hp-ux web server suite

january 2003

migration guide

Migration Guide

HP Apache-based Web Server Version 1.3.x to HP-UX Web Server Suite

January 16, 2003

Table of Contents

Glossary

ASF	Apache Software Foundation
CA	Certificate Authority
hpuxws	HP-UX Web Server Suite
HP Apache	HP Apache-based Web Server
IPF	Itanium Processor Family
IPv6	Internet Protocol Version 6
JDK	Java Development Kit
JRE	Java Runtime Environment
LDAP	Lightweight Directory Access Protocol
MPM	Multi-Processing Module
PA-RISC	Precision Architecture, Reduced Instruction Set Computing
PHP	PHP Hypertext Preprocessor
RSA	RSA Security Inc.
WebDAV	Web-based Distributed Authoring and Versioning

Revision History

January 16, 2003	Version 1	

Legal Notices

The information in this document is subject to change without notice.

Warranty Disclaimer. HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS INFORMATION, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance
or use of this material.

Restricted Rights Legend. Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies. Rights for non-DOD U.S. Government Department and Agencies are as set forth in FAR 52.227-19 (c)(1,2).

Acknowledgements. This product includes software developed by the Apache Software Foundation
This documentation is based on information from the Apache Software Foundation (http://www.apache.org).

This product includes software developed by the OpenLDAP Project (http://www.openldap.org).

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product includes software developed by the Java Apache Project for use in the Apache JServ Servlet Engine (http://java.apache.org/) .

This product includes software developed by Ralf S. Engelschall (rse@engelschall.com) for use in the mod_ssl project (http://www.modssl.org/).

This product includes software developed by Ralf S. Engelschall (rse@engelschall.com).

This product includes PHP, freely available from (http://www.php.net).

1 Using this Guide

This guide covers the migration from HP Apache-based Web Server Version 1.3.x to HP-UX Web Server Suite on servers running HP-UX 11.0 (PA-RISC), 11i (PA-RISC) and HP-UX 11i Version 1.5 or later, Itanium Processor Family (IPF).

This guide can be used to:

- Understand the differences between HP Apache-based Web Server Version 1.3.x and HP-UX Web Server Suite .
- Install the HP-UX Web Server Suite .
- Migrate the configuration of your HP Apache-based Web Server from Version 1.3.x to HP-UX Web Server Suite .
- Verify that your HP-UX Web Server Suite migration was successful.

There are three basic sections to the guide.

- **Section 2:** Quick Guide to Migration Solutions
  For many users this will be the only section required.
- **Sections 3-7:** Detailed Installation and Migration Steps
  For users who want a more detailed discussion of a particular step.
- **Appendices:** A complete listing of module changes
  For users who want an exhaustive overview of all changes to modules, directives, and options.

2 Quick Guide to Migration Solutions

Four alternative solutions are presented. Solutions A & B are simple installations that require little knowledge of Apache. If you have heavily customized any part of Apache you should refer to the detailed sections later in the document. Solutions C & D are more complex and assume a familiarity with Apache's configuration files.

2.1 Quick Guide Assumptions

A. 1.3.x refers to HP Apache-based Web Server 1.3.x.
   hpuxws refers to HP-UX Web Server Suite.
B. Files are in their default locations. For example:
   1.3.x is installed in the default locations:
   `<apache root> = /opt/apache and <tomcat root> = /opt/tomcat`
   HP-UX Web Server Suite is installed in the default locations:
   `<apache root> = /opt/hpws/apache  <tomcat root> = /opt/hpws/tomcat`
   `<webmin root> = /opt/hpws/webmin and <xmltools root> = /opt/hpws/xmltools`
C. The Apache documents are available at `<apache root>/hpws_docs` and `/opt/hpws/hp_docs/apache`.
   The Tomcat documents are available at `<tomcat root>/hpws_docs` and `/opt/hpws/hp_docs/tomcat.` The XML Tools documents are available at `/opt/hpws/hp_docs/xmltools.` The Webmin documents are available at `/opt/hpws/hp_docs/webmin.`
   (Note: The file `/opt/hpws/util/altroot.sh` can be used to move Apache's root. Type "`altroot.sh -h`" for help with the script)
D. The customer does not need IPv6 support.

2.2 Preliminary Steps

2.2.1 HP-UX Web Server Suite Requirements

HP-UX 11.0, 11i, 11i version 1.5, or later.
90 MB of disk space without XML Tools installed. 190MB of disk space for all four components installed.
Perl 5.6.1
Java JDK 1.2.2.4 or later

2.2.2 HP-UX Web Server Suite Installation

A. Stop Apache. For example if you are using HP Apache 1.3.x. then as root type:
   `/opt/apache/bin/apachectl stop`
B. Verify that all httpd processes have stopped by typing:
   `ps -e | grep httpd`
C. Install using the HP SW Depot Web Release,
   software.hp.com/ →Featured Products→HP-UX Apache-based Web Server.
   OR software.hp.com/ search for "HP Apache-based Web Server".
   The latest versions are always available online from HP SW Depot.
   Or install using the HP-UX 11.0/11i Application Release CDs (when available).

2.3 Running 1.3.x and HP-UX Web Server Suite on the same machine

Solutions A & B are two simple alternative installations.

2.3.1 Solution A: Running on the SAME machine but at DIFFERENT times.

If you installed HP-UX Web Server Suite on a machine that has 1.3.x then you are done. No additional steps are required. HP-UX Web Server Suite and HP Apache 1.3.x can coexist on the same machine. However because they use the same port assignments, they cannot be run at the same time without conflicting with each other.

2.3.2 Solution B: Running on the SAME machine at the SAME time.

You need to change the port assignments so that the two Apaches don't conflict. The utility script `/opt/hpws/util/ports.sh` can be used to list ports and files for you to manually edit.
For example type:
`/opt/hpws/util/ports.sh`

You will see output similar to the following:

```
Apache installed at /opt/hpws/apache
Tomcat installed at /opt/hpws/tomcat
Webmin installed at /opt/hpws/webmin
Xmltools installed at /opt/hpws/xmltools
```

Default locations of configuration files:

```
Apache:       /opt/hpws/apache/conf/httpd.conf
Apache (SSL): /opt/hpws/apache/conf/ssl.conf
LDAP:         /opt/hpws/apache/conf/ldap.conf
Tomcat:       /opt/hpws/tomcat/conf/server.xml
mod_jk:       /opt/hpws/tomcat/jk/apache2/mod_jk.conf
mod_jk:       /opt/hpws/tomcat/jk/apache2/workers.properties
Webmin:       /opt/hpws/webmin/conf/miniserv.conf
```

The following ports have been identified:

```
HTTP port:          80
HTTPS port:         443
Tomcat ports:       8005 8081 8009
Webmin port:        10000
LDAP port:          389
```

Then go to each file and manually change the port numbers to for example:

```
HTTP port:          8080
HTTPS port:         8443   (note 1)
Tomcat ports:       8105,8181,8109
Webmin port:        10100
LDAP port:          389    (note 2)
```

Note 1. Your SSL URL will need to be of the form `https://yourserver.com:8443` since you are no longer using the standard SSL port.

Note 2. The LDAP port doesn't change since it is the LDAP server's port. HP Apache-based Web Server or HP-UX Apache-based Web Server is an LDAP client.

2.4 Modifying httpd.conf

Solutions C & D are two alternative migration methods based on opposite philosophies. Method C edits the new configuration files to correspond to your needs. Method D edits copies of your old files.

2.4.1 Solution C: Migrating to HP-UX Web Server Suite using the NEW HP-UX Web Server Suite configuration files as a base.

A. Save a copy of `/opt/hpws/apache/conf/httpd.conf`
B. Find the differences between the old and new configuration files. For example:
   `diff /opt/apache/conf/httpd.conf /opt/hpws/apache/conf/httpd.conf`
C. Edit `/opt/hpws/apache/conf/httpd.conf`
D. Skip to the "Directives Common to C & D Solutions" section below.

2.4.2 Solution D: Migrating to HP-UX Web Server Suite using your OLD 1.3.x configuration files as a base.

A. Save a copy of `/opt/hpws/apache/conf/httpd.conf`
B. Copy your 1.3 httpd.conf file to `/opt/hpws/apache/conf/httpd.conf`
C. Find the differences between the old and new configuration files.
D. Edit `/opt/hpws/apache/conf/httpd.conf`
E. `LoadModule`
   The module location has changed from the `libexec` to `modules` directory. So you need to replace `libexec` with `modules` in the "Dynamic Shared Object (DSO) Support" section.

F. `mod_dav`
This module enables "Web-based Distributed Authoring and Versioning" (webDAV)
See the FAQ at www.hp.com/products1/unix/webservers/apache/faqs/index.html for more details.

G. `mod_perl`
Enabling Perl has changed. In HP-UX Apache-based Web Server "`ModPerl::`" has replaced "`Apache::`".
The new way is:

```
<IfModule mod_perl.c>
  PerlModule ModPerl::Registry
<Files *.pl>
     SetHandler perl-script
     PerlHandler ModPerl::Registry::handler
     Options +ExecCGI
   PerlOptions +ParseHeaders
  </Files>
  </IfModule>
```

H. `mod_php`
Enabling PHP has changed. For example in 1.3.x you did:

```
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
```

But in HP-UX Apache-based Web Server you do:

```
<Files *.php>
 SetOutputFilter PHP
 SetInputFilter PHP
</Files>
```

In addition, to use `PATH_INFO`, you must explicitly set directive `AcceptPathInfo`, otherwise a 404 response will be returned.

I. `mod_proxy`
The method of invoking `mod_proxy` has changed radically.
See httpd.apache.org/docs-2.0/mod/mod_proxy.html.
For example in 1.3.x you do:

```
<Directory proxy: *>
    ...
</Directory >
```

But in HP-UX Apache-based Web Server you do:

```
<Proxy *>
    ...
</Proxy>
```

J. `mod_suexec`
To enable suexec you need to load the module with
`LoadModule suexec_module modules/mod_suexec.so`
And you need to setup the following directive:

```
<IfModule mod_suexec.c>
    SuexecUserGroup TheUser TheUsersGroup
</IfModule>
```

See httpd.apache.org/docs-2.0/mod/mod_suexec.html.

K. `mod_jk`
This module is the connector used by Apache to communicate with Tomcat servlet container. Mod_jk replaces the mod_jserv connector which is used by the Apache JServ servlet engine To enable this uncomment the line `#Include /opt/hpws/tomcat/jk/apache2/mod_jk.conf in /opt/hpws/apache/conf/httpd.conf`. See http://jakarta.apache.org/tomcat/tomcat-4.1-doc/jk2/jk/aphowto.html for more information.

L. `AddCharset`
You may wish to take advantage of the many new character sets added to HP-UX Apache-based Web Server. Previously there were 8 now there are 28. See httpd.apache.org/docs-2.0/mod/mod_mime.html#addcharset.

M. `AddHandler type-map var`
Previously this was commented out. It is now enabled by default to allow the Apache "It Worked" page (http://yourserver.com/index.html) to be distributed in multiple languages.

N. `ErrorDocument`
ASF documentation says that this directive must now have a closing quote. For example
`ErrorDocument 403 "Some Message"`
Currently if you leave off the closing quote no error is reported but this may change in the future.

O. `BrowserMatch`
You need to enable the following if you are going to use "Web-based Distributed Authoring and Versioning" (WebDAV) methods with Microsoft WebFolders.
```
BrowserMatch "Microsoft Data Access Internet Publishing Provider" redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
```
See `mod_dav` above.

P. `IndexOptions`
The new `VersionSort` option enables numerical sorting rather than alphabetical sorting.
`IndexOptions FancyIndexing VersionSort`

Q. `ReadmeName` and `HeaderName`
These directives determine the filename to be appended and prepended to the directory listing. The defaults have been change from `README` and `HEADER` to `README.html` and `HEADER.html`.

R. Continue with the "Directives Common to C & D Solutions" section below.

2.4.3 Directives Common to C & D Solutions

These apply to both Solutions C and D.

A. `ServerType`
The `ServerType` directive has been eliminated since HP-UX Web Server Suite can only be `ServerType standalone`.

B. `Port`
The `Listen` directive is now required and the `Port` directive has been eliminated. For example if you had set `Port 80` you should change it to `Listen 80`.

C. `ServerName`
The `ServerName` directive now supercedes the functionality of the `Port` directive. If no `ServerName` or associated port number is indicated then Apache will try to deduce them. For reliablity and predictability you should explicitly set the servername and port. For example:
`ServerName yourserver.com:80.`

D. Process handling directives.
See Migrating Process Handling.

♦`StartServers and ThreadsPerChild`
There is not much reason to change these values since the number of child processes is dynamically set based on load.

♦`MaxClients`
This is the maximum number of simultaneous client connections. Performance is based on transactions per second and is only indirectly related to `MaxClients` so you normally won't change this value.

♦`MaxRequestsPerChild`
This directive sets the limit on the number of requests that an individual child server process will handle. After `MaxRequestsPerChild` requests, the child process will die. If it is set to 0 then the process will never die. If you load modules that are not distributed by HP, you may have memory leaks. In that case you may want to set `MaxRequestsPerChild` to, for example, 500 so that the process will die and the memory will be freed. In order to maximize performance HP-UX Apache-based Web Server sets it to 0.

♦`MinSpareThreads and MaxSpareThreads`
Child processes are created until the number of idle threads are greater than `MinSpareThreads`. Child processes are killed until the number of idle threads are less than `MaxSpareThreads`. The defaults should work well with most servers. If you have extremely high loads then increase `MinSpareThreads` and `MaxSpareThreads`. This will not increase transactions per second but it will lower the impact of especially demanding transactions on other less demanding ones.

♦MinSpareServers and MaxSpareServers
These have been replaced with MinSpareThreads and MaxSpareThreads.

E. Modules not in HP-UX Apache-based Web Server distribution
Check that you have the same modules in LoadModule as you had in 1.3.x. 1.3.x modules will not work with HP-UX Web Server Suite and will need to be rewritten. See Writing 2.x Modules.

F. mod_auth_digest
mod_digest has been renamed to mod_auth_digest.

G. mod_proxy, mod_cache, mod_disk_cache, and mod_file_cache
mod_proxy has been decomposed into a more logical structure with separate modules. The cache functionality has been split out into other cache modules such as mod_cache, mod_disk_cache, and mod_file_cache. Generally they use similar cache directives as used in the old mod_proxy.

H. mod_cgi.
See Migrating Common Gateway Interface (CGI) and
httpd.apache.org/docs-2.0/mod/mod_cgid.html.
You can improve CGI performance by enabling the CGI daemon (cgid) module instead of cgi.
```
LoadModule cgid_module modules/mod_cgid.so
#LoadModule cgi_module modules/mod_cgi.so
Scriptsock <path>
```
You need to enable the Scriptsock directive which defines the UNIX socket for communicating with cgid.
```
Scriptsock            logs/cgisock
```

I. mod_define
mod_define has been eliminated. The Define directive allowed you to set a variable for later use in the configuration file. A future module may implement this functionality.

J. Logging directives.
See Migrating Logging
AgentLog, RefererLog, and RefererIgnore
The logging directives AgentLog, RefererLog, and RefererIgnore have been eliminated. They can be replaced with LogFormat and CustomLog directives.

K. mod_ssl
♦SSL configuration is in a separate configuration file called ssl.conf.
♦The default session cache is now:
```
SSLSessionCache         shmcb:logs/ssl_scache(512000)
```
♦ca-bundle.crt is not distributed with HP-UX Apache-based Web Server. You can use the certmig tool to extract ca certificates from a Netscape client. For example:
```
/opt/hpws/apache/util/test_certmig.sh -E -d $HOME/.netscape
```
See /opt/hpws/hp_docs/apache/utilities.user.guide.
♦ Logging directives have changed
SSLLog and SSLLogLevel have been eliminated. Use Apache logging directives ErrorLog and LogLevel instead.

2.5 Start

To start Apache, as root type :
```
/opt/hpws/apache/bin/apachectl start
```

2.6 Verify

In a browser enter the URL:
```
http://yourserver.com
```

3 Getting More Information

The HP Apache-based Web Server or HP-UX Web Server Suite is built on the open source Apache HTTP Server software developed by the Apache Software Foundation (ASF) with additional open source components and HP proprietary content integrated and bundled by HP.

The following table lists resources for HP Apache-based Web Server or HP-UX Web Server Suite. For the latest HP Apache-based Web Server or HP-UX Web Server Suite information, see www.hp.com/go/webserver.

HP Apache-based Web Server or HP-UX Web Server Suite Resources

Resources	Location
HP Apache-based Web Server or HP-UX Web Server Suite information library, technical tips, FAQs	www.hp.com/products1/unix/webservers/apache/index.html
HP Apache-based Web Server or HP-UX Web Server Suite product bundle	www.hp.com/go/webserver Click "downloads"
HP Apache or HP-UX web Server Suite product bundle for IPv6	software.hp.com/ search for "IPv6"
HP Developer & Solution Partner Portal	www.hp.com/ search for "HP Apache-based Web Server"

Integrated Applications

Resources	Location
HP-UX Workload Manager	www.hp.com/go/wlm
BEA Weblogic connector to Apache (mod_wl)	www.bea.com
BroadVision connector to Apache (mod_bv)	www.broadvision.com

The following table lists general information about Apache and the open source add-on products bundled by HP into the HP Apache-based Web Server or HP-UX Web Server Suite.

Apache Open Source Resources

Resources	Location
General	
Apache HTTP Web Server General Information	httpd.apache.org/
Apache Software Foundation (ASF)	www.apache.org/
Apache 1.3 User's Guide	httpd.apache.org/docs/
Apache 2 User's Guide	httpd.apache.org/docs-2.0/
Developer resources	dev.apache.org
IPv6	
IPv6 Home Page	www.ipv6.org
LDAP authentication	
OpenLDAP	www.openldap.org/
auth_ldap	www.rudedog.org/auth_ldap
Perl	
Perl Interpreter	www.software.hp.com/, search for "Perl v.5.6.1"

Resources	Location
General information on the mod_perl module	perl.apache.org/
PHP	
PHP User's Guide	www.php.net/docs.php
General information on PHP	www.php.net
Servlets	
Tomcat User's Guide	jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html
Apache JServ - Servlets Development Kit 2.0 for developing servlets. Later versions may work but they have not been tested. This is required by JServ in HP Apache 1.3.x	java.sun.com/products/servlet/archive.html
SSL	
OpenSSL User's Guide	www.openssl.org/docs/
mod_ssl User's Guide	www.modssl.org/docs
WebDAV	
webDAV Home Page	www.webdav.org/
Webmin	
Webmin User's Guide	www.swelltech.com/support/webminguide/index.html
Webmin Home Page	www.webmin.com/
XML Tools	
Cocoon Documents	xml.apache.org/cocoon/index.html
Xerces Documents	xml.apache.org/xerces2-j/index.html
Xalan Documents	xml.apache.org/xalan-j/index.html
FOP Documents	xml.apache.org/fop/index.html
Batik Documents	xml.apache.org/batik /index.html

4 What's New and Changed in HP-UX Web Server Suite

This version is the first release of a more flexible HP-UX Web Server Suite. Beginning with this version, HP-UX Apache-based Web Server, HP-UX Tomcat-based Servlet Engine, and HP-UX Webmin-based Admin can be installed together (as in previous versions of HP Apache-based Web Server) or installed separately as standalone components. A new feature, HP-UX XML Web Server Tools, has been added and can be installed separately or with other components. Each is installed in its own directory with a common base root of /opt/hpws/

This section contains an overview of the new and changed features in HP-UX Web Server Suite.

4.1 File Location Differences Between HP Apache 1.3.x and HP-UX Web Server Suite

HP Apache 1.3.x files are located in `/opt/apache`.
In HP-UX Web Server Suite these files are located in `/opt/hpws` under their respective directories.

The following table compares HP Apache 1.3.x and HP-UX Web Server Suite configuration files.

