

# Service Request Catalog 9.32 Deployment and Sizing Guide



For supported Windows® operating systems

## Contents

Service Request Catalog	2
Service Manager	2
Deployment Guide	2
Pre-Conditions	2
Prepare Server for SRC Deployment	2
Service Manager 9.32 Installed	2
Client Operating System	2
SRC Deployment Models	2
Standalone Deployment Model	2
Horizontal Scaling Model	3
SRC Standalone Deployment Guide	4
Deploy and Configure the SRC Instance	4
KM Search Engine Installation and Configuration	4
Verify Your SRC Standalone Deployment	4
SRC Horizontal Scaling Deployment Guide	4
Deploy and Configure the Apache HTTP Server (httpd)	4
Deploy and Configure Multiple SRC Application Instances	15
Verify the Integration	16
Sizing Guide	17
Benchmark Performance Testing	17
Lab Topologic Architecture	17
JDK and Apache Tomcat specifications	19
Dataset	19
SRC Settings	20
SM Settings	20
Sizing Reference Recommended	21
Support	22

# In this Version

This document presents how to deploy SRC 9.32 as a standalone model (the simplest model), or as a horizontally-scaled model using an Apache HTTP Server to load balance client traffic to different SRC 9.32 instances. It also provides a sizing guide to help you determine optimal performance. We recommend reading the document thoroughly to gain a complete sense of the SRC 9.32 deployment, architecture, and how it acts together with Service Manager 9.32.

## Service Request Catalog

Service Request Catalog is a self-service web interface to request IT goods and services, manage those requests, or approve requests for others. This release integrates with Service Manager 9.32, which provides the catalog content, fulfillment, and approval workflow for each request.

## Service Manager

HP Service Manager is a comprehensive and fully integrated IT Service Management software suite that enables IT organizations to improve service levels, balance resources, control costs, and mitigate risk exposure to the organization. It enables you to manage services using a "lifecycle" approach, with consistent improvement built into the governance model.

## Deployment Guide

The following section provides an example of how to deploy SRC 9.32 on a Windows Server 2008.

### Pre-Conditions

#### Prepare Server for SRC Deployment

It is recommended to deploy SRC on a Windows Server 2008 separate from that of Service Manager for better performance. Before you start the installation, it is assumed that you have prepared the server and installed the following third-party resources:

- Sun Java JRE 7.0.15 or later release
- Apache Tomcat 7

In this document, performance testing for SRC is performed on a 64-bit Windows Server 2008 machine. For more details of the configuration, refer to [Lab Topologic Architecture chapter](#).

#### Service Manager 9.32 Installed

SRC 9.32 works together with Service Manager 9.32. Before you start the deployment, you must have an instance of Service Manager 9.32 installed.

**Note:** If you need help installing Service manager, refer to the *Service Manager 9.32 Interactive Installation Guide*.

#### Client Operating System

To access the SRC UI Web service, you must install Adobe Flash Player 10.3 (or later) on the client. For details, refer to the *Service Request Catalog 9.32 Support Matrix*.

### SRC Deployment Models

This section presents two different kinds of SRC deployment models, a standalone deployment model and a horizontally-scaled model. The former is the simplest and most basic deployment model and includes only a single SRC instance. The latter scales multiple SRC instances horizontally to support more users.

#### Standalone Deployment Model

The architecture shown in Figure 1 is the standalone model, and includes the following components:

1. Service Request Catalog 9.32

2. The Knowledge Management (KM) Search Engine
3. Service Manager 9.32
4. Database

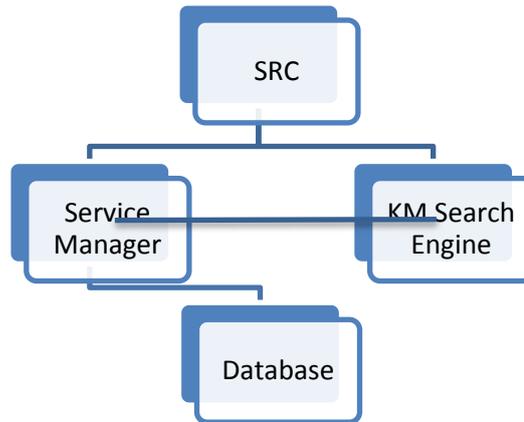


Figure -1 Standalone Deployment Model

**Note:** Please refer to [SRC Standalone Deployment Guide](#) for deployment guide.

### Horizontal Scaling Model

To support more users, it is recommended to scale multiple SRC applications horizontally, by using an Apache HTTP Server as a software load balancer to direct client traffic to multiple SRC instances. This model also supports a hardware load balancer (such as Cisco ACE 4710) which can direct SRC requests to multiple Service Manager nodes.

See the following 'Figure -2 Horizontal Scaling Models' for more details on this architecture.

