

HP Service Manager

Software Version: 9.41

For the supported Windows® and UNIX® operating systems

Solr Search Engine Guide

Document Release Date: September 2015
Software Release Date: September 2015



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notice

© 1994-2015 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

For a complete list of open source and third party acknowledgements, visit the HP Software Support Online web site and search for the product manual called HP Service Manager Open Source and Third Party License Agreements.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: <https://softwaresupport.hp.com/>.

This site requires that you register for an HP Passport and to sign in. To register for an HP Passport ID, click **Register** on the HP Support site or click **Create an Account** on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support site at: <https://softwaresupport.hp.com>.

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

HP Software online 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 website 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 to sign in. Many also require a support contract. To register for an HP Passport ID, click **Register** on the HP Support site or click **Create an Account** on the HP Passport login page.

To find more information about access levels, go to: <https://softwaresupport.hp.com/web/softwaresupport/access-levels>.

HPSW Solutions Catalog accesses the HPSW Integrations and Solutions Catalog portal website. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this website is <https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01702710>.

About this PDF Version of Online Help

This document is a PDF version of the online help. This PDF file is provided so you can easily print multiple topics from the help information or read the online help in PDF format. Because this content was originally created to be viewed as online help in a web browser, some topics may not be formatted properly. Some interactive topics may not be present in this PDF version. Those topics can be successfully printed from within the online help.

Contents

- Chapter 1: Introduction to the Solr Search Engine Guide 6
- Chapter 2: Overview of the Solr Search Engine 7
- Chapter 3: Upgrading from the K2 Search Engine 10
- Chapter 4: Installing the Solr Search Engine 11
 - Meet the Solr Search Engine Requirements 11
 - Install the Solr Search Engine 12
 - Uninstall the Solr Search Engine 14
 - Before You Start the Solr Search Engine 15
 - Start and Stop the Solr Search Engine 16
- Chapter 5: Managing Knowledgebase Search Servers 19
 - Recommended Search Server Configurations 19
 - Add a Virtual Search Server 21
 - Verify Knowledgebase Search Server Connectivity 26
 - Specify a Primary Searcher 27
 - Specify a Search Server for Each Knowledgebase 27
- Chapter 6: Configuring the Solr Search Engine 29
 - Edit the Knowledge Management Environment Record 30
 - Managing Knowledgebases 32
 - Add an sclib Knowledgebase 32
 - Add a weplib Knowledgebase 45
 - Edit the List of MIME Types 52
 - Add an fsyslib Knowledgebase 52
 - Delete a Knowledgebase 58
 - Configure Advanced Search for Knowledge Management 60
 - Enabling Languages for KM Search 61

Supported Languages for the Solr Search Engine	61
Activate a Knowledge Management language	63
Enable Languages in the Solr Search Engine	64
Create Search Engine Thesaurus Files	67
Modify Stop Words	69
Add a New KM Message to the scmessage Table	69
Create a Hitlist with Multilingual Labels	71
Chapter 7: Indexing the Knowledgebases	72
Perform a Full Re-Index on a Knowledgebase	74
Chapter 8: Enforcing Mandanten Security in Knowledge Management	77
Update a KM Search Security Script for Mandanten Security	78
Chapter 9: Searching the Knowledgebases	85
Send Documentation Feedback	86

Chapter 1: Introduction to the Solr Search Engine Guide

Note: Starting with version 9.41, Service Manager supports both the Solr Search Engine and the IDOL Search Engine for Knowledge Management (KM) search. The IDOL Search Engine is available only when you have Smart Analytics installed and enabled. Once Smart Analytics is enabled, the Solr Search Engine is disabled. For information about how to install and configure Smart Analytics and use Smart Search, see the *Smart Analytics Administrator and User Guide*.

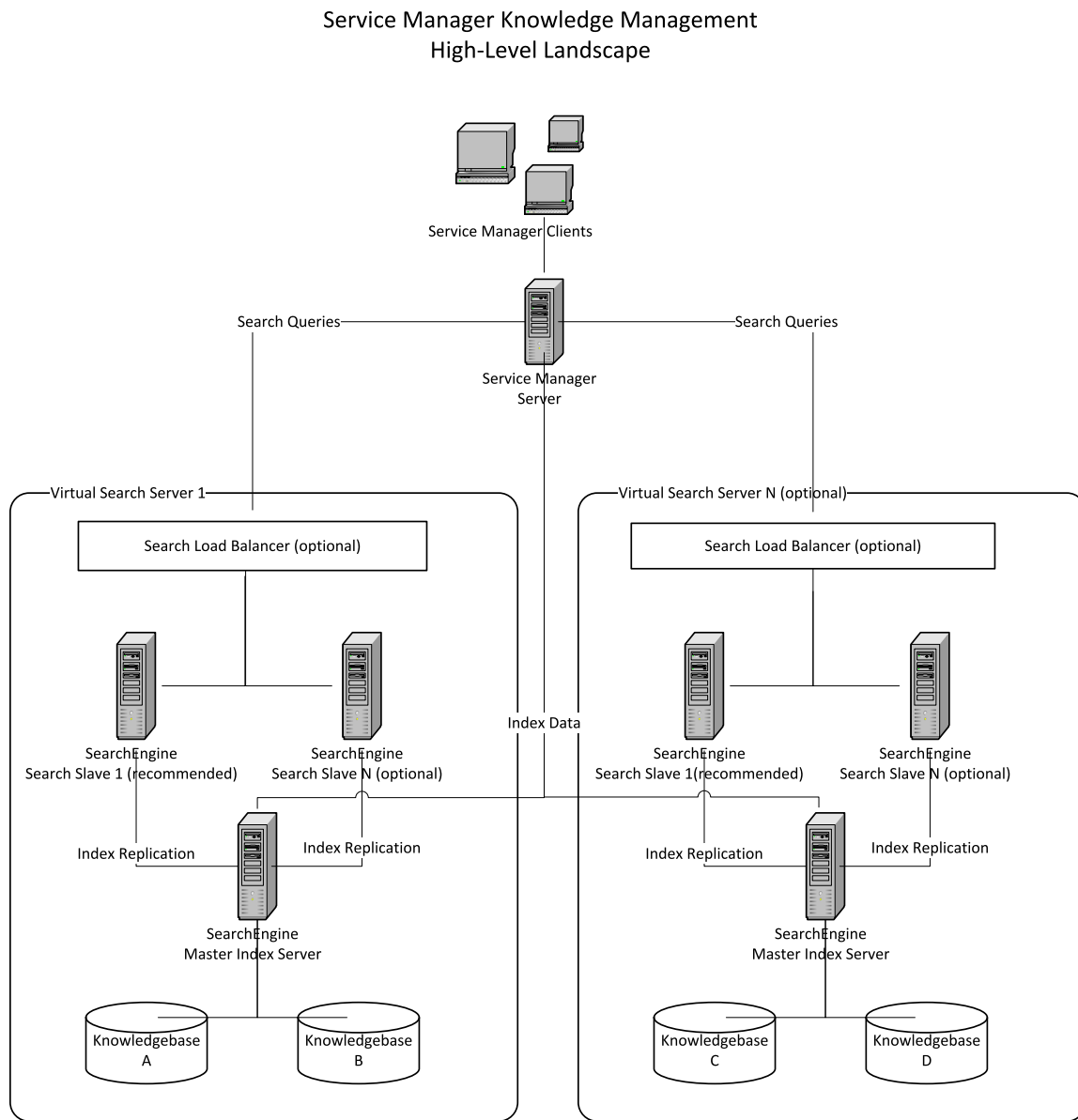
This guide describes how to install the Solr-based Knowledge Management search engine, set up search servers, configure the search engine, and re-index knowledgebases. It also describes how to enforce Mandanten security in the Knowledge Management module where the Solr search engine is used.

Chapter 2: Overview of the Solr Search Engine

The Solr-based search engine enables the Knowledge Management module to index over 1,200 unique file formats, including the latest versions of Microsoft Office and OpenOffice formats, PDF, HTML/XML, compression, image, audio, etc.

Due to its flexible architecture, the Solr search engine provides scalability and improves indexing performance by supporting the use of multiple index servers. It supports high availability architectures, which include decoupling of search servers from index servers, replication of the search server to multiple servers, and the addition of a load balancer across multiple search servers; in addition, it can provide fail-safe capabilities, such as the creation of a second index server or search server for failover and the ability to switch to a backup server immediately without having to restart or log out and log back in to your Service Manager server.

The following diagram illustrates an example Service Manager Knowledge Management high-level landscape.



A Knowledge Management search server is composed of three parts: an indexer, a searcher, and a crawler. These server parts are responsible for the following:

- **Indexer:** Indexes documents into searchable data
- **Searcher:** Provides results to users' search requests
- **Crawler:** Indexes the file system and web content

Knowledgebases are assigned to a Knowledge Management search server. If you have a single search server definition, all knowledgebases will be indexed and searched according to that configuration.

When one or more slave servers are defined, all knowledgebases assigned to this virtual search engine will be replicated to each slave. Replication happens when a knowledgebase is re-indexed or the index is updated. Depending on the size of the knowledgebases, they may not be immediately searchable while the replication process is running. If this is a new slave server, you will have to wait for the replication process to finish before you are able to search. Subsequent updates or re-indexes will happen in the background. The slave server will continue to serve search requests on the old knowledgebase until the updated knowledgebase comes on line. It will then automatically begin serving search requests against the new knowledgebase.

Supported Platforms

The search engine runs on multiple platforms, with the same server compatibility as Service Manager.

Language Support

Due to its up-to-date technology, the search engine offers improved Asian language support. Thesaurus maintenance is a lot easier compared to the K2 search engine, because it can now be done through text-based editing. For more information, see ["Supported Languages for the Solr Search Engine" on page 61](#) and ["Create Search Engine Thesaurus Files" on page 67](#).

Upgrade

Upgrading from a custom legacy search engine to the cutting edge Solr search engine is invisible to the end-user, and the administrator only needs to assign the new Solr search servers to the existing knowledgebases and re-index. The Search Engine Management has been greatly simplified – there is no more need for mapped drives and complex environment records. For more information, see ["Upgrading from the K2 Search Engine" on page 10](#).

Flexible Installation of File/Web Crawlers

The File/Web Crawlers are no longer chained to the search engine, and can be located separately, with many new website formats supported.

Chapter 3: Upgrading from the K2 Search Engine

The Service Manager 9.41 client and server support only the Service Manager 9.3x Applications, which can only work with the Solr Search Engine.

Once you have upgraded your server and clients to Service Manager 9.41, you must uninstall the K2 Search Engine and install the Solr Search Engine.

Once you run the applications upgrade for Service Manager 9.41, you will lose support for the K2 Search Engine, which affects script libraries, menus, the search library (including advanced search), and how you manage knowledgebases.

To upgrade to the Solr Search Engine, you will need to do the following:

1. Install the Solr Search Engine.
 - a. ["Meet the Solr Search Engine Requirements" on page 11](#)
 - b. ["Install the Solr Search Engine" on page 12](#)
2. Configure Service Manager to connect to the new Solr Search Engine and KM Web Crawler by entering the host names, ports, etc., as the old connection information will not work. See ["Managing Knowledgebase Search Servers" on page 19](#).
3. Re-index all of your knowledgebases, as the old indexes will not work. See ["Perform a Full Re-Index on a Knowledgebase" on page 74](#).

Tip: If your knowledgebases contain large amounts of data, re-index them before going into production (live) mode.

4. If you have any tailored forms, follow the normal Service Manager applications tailoring procedures to update the new versions. For more information, see the *HP Service Manager Tailoring Best Practices Guide*.
5. If you have modified your dbdict, merge your dbdict changes.

Note: If you do not have the administrative experience necessary to manage migrating to the Solr Search Engine, you should get assistance from your local application developers and database administrators.

Chapter 4: Installing the Solr Search Engine

The section describes the steps to install a single KM Solr Search Engine instance.

Note: While a single instance may be suitable for pre-production testing, you may want to install multiple instances for a more robust production system. For information about recommended search server configurations, see "[Managing Knowledgebase Search Servers](#)" on page 19.

Meet the Solr Search Engine Requirements

Ensure that the target system complies with the installation requirements, as listed in the *Service Manager 9.41 Support Matrix* on the HP Support Matrices web site (<https://softwaresupport.hp.com/group/softwaresupport/support-matrices>).

HP recommends using the following configurations for the Solr Search Engine:

- RAM: Minimum of 8GB with 4GB dedicated to the JVM that hosts the Solr Search Engine. For better performance, 16GB of RAM with 8GB dedicated to the JVM that hosts the Solr Search Engine.

Note: It is important to consider the size of the indexes when allocating RAM. Performance is greatly improved if there is enough RAM available to the OS, so that all the index files can be easily cached by the OS as disk seeks to load stored fields and other data from the KM index files, which can slow performance. With 4GB of RAM for the OS, an index of approximately 3GB in size could be cached easily. However, if that number is doubled to 8GB of RAM for the OS, a 6GB or 7GB index could be cached.

- The servers should ideally have at least four processors (no less than two).
- 800 MHz or higher processor
- 400 MB of disc space designated for /tmp (UNIX) and \TEMP (Windows)

Install the Solr Search Engine

Running the Knowledge Management (KM) Search Engine installer will automatically install the Solr Search Engine, KM Web Crawler, and an embedded Apache Tomcat Server. The Solr Search Engine and KM Web Crawler can be installed separately.

The Solr Search Engine is highly configurable. While a single instance may be suitable for pre-production testing, you may want to install more instances on several machines for a more robust production system. For recommendations on production system configurations, see "[Managing Knowledgebase Search Servers](#)" on page 19.

To install the Service Manager 9.41 Solr Search Engine, you need to perform the following tasks.

Task 1. Install the Service Manager 9.40 Solr Search Engine.

To install the Service Manager 9.40 Solr Search Engine using the installation wizard, follow these steps:

1. Insert the Solr Search Engine installation DVD into the appropriate drive of the server.
2. Start the Installer:
 - o To start the installer from Internet Explorer:
 - i. If you are installing on a system that has auto-run enabled, the DVD browser starts automatically.
If auto-run is disabled, start the DVD browser manually by navigating to the DVD directory and opening ClickMe.html.
 - ii. For Windows, click **Knowledge Management > HP Service Manager Knowledge Management Search Engine for Windows.**

For Unix, click **Knowledge Management > HP Service Manager Knowledge Management Search Engine for Linux.**

Note: This installer is for all supported Unix (Linux, AIX, HP-UX, and Solaris) platforms.