Configuration File Differences Between HP Apache 1.3.x and HP-UX Web Server Suite

Component	HP Apache 1.3.x	HP-UX Web Server Suite
Apache Server (httpd)	`/opt/apache/conf/httpd.conf`	`/opt/hpws/apache/conf/httpd.conf`
mod_ssl	Included inside of: `/opt/apache/conf/httpd.conf`	`/opt/hpws/apache/conf/ssl.conf` New in HP-UX Web Server Suite and included by httpd.conf
mod_file_cache	Not available	`/opt/hpws/apache/conf/cache.conf`
Tomcat	`/opt/tomcat/conf/server.xml`	`/opt/hpws/tomcat/conf/server.xml`
mod_jserv	`/opt/apache/conf/jserv/jserv.conf`	Replaced with `mod_jk`
mod_jk	1.3.26 and later: `/opt/tomcat/conf/jk/mod_jk.conf` and `/opt/tomcat/conf/jk/ workers.properties`	`/opt/hws/tomcat/jk/apache2/mod_jk.conf` and `/opt/hpws/tomcat/jk/apache2/ workers.properties`
Webmin	`/opt/apache/webmin/conf/miniserv.conf`	`/opt/hpws/webmin/conf/miniserv.conf`
auth_ldap	1.3.26.06 and later: `/opt/apache/conf/ldap.conf`	`/opt/hpws/apache/conf/ldap.conf`

Note: If you do not need the functionality then you can comment out the corresponding "include" directive. For example if you do not need ldap authentication then the line would look like:
`#Include /opt/hpws/apache/conf/ldap.conf`

4.2 Version and Module Numbers

Version Numbers of HP Apache and HP-UX Web Server Suite Components

Component	HP Apache 1.3.26 on 11.0/11i and Version 1.5 or later (PA-RISC & IPF)	HP-UX Web Server Suite on 11.0/11i and 11i Version 1.5 or later (PA-RISC & IPF)
Apache Web Server	1.3.26	2.0.43
OpenSSL	0.9.6g	0.9.6g
mod_ssl	2.8.10	A standard module of ASF Apache
BSAFE ® Crypto-C	PA 5.2/IPF 5.2.1	PA 5.2/IPF 5.2.1
Tomcat Servlet Container	3.3.1	4.1.12
Apache Connector to Tomcat	mod_jserv / mod_jk	mod_jk
Apache JServ	1.1.1	Not available. See preceding table row for equivalent.
mod_jk	1.2.0	1.2.0
mod_perl	1.27	1.99_07
PHP	4.2.2	4.2.3
auth_ldap	1.6	1.5.4
OpenLDAP SDK	2.0.7	2.0.7
Stunnel	3.14	3.14
Webmin	0.980	1.020
Xerces	Not available	2.2.1
Xalan	Not available	2.4.1
Batik	Not available	1.5
FOP	Not available	0.20.4
Cocoon	Not available	2.0.3

Note: The version numbers in the preceding table are correct for the HP Apache-based Web Server 1.3.26.06 and HP-UX Web Server Suite initial releases. For later releases see the Release Notes in the documentation directory, or on the Software Depot web site: software.hp.com/ → Featured Products → HP-UX Apache-based Web Server or www.hp.com/go/webserver→Click “downloads”

4.3 New Multi-Processing Module in HP-UX Apache-based Web Server

The Apache HTTP server 2.x can be built with one of several different multiprocessing modules (MPMs). These MPMs are beos, os2, perchild, prefork, winnt and worker. Apache 1.3.x is equivalent to "prefork". HP-UX Apache-based Web Server is built with the "worker" module. This module implements a hybrid, multiprocess, multithreaded server and provides high scalability with improved performance.

In the worker MPM a single control process is responsible for launching child processes. A fixed number of threads is created by each child process. This is specified in the `ThreadsPerChild` directive. Individual threads listen for connections and serve them when they arrive. A pool of spare, idle threads is created to be ready to handle these connections. The `MinSpareThreads` and `MaxSpareThreads` directives determine the range of this pool. Processes will be dynamically created until there are idle threads equal to at least `MinSpareThreads`. Processes will be destroyed until there are idle threads less than or equal to `MaxSpareThreads`. The `MaxRequestsPerChild` directive limits the number of requests that an individual child server process will handle. It controls how frequently the server recycles processes by killing old ones and launching new ones. If `MaxRequestsPerChild` is set to 0 then processes are never recycled.

For more information on the worker MPM, see httpd.apache.org/docs-2.0/mod/worker.html.

4.4 Summary of New Features in the HP-UX Web Server Suite

Following is a summary of the new features in HP-UX Web Server Suite.

Threading

HP-UX Apache-based Web Server runs in a hybrid multi-process, multi-threaded mode for improved scalability.

Multi-protocol Support

One of the new capabilities of Apache is support for multiple protocols. Users can write filters that implement ftp and other protocols. (`mod_echo` is provided as an example.)

Apache Portable Runtime (APR)

This new Apache API for modules has changed significantly for HP-UX Apache-based Web Server. In HP-UX Apache-based Web Server module ordering is done largely automatically. Module ordering is also done per-hook to allow more flexibility. Hooks allow modules to interact with Apache at many points in Apache's processing chain. New calls have been added that provide additional module capabilities without patching the core Apache server.

Filtering

Apache modules may now be written as filters that act on the stream of content as it is delivered to or from the server. For example, the output of CGI scripts can be parsed for Server-Side Include directives by `mod_include`.

IPv6 Support

On systems where IPv6 is supported by the underlying OS, Apache gets IPv6 listening sockets by default. Additionally, the `Listen`, `NameVirtualHost`, and `<VirtualHost>` directives support IPv6 numeric address strings (e.g., `"Listen [fe80::1]:8080"`).

Note:

a. HP-UX Web Server Suite for IPv6 is available on 11i (PA-RISC) as a separate HP–UX Web Server Suite product and requires an additional IPv6 networking product to be installed. HP plans to have IPv6 support integrated into 11i version 2.

b. Other HP-UX releases and IPF are not supported at this time.

WebDAV

WebDAV (“Web-based Distributed Authoring and Versioning”) is a set of extensions to the HTTP protocol that allows users to create, move, copy, and delete files (i.e. HTML, images, presentations) and directories on a remote server. HP-UX Apache-based Web Server implements WebDAV using the `mod_webdav` and `mod_webdav_fs` modules, www.webdav.org/. WebDAV is an IETF standard for collaborative authoring on the web. With HTTP, WebDAV can use strong authentication (certificates), encryption, proxy support, and caching.

PHP to Oracle Database Connectivity (PA-RISC only)

PHP can be used to access Oracle 8.1.6. The database can reside on either the same server as Apache (local) or on a different server (remote).

LDAP Authentication

`Auth_ldap` is the connector between Apache and an LDAP directory server that allows Apache to authenticate HTTP clients by utilizing entries in an LDAP directory. `Auth_ldap` supports iPlanet (Netscape) Directory Server and OpenLDAP Directory Server. `Auth_ldap` can be configured to use the `stunnel` program for secure SSL queries to the LDAP server. The `stunnel` binary is located in `/opt/hpws/apache/stunnel/sbin`. OpenLDAP SDK resides in `/opt/hpws/apache/lib/LDAP`.

Webmin

Webmin has HP added functionality in administrating Apache 2.0 features, easily displaying default values, and accessing log files. You can also generate keys and certificates for your server or for your own self signed Certificate Authority (CA).

XML Tools

XML Web Server Tools is a group of standards-based XML products that run in the Java environment.

- *Xerces* provides XML parsing and generation
- *Xalan* is an XSLT stylesheet processor for transforming XML documents into HTML, text, or other XML document types.
- *Cocoon* is a framework for XML web publishing that brings a whole new world of abstraction and ease to consolidated web site creation and management based on the XML paradigm and related technologies.
- *FOP* is a print formatter driven by XSL formatting objects. It is a Java 1.2 application that reads a formatting object tree and then turns it into a PDF document. The formatting object tree, can be in the form of an XML document (output by an XSLT engine like Xalan) or can be passed in memory as a DOM Document or (in the case of Xalan) SAX events.
- *Batik* is a Java-based toolkit for applications or applets that want to use images in the Scalable Vector Graphics (SVG) format for purposes such as parsing, viewing, generation or manipulation on either the client side or the server side.

New Utilities

These utilities are found in `/opt/hpws/util`. For more information, please see: `/opt/hpws/hp_docs/utilities.user.guide`

a. altroot.sh

Alternate Root Utility. After installing HP-UX Web Server Suite into the default `/opt/hpws` directory, this script can be used to move it into another directory.

b. cache_util.pl

This interactive utility helps in the creation of cache.conf file for use with `mod_file_cache`. Performance of HP-UX Apache-based Web Server can be improved for serving of static content by using `mod_file_cache`. Frequently accessed static files can be pre-loaded into memory and served directly in order to avoid frequent disk access. Files to be cached by `mod_file_cache` are listed in `cache.conf` file.

c. chroot_os_cp.sh

Chroot copy utility. This is a helper script that sets up chroot by copying typical files used by HP-UX Apache-based Web Server into the chroot directory. The copied set of files allows demo web pages to run. A very secure Apache may require some of these files to be deleted. If you need additional things to be done as part of chroot set-up, you are encouraged to customize this script for your site.

d. mkcert.sh

SSL Certificate Generation Utility. This script generates private keys, certificate signing requests, and certificates for the CA, server, and client. Before you use the `mod_ssl`, you should prepare the SSL certificate system by running the `'mkcert.sh'` command.

e. ports.sh

Port List Utility. This script lists the ports being configured by the HP-UX Apache-based Web Server. In this distribution, ports are configured for Apache, Apache(SSL), Tomcat, mod_jk, Webmin, and LDAP.

f. stunnel_ctl.sh
This is a wrapper utility for starting up the stunnel program. Stunnel is used for SSL connections between Apache and an LDAP directory server. More information on configuring an SSL connection is in the `ldap.admin.guide`.

g. test_certmig.sh
Certificate Migration Utility. This utility is a wrapper around certmig. It can be used to import, extract and list the certificates in an iPlanet 4.1.x Certificate database. For usage information,
Type "`/opt/hpws/apache/util/test_certmig.sh -h`".

Additional features that are part of this release are:

Chroot
Chroot causes the named directory to become the root directory, the starting point for path searches. A malicious user cannot get to the root file system. Our chroot includes SSL enhancements. We include a script, `/opt/hpws/apache/util/chroot_os_cp.sh`, that can be used to copying OS files to your chroot directory.

certmig
The certmig utility makes sharing of certificates between the Netscape Enterprise Server (4.x and above) and any server that supports PKCS#12 formats possible. The certmig utility is an extension of the pk12util utility, provided by the Mozilla community. In addition to the pk12util functionality, certmig lists and extracts certificates from Netscape certificate databases.

Certmig is installed in `/opt/hpws/apache/bin/certmig`. For more information, see `/opt/hpws/hp_docs/apache/certmig.user.guide`. A helper script, `test_certmig.sh`, is located in `/opt/hpws/apache/util` directory. For more information on this script, see `/opt/hpws/hp_docs/apache/utilities.user.guide`.

mod_perl 1.99_07
`mod_perl` is an add-on Apache module that glues together the Perl runtime library, server software and an object-oriented Perl interface to Apache's C language API. It enables Apache modules to be written entirely in Perl and improves performance of Perl cgi scripts. The Prerequisites section in the `apache.admin.guide` contains requirements for `mod_perl`. It is still under development because of its dependency on Apache's API.

auth_ldap 1.5.4
Auth_ldap is the connector between Apache and an LDAP directory server module allowing Apache to authenticate HTTP clients by using entries in an LDAP directory. Auth_ldap supports iPlanet(Netscape) Directory Server and OpenLDAP Server and can be configured to use the stunnel program for secure SSL queries to the LDAP server. Stunnel is started and stopped using the `/opt/hpws/apache/util/stunnel_ctl.sh` utility.

More information on setting up auth_ldap and stunnel can be found in `/opt/hpws/hp_docs/apache/ldap.admin.guide`

apr_shm
apr_shm is a library that abstracts the usage of shared memory on UNIX platforms. It was previously called "Shmem" in this release. apr_shm support allows the `/opt/hpws/apache/conf/ssl.conf` `SSLSessionCache` directives `shmht:/opt/hpws/apache/logs/ssl_scache(51200)` and `shmcb:/opt/hpws/apache/logs/ssl_scache(51200)` to be used. 51200 represents the size of the shared memory being created and can be changed based on the system resources. Shmht refers to the

hash table method of session caching and shmcb refers to the circular buffer method of session caching. Please note that either shmht or shmcb can be specified.

Tomcat 4.1.12

Tomcat is an implementation of the Java Servlets 2.3 and JavaServer Pages 1.2 specifications. Tomcat can work either standalone or with the HP-UX Apache-based Web Server depending on the configuration. Tomcat is pre-configured to run with HP-UX Apache-based Web Server. Tomcat listens on port number 8081.

Tomcat is installed in `/opt/hpws/tomcat`. For more information, please see `/opt/hpws/hp_docs/tomcat/tomcat.admin.guide`

OpenSSL 0.9.6g

OpenSSL is a fully-featured Open Source toolkit implementing the Secure Sockets Layer and Transport Layer Security protocols with full-strength cryptography worldwide.

OpenSSL is installed in `/opt/hpws/apache/ssl/openssl_bsafe/bin/openssl`.

PHP 4.2.3

PHP is an HTML embedded, server-side, cross-platform, scripting language with support for database access. This version includes the security fix to correct POST vulnerabilities in versions 4.2.0 and 4.2.1. For more information see: http://www.php.net/release_4_2_2.php

For more information on PHP, please see `php.admin.guide` and `php.user.guide` in the `/opt/hpws/hp_docs/apache` directory.

Webmin 1.020

Administration and Configuration GUI, a customized version of Webmin for HP-UX Web Server Suite. Webmin can configure the MPM directives that are new to Apache 2.0. It has also been enhanced to generate Certificate Authority (CA) Server Keys and Certificates using the OpenSSL toolkit.

Webmin is installed in the `/opt/hpws/webmin/` directory. For more information on Webmin, please see `/opt/hpws/hp_docs/webmin/webmin.admin.guide`

Apache modules in C++

HP-UX Apache-based Web Server supports loading of Apache modules written in C++. More information regarding building and using C++ modules can be found in `/opt/hpws/apache/build/examples/README` and in the FAQ under Troubleshooting - "Why does my C++ module fail to load?".

Automatic Restart of Apache/Tomcat/Webmin

Apache/Tomcat/Webmin can be started automatically on reboot. More information on customization/configuration of this feature can be found at in `/opt/hpws/hp_docs/apache/apache.admin.guide`

The Apache product is built with options that provide maximum flexibility to allow enabling of new modules or disabling of existing modules. The modules included by default as shared objects with the HP-UX Apache-based Web Server are listed below. These modules are located in the `/opt/hpws/apache/modules/` directory.

For a detailed description of the standard Apache modules, please see
http://httpd.apache.org/docs-2.0/mod/index-bytype.html.

For more detailed information about new and changed features included in each HP-UX Web Server Suite release, see the release notes for each release on the web site:

www.hp.com/go/webserver
Click→"downloads"

For additional information, see:

httpd.apache.org/docs-2.0/

Components & Features of Apache and HP-UX Web Server Suite

Components & Features	Apache 1.3 from ASF[1]	HP Apache 1.3.27 PA/IPF	Apache 2.0 from ASF[1]	HP–UX Web Server Suite PA / IPF [2]
Latest HP Apache Available on HP SW Depot (Aug. 2002)		1.3.27		2.0.43
Apache Web Server	1.3.26	1.3.27	2.0.43	2.0.43
SuEXEC	yes		yes	yes
IPv6**			yes	PA only
*Auto-restart Apache, Tomcat, or Webmin		yes		yes
*Shared Memory Caching		MM v.1.2.1		apr_shm
*Support Apache modules written in C++		yes		yes
*altroot.sh				yes
*cache_util.sh				yes
*ports.sh				yes
Security				
mod_ssl		2.8.11	Built-in	Built-in
OpenSSL		0.9.6g		0.9.6g
auth_ldap		1.6		1.5.4
Stunnel		3.1.4		3.1.4
*Chroot		yes		Yes
*Certmig		PA only		PA only
*test_certmig.sh				PA only
*mkcert.sh		yes		yes
*stunnel_ctl.sh				yes
Scripting				
mod_perl		1.27		1.99_07
PHP		4.2.2		4.2.3
PHP with Oracle extension				yes
Miscellaneous				
mod_dav/mod_dav_fs			yes	yes
mod_proxy	yes	yes	yes	yes
mod_define		yes		
Java				
Tomcat Servlet Container		3.3.1		4.1.12
Apache connector to Tomcat		mod_jserv mod_jk 1.2.0		mod_jk 1.2.0
Apache JServ		1.1.1		
Administration				
Webmin		0.980		1.020
Xmltools				
Xerces				yes
Xalan				yes
Batik				yes
FOP				yes
Cocoon				yes

Notes:

[1] "Apache from ASF" indicates that if you were to go to httpd.apache.org, download and build Apache yourself, these features/components would be included in the standard distribution.

[2] PA-RISC binaries are 32-bit and IPF binaries are 64-bit.

* Indicates HP added feature. Not available in Open Source.

** IPv6 is only available on PA-RISC 11i (11.11) with IPv6 product (T1306AA) installed and 11i Version 1.6 (11.23). Since Apache is dependent on other products such as Java and Perl, some components are not completely supported.

Note: The version numbers in the preceding table are correct for the HP Apache-based Web Server 1.3.27 and HP-UX Web Server Suite initial release. For later releases see the Release Notes in the documentation directory, or on the web site:

www.hp.com/go/webserver
Click →"downloads"

5 Preparing for Installing HP-UX Web Server Suite

5.1 Hardware and Software Requirements

The HP-UX Web Server Suite runs on HP-UX 11.0, 11i, or 11i Version 1.5 or later. There are a small number of required patches that affect functionality. These can be reviewed after installation. They are listed in the Admin Guides for each component which can be found in the directory `/opt/hpws/hp_docs`.

The following table shows the necessary hardware and software for installing and running the HP Apache or HP-UX Web Server Suite. These requirements should be satisfied before beginning a migration.

Hardware and Software Requirements

HP Apache-based Web Server Products	HP-UX Platform	Disk Space	mod_perl	Webmin	Java Servlets and JSPs
HP Apache-based Web Server versions 1.3.x					
v.1.3.27 PA-RISC product # B9415AA	HP-UX 11.0 or 11i	50-60 MB	Perl v.5.6.1	Perl 5 or greater	HP JDK 1.2.2.04 or higher (JDK 1.3 or higher recommended) JSDK 2.0 for ApacheJServ 2.0 servlets
v.1.3.27 IPF Itanium Processor Family (IPF) Pre-enabled for 64-bit Perl product # B9415AA	HP-UX 11i Version 1.5 or later	50-60MB	64-bit Perl for IPF	Perl 5 or greater	HP JDK 1.3.0 for IPF or higher JSDK 2.0 for ApacheJServ 2.0 servlets
HP-UX Web Server Suite					
PA-RISC	HP-UX 11.0 or 11i	200-220MB *	Perl v.5.6.1	Perl 5 or greater	HP JDK 1.2.2.04 or higher (JDK 1.3 or higher recommended)
PA-RISC with IPv6	HP-UX 11i IPv6 product # T1306AA	200-220MB *	Perl v.5.6.1	Perl 5 or greater	HP JDK 1.4 or higher
IPF	HP-UX 11i Version 1.5 or later	200-220MB *	64-bit Perl for IPF	Perl 5 or greater	HP JDK 1.3.0 for IPF or higher

5.2 Disk Space Requirements

200 to220MB of disk space is needed to install the entire HP-UX Web Server Suite. Installed separately, each product uses the:

HP-UX Apache-based Web Server	55 MB
HP-UX Tomcat-based Servlet Engine	20 MB
HP-UX Webmin-based Admin	5 MB
HP-UX XML Web Server Tools	115 MB
To use Cocoon with Tomcat add	40 MB.

5.3 Perl Requirements

Perl is needed when you are using perl scripts, `mod_perl,` or `Webmin`. The Release Notes bundled with the product describe how to configure `mod_perl` and `Webmin`.

`apxs` is a utility perl script provided by Apache for compiling and installing modules. HP Apache-based Web Server 1.3.x for IPF expects perl to be at `/usr/contrib/Q4/bin`. All other versions expect Perl to be at `/opt/perl/bin/perl`.

The `mod_perl` module is an add-on that is compiled into HP Apache or HP-UX Apache-based Web Server but is not configured by default. Using `mod_perl` enables Perl CGI to run faster and allows Apache add-on modules to be written in Perl.

The `Webmin` tool is the web-based GUI administrator for HP Apache or HP-UX Web Server Suite. The `Webmin` tool requires Perl version 5.002 or higher; HP-UX Web Server Suite `mod_perl` requires Perl v 5.6.1. Therefore, using Perl 5.6.1 is recommended. (Perl 5.6.1 is the same version as was required by HP Apache 1.3.19.20 and up.)

On IPF, Apache 2.x is a native 64-bit application. Its `mod_perl` modules require respective 64-bit perl libraries v 5.6.1.

To download Perl 5.6.1, go to HP Software Depot at software.hp.com, search for "Perl v.5.6.1".

5.4 Java Development Kit (JDK) Requirements

As part of its distribution, HP Apache-based Web Server or HP-UX Web Server Suite includes a servlet and JSP container. Apache 1.3.x bundled Tomcat and JServ; HP-UX Web Server Suite bundles only Tomcat (JServ is being phased out). If you use a servlet/JSP container you need to have the HP-UX Java Developer's Kit (JDK) release 1.2.2.04 or later. However it is strongly recommended to use version 1.3.0.2 or later.

The latest versions of Java can be downloaded from: www.hp.com/go/java.

The IPv6 version of HP-UX Web Server Suite requires JDK 1.4 only if your Tomcat servlets or JSPs use IPv6 addresses. For example in many cases Apache will handle the long IPv6 addresses and Tomcat will be insulated from them. In this situation the earlier version of the JDK is sufficient. In other words if you use Java classes that need IPv6 support then JDK 1.4 is required. If you use Java classes without reference to IP addresses then the earlier JDK is adequate.