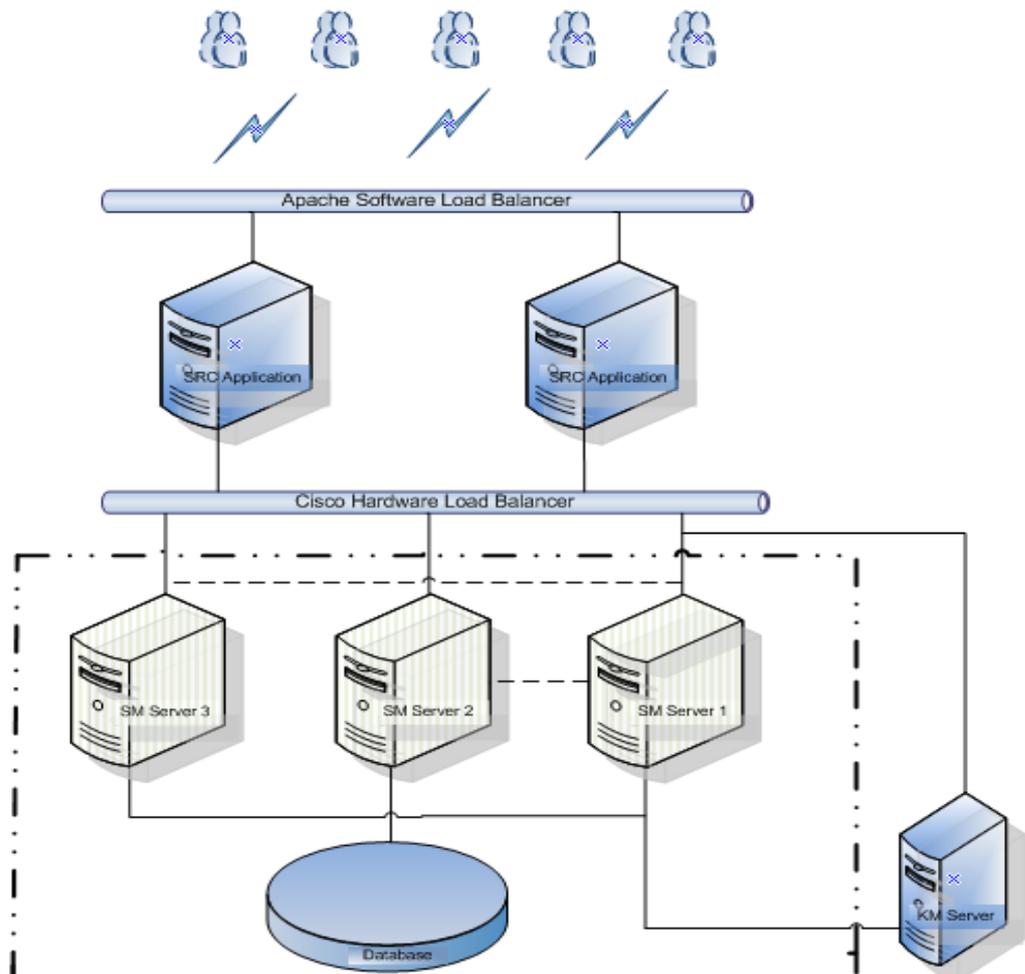


Figure -2 Horizontal Scaling Models

**Notes:**

1. To configure the Cisco hardware load balancer, refer to document *Service Request Catalog HWLB Configuration Guide -Cisco ACE 4710*.
2. The KM Search Engine supports the Solr search engine. The K2 engine is not supported;
3. KM does not support HTTPS for the Solr search engine;
4. Refer to the *Service Manager 9.32 Search Engine Guide* for KM search engine configuration details.
5. The Service Manager deployment is marked by the bold dotted boundary line. For details on this aspect of the figure, refer to the sizing and deployment documentation for Service Manager.

For the horizontally-scaled model, refer to the [SRC Horizontal Scaling Deployment Guide](#).

**SRC Standalone Deployment Guide**

Before you deploy SRC, an instance of Service Manager 9.32 must be installed. During the deployment, you will need the hostname of the server on which Service Manager installed and the port number of the Service Manager instance. To finish the SRC standalone deployment, follow these steps:

**Deploy and Configure the SRC Instance**

This section requires both JDK and Apache Tomcat installed as the preconditions. Refer to the *SRC 9.32 Interactive Installation Guide* for the installation details.

**KM Search Engine Installation and Configuration**

1. For installation of KM, refer to *Service Manager 9.32 Knowledge Management Search Engine Guide*.

To configure Knowledge Management for SRC, refer to the 'Enable Knowledge Management' section in the *Service Request Catalog 9.32 Customization Guide* document.

**Verify Your SRC Standalone Deployment**

2. Using Internet Explorer, access the SRC UI by specifying a URL in the following format:  
`http://<SRC-Server-IP>:<Port>/<path>/secure/main.jsp`  
 For example: <http://localhost:8080/src932/secure/main.jsp>
3. If successful, you should see the SRC login page. If so, enter "falcon" in the User Name field, and then click Submit to log in to the SRC system.

**SRC Horizontal Scaling Deployment Guide**

The architecture of this model refers to Figure 2 in the [Horizontal Scaling Model](#) section. Topics presented in this chapter are shown below:

1. Deploy and Configure the Apache HTTP Server (httpd)
2. Deploy and Configure Multiple SRC Application Instances
3. Verify the Integration

**Deploy and Configure the Apache HTTP Server (httpd)**

The Apache HTTP Server can be used as a software load balancer to direct user traffic to SRC instances, as shown in Figure 2 of chapter Horizontal Scaling Model. To deploy using the Horizontal Scaling model, follow these steps:

1. Download the Apache HTTP Server 2.0.64 MSI Installer from following link:  
<http://httpd.apache.org/download.cgi>
2. Double-click the MSI Installer to install Apache HTTP Server.  
 Note: You may change the default installation location. In this document, the installation path is as follows:  
 C:\ApacheGroup\
3. After the installation is complete, specify APACHE\_HOME as a system variable and set it to the following value:  
 C:\ApacheGroup\Apache2.