- To start the installer from the command prompt:
 - i. Navigate to the \KnowledgeManagement directory on the installation media.
 - ii. Run the installer.
 - **For Windows:** At the prompt, type **kmsetup.exe**, and press Enter to start the installation wizard.
 - **For Unix:** At the prompt, type: **sh ./kmsetup.bin** and press Enter to start the installation wizard.
 - iii. The download page opens. Download the installer for your platform.
 - iv. After downloading, change directory to the download directory. The installation wizard starts.
- 3. From the Choose Locale dialog box, choose a language for the installation program. (The default is English).
- 4. Click **OK**.
- 5. Click **Next**. The license agreement is displayed.
- 6. Once you have read and agree to the terms of the license agreement, select **I accept the terms of the License Agreement** and then click **Next**.
- 7. Choose one of the following installation sets and then click **Next**:
 - **Typical:** Installs the Solr Search Engine, KM Web Crawler, and Tomcat Server.
 - **Solr Search Engine:** Installs the Solr Search Engine only with Tomcat Server.
 - **KM Crawler:** Installs the KM Crawler only with Tomcat Server. If you do not plan to crawl file systems or web content, you do not need to install the crawler. The crawler can also be installed by itself on another machine to reduce resource consumption between the crawling process and the indexer.
- 8. Choose an Installation folder and then click **Next**. The default installation directory opens. For example: C:\Program Files (x86)\HP\Service Manager 9.40\SearchEngine or /opt/HP/ServiceManager9.40/SearchEngine.

Note: If necessary, click **Choose** to choose a different location.

9. On the Tomcat Port Selection screen, enter the following port numbers and then click **Next**:
 - Tomcat Port number (the default is 8080)
 - Tomcat Shutdown Port number (the default is 8005)

Note: If the default ports are in use on the server host, you need to use other ports (for example, 8180 and 8105). Record these settings, as you will need them to configure Service Manager to communicate with the Solr Search Engine. See ["Add a Virtual Search Server" on page 21](#).

10. A pre-installation summary displays the following information:
 - Product name selected for installation
 - Installation folder
 - Disk space information for installation target with required bytes and available bytes
11. Click **Install**. The installation begins.
12. The Solr Search Engine is installed. Click **Done** to exit the installer.

Task 2. Apply the Service Manager 9.41 Solr Search Engine patch.

For detailed steps, see the Service Manager 9.41 Release Notes, which you can download from the following website:

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM01384297>

Uninstall the Solr Search Engine

The Windows and UNIX uninstall folder and program are the same. You can uninstall the Solr Search Engine as follows:

1. In the <Service Manager Installation Directory>\Search_Engine directory, select **Search_Engine_Uninstall**.
2. Click **Change_or_uninstall.exe**. A message displays, stating that the HP Solr Search Engine 1.00 and its features will be removed, except files and folders created after the installation.

3. Click **Next** and then select one of the following options:
 - **Complete Uninstall:** To remove all features and components of the HP Solr Search Engine 1.00 that were installed by the installer, except files and folders created after the installation
 - **Uninstall Specific Features:** To remove specific features of the HP solr Search Engine 1.00 that were installed by the installer.
4. Click **Next**. The uninstaller begins to remove the features. When the uninstall is complete, a list of the files that have been removed is displayed.

The uninstall process intentionally preserves the files that have changed since the initial installation. You must manually remove these files if you want to completely uninstall the Solr Search Engine from your system.

Before You Start the Solr Search Engine

Once you have installed a Solr Search Engine instance, you need to start it. Before doing so, you need to do the following:

1. Create new system variables on the Search Engine server host:
 - **Variable:** JAVA_HOME
Value: = <home folder of JDK>. For example: C:\Program Files\Java\jdk1.8.0_31

Note: The Service Manager 9.41 Solr Search Engine requires JDK 1.8 (Update 51 or greater). The latest JDK 1.8 is recommended.

- **Variable:** JAVA_TOOL_OPTIONS
Value: -Dfile.encoding=UTF8
2. Add one of the following lines to the sm.ini file, found in the <Service Manager install>\Server\RUN directory.

plugin0: kmpugin.dll (if the SM server is running on Windows)

plugin0: libkmpugin.so (if the SM server is running on Unix)

3. If your sm.ini file already contains the following line, remove it.

```
KMSearchEngineClass:com.hp.ov.sm.server.plugins.knowledgemanagement.solr.KMSolrSearch
```

Note: This parameter is no longer needed for the Service Manager 9.41 or later Solr Search Engine. If it is present in the `sm.ini` file, a warning message will occur in the Service Manager server log (`sm.log`).

4. Once you have modified the `sm.ini` file, restart the Service Manager server.

Start and Stop the Solr Search Engine

You can start and stop the Solr Search Engine by using the command line scripts or through a Windows Service.

Security best practices

Since the Search Engine uses Tomcat as the web server, be sure to follow the following best practices (see also the Apache documentation for information on Tomcat security best practices).

- On Windows operating systems, you need to create a user account that has been granted the **Log on as a service** right and has **Full Control** permissions to the search engine installation directory (default: `C:\Program Files\HP\ServiceManagerxx\SearchEngine`). Further, make sure the **HP KM Search Engine - Master** or **HP KM Search Engine - Slave** service is set to run as this user.
- On Unix systems, follow these steps:
 - a. Install the Search Engine as root.
 - b. Create a user to run the Search Engine. Suppose the user is `SearchEngineUser`.
 - c. Run the following command to change the owner of the Search Engine installation directory:

```
sudo chown -R SearchEngineUser Search_Engine_install_directory
```
 - d. Run the following command to start the Search Engine:

```
sudo -u SearchEngineUser Search_Engine_install_directory/startup.sh
```

For more information about the `startup.sh` script and running the Search Engine as a Windows service, see ["Start and stop the Search Engine using scripts" on the next page](#) and ["Start and stop the Search Engine as a Windows service" on the next page](#).

Start and stop the Search Engine using scripts

Run the following executable scripts in the Solr Search Engine installation folder to start and stop the Search Engine.

Search engine startup scripts

Name	Purpose
startup.cmd	Starts the master Solr Search Engine server on a Windows system. Starts the KMCrawler that has been installed separately.
startup.cmd slave	Starts the slave Solr Search Engine server on a Windows system.
startup.sh	Starts the master Solr Search Engine server on a UNIX system.
startup.sh slave	Starts the slave Solr Search Engine server on a UNIX system.

Search engine shutdown scripts

Name	Purpose
shutdown.cmd	<ul style="list-style-type: none">Shuts down the master Solr Search Engine server on a Windows system.Shuts down the KMCrawler if it was installed separately.
shutdown.cmd slave	Shuts down the slave Solr Search Engine server on a Windows system.
shutdown.sh	<ul style="list-style-type: none">Shuts down the master Solr Search Engine server on a UNIX system.Shuts down the KMCrawler if it was installed separately.
shutdown.sh slave	Shuts down the slave Solr Search Engine server on a UNIX system.

Start and stop the Search Engine as a Windows service

You can register and install the Search Engine as a Windows service, and then start and stop the Solr Search Engine as a Windows service.

Register and install the Search engine as a Windows service

Run the following scripts to register the Solr Search Engine as a Windows service. Once the Solr Search Engine is registered as a Windows service, then you can use startup and shutdown scripts through the

Windows service. The name of the Windows service in the Windows console is: **HP KM Search Engine**.

The following table lists the registration scripts for Windows systems.

Name	Purpose
installservice.cmd install master	Installs the master Solr Search Engine server as a Windows Service.
Installservice.cmd install slave <identifier>	Installs the slave server as a Windows Service.
installservice.cmd install	Installs the KMCrawler as a Windows Service when the KMCrawler has been installed separately.
installservice.cmd remove master	Removes the master Windows Service.
installservice.cmd remove slave <identifier>	Removes the slave Windows Service.
installservice.cmd remove	Removes the KMCrawler Windows Service.

Start and stop the Solr Search Engine as a Windows Service

1. From the Windows Start menu, select **Control Panel > Administrative Tools > Services**.
2. Select the service that you want to start, for example: **HP KM Search Engine - Master**, **HP KM Search Engine - Slave <identifier>**, or **HP KM Search Engine - Crawler**.
3. To start the service, right-click the service and select **Start**.
4. To stop the service, right-click the service and select **Stop**.

Once you have installed and started your search engine instances, you are ready to configure them and verify their connectivity in Knowledge Management. For more information, see "[Managing Knowledgebase Search Servers](#)" on page 19.

Chapter 5: Managing Knowledgebase Search Servers

Deploying a single search engine instance is not recommended for production environments. For information about recommended configurations for a production environment, see "[Recommended Search Server Configurations](#)" below.

Recommended Search Server Configurations

The following provides recommended configurations of the Solr Search Engine for a production environment.

Recommended configuration for improved performance

A search server with a master and a slave is the recommended minimum configuration for a production system. You must install a separate instance of the search engine on a separate port or a separate machine; you cannot define a slave server with the same parameters as the master server.

For improved system performance and to allow indexes to be replaced, you can separate the indexer and searcher onto two separate machines, as follows:

- On machine A, perform a full install of the Solr search engine.
- On machine B, install the searcher and indexer components (without the crawler).
- Add two servers for the search server: Machine A as the master server and machine B as a slave server.
- Set machine B as the primary searcher.

In this configuration, all indexing and crawling will be performed by machine A, and all searching will be performed by machine B. Indexes will automatically be replicated to machine B when changes are made to the original indexes on machine A, the master server.

See the following figure for an example of this configuration.

KM Search Servers

Server Name:

Hostname: Port: Server Type:

Primary Searcher	Hostname	Port	Server Type
false	mli6	8088	master
true	mli85	8089	slave

Recommended configuration for improved performance and failover

For better performance and to provide failover capabilities, add a second slave server and configure a load balancer to handle search requests for both machines A and B. In this scenario, you will have three machines. In this setup, all indexing and crawling will be performed on Machine A.

- Machine A will have a full install of the Solr search engine.
- Machines B and C will only need the searcher and indexer components of the install (without the crawler).
- Load balancer is installed and configured on machine B, machine C, or on another system.

Note: The Solr search engine does not ship with a load balancer.

- For the search server, you will add four servers:
 - Machine A as the master server.
 - Machines B and C as slave servers.
 - Load balancer is added as a reference and is set as the primary searcher. Search requests will be directed to the load balancer, which will then redirect search requests to either machine B or C, depending on the load. Machine B or C will perform the actual search task.

The configurations above can be replicated to cover any loads you may have. If you have large knowledgebases, you can create additional search servers of any configuration, and then assign the

knowledgebases individually to each search server. See ["Overview of the Solr Search Engine" on page 7](#) for a diagram of a Service Manager Knowledge Management high-level landscape.

For information about adding a search server cluster, see ["Add a Virtual Search Server" below](#).

Add a Virtual Search Server

User Role: System Administrator, KMAAdmin

The Knowledge Management search server form displays settings used for configuring the Knowledge Management search servers (or search engines) to provide connectivity to Service Manager for knowledgebase searching. The search engines can be installed on separate machines to enhance performance.

A virtual search server must contain one master server and may also contain several slave servers or a load balancer server. In most cases, a single virtual server should be sufficient for most organizations. However, your particular organization may choose to encapsulate your knowledgebases into multiple virtual servers for performance reasons or to prevent all knowledgebases from going down at the same time. See ["Overview of the Solr Search Engine" on page 7](#) for the diagram of an example virtual server configuration.

To add a virtual search server, add a master first and then add slave or load balancer servers to it as needed. The following describes the three types of search servers:

- **Master:** Indexing will always happen on this server. You can only have one master server per virtual search engine. This is the minimum server definition needed for a working virtual search engine configuration.
- **Slave:** Slave servers are optional. When defined and the Primary Searcher field is set to true, all search requests for knowledgebases assigned to this virtual search engine will be directed to this server.
- **Load Balancer:** Load balancers are optional. When defined and the Primary Searcher field is set to true, all search requests will be directed to this server. You also need to install a load balancer (an Apache web server) for load balancing of the virtual search server.

Note: For load balancing of the Solr search engine, only Apache has been tested and is currently supported by HP.

Primary Searcher

If a virtual server contains one master and only one slave, the slave must be the primary searcher. If a virtual server contains one master and more than one slave, a load balancer is needed, and the load balancer must be the primary searcher.

Search requests are always initially directed to this server. If the primary searcher is a load balancer, it will redirect the requests to the other servers in the virtual server group.

To add a new virtual search server:

1. From the navigator menu, select **Knowledge Management > Configuration > Configure Search Servers**.
2. In the **Server Name** field, enter the server name that specifies the name for the virtual search server - Knowledge Management search server. Choose a unique name to describe the search server, as in "Production" or "Development."

Caution: You cannot change server names once the record is added.

3. Click **Add**. The fields for the virtual search server will be displayed.

Note: All fields are required for adding or editing search server records.

4. Add a master to the new search server.
 - a. Complete the following fields:
 - **Hostname:** Enter the server name (or IP address) of the machine where the master server is installed.

Caution: Do not specify **localhost** as the Hostname of a search server, no matter whether it is a master, slave or load balancer.

- **Port:** The search engine requires an open port for communication. For most Tomcat web servers, the default port is **8080**. The web server listens to this port number for the search engine.

Note: Ensure that the port you select is not blocked or otherwise restricted by a firewall.

- **Server Type:** Select **Master**.

- b. Click **Add**. The master search server is added to the virtual search server.
- c. Select the master server from the table and then click the **Verify Server** button to send a test ping to that server. For more information, see ["Verify Knowledgebase Search Server Connectivity" on page 26](#).

Optionally, you can continue to add slave or load balancer servers to the search server.

5. Optionally, add a slave search server.

- a. In the **Server Name** and **Port** fields, enter the server name (or IP address) and port of the slave server host.

Caution: You should not designate any search server as localhost.

- b. Select **Slave** for **Server Type**.
- c. Click **Add**. The slave server is added to the table.
- d. Verify the slave server connectivity. See ["Verify Knowledgebase Search Server Connectivity" on page 26](#).

The following figure shows an example master and slave configuration.

KM Search Servers

Server Name:

Hostname: Port: Server Type:

Primary Searcher	Hostname	Port	Server Type
false	mli6	8088	master
true	mli85	8089	slave

6. Optionally, add a load balancer search server and configure load balancing:

- a. Make sure you have already added one master and at least two slave search servers, as described above.