5.5 Java Servlet Development Kit (JSDK 2.x)

Apache 1.3.x requires JSDK 2.x in order to support Apache's JServ. Since the JServ module is no longer used in HP-UX Web Server Suite, JSDK 2.x is no longer required.

If you are using JServ on HP Apache 1.3.x, you must migrate to `tomcat/mod_jk` when you upgrade to HP-UX Web Server Suite. After you install HP-UX Web Server Suite, perform the steps described in Migrating Tomcat and Java.

See also the "`tomcat.migration.guide`", which is available at:

```
/opt/hpws/hp_docs/tomcat
```

6 Installing HP-UX Web Server Suite

The following procedures describe two methods of upgrading to HP-UX Web Server Suite. Pick the procedure that is appropriate to your site environment.

- Installing HP-UX Web Server Suite on a Server Where 1.3.x is Not Running
  Pick this method if it is OK to have the web server you are upgrading unavailable while you are installing HP-UX Web Server Suite.

- Installing HP-UX Web Server Suite on a Server Concurrently Running 1.3.x
  Pick this method if you wish to have HP Apache 1.3.x continuously running on the server that you are upgrading to HP-UX Web Server Suite.

6.1 Installing HP-UX Web Server Suite on a Server Where 1.3.x is Not Running

Step 1: Preparing the Web Server Environment

If you have not already done so, see the preceding section Preparing for Installing HP-UX Web Server Suite, before continuing with the installation. (A reminder… backup your system.)

Step 2: Remove Technology Preview Installation

If you participated in the Technology Preview Program and still have HP Apache-based Web Server Version 2.0.0 Tech Preview installed on your system, make sure to remove it at this time. Save any Apache configuration files, ssl certificates, Tomcat configuration files, and webapps that you may have changed or added.

To check to see if it is installed, type:

```
/usr/sbin/swlist | grep B9416AA
```

The product is installed if it is listed:

```
  B9416AA          2.0.00.00.02   HP Apache-based Web Server with Strong
(128bit) Encryption
```

To stop Apache, Tomcat, and Webmin as root type:

```
/opt/hpapache2/bin/apachectl stop
/opt/hpapache2/tomcat/bin/shutdown.sh
/opt/hpapache2/webmin/webmin-init stop
```

To uninstall the Technology Preview, as root, type

```
/usr/sbin/swremove B9416AA
```

Also remove the directories completely. As root type

```
rm –rf /opt/hpapache2
```

Step 3: Stopping HP Apache 1.3.x

Before you install hpws, (which uses default ports 80 and 443) you want to make sure that HP Apache Version 1.3.x (which uses the same default ports) is stopped. This ensures that there will be no conflict in port numbers between the two versions of HP Apache.

To stop Apache, Tomcat, and Webmin, as root type:

```
/opt/apache/bin/apachectl stop
/opt/tomcat/bin/shutdown.sh
/opt/apache/webmin/webmin-init stop
```

Step 4: Performing the Installation

The HP-UX Web Server Suite product bundle is part of HP-UX. There are two products. One is for IPv4 and has a product number of B9416AA. The other is for the new long IP addresses defined by IPv6 and it has product number of B9416BA. This example uses the IPv4 product. To install HP-UX Web Server Suite, you can use one of the two following methods:

- Install using the HP SW Depot Web Release, www.hp.com/go/webserver→Click "downloads".
- Install using the HP-UX 11.0/11i Application Release CDs (when available).

Step 5: Startup After Installation

Type the following command line to start Apache after installation.
`/opt/hpws/apache/bin/apachectl start`

Step 6: Perform a Quick Check of the HP-UX Web Server Suite Installation

Access the index.html page by typing:
`http://yourserver.com`
You should see the HP-UX Apache-based Web Server home page.

Execute the test CGI script by typing:
`http://yourserver.com/cgi-bin/test-cgi`
You should see several lines of text listing environment variables. The first line should be:
`CGI/1.0 test script report:`

Step 7: Stop HP-UX Apache-based Web Server

Stop HP-UX Apache-based Web Server, if it is running, before continuing with the migration. As root, type:
`/opt/hpws/apache/bin/apachectl stop`

Step 8: Migrating to an HP-UX Web Server Suite Environment

You now need to configure your HP-UX Web Server Suite, to retain any customizations you have made to your HP Apache-based Web Server 1.3.x environment. Go to Migrating Your Apache Configuration from 1.3.x to HP-UX Apache-based Web Server for detailed instructions.

Also, look at the release documents that are bundled with the product to become familiar with HP-UX Web Server Suite's capability. For administrator guides, user guides and configuration information, see:

```
/opt/hpws/hp_docs
```

6.2 Installing HP-UX Apache-based Web Server on a Server Concurrently Running 1.3.x

HP Apache 1.3.x and HP-UX Apache-based Web Server
can run simultaneously on the same machine if they use different port numbers or use different IP addresses. Each web server has its own binary.

Both HP Apache-based Web Server 1.3.x and HP-UX Web Server Suite use port 80 and port 443 (SSL) by default. If you want to maximize availability during the migration by keeping HP Apache 1.3.x running on the server you are upgrading, you can temporarily change ports using the following procedure.

Step 1: Preparing the Web Server Environment

If you have not already done so, see the preceding section Preparing for Installing HP-UX Web Server Suite, before continuing with the installation. (A reminder… backup your system.)

Step 2: Remove Technology Preview Installation

If you participated in the Technology Preview Program and still have HP Apache-based Web Server Version 2.0.0 Tech Preview installed on your system, make sure to remove it at this time. Save any Apache configuration files, ssl certificates, Tomcat configuration files, and webapps that you may have changed or added.

To check to see if it is installed, type:

```
/usr/sbin/swlist | grep B9416AA
```

The product is installed if it is listed:

```
  B9416AA          2.0.00.00.02   HP Apache-based Web Server with Strong
(128bit) Encryption
```

To stop Apache, Tomcat, and Webmin as root type:

```
/opt/hpapache2/bin/apachectl stop
/opt/hpapache2/tomcat/bin/shutdown.sh
/opt/hpapache2/webmin/webmin-init stop
```

To uninstall the Technology Preview, as root, type

```
/usr/sbin/swremove B9416AA
```

Also remove the directories completely. As root type

```
rm –rf /opt/hpapache2
```

Step 3: Performing the Installation

While HP Apache 1.3.x is running, go ahead and perform the HP-UX Web Server Suite installation.

The HP-UX Web Server Suite product bundle is part of HP-UX. There are two products. One is for IPv4 and has a product number of B9416AA. The other is for the new long IP addresses defined by IPv6 and it has product number of B9416BA. This example uses the IPv4 product. To install HP-UX Web Server Suite, you can use one of the two methods listed following:

- Install using the HP SW Depot Web Release, www.hp.com/go/webserver→Click “downloads”.
- Install using the HP-UX 11.0/11i Application Release CDs (when available).

Step 4: Assigning Port Numbers

You need to modify the `httpd.conf` and the `ssl.conf` files to assign non-defaults ports in HP-UX Web Server Suite

Default Port Assignments in HP-UX Web Server Suite:

Check that the proposed new port assignments are not already being used by typing:
`netstat -a | egrep "8080|8443"`
Select other unused port numbers if necessary.

In `/opt/hpws/apache/conf/httpd.conf` the port is configured by:

```
Listen  80
```

To eliminate the conflict change it to:

```
Listen 8080
```

In `/opt/hpws/apache/conf/ssl.conf` the ports are configured by:

```
Listen 443
ServerName www.yourserver.com:443
VirtualHost <www.yourserver.com:443>
```

To eliminate conflicts change them to:

```
Listen 8443
ServerName www.yourserver.com:8443
VirtualHost <www.yourserver.com:8443>
```

Step 5: Startup After Changing Port Assignments

Start up HP-UX Web Server Suite by typing as root:
`/opt/hpws/apache/bin/apachectl startssl`

Step 6: Perform a Quick Check of the HP-UX Web Server Suite Installation

In a browser enter the following URLs.

Access the index.html page by typing:
`http://yourserver.com:8080`
You should see the HP-UX Apache-based Web Server home page.

Execute the test CGI script by typing:
`http://yourserver.com:8080/cgi-bin/test-cgi`
You should see several lines of text listing environment variables. The first line should be:
`CGI/1.0 test script report:`

Access the index.html page securely by typing:
`https://yourserver.com:8443`
You should see the HP-UX Apache home page.

Step 7: Stop HP–UX Apache

Stop HP-UX Web Server Suite, if it is running, before continuing with the migration. As root, type:
`/opt/hpws/apache/bin/apachectl stop`

Step 8: Migrating to an HP-UX Web Server Suite Environment

You now need to configure your HP-UX Web Server Suite, to retain any customizations you have made to your HP Apache-based Web Server 1.3.x environment. Go to Migrating Your HP Apache Configuration from 1.3.x to HP-UX Web Server Suite for detailed instructions.

Also, look at the release documents that are bundled with the product to become familiar with HP-UX Web Server Suite's capability. For administrator guides, user guides and configuration information, see:
`/opt/hpws/hp_docs`

7 Migrating Your Apache Configuration from 1.3.x to HP-UX Apache-based Web Server

During the installation of HP-UX Web Server Suite, the new default `httpd.conf` file that configures the operation of HP-UX Web Server Suite is placed in `/opt/hpws/apache/conf/httpd.conf`. You will need to make sure that this new file incorporates any customizations that you made to the 1.3.x version of the file `/opt/apache/conf/httpd.conf`.

You can accomplish these migration changes by two methods. You can edit your old 1.3.x `httpd.conf` file to bring it into conformance with HP-UX Web Server Suite requirements (usually the most convenient method). If you use this method, make sure the updated `httpd.conf` file is placed in `/opt/hpws/apache/conf`. As an alternative method, you can transfer your 1.3.x `httpd.conf` file customizations to the new HP-UX Web Server Suite version of the `httpd.conf` file. See Quick Guide to Migration Solutions

The following sections list the features that are new, changed, and deleted in HP-UX Web Server Suite, and the changes you may need to make.

You should follow the suggested order of migration because some things will not work unless previous steps have been done.

7.1 Migrating the HP Apache Core

To migrate the HP Apache Core, see the following sections:

- Migrating Process Handling
- Migrating Module Loading
- Migrating Logging
- Migrating the Printing of Error Messages
- Migrating Port Number Assignments
- Migrating Access to Configuration Files

7.1.1 Migrating Process Handling

In HP-UX Apache-based Web Server you should not have to adjust the Process Handling directives because Apache will dynamically adjust the number of processes and threads based on the load. If you have extremely high loads then you should increase the `MinSpareThreads` and `MaxSpareThreads` directives.

Process directives have changed between HP Apache 1.3.x and HP-UX Web Server Suite Apache.
HP Apache 1.3.x is process-oriented and the directives in Table 6 reflect that fact. More information is available at httpd.apache.org/docs/mod/core.html.

HP Apache 1.3.x Process Directives in /opt/apache/conf/httpd.conf

Process Directive	Default Value	Description
`StartServers`	`5`	*`StartServers`* refers to number of servers to initially start.
`MaxClients`	`150`	*`MaxClients`* refers to the maximum number of child processes running simultaneously. It directly translates to the maximum number of concurrent requests served by the web server.

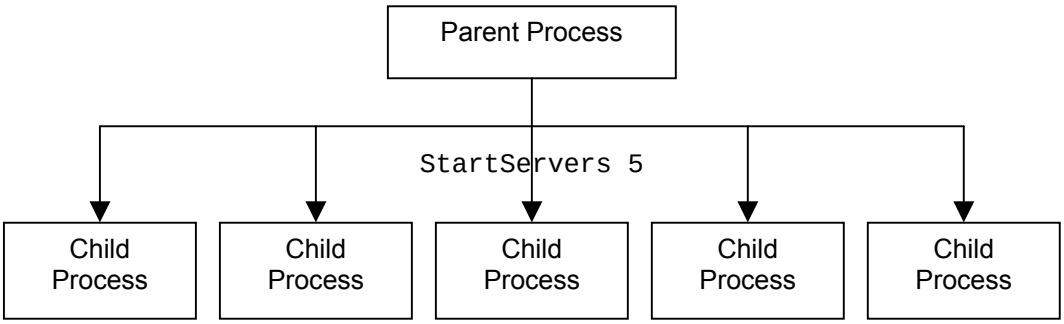
Process Directive	Default Value	Description
MinSpareServers	5	*MinSpareServers* sets the desired minimum number of idle child processes at any time. It serves as a trigger point for Apache to automatically spawn new processes when the number of idle processes falls below *MinSpareServers.*
MaxSpareServers	10	*MaxSpareServers* sets the desired maximum number of idle child processes at any time. It serves as a trigger point for Apache to automatically kill spare idle processes, when the number of idle processes goes above *MaxSpareServers.*
MaxRequestsPerChild	0	*MaxRequestsPerChild* sets the number of requests a child process will handle, before it is killed (aged-out). A value of 0 says that the child process would never expire.

The following table shows the default process directives in HP-UX Apache-based Web Server. HP-UX Apache-based Web Server is thread-oriented as reflected in the following table. More information is available at httpd.apache.org/docs-2.x/mod/worker.html.

HP-UX Apache Web Server Process Directives in /opt/hpws/apache/conf/httpd.conf

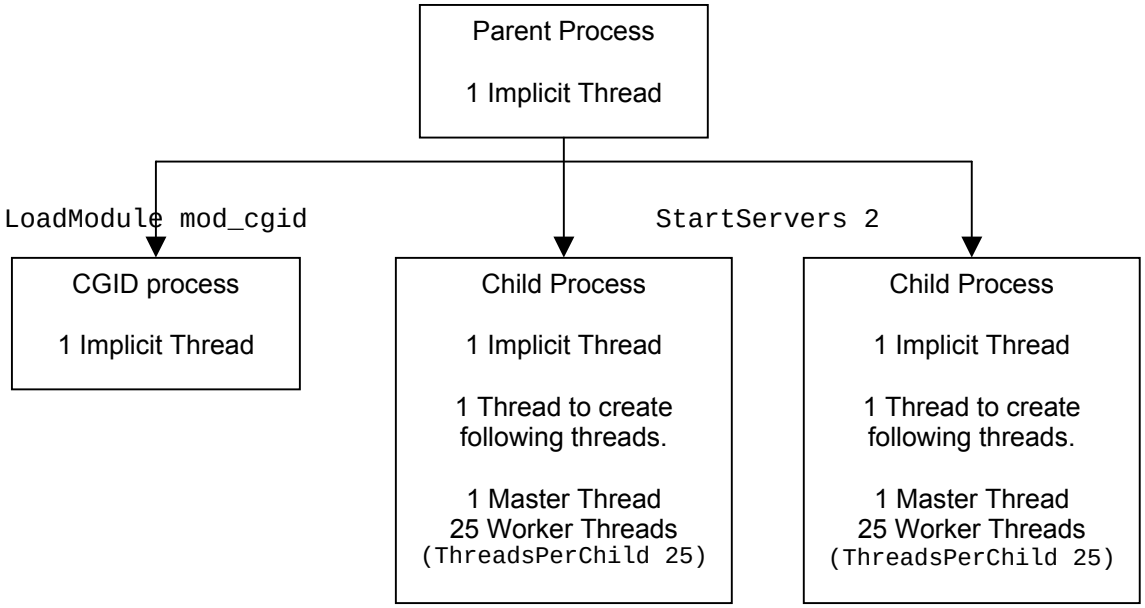
Process Directive	Default Value	Description
StartServers ThreadsPerChild	2 25	*StartServers* refers to number of server processes to start at start-up time. However, the number of "workers" available to serve requests is dependent on the ThreadsPerChild directive. Hence, in this example, the number of workers available is 2*25 = 50.
MaxClients	8	*MaxClients* refers to the maximum number of child processes running simultaneously. In conjunction with the ThreadsPerChild directive, it translates to maximum number of concurrent requests served, in this case 8*25=200 requests.
MinSpareThreads	25	*MinSpareThreads* sets the desired minimum number of idle threads at any time. It serves as a trigger point for Apache to automatically spawn a new process, when the number of idle threads falls below 25 (in this example). Spawning of each new process directly translates into 25 (ThreadsPerChild) new threads.
MaxSpareThreads	75	*MaxSpareThreads* sets the desired maximum number of idle threads at any time. It serves as a trigger point for Apache to automatically kill spare idle processes, when the number of idle processes goes above 75 (as in this example). Killing of each process directly translates into 25 (ThreadsPerChild) killed threads.
MaxRequestsPerChild	0	*MaxRequestsPerChild* sets the number of requests a child process will handle, before it is killed (aged-out). A value of 0 says that the child process would never expire.

HP Apache 1.3.x Multiprocess Diagram



`MinSpareServers 5`
`MaxSpareServers 10` so a quiet server with no connections would have a minimum of 5 and maximum of 10 child processes running.

HP-UX Apache-based Web Server Worker Multiprocess Multithread (MPM) Diagram



`MinSpareThreads 25`
`MaxSpareThreads 75` so a quiet server with no connections would have a minimum of 3 and a maximum of 4 child processes running.

Verification:
Type:
`ps -f | grep httpd`
With the default settings you should see 4 httpd processes

7.1.2 Migrating Module Loading

In HP Apache 1.3.x the `AddModule` directives specified what modules to enable, and the `LoadModule` directives specified in what order the modules were to be loaded. However, in Apache 2.0, the order in which module loading occurs is handled automatically. Therefore, the `AddModule` and `ClearModuleList` directives have been removed. The `LoadModule` directive is the only one required to specify which modules to load.

Loading Modules in HP Apache 1.3.x:

Modules were loaded in HP Apache 1.3.x with the following directives in `/opt/apache/conf/httpd.conf`

```
AddModule mod_xyz
...
LoadModule mod_xyz libexec/mod_xyz.so
```

Loading Modules in HP-UX Apache-based Web Server:

Let's say you have added 1.3.x modules to the HP Apache 1.3x `httpd.conf` file. To move these modules to HP-UX Apache, first make sure you have the 2.x versions of the modules. (If you have user-created modules, see Porting User-Crea
ted Modules to HP Apache 2.x for more information.) For example, suppose that in 1.3.x you had added the *mod_xyz* module, and now want to add it to HP-UX Apache-based Web Server. Your entry in the `/opt/hpws/apache/conf/httpd.conf` file would look like this:

```
LoadModule mod_xyz modules/mod_xyz.so
```

7.1.3 Migrating Logging

The `AgentLog`, `RefererLog`, and `RefererIgnore` directives have been removed in HP-UX Apache Web Server. Agent and referer logs are still available using the `CustomLog` and `LogFormat` directives of `mod_log_config`. The `SetEnvIf` directive of `mod_setenvif` can be used for ignoring referers.

Logging in HP Apache 1.3.x:

The `AgentLog` directive was defined in `mod_log_agent`. `RefererLog` and `RefererIgnore` directives were defined in `mod_log_referer`. Logging was enabled by default in HP Apache 1.3.x.

```
AgentLog logs/agent_log
RefererLog logs/referer_log
RefererIgnore www.yourserver.com
```

Logging in HP-UX Apache-based Web Server:

By default logging is disabled in HP-UX Apache-based Web Server because it affects performance. The procedure is to first define a format with the LogFormat directive and then use the CustomLog directive to attach it to a specific log file. 1.3.x functionality can achieved with the following directives:

```
LogFormat "%{User-agent}i" agent
CustomLog logs/agent_log agent

LogFormat "%{Referer}i -> %U" referer
SetEnvIf Referer www\.yourserver\.com server-request
CustomLog logs/referer_log referer env=!server-request
```

7.1.4 Migrating the Printing of Error Messages

The 1.3.x method continues to work but according to ASF `ErrorDocument` directive information, you must provide the closing quote. It is unclear if ASF will enforce this in future releases.

Printing Error Messages in HP Apache 1.3.x:

Error messages were specified in 1.3 x in `/opt/apache/conf/httpd.conf`.

The `ErrorDocument` directive used a quote only at the beginning of the argument to indicate a text message.

```
ErrorDocument 403 "Some Message
```

Printing Error Messages in HP-UX Apache-based Web Server:

The `ErrorDocument` directive no longer uses a quote at the beginning of the argument to indicate a text message. Instead enclose the message in double quotes.

Error messages are specified in 2.x in `/opt/hpws/apache/conf/httpd.conf`:

```
ErrorDocument 403 "Some Message"
```

Verify

Currently no error is generated if you do it the old way.

7.1.5 Migrating Port Number Assignments

HP Apache-based Web Server 1.3.x and HP-UX Web Server Suite can run simultaneously on the same machine if they use different port numbers or use different IP addresses. Each web server has its own binary.

HP-UX Web Server Suite can be installed in any location allowing multiple Apache 2.x installations to run on the same machine provided they use unique port numbers or use different IP addresses.