4. Check whether the mod\_jk.so file exists in the <APACHE\_HOME>/module directory; If it does not, download this file and place it into the folder<APACHE\_HOME>/module. This file is a core component and is the conduit between the Apache HTTP Server and the Tomcat Web application server.
5. Create a workers.properties file in the <APACHE\_HOME>/conf directory. The following template is created for multiple Tomcat instances, you may modify this example according to your configuration:

---

```

worker.list=loadbalancer
worker.tomcat1.port=8001
worker.tomcat1.host=serverHostname
worker.tomcat1.type=ajp13
worker.tomcat1.lbfactor=1

worker.tomcat2.port=8002
worker.tomcat2.host= serverHostname
worker.tomcat2.type=ajp13
worker.tomcat2.lbfactor=1

worker.tomcat3.port=8003
worker.tomcat3.host= serverHostname
worker.tomcat3.type=ajp13
worker.tomcat3.lbfactor=1
.....
worker.tomcatN.port=800N
worker.tomcatN.host= serverHostname
worker.tomcatN.type=ajp13
worker.tomcatN.lbfactor=1

worker.loadbalancer.type=lb
worker.loadbalancer.balance_workers=tomcat1, tomcat2, ..., tomcatN
worker.balancer.sticky_session=True
    
```

**Note:** Make sure to set sticky\_session to True to ensure that requests are always routed back to the node which originally assigned the JsessionID.

6. Modify httpd.conf file in ./conf directory according the following example:

---

```

# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
ServerRoot "C:/ApacheGroup/Apache2"

# PidFile: The file in which the server should record its process
# identification number when it starts.
PidFile logs/httpd.pid

#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300

#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On

#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100

#
# KeepAliveTimeout: Number of seconds to wait for the next request from the
    
```

---

```

# same client on the same connection.
#
KeepAliveTimeout 15

###
### Server-Pool Size Regulation (MPM specific)
###

# WinNT MPM
# ThreadsPerChild: constant number of worker threads in the server process
# MaxRequestsPerChild: maximum number of requests a server process serves
<IfModule mpm_winnt.c>
ThreadLimit 9000
ThreadsPerChild 3000
MaxRequestsPerChild 0
</IfModule>

#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
Listen 16.103.225.211:8088
#
# Dynamic Shared Object (DSO) Support
# Example:
# LoadModule foo_module modules/mod_foo.so
#
LoadModule access_module modules/mod_access.so
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule asis_module modules/mod_asis.so
LoadModule auth_module modules/mod_auth.so
#LoadModule auth_anon_module modules/mod_auth_anon.so
#LoadModule auth_dbm_module modules/mod_auth_dbm.so
#LoadModule auth_digest_module modules/mod_auth_digest.so
LoadModule autoindex_module modules/mod_autoindex.so
#LoadModule cern_meta_module modules/mod_cern_meta.so
LoadModule cgi_module modules/mod_cgi.so
#LoadModule dav_module modules/mod_dav.so
#LoadModule dav_fs_module modules/mod_dav_fs.so
LoadModule dir_module modules/mod_dir.so
LoadModule env_module modules/mod_env.so
#LoadModule expires_module modules/mod_expires.so
#LoadModule file_cache_module modules/mod_file_cache.so
#LoadModule headers_module modules/mod_headers.so
LoadModule imap_module modules/mod_imap.so
LoadModule include_module modules/mod_include.so
#LoadModule info_module modules/mod_info.so
LoadModule isapi_module modules/mod_isapi.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule mime_module modules/mod_mime.so
#LoadModule mime_magic_module modules/mod_mime_magic.so
#LoadModule proxy_module modules/mod_proxy.so
#LoadModule proxy_connect_module modules/mod_proxy_connect.so
#LoadModule proxy_http_module modules/mod_proxy_http.so
#LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
LoadModule negotiation_module modules/mod_negotiation.so
#LoadModule rewrite_module modules/mod_rewrite.so
LoadModule setenvif_module modules/mod_setenvif.so
#LoadModule speling_module modules/mod_speling.so
#LoadModule status_module modules/mod_status.so
#LoadModule unique_id_module modules/mod_unique_id.so
LoadModule userdir_module modules/mod_userdir.so
#LoadModule usertrack_module modules/mod_usertrack.so
#LoadModule vhost_alias_module modules/mod_vhost_alias.so

```

---

```

#LoadModule ssl_module modules/mod_ssl.so

#
# ExtendedStatus controls whether Apache will generate "full" status
# information (ExtendedStatus On) or just basic information (ExtendedStatus
# Off) when the "server-status" handler is called. The default is Off.
#
#ExtendedStatus On

#### Section 2: 'Main' server configuration
#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed. This address appears on some server-generated pages, such
# as error documents. e.g. admin@your-domain.com
#
ServerAdmin admin@devlab.ad

#
# UseCanonicalName: Determines how Apache constructs self-referencing
# URLs and the SERVER_NAME and SERVER_PORT variables.
# When set "Off", Apache will use the Hostname and Port supplied
# by the client. When set "On", Apache will use the value of the
# ServerName directive.
#
UseCanonicalName Off

#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "C:/ApacheGroup/Apache2/htdocs"

#
# Each directory to which Apache has access can be configured with respect
# to which services and features are allowed and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very restrictive set of
# features.
#
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#

#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "C:/ApacheGroup/Apache2/htdocs">

#
# Possible values for the Options directive are "None", "All",
# or any combination of:
# Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.

