search in `foo`'s `bin` directory first, and then to search other system-wide directories. Using the `su` command to switch users does not affect this `PATH` on Tru64 UNIX. However, when using `su` on HP-UX, `PATH` changes to the following:

```
/usr/bin:/usr/sbin:/sbin
```

If you are accustomed to using non-default `PATH` values when using `su`, be aware that HP-UX shells do not retain your additions to `PATH`.

While `MANPATH` and `NLSPATH` retain user-defined values when changing to superuser, because of differences in the Tru64 UNIX and HP-UX directory structures, the location of commonly used pieces of functionality might be in new places.

The `MANPATH` environment variable defines the location of man pages on a platform. On Tru64 UNIX, the default definition is:

```
MANPATH=/usr/share/%L/man:/usr/share/man:/usr/dt/share/man:/usr/local/man
```

On HP-UX, the default definition is:

```
MANPATH=/usr/share/man/%L:/usr/contrib/man/%L:/usr/local/man/%L:/usr/share/man:/usr/contrib/man:/usr/local/man
```

For each definition, `%L` is the name of the current locale. Therefore, if you are running in a Japanese locale and translated Japanese man pages are installed on your system, `MANPATH` finds and displays the Japanese language pages. However, on Tru64 UNIX, `MANPATH` searches `/usr/share/%L/man` first; on HP-UX, it searches `/usr/share/man/%L` first.

The default definitions of `NLSPATH` on Tru64 UNIX and HP-UX are also different. `NLSPATH` defines the location of translated program messages. On Tru64 UNIX, the default definition is:

```
NLSPATH=/usr/lib/nls/msg/%L/%N
```

On HP-UX, the default definition is:

```
NLSPATH=/usr/lib/nls/msg/%L/%N.cat:/usr/lib/nls/%l/%t/%c/%N.cat
```

See `NLSPATH(3)` on Tru64 UNIX or `environ(5)` on HP-UX for descriptions of the variable portions of these values.

The last segment of the paths differs on each platform. On Tru64 UNIX, it is %N, which specifies the name of the message catalog to open. On HP-UX, it is %N.cat. Both platforms use message catalogs of the form %N.cat; for example, grep.cat, sort.cat, and date.cat. However, on Tru64 UNIX, the .cat extension is considered part of the %N parameter. On HP-UX, only the application name is considered part of %N, and the .cat extension is explicitly listed within NLSPATH.

If you set NLSPATH within a shell script or in your interactive shell, you must include the .cat extension when migrating to HP-UX.

umask differences

The shells include the built-in command `umask` for setting or displaying the file mode creation mask. See `chmod(1)` on Tru64 UNIX for information on mask values. The system default mask value is `022`. On Tru64 UNIX, the `login` command does not change this value; on HP-UX, `login` changes the mask to `0` for non-secured logins. If you depend on the Tru64 UNIX default value, you might need to add the `umask` command to shell scripts.

differences in values assigned to shell variables

The naming conventions for some shell variables differ from Tru64 UNIX to HP-UX. Common shell start-up scripts like `.login`, `.profile`, `.cshrc`, and `.kshrc` are examples of scripts that might contain references to Tru64 UNIX specific values, but you should check other shell scripts for such values. For example, a shell script may assign a value to one of the locale-specific environment variables (`LANG`, `LC_ALL`, `LC_CTYPE`, `LC_COLLATE`, `LC_MESSAGES`, `LC_MONETARY`, `LC_NUMERIC`, or `LC_TIME`) to initialize the linguistic and cultural conventions used within the process. For example, the following Tru64 UNIX assignments set `LANG` to a locale that uses Spanish (`es`) with Spain's cultural rules (`ES`), and uses the ISO 8859-1 codeset:

```
setenv LANG es_ES.ISO8859-1          /* C shell */
LANG=es_ES.ISO8859-1; export LANG    /* Bourne/Korn/POSIX shells */
```

On Tru64 UNIX, the naming conventions for locales are similar, but not identical, to what HP-UX uses. For the previous example, the value of `LANG` on HP-UX is as follows:

```
setenv LANG es_ES.iso88591          /* C shell */
LANG=es_ES.iso88591; export LANG    /* Korn/POSIX shells */
```

See the *Tru64 UNIX to HP-UX 11i Porting Guide* for mapping between Tru64 UNIX and HP-UX locale names.

inlib and rmlib built-ins

The Tru64 UNIX `inlib` and `rmlib` built-in commands for using shared libraries are not currently supported. However, some older scripts might still include these commands, so you do need to check for references to them.