Important: You must install a load balancer (for example, an Apache web server) of your own choice, as no load balancer is provided with Knowledge Management. You must choose a unique port for your load balancer; do not use a port defined for a master server or slave server. A minimum of two slave servers is recommended for load balancing. Do not load balance a single slave server with the master server. If the master server is re-indexing, the load balancer may send a search request to the master server and the search will fail.

The following example describes how you can configure load balancing for a virtual search server, using an Apache web server as a load balancer. This example assumes the following virtual search server configuration is used.

KM Search Servers

Server Name:

Hostname: Port: Server Type:

Primary Searcher	Hostname	Port	Server Type
false	mli2	8082	master
false	mli3	8083	slave
false	mli4	8084	slave
true	mli5	8080	Load Balancer

- b. Set one of the slaves as the Primary Searcher. For details, see ["Specify a Primary Searcher" on page 27.](#)
- c. Perform a full re-indexing of an existing knowledgebase. For details, see ["Perform a Full Re-Index on a Knowledgebase" on page 74.](#)

- d. Click **Search Knowledgebase**, select the indexed knowledgebase, and verify the search functionality is working fine.
- e. Install an Apache 2.x server on a machine that you want to use as the load balancer server.
- f. Modify the httpd.conf file in the conf directory to enable the proxy server.
 - i. Uncomment the following loadModule lines.

```
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_module modules/mod_proxy.so
```

- ii. Add the following proxy server configuration to the end of the file.

```
#example
ProxyRequests Off
ProxyPass /KMCores balancer://mycluster/
ProxyPassReverse /KMCores balancer://mycluster/
<Proxy balancer://mycluster>
BalancerMember http://mli3:8083/KMCores
BalancerMember http://mli4:8084/KMCores
</Proxy>
```

Where: **mycluster** is a descriptive name for the search server cluster, while **mli3:8083** and **mli4:8084** are the host names and ports of the slaves.

- g. Start the Apache server.
- h. Verify that the proxy server works fine for load balancing (http://mli5:8080/KMCores).
 - i. Add the load balancer server to the virtual search server.
 - i. In the **Server Name** and **Port** fields, enter the server name (or IP address) and port of the Apache server.
 - ii. Select **Load Balancer** for **Server Type**, and click **Add**. The load balancer server is added.
 - j. Set the load balancer server as the Primary Searcher. For details, see ["Specify a Primary Searcher" on page 27](#).

The load balancing configuration is complete.

To delete an existing server from a virtual search server (that is, a search server cluster):

1. Select the server from the table.
2. Click the **Delete Server** button.

Caution: You should not delete a Primary Search server. You need to select another server to handle search requests. Once the Primary Searcher server setting is changed to another server, you can delete the server.

To delete an existing virtual search server:

Delete a virtual search server only if there are no knowledgebases assigned to it. If no knowledgebases point to the server, you can delete the server.

1. On the **Knowledge Management > Configuration > Knowledgebases** screen, change the Search Server Name to an alternate server.
2. Click **Knowledge Management > Configuration > Configure Search Servers**.
3. Click **Search**.
4. Select the virtual search server you want to delete.
5. Click **Delete**.

Verify Knowledgebase Search Server Connectivity

User Role: System Administrator, KMAdmin

When a search server is added or edited, you need to test whether Service Manager is communicating with the server.

To verify the knowledgebase search server is communicating with Service Manager:

1. From the navigator menu, select **Knowledge Management > Configuration > Configure Search Servers** and then click **Search**.
2. Select the server name you want to verify.
3. Click **Verify Server**. If the search server is correctly set up, an information window displays with the following message: **Search Engine Connected**.

Once the server settings are verified, you may index and search the knowledgebases.

Specify a Primary Searcher

User Role: System Administrator, KMAdmin

When you have configured a search server with a master and one or more slaves/load balancers, you need to specify one of these servers as a Primary Searcher. All search requests for knowledgebases assigned to this virtual search engine will be redirected to this server. If the primary searcher server goes down, you can set the Primary Searcher field to 'true' on another server to restore search functionality. For more information on search server configuration considerations, see "[Recommended Search Server Configurations](#)" on page 19.

To specify a primary searcher:

1. Click **Knowledge Management > Configuration > Configure Search Servers**.
2. Click **Search**. A list of search servers (if configured) is displayed.
3. Double-click a search server from the list to open the record.

Note: If the search server contains only a master, its Primary Searcher field is set to true by default; if the search server contains a master and one or more slaves/load balancers, the master is by default specified as a primary searcher, but you can change the default setting.

4. Select a server from the record, click the **Set Primary Search Server** button. The server's **Primary Searcher** field changes from **false** to **true**.
5. Click **Save** to save the search server record.

Specify a Search Server for Each Knowledgebase

User Role: System Administrator, KMAdmin

All default knowledgebases will map to the search server that is added first. An administrator can add a second search server to point to an additional search engine installation, if needed. Individual knowledgebases can then be separated onto different search servers.

To specify a search server for a knowledgebase:

1. Navigate to **Knowledge Management > Configuration > Knowledgebases**, and click **Search**.
2. Select a knowledgebase from the list.
3. In the Search Server field, select a search server.
4. Click **Save**.

You do not have to log out or restart your Service Manager server for the changes to take effect.

Once you have assigned a search server for each knowledgebase, you are ready to perform a full re-indexing of the knowledgebases. For more information, see ["Indexing the Knowledgebases" on page 72](#).

Chapter 6: Configuring the Solr Search Engine

Before you can use the Knowledge Management (KM) Solr Search Engine, you will need to set up the KM environment, configure the Search Engine, create indexes, and configure the Nutch Web Crawler.

Important: To configure the KM Solr Search Engine and Nutch Web Crawler, you should be an experienced System Administrator who is familiar with your installation.

The following checklist outlines the tasks that you should complete to configure the KM Solr Search Engine and Nutch Web Crawler:

1. Edit the Knowledge Management environment record if needed. See ["Edit the Knowledge Management Environment Record" on the next page](#).
2. Set up and verify server connectivity for multiple servers. Since the KM Solr search engine and Nutch Web Crawler can be installed on multiple servers, you need to set up connectivity for all servers.
 - Configure a search server cluster. For information on the master + slave + load balancer architecture, see ["Managing Knowledgebase Search Servers" on page 19](#).
 - List all servers in a cluster. See ["Add a Virtual Search Server" on page 21](#).
 - Assign the virtual search server to existing knowledgebases. See ["Managing Knowledgebases" on page 32](#).
3. Add knowledgebases if needed. You can add three types of knowledgebases: sclib, weplib, and fsyslib. For more information, see ["Add an sclib Knowledgebase" on page 32](#), ["Add a weplib Knowledgebase" on page 45](#), and ["Add an fsyslib Knowledgebase" on page 52](#).
4. Enable your language(s). Out-of-the-box, only English is enabled.

You will need to add your language(s) to the search engine's collection configuration file, known as the schema. This schema file is located in the Service Manager server directory. For details, see ["Enable Languages in the Solr Search Engine" on page 64](#).

You will also need to activate your language(s) in the **language** table. For details, see ["Activate a Knowledge Management language" on page 63](#). For information on which languages are supported by the search engine that can be enabled for Knowledge Management, see the KM Identifier values in the **language** table.

5. If needed, create your own thesaurus dictionaries for use when searching content. The KM Solr Search Engine supports thesaurus files (or dictionaries) for individual languages, but none are provided out-of-the-box. To create your own thesaurus dictionaries, see ["Create Search Engine Thesaurus Files" on page 67](#).
6. Modify the stop words for your language(s) if needed. See ["Modify Stop Words" on page 69](#).
7. When you first install the KM Solr search engine after having used the K2 search engine, you will need to re-index all of your knowledgebases, as the old indexes will not work. Once the new index has been created, you can re-index all of your knowledgebases. Read about the indexing process in ["Indexing the Knowledgebases" on page 72](#), and also see ["Perform a Full Re-Index on a Knowledgebase" on page 74](#).
8. Make sure that the KMUpdate process is started. See ["Indexing the Knowledgebases" on page 72](#).

Edit the Knowledge Management Environment Record

User role: System Administrator, KMAdmin

Before you can use Knowledge Management and the search engine, you must configure settings. The Knowledge Management Environment record allows you to set up connectivity and define options, such as enabling adaptive learning and setting the number of documents to be returned from a search. This record contains default settings. However, you can configure these settings to meet your business needs.

To configure the Knowledge Management application environment settings, do the following:

1. Click one of the following:
 - **Knowledge Management > Administration > Environment**
 - **System Administration > Ongoing Maintenance > Environment Records > Knowledge Management Environment**
2. Select new options or clear default options. Your changes redefine the Knowledge Management environment for all users.
3. **Assign the Default Knowledge View Group to all operators** who will have permission to search knowledge. A check mark ensures that all operators are able to view any documents in those document categories to which the default knowledge view group has access.

4. Select the check box to enable **Use Adaptive Learning to enhance search results** to weight search results based on usage. You can artificially weight documents using Adaptive Learning by adding a phrase and adding a number of occurrences with the phrase to specify that the term or phrase occurs in the document a specified number of times. If the phrase is not literally present, this can have the effect of putting the phrase, or word, in the document. It also simulates the number of times the phrase is added to the document index, based on the quantity you choose to apply.

Notes:

- Clear the check box to disable Adaptive Learning.
 - When you disable Adaptive Learning, you need to go to the "Manage Knowledgebases" screen and reindex all knowledgebases of type sclib.
 - Adaptive Learning data is not deleted when Adaptive Learning is disabled.
5. Specify the **Maximum number of documents** to be returned from a search.
 6. Specify the **Default expiration period** of time a document should be stored in the document queue before it expires.

Note: The time period specified here will be over-ridden if an expiration date was specified when the document was created using the Contribute New Document function.

7. Specify the style text for search results.
8. If you have deployed Service Request Catalog for Service Manager, enable Knowledge Management search for Service Request Catalog users.
 - a. Select the **SRC?** check box.
 - b. In the Search Server field, select a virtual search server from the list.
 - c. Click **Full Reindex**.

Note: Only the Service Manager knowledge library is available for Service Request Catalog users to search.

9. When you have finished making your changes, click **Save** and **OK**.
10. Log out of Service Manager and then log back in again for your changes to take effect.

Warning: When these fields are not set correctly and a user attempts to access a knowledgebase,

the Manage Knowledgebase form displays an error, stating that the search engine is incorrect or not found, and asking you to check the environment settings.

Managing Knowledgebases

There are three types of knowledgebases that you can add to Knowledge Management in Service Manager. You can add an sclib knowledgebase, an fsyslib knowledgebase, and a weblib knowledgebase. An sclib knowledgebase is created from a table in Service Manager. For example the out-of-the-box Incident_Library uses the probsummary table. A weblib knowledgebase is created by using web crawling to browse and index an external web site. The system creates an fsyslib knowledgebase when it crawls a file system.

The Knowledgebases feature enables administrators to add and delete knowledgebases. In order to add a knowledgebase and make it functional you need to:

- Identify the type of knowledgebase
- Map fields for indexing and searching
- If needed, update the four ScriptLibrary scripts that were created when you added the knowledgebase:
 - <LibraryNameHere>_kmaccess
 - <LibraryNameHere>_kmcategoridxscript
 - <LibraryNameHere>_kmprocesslibcriteria
 - <LibraryNameHere>_kmsearchsecurity
- For sclib type knowledgebases only, create a new tab for the knowledgebase for advanced searching

Add an sclib Knowledgebase

User role: System Administrator, KMAAdmin

The **sclib** type is used to index data contained in Service Manager tables, such as knowledge articles or other records. Out-of-the-box, there are five sclib knowledgebases:

- Incident_Library: used to index data of all Incident records.
- Interaction_Library: used to index data of all Interaction records.
- Knowledge_Library: used to index data of all knowledge articles.

- **KnownError_Library:** used to index data of all Known Error records.
- **Problem_Library:** used to index data of all Problem records.

You may need to add more sclib knowledgebases in your production environment. To do so, you need to perform the following tasks.

Task 1: Add an sclib knowledgebase record.

To add a sclib knowledgebase record:

1. Click **Knowledge Management > Configuration > Knowledgebases.**
2. Type a name for the new knowledgebase.
3. Type a display name for the new knowledgebase.
4. Select **sclib** in the Type list.
5. Click **Add.**
6. Type the Refresh Interval rate on the Status tab so that the selected knowledgebase index is updated at the specified interval. Each interval unit is five minutes (default). You may increase the interval to slow down the time between updates, or you may set the number at zero (0) to disable updates to the index.
7. On the **Type information** tab, provide the necessary information. The system creates a default Knowledgebase access script, a Search security script, and Category index scripts.

Field	Description
HP Service Manager Table Name:	This is the Service Manager table that will be indexed. A valid Service Manager table is required. For example: kmdocument .
HP Service Manager Table Query:	Used to enter a Service Manager style query to limit what records in the table are indexed. For example, a query to return only documents that are neither draft nor retired in the kmdocument table: <code>status ~= "draft" and status ~= "retired"</code> . A blank query indicates that all records will be indexed.
Document ID Field:	Every table in Service Manager has a unique ID field and this field identifies

Field	Description
	the field name of the ID field. The indexer uses this field to uniquely identify each document in the index. This is a required field for indexing a knowledgebase.
Index Attachments	<p>If the table being indexed has attachments, select this check box to have the attachments indexed.</p> <p>Caution: The search engine can index a wide variety of data types. However, indexing attachments is processor intensive. Each attachment must be extracted from the attachment table, written out to the local file system, and then streamed to the search engine server.</p>
Skip These Extensions:	<p>A semicolon-separated list of file extensions that should not be indexed nor extracted. Certain file types either cannot be indexed, or provide no relevance. By providing these extensions, you can increase index performance.</p> <p>Note: Sample gif;jpg without any spaces.</p>
Knowledgebase access script:	This script specifies the script the system uses to determine if a particular user has rights to access the knowledgebase. See the default script for detailed information.
Search security script:	This script further limits what users have access to, when they have access to the knowledgebase. This script returns a query string that is added to the user's normal query to limit the scope of the particular user's access to documents in the knowledgebase. See the default script for detailed information.
Category index script:	This script processes the document category so that the indexer can translate the document's category into a string that the search engine can use later to find the document based on the user's category access.
Advanced Search Script:	This script is used to build and return a string of library-specific query values that were entered by the user under the tabs in the Advanced Search screen. Tailor this script when a knowledgebase has a tab in the Advanced Search screen and you wish to modify the fields available for Advanced Search.
Default Locale:	Specifies the default language used by the search engine when searching and indexing. By default, the language code is English.