```

---

```

#
# The Options directive is both complicated and important. Please see
# http://httpd.apache.org/docs/2.0/mod/core.html#options
# for more information.
#
    Options Indexes FollowSymLinks

#
# AllowOverride controls what directives may be placed in .htaccess files.
# It can be "All", "None", or any combination of the keywords:
# Options FileInfo AuthConfig Limit
#
    AllowOverride None

#
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all
</Directory>

#
# UserDir: The name of the directory that is appended onto a user's home
# directory if a ~user request is received. Be especially careful to use
# proper, forward slashes here. On Windows NT, "Personal/My Website"
# is a more appropriate choice.
#
UserDir "My Documents/My Website"
#
# DirectoryIndex: sets the file that Apache will serve if a directory
# is requested.
#
# The index.html.var file (a type-map) is used to deliver content-
# negotiated documents. The MultiViews Option can be used for the
# same purpose, but it is much slower.
#
DirectoryIndex index.html index.html.var

#
# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives. See also the AllowOverride
# directive.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess and .htpasswd files from being
# viewed by Web clients.
#
<FilesMatch "^\.ht">
    Order allow,deny
    Deny from all
</FilesMatch>

#
# TypesConfig describes where the mime.types file (or equivalent) is
# to be found.
#
TypesConfig conf/mime.types

#
# DefaultType is the default MIME type the server will use for a document
# if it cannot otherwise determine one, such as from filename extensions.
# If your server contains mostly text or HTML documents, "text/plain" is
# a good value. If most of your content is binary, such as applications
# or images, you may want to use "application/octet-stream" instead to

```

---

```

# keep browsers from trying to display binary files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the server to use various hints from the
# contents of the file itself to determine its type. The MIMEMagicFile
# directive tells the module where the hint definitions are located.
#
<IfModule mod_mime_magic.c>
    MIMEMagicFile conf/magic
</IfModule>

#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
HostnameLookups Off

#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here. If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog logs/error.log

#
# LogLevel: Control the number of messages logged to the error.log.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%[Referer]i\" \"%[User-Agent]i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%[Referer]i -> %U" referer
LogFormat "%[User-agent]i" agent

# You need to enable mod_logio.c to use %l and %O
#LogFormat "%h %l %u %t \"%r\" %>s %b \"%[Referer]i\" \"%[User-Agent]i\" %l %O" combinedio

#
# The location and format of the access logfile (Common Logfile Format).
# If you do not define any access logfiles within a <VirtualHost>
# container, they will be logged here. Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles, transactions will be
# logged therein and *not* in this file.
#
CustomLog logs/access.log common

#
# ServerTokens
# This directive configures what you return as the Server HTTP response
# Header. The default is 'Full' which sends information about the OS-Type

```

---

```

# and compiled in modules.
# Set to one of: Full | OS | Minor | Minimal | Major | Prod
# where Full conveys the most information, and Prod the least.
#
ServerTokens Full

#
# Optionally add a line containing the server version and virtual host
# name to server-generated pages (internal error documents, FTP directory
# listings, mod_status and mod_info output etc., but not CGI generated
# documents or custom error documents).
# Set to "EMail" to also include a mailto: link to the ServerAdmin.
# Set to one of: On | Off | EMail
#
ServerSignature On

#
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename then the server will
# require it to be present in the URL. So "/icons" isn't aliased in this
# example, only "/icons/". If the fakename is slash-terminated, then the
# realname must also be slash terminated, and if the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed directory listings. If you
# do not use FancyIndexing, you may comment this out.
#
Alias /icons/ "C:/ApacheGroup/Apache2/icons/"

<Directory "C:/ApacheGroup/Apache2/icons">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

#
# This should be changed to the ServerRoot/manual/. The alias provides
# the manual, even if you choose to move your DocumentRoot. You may comment
# this out if you do not care for the documentation.
#
AliasMatch ^/manual(?:/(?:de|en|es|fr|ja|ko|ru|tr))?(/.*)?$ "C:/ApacheGroup/Apache2/manual$1"

<Directory "C:/ApacheGroup/Apache2/manual">
    Options Indexes
    AllowOverride None
    Order allow,deny
    Allow from all

<Files *.html>
    SetHandler type-map
</Files>
# .tr is text/troff in mime.types!
<Files *.html.tr.utf8>
    ForceType text/html
</Files>

    SetEnvIf Request_URI ^/manual/(de|en|es|fr|ja|ko|ru|tr)/ prefer-language=$1
    RedirectMatch 301 ^/manual(?:/(de|en|es|fr|ja|ko|ru|tr)){2,}/.*?$ /manual/$1$2
</Directory>

#
# ScriptAlias: This controls which directories contain server scripts.
# ScriptAliases are essentially the same as Aliases, except that

```

---

```

# documents in the realname directory are treated as applications and
# run by the server when requested rather than as documents sent to the client.
# The same rules about trailing "/" apply to ScriptAlias directives as to
# Alias.
#
ScriptAlias /cgi-bin/ "C:/ApacheGroup/Apache2/cgi-bin/"

#
# "C:/ApacheGroup/Apache2/cgi-bin" should be changed to whatever your ScriptAliased
# CGI directory exists, if you have that configured.
#
<Directory "C:/ApacheGroup/Apache2/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>
#
# IndexOptions: Controls the appearance of server-generated directory
# listings.
#
IndexOptions FancyIndexing VersionSort

#
# AddIcon* directives tell the server which icon to show for different
# files or filename extensions. These are only displayed for
# FancyIndexed directories.
#
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip

AddIconByType (TXT,/icons/text.gif) text/*
AddIconByType (IMG,/icons/image2.gif) image/*
AddIconByType (SND,/icons/sound2.gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*

AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrm .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core

AddIcon /icons/back.gif .
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^^DIRECTORY^^
AddIcon /icons/blank.gif ^^BLANKICON^^

#
# DefaultIcon is which icon to show for files which do not have an icon
# explicitly set.
#
DefaultIcon /icons/unknown.gif

#
# ReadmeName is the name of the README file the server will look for by