References to these commands should have been replaced with newer syntax as described in the Tru64 UNIX man page `loader(5)`. If you have not made these changes, you must do so when moving to HP-UX shells because they do not support `inlib` or `rmlib`.

using `/usr/bin/sh`

Tru64 UNIX and HP-UX both supply a `/usr/bin/sh` shell. Despite the matching names, the shell types are quite different. On Tru64 UNIX, `/usr/bin/sh` is the Bourne shell, but on HP-UX, `/usr/bin/sh` is the POSIX shell. Beginning with the 11i Version 1.5 release, HP-UX does not provide a Bourne shell. However, Tru64 UNIX does provide a POSIX shell at `/usr/bin/posix/sh`. For detailed descriptions of these shells, see the following man pages:

- `sh(1b)` — Tru64 UNIX Bourne shell
- `sh(1p)` — Tru64 UNIX POSIX shell
- `sh-posix(1)` — HP-UX POSIX shell

It is very common to use `/usr/bin/sh` for shell scripts. Because this name has different meanings on the two platforms, scripts that work on Tru64 UNIX need changes in order to work correctly on HP-UX. One method to find functional mismatches is to change references in Tru64 UNIX shell scripts from the Bourne shell to the POSIX shell, and then test the script on Tru64 UNIX. For example, if you have the following line in a script:

```
#!/usr/bin/sh
```

You would change the reference to the following:

```
#!/usr/bin/posix/sh
```

Similarly, if you have `BIN_SH` set in `/etc/profile` or `.profile`, you would also change that reference. Note that this change is for testing purposes only. This can help you determine what, if anything, works for a Bourne shell script that does not work for a POSIX shell script. After making any needed changes, you should restore the original shell designation in the script before attempting to run it on HP-UX.

The following sections describe some of the functionality or syntax available within `/usr/bin/sh` on Tru64 UNIX that is not supported or is supported differently in `/usr/bin/sh` on HP-UX. For readability, all references to

Tru64 UNIX refer to the `sh` environment on Tru64 UNIX. Similarly, all references to HP-UX refer to the `sh` environment on HP-UX.

reserved file descriptors on HP-UX

On HP-UX, the POSIX shell reserves some file descriptors for internal use. For HP-UX Release 10.10 and later, file descriptors 24 through 30 are reserved. Applications that use these and fork a subshell should not depend upon them surviving in the subshell or its descendants.

Check any scripts you have written for use on Tru64 UNIX. If they depend on the listed file descriptors, you must change the scripts in order to work on HP-UX.

shell variable differences

Although the `sh` implementations on Tru64 UNIX and HP-UX support many of the same shell variables, the following variables are supported on Tru64 UNIX, but are not available on HP-UX:

MAILMSG

Contains a mail notification message. For example, the variable might be set as follows:

```
MAILMSG="You have new mail."
```

On HP-UX, you must changes references to this variable to `MAILPATH`, which allows you to define a mail notification message.

SHACCT

Contains the name of a file that you own. If the variable is set, `sh` writes an accounting record in the file for each shell script executed. You can then use accounting programs such as `acctcom` and `acctcms` to analyze the data collected.

On HP-UX, you must remove this variable from scripts. However, HP-UX does support the named accounting programs. See the HP-UX man pages `acctcom(1M)` and `acctcms(1M)` for information on collecting and analyzing shell script data.