Assigning Ports in HP Apache 1.3.x:

Ports were assigned in 1.3.x in the "`/opt/apache/conf/httpd.conf`" file.

For example, if you wanted HP Apache to accept requests on Port 80, the following directive would be specified:

```
Port 80
```

However, if you wanted the same apache server to listen in on an additional port, say 8000, you would use the following directives together.

```
Port 80
Listen 80
Listen 8000
```

Assigning Ports in HP-UX Web Server Suite:

In the 2.x version of Apache the directives `Port` and `BindAddress` have been removed. Equivalent functionality is provided by the `Listen` directive. The `Listen` directive tells the server to accept incoming requests only on the specified port or address-and-port combinations. If only a port number is specified in the `Listen` directive, the server listens to the given port on all interfaces. If an IP address is given as well as a port, the server will listen on the given port and interface. Multiple `Listen` directives may be used to specify a number of addresses and ports to listen to. The server will respond to requests from any of the listed addresses and ports.

To accept requests on Port 80, the directive would be specified in `/opt/hpws/apache/conf/httpd.conf.`

```
Listen 80
```

However, if you wanted the same apache server to listen in on an additional port, say 8000, you would use the following directives together.

```
Listen 80
Listen 8000
```

To make the server accept connections on two specified interfaces and port numbers, use

```
Listen 192.170.2.1:80
Listen 192.170.2.5:8000
```

Binding to a Particular Address in HP Apache 1.3.x and HP-UX Web Server Suite:

In Apache 1.3.x, BindAddress made the server bind to just the specified address. If the argument was * (an asterisk), the server bound to all interfaces currently marked as up on the server. The Port directive sets which port to bind to. Only one BindAddress should be used.

In 2.0, the Listen directive can be used to achieve the same result.

HP Apache 1.3.x	HP Apache 1.3.x Alternate Method	HP-UX Apache-based Web Server
`BindAddress *` `Port 80`	`Listen *:80`	`Listen *:80`

`BindAddress`
`111.222.333.444`
`Port 80`
	`Listen`
	`111.222.333.444:80`
		`Listen 111.222.333.444:80`

`BindAddress`
`yourserver.com`
`Port 80`
	No equivalent. Can use only IP address.	No equivalent. Can use only IP address.

No support for IPv6	No support for IPv6	`Listen [fe80::1]:80`

Verify:

Stop and start Apache.
The default Apache configuration starts four httpd processes. Verify that they are all running by typing:
`ps -e | grep httpd`

Note:
In both HP Apache 1.3.x and HP-UX Apache-based Web Server, `Listen` does not implement Virtual Hosts. It only tells the main server what addresses and ports to listen to. If no `<VirtualHost>` directives are used, the server will behave the same for all accepted requests. However, `<VirtualHost>` can be used to specify a different behavior for one or more of the addresses and ports. To implement a `VirtualHost`, the server must first be told to listen to the address and port to be used. Then a `<VirtualHost>` section should be created for a specified address and port to set the behavior of this virtual host. If the `<VirtualHost>` is set for an address and port that the server is not listening to, it cannot be accessed.

For more information, see the documentation on Listen directive, Virtual Hosts, DNS Issues or <VirtualHost> section. See also, Setting which addresses and ports Apache uses.

7.1.6 Migrating Access to Configuration Files

In Version 1.3.x, HP Apache commented out the `ResourceConfig` and `AccessConfig` directives and the files `srm.conf` and `access.conf` had only comments. Since Apache 1.3.6 directives from `srm.conf` and `access.conf` have been included within `httpd.conf`. The `ResourceConfig` and `AccessConfig`

directives have been removed in HP-UX Apache-based Web Server. Existing instances of these directives can be replaced with the "`Include`" directive which has equivalent functionality.

If you were making use of the default values of the `ResourceConfig` and `AccessConfig` directives in HP Apache 1.3.x without including them in the configuration files, then in HP-UX Apache-based Web Server you may need to add `Include conf/access.conf` and `Include conf/srm.conf` directives in your `/opt/hpws/apache/conf/httpd.conf` file.

In order to assure that Apache reads the configuration files in the same order as implied by the older directives, the `Include` directives should be placed at the end of `httpd.conf`, with the one for `srm.conf` preceding the one for `access.conf`.

7.2 Migrating Included Modules

See Module Changes in the Appendix.

7.2.1 Migrating mod_proxy

The 1.3.x mod_proxy module included both proxy and cache functionality. HP-UX Apache-based Web Server seperates these two. Now `mod_proxy` just does proxying.

`mod_cache` implements an RFC 2616 compliant HTTP content cache that can be used to cache either local or proxied content. `mod_cache` requires the services of one or more storage management modules. Two storage management modules are included in HP-UX Apache-based Web Server :

- `mod_disk_cache`
  A disk based storage manager, generally used for proxy caching. For a cached file the proxy server will serve the request directly and avoid the overhead of sending the request on to the web server.
- `mod_file_cache`
  An mmap based and/or file handle based storage manager that allows the server file system to control whether the file is in memory or not. This replaces all the functionality of the 1.3.x module, `mod_mmap_static`.
- `mod_mem_cache`
  An in-memory based storage manager, primarily useful for caching local content.

7.2.2 Migrating How Documents are Cached by Proxy Servers

If it is set, the `CacheNegotiatedDocs` directive allows content-negotiated documents to be cached by proxy servers. Prior to HP-UX Web Server Suite, `CacheNegotiatedDocs` did not take an argument; it was turned on by the presence of the directive by itself.

In HP-UX Web Server Suite the `CacheNegotiatedDocs` directive takes the argument `on` or `off`. In the `httpd.conf` file, replace existing instances of `CacheNegotiatedDocs` with `CacheNegotiatedDocs on`.

7.3 Migrating Security

To migrate Security from HP Apache 1.3.x to HP-UX Apache-based Web Server, see the following sub-sections:

7.3.1 Migrating SSL

In the 1.3.x version of HP Apache-based Web Server, SSL configuration information was provided in the `httpd.conf` file. In the HP-UX Apache-based Web Server SSL configuration information has been moved to `ssl.conf`.

To migrate SSL from 1.3.x to 2.x, you should move your customized SSL configuration information from the 1.3.x version of the `/opt/apache/conf/httpd.conf` file to `/opt/hpws/apache/conf/ssl.conf`.

In addition you need to edit your `ServerName` directives by adding a colon and port number after each `ServerName` directive. You also need to replace your SSL logging directives with Apache logging directives. See the following example.

SSL in HP Apache 1.3.x:

SSL was enabled in 1.3.x with the following entry in `/opt/apache/conf/httpd.conf` .

```
##
## SSL Virtual Host Context
##
<VirtualHost _default_:443>
#  General setup for the virtual host
DocumentRoot "/opt/apache/htdocs"
ServerName yourserver.com
```

SSL Logging has its own directives in `/opt/apache/conf/httpd.conf`.

```
SSLLog      /opt/apache/logs/ssl_engine_log
SSLLogLevel info
```

SSL in HP-UX Apache-based Web Server:

You can no longer proxy through SSL.

SSL is enabled with the following entry in `/opt/hpws/apache/conf/ssl.conf` .

```
##
## SSL Virtual Host Context
##
<VirtualHost _default_:443>
#  General setup for the virtual host
DocumentRoot "/opt/hpws/apache/htdocs"
ServerName yourserver.com:443
```

SSL logging uses the Apache directives in `/opt/hpws/apache/conf/ssl.conf` .

```
ErrorLog logs/error_log
LogLevel info
```

7.3.2 Migrating Chroot

`Chroot` remains unchanged between HP Apache 1.3.x and HP-UX Web Server Suite.

7.4 Migrating Scripting Modules

To migrate Scripting Modules from HP Apache 1.3.x to HP-UX Apache-based Web Server, see the following sub-sections:

- Migrating PHP
- Migrating Perl

- Migrating CGI
- Migrating Server-Side Includes

7.4.1 Migrating PHP

PHP in HP Apache 1.3.x:

PHP was enabled in 1.3.x with the following entries in the `/opt/apache/conf/httpd.conf` file:

```
LoadModule php4_module libexec/libphp4.so
AddModule mod_php4.c
```

PHP in HP-UX Apache-based Web Server:

To enable PHP uncomment the following lines or make sure they are present in `/opt/hpws/apache/conf/httpd.conf`:

```
LoadModule php4_module modules/libphp4.so

<Files *.php>
    SetOutputFilter PHP
    SetInputFilter PHP
</Files>
```

To use `PATH_INFO`, you must explicitly set directive `AcceptPathInfo`, otherwise a 404 response will be returned. In 1.3, `PATH_INFO` is enabled by default, so a PHP script `/script.php` will be invoked if a request is made to the location `/script.php/foo/bar`, passing a `PATH_INFO` of `/foo/bar` to the script.

7.4.2 PHP Oracle support in HP-UX Apache-based Web Server

PHP Oracle support is new at HP-UX Web Server Suite.

To enable PHP Oracle, edit the file `/opt/hpws/apache/bin/apachectl.` Uncomment the four individual lines that each begin with export:

```
export ORACLE_HOME=/opt/oracle
export ORACLE_SID=dummy_sid
export
   LD_PRELOAD="$LD_PRELOAD:$ORACLE_HOME/JRE/lib/PA_RISC/native_threads/libjava.sl"
export SHLIB_PATH="$SHLIB_PATH:$ORACLE_HOME/lib"
```

7.4.3 Migrating Perl

Perl in HP Apache 1.3.x:

Perl was enabled in 1.3.x with the following entry in `/opt/apache/conf/httpd.conf` .

```
AddModule mod_perl.c
<IfModule mod_perl.c>
   PerlModule Apache::Registry
   <Files *.pl>
       SetHandler perl-script
       PerlHandler Apache::Registry::handler
       Options ExecCGI
   </Files>
</IfModule>
```

For backward `mod_perl` compatability use perl module `Apache::compat.`

See http://perl.apache.org/docs/2.0/user/compat/compat.html for incompatabilities between mod_perl 1.0 and 2.0

Perl in HP-UX Apache-based Web Server:

Perl is enabled with the following entry in /opt/hpws/apache/conf/httpd.conf: .

```
LoadModule perl_module modules/mod_perl.so
<IfModule mod_perl.c>
    PerlModule ModPerl::Registry
    <Files *.pl>
        SetHandler perl-script
        PerlHandler ModPerl::Registry::handler
        Options ExecCGI
        PerlOptions +ParseHeaders
    </Files>
</IfModule>
```

7.4.4 Migrating Common Gateway Interface (CGI)

The module mod_cgid is new in HP-UX Apache-based Web Server.

Forking a process from a multi-threaded server can be a very expensive operation because the new process will replicate all the threads of the parent process. In order to avoid incurring this expense on each CGI invocation, in HP-UX Apache-based Web Server mod_cgid creates an external daemon that is responsible for forking child processes to run CGI scripts. The main server communicates with this daemon using a unix domain socket.

At the user level, this module is identical in configuration and operation to mod_cgi. The only exception is the additional directive Scriptsock which gives the name of the socket to use for communication with the cgid daemon.

CGI in HP Apache 1.3.x:

CGI was enabled in 1.3.x with the following entry in /opt/apache/conf/httpd.conf .

```
AddModule  mod_cgi.c
...
LoadModule cgi_module  libexec/mod_cgi.so
```

CGI in HP-UX Apache-based Web Server:

The cgid daemon is turned on in HP-UX Apache-based Web Server by default. The Scriptsock directive is used to set the UNIX socket for communicating with cgid.

The relevant parts of httpd.conf are:

```
LoadModule cgid_module modules/mod_cgid.so
#LoadModule cgi_module modules/mod_cgi.so

<IfModule mod_cgid.c>
   Scriptsock            logs/cgisock
</IfModule>
```

7.4.5 Migrating Server-Side Includes (SSI)

In HP Apache 1.3.x:

In 1.3.x Server-Side Includes were provided by `mod_include`. Any document with handler of "`server-parsed`" was handled by `mod_include`, if the `Includes` option was set. If documents containing server-side include directives were given the extension `.shtml`, the following directives made Apache parse them and assign the resulting document the mime type of text/html:

```
AddType text/html .shtml
AddHandler server-parsed .shtml

Options +Includes
```

In HP-UX Apache-based Web Server :

Server-Side Includes provided by mod_include are now implemented using the new filter system rather than as a handler. This provides much more power and flexibility, but requires the use of the SetOutputFilter to activate server-parsed content. If you were using AddHandler server-parsed `.shtml` in HP Apache 1.3.x, you can get similar functionality in HP-UX Apache-based Web Server by using,

```
<FilesMatch "\.shtml(\..+)?$">
   SetOutputFilter INCLUDES
<FilesMatch>
```

7.5 Migrating Java

HP Apache 1.3.x has two servlet containers for running java server applications. One container is JServ and the other is Tomcat. In 1.3.x mod_jserv connects Apache to JServ or Tomcat. In 1.3.27, mod_jk is also available to connect Apache to Tomcat. Migrating from 1.3.x to HP-UX Apache-based Web Server is mainly a path change. HP-UX Apache-based Web Server does not support the mod_jserv connector or the JServ container.

7.5.1 Migrating Tomcat with mod_jserv to Tomcat with mod_jk

The new HP-UX Tomcat-based Servlet Engine uses Tomcat 4.1. Please refer to /op/hpws/hp_docs/tomcat/tomcat.migration.guide for more information.

In 1.3.x, Tomcat uses mod_jserv or mod_jk as the connector while in 2.x Tomcat uses mod_jk.

A) File structure

1.3.x with Tomcat (mod_jserv)

```
/opt/apache/conf/
   httpd.conf
/opt/tomcat/conf/
   tomcat.conf
   server.xml
```

1.3.27 with Tomcat (mod_jk)

```
/opt/apache/conf/
   httpd.conf
/opt/tomcat/conf/
   server.xml
   apps-examples.xml
   jk/
         mod_jk.conf
         workers.properties
```

2.x

```
/opt/hpws/apache/conf/httpd.conf
/opt/hpws/tomcat/conf/server.xml
/opt/hpws/tomcat/jk/apache2/mod_jk.conf
/opt/hpws/tomcat/jk/apache2/workers.properties
```

B) Apache configuration file

In the 1.3.x file `httpd.conf`:

```
Include /opt/tomcat/conf/tomcat.conf
```

In the 1.3.27 file `httpd.conf`:

```
Include /opt/tomcat/conf/jk/mod_jk.conf
```

In the 2.x file `httpd.conf`:

```
Include /opt/hpws/tomcat/jk/apache2/mod_jk.conf
```

C) Passing URLs from Apache to Tomcat

In the 1.3.x file `tomcat.conf`:

```
ApJServMount /examples /root
```

In the 1.3.27 and 2.x file `mod_jk.conf`:

```
JkMount /examples ajp13
JkMount /examples/* ajp13
```

D) Log files

In the 1.3.x file

`tomcat.conf`:

```
ApJServLogLevel debug
```

`server.xml`:

```
<Logger name="tc_log"
    path="logs/tomcat.log"
    customOutput="yes" />
```

In the 1.3.27 file

`mod_jk.conf`:

```
JkLogFile /opt/tomcat/logs/jk.log
JkLogLevel emerg
```

`server.xml`:

`<LogSetter` can be used to create servlet_log, JASPER_LOG, and tag_pool_log log files.

`<AccessLogInterceptor` can be used to create access logs.

In the 2.x file

`mod_jk.conf`:

```
JkLogFile /opt/hpws/tomcat/logs/jk.log
JkLogLevel emerg
```

`server.xml`:

```
<!-- Global logger unless overridden at lower levels -->
<Logger className="org.apache.catalina.logger.FileLogger"
prefix="catalina_log." suffix=".txt"
timestamp="true"/>
```

E) Defining the connection worker

In the 1.3.x file `tomcat.conf`:

```
ApJServDefaultProtocol ajpv12
ApJServDefaultPort 8007
```

In the 1.3.27 and 2.x file `workers.properties`:

```
worker.ajp12.port=8007
worker.ajp12.host=localhost
worker.ajp12.type=ajp12
worker.ajp13.port=8009
worker.ajp13.host=localhost
worker.ajp13.type=ajp13
```

F) Web application contexts

In the 1.3.x file `server.xml`:

```
<Context path="/examples"
```

```
    ...
```

In the 1.3.26 file `apps-examples.xml`:

```
    <Context path="/examples"
    ...
```

In the 2.x file `server.xml`:

```
    <Context path="/examples"
```

G) Handling .jsp files

In the 1.3.x file `tomcat.conf`:

```
    ApJServMount default /root
    AddType text/jsp .jsp
    AddHandler jserv-servlet .jsp
```

In 1.3.26 and 2.x:

Tomcat is a servlet/jsp container so a web application like "examples" handles jsp's properly. For instance Apache will pass URIs starting with "`/examples`" to Tomcat and Tomcat will handle the `*.jsps`. In Tomcat, the JSP interceptor handles loading of JSP pages. Its behavior can be customized in the `server.xml` file.

7.5.2 Migrating JServ to Tomcat

If you are migrating from JServ then see:
`/opt/hpws/hp_docs/tomcat/tomcat.migration.guide`
Also see the following:
jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html

Here is an example with steps to get the following URL to work with a "HelloWorldExample" servlet:
`yourserver.com/my_app/HelloWorldExample`

1. Assumptions
   A) All your java classes are in:
   `/my_servlets`
2. Create a myapp directory inside Tomcat's `/opt/hpws/tomcat/webapps`
   ```
   $ cd /opt/hpws/tomcat/webapps
   $ mkdir myapp
   $ mkdir myapp/WEB-INF
   ```
3. Create a symbolic link so servlets in /my_servlets are visible inside WEB-INF
   ```
   $ cd /opt/hpws/tomcat/webapps/myapp/WEB-INF
   $ ln -s /my_servlets classes
   ```
4. Create a web.xml file in /opt/hpws/tomcat/webapps/myapp/WEB-INF
   With the following contents:
   ```
   <?xml version="1.0" encoding="ISO-8859-1"?>

   <!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application
   2.3//EN" "http://java.sun.com/j2ee/dtds/web-app_2_3.dtd">

<web-app>

    <servlet-mapping>
        <servlet-name>
            invoker
        </servlet-name>
        <url-pattern>
            /*
        </url-pattern>
    </servlet-mapping>
   ```

```
</web-app>
```

Restart Tomcat and at this point we should be able to access servlets by using http://yourserver.com:8081/myapp. To access servlets through Apache we need one more step:

5. Map tomcat context to apache Edit `/opt/hpws/apache/conf/httpd.conf` and uncomment the following line:
   `Include /opt/hpws/tomcat/jk/apache2/jk/mod_jk.conf`

   Edit `/opt/hpws/tomcat/jk/apache2/mod_jk.conf` and add:

   ```
   JkMount /servlet ajp13
   JkMount /servlet/* ajp13
   ```

6 Add servlets to CLASSPATH
   Tomcat doesn't read the CLASSPATH environment variable so you have to do the following:

   Edit `/opt/hpws/tomcat/bin/setenv.sh`
   Below the line:
   `CATALINA_OPTS="-XdoCloseWithReadPending"`
   Add the following as a single line:

   ```
   CATALINA_OPTS="$CATALINA_OPTS -
   Dorg.apache.tomcat.apps.classpath=/my_servlets"
   ```

   Note: We add `/my_servlets` to the CLASSPATH by setting the following system property:
   `org.apache.tomcat.apps.classpath`

7. Verify
   Stop and start tomcat:

   ```
   cd /opt/hpws/tomcat/bin
   ./shutdown.sh
   ./startup.sh
   ```

   Stop and start apache:

   ```
   cd /opt/hpws/apache/bin
   ./apachectl stop
   ./apachectl start
   ```

   Enter URL in browser `yourserver.com/myapp/<your servlet>`

8. Create test environment (if previous step fails). In case you don't have your own servlets, here are steps to setup a test environment based on servlets available in Tomcat.

   Create servlet home

   ```
   mkdir /my_servlets
   ```

   Add a test servlet

   ```
   cp  /opt/hpws/tomcat/webapps/examples/WEB-INF/classes/HelloWorldExample
   /my_servlets
   ```

   Stop and start tomcat:

   ```
   cd /opt/hpws/tomcat/bin
   ./shutdown.sh
   ./startup.sh
   ```

   Stop and start apache:

   ```
   cd /opt/hpws/apache/bin
   ./apachectl stop
   ./apachectl start
   ```

Enter URL in browser:
`yourserver.com/myapp/HelloWorldExample`
See in browser:
`Hello World!`

Note:
Notice the different functions of the following:

a. `/myapp` causes Apache to connect with Tomcat.
b. `/my_servlets` is the actual location of the servlets.

If you prefer you can use the name "`/my_servlets`" for both.

c. This scenario assumes that the servlets do not need to be reloadable. In other words if you recompile the servlet you will have to restart Tomcat for the changes to be recognized. Servlets can be made reloadable by referencing them in `/opt/hpws/tomcat/webapps/ROOT/WEB-INF/web.xml` and creating individual symbolic links to the classes in `/opt/hpws/tomcat/webapps/ROOT/WEB-INF/classes`.