```

---

```

# default, and append to directory listings.
#
# HeaderName is the name of a file which should be prepended to
# directory indexes.
ReadmeName README.html
HeaderName HEADER.html

#
# IndexIgnore is a set of filenames which directory indexing should ignore
# and not include in the listing. Shell-style wildcarding is permitted.
#
IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t

#
# DefaultLanguage and AddLanguage allows you to specify the language of
# a document. You can then use content negotiation to give a browser a
# file in a language the user can understand.
#
AddLanguage ca .ca
AddLanguage cs .cz .cs
AddLanguage da .dk
AddLanguage de .de
AddLanguage el .el
AddLanguage en .en
AddLanguage eo .eo
AddLanguage es .es
AddLanguage et .et
AddLanguage fr .fr
AddLanguage he .he
AddLanguage hr .hr
AddLanguage it .it
AddLanguage ja .ja
AddLanguage ko .ko
AddLanguage ltz .ltz
AddLanguage nl .nl
AddLanguage nn .nn
AddLanguage no .no
AddLanguage pl .po
AddLanguage pt .pt
AddLanguage pt-BR .pt-br
AddLanguage ru .ru
AddLanguage sv .sv
AddLanguage tr .tr
AddLanguage zh-CN .zh-cn
AddLanguage zh-TW .zh-tw

#
# LanguagePriority allows you to give precedence to some languages
# in case of a tie during content negotiation.
#
# Just list the languages in decreasing order of preference. We have
# more or less alphabetized them here. You probably want to change this.
#
LanguagePriority en ca cs da de el eo es et fr he hr it ja ko ltz nl nn no pl pt pt-BR ru sv tr zh-CN zh-TW

#
# ForceLanguagePriority allows you to serve a result page rather than
# MULTIPLE CHOICES (Prefer) [in case of a tie] or NOT ACCEPTABLE (Fallback)
# [in case no accepted languages matched the available variants]
#
ForceLanguagePriority Prefer Fallback

#
# Commonly used filename extensions to character sets. You probably
# want to avoid clashes with the language extensions, unless you
# are good at carefully testing your setup after each change.

```

---

```

# See http://www.iana.org/assignments/character-sets for the
# official list of charset names and their respective RFCs.
#
AddCharset ISO-8859-1 .iso8859-1 .latin1
AddCharset ISO-8859-2 .iso8859-2 .latin2 .cen
AddCharset ISO-8859-3 .iso8859-3 .latin3
AddCharset ISO-8859-4 .iso8859-4 .latin4
AddCharset ISO-8859-5 .iso8859-5 .latin5 .cyr .iso-ru
AddCharset ISO-8859-6 .iso8859-6 .latin6 .arb
AddCharset ISO-8859-7 .iso8859-7 .latin7 .grk
AddCharset ISO-8859-8 .iso8859-8 .latin8 .heb
AddCharset ISO-8859-9 .iso8859-9 .latin9 .trk
AddCharset ISO-2022-JP .iso2022-jp .jis
AddCharset ISO-2022-KR .iso2022-kr .kis
AddCharset ISO-2022-CN .iso2022-cn .cis
AddCharset Big5 .Big5 .big5
# For russian, more than one charset is used (depends on client, mostly):
AddCharset WINDOWS-1251 .cp-1251 .win-1251
AddCharset CP866 .cp866
AddCharset KOI8-r .koi8-r .koi8-ru
AddCharset KOI8-ru .koi8-uk .ua
AddCharset ISO-10646-UCS-2 .ucs2
AddCharset ISO-10646-UCS-4 .ucs4
AddCharset UTF-8 .utf8

# The set below does not map to a specific (iso) standard
# but works on a fairly wide range of browsers. Note that
# capitalization actually matters (it should not, but it
# does for some browsers).
#
# See http://www.iana.org/assignments/character-sets
# for a list of sorts. But browsers support few.
#
AddCharset GB2312 .gb2312 .gb
AddCharset utf-7 .utf7
AddCharset utf-8 .utf8
AddCharset big5 .big5 .b5
AddCharset EUC-TW .euc-tw
AddCharset EUC-JP .euc-jp
AddCharset EUC-KR .euc-kr
AddCharset shift_jis .sjis

#
# AddType allows you to add to or override the MIME configuration
# file mime.types for specific file types.
#
#AddType application/x-tar .tgz
#
# AddEncoding allows you to have certain browsers (Mosaic/X 2.1+) uncompress
# information on the fly. Note: Not all browsers support this.
# Despite the name similarity, the following Add* directives have nothing
# to do with the FancyIndexing customization directives above.
#
#AddEncoding x-compress .Z
#AddEncoding x-gzip .gz .tgz
#
# If the AddEncoding directives above are commented-out, then you
# probably should define those extensions to indicate media types:
#
AddType application/x-compress .Z
AddType application/x-gzip .gz .tgz

#
# AddHandler allows you to map certain file extensions to "handlers":
# actions unrelated to filetype. These can be either built into the server

```

---

```

# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased directories:
# (You will also need to add "ExecCGI" to the "Options" directive.)
#
#AddHandler cgi-script .cgi

#
# For files that include their own HTTP headers:
#
#AddHandler send-as-is asis

#
# For server-parsed imagemap files:
#
#AddHandler imap-file map

#
# For type maps (negotiated resources):
# (This is enabled by default to allow the Apache "It Worked" page
# to be distributed in multiple languages.)
#
AddHandler type-map var
#
# The following directives modify normal HTTP response behavior to
# handle known problems with browser implementations.
#
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4.0b2;" nokeepalive downgrade-1.0 force-response-1.0
BrowserMatch "RealPlayer 4\." force-response-1.0
BrowserMatch "Java/1\." force-response-1.0
BrowserMatch "JDK/1\." force-response-1.0
#
# The following directive disables redirects on non-GET requests for
# a directory that does not include the trailing slash. This fixes a
# problem with Microsoft WebFolders which does not appropriately handle
# redirects for folders with DAV methods.
# Same deal with Apple's DAV filesystem and Gnome VFS support for DAV.
#
BrowserMatch "Microsoft Data Access Internet Publishing Provider" redirect-carefully
BrowserMatch "MS FrontPage" redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
BrowserMatch "^WebDAVFS/1.[0123]" redirect-carefully
BrowserMatch "^gnome-vfs" redirect-carefully
BrowserMatch "^XML Spy" redirect-carefully
BrowserMatch "^Dreamweaver-WebDAV-SCM1" redirect-carefully
#
# Bring in additional module-specific configurations
#
<IfModule mod_ssl.c>
    Include conf/ssl.conf
</IfModule>
#### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them. Most configurations
# use only name-based virtual hosts so the server doesn't need to worry about
# IP addresses. This is indicated by the asterisks in the directives below.
#
#</VirtualHost>
LoadModule jk_module modules/mod_jk.so
JkWorkersFile C:\ApacheGroup\Apache2\conf\workers.properties
JkLogLevel debug
JkLogFile logs/mod_jk.log
Alias /src932 C:\tomcat1\webapps\src932