The following figure shows the **Type information** tab of an example sclib knowledgebase named ConfigurationItem_Library.

Knowledgebase Name: ConfigurationItem_Library
 Display Name: Configuration Items
 Type: sclib
 Search Server Name: ml6

Status
 Type information
 Field Definitions
 Errors

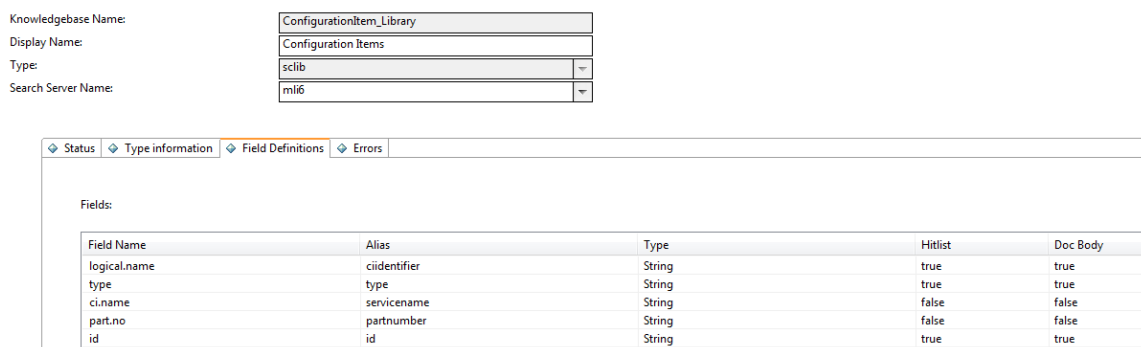
HP Service Manager Table Name: device
 HP Service Manager Table Query:
 Document ID Field: id
 Index Attachments
 Skip these extensions: jpg;bmp;gif;exe;unl;unsafe
 Knowledgebase access script: ConfigurationItem_Library_kmaccess
 Search security script: ConfigurationItem_Library_kmsearchsecurity
 Category index script: ConfigurationItem_Library_kmcategoryidscript
 Advanced Search Script: ConfigurationItem_Library_kmprocesslibcriteria
 Default Locale: English

8. Provide the necessary field definition data for the new knowledgebase.

Field	Description
Field Name	Specifies the field name in the Service Manager table included in the index.
Alias	Specifies the name that the field is to be indexed as. You can make use of the Alias field to have a single common field name for searching and for the hitlist. For example, you may wish to alias different fields from different tables as "Title" so they can be searched using Advanced Search. Fields can have more than one alias. Separate these fields with a semicolon. An alias can be the same name as the field name. If the alias name includes ".", the system converts the period to an underscore when indexed.
Type	Indicates whether the field is a plain text string, a rich text string or a date type. The indexer ignores HTML markup in rich text strings and indexes plain text strings completely.
Hitlist	Defines what fields are available on the search hitlist. Fields marked as "true" in the hitlist column are available to be included on a search results hitlist display.
Doc Body	This column is a Boolean (True/False) field. By setting the field in this column to "true," the system indexes the field's content as part of the document body and also as an individual field. The search engine searches only the document body in the simple search. Fields with the Doc Body marked as "false" can only be searched by doing a field

Field	Description
	search with the Advanced Search.
Delete Field	Used to delete a field from the table. <ol style="list-style-type: none"> a. Select the field you want to delete. <ul style="list-style-type: none"> • In the Windows client, click a field to select it. • In the web client, hold down the Shift key as you click a field to select it. b. Click the Delete Field button to delete the field.

The following figure shows the **Field Definitions** tab of the ConfigurationItem_Library knowledgebase. These fields are defined in the **device** table, and you need to specify an Alias for each of them.



9. Configure the **Status** tab.

The **Status** tab for an sclib knowledgebase provides information about the selected knowledgebase's index and details about the search servers to which the knowledgebase is connected. This information includes:

Field	Description
Master, Slave, or Load Balancer	This field provides information about the search servers that are connected to the selected knowledgebase. Details on each server connection include type of server (Master, Slave, or Load Balancer), state of the server, date the server was created, and the number of documents that were indexed in the knowledgebase. For more information, see State, Created, and Docs .

Field	Description
	To make changes to a search server, go to Knowledge Management > Configuration > Configure Search Servers and select the search server.
State	Indicates whether the knowledgebase is on-line, off-line, or replicating (if it is a slave server). If State is blank, the knowledgebase is either off-line or the search engine is not connected.
Created	Displays the date and time the knowledgebase was created. This value will change when the Full Reindex button is pressed.
Docs	Displays the number of documents currently indexed in the knowledgebase . This number may or may not exactly match the number of records in the table being indexed. Factors that can contribute to this include: Using a selection query other than "true" on the Type Information tab or when a document is indexed into multiple document categories. It is indexed once for each document category to which it is assigned.
Refresh Interval	<p>Displays the current interval used to update the selected knowledgebase index. Each interval unit is 5 minutes (default). You may increase the interval, which slows down time between updates, by increasing this number. Setting this number to 0 (zero) disables updates to the index. To re-start indexing, reset the interval to a value greater than zero.</p> <p>Note: Changes made to the table being indexed will be cached even if the update interval is set to 0. These changes will be processed once the interval is set higher than 0. A full re-index removes all changes for this knowledgebase from the change cache.</p>

The tab also includes buttons for indexing.

Button	Description
Full Reindex:	<p>When selected, the search engine performs a full re-index of a knowledgebase. If the index does not exist, it will be created. If it does exist, it will be deleted and re-created. A full re-index will remove all changes for this knowledgebase from the change cache since they will no longer be relevant.</p> <p>When you click the Full Reindex button, the KMUpdate process begins initialization processing to start indexing a knowledgebase, but because indexing runs as a background process, the search engine does not begin indexing until the specified refresh interval is reached. Doing a full re-index on a large knowledgebase may have</p>

Button	Description
	<p>a significant impact on system resources.</p> <p>Certain actions require a full re-index. These actions are:</p> <ul style="list-style-type: none"> ○ Initial setup (no indexes exist). ○ Changing any value on the Type Information tab or the Field Definitions tab. ○ When a large number of changes or new documents have been added. For example, when you import new documents. ○ When search engine performance becomes sluggish. <p>As updates are applied to the index, they are added as incremental index files. If you have a knowledgebase that has had many changes applied, these incremental index files can slow the search engine down, since it must perform your query on each one. A full re-index builds a new clean index that performs better. This is similar to defragmenting a hard drive.</p> <p>Note: If you have a single master server defined, users will not be able to search the knowledgebase being re-indexed until the process completes successfully. If you have one or more slave servers, users will continue to search the old knowledgebase while the index is being re-created on the master server. Once indexing is complete, the old knowledgebase will be replaced by the new knowledgebase on the slave servers automatically and seamlessly, without disrupting user searchers.</p>
Refresh Statistics	When selected, the search engine refreshes the statistics of the indexing process to show how many documents are indexed and searchable at that time in the process.

10. Click **Save**.
11. Click **Full Reindex** on the **Status** tab to re-index the documents in the knowledgebase.
12. Click **Refresh Statistics** on the **Status** tab to refresh statistics tracking knowledge documents at various stages.

The following figure shows that a Full Re-indexing has been completed for the example ConfigurationItem_Library knowledgebase.

Knowledgebase Name:	ConfigurationItem_Library
Display Name:	Configuration Items
Type:	sclib
Search Server Name:	mli6

◆ Status ◆ Type information ◆ Field Definitions ◆ Errors

Master: mli6:8088
State: online
Created: Fri Mar 09 16:20:17 CST 2012
Docs: 861

Refresh Interval: 1
0 = No Updates, All others multiply by 5 (eg. 1 interval = 5 minutes)

Full Reindex Refresh Statistics

Now, you can perform simple searches in the new knowledgebase. However, if you want to enable the Advanced Search functionality for the knowledgebase, you need to complete the following additional tasks.

Task 2: Add new fields in the kmquery table.

You must add new fields to the kmquery dbdict record, which specifies the fields available for searching during an advanced search of the new knowledgebase.

When you add a field, use the naming convention <element_name>lib.<field_name> to make it available for advanced searching. Taking the ConfigurationItem_Library knowledgebase for example, you can use **ci** or **configurationitem** for <element_name>, followed by the field's Alias name defined on the **Field Definitions** tab. For example, for the **logical.name** field, you can add an entry of **clib.ciidentifier**.

Note: These field names are used in the format input and are mapped into the query in the <LibraryNameHere>kmprocesslibcriteria script library.

When you add a new sclib type library, the system adds a new Boolean field. For example, if you named your new knowledgebase ConfigurationItem_Library, the system adds a Boolean field named ConfigurationItemLibrary.

incidentlib.priority	character	1	132	INCIDENTLIB_PRIORITY	VARCHAR(40)	m1
interactionlib.logicalname	character	1	133	INTERACTIONLIB_LOGICALNAME	VARCHAR(200)	m1
interactionlib.impact	character	1	134	INTERACTIONLIB_IMPACT	VARCHAR(40)	m1
interactionlib.severity	character	1	135	INTERACTIONLIB_SEVERITY	VARCHAR(40)	m1
interactionlib.priority	character	1	136	INTERACTIONLIB_PRIORITY	VARCHAR(40)	m1
yahoo	logical	1	137			
ConfigurationItemLibrary	logical	1	138			

As an example, the following steps illustrate how to add new fields in the kmquery dbdict record for the ConfigurationItem_Library knowledgebase.

1. Log in to the Service Manager Windows client as a system administrator.
2. Type **dbdict** in the command line, and press Enter.
3. In the **File Name** field, type **kmquery**, and click **Search**. The kmquery dbdict record opens.
4. On the **Fields** tab, place the cursor on the **descriptor** row, and click **New Field/Key** to add the following example fields to the kmquery dbdict record.

ConfigurationItemLibrary	logical	1	138	CONFIGURATIONITEMLIBRARY	CHAR(1)	m1
cilib.ciidentifier	character	1	139	CILIB_IDENTIFIER	VARCHAR(200)	m1
cilib.type	character	1	140	CILIB_TYPE	VARCHAR(60)	m1
cilib.status	character	1	141	CILIB_STATUS	VARCHAR(60)	m1
cilib.partnumber	character	1	142	CILIB_PARTNUMBER	VARCHAR(60)	m1
cilib.id	character	1	143	CILIB_ID	VARCHAR(60)	m1

5. Click **OK**.
6. Select **SM Alters**. A message displays: **Record updated in the dbdict file**.

Task 3: Add the new fields to the Advanced Search Script of the knowledgebase.

Once you have added the Advanced Search fields in the kmquery dbdict, add the fields in the advanced search script that was created when the knowledgebase record was added. You can find this script on the **Type information** tab of the sclib knowledgebase record. For example, for the ConfigurationItem_Library knowledgebase, the Advanced Search Script is named **ConfigurationItem_Library_kmprocesslibcriteria**.

As an example, the following steps illustrate how to add the new fields in task 2 to the **ConfigurationItem_Library_kmprocesslibcriteria** script.

1. Log in to Service Manager as a system administrator.
2. Type **sl** in the command line, and press Enter.
3. In the **Name** field, type **ConfigurationItem_Library_kmprocesslibcriteria**, and then click **Search**.
4. Add the following lines to the script.


```
function ProcessLibCriteria(KMQuery)
{
var getenginename = system.library.KMSearchInterface.getEngineName();
var strQuery = "";
if (getenginename == "K2")
{
if(KMQuery.cilib_ciidentifier != null)
strQuery += " <AND> (ciidentifier <CONTAINS> " + KMQuery.cilib_ciidentifier + "
<AND> _style <CONTAINS> ConfigurationItem_Library)";
if(KMQuery.cilib_type != null)
strQuery += " <AND> (type <CONTAINS> " + KMQuery.cilib_type + " <AND> _style
<CONTAINS> ConfigurationItem_Library)";
y)";
if(KMQuery.cilib_status != null)
strQuery += " <AND> (status <CONTAINS> " + KMQuery.cilib_status + " <AND> _
style <CONTAINS> ConfigurationItem_Library)";
if(KMQuery.cilib_partnumber != null)
strQuery += " <AND> (partnumber <CONTAINS> " + KMQuery.cilib_partnumber + "
<AND> _style <CONTAINS> ConfigurationItem_Library)";
if(KMQuery.cilib_id != null)
strQuery += " <AND> (id <CONTAINS> " + KMQuery.cilib_id + " <AND> _style
<CONTAINS> ConfigurationItem_Library)";

}
else
{
if(KMQuery.cilib_ciidentifier != null)
strQuery += " (ciidentifier : " + KMQuery.cilib_ciidentifier + " AND
knowledgebase_name: ConfigurationItem_Library)";
if(KMQuery.cilib_type != null)
strQuery += " (type: \" + KMQuery.cilib_type + "\" AND knowledgebase_name:
ConfigurationItem_Library)";
if(KMQuery.cilib_status != null)
strQuery += " (status: " + KMQuery.cilib_status + " AND knowledgebase_name:
ConfigurationItem_Library)";
```

```
if(KMQuery.cilib_partnumber != null)

strQuery += " (partnumber: " + KMQuery.cilib_partnumber + " AND knowledgebase_
name: ConfigurationItem_Library)";

if(KMQuery.cilib_id != null)

strQuery += " (id: " + KMQuery.cilib_id + " AND knowledgebase_name:
ConfigurationItem_Library)";

}

return strQuery;

}
```

5. Save and compile the script. A message should display: **Successful compilation of JavaScript function or expression.**

Task 4: Create a viewing format for the knowledge candidates for the knowledgebase.

Use Forms Designer to create a viewing form for the knowledge candidates for the new knowledgebase. When users click a search result (knowledge candidate) of the new knowledgebase, it opens in this form.

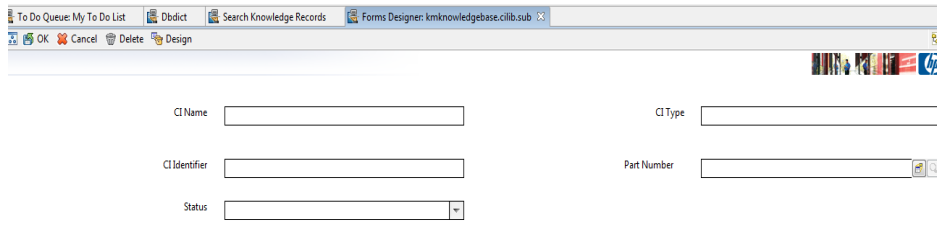
Task 5: Add a tab to the Knowledgebase Advanced Search form.