TIMEOUT

Contains the number of minutes from the time the Tru64 UNIX `sh` displays its prompt until you enter a command. If you do not enter a command in the specified amount of time, the shell exits. The `sh` on HP-UX supports the similar shell variable `TMOUT`, but `TMOUT` contains the number of seconds, rather than minutes, after which the shell

terminates if no command is entered. If a Tru64 UNIX script contains the following line:

```
TIMEOUT=10
```

You must change this line to the following in order to maintain the same script functionality:

```
TMOUT=600
```

The 600-second limit in `TMOUT` is the same as the 10-minute limit in `TIMEOUT`.

built-in command differences

Although the `sh` implementations on Tru64 UNIX and HP-UX support many of the same built-in commands, the following commands are supported differently on Tru64 UNIX and HP-UX:

`hash`

When using this built-in without options on Tru64 UNIX, the shell displays information about remembered commands, including the number of times a command has been run by the shell, and a measure of the work required to locate a command in the search command. For example:

```
$ hash
hits  cost  command
5     3     /usr/bin/ls
2     3     /usr/bin/man
1     3     /usr/bin/printf
```

On HP-UX, `hash` only returns a remembered list of utilities.

`return`

On Tru64 UNIX, `return` can only be executed within a shell function; if you do not do so, it returns an error. On HP-UX, if you invoke `return` when you are not in a shell function or in a `.` script, it has the same effect as an `exit` command.

`set -x`

Using `set` with the `-x` option displays commands and their arguments as they are executed. On Tru64 UNIX, the `-x` option is global; command names are printed whether they are in the main body or within a function. On HP-UX, however, command names are not printed when they occur within a function.

ulimit

On Tru64 UNIX, `ulimit` can take the `-h` option, which sets or displays the current hard resource setting. For example:

```
$ ulimit -h
cputime      unlimited
filesize    unlimited
datasize    1048576 kbytes
stacksize   32768 kbytes
:
```

The `-h` option is not supported on HP-UX. You must change references to the `-h` option to the `-H` option in scripts.

On Tru64 UNIX, you must be superuser to use the `-H` option; on HP-UX, any user can use the `-H` option. Therefore, if you have scripts that use `-H` so that only superuser can perform a given function, be aware that this restriction will no longer be enforced when using `-H` on the HP-UX version of `ulimit`.

pipe character differences

In the past, the circumflex (^) was supported as a pipe character within `sh`. Most modern shell scripts use the vertical line (|) as the pipe character, but Tru64 UNIX continues to allow ^ for this purpose. However, HP-UX does not.

Suppose you have a shell script or alias that searches directories for the number of files with read and execute permissions. Using pipes, the command might look like one of the following:

```
ls -l ^ grep r-x ^ wc -l
ls -l | grep r-x | wc -l
```

The first example uses the historical pipe character ^, while the second uses the modern character | (vertical line). Check your scripts and aliases for the first usage. If you have any that use the circumflex, you must change them use to the vertical line. This enables them to run in `sh` on HP-UX.

command substitution syntax differences

Tru64 UNIX supports the Bourne shell syntax of enclosing arguments within grave accents (` `) for command substitution. The shell first executes the command or commands within the grave accents, and then replaces the whole expression with their output. This feature is often used in assignment statements:

```
today=`date`
```

This statement assigns the string representing the current date to the `today` variable. You might then use this variable in a login script as follows:

```
echo "Good morning. Today's date is $today"
```

HP-UX does support using grave accents for command substitution, but the preferred syntax is to enclose a command in parentheses preceded by a dollar sign, `$(...)`; for example:

```
today=$(date)
```

You should replace uses of grave accents for command substitution with the dollar sign and parentheses syntax.

the C shell

Both Tru64 UNIX and HP-UX include implementations of the widely used C shell (`/usr/bin/csh`). The C shell is often a user's interactive shell, and is less commonly used for scripts, but it does see use in shell scripts as well. This section describes functionality present in the Tru64 UNIX C shell that either is not available on the HP-UX version, or is implemented in a different way.

For readability, all references to Tru64 UNIX refer to the `csh` environment on Tru64 UNIX. Similarly, all references to HP-UX refer to the `csh` environment on HP-UX.

command line editing

The Tru64 UNIX C shell provides full command-line editing support which allows you to recall and edit commands as if you were using an editor. This capability is in addition to the history mechanism. You must set the shell variable `editmode` to use this capability. See the `csh(1)` man page for details.