```

```
JkMount /src932/*.jsp loadbalancer
JkMount /src932/secure/* loadbalancer
JkMount /src932/messagebroker/* loadbalancer
JkMount /src932/configuration/* loadbalancer
JkMount /src932/* loadbalancer
```

```
<Location "/src932/WEB-INF/">
AllowOverride None
deny from all
</Location>
```

**Note:** Make sure all the roots, directories, and alias in httpd.conf file are configured for the correct paths to your Apache HTTP Server installation directory.

After these configuration steps are complete, Apache can function as a software load balancer for multiple SRC instances.

### Deploy and Configure Multiple SRC Application Instances

To deploy and configure multiple SRC application instances, follow these steps:

1. Follow the steps in section Deploy and Configure the SRC Instance to deploy the first SRC instance into Apache Tomcat. For example, specify the path of the first SRC instance as the following:  

```
C:\tomcat1\webapps\src932\
```
2. Find the server.xml file in the C:\tomcat1\conf directory and adapt the contents as shown in the following example:

---

#### *Step 1: change parameter 'Connector prot' to avoid conflicts*

```
<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at:
Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL HTTP/1.1 Connector on port 8080
-->
<Connector port="8081" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />
```

#### *Step 2: definition for AJP 1.3*

```
<!-- Define an AJP 1.3 Connector on port 8001 -->
<Connector port="8001" protocol="AJP/1.3" redirectPort="8443" />
```

#### *Step 3: configure Catalina engine*

```
<!-- An Engine represents the entry point (within Catalina) that processes
every request. The Engine implementation for Tomcat stand alone
analyzes the HTTP headers included with the request, and passes them
on to the appropriate Host (virtual host).
Documentation at /docs/config/engine.html -->
<Engine name="Catalina" defaultHost="localhost" jvmRoute="tomcat1">
```

#### *Step 4: configure Host and Context*

```
<Host name="localhost" appBase="webapps"
unpackWARs="true" autoDeploy="true">
<Context docBase="C:\tomcat1\webapps\src-9.32" path="/src-9.32" reloadable="false"/>
```

---

3. Modify the Catalina.bat file in the C:\tomcat1\bin directory by setting the CATALINA\_HOME parameter to the first Tomcat directory:

---

```
rem Guess CATALINA_HOME if not defined
```

```

set "CURRENT_DIR=%cd%"
set CATALINA_HOME=C:\tomcat1

if not "%CATALINA_HOME%" == "" goto gotHome
set "CATALINA_HOME=%CURRENT_DIR%"
if exist "%CATALINA_HOME%\bin\catalina.bat" goto okHome

cd ..
set "CATALINA_HOME=%cd%"
cd "%CURRENT_DIR%"
:gotHome
    
```

---

4. Deploy a second SRC instance. Copy the tomcat1 folder, where the first SRC instance is deployed to desktop and rename it 'tomcat2'. Then, copy it back to the same directory as tomcat1. Modify the server.xml and catalina.bat files as shown in the following example for the second tomcat instance.
- 

**Step 1:** Change 'Connector port' parameter to avoid conflicts

```

<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at :
Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL HTTP/1.1 Connector on port 8080
-->
    
```

```

<Connector port="8082" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />
    
```

**Step 2:** Specify the AJP 1.3 protocol

```

<!-- Define an AJP 1.3 Connector on port 8002 -->
    
```

```

<Connector port="8002" protocol="AJP/1.3" redirectPort="8443" />
    
```

**Step 3:** Configure the Catalina engine

```

<!-- An Engine represents the entry point (within Catalina) that processes
every request. The Engine implementation for Tomcat stand alone
analyzes the HTTP headers included with the request, and passes them
on to the appropriate Host (virtual host).
Documentation at /docs/config/engine.html -->
    
```

```

<Engine name="Catalina" defaultHost="localhost" jvmRoute="tomcat2">
    
```

**Step 4:** Configure the Host and Context

```

<Host name="localhost" appBase="webapps" unpackWARs="true" autoDeploy="true">
    
```

```

<Context docBase="C:\tomcat2\webapps\src932" path="/src932" reloadable="false"/>
    
```

**Modify the Catalina.bat file for SRC Instance 2:**

```

rem Guess CATALINA_HOME if not defined
set "CURRENT_DIR=%cd%"
set CATALINA_HOME=C:\tomcat2
if not "%CATALINA_HOME%" == "" goto gotHome
set "CATALINA_HOME=%CURRENT_DIR%"
if exist "%CATALINA_HOME%\bin\catalina.bat" goto okHome
cd ..
set "CATALINA_HOME=%cd%"
cd "%CURRENT_DIR%"
:gotHome
    
```

---

**Note:** Refer to above configuration steps to deploy more SRC instances.

### Verify the Integration

After you complete the configuration described in the Deploy and Configure Apache HTTP Server (httpd) and Deploy and Configure Multiple SRC Application Instances chapters, verify whether the Apache HTTP Server is working as a software load balancer with multiple SRC instances. To do this, follow these steps:

1. Restart all the SRC instances.

2. Start the Apache HTTP Server. In a Command Prompt window, change to the <APACHE\_HOME>/bin directory, and then execute the following command: `apache -k start`

Open Internet Explorer and access the SRC UI Web service by using an appropriate URL, such as the following:  
<http://localhost:8088/src932/secure/main.jsp>

**Note:** In the access.log file, in the ./logs path of the Apache HTTP Server, you may view the details of the request and response status.