Use the Forms Designer to add a new widget to include the fields that have been added in the kmquery dbdict. Set the visible condition for the widget to use the same Boolean field name the system added when you added the new knowledgebase.

As an example, the following steps illustrate how to add a widget for the ConfigurationItem_Library knowledgebase.

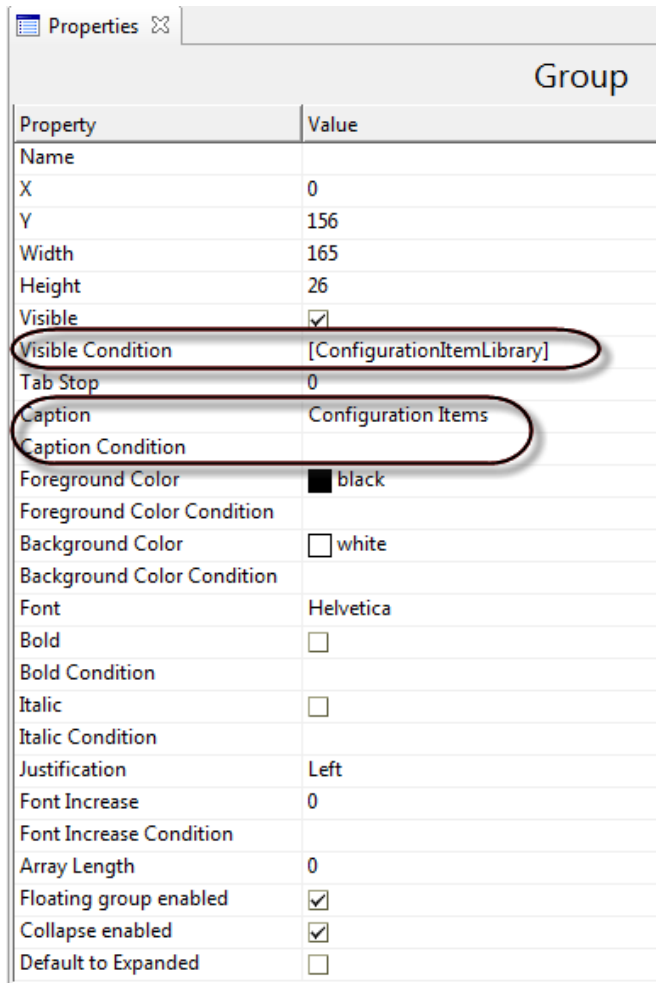
1. Log in to Service Manager as a system administrator.
2. Type **fd** in the command line, and press ENTER.
3. Create a subform for the ConfigurationItem_Library knowledgebase.
 - a. In the Form field, type **kmknowledgebase.cilib.sub**, and then click **New**.
 - b. Click **No** to create the subform without using the form wizard.

- c. Create the subform with all fields you have added to the kmquery dbdict record.



- 4. Add a widget for the ConfigurationItem_Library knowledgebase.
 - a. In Forms Designer, open the **kmknowledgebase.advsearch.g** form in Design mode. Out-of-the-box, this form contains the following Group controls: **Knowledge Library**, **Known Errors**, **Problems**, **Incidents**, and **Interactions**.
 - b. Create a new Group widget by copying an existing Group control, for example, the **Interactions** group.
 - c. Change the following properties of the new group:
 - Visible Condition: **[ConfigurationItemLibrary]** (This is the Boolean field name the system added to the kmquery dbdict record when you added the new knowledgebase)
 - Caption: **Configuration Items**

- **Caption Condition:** Leave this setting empty.



Properties		Group
Property	Value	
Name		
X	0	
Y	156	
Width	165	
Height	26	
Visible	<input checked="" type="checkbox"/>	
Visible Condition	[ConfigurationItemLibrary]	
Tab Stop	0	
Caption	Configuration Items	
Caption Condition		
Foreground Color	<input checked="" type="checkbox"/> black	
Foreground Color Condition		
Background Color	<input type="checkbox"/> white	
Background Color Condition		
Font	Helvetica	
Bold	<input type="checkbox"/>	
Bold Condition		
Italic	<input type="checkbox"/>	
Italic Condition		
Justification	Left	
Font Increase	0	
Font Increase Condition		
Array Length	0	
Floating group enabled	<input checked="" type="checkbox"/>	
Collapse enabled	<input checked="" type="checkbox"/>	
Default to Expanded	<input type="checkbox"/>	

- d. Select the subform enclosed in the group, and change the Format property to **kmknowledgebase.cilib.sub**.

Subformat	
Property	Value
Name	
X	0
Y	2
Width	165
Height	22
Visible	<input checked="" type="checkbox"/>
Visible Condition	
Tab Stop	0
Format	kmknowledgebase.cilib.sub
Virtual Join	<input type="checkbox"/>
Display Blank	<input checked="" type="checkbox"/>
Display Using Table	<input type="checkbox"/>
Input	

- e. Click **OK**.
5. Verify that KM Advanced Search is enabled for the new knowledgebase.
 - a. Navigate to **Knowledge Management > Search Knowledgebase**.
 - b. In the **Search In** section, select and then deselect the Configuration Items library, to verify that the Configuration Items widget displays and disappears accordingly.

Advanced Search

Search for:

Query Language: English

Filter by...

All of these words	<input type="text"/>
This exact phrase	<input type="text"/>
Any of these words	<input type="text"/>
None of these words	<input type="text"/>

Search in

- Knowledge Library
- Known Errors
- Problems
- Incidents
- Interactions
- Configuration Items

Configuration Items

CI Name	<input type="text"/>	CI Type	<input type="text"/>
CI Identifier	<input type="text"/>	Part Number	<input type="text"/> <input type="button" value="v"/>
Status	<input type="text"/>		

- c. Enter values in certain Configuration Item fields, and then click **Search**, to verify that Advanced Search works correctly for configuration item records.

Add a weblib Knowledgebase

User role: System Administrator

Web crawling enables administrators to make outside knowledge sources accessible to users and reduce administration costs. With this feature, you can:

- Index and search external Web content, such as RightAnswers and Microsoft's TechNet.
- Index and search Intranet content, such as internal corporate web content.

If you have large number of documents that you do not want to import into Service Manager but still wish to index and search, you can publish these documents to a web server for crawling with the web crawler. You can also crawl these documents with the file system crawler.

Indexing requirements for web crawling

The indexing requirements for a weplib-type knowledgebase are different than those of an sclib type. For weplib types, Service Manager stores only information used to create the knowledgebase (collection) format and the information used by the search engine to index the pages being crawled.

Log files for web crawling

Log files are generated during indexing for the web pages being crawled. These log files are located in the following directory:

```
C:\Program Files\HP\Service Manager 9.30\Search_Engine\kmcrawler\logs
```

The log files end in .log.

The two most important log files in this folder are skipkeys.log and joberror.log. The skipkeys.log provides detailed information as to why a particular page was skipped during indexing. For example, if a page is an excluded mime type or if the path depth has been exceeded, it is logged in this file. The joberror.log contains the errors encountered by the indexer. For example, if you provided a start URL that could not be reached, it is logged here.

To add a new weplib knowledgebase, you need to complete the following tasks.

Task 1: Add a weplib type knowledgebase record.

To add a weplib type knowledgebase record:

1. Click **Knowledge Management > Configuration > Knowledgebases**.
2. Type a unique name for the new knowledgebase in the Knowledgebase name field (required). For example, **My_Web**.
3. Type a display name for the knowledgebase (required). For example, **My Web**.
4. Select **weplib** in the Type field list.

5. Click **Add**.

The knowledgebase record is added. Continue to configure the tabs of the record and test the knowledgebase, as described in the following tasks.

Task 2: Configure the Type information tab.

The Type Information tab maintains information about the knowledgebase's source and security. Configure the following **Type information** settings.

Field	Description
URLs	<p>URL: Type the URLs in the space provided. You can enter more than one URL, but use only one URL per line.</p> <p>Note: The "http://" prefix is required for all URL's. For example: <code>http://www.bbc.com</code></p>
URL path depth	<p>Limits indexing to the specified number of path segments in the URL or file system path. The default is 100 path segments.</p> <p>The path length is determined as follows:</p> <ul style="list-style-type: none"> • The host name and drive letter are not included. For example, neither <code>www.spider.com:80/</code> or <code>C:\</code> are included in determining the path length. • All elements following the host name are included. • The actual file name, if present, is included. For example, <code>/world.html</code> is included when determining the path length. • Any directory paths between the host and the actual file name are included. <p>For example: For the following URL, the path length would be 4: <code>http://www.spider:80/comics/fun/funny/world.html</code></p> <p>where: comics = 1 segment fun = 1 segment funny = 1 segment world.html = 1 segment</p>
Max links to follow	<p>Specifies the maximum number of levels from the starting URL that an indexing includes. If you see extremely large numbers of documents in a knowledgebase where you do not expect them, consider experimenting with this option, in conjunction with the Content options, to reduce the reach of the index.</p> <p>Specify a number between 0 and 255. The default is 255, which is equivalent to there are no limits on the number of jumps.</p>

Field	Description
Constrain indexing to host domains	By default, links are not followed outside the hosts provided in the URLs. Un-checking this box lets the indexer index outside the hosts specified.
Mime Types	<ul style="list-style-type: none"> • Unlimited: The search engine supports over 1,200 different MIME types. Selecting "Unlimited" will include this full list of MIME types during crawling. • Include:List the MIME file types to be included in the index. Specify the file type by using the MIME specification. Enter only one MIME type per line. You may use the "*" wildcard only for MIME types. For example, "text/*". This example will include all the MIME types that start with "text/". • Exclude: List the file types NOT to be included in the index. Specify the file type by using the MIME specification. Enter only one MIME type per line. You may use the "*" wildcard only for MIME types. For example, "text/*". This example excludes all the MIME types that start with "text/". <p>The default values in the drop-down list for MIME types can be configured as follows:</p> <ol style="list-style-type: none"> 1. Go To Tailoring > Database Manager. 2. Search for the kmmimetypes table to retrieve the full list of MIME types. Note: You can edit the list of MIME types. See "Edit the List of MIME Types" on page 52. 3. Check the Mimelist box, so that the all available MIME types will appear in the drop-down list. 4. Save your changes.
Proxy Configuration	<ul style="list-style-type: none"> • Proxy Host: Provides the proxy host name when the web site being crawled uses a proxy server. • Port: The port number used by the proxy server. • Username: The user name required by the proxy server for access. This may not be required for all proxy servers. • Password: The password associated with the user name provided. This may not be required by all proxy servers.
Security Scripts	<ul style="list-style-type: none"> • Knowledgebase access script: This script specifies the script the system uses to determine if a particular user has rights to access the knowledgebase. See the default script for more detailed information. • Search security script: This script further limits what the user has access to when accessing the knowledgebase. This script returns a query string that is

Field	Description
	<p>added to the user's normal query to limit the scope of the particular user's access to documents in the knowledgebase. See the default script for more detailed information.</p> <ul style="list-style-type: none"> • Category index script: This script processes the document category so that the indexer can translate the document's category into a string that the search engine can use later to find the document based on the user's category access. • Advanced search script: This script processes a string of library-specific query values using the KMQuery object. The values in KMQuery were entered by the user under the tabs in the Advanced Search screen. You can tailor this script if this knowledgebase has a tab in the Advanced Search screen. • Default locale: Specifies the default language used by the search engine when searching and indexing. By default, the language code matches the language you logged in as.

Task 2: Configure the Status tab.

The Status tab for a weplib knowledgebase displays information about the selected knowledgebase's index. This information includes the following:

Field	Description
Master, Slave, or Load Balancer	<p>This field provides information about the search servers that are connected to the selected knowledgebase. Details on each server connection include type of server (Master, Slave, or Load Balancer), state of the server, date the server was created, and the number of documents that were indexed in the knowledgebase. For more information, see the State, Created, and Docs settings.</p> <p>To make changes to a search server, go to Knowledge Management > Manage Search Servers and select the search server.</p>
State	Indicates whether the knowledgebase is on-line, off-line, or replicating (if it is a slave server). If State is blank, the knowledgebase is either off-line or the search engine is not connected. If the knowledgebase has not been created, this field is blank.
Created	Displays the knowledgebase creation date and time. If the knowledgebase has not been created, this field is blank.
Docs	Displays the number of documents contained in the index. If the knowledgebase has not been created, this field is blank. If the knowledgebase has been created but not indexed, this field will show 0 documents.
Status	Since the index is created separately from the knowledgebase creation, the Status field

Field	Description
	displays the current status of the index. Values can be "Not Created," "Created," "Running," and "Finished."
Schedule this index?	You can choose to schedule the index to start on a particular day and time. Select the Schedule this index? check box to enable scheduling.
Start Date	Select the first day you wish this index to run. Adjust the time values to the time of day the index should run.
Frequency	You can schedule the index to run once, hourly, daily, weekly, or monthly. For example, if you scheduled the index to run on Monday, May 5, at 6 p.m. and selected an hourly frequency, the index would run the first time on Monday, May 5, at 6 p.m. and would then run again at 7 p.m., 8 p.m., and so on until you updated the schedule. Note: When indexing, the search engine only indexes documents that have changed since the last run.
Crawler Host	Specify the host name of a search server for web crawling. It can be a master or slave search server; however for optimized performance you can use a dedicated server for crawling only.
Crawler Port	Specify the Tomcat port of the crawler host.

The tab also includes buttons for indexing.

Button	Description
Initialize Index	Once you have all your settings specified for a new knowledgebase, click this button. The system sends the settings to the search engine. The search engine creates the empty knowledgebase for index. After a knowledgebase and index have been created, click this button to erase the index and create an empty knowledgebase. You should create a new knowledgebase if you change the parameters on either the Type information tab or Field Definitions tab.
Start Indexing	Click this button to start an index manually. Instead of scheduling index updates, you can use this button if the file system you are crawling does not change or if you want to test index settings once.
Stop Indexing	Click this button to stop an indexing process that is running. Click the Start Indexing button to resume indexing from the point where the index was stopped. To start indexing from the beginning, click Initialize Index .
Refresh Statistics	When selected, the search engine refreshes the statistics of the indexing process to show how many documents are indexed and searchable at that time in the process.

Task 3: Index the weblib knowledgebase.

1. On the **Status** tab, click **Initialize Index**.
2. Click **Start Indexing**.
3. Click **Refresh Statistics** to monitor the indexing status. Once the documents have been indexed on the crawler and replicated to the searcher, you can continue to perform a test search on the weblib knowledgebase.

The following figure shows an example.