HP-UX's C shell currently does not include support for command line editing. This functionality is planned for a future release, but is not available as of HP-UX 11i v1.6. If you have been using the Tru64 UNIX C shell as your interactive shell, and are accustomed to the command-line editing support, the HP-UX POSIX shell (`/usr/bin/sh`) does include command-line editing, so you may choose to use it instead. Another alternative is to use the history and file name completion features within the HP-UX C shell to get a subset of the command line editing capabilities.

predefined shell variable differences

Although the Tru64 UNIX and HP-UX C shells support many of the same predefined shell variables, the following variables are supported on Tru64 UNIX, but are not available on HP-UX:

filec

Both C shells support file name completion, but it is always enabled on HP-UX. Therefore, you do not need to set a shell variable to get this functionality. In contrast, on Tru64 UNIX, you must set the shell variable `filec` to enable file name completion; for example:

```
% set filec=1
```

On HP-UX, remove this variable from shell scripts or configuration files.

histchars

This variable allows you to specify your own selected characters to use within history substitutions. The default characters are exclamation point (!) as the history substitution character, and circumflex (^) as the quick substitution character, but when you set `histchars`, you can define other characters.

On HP-UX, you must remove references to `histchars` in shell scripts and configuration files. If you have used non-default characters in history substitutions when using `csh` as your interactive shell, you must now use the default characters (! and ^) on HP-UX.

built-in command differences

Although the C shell implementations on Tru64 UNIX and HP-UX support many of the same built-in commands, the following commands are supported on Tru64 UNIX, but are not available on HP-UX:

`unlimit [-h] [resource]`

Removes the limitation on the named resource. Such limitations are set using the `limit` built-in command. If no resource is specified, all resource limitations are removed. When used with `-h`, corresponding hard limits are removed. Only the superuser can use the `-h` option.

`which [-U] name ...`

Takes a list of names and looks for the files that would be executed had these names been given as commands. Each name argument is expanded if it is aliased and searched for along your path. Both the aliases and path are taken from your `.cshrc` file. If you do not have a `.cshrc` file, the path is the current `PATH` from your environment.

The Tru64 UNIX `which` supports aliases, as follows:

```
% alias ls 'ls -F'
% which ls
alias/ls 'ls -f'
```

The HP-UX which does not, as follows:

```
% alias ls 'ls -F'  
% which ls  
/usr/bin/ls
```

The following command is supported differently on Tru64 UNIX and HP-UX:

time

Both C shells support the `time` built-in, but the command's output differs across the two platforms. On Tru64 UNIX, the output might look as follows:

```
% time  
15.86u 3.35s 2:11:20 0% 100+127k 2031+1170io 1683pf+0w
```

On HP-UX, the output might look as follows:

```
% time  
1.7u 0.7s 1:33:10 0%
```

The information returned in the first four fields of the Tru64 UNIX `time` built-in is the same as the four fields that HP-UX returns. Tru64 UNIX also returns additional information. See the description of `time` in the section "Predefined and Environment Variables" in the Tru64 UNIX `csch(1)` man page for information about all the time fields. If you are using any of the information returned in Tru64 UNIX specific fields, you must modify your scripts to remove these dependencies.

setenv behavioral difference

The built-in command `setenv` lets you set the value of a named environment variable. For example, the following command line assigns the value `localprinter` to the `PRINTER` environment variable.

```
% setenv PRINTER localprinter
```

If you omit the value of the named environment variable, the Tru64 UNIX C shell sets the value to null, as follows:

```
% setenv FOO  
% printenv  
:  
:  
FOO=  
%
```

In the HP-UX C shell, omitting the variable value is an error, as follows:

```
% setenv FOO  
setenv: Too few arguments.
```

-b command line option

Tru64 UNIX's C shell includes the `-b` option for increased security. The option forces a break from option processing, causing any further shell arguments to be treated as nooption arguments. All set-user-ID scripts must include this option on Tru64 UNIX.