## Sizing Guide

This section provides sizing guide information to SRC 9.32 customers to achieve a satisfied user experience based on performance testing with 400 concurrent users. The principles in this guideline are based on the critical data that has real impact on Service Request Catalog performance and the tests conducted by HP SRC Performance team. Recommendations in section Sizing Reference Recommended results from Benchmark Performance Testing and the data volume in Service Manager Database are 1 million interactions.

## Benchmark Performance Testing

### Lab Topologic Architecture

This performance testing is conducted on the following lab topologic architecture shown in 'Figure -3 Lab Topology':

A Cisco hardware load balancer balances user requests from the SRC instance to three different Service Manager instances and one KM Search Engine with persistence. The number of nodes that each Service Manager server can support depends on the server configuration.

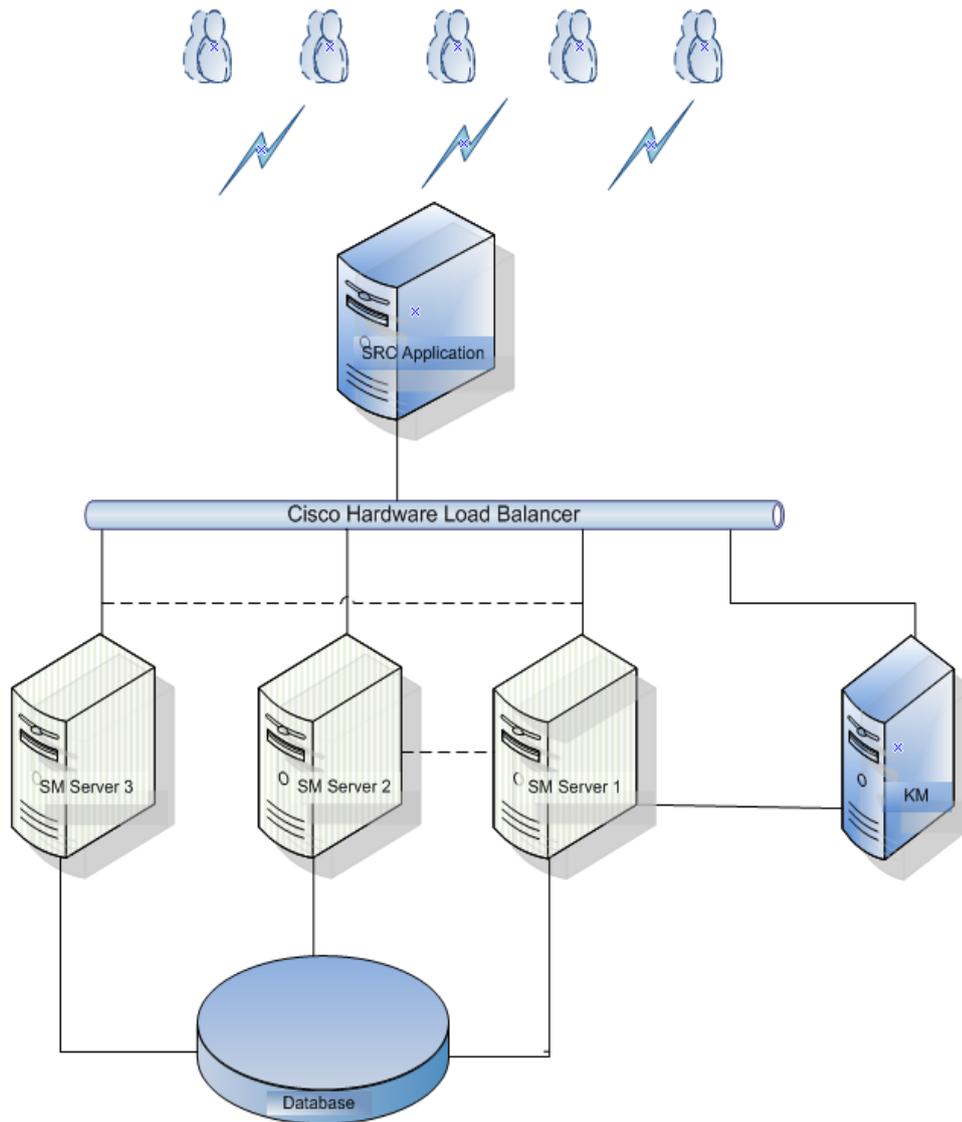


Figure -3 Lab Topology

**Note:** The Servers used in this performance testing are all virtual machines, and the network latency (round trip) is less than 1ms.