Display Name:	<input type="text" value="My Web"/>
Type:	<input type="text" value="weblib"/>
Search Server Name:	<input type="text" value="mli6"/>

◆ Status ◆ Type information ◆ Field Definitions

Master: mli6:8088
State: online
Created: Mon Mar 19 15:49:43 CST 2012
Docs: 306

Schedule this index?

Start Date: Crawler Host:
Frequency: Crawler Port:

Task 4: Search in the weblib knowledgebase.

Perform a search in the new weblib knowledgebase to verify that it is set up properly.

1. Click **Knowledge Management > Search Knowledgebase**.
2. In the Search In section, select only the new weblib knowledgebase.
3. Enter a search word in the search box, and click **Search**.

The following figure illustrates a list of search results of an example weblib knowledgebase.



Edit the List of MIME Types

User role: System Administrator, KMAAdmin

When indexing file systems and web sites, there are MIME type file extensions recognized by the Solr search engine. The web server passes along MIME-type information based on its own internal tables.

To edit the list of MIME types to be included or excluded in the drop-down lists:

1. Click **Tailoring > Database Manager**.
2. In the Table field, type **kmmimetypes** and then click **Search**.
3. Based on your business needs, edit the "include" and "exclude" MIME type selections.

Note: MIME types with the Mimest checkbox selected (or checked) will display in the drop-down selections.

- a. Select the MIME type **Mimest** check boxes that are blank, so that those MIME types will be included in the drop-down selections.
 - b. Uncheck the MIME type **Mimest** check boxes that are currently selected, so that those MIME types will be excluded from the drop-down selections.
 - c. Click the **Unlimited** radio button to store all MIME types in the kmmimetypes table, so that all MIME types are used.
4. Save your changes.

Add an fsyslib Knowledgebase

User role: System Administrator, KMAAdmin

File systems can be crawled to create fsyslib-type knowledgebases. File system crawling enables administrators to make the knowledge in a file system outside of Service Manager accessible to users and to reduce administration costs.

Note: The KM Crawler can crawl only local file systems on the crawler host.

To add an fsyslib-type knowledgebase:

1. Click **Knowledge Management > Configuration > Knowledgebases**.
2. Type a unique name for the new knowledgebase in the Knowledgebase name field (required). For example, **My_FYS**.
3. Type a display name for the knowledgebase (required). For example, **My File System**.
4. Select **fsyslib** in the Type field list.
5. Click **Add**. The knowledgebase record is added.
6. Configure the **Type information** tab.

Field	Description
Directories	<ul style="list-style-type: none"> o Start Path: Type the Universal Naming Convention (UNC) directory path. You can enter more than one path, but use only one path per line. If the starting directory is hosted on a UNIX server, also specify: o Replace this UNIX path: The UNIX directory to substitute with the Windows mapping to the UNIX server path. o With this Windows mapping: The Windows mapping to the UNIX path. This translation of the mapped path is necessary so that the Windows client loads the knowledge article from the correct location when displaying it in the hitlist after a search. <p>For example: Start Path: /samba/test/webcrawltest Replace this UNIX path: /samba/test With this Windows mapping: \\blade100\test\<UNIX mapping on Windows></p>
Path depth:	Limits indexing to the specified number of path segments in the file system path. The default is 100 path segments.

Field	Description
	<p>The path length is determined as follows:</p> <ul style="list-style-type: none"> ○ The host name (for example, \\hostname) is not included in determining the path length. ○ All elements following the host name are included and determine the path length in the path name, including the actual file name (for example, \world.htm) if it is present. ○ Any directory paths between the host and the actual file name are included. <p>Example: For the following UNC path, the path length would be 4: \\host\comics\fun\funny\world.html</p> <p>where: comics = 1 segment fun = 1 segment funny = 1 segment world.html = 1 segment</p>
<p>Mime Types</p>	<ul style="list-style-type: none"> ○ Unlimited: The search engine supports over 1,200 different MIME types. Selecting "Unlimited" will include this full list of MIME types during crawling. ○ Include: List the MiIME file types to be included in the index. Specify the file type by using the MIME specification. Enter only one MIME type per line. You may use the "*" wildcard only for MIME types. For example, "text/*". This example will include all the MIME types that start with "text/". ○ Exclude: List the file types NOT to be included in the index. Specify the file type by using the MIME specification. Enter only one MIME type per line. You may use the "*" wildcard only for MIME types. For example, "text/*". This example excludes all the MIME types that start with "text/". <p>The default values in the drop-down list for MIME types can be configured as follows:</p> <ol style="list-style-type: none"> a. Go To Tailoring > Database Manager. b. Search for the kmmimetypes table to retrieve the full list of MIME types. c. Check the Mimelist box, so that the all available MIME types will appear in the

Field	Description
	<p>drop-down list.</p> <p>d. Save your changes.</p>
Security Scripts	<ul style="list-style-type: none"> o Knowledgebase access script: This script specifies the script the system uses to determine if a particular user has rights to access the knowledgebase. See the default script for more detailed information. o Search security script: This script further limits what the user has access when given access to the knowledgebase. This script returns a query string that is added to the user's normal query to limit the scope of the particular user's access to documents in the knowledgebase. See the default script for more detailed information. o Category index script: This script processes the document category so that the indexer can translate the document's category into a string that the search engine can use later to find the document based on the user's category access. o Advanced search script: This script processes a string of library-specific query values using the KMQuery object. The values in KMQuery were entered by the user under the tabs in the Advanced Search screen. You can tailor this script if this knowledgebase has a tab in the Advanced Search screen. o Default locale: Specifies the default language used by the search engine when searching and indexing. By default, the language code matches the language you logged in as.

7. Configure the **Field Definitions** tab.

Column	Description
Field Type	<p>Select Constant. You may alias the 'constant' field as you would any other field. Every document indexed will have a field with the name you specified and the value listed in this field. Constant fields can be used for search security, categorization, or similar types of data that you do not have edit access to.</p> <p>Example constant: \$PASSAGE_BASED_SUMMARY</p>
Field Name	<p>Define a unique name for your 'constant' field in the Field Name.</p>
Alias	<p>This is the name of the field as it is to be indexed. You can make use of the Alias field</p>

Column	Description
	to have a single common field name for searching and for the hitlist.
Data Type	Specify the Data Type for date fields to allow date range searching.
Hitlist	Defines what fields are available on the search hitlist. Fields marked as 'true' in the Hitlist field are available to be included on a search results hitlist display.
Value	Specify the value for the 'constant' field. Note: The Value field is not used for a field type meta tag.

8. View the **Status** tab.

Field	Description
Master, Slave, or Load Balancer	This field provides information about the search servers that are connected to the selected knowledgebase. Details on each server connection include type of server (Master, Slave, or Load Balancer), state of the server, date the server was created, and the number of documents that were indexed in the knowledgebase. For more information, see State , Created , and Docs . To make changes to a search server, go to Knowledge Management > Configuration > Configure Search Servers and select the search server.
State	Displays whether the collection is online or offline. If the knowledgebase has not been created, this field is blank.
Created	Displays the knowledgebase creation date and time. If the knowledgebase has not been created, this field is blank.
Docs	Displays the number of documents contained in the index. If the knowledgebase has not been created, this field is blank. If the knowledgebase has been created but not indexed, this field will show 0 documents.
Status	Since the index is created separately from the knowledgebase creation, the Status field displays the current status of the index. Values can be "Not Created", "Created", "Running", and "Finished".
Schedule this index?	You can choose to schedule the index to start on a particular day and time. Select the Schedule this index? check box to enable scheduling.

Field	Description
Start Date	Select the first day you wish this index to run. Adjust the time values to the time of day the index should run.
Frequency	<p>You can schedule the index to run once, hourly, daily, weekly, or monthly. For example, If you scheduled the index to run on Monday, May 5, at 6 p.m. and selected hourly for the frequency, the index would run the first time on Monday, May 5, at 6 p.m. and would then run again at 7 p.m., 8 p.m., and so on until you updated the schedule.</p> <p>When indexing, the search engine only indexes documents that have changed since the last run.</p>
Crawler Host	Specify the host name of a search server for web crawling.
Crawler Port	Specify the Tomcat port of the crawler host.

This tab also contains the following indexing buttons:

Button	Description
Initialize Index	Once you have all your settings specified for a new knowledgebase, click this button. The system sends the settings to the search engine. The search engine creates the empty knowledgebase for index. After a knowledgebase and index have been created, click this button to erase the index and create an empty knowledgebase. You should create a new knowledgebase if you change the parameters on either the Type information tab or Field Definitions tab.
Start Indexing	Click this button to start an index manually. Instead of scheduling index updates, you can use this button if the file system you are crawling does not change or if you want to test index settings once.
Stop Indexing	Click this button to stop an indexing process that is running. Click the Start Indexing button to resume indexing from the point where the index was stopped. To start indexing from the beginning, see Initialize Index .
Refresh Statistics	When selected, the search engine refreshes the statistics of the indexing process to show how many documents are indexed and searchable at that time in the process.

- Click **Initialize Index** to index the new fsyslib knowledgebase.

Display Name: My File System
Type: fsyslib
Search Server Name: mli6

Status | Type information | Field Definitions

Master: mli6:8088
State: online
Created: Tue Mar 20 11:20:11 CST 2012
Docs: 0

Schedule this index?

Start Date: [] Crawler Host: mli6
Frequency: [] Crawler Port: 8088

Initialize Index Start Indexing Stop Indexing Refresh Statistics

10. Click **Save**.

11. Perform a test search on the new fsyslib knowledgebase to verify that it is set up correctly.

Search Knowledgebase

Integration [] Search [] Advanced... []

Search In

- Problems
- Incidents
- Interactions
- Configuration Items
- My File System
- My Web

0 documents found. 0 documents searched.

Delete a Knowledgebase

You can delete an existing knowledgebase, using a Delete button available on the Knowledgebase Maintenance form. When you delete a knowledgebase, the system automatically performs certain clean-ups needed for the removed knowledgebase, however for a sclib type knowledgebase, you still need to manually undo the changes to the system you made when adding the knowledgebase.

When you delete a knowledgebase, the system automatically:

- Removes four triggers on the table whose records were indexed: after.add.KM.<tablename>, before.update.KM.<tablename>, after,update.KM.<tablename>, and before.delete.KM.<tablename>.
- Removes the Boolean field from the kmquery table that corresponds to the knowledgebase.

- Removes references to the new knowledgebase from three functions in the KMSearch ScriptLibrary: `getAvailableKnowledgeBases`, `getSelectedCollections`, and `getSelectedCollectionsString`.
- Removes the `kmquery.default` display options that reference the removed knowledgebase.

To delete a knowledgebase:

1. Click **Knowledge Management > Configuration > Knowledgebases**.
2. Click **Search**.
3. Select the knowledgebase you want to delete.
4. In the knowledgebase detail form, click **Delete** and then click Yes.
5. If it is a weplib or fsyslib type knowledgebase, you do not need to do anything else.
6. If it is an sclib type knowledgebase, continue to clean up any references to the knowledgebase by undoing the changes to the system that you made when you added the knowledgebase.
 - a. Remove the tab for the knowledgebase on the `kmknowledgebase.advsearch.g` form, if you created one.
 - b. Remove any knowledgebase-specific fields from the `kmquery dbdict` that you previously added.
 - c. Remove any knowledgebase-specific references that you added to the KMSearch ScriptLibrary.
 - d. Delete the read-only viewing format that you created for the knowledge candidates for the knowledgebase.
 - e. Remove the format name that you added to the `kmquery.linkrequest` process record.
 - f. Remove the links to this knowledgebase from the `kmquery link` record for an advanced search record.

Configure Advanced Search for Knowledge Management

User role: System Administrator

Prior to version 9.32, the KM Solr Search Engine does not support Full Match mode for advanced search against KM fields. For example, if you enter "Phone Troubleshooting" in the Title field, your search will return records whose title contains "phone" and also records whose doc body contains "troubleshooting".

Starting with version 9.32, the Solr Search Engine allows you to configure Advanced Search for KM, to enable or disable Full Match mode. You do so by configuring the Advanced Search Script (<library name>_kmprocesslibcriteria) for each sclib type knowledgebase.

To configure Advanced Search for a sclib knowledgebase:

1. Make sure you have already added each query field to dbdict kmquery.
2. Navigate to **Knowledge Management > Configuration > Knowledgebases** and open the knowledgebase.
3. On the **Type information** tab, click the Find button next to the Advanced Search Script field. The <library name>_kmprocesslibcriteria record opens.
4. For each "field=value" type query field, add an entry to the field mapping section, using this format:

```
["<query_field_name>", "<search_engine_alias>", <true or false>]
```

Where:

- <query_field_name> is the name of the field you added to dbdict "kmquery";
- <search_engine_alias> is the alias specified for the query field in the **Field Definitions** tab of the knowledgebase;
- <true or false>: This part indicates if Advanced Search uses full match mode for this field (when set to true, full match is used).

The following are examples:

```
this.fieldmapping = [  
  ["incidentlib_number", "number", false],  
  ["incidentlib_status", "problemstatus", true],  
  ...
```

5. If you use custom search criteria that are not simply a "field=value" query, add it to the `processSpecial` function defined in the `<library name>_kmprocesslibcriteria` record.
6. Save the advanced search script.

Enabling Languages for KM Search

The Solr search engine supports a set of languages, however only some of them are officially supported in Service Manager. For more information, see ["Supported Languages for the Solr Search Engine" below](#).

To enable a language for KM search, you must activate the language for KM from the language table, and enable it in the Solr search engine schema file.

- ["Activate a Knowledge Management language" on page 63](#)
- ["Enable Languages in the Solr Search Engine" on page 64](#)

Once you have enabled your languages, you can continue to do the following:

- ["Create Search Engine Thesaurus Files" on page 67](#)
- ["Modify Stop Words" on page 69](#)
- ["Add a New KM Message to the scmessage Table" on page 69](#)
- ["Create a Hitlist with Multilingual Labels" on page 71](#)

Supported Languages for the Solr Search Engine

Technically, the Solr search engine supports all languages listed in the `schemastub.xml` file, however only some of them are officially supported in Service Manager.

Important: For Knowledge Management, HP only provides official support for languages that are officially supported in Service Manager.

The following table lists all languages technically supported in the Solr search engine, and indicates which of them are officially supported in Service Manager.