7.5.3 Methods to Enable Servlets in Tomcat

The class-loader-howto.html of tomcat-4.1-doc refers to configuring what classes are available and the scope of their availability. Tomcat uses some classes (in particular xml parsers) that applications may want to replace with other classes. In order to avoid conflicts Tomcat separates Tomcat-required classes from application-required classes. To accomplish this your CLASSPATH environment variable is ignored by Tomcat.

The following summarizes ways to make your classes available to Tomcat and to each other.

Enabling Classes Based on Scope of Class Availability
A) Classes that are used in a particular web application.
*.jar files
`/opt/hpws/tomcat/webapps/<app name>/WEB-INF/lib`
*.class files
`/opt/hpws/tomcat/webapps/<app name>/WEB-INF/classes`

B) Classes that are shared by all web applications.
*.jar files
Place the jar files in the following directory:
`/opt/hpws/tomcat/shared/lib`
*.class files
`/opt/hpws/tomcat/shared/classes`

7.6 Other Migration Tasks

See the following sub-sections for additional migration tasks:

- Migrating the Selection of Server Type
- Migrating Webmin

7.6.1 Migrating the Selection of Server Type

In HP Apache 1.3.x:

In 1.3.x, the `ServerType` directive could be set to one of the two values in the `/opt/apache/conf/httpd.conf file`:

```
   ServerType inetd
OR
   ServerType standalone
```

In HP-UX Apache-based Web Server:

In Apache 2.x, the `ServerType` directive has been removed and server type is selected through the choice of an MPM. HP-UX Apache-based Web Server, integrates the worker MPM, which supports only one behavior, that of `standalone`. This behavior does not require any directive. There is currently no MPM designed by ASF to be launched by `inetd`.

7.6.2 Migrating Webmin

New functionality has been added to HP-UX Webmin-based Admin, the GUI interface that enables you to administer the HP-UX Web Server Suite.

Starting HP-UX Webmin-based Admin:

To Start Webmin on HP-UX Web Server Suite, perform the following steps.

Log in as root, enter

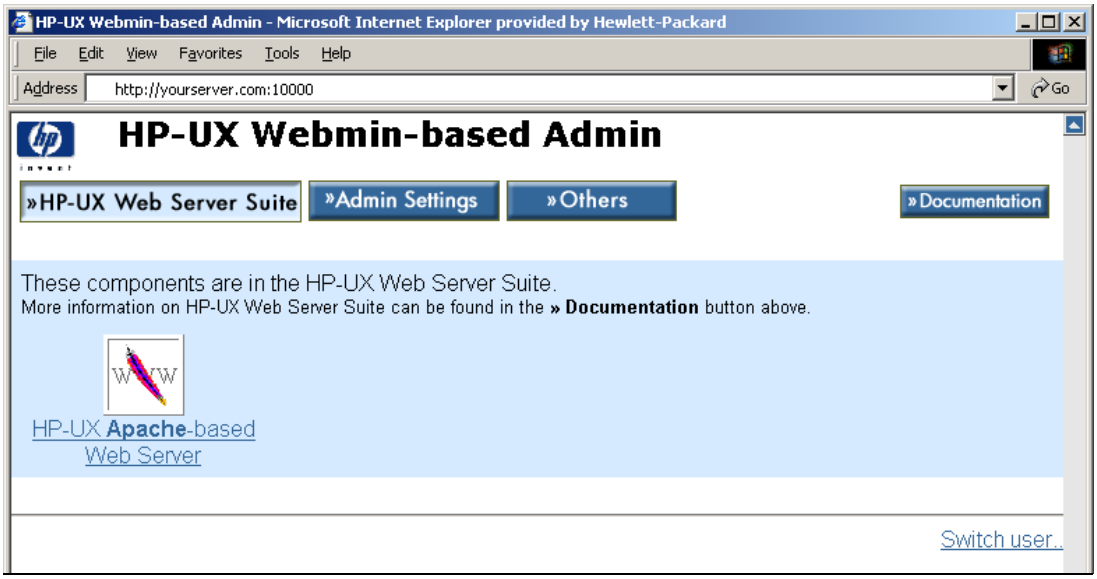
```
/opt/hpws/webmin/webmin-init start
```

Point the browser to `http://yourserver.com:10000`

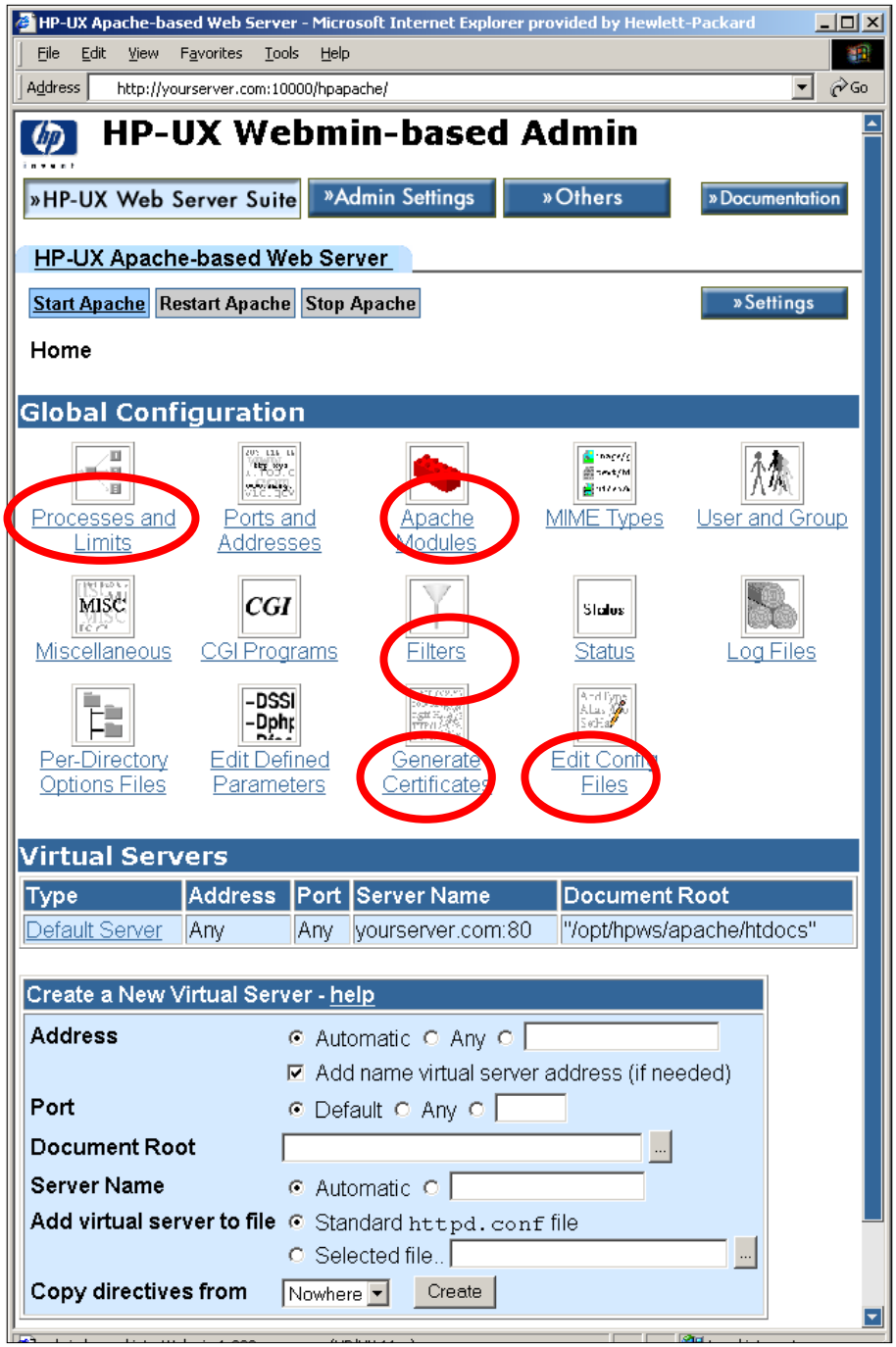
Login : `admin`
Default Password : `hp.com`

To change the password, see
`/opt/hpws/hp_docs/webmin/webmin.admin.guide`

For more information see:
`/opt/hpws/hp_docs/webmin/webmin.admin.guide`



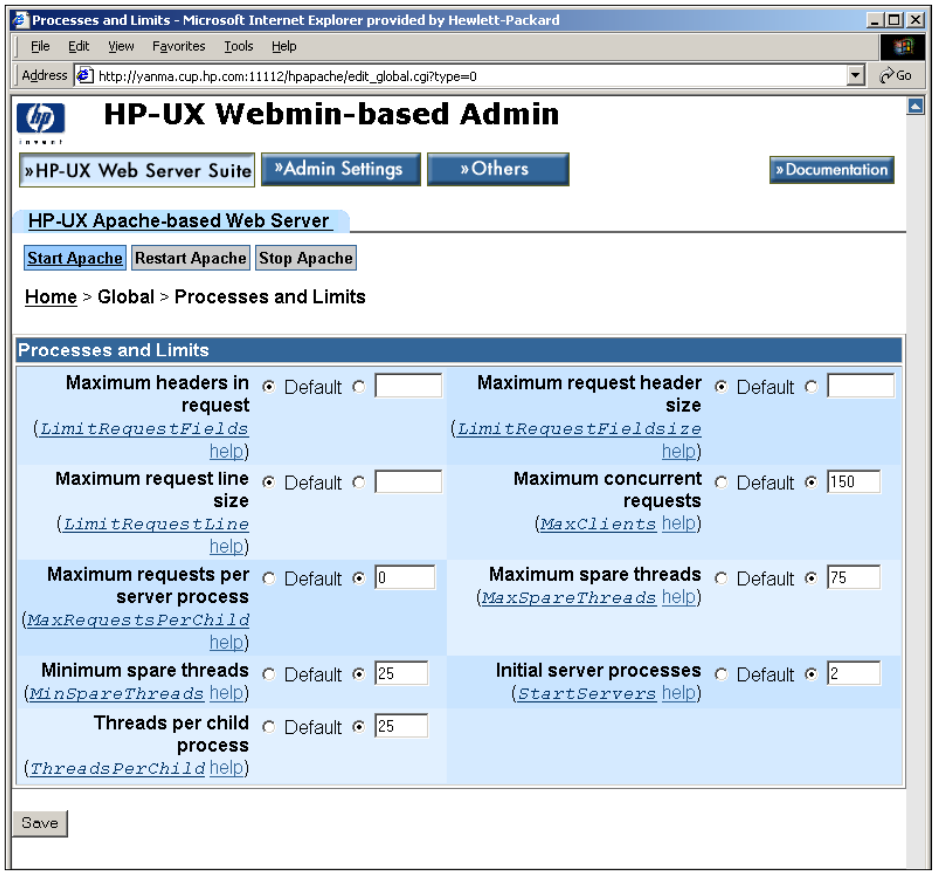
7.6.2.1 Changes to “HP-Apache-based Web Server” screen in HP-UX Webmin-based Admin:



Once you bring up HP-UX Webmin-based Admin, to get to the HP-UX Apache-based Web Server page, click on the “`HP-UX Apache-based Web Server`” icon. This will take you to the “`Default/Global Server`” page as shown following.

The major areas of new or changed Webmin functionality in HP-UX Web Server Suite have been circled.

7.6.2.2 Changed “Processes and Limits” screen in HP-UX Webmin-based Admin



The HP-UX Apache-based Web Server screen “Processes and Limits” now displays the directives corresponding to the MPM worker module (as shown in the table header).

See Migrating Process Handling for more detailed information about the worker directives.

7.6.2.3 Changed “Apache Modules” screen in HP-UX Webmin-based Admin:

The “Apache Modules” screen is grouped into five categories:

1) Statically linked-in modules
2) Standard Apache modules (DSO)
3) Standard HP-UX Apache Add-on modules (DSO)
4) Other
5) Add new modules

The first section contains modules that are always loaded and are not configurable.

The modules in the second section are available in the standard Apache distribution.

The third set of modules are not in Apache’s standard distribution, but have been included in HP-UX Apache-based Web Server. These modules include auth_ldap, mod_jk, mod_perl and mod_php4.

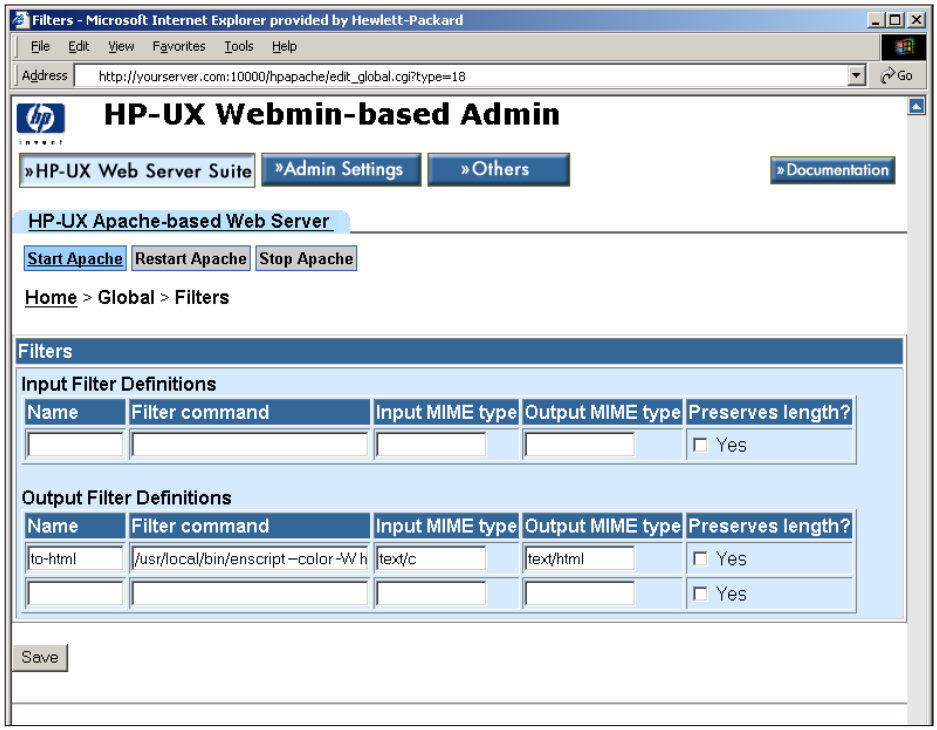
The fourth section, “Other” is blank in this example, but will contain any other modules the user adds that are not part of the HP-UX Apache-based Web Server distribution. For example if the user adds their personal module “mod_personal”, it will show up in this section.

The final section is where users can add their own modules. These input boxes require the Module Name and the Location of the Module. This is uses the LoadModule directive. For example, “foo_module” and “modules/mod_foo” would be the two required fields.

Note for mod_ssl and auth_ldap:
Since the configuration directives for mod_ssl and auth_ldap are not in the standard configuration file (conf/httpd.conf), these two modules are enabled differently. Instead, the directive “Include path/to/config_file” is uncommented to load the module. This means that when mod_ssl is enabled, the directives in the “conf/ssl.conf” file are automatically included. Likewise, for auth_ldap, the directives in the “conf/ldap.conf” file are automatically included.

Note for mod_ssl:
If one makes a change to the settings Apache when the mod_ssl module is loaded, one has to stop and then start Apache for these settings to take effect. If mod_ssl is not loaded a simple restart should work.

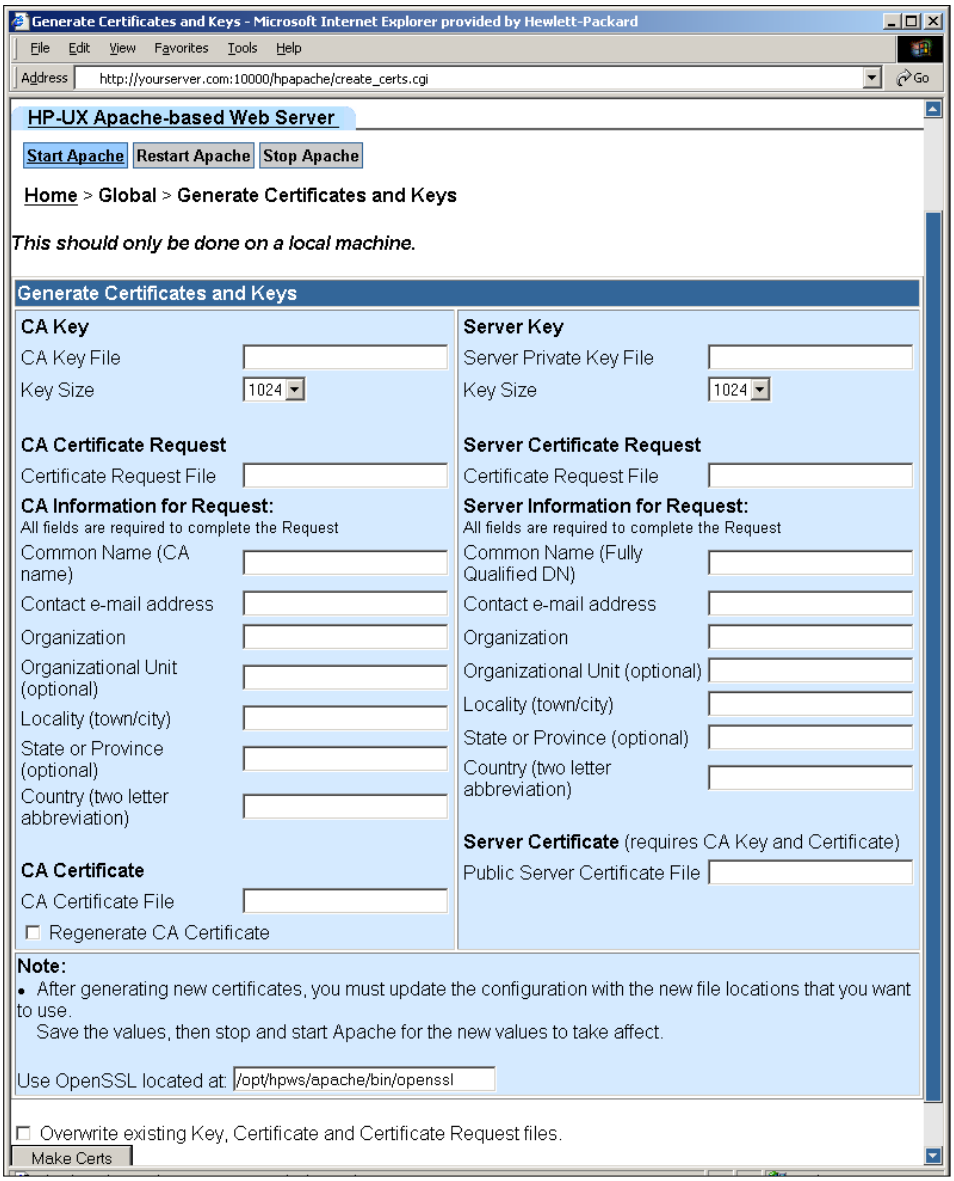
7.6.2.4 New “Filters” screen in HP-UX Webmin-based Admin:



The HP-UX Apache-based Web Server screen “Filters” allows configuration of filter directives.

See Writing 2.x Modules - Bibliography for references on filters.

7.6.2.5 New “Generating Certificates and Keys” in HP-UX Webmin-based Admin



This screen allows you to generate keys and certificates for your server or for your own self signed Certificate Authority (CA). You can use a previously generated CA certificate and key to generate new Server certificates or Keys.