The HP-UX C shell does not include the `-b` option.

internationalization support differences

The Tru64 UNIX `csh` includes more internationalization support than does the HP-UX `csh`. Both C shells, however, support the use of internationalized character classes in commands. The following command returns all file names that begin with a lowercase letter as defined by the current locale on both platforms:

```
% ls [[:lower:]]*
```

However, Tru64 UNIX also supports internationalized equivalence classes and range expressions, while the HP-UX `csh` does not.

The following example assumes that you have a directory with the following files:

```
13          Alpha      cotest      estim        t  
alpha        tre          chile      ESTIM      eur  ka
```

If you are running in a French locale and want to list all files that begin with a form of the letter `e`, using equivalence class syntax on Tru64 UNIX returns the following:

```
% ls [[:e=]]*
estim    ESTIM      t      eur  ka
```

HP-UX does not support equivalence class matching within the C shell, so given the same circumstances as the previous example, HP-UX returns the following:

```
% ls [[:e=]]*
Aucune concordance.
```

"Aucune concordance" means "No match" in French.

In addition to the difference in equivalence class support, the behavior of range checking for multi-character collating symbols differs between the shells. When running under Spanish locales given the same directory of files, Tru64 UNIX returns the following:

```
% ls [  -c]*
  tre    cotest
```

HP-UX returns the following:

```
% ls [â-c]*
âtre  cotest  chile
```

In traditional Spanish, "ch" is a multi-character collating symbol that sorts after "c", so a range that ends with "c" should not match filenames that begin with "ch".

There is also a difference in the behavior of range expressions that include multi-character collating symbols. Again, using the Spanish locale, Tru64 UNIX returns the following:

```
% ls [a-[.ch.]]*
alpha  Alpha  âtre  cotest  chile
```

HP-UX returns the following:

```
% ls [a-[.ch.]]*
No se encuentra correspondencia.
```

Note: The differences with respect to multi-character collating symbols exist within the C shells only. Both platforms support multi-character collating symbols within `grep`, and return identical results for when using `grep` or other regular expression-based commands.

command line expression differences

Within command line expressions in the Tru64 UNIX C shell, `l` is supported as a primitive file inquiry operand to identify symbolic links. The `l` primitive is not supported in the HP-UX C shell.

the Korn shell

The Korn shell (`/usr/bin/ksh`) is an interactive command interpreter and command programming language. On Tru64 UNIX, `ksh` is identical to the POSIX shell (`/usr/bin/posix/sh`). The man pages for these shells are `ksh(1)` and `sh(1p)`.

On HP-UX, the Korn shell (`/usr/bin/ksh`) is similar, but not identical to the POSIX shell (`/usr/bin/sh`). The man pages for these shells are `ksh(1)` and `sh-posix(1)`. There are more similarities between the Tru64 UNIX Korn/POSIX shell and the HP-UX POSIX shell than with the HP-UX Korn shell. When porting `ksh` scripts to HP-UX, it might be easier to port them to the HP-UX POSIX shell.

The "using `/usr/bin/sh`" section describes some aspects of the HP-UX POSIX shell. This section only covers differences between the Tru64 UNIX Korn and POSIX shells, and the HP-UX Korn shell.

For readability, all references to Tru64 UNIX refer to the `ksh` environment on Tru64 UNIX. Similarly, all references to HP-UX refer to the `ksh` environment on HP-UX.

environment variable differences

On Tru64 UNIX, the `COLUMNS` environment variable is not automatically set. You can assign a value to it if you prefer. However, on HP-UX, the Korn shell automatically sets a value for `COLUMNS`. In a windowed environment, the value automatically changes when the shell detects that the window size has changed.

special ksh command differences

While both the Tru64 UNIX and HP-UX Korn shells support many of the same special commands, Tru64 UNIX includes additional options with the following commands:

`export`

Tru64 UNIX supports the `-p` option that outputs the names and values of all exported variables, one per line, in the following format:

```
export -variable=-value
```

`readonly`

Tru64 UNIX supports the `-p` option that outputs the names and values of all read-only variables, one per line, in the following format:

```
readonly -variable=-value
```

`set`

Tru64 UNIX supports two options for this command that are not available on HP-UX. They are `-b`, which causes the shell to notify the user asynchronously of background job completions, and `-C`, which prevents existing files from being overwritten by the shell's redirection operator (`>`).