Languages Supported in Solr Search Engine	Officially Supported in Service Manager?
ARABIC	Yes
CHINESE SIMPLIFIED/TRADITIONAL	Yes
CZECH	Yes
DANISH	No
DUTCH (STANDARD)	Yes
ENGLISH	Yes
FINNISH	No
FRENCH	Yes
GERMAN	Yes
GREEK	No
HEBREW	Yes
HUNGARIAN	Yes
INDONESIAN	No
ITALIAN	Yes
JAPANESE	Yes
KOREAN	Yes
NORWEGIAN NYNORSK	No
POLISH	Yes
PORTUGUESE	Yes
ROMANIAN	No
RUSSIAN	Yes
SLOVAK	No
SPANISH	Yes
SWEDISH	No

Languages Supported in Solr Search Engine	Officially Supported in Service Manager?
THAI	No
TURKISH	No
VIETNAMESE	No

Activate a Knowledge Management language

User role: System Administrator

Languages in the Solr search engine are identified by language codes known as "KM Identifier" values. When a KM Identifier value is assigned to a language, the language can be activated and enabled to be used by the search engine.

Note: Languages not containing a valid KM Identifier value are not supported by the search engine and should not be enabled for Knowledge Management. If they are enabled, the search engine will default to the English language processing rules for both searching and indexing.

To activate a language that has a KM Identifier value for the search engine:

Note: By default, the English language is activated.

1. Click **Tailoring > Database Manager**.
2. In the Table field, type **language** and then click **Search**.
3. Double-click the language form and click **Search**. A list of language identification records is displayed.
4. Select the applicable language identification record. The record is displayed.

Note: There are three Chinese language records (Chinese Simplified, Chinese Traditional, and Chinese). Service Manager differentiates between Chinese traditional and Chinese simplified, but the search engine uses only one language file for both languages. To enable Chinese for the search engine, select Chinese.

5. Look at the KM Identifier field. If there is a valid KM Identifier value, then this language can be activated and enabled in the search engine. Following are some examples of the valid KM Identifier values that are stored in the language table.

Language	KM Identifier value
Arabic	ar
English	en
German	de
Spanish	es

6. Select the **Active for Knowledge Management** field to make the language available to be used in the Solr search engine.
7. Click **Save** and **OK**.
8. Users must log out and then log back in for the language activation to take affect.

Enable Languages in the Solr Search Engine

User role: System Administrator

The Solr search engine provides out-of-box languages that can be enabled to allow users to search for knowledge documents using key words in those languages. You can enable or disable languages in the Solr search engine that contain a valid "KM Identifier" value.

Note: HP recommends that you enable only those languages that are applicable to your knowledgebase. By default, English is the only language enabled.

To enable or disable languages in the Solr search engine:

1. Update the applicable `fieldType`, `field`, and `copyField` language entries in the **schemastub.xml** file.
 - a. Locate the **schemastub.xml** file in the Service Manager home directory.
 - i. In the Windows client, the default directory is: *C:\Program Files\HP\Service Manager 9.30*
 - ii. In the Linux environment, the install path is: */apps/sm930/*
 - iii. Once you locate the Service Manager home directory, the path to the **schemastub.xml** file is: */Server/RUN/km/styles/schemastub.xml*
 - b. In the **schemastub.xml** file, find the reference for the language `fieldType` you want to

enable (for example, **text_zh** for the Chinese language) and uncomment the entry.

Note: The following XML comment tags start and end (surround) the language entry:

```
<!--<fieldType> and </fieldType>-->
```

Following is an example of the Chinese language entry:

```
<!-- CHINESE SIMPLIFIED/TRADITIONAL -->

<!-- <fieldType name="text_zh" class="solr.TextField" positionIncrementGap="100">

<analyzer type="index">

<tokenizer class="com.teragram.solr.AsianTaggingTokenizerFactory"
filename="../../kmsearchengine/languages/data/zh.uhtagger" />

</analyzer>

<analyzer type="query">

<tokenizer class="com.teragram.solr.AsianTaggingTokenizerFactory"
filename="../../kmsearchengine/languages/data/zh.uhtagger" />

<filter class="solr.SynonymFilterFactory" synonyms="../../languages/thesaurus/synonyms_
zh.txt"/>

</analyzer>

</fieldType>

-->
```

- c. Locate the solr field section to uncomment the applicable fields. You will see a list of language-specific docbody and adlearn fields (except docbody, docbody_en, adlearn, and adlearn_en will be commented).
- d. Uncomment the applicable language field entry.

Note: Do not comment out the default docbody and adlearn fields. By default, the docbody_en and adlearn_en fields are enabled. You may safely comment them out if your system does not require English language processing.

For Example:

```
<!-- <field name="docbody_zh" type="text_zh" indexed="true" stored="true"
multiValued="true"/> -->
```

```
<!-- <field name="attachment_zh" type="text_zh" indexed="true" stored="true"
multiValued="true"/> -->
```

```
<!-- <field name="adlearn_zh" type="text_zh" indexed="true" stored="true"
multiValued="true"/> -->
```

- e. Uncomment the applicable language `copyField` entry.

Important: The three field entries (`fieldType`, `field`, and `copyField`) must match (you must uncomment all three fields to enable the language).

For example:

```
<!-- <copyField source="docbody_zh" dest="docbody"/> -->
```

```
<!-- <copyField source="attachment_zh" dest="attachment"/> -->
```

- f. Save your changes.
2. Update the Database Manager files.
 - a. Click **Tailoring > Database Manager**.
 - b. In the Table field, type **language** and then click **Search**.
 - c. Double-click the `language` form and click **Search**. A list of language identification records is displayed.
 - d. Select the applicable language identification record. For this example, there are two records: **Chinese Simplified** (zh-Hans) and **Chinese Traditional** (zh-Hant). The record is displayed.
 - e. Select the **Active for Knowledge Management** field to make the language available for Knowledge Management.

Note: The Service Manager Language table may contain language entries that are not supported by the search engine. Only languages with a valid **KM Identifier** value should be enabled.

- f. Click **Save** and **OK**. Your language should now be available in the locale dropdown on the Advanced Search screen, Contribute Knowledge screens, and for the default locale dropdown on the Manage Knowledgebases Type Information tab.
3. To disable a language that is no longer being used, do the following:
 - a. Locate and comment the following entries in the **schemastub.xml** file:
fieldType, field, and copyField

Important: The three field entries (fieldType, field, and copyField) must match (you must comment all three field entries to disable the language).
 - b. Save your changes.
 - c. Go to the Database Manager identification record and uncheck the **Active for Knowledge Management** field to make the language unavailable for Knowledge Management.
 - d. Save your changes.

Create Search Engine Thesaurus Files

The search engine uses the thesaurus when doing a simple search. A synonym search is a type of search that locates occurrences of either the search term or any of its synonyms. For example, a synonym search for computer might return documents that contain laptop or desktop .

Thesaurus expansion happens automatically for terms entered into the simple search screen and is not currently supported for advanced searches. The search engine performs thesaurus expansion on words in the natural language query box. Thesaurus expansion is done using the dictionary that matches your login language. To use a different language dictionary, change the query language parameter on the advanced search form.

Note: A synonym search term containing a phrase is not supported.

The search engine supports thesaurus files (or dictionaries) for individual languages, but none are provided out-of-box. You may create your own thesaurus dictionaries for use when searching content, or thesaurus expansion can occur automatically for terms that are entered into a simple search screen that is not currently supported by advanced search.

To add a thesaurus file for a certain language:

1. Go to the `{SERVICE_MANAGER_HOME}/Search_Engine/kmsearchengine/languages/thesaurus` folder, and create an empty text file named `synonyms_synonyms_<language id code>.txt`.

The thesaurus file name format includes the two-character language id. For example, the English thesaurus text file name is `synonyms_en.txt` and the French thesaurus text file name is `synonyms_fr.txt`.

2. Add content to the thesaurus file. The thesaurus file format is as follows:

```
# blank lines and lines starting with pound are comments.
#Explicit mappings match any token sequence on the left hand side of
  #"=>" and replace with all alternatives on the right hand side.
#Examples:
laptop, desktop => computer
#Equivalent synonyms may be separated with commas
#NOTE: When using commas in files, ensure that single-byte commas
#are used instead of double-byte commas.
#Examples:
foozball , foosball
universe , cosmos
#"computer, laptop, desktop" is equivalent to the explicit mapping:
computer, laptop, desktop => computer
#multiple synonym mapping entries are merged.
foosball => foosball
foozball => fozzball
#is equivalent to
foosball => foosball, fozzball
```

Caution: When using commas to separate terms in files, you must use the single-byte commas instead of double-byte commas.

3. Save the file in UTF-8 encoding.

Caution: Because UTF-8 is part of the Unicode standard which enables you to encode text in practically any script and language, be sure you save your files in UTF-8 encoding.

Modify Stop Words

User role: System Administrator

A stop-word list is a list of terms that can be ignored when the search engine is searching or indexing. Typically, stop-word lists include short and common words or prepositions, such as "a," "the," or "with" in English. However, they may also include longer words, such as long number strings, or words that are too common to be useful as search targets, such as the term "internet."

Stop words are removed from words entered in the "Search for" box unless they are enclosed in double quotes (phrase search). They are not removed during indexing to allow for phrase searching.

Stop words are stored in Service Manager in lists by specific language. Not all languages support stop words (for example, Japanese and Chinese). Adjust the list of stop words by either adding or removing words from this list.

The stop-word list for your log-in language is used by default, and is loaded once when you first log in. Changing the query language parameter on the advanced search screen changes the stop-word list used. The new stop-word list is loaded each time you search in a language other than your log-in language. This may cause a delay in your search being submitted as the stop-word list is loaded. If you need to perform extensive searches in a language other than your log-in language, HP recommends that you log out and then log back in the other language to reduce this delay.

To modify stop words:

1. Click **Knowledge Management > Configuration > Stopwords**.
2. Click **Search**.
3. Select the record for the language code you wish to change.
4. Add a new word or modify an existing word.
5. Click **Save**.

Add a New KM Message to the scmessage Table

User role: System Administrator

Use this procedure to add a new message (token), which makes a label multilingual. After you create this new message, use the Message ID to update the label field for the hitlist you want to be multilingual.

1. Type **db** in the command line or navigate to Database Manager (click **Tailoring > Database Manager**).
2. Type **scmessage** in the Table box.
3. Click **Search**.
4. Type **km** in the Class box of the HP Service Manager Message form.
5. Click **Search**.
6. Review the list of existing km-related system messages.
7. Add a new record where the class is km, the language code is the language of the string being added, the Message ID is unique within the class, and the text string is the text you want to display in the label of the hitlist.
8. Make a note of the Message ID number. You will need it to update the label field and when you add the text strings for the other languages you want to use for this hitlist label.
9. Click **Save** to create the new record.

Use this procedure to add message text in an additional language for an existing message ID.

1. Type **db** in the command line or navigate to Database Manager (click **Tailoring > Database Manager**).
2. Type **scmessage** in the Table text box.
3. Click **Search**.
4. Type **km** in the Class box of the HP Service Manager Message form.
5. Click **Search**.
6. Review the list of existing km-related system messages to find the message number of the message for which you want to add the new language text.
7. Add a new record where the class is km, the language code is the language of the string being added, the Message ID matches the Message ID of the label token for this language, and the text string is the text in the language you want to display in the label of the hitlist.
8. Click **Save** to create the new record.

Create a Hitlist with Multilingual Labels

User role: System Administrator

In addition to creating new document views, administrators can configure the labels for the document view to display in languages other than English.

Note: This procedure is for a hitlist that is not multilingual-enabled. Before you begin, make sure you create a message number ID for each multilingual label and that the message number is defined to display labels in multiple languages.

To create a hitlist with multilingual labels:

Note: The default hitlist is multilingual-enabled in the out-of-box system.

1. Click **Knowledge Management > Configuration > Configure Hitlists**.
2. Click **Search**.
3. Select the hitlist to update.
4. Type the label delimiter (*SCMSG*123*SCMSG*) in the Label field for the label you are making multilingual. The 123 should be the message number ID from the `scmessage` table for the unique message number ID for this label.
5. Continue adding additional label delimiters for each of the multilingual labels in the hitlist.
6. Click **Save**.

Chapter 7: Indexing the Knowledgebases

Indexing is performed in the background. A background process, `KMUpdate`, is responsible for starting the indexer. Indexing a knowledgebase includes submitting records to the `kmknowledgebaseupdates` table.

Caution: There can be only **one** `KMUpdate` process running at any time. Starting more than one `KMUpdate` process causes unpredictable behavior on the search engine server.

When scheduled, `KMUpdate` runs every 5 minutes by default. This Repeat Interval is defined in the `KMUpdate` schedule record. When `KMUpdate` runs, the indexer starts. The indexer first queries for all knowledgebases of `sclib` type. Each knowledgebase has a field called `interval`. There is also an internal field called `current interval`. The indexer first checks that the interval value is not set to zero. An interval of zero causes the indexer to skip any further processing on this knowledgebase and the indexer then moves to the next knowledgebase in the list. If the interval is greater than zero, the indexer compares the value of `interval` to the value of the internal interval. If they do not match, the internal interval is incremented by 1 and saved. The indexer then skips any further processing of this knowledgebase and moves to the next knowledgebase in the list.

When the values of `interval` and internal interval match, the indexer queries the `kmknowledgebaseupdates` table for records matching the knowledgebase and begins processing them. Because of processing time, the intervals cannot be based on elapsed time. An interval of two would be 10 minutes only if all records from any knowledgebase with an interval of one were processed in under five minutes. Once complete, the indexer will move to the next knowledgebase in the list. `KMUpdate` is suspended during indexing so that it cannot start any new indexing until the indexer completes all knowledgebases in its list to prevent overlapping indexing.

Indexing a knowledgebase includes submitting records to the `kmknowledgebaseupdates` table. There are two ways to submit records to the `kmknowledgebaseupdates` table.

- When an administrator selects **Full Reindex** on the Knowledgebase Maintenance form (the Status tab), the system performs the following processing:
 - a. Delete the old knowledgebase index.
 - b. Create a new empty knowledgebase index based on the current knowledgebase settings.
 - c. Remove all records for the knowledgebase from the `kmknowledgebaseupdates` table.

- d. Execute the query provided on the Type Information tab, adding all matching records to the `kmknowledgebaseupdates` table.
- When any record is modified in any table that has a knowledgebase associated with it, there are triggers in the tables associated with a knowledgebase that cause records to be submitted to the `kmknowledgebaseupdates` table.