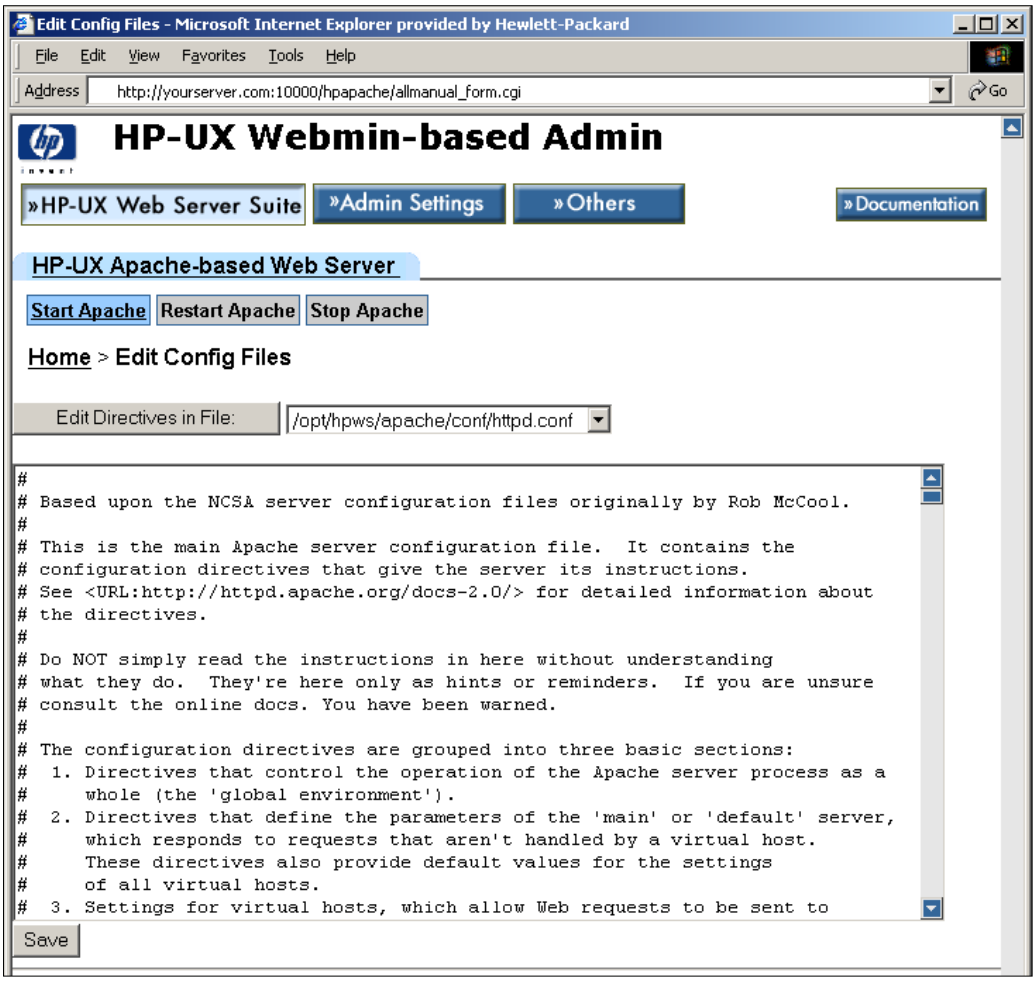
To generate only a CA Key, specify the “CA Key File” and “Key Size” and click “Make Certs” at the bottom of the screen. If there is already a file with the same name at that location, and you wish to overwrite it, check the box labeled “Overwrite existing Key, Certificate and Certificate Request files.”

Creating a CA Certificate Request depends on the CA Key, so both file locations must be specified, and the Information for the Request must be filled out as well. Organizational Unit and State or Province may remain blank.

Creating a CA Certificate depends on both the CA Key and Certificate Request. To ensure that the user doesn’t accidentally overwrite the CA Certificate, an extra box must be checked to “(Re)generate CA Certificate.”

Generating a Server Key, Certificate and Certificate Request have similar dependencies upon each other. The Server Certificate has an additional dependency on the CA Key and CA Certificate.

7.6.2.6 New “Edit Config Files” screen in HP-UX Webmin-based Admin:



This new screen allows users to directly edit and save configuration files, including the httpd.conf file. Any “Included” files can be selected through the dropdown menu. This is a new feature as of Webmin 1.020

7.7 Migrating the Printing of Virtual Host Configuration

The httpd command line option `-S`, which was used for printing the virtual host configuration has been replaced by `-t -D DUMP_VHOSTS`.

7.8 Writing 2.x Modules

The main semantic change is that operations defined in the module configuration structure and the handler array are now defined in the register_hooks function.

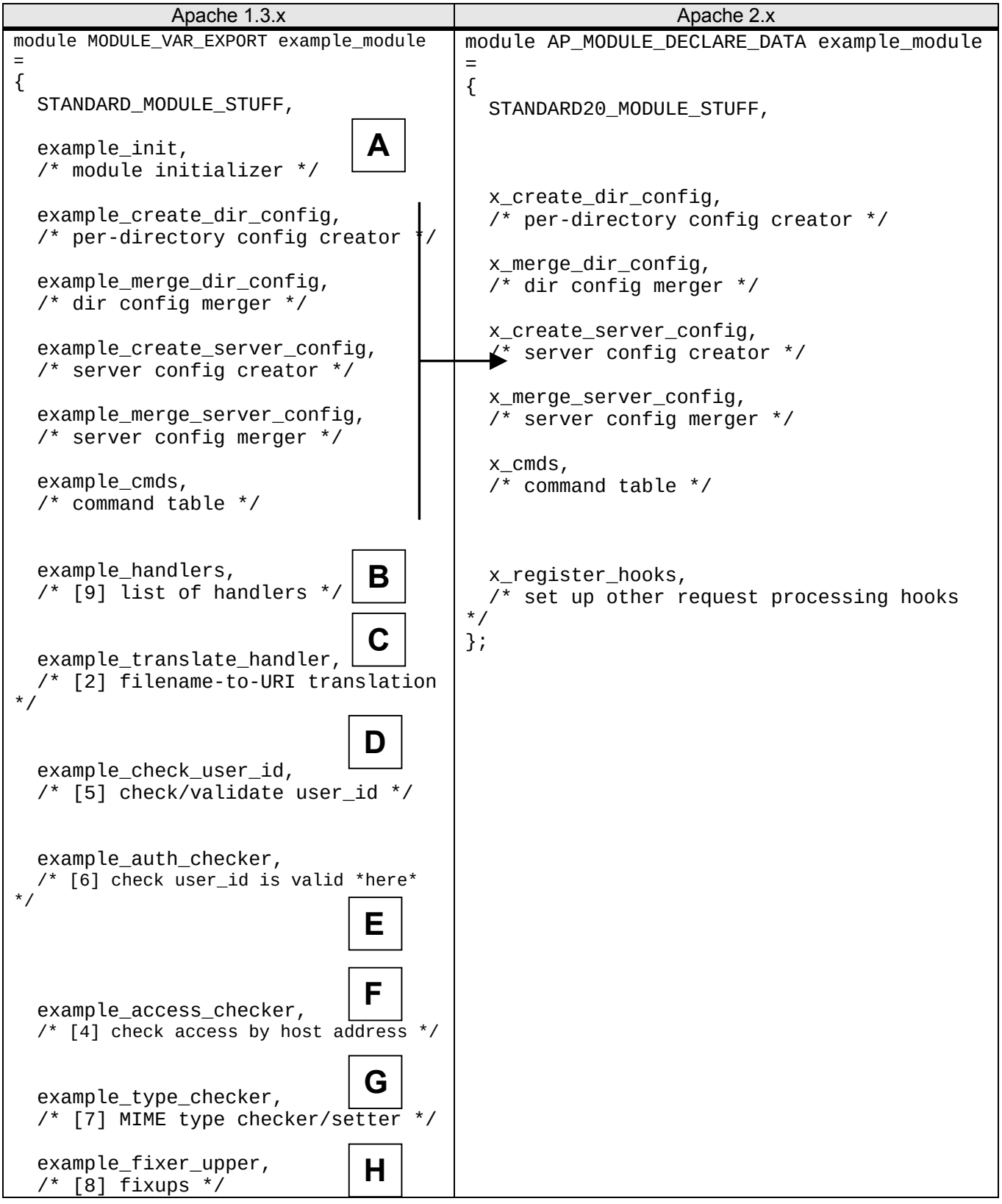
The following code is based on `mod_example.c` versions 1.3.x and 2.x. The 2.x version can be found at: `/opt/hpws/apache/build/examples/mod_example.c`.
See `/opt/hpws/apache/build/examples/README` for instructions on building `mod_example`.

7.8.1 Module Configuration

The “module definition for configuration” has changed dramatically. The register_hooks function is now used to setup options that were previously defined in the configuration module.

In the "Module definition for configuration" if a particular callback is not needed, replace its routine name below with NULL. For Apache 1.3 the number in brackets indicates the order in which the routine is called during request processing. A specific module will only use those hooks that it finds necessary.

Module definition for configuration



```
  example_logger,
  /* [10] logger */

  example_header_parser,
  /* [3] header parser */

  example_child_init,
  /* process initializer */

  example_child_exit,
  /* process exit/cleanup */

  example_post_read_request
  /* [1] post read_request handling
*/
};
```

I J K L M

There is no corresponding hook in 2.x. You would parse the bucket brigade instead. For information on bucket brigades see www.onlamp.com/pub/a/apache/2001/09/20/apache 2.html

For module initialization hook to pre_ or post_ config as necessary.

```
static void x_register_hooks(apr_pool_t *p)
{
    ap_hook_pre_config(x_pre_config,
                       NULL,
                       NULL,
                       APR_HOOK_MIDDLE);

    ap_hook_post_config(x_post_config,
                        NULL,
                        NULL,
                        APR_HOOK_MIDDLE);

    ap_hook_open_logs(x_open_logs,
                      NULL,
                      NULL,
                      APR_HOOK_MIDDLE);

    ap_hook_child_init(x_child_init,
                       NULL,
                       NULL,
                       APR_HOOK_MIDDLE);


    ap_hook_handler(x_handler,
                    NULL,
                    NULL,
                    APR_HOOK_MIDDLE);

    ap_hook_quick_handler(x_quick_handler,
                          NULL,
                          NULL,
                          APR_HOOK_MIDDLE);

    ap_hook_pre_connection(x_pre_connection,
                           NULL,
                           NULL,
                           APR_HOOK_MIDDLE);

  /* This module doesn't have a process
```

A A B

```
   connection phase, but I am leaving
   the code in, in-case somebody wants to
   add one. */
/* ap_hook_process_connection
                (x_fixer_upper,
                 NULL,
                 NULL,
                 APR_HOOK_MIDDLE); */


ap_hook_post_read_request(x_post_read_request,
          M               NULL,
                          NULL,
                          APR_HOOK_MIDDLE);

ap_hook_log_transaction(x_logger,
          I                NULL,
                           NULL,
                           APR_HOOK_MIDDLE);

ap_hook_http_method(x_http_method,
                    NULL,
                    NULL,
                    APR_HOOK_MIDDLE);

ap_hook_default_port(x_default_port,
                     NULL,
                     NULL,
                     APR_HOOK_MIDDLE);

ap_hook_translate_name(x_translate_handler,
          C          NULL,
                     NULL,
                     APR_HOOK_MIDDLE);

ap_hook_check_user_id(x_check_user_id,
          D               NULL,
                          NULL,
                          APR_HOOK_MIDDLE);

ap_hook_fixups(x_fixer_upper,
          H          NULL,
                     NULL,
                     APR_HOOK_MIDDLE);

  ap_hook_type_checker(x_type_checker,
          G               NULL,
                          NULL,
                          APR_HOOK_MIDDLE);

ap_hook_access_checker(x_access_checker,
          F               NULL,
                          NULL,
                          APR_HOOK_MIDDLE);



ap_hook_auth_checker(x_auth_checker,
                    NULL,
          E         NULL,
```

	APR_HOOK_MIDDLE); ap_hook_insert_filter(x_insert_filter, NULL, NULL, APR_HOOK_MIDDLE); }

7.8.2 Handlers

The 1.3.x handlers array is now a series of ap_hook_handler calls in the register_hooks function that we saw previously.

List of content handlers available from this module

Apache 1.3.x	Apache 2.x

static const handler_rec example_handlers[]
=
{
{"example-handler", example_handler},
{NULL}
};
	static const handler_rec x_handlers[] =
	{
	{"example-handler", x_handler},
	{NULL}
	};
	
	static void register_hooks(apr_pool_t *p)
	{
	...
	ap_hook_handler(x_handler,
	B
	NULL,
	NULL,
	APR_HOOK_MIDDLE);
	...
	}

7.8.3 Directives List

Notice that in the example NO_ARGS becomes the `AP_INIT_NO_ARGS macro.`

List of directives specific to our module

Apache 1.3.x	Apache 2.x

`static const command_rec example_cmds[] =`
`{`
`  {`
`   "Example",   /* directive name */`
`   cmd_example, /* config action routine */`
`   NULL,  /* argument to include in call */`
`   OR_OPTIONS,  /* where available */`
`   NO_ARGS,     /* arguments */`
`   "Example directive - no arguments"`
`                /* directive description */`
`  },`
`  {NULL}`
`};`
	`static const command_rec x_cmds[] =`
	`{`
	`  AP_INIT_NO_ARGS(`
	`    "Example",   /* directive name */`
	`    cmd_example, /* config action routine */`
	`    NULL,        /* arg to include in call */`
	`    OR_OPTIONS,  /* where available */`
	`    "Example directive - no arguments"`
	`                 /* directive description */`
	`  ),`
	`  {NULL}`
	`};`

7.8.4 Initialization and Exiting

Notice that you now set a cleanup register.

Child process init handling

Apache 1.3.x	Apache 2.x

`static void`
`  example_child_init(server_rec *s,`
`                    pool *p)`
`  {`
`    char *note;`
`    char *sname = s->server_hostname;`
 	
`    /* Set up any module cells that ought`
`       to be initialised. */`
 	
`    setup_module_cells();`
 	
`    /* The arbitrary text we add to our`
`       trace entry indicates for which`
`       server we're being called. */`
 	
`    sname = (sname != NULL) ? sname : "";`
`    note = ap_pstrcat(p,`
`              "example_child_init(",`
`              sname,`
`              ")",`
`              NULL);`
`    trace_add(s, NULL, NULL, note);`
`}`
	`static void`
	`  x_child_init(apr_pool_t *p,`
	`               server_rec *s)`
	`  {`
	`    char *note;`
	`    char *sname = s->server_hostname;`
	
	`    /* Set up any module cells that ought`
	`       to be initialised. */`
	
	`    setup_module_cells();`
	
	`    /* The arbitrary text we add to our trace`
	`       entry indicates for which server we're`
	`       being called. */`
	
	`    sname = (sname != NULL) ? sname : "";`
	`    note = apr_pstrcat(p,`
	`                       "x_child_init(",`
	`                       sname,`
	`                       ")", NULL);`
	`    trace_add(s, NULL, NULL, note);`
	
	`    apr_pool_cleanup_register(p,`
	`                              s,`
	`                              x_child_exit,`
	`                              x_child_exit);`
	`}`

Child process exit handling

Apache 1.3.x	Apache 2.x

static void
  example_child_exit(server_rec *s,
  pool *p)
{
 	
  char *note;
  char *sname = s->server_hostname;
 	
  /* The arbitrary text we add to our
  trace entry indicates for which
  server we're being called. */
 	
  sname = (sname != NULL) ? sname : "";
  note = ap_pstrcat(p,
  "example_child_exit(",
  sname,
  ")",
  NULL);
  trace_add(s, NULL, NULL, note);
}
	static apr_status_t
	  x_child_exit(void *data)
	
	{
	  char *note;
	  server_rec *s = data;
	  char *sname = s->server_hostname;
	
	  /* The arbitrary text we add to our
	  trace entry indicates for which
	  server we're being called. */
	
	  sname = (sname != NULL) ? sname : "";
	  note = apr_pstrcat(s->process->pool,
	  "x_child_exit(",
	  sname,
	  ")",
	  NULL);
	  trace_add(s, NULL, NULL, note);
	  return APR_SUCCESS;
	}

7.8.5 Functions and Types

Many "ap_" functions have new "apr_" versions. "apr" stands for "Apache Portable Runtime". "apu" stands for "Apache Portable Runtime Utilities". The following tables list common changes. For more info on changes look at ap_compat.h, apu_compat.h, and apr_ compat.h. These compatibility headers allow old function names to work in the new environment.

Change from old to new functions

Apache 1.3.x	Apache 2.x

	apr_pool_cleanup_register(apr_pool_t* p,
	  server_rec* s,
	  my_child_exit,
	  my_child_exit);

/* Allocate space from pool supplied*/
tocfg *cfg;
cfg = (tocfg *) ap_pcalloc(p,
  sizeof(tocfg)
  );
	/* Allocate space from pool supplied*/
	tocfg *cfg;
	cfg = (tocfg *) apr_pcalloc(p,
	  sizeof(tocfg)
	  );

ap_palloc(...);	/* No change to signature */ apr_palloc(...);

/* Allocate memory for pool use */
toMyPool = ap_make_sub_pool(r->pool);
	/* Allocate memory for pool use */
	apr_make_sub_pool(&toMyPool,
	  r->pool,
	  NULL);

Apache 1.3.x	Apache 2.x

`ap_snprintf( valueString,`
`sizeof(valueString),`
`"%s\n",`
`value);`
	`apr_snprintf( valueString,`
	`sizeof(valueString),`
	`"%s\n",`
	`value);`

`ap_table_do( ... );`	`/* Second argument has changed */` `apr_table_do( ... );`
`ap_pool_destroy( ... );`	`/* No change to signature */` `apr_pool_destroy( ... );`
`ap_table_set( ... );`	`/* No change to signature */` `apr_table_set( ... );`
`ap_table_set( ... );`	`/* No change to signature */` `apr_table_set( ... );`
`ap_make_array( ... );`	`/* No change to signature */` `apr_array_make( ... );`
`ap_push_array( ... );`	`/* No change to signature */` `apr_array_push( ... );`
`ap_pstrdup( ... );`	`/* No change to signature */` `apr_pstrdup( ... );`
`ap_send_http_header( r );`	`/* This function was removed in 2.0. It is defined in ap_compat.h to do nothing */` `ap_send_http_header( r );`

Change in existing functions

Apache 1.3.x	Apache 2.x

`ap_log_error( APLOG_MARK,`
`APLOG_DEBUG,`
`r->server,`
`"error text")`
	`ap_log_error( APLOG_MARK,`
	`APLOG_DEBUG,`
	`0,`
	`r->server,`
	`"error text")`

Change from old to new types

Apache 1.3.x	Apache 2.x
`pool`	`apr_pool_t`
`array_header`	`apr_array_header_t`

Change from 2.0.32 to 2.0.39 functions

Apache 2.0.32	Apache 2.0.39
`apr_lock_create  ( ... );`	`apr_thread_mutex_create  ( ... );`
`apr_lock_acquire ( ... );`	`apr_thread_mutex_lock    ( ... );`
`apr_lock_release ( ... );`	`apr_thread_mutex_unlock  ( ... );`
`apr_lock_destroy ( ... );`	`apr_thread_mutex_destroy ( ... );`

7.8.6 Header files

Note: You can use ap_compat.h in HP-UX Apache-based Web Server to allow compatability with 1.3.x. It maps function names and signatures to the HP-UX Apache-based Web Server.

Header files

Apache 1.3.x	Apache 2.x

`#include "httpd.h"`
`#include "http_config.h"`
`#include "http_core.h"`
`#include "http_log.h"`
`#include "http_main.h"`
`#include "http_protocol.h"`
`#include "util_script.h"`
 	
`#include <stdio.h>`
	`#include "httpd.h"`
	`#include "http_config.h"`
	`#include "http_core.h"`
	`#include "http_log.h"`
	`#include "http_main.h"`
	`#include "http_protocol.h"`
	`#include "http_request.h"`
	`#include "util_script.h"`
	`#include "http_connection.h"`
	
	`#include "apr_strings.h"`
	
	`#include <stdio.h>`

7.8.7 Bibliography

For information on writing HP-UX Web Server Suite modules see:
httpd.apache.org/docs-2.0/developer/
www.onlamp.com/pub/a/apache/2001/09/27/apache_2.html

For information on HP-UX Web Server Suite filters see:
Official Apache Software Foundation document
httpd.apache.org/docs-2.0/filter.html
Overview, "Writing Filters for Apache 2.0", 08/23/2001
www.onlamp.com/pub/a/apache/2001/08/23/apache_2.html
"Writing Apache 2.0 Output Filters", 09/13/2001
www.onlamp.com/pub/a/apache/2001/09/13/apache_2.html
"Writing Input Filters for Apache 2.0", 09/20/2001
www.onlamp.com/pub/a/apache/2001/09/20/apache_2.html

For information on writing Apache 1.3.x modules see:
Writing Apache Modules with Perl and C, by Stein and MacEachern, O'Reilly and Associates Inc.

7.9 Miscellaneous Changes

- Graceful restarts of the server are now executed by signaling the parent process with WINCH rather than USR1.
- The httpd command line option -X still exists but most MPMs allow the same functionality to be requested by using the -D ONE_PROCESS command line option. In addition, a -D NO_DETACH command line option is available.

8 Tuning HP-UX Web Server Suite Performance

The following are suggestions only and may not apply to your specific situation.
See the FAQ at www.hp.com/products1/unix/webservers/apache/faqs/index.html for the latest information on improving performance.