The following table describes lab configurations:

SRC Server	Configuration
Windows	Windows Server 2008 R2 Enterprise
System	64-bit Operation System
Memory	16.0 GB
Processor	Intel(R)Xeon(R)CPU E7-L8867 @2.13GHz 2.13GHz
CPU	8
SM Server -1	Configuration
Windows	Windows Server 2008 R2 Enterprise
System	64-bit Operation System
Memory	24.0 GB
Processor	Intel(R)Xeon(R)CPU E7-L8867 @2.13GHz 2.13GHz
CPU	8
SM Server -2	Configuration

Windows	Windows Server 2008 R2 Enterprise
System	64-bit Operation System
Memory	32.0 GB
Processor	Intel(R)Xeon(R)CPU E7-L8867 @2.13GHz 2.13GHz
CPU	8
<b>SM Server -3</b>	<b>Configuration</b>
Windows	Windows Server 2008 R2 Enterprise
System	64-bit Operation System
Memory	32.0 GB
Processor	Intel(R)Xeon(R)CPU E7-L8867 @2.13GHz 2.13GHz
CPU	8
<b>Database Server</b>	<b>Configuration</b>
Windows	Windows Server 2008 R2 Enterprise
System	64-bit Operation System
Memory	96.0 GB
Processor	Intel(R)Xeon(R)CPU X5670 @2.93GHz 2.93GHz
CPU	24
<b>KM Server</b>	<b>Configuration</b>
Windows	Windows Server 2008 R2 Enterprise
System	64-bit Operation System
Memory	16.0 GB
Processor	Intel(R)Xeon(R)CPU E7-L8867 @2.13GHz 2.13GHz
CPU	8

#### JDK and Apache Tomcat specifications

1. JDK 1.7.0\_15, 64-bit
2. Apache Tomcat 7.0.26, 64-bit
3. Threads settings in server.xml: maxThreads="600" minSpareThreads="100" acceptCount="575" connectionTimeout="180000" maxKeepAliveRequests="1000"

#### Dataset

Testing is base on 1,000,000 interactions, 6,000 Service Categories, more than 20,000 catalog items, 10000 subscriptions, 300,000 users with Operator, SDA, ESS roles, and more than 130,000 knowledge management documents in an Oracle 11.2.0 database. The critical data types in regards to Service Request Catalog performance are the service category, catalog items, interactions, km documents and users.

Data Type	Volumn	Criteria
interaction	1,000,000	1. Login transaction server response time is not more than 10s.
service catalog	6,000	
service catalog item	27,000	
change	100,000	2. Typical transaction server response time is not more than 5s.
quote	100,000	
km documents	133,000	3.KM Search Engine server response time is not more than 1s.
km attachments	6,000	
Subscription	10,000	
Operator	5000	
SDA	5000	
ESS	30,000	

## **SRC Settings**

*applicationContext.properties:*

```
src.sm.defaultMaxConnectionsPerHost=60
src.sm.maxTotalConnections=60
src.refreshCatalogAfterEvery=36000
src.km.enabled=true
km.defaultMaxConnectionsPerHost=60
km.maxTotalConnections=60
sm.km.doctype.expireDuration=3600
src.requestedFor.editable=true
src.security.mode=default
ehcache.disk.store.dir=E:/src-cache
```

## **SM Settings**

*Primary host*

*sm.ini*

```
grouplicenseip:16.155.197.39
groupname:sm932hwlb
groupmcastaddress:224.0.0.225
groupsubnetaddress:255.255.255.0
groupport:13081
groupbindaddress:16.155.197.39
ir_asynchronous:1
JVMOption999:../lib/mbeanclient-9.32.war
webservices_sessiontimeout:180
Threadsperprocess:100
sessiontimeout:3
Heartbeatinterval:5
maxKeepAliveRequests:100
dbcachequery:extaccess
JVMOption0:-Xms512M
JVMOption1:-Xmx512M
```

*sm.cfg*

```
sm -httpPort:62271 -log:../logs/sm.62271.log
sm -httpPort:62272 -log:../logs/sm.62272.log
sm -httpPort:62273 -log:../logs/sm.62273.log
sm -httpPort:62274 -log:../logs/sm.62274.log
sm -httpPort:62275 -log:../logs/sm.62275.log
```

*Secondary Host*

*sm.ini*

```
grouplicenseip:16.155.197.39
groupname:sm932hwlb
groupmcastaddress:224.0.0.225
groupsubnetaddress:255.255.255.0
groupport:13081
```

```
groupbindaddress:16.155.194.36
JVMOption999:../lib/mbeanclient-9.32.war
webservices_sessiontimeout:180
Threadsperprocess:100
sessiontimeout:3
Heartbeatinterval:5
maxKeepAliveRequests:100
dbcachequery:extaccess
JVMOption0:-Xms512M
JVMOption1:-Xmx512M
```

*sm.cfg*

```
sm -httpPort:62281 -log:../logs/sm.62281.log
sm -httpPort:62282 -log:../logs/sm.62282.log
sm -httpPort:62283 -log:../logs/sm.62283.log
sm -httpPort:62284 -log:../logs/sm.62284.log
sm -httpPort:62285 -log:../logs/sm.62285.log
sm -sync
```

### **Sizing Reference Recommended**

- We recommend that the size of the attachment a user uploads while submitting service or support requests is not larger than 30M. Otherwise, users may encounter much longer uploading times and/or unexpected exceptions.
- The hard disk on each SRC server should meet the actual requirement of storing items from users. Therefore we recommend a minimum of 60 GB of free space.
- We recommend that you adjust the max threads in server.xml of Apache Tomcat according to the number of potential customers.

## Support

You can visit the HP Software support web site at: [www.hp.com/go/hpssoftwaresupport](http://www.hp.com/go/hpssoftwaresupport)

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require an active support contract. To find more information about support access levels, go to the following URL:

[http://h20230.www2.hp.com/new\\_access\\_levels.jsp](http://h20230.www2.hp.com/new_access_levels.jsp)

To register for an HP Passport ID, go to the following URL:

<http://h20229.www2.hp.com/passport-registration.html>

**Sign up for updates**  
[hp.com/go/getupdated](http://hp.com/go/getupdated)

  
Share with colleagues

  
Rate this document