If you use `set` with either of these options, you must change your command or script. For `-C`, you can use `noclobber` instead. For example:

```
$ set -o noclobber    # same as set -C
$ set +o noclobber   # same as set +C
```

typeset

Tru64 UNIX supports the `-H` option, which provides system-to-hostname file mapping on machines that restrict the set of characters in file names. This is not supported in `ksh` on HP-UX.

ulimit

Tru64 UNIX supports the following additional options for setting or displaying resource limits:

- `-n` The number of file descriptors
- `-v` The number of kilobytes for virtual memory

unask

Tru64 UNIX supports the `-S` option, which produces Symbolic output.

unalias

Tru64 UNIX supports the `-a` option, which removes all aliases from the current shell execution environment.

unset

Tru64 UNIX supports the `-v` option for designating that named values refer to variables, not functions.

conditional expressions difference

Tru64 UNIX supports the `-e` option, which tests if a file exists; HP-UX does not. Instead, use the `-a` option within such expressions, or use the HP-UX POSIX shell, which supports the `-e` option.

negation command support

Tru64 UNIX supports the negation command (`!`); HP-UX does not. For example, the following command line is successful on Tru64 UNIX:

```
$ if ! grep -q foo /etc/passwd; then echo no foo; fi
no foo
```

On HP-UX, the same command line is not successful, as follows:

```
$ if ! grep -q foo /etc/passwd; then echo no foo; fi
ksh: !: not found
```

The HP-UX POSIX shell supports the negation command. Therefore, you might want to use that shell instead.

reserved file descriptors on HP-UX

On HP-UX, the Korn shell reserves file descriptors 10 and 54 through 60 for internal use. Applications that use these and fork a subshell should not depend upon them surviving in the subshell or its descendants.

Check any scripts you have written for use on Tru64 UNIX. If they depend on the listed file descriptors, the scripts must change in order to work on the HP-UX `ksh`.

for more information

For general information about Tru64 UNIX shells, see the *Tru64 UNIX Command and Shell User's Guide* and the appropriate man pages, as listed previously. For general information about HP-UX shells, see *HP-UX Shells: User's Guide* and the man pages, as listed previously.

For more information on porting from Tru64 UNIX to HP-UX, see the following:

- *Tru64 UNIX to HP-UX 11i Porting Guide* at the following location:
http://h30097.www3.hp.com/transition/apps/porting_guide.html
- Tru64 UNIX to HP-UX application transition Web site:
<http://www.hp.com/go/tru64appmigration>

For more information about HP-UX development tools, especially the C compiler and linker, go to the following location:

<http://www.docs.hp.com/hpux/dev/index.html>

Note

Be sure to check the operating system versions to which the manuals apply; otherwise, you might try to use an Itanium® architecture feature on PA-RISC, or vice versa.

For all HP-UX documentation, go to the following location:

<http://www.docs.hp.com>

The following HP-UX documents might be useful:

- *Shells: User's Guide*
- *Technical Addendum to the Shells: User's Guide*
- *HP-UX 9.x - 11i Version 1.6 Internationalization Features White Paper*

The HP IT Resource Center at <http://www.itrc.hp.com> is useful for searches. From the main page, click on search, enter the search text and check the ITRC forums option.

Legal notices

© Copyright 2001, 2003, 2005 Hewlett-Packard Development Company, L.P.

Intel®, Itanium®, Itanium® 2, and Itanium®-compatible are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

POSIX® is a registered trademark of the Institute of Electrical and Electronic Engineers, Inc.

UNIX® is a trademark of The Open Group in the U.S. and/or other countries.

All other product names mentioned herein may be trademarks of their respective owners.

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

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.