8.1 Basic HTTP

For the basic http server do the following:

1. Disable indexing of directories in "httpd.conf".
   With the `Options` directive don't use `Indexes` or `All` which enable index generation. Indexes are dynamically generated for every access. It is much better to create an `index.html` file instead so that the server does not need to dynamically generate text.
2. Logging
   Disable logging as much as possible. If necessary write each log to their own file system using different high speed disks. This will decrease delays caused by disk scheduling.
3. `KeepAlive on`
   With this directive set to on Apache will attempt to maintain a rapid dialog with the client. `KeepAlive on` does not mean that the connection will be maintained too long. The `KeepAliveTimeout` and `MaxKeepAliveRequests` directives restrict clients from excessive, persistent connection times.
4. `HostNameLookups off`
   With this directive set to `off` Apache will log IP addresses instead of host names. Even though Apache caches DNS lookups, this can be very time consuming.
5. `Options FollowSymLinks`
   not
   `Options FollowSymLinksIfOwnerMatch`
   `FollowSymLinksIfOwnerMatch` causes Apache to check the entire path for symbolic links and check that the ownership is the same as the server or virtual host. This can be time consuming. The `FollowSymLinks` directive is less secure because it does no checking. If neither `FollowSymLinks` nor `FollowSymLinksIfOwnerMatch` directives are used then Apache will not follow symbolic links. This slows Apache because it has to check for symbolic links so it can avoid them but this situation is the most secure.
6. For local directory URL's put a forward slash, "/", at the end. That way the server can avoid needless file searches.
7. Deny access to well mannered spiders or web crawlers with a "robots.txt" file under the document root. The file has "User-Agent" and "Disallow" directives that limit access to specified URLs by the spider. This will eliminate some unnecessary web traffic.
8. Verify that there are no performance bottlenecks between servers and databases.
9. For images:
   Specify height and width.
   Keep size small.
   Minimize number of images.
   Avoid animations.
   Don't use macromedia flash on home pages.
10. Don't combine large and small images into same file.
    Put alternate large images into a separate file since many users will not choose to view the large image.

8.2 SSL

By default HP-UX Apache-based Web Server Suite SSL is tuned to maximize performance. See `/opt/hpws/apache/hpws_docs/ssl.admin.guide` for a description.

8.3 PHP

For PHP some suggestions are:

1. Use only one echo or print statement in the script. Use array or concatenation to compose the html then send with only one echo or print.
2. Use the set of output buffer (ob_*) functions newly available in PHP4. For example ob_start.

3. Use an accelerator. Search the web for "PHP Accelerator" and you will find several accelerators, some of which are free.

8.4 Tomcat

For the latest Tomcat suggestions see:
`/opt/hpws/hp_docs/tomcat/tomcat.admin.guide`

1. Avoid JSP's.
   Instead use Perl, C, C++, or Java modules. JSPs automatically create and compile servlets which they then run. Typically the JSP will only create the servlet the first time it is accessed. Still the JSP has to call the servlet so there is some delay.

The following is based on:
jakarta.apache.org/tomcat/tomcat-3.3-doc/tomcat-ug.html

which contains the complete Tomcat User Guide.

2. Modify and Customize the Batch Files

As stated in the previous sections, the startup scripts are here for your convenience. Yet, sometimes the scripts that are needed for deployment should be modified:

To set resource limits such as maximum number of descriptors.

To add new PATH/LD_LIBRARY_PATH entries (for example, JDBC drivers DLLs).

To modify the JVM command line settings.

Make sure that you are using a specific JVM (out of the two or three JVMs installed on your machine).

To switch user from root to some other user using the "su" UNIX command.

Some of these changes can be done without explicit changes to the basic scripts; for example, the tomcat script can use an environment variable named TOMCAT_OPTS to set extra command line parameters to the JVM (such as memory setting etc.).

On UNIX you can also create a file named ".tomcatrc" in your home directory and Tomcat will take environment information such as PATH, JAVA_HOME, TOMCAT_HOME and TOMCAT_INSTALL from this file. On NT however (and also on UNIX when the modifications are for something such as the JVM command line) you are forced to rewrite some of the startup script.

3. Modify the Default JVM Settings

The default JVM settings in the tomcat script are very naive; everything is left for defaults. There are a few things that you should consider to improve your Tomcat performance:

Modify your JVM memory configuration. Normally the JVM allocates an initial size for the Java heap and that's it, if you need more then this amount of memory you will not get it.

Nevertheless, in loaded sites, giving more memory to the JVM improves Tomcat's performance. You should use command line parameters such as -Xms/-Xmx/-ms/-mx to set the minimum/maximum size of the Java heap (and check to see if the performance was improved).

Modify your JVM threading configuration. The SUN JDK1.2.2 for Linux comes with support for both, green and native threads. In general native threads are known to provide improved performance for I/O bound applications, green threads on the other hand put less stress on the machine. You should experiment with these two threading models and see which model is better for your site (in general, native threads are better).

Select the best JVM for the task. If your application does not require a specific JDK functionality, you should benchmark the two JVMs and select the better one.

4. Disable Servlet Auto-Reloading

Servlet auto-reloading is really useful for development time. However it is very expensive (in performance degradation terms) and may put your application in strange conflicts when classes that were loaded by a certain classloader cannot cooperate with classes loaded by the current classloader.

So, unless you have a real need for class reloading during your deployment you should turn off the reloadable flag in your contexts. You can disable reloading globally by removing the ReloaderInterceptor found in the server.xml file.

9 Verifying the Migration

After migration check for errors:

- Check `/opt/hpws/apache/logs/error_log` for error messages
- If errors occurred, determine the cause and adjust configuration files.
- After manual updates, start HP-UX Web Server Suite. As root type:

  ```
  /<server_root>/bin/apachectl start
  ```

- Verify servlet and JSP migration by checking a known servlet and a known JSP page.
- Run customer applications and access customer files.
- Run previous performance and load tests. Tune HP Apache directives and HP-UX kernel parameters if the migrated web server is not meeting expected performance.

10 Appendix

10.1 Directive Changes by Name

The table includes all removed, changed, or added directives.

	Status	Module	How to migrate from 1.3.x
__END__	Removed	mod_perl	
=back	Removed	mod_perl	
=cut	Removed	mod_perl	
=pod	Removed	mod_perl	
AcceptMutex	Added	core	
AcceptPathInfo	Added	core	
AccessConfig	Removed	core	Migrating Access to Configuration Files
AddInputFilter	Added	mod_mime	
AddModule	Removed	core	Migrating Module Loading
AddOutputFilter	Added	mod_mime	
AddOutputFilterByType	Added	core	
AgentLog	Removed	mod_log_agent	Migrating Logging
ApJServAction	Removed	mod_jserv	
ApJServBalance	Removed	mod_jserv	
ApJServDefaultHost	Removed	mod_jserv	
ApJServDefaultPort	Removed	mod_jserv	
ApJServDefaultProtocol	Removed	mod_jserv	
ApJServEnvVar	Removed	mod_jserv	
ApJServHost	Removed	mod_jserv	
ApJServLogFile	Removed	mod_jserv	
ApJServLogLevel	Removed	mod_jserv	
ApJServManual	Removed	mod_jserv	
ApJServMount	Removed	mod_jserv	
ApJServMountCopy	Removed	mod_jserv	
ApJServProperties	Removed	mod_jserv	
ApJServProtocolParameter	Removed	mod_jserv	
ApJServRetryAttempts	Removed	mod_jserv	
ApJServRoute	Removed	mod_jserv	
ApJServSecretKey	Removed	mod_jserv	
ApJServShmFile	Removed	mod_jserv	
ApJServVMInterval	Removed	mod_jserv	
ApJServVMTimeout	Removed	mod_jserv	
AuthDBMType	Added	mod_auth_dbm	
AuthDigestShmemSize	Added	mod_auth_digest	
AuthLDAPAuthoritative	Added	auth_ldap	
AuthLDAPBindDN	Added	auth_ldap	
AuthLDAPBindPassword	Added	auth_ldap	
AuthLDAPCacheCompareOps	Added	auth_ldap	
AuthLDAPCacheSize	Added	auth_ldap	
AuthLDAPCacheTTL	Added	auth_ldap	
AuthLDAPCertDBPath	Added	auth_ldap	
AuthLDAPCompareDNOnServer	Added	auth_ldap	
AuthLDAPDereferenceAliases	Added	auth_ldap	
AuthLDAPEnabled	Added	auth_ldap	
AuthLDAPGroupAttribute	Added	auth_ldap	
AuthLDAPGroupAttributeIsDN	Added	auth_ldap	
AuthLDAPOpCacheSize	Added	auth_ldap	
AuthLDAPOpCacheTTL	Added	auth_ldap	

Directive	Status	Module	How to migrate from 1.3.x
AuthLDAPRemoteUserIsDN	Added	auth_ldap	
AuthLDAPStartTLS	Added	auth_ldap	
AuthLDAPURL	Added	auth_ldap	
BindAddress	Removed	core	
BufferedLogs	Added	mod_log_config	experimental
CacheDefaultExpireMin	Added	mod_cache	
CacheDirLength	Added	mod_disk_cache	
CacheDirLength	Removed	mod_proxy	
CacheDirLevels	Added	mod_disk_cache	
CacheDirLevels	Removed	mod_proxy	
CacheDisable	Added	mod_cache	
CacheEnable	Added	mod_cache	
CacheExpiryCheck	Added	mod_disk_cache	
CacheFile	Added	mod_file_cache	
CacheGcClean	Added	mod_disk_cache	
CacheGcDaily	Added	mod_disk_cache	
CacheGcInterval	Added	mod_disk_cache	
CacheGcInterval	Removed	mod_proxy	
CacheGcMemUsage	Added	mod_disk_cache	
CacheGcUnused	Added	mod_disk_cache	
CacheIgnoreCacheControl	Added	mod_cache	
CacheIgnoreNoLastMod	Added	mod_cache	
CacheMaxFileSize	Added	mod_disk_cache	
CacheMaxStreamingBuffer	Added	mod_cache	
CacheMinFileSize	Added	mod_disk_cache	
CacheNegotiatedDocs	Changed	mod_negotiation	How to Migrate Documents Cached by Proxy Servers
CacheRoot	Added	mod_disk_cache	
CacheRoot	Removed	mod_proxy	
CacheSize	Added	mod_disk_cache	
CacheSize	Removed	mod_proxy	
CacheTimeMargin	Added	mod_disk_cache	
CaseFilter	Added	mod_casefilter	
CaseFilterIn	Added	mod_casefilterin	
CharsetDefault	Added	mod_charset_lite	
CharsetOptions	Added	mod_charset_lite	
CharsetSourceEnc	Added	mod_charset_lite	
ClearModuleList	Removed	core	Migrating Module Loading
CookieDomain	Added	mod_usertrack	
CookieStyle	Added	mod_usertrack	
DAV	Added	mod_dav	
DAVDepthInfinity	Added	mod_dav	
DAVLockDB	Added	mod_dav_fs	
DAVMinTimeout	Added	mod_dav	
Define	Removed	mod_define	
DeflateBufferSize	Added	mod_deflate	
DeflateFilterNote	Added	mod_deflate	
DeflateMemLevel	Added	mod_deflate	
DeflateWindowSize	Added	mod_deflate	
ErrorDocument	Changed	core	Migrating the Printing of Error Messages
ExcessRequestsPerChild	Removed	core	
ExtFilterDefine	Added	mod_charset_lite	
ExtFilterDefine	Added	mod_ext_filter	
ExtFilterOptions	Added	mod_charset_lite	
ExtFilterOptions	Added	mod_ext_filter	

Directive	Status	Module	How to migrate from 1.3.x
FancyIndexing	Removed	mod_authindex	
FileETag	Added	core	
ForceLanguagePriority	Added	mod_negotiation	
Header echo	Added	mod_headers	
HeaderName *filename*	Changed	mod_authindex	
Include	Changed	core	Migrating Access to Configuration Files
IndexOptions FancyIndexing	Added	mod_authindex	
IndexOptions HTMLTable	Added	mod_authindex	
IndexOptions IgnoreClient	Added	mod_authindex	
IndexOptions SuppressIcon	Added	mod_authindex	
IndexOptions SuppressRules	Added	mod_authindex	
IndexOptions VersionSort	Added	mod_authindex	
JkAutoMount	Added	mod_jk	
JkCERTSIndicator	Added	mod_jk	
JkCIPHERIndicator	Added	mod_jk	
JkEnvVar	Added	mod_jk	
JkExtractSSL	Added	mod_jk	
JkHTTPSIndicator	Added	mod_jk	
JkKEYSIZEIndicator	Added	mod_jk	
JkLogFile	Added	mod_jk	
JkLogLevel	Added	mod_jk	
JkLogStampFormat	Added	mod_jk	
JkMount	Added	mod_jk	
JkMountCopy	Added	mod_jk	
JkOptions	Added	mod_jk	
JkSESSIONIndicator	Added	mod_jk	
JkWorker	Added	mod_jk	
JkWorkersFile	Added	mod_jk	
LimitXMLRequestBody	Added	core	
Listen	Changed	core	Supports Ipv6
LoadModule	Changed	core	Migrating Module Loading
LogFormat	Changed	mod_log_config	
MaxServers	Removed	core	
MaxSpareThreads	Added	core	
MCacheMaxObjectCount	Added	mod_mem_cache	
MCacheMaxObjectSize	Added	mod_mem_cache	
MCacheMinObjectSize	Added	mod_mem_cache	
MCacheRemovalAlgorithm	Added	mod_mem_cache	
MCacheSize	Added	mod_mem_cache	
MinSpareThreads	Added	core	
mmapfile	Added	mod_file_cache	(Directive is in ASF Apache 1.3.x but not in HP Apache 1.3.x)
ModMimeUsePathInfo	Added	mod_mime	
MultiviewsMatch	Added	mod_mime	
NameVirtualHost	Changed	core	Supports Ipv6
NoCache	Removed	mod_proxy	
<Perl >	Removed	mod_perl	Not yet implemented
PerlFreshRestart	Removed	mod_perl	
PerlInitHandler	Added	mod_perl	
PerlInterpMax	Added	mod_perl	
PerlInterpMaxRequests	Added	mod_perl	
PerlInterpMaxSpare	Added	mod_perl	
PerlInterpMinSpare	Added	mod_perl	
PerlInterpScope	Added	mod_perl	
PerlInterpStart	Added	mod_perl	

Directive	Status	Module	How to migrate from 1.3.x
PerlOpmask	Removed	mod_perl	
PerlOptions	Added	mod_perl	
PerlScript	Removed	mod_perl	
PerlSwitches	Added	mod_perl	
PerlTrace	Added	mod_perl	
PHPINIDir	Added	mod_php	
Port	Removed	core	Migrating Port Number Assignments
ProtocolEcho	Added	mod_echo	
<Proxy >	Added	mod_proxy	(A Container for other directives)
ProxyErrorOverride	Added	mod_proxy	
ProxyIOBufferSize	Added	mod_proxy	
<ProxyMatch >	Added	mod_proxy	(A Container for other directives)
ProxyMaxForwards	Added	mod_proxy	
ProxyPass	Changed	mod_proxy	
ProxyPassReversed	Changed	mod_proxy	
ProxyPreserveHost	Added	mod_proxy	
ProxyRemoteMatch	Added	mod_proxy	
ProxyTimeout	Added	mod_proxy	
RefererIgnore	Removed	mod_log_referer	Migrating Logging
RefererLog	Removed	mod_log_referer	Migrating Logging
RemoveCharset	Added	mod_mime	
RemoveInputFilter	Added	mod_mime	
RemoveLanguage	Added	mod_mime	
RemoveOutputFilter	Added	mod_mime	
RequestHeader	Added	mod_headers	
ResourceConfig	Removed	core	Migrating Access to Configuration Files
ScoreBoardFile	Added	core	
Scriptsock	Added	mod_cgid	
SendBufferSize	Removed	core	
ServerLimit	Added	core	
ServerName	Changed	core	Migrating Security (SSL)
ServersSafetyLimit	Removed	core	
ServerTokens *major*	Changed	core	New option
ServerType	Removed	core	Migrating the Selection of Server Type
SetInputFilter	Added	core	
SetOutputFilter	Added	core	
SSIEndTag	Added	mod_include	
SSIErrorMsg	Added	mod_include	
SSIStartTag	Added	mod_include	
SSITimeFormat	Added	mod_include	
SSIUndefinedEcho	Added	mod_include	
SSLLog	Removed	mod_ssl	Use ErrorLog instead
SSLLogLevel	Removed	mod_ssl	Use LogLevel instead
SuexecUserGroup	Added	mod_suexec	
ThreadLimit	Added	core	
UnsetEnv	Changed	mod_env	
<VirtualHost>	Changed	core	Supports Ipv6
VirtualScriptAlias	Changed	mod_vhost_alias	

10.2 Module Changes

The tables include all removed, changed, or added modules or directives.

10.2.1 Core

Notes	Status	How to migrate from HP Apache 1.3.x
These directives don't come from a module. They are part of Apache core.		ASF Apache 1.3.x doc on core directives. ASF Apache 2.x doc on core directives

Directive	Status	Description
<VirtualHost>	Changed	Now supports Ipv6
AcceptMutex	Added	The system mutex implementation to use for the accept mutex
AcceptPathInfo	Added	Set to on or off for PATH_INFO to be accepted by handlers, or default for the per-handler preference
AccessConfig	Removed	The filename of the access config file
AddModule	Removed	The name of a module
AddOutputFilterByType	Added	Output filter name followed by one or more content-types
BindAddress	Removed	'*', a numeric IP address, or the name of a host with a unique IP address
ClearModuleList	Removed	Not needed since modules now inform Apache of their prefered load order.
ErrorDocument	Changed	Now requires a closing " at end of description.
ExcessRequestsPerChild	Removed	Maximum number of requests a particular child serves after it is ready to die.
FileETag	Added	Specify components used to construct a file's ETag
Include	Changed	
LimitXMLRequestBody	Added	Limit (in bytes) on maximum size of an XML-based request body
Listen	Changed	Now supports Ipv6
LoadModule	Changed	Order is no longer important. The module, itself, tells Apache when it should be loaded.
MaxServers	Removed	Deprecated equivalent to MaxSpareServers
MaxSpareThreads	Added	Maximum number of idle children
MinSpareThreads	Added	Minimum number of idle children, to handle request spikes
NameVirtualHost	Changed	Now supports Ipv6
Port	Removed	A TCP port number
ResourceConfig	Removed	The filename of the resource config file
ScoreBoardFile	Added	A file for Apache to maintain runtime process management information
SendBufferSize	Removed	Send buffer size in bytes
ServerLimit	Added	Maximum value of MaxClients for this run of Apache
ServerName	Changed	Now includes the port number.
ServersSafetyLimit	Removed	Deprecated equivalent to MaxClients
ServerTokens *major*	Changed	New option prints major version only
ServerType	Removed	'inetd' or 'standalone'
SetInputFilter	Added	filter (or ; delimited list of filters) to be run on the request body
SetOutputFilter	Added	filter (or ; delimited list of filters) to be run on the request content
ThreadLimit	Added	Maximum worker threads in a server for this run of Apache

10.2.2 auth_ldap

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module provides client authorization by querying an LDAP server.	New	auth_ldap Home Page For info on LDAP servers see: openldap Home Page

Directive	Status	Description
AuthLDAPAuthoritative	Added	Set to 'off' to allow access control to be passed along to lower modules if the UserID and/or group is not known to this module

Directive	Status	Description
AuthLDAPBindDN	Added	DN to use to bind to LDAP server. If not provided, will do an anonymous bind.
AuthLDAPBindPassword	Added	Password to use to bind to LDAP server. If not provided, will do an anonymous bind will be done.
AuthLDAPCacheCompareOps	Added	Set to no to disable caching of LDAP compare operations. Defaults to yes.
AuthLDAPCacheSize	Added	Sets the maximum amount of memory in bytes that the LDAP search cache will use. Zero means no limit; -1 disables the cache. Defaults to 10KB.
AuthLDAPCacheTTL	Added	Sets the maximum time (in seconds) that an item can be cached in the LDAP
AuthLDAPCompareDNOnServer	Added	
AuthLDAPDereferenceAliases	Added	Determines how aliases are handled during a search. Can bo one of the
AuthLDAPEnabled	Added	Set to off to disable auth_ldap, even if it's
AuthLDAPGroupAttribute	Added	A list of attributes used to define group membership - defaults to
AuthLDAPGroupAttributeIsDN	Added	
AuthLDAPOpCacheSize	Added	Sets the initial size of the LDAP operation cache (for bind and compare
AuthLDAPOpCacheTTL	Added	Sets the maximum time (in seconds) that an item is cached in the LDAP
AuthLDAPRemoteUserIsDN	Added	Set to, , auth_ldap_cmds
AuthLDAPStartTLS	Added	Set to 'on' to start TLS after connecting to
AuthLDAPURL	Added	URL to define LDAP connection. This should be an RFC 2255 complaint
AuthLDAPCertDBPath	Added	Specifies the file containing Certificate Authority certificates for validating secure LDAP server certificates. This file must be the cert7.db database used by Netscape Communicator}

10.2.3 mod_auth_dbm

Module Notes	Status	How to migrate from HP Apache 1.3.x
Now supports multiple DBM-like databases using the AuthDBMType directive	Changed	ASF Apache 2.x doc on mod_auth_dbm ASF Apache 1.3.x doc on mod_auth_dbm

Directive	Status	Description
AuthDBMType	Added	what type of DBM file the user file is

10.2.4 mod_auth_digest

Module Notes	Status	How to migrate from HP Apache 1.3.x
Includes additional support for session caching across processes using shared memory. This module, which was experimental in HP Apache 1.3.x, is a standard module in 2.x. In 1.3.x this module was called mod_digest.	Changed	ASF Apache 2.x doc on mod_auth_digest ASF Apache 1.3.x doc on mod_digest

Directive	Status	Description
AuthDigestShmemSize	Added	The amount of shared memory to allocate for keeping track of clients

10.2.5 mod_autoindex

Module Notes	Status	How to migrate from HP Apache 1.3.x
Autoindex'ed directory listings can now be configured to use HTML tables for cleaner formatting, and allow finer-grained control of sorting, including version-sorting, and wildcard filtering of the directory listing.	Changed	ASF Apache 2.x doc on mod_autoindex ASF Apache 1.3.x doc on mod_autoindex

Directive	Status	
FancyIndexing	Removed	It is now an IndexOptions option.
HeaderName *filename*	Changed	Commands within *filename* have been changed to reorganize the query arguments for column sorting and introduce an entire group of new query options..
IndexOptions FancyIndexing	Added	Replaces the FancyIndexing directive
IndexOptions HTMLTable	Added	Experimental. When used with fancy indexing it constructs a simple table. This will confuse older browsers.
IndexOptions IgnoreClient	Added	Eliminates all client control over the output.
IndexOptions SuppressIcon	Added	Suppress icon in fancy indexing listings.
IndexOptions SuppressRules	Added	Suppress horizontal rule lines (HR Tags) in directory listings.
IndexOptions VersionSort	Added	This causes files containing version numbers to sort in a natural way.

10.2.6 mod_case_filter and mod_case_filter_in

mod_case_filter

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module converts the output to uppercase.	New	

Directive	Status	Description
CaseFilter	Added	Run a case filter on this host

mod_case_filter_in

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module converts the input to uppercase. It does NOT convert request headers.	New	

Directive	Status	Description
CaseFilterIn	Added	Run an input case filter on this host

10.2.7 mod_cgid

Module Notes	Status	How to migrate from HP Apache 1.3.x
mod_cgid creates an external daemon that is responsible for forking child processes to run CGI scripts.	New	How to Migrate Common Gateway Interface (CGI) ASF Apache 2.x doc on mod_cgid ASF Apache 2.x doc on mod_cgi ASF Apache 1.3.x doc on mod_cgi

Directive	Status	Description
Scriptsock	Added	the name of the socket to use for communication with the cgi daemon.

10.2.8 mod_charset_lite

Module Notes	Status	How to migrate from HP Apache 1.3.x
mod_charset_lite	New	

Directive	Status	Description
CharsetDefault	Added	Name of default charset
CharsetOptions	Added	Valid options: ImplicitAdd, NoImplicitAdd, DebugLevel=n
CharsetSourceEnc	Added	Source (html, cgi, ssi) file charset
ExtFilterDefine	Added	Define an external filter
ExtFilterOptions	Added	/* same as SetInputFilter/SetOutputFilter */ valid options: DebugLevel=n, LogStderr, NoLogStderr

10.2.9 mod_dav and mod_dav_fs

mod_dav

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module implements the HTTP Distributed Authoring and Versioning (DAV) specification for posting and maintaining web content	New	Summary of New Features in the HP Apache Version 2.x See also: The webDAV Home Page

Directive	Status	Description
DAV	Added	specify the DAV provider for a directory or location
DAVDepthInfinity	Added	allow Depth infinity PROPFIND requests
DAVMinTimeout	Added	specify minimum allowed timeout
DAVParam	Added	DAVParam <parameter name> <parameter value>

mod_dav_fs

Module Notes	Status	How to migrate from HP Apache 1.3.x
Required by mod_dav	New	See `mod_dav` above.

Directive	Status	Description
DAVLockDB	Added	specify a lock database

10.2.10 mod_define

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module does not yet have a 2.x counterpart.	Removed	

Directive	Status	Description
Define	Removed	Define a configuration variable

10.2.11 mod_deflate

Module Notes	Status	
	New	

Directive	Status	Description
DeflateBufferSize	Added	Set the Deflate Buffer Size
DeflateFilterNote	Added	Set a note to report on compression ratio
DeflateMemLevel	Added	Set the Deflate Memory Level (1-9)
DeflateWindowSize	Added	Set the Deflate window size (1-15)

10.2.12 mod_echo

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module impliments the echo protocol which simply returns the request unchanged.	New	ASF Apache 2.x doc on mod_echo

Directive	Status	Description
ProtocolEcho	Added	Run an echo server on this host

10.2.13 mod_env

Module Notes	Status	How to migrate from HP Apache 1.3.x
This modifies the environment which is passed to CGI scripts and SSI pages	Changed	ASF Apache 2.x doc on mod_env ASF Apache 1.3.x doc on mod_env

Directive	Status	Description
UnsetEnv	Changed	Now it will not unset a SetEnv and PassEnv directive following that UnSetEnv within the same container.

10.2.14 mod_ext_filter

Module Notes	Status	
Pass the response body through an external program before delivery to the client. This works like a unix filter.	New	ASF Apache 2.x doc on mod_ext_filter

Directive	Status	Description
ExtFilterDefine	Added	Define an external filter
ExtFilterOptions	Added	same as SetInputFilter/SetOutputFilter

10.2.15 mod_headers

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module is much more flexible in HP Apache 2.x. It can now modify request headers used by mod_proxy, and it can conditionally set response headers	Changed	ASF Apache 2.x doc on mod_headers ASF Apache 1.3.x doc on mod_headers

Directive	Status	Description
Header echo	Changed	Added "echo" option. For example "Header echo regex" will cause any headers received on the request that match regex to be echoed to [included in] the response headers.
RequestHeader	Added	an action, header and value

10.2.16 mod_include

Module Notes	Status	How to migrate from HP Apache 1.3.x
New directives allow the default start and end tags for SSI elements to be changed and allow for error and time format configuration to take place in the main configuration file rather than in the SSI document. Results from regular expression parsing and grouping (now based on Perl's regular expression syntax) can be retrieved using mod_include's variables $0 .. $9.	Changed	ASF Apache 2.x doc on mod_include ASF Apache 1.3.x doc on mod_include

Directive	Status	Description
SSIEndTag	Added	SSI End String Tag
SSIErrorMsg	Added	a string
SSIStartTag	Added	SSI Start String Tag
SSIUndefinedEcho	Added	
SSITimeFormat	Added	a strftime(3) formatted string

10.2.17 mod_jk

Module Notes	Status	How to migrate from HP Apache 1.3.x
The mod_jk module provides better support for SSL. Also, Tomcat supports many web servers through a compatibility layer named the jk library. The layered approach provided by the jk library makes it easier to support both Apache1.3.x and Apache2.x.	New	Migrating Tomcat and Java to Version 2.x See also: mod_jk User Guide and Apache Tomcat Home Page

Directive	Status	Description
JkAutoMount	Added	automatic mount points to a Tomcat worker
JkCERTSIndicator	Added	Name of the Apache environment that contains SSL client certificates
JkCIPHERIndicator	Added	Name of the Apache environment that contains SSL client cipher
JkEnvVar	Added	Adds a name of environment variable that should be sent from web server to servlet-engine
JkExtractSSL	Added	Turns on SSL processing and information gathering by mod_jk
JkHTTPSIndicator	Added	Name of the Apache environment that contains SSL indication
JkKEYSIZEIndicator	Added	Name of the Apache environment that contains SSL key size in use
JkLogFile	Added	Full path to the Jakarta mod_jk module log file
JkLogLevel	Added	The Jakarta mod_jk module log level, can be debug, info, error or emerg
JkLogStampFormat	Added	The Jakarta mod_jk module log format, follow strftime synthax
JkMount	Added	A mount point from a context to a Tomcat worker
JkMountCopy	Added	Should the base server mounts be copied to the virtual server
JkOptions	Added	Set one of more options to configure the mod_jk module
JkSESSIONIndicator	Added	Name of the Apache environment that contains SSL session
JkWorker	Added	worker property
JkWorkersFile	Added	the name of a worker file for the Jakarta servlet containers

10.2.18 mod_jserv

Module Notes	Status	How to migrate from HP Apache 1.3.x
Replaced with mod_jk	Removed	The Apache JServ Project

Directive	Status	Description
ApJServAction	Removed	Apache JServ action mapping extension to servlets.
ApJServBalance	Removed	Apache JServ load-balancing server set.
ApJServDefaultHost	Removed	The default host running Apache JServ.
ApJServDefaultPort	Removed	The default port on which Apache JServ is running on.
ApJServDefaultProtocol	Removed	The default protocol used for connecting to Apache JServ.
ApJServEnvVar	Removed	Apache JServ: protocol ajpv12 : env var to send to the server
ApJServHost	Removed	Apache JServ host definition.
ApJServLogFile	Removed	Apache JServ log file relative to Apache root directory.
ApJServLogLevel	Removed	Apache JServ log verbosity.
ApJServManual	Removed	Whether Apache JServ is running in manual or automatic mode.
ApJServMount	Removed	Where Apache JServ servlets will be mounted under Apache.
ApJServMountCopy	Removed	Whether <VirtualHost> inherits base host mount points or not.
ApJServProperties	Removed	The full pathname of jserv.properties file.
ApJServProtocolParameter	Removed	Apache JServ protocol-dependant property.
ApJServRetryAttempts	Removed	Apache JServ: retry attempts (1s appart) before returning server error
ApJServRoute	Removed	Apache JServ host routing identifier.
ApJServSecretKey	Removed	Apache JServ secret key file relative to Apache root directory.
ApJServShmFile	Removed	The full pathname of shared memory file.
ApJServVMInterval	Removed	Apache JServ: the interval between 2 polls of the JVM
ApJServVMTimeout	Removed	Apache JServ: the amount of time given for the JVM to start or stop

10.2.19 mod_log_agent

Module Notes	Status	How to migrate from HP Apache 1.3.x
This deprecated module has been replaced with `mod_log_config`.	Removed	ASF Apache 1.3.x doc on mod_log_agent

Directive	Status	Description
AgentLog	Removed	the filename of the agent log

10.2.20 mod_log_config

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module provides logging flexibility.	Changed	ASF Apache 2.x doc on mod_log_config ASF Apache 1.3.x doc on mod_log_config

Directive	Status	Description
BufferedLogs	Added	Enable Buffered Logging (experimental)
LogFormat	Changed	Added options %D logs time it takes to a request in milliseconds %X logs connection status at the end of the response as follows: 'X' - connection aborted before the response completed. '+' - connection may be kept-alive by the server. '-' - connection will be closed by the server.

10.2.21 mod_log_referer

Module Notes	Status	How to migrate from HP Apache 1.3.x
This deprecated module has been replaced with `mod_log_config`.	Removed	ASF Apache 1.3.x doc on mod_log_referer

Directive	Status	Description
RefererIgnore	Removed	referer hostnames to ignore
RefererLog	Removed	the filename of the referer log

10.2.22 mod_mime

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module is used to associate various bits of "meta information" with files by their filename extensions. This information is sent to the browser, and participates in content negotiation,	Changed	ASF Apache 2.x doc on mod_mime ASF Apache 1.3.x doc on mod_mime

Directive	Status	Description
AddInputFilter	Added	input filter name (or ; delimited names) followed by one or more file extensions
AddOutputFilter	Added	output filter name (or ; delimited names) followed by one or more file extensions
ModMimeUsePathInfo	Added	Set to 'yes' to allow mod_mime to use path info for type checking
MultiviewsMatch	Added	NegotiatedOnly (default), Handlers and/or Filters, or Any
RemoveCharset	Added	one or more file extensions
RemoveInputFilter	Added	one or more file extensions
RemoveLanguage	Added	one or more file extensions
RemoveOutputFilter	Added	one or more file extensions

10.2.23 mod_negotiation

Module Notes	Status	How to migrate from HP Apache 1.3.x
A new ForceLanguagePriority directive can be used to assure that the client receives a single document in all cases, rather than NOT ACCEPTABLE or MULTIPLE CHOICES responses. In addition, the negotiation and MultiViews algorithms have been cleaned up to provide more consistent results and a new form of type map that can include document content is provided.	Changed	ASF Apache 2.x doc on mod_negotiation ASF Apache 1.3.x doc on mod_negotiation

Directive	Status	Description
CacheNegotiatedDocs	Changed	
ForceLanguagePriority	Added	Force LanguagePriority elections, either None, or Fallback and/or Prefer

10.2.24 mod_perl

Module Notes	Status	How to migrate from HP Apache 1.3.x
This is a very powerful module that basically does two things. It increases performance by causing Apache to run its own Perl interpreters that cache Perl scripts. It also provides an api that allows access to Apache internals.	Changed	Migrating Perl Main mod_perl web site Documentation on mod_perl modules

Directive	Status	Description
__END__	Removed	Stop reading config
<Perl	Added	NOT YET IMPLEMENTED
=back	Removed	End of =over
=cut	Removed	End of POD
=pod	Removed	Start of POD
PerlFreshRestart	Removed	Tell mod_perl to reload modules and flush Apache::Registry cache on restart
PerlInitHandler	Added	Subroutine name
PerlInterpMax	Added	Max number of running Perl interpreters
PerlInterpMaxRequests	Added	Max number of requests per Perl interpreters
PerlInterpMaxSpare	Added	Max number of spare Perl interpreters
PerlInterpMinSpare	Added	Min number of spare Perl interpreters
PerlInterpScope	Added	Scope of a Perl interpreter
PerlInterpStart	Added	Number of Perl interpreters to start
PerlOpmask	Removed	Opmask File
PerlOptions	Added	Perl Options
PerlScript	Removed	this directive is deprecated, use `PerlRequire'
PerlSwitches	Added	Perl Switches
PerlTrace	Added	Trace level

10.2.25 mod_php

Module Notes	Status	How to migrate from HP Apache 1.3.x
	Changed	

Directive	Status	Description
PHPINIDir	Added	Directory containing the php.ini file

10.2.26 mod_proxy, mod_cache, mod_disk_cache, mod_file_cache, and mod_mem_cache

See Migrating mod_proxy

mod_proxy

Module Notes	Status	How to migrate from HP Apache 1.3.x
The proxy module has been completely rewritten to take advantage of the new filter infrastructure and to implement a more reliable, HTTP/1.1 compliant proxy. In addition, new <Proxy> configuration sections provide more readable (and internally faster) control of proxied sites; overloaded <Directory "proxy:..."> configuration are not supported. The module is now divided into specific protocol support modules including proxy_connect, proxy_ftp and proxy_http.	Changed	ASF Apache 2.x doc on mod_proxy ASF Apache 1.3.x doc on mod_proxy

Directive	Status	Description
<Proxy >	Added	Container for directives affecting resources located in the proxied location
<ProxyMatch >	Added	Container for directives affecting resources located in the proxied location, in regular expression syntax
CacheDirLength	Removed	The number of characters in subdirectory names
CacheDirLevels	Removed	The number of levels of subdirectories in the cache
CacheGcInterval	Removed	The interval between garbage collections, in hours
CacheRoot	Removed	The directory to store cache files
CacheSize	Removed	The maximum disk space used by the cache in Kb
NoCache	Removed	A list of names, hosts or domains for which caching is *not* provided
ProxyErrorOverride	Added	use our error handling pages instead of the servers' we are proxying
ProxyIOBufferSize	Added	IO buffer size for outgoing HTTP and FTP connections in bytes
ProxyMaxForwards	Added	The maximum number of proxies a request may be forwarded through.
ProxyPass	Changed	Adds "ProxyPass path !" to exclude a path from being proxied
ProxyPassReversed	Changed	Adds ProxyPassReverse url when placed in a <location> directive
ProxyPreserveHost	Added	on if we should preserve host header while proxying
ProxyRemoteMatch	Added	A regex pattern and a proxy server
ProxyTimeout	Added	Set the timeout (in seconds) for a proxied connection. This overrides the server timeout

mod_cache

Module Notes	Status	How to migrate from HP Apache 1.3.x
An RFC 2616 compliant HTTP content cache. `mod_cache` requires the services of one or more storage management modules such as `mod_disk_cache` or `mod_file_cache`.	New	ASF Apache 2.x doc on mod_cache

Directive	Status	Description
CacheDefaultExpireMin	Added	The default time in Minutes to cache a document
CacheDisable	Added	A partial URL prefix below which caching is disabled
CacheEnable	Added	A cache type and partial URL prefix below which caching is enabled
CacheIgnoreCacheControl	Added	Ignore requests from the client for uncached content
CacheIgnoreNoLastMod	Added	Ignore Responses where there is no Last Modified Header
CacheMaxExpireMin	Added	The maximum time in Minutes to cache a document
CacheMemEntrySize	Added	The maximum size (in bytes) that a entry can take
CacheMemSize	Added	The maximum space used by the cache in Kb
CacheOn	Added	On if the transparent cache should be enabled
CacheMaxStreamingBuffer	Added	Maximum number of bytes of content to buffer for a streamed response

mod_disk_cache

Module Notes	Status	How to migrate from HP Apache 1.3.x
A disk based storage manager, generally used for proxy caching.	New	

Directive	Status	Description
CacheDirLength	Added	The number of characters in subdirectory names
CacheDirLevels	Added	The number of levels of subdirectories in the cache
CacheExpiryCheck	Added	On if cache observes Expires date when seeking files
CacheGcClean	Added	The time in hours to retain unchanged files that match a url
CacheGcDaily	Added	The time of day for garbage collection (24 hour clock)
CacheGcInterval	Added	The interval between garbage collections, in hours
CacheGcMemUsage	Added	The maximum kilobytes of memory used for garbage collection
CacheGcUnused	Added	The time in hours to retain unused file that match a url
CacheMaxFileSize	Added	The maximum file size to cache a document
CacheMinFileSize	Added	The minimum file size to cache a document
CacheRoot	Added	The directory to store cache files
CacheSize	Added	The maximum disk space used by the cache in Kb
CacheTimeMargin	Added	The minimum time margin to cache a document

mod_file_cache

Module Notes	Status	How to migrate from HP Apache 1.3.x
An mmap based and/or file handle based storage manager that allows the server file system to control whether the file is in memory or not. This is an extension of the 1.3.x mod_mmap_static module	New	ASF Apache 2.x doc on mod_file_cache ASF Apache 1.3.x doc on mod_mmap_static

Directive	Status	Description
cachefile	Added	A space separated list of files to add to the file handle cache at config time
mmapfile	Added	A space separated list of files to mmap at config time

mod_mem_cache

Module Notes	Status	How to migrate from HP Apache 1.3.x
	New	

Directive	Status	Description
MCacheMaxObjectCount	Added	The maximum number of objects allowed to be placed in the cache

MCacheMaxObjectSize	Added	The maximum size (in bytes) of an object to be placed in the cache
MCacheMinObjectSize	Added	The minimum size (in bytes) of an object to be placed in the cache
MCacheRemovalAlgorithm	Added	The algorithm used to remove entries from the cache (default: GDSF)
MCacheSize	Added	The maximum amount of memory used by the cache in KBytes

10.2.27 mod_setenvif

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module allows environment variables to be set based on request characteristics.	Changed	ASF Apache 2.x doc on mod_setenvif ASF Apache 1.3.x doc on mod_setenvif

Directive	Status	Description
SetEnvIf	Changed	Allows regular expressions to be specified in the directive attribute field. For example: SetEnvIf ^TS* [a-z].* HAVE_TS will cause HAVE_TS to be set if any of the request headers begins with "TS" and has a value that begins with any character in the set [a-z].
SetEnvIfNoCase	Changed	Same as SetEnvIf

10.2.28 mod_ssl

Module Notes	Status	How to migrate from HP Apache 1.3.x
New module in ASF Apache 2.0. This module is an interface to the SSL/TLS encryption protocols provided by OpenSSL. Previous versions of HP Apache used mod_ssl from modssl.org.	New	ASF Apache 2.x doc on mod_ssl See also: mod_ssl User's Guide OpenSSL User's Guide

Directive	Status	Description
SSLLog	Removed	Use ErrorLog
SSLLogLevel	Removed	Use LogLevel

10.2.29 mod_suexec

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module Aalows CGI scripts to run as a specified user and group	New	ASF Apache 2.x doc on mod_suexec

Directive	Status	Description
SuexecUserGroup	Added	User and group for spawned processes

10.2.30 mod_usertrack

Module Notes	Status	How to migrate from HP Apache 1.3.x
This module uses cookies to provide for a *clickstream* log of user activity on a site.	Changed	ASF Apache 2.x doc on mod_usertrack ASF Apache 1.3.x doc on mod_usertrack

Directive	Status	Description
CookieDomain	Added	domain to which this cookie applies
CookieStyle	Added	'Netscape', 'Cookie' (RFC2109), or 'Cookie2' (RFC2965)

10.2.31 mod_vhost_alias

Module Notes	Status	How to migrate from HP Apache 1.3.x
Provides for dynamically configured mass virtual hosting	Changed	ASF Apache 2.x doc on mod_vhost_alias ASF Apache 1.3.x doc on mod_vhost_alias

Directive	Status	Description
VirtualScriptAlias	Changed	When mod_vhost_alias is used with no VirtualScriptAlias directives then cgi-bin works like a regular directory.