Managing the KMUpdate process

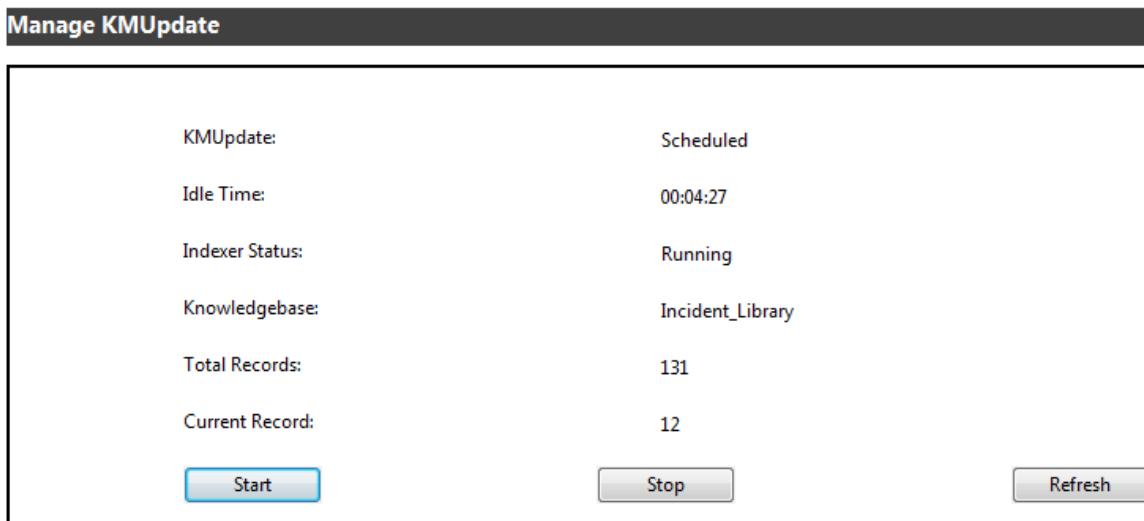
The KMUpdate process controls indexing, which runs in the background. You use the Update Indexes form to stop and restart indexing, and to view the status statistics relating to indexing. If indexing has not started when you stop the KMUpdate process, the interval counter resets and the interval does not begin counting down until you start indexing.

Tip: You can also type `status` in the Command window to display any processes currently running in the system, and this will include the KMUpdate process. You can use the `kill` command to stop indexing or stop any scheduled indexing. Typically, this is only accessible to and done by a System Administrator.

To access the Update Indexes form, navigate to **Knowledge Management > Configuration > Update Indexes**.

Field	Description
KMUpdate:	Displays whether or not the KMUpdate process is scheduled: Scheduled, Not Scheduled .
Idle Time:	When the KMUpdate process is scheduled, this field displays the amount of time the process has been idle. When this count reaches the Repeat Interval of the KMUpdate schedule record (default: 5 minutes), the KMUpdate process starts the indexer. If the indexer finds pending updates to the scheduled knowledgebase, the indexer processes them. Otherwise, the indexer updates the internal interval counter.
Indexer Status:	Displays the status of indexing: Idle , or Running .
Knowledgebase:	Displays the name of the knowledgebase currently being indexed.
Total Records:	Displays the total number of records in the knowledgebase being indexed.
Current Record:	Displays the current number of records that have been processed. Click Refresh to view the most current statistics for indexing that is in progress. (For the web client, click More > Refresh Statistics .)

The following figure shows a scenario where the indexer is indexing the Incident_Library knowledgebase.



Perform a Full Re-Index on a Knowledgebase

User Role: System Administrator, KMAdmin

After you have upgraded from the K2 Search Engine to the Solr Search Engine or after you have installed the Solr Search Engine, you need to perform a full re-indexing for all of your knowledgebases.

Knowledgebase maintenance error reporting

The Knowledgebase Maintenance form contains an Errors tab that lists any errors found during indexing. The listed errors only apply to the selected knowledgebase. Once the error is corrected and the document re-indexed the system removes the error from the list.

The Errors tab displays the document ID for the document containing the error and an error message that identifies the error found during indexing. When a document contains an error, you should edit the document and submit it again so that the document can be indexed during next indexing interval. This will make the document available in the knowledgebase.

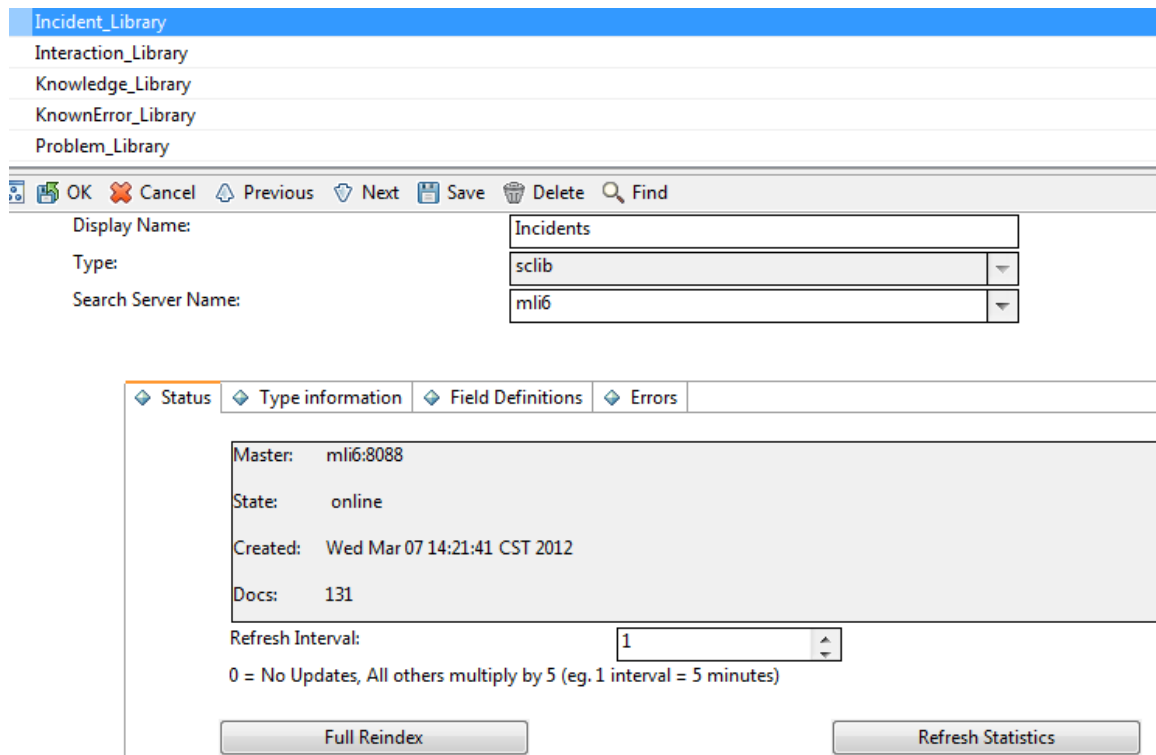
To perform a full re-indexing of a knowledgebase:

1. Navigate to **Knowledge Management > Configuration > Knowledgebases**, and click **Search**.
2. Select a knowledgebase from the list.
3. Check that the **Search Server Name** field contains the name of a search server that is connected.

4. On the **Status** tab, click **Full Reindex**.

The Search Engine starts to index all records in the knowledgebase. Once the indexing is complete, the knowledgebase state changes from **offline** to **online**, and the creation time of the indexes displays in the **Created** field.

5. Click **Refresh Statistics**. The total number of records that have been indexed displays.



6. Click the **Errors** tab to see if any errors occurred during the indexing and fix them as needed.
7. Do a search against this knowledgebase to verify that the search works properly.
 - a. Click **Knowledge Management > Search Knowledgebase**.
 - b. Select the knowledgebase in the **Search In** section.

If the knowledgebase has not been successfully indexed, it will be in offline state and will not display in the **Search In** section.

- c. Enter a search word.
- d. Click **Search**.

A list of search results is displayed, together with a message that resembles the following:
"XXX records found. YYY records searched."

Chapter 8: Enforcing Mandanten Security in Knowledge Management

User role: System Administrator

Service Manager offers a security feature called Mandanten for any searches performed within Service Manager. Because the Knowledge Management module uses a third-party search engine (the Solr search engine), it does not apply the settings for Mandanten protection that may have been defined for Service Manager searches against these tables by the customer. You can utilize Mandanten protection for searches executed by the Knowledge Management module to ensure that all searches against these tables comply with the security requirements defined by Mandanten.

Introduction to Mandanten Security

Typically, Mandanten is set up based on the company of the user who is accessing the system, though it can be set up based on any value in any table that needs to be protected. Mandanten protection is set up on a per-table basis. The operator can be a member of one, many, or no security groups. The security groups (`scsecuritygroups` table) set values that define which records the user is allowed to see based on the content of the mandant field; the mandant field is set up for each table in the `scmandant` record. More flexible queries for each table and security group can be added to the `scaccess` table. However, when a user enters a search anywhere within Service Manager, the Mandanten restrictions are appended to that query upon execution, and restricted records will not be part of the returned record set.

Queries executed outside of Service Manager, such as with the Solr search engine, are not Mandanten protected. Information shown in the Knowledge Management hit list is not yet retrieved from the Service Manager internal files. When you select a record from the hit list for viewing, it will then access the Service Manager internal file (such as `probsummary`) that is under Mandanten protection. Access to the record will then be denied based on the Mandanten restrictions, even if the record was displayed in the hit list. To prevent this from happening, update the KM search security scripts to read Mandanten settings and apply these settings to the hitlist as well. For details, see ["Update a KM Search Security Script for Mandanten Security" on the next page](#).

How to Enable Mandanten Security in Knowledge Management

To enable Mandanten security in Knowledge Management, you need to:

1. Set up Mandanten protection according to the online help documentation:
 - The operator needs to belong to one or many security groups.
 - Security groups must have one or many “include” and/or “exclude” values.
 - The `scmandant` file must have a record for the table to protect and define a field in that table as the mandant field.
2. Ensure that all fields used in the `scmandant` and `scaccess` files are defined in the Knowledgebase record's **Field Definitions** tab. See ["Add an sclib Knowledgebase" on page 32.](#)
3. Modify the search security script for the library that uses Mandanten protection, as described in ["Update a KM Search Security Script for Mandanten Security" below.](#)

Note: Make sure to run the full re-index as an operator without Mandanten limitations, since the Mandanten query that enforces security on the originating table will limit the records read during the re-index operation.

Update a KM Search Security Script for Mandanten Security

User role: System Administrator

To enable Mandanten security for a library, you need to update its Search Security Script. The following steps use the Incident Library as an example.

1. Open the `Incident_Library_kmsearchsecurity` JavaScript in the ScriptLibrary. The out-of-box security script is as follows:

```
function getSecurityInfo(user, record)
{
```

```
return "";
}
```

2. Change the script above to the following.

```
* Convert the security control from K2 to Solr.
* The security settings come from the security group / field, and Mandanten
security restriction query string.
* There is no need to change the record type for different libraries,
* because the record passed in defines the library and contains the record type
information.
*/

if (!('unique' in Array.prototype)) { /* add function to remove the duplicated
element in array, sort the array first */
    Array.prototype.unique = function(that /* opt */) {
        this.sort();
        for ( var i = 1; i < this.length; i++ ) {
            if ( this[i] === this[ i - 1 ] ) {
                this.splice( i--, 1 );
            }
        }
    };
}

if (!('forEach' in Array.prototype)) { /* add function to iterate element in
array */
    Array.prototype.forEach = function(action, that /* opt */) {
        for ( var i = 0, n = this.length; i < n; i++ )
            if (i in this)
                action.call(that, this[i], i, this);
    };
}

/**
 * generate the security information according to the user passed in.
 * @param user, user name
 * @param record, obsoleted parameter.
 * @returns {String}
 */
function getSecurityInfo(user, record) { /* record stands for the current
library record */
    var querystr = "";
    /* as the table name is already stored in library record, no need to change it
for different library*/
    var tablename = record.sclibtablename;
    var operatorFile = new SCFile("operator");
    var rc = operatorFile.doSelect("name=\"" + user + "\"");
```

```

    if (rc == RC_SUCCESS) {
        var securitygroups = operatorFile.security_group;
        var mandant = new SCFile("scmandant");
        var rc_mandant = mandant.doSelect("filename=\"" + tablename + "\"");
        if (rc_mandant == RC_SUCCESS) {
            var mandantField = mandant.fieldname;
            querystr = checkSecurityGroup(securitygroups, mandantField, tablename);
        }
    }
    if (querystr == "()") {
        querystr = "";
    }
    return querystr;
}

/**
 * generate the query string according to the security group passed in
 * @param groups, security group ID
 * @param field, field to add security control
 * @returns {String}, query string contains the mandant security setting.
 */
function checkSecurityGroup(groups, field, tablename) {
    if(field == null) field = "";
    field = field.replace('.', '');
    var secgroup = new SCFile("scsecuritygroup");
    var sm_query = "";
    var includelist = new Array(); /* value list within the security control */
    var excludelist = new Array(); /* value list outside the security control */
    var restrictions = new Array();
    var sql = "security.id isin " + system.functions.str(groups);
    var rc = secgroup.doSelect(sql);
    while(rc == RC_SUCCESS){
        includeList = includeList.concat(secgroup.include.toArray());
        excludeList = excludeList.concat(secgroup.exclude.toArray());
        var restriction = get_scaccess_query(secgroup.security_id, tablename);
        if(restriction) restrictions.push(restriction);
        rc = secgroup.getNext();
    }
    if(field && includeList.length>0 ){
        includeList.unique();
        sm_query = field + ":((" + includeList.toString() + ")"; /* add include value
list to query */
    }

    if(excludeList.length>0 && field){ /** add exclude value list to query */
        excludeList.unique();
        if(sm_query) {

```



```

        sm_query += " AND ";
    }
    sm_query += "NOT " + field + ":((" + excludeList.toString() +)";
}
sm_query = sm_query.replace(/,/g, ' '); /* add a whitespace after separator in
array ',' => ', ' */

restrictions.forEach(function(item){ /* add mandanten restrict query to km
search string*/
    if(sm_query){
        sm_query += ' AND ' + item;
    } else {
        sm_query += item;
    }
});
return sm_query;
}

/**
 * get restrict query for security group
 * TODO: get the solr field name from library
 * for now, the solr field name simply removes the '.'(dot) in SCFile field
name.
 */
function get_scaccess_query(groupId, tablename){
    var restriction = '';
    var scaccess=new SCFile("scaccess");
    var rc=scaccess.doSelect("filename=\"\" + tablename +\"\"\" + "and
security.id=\\\""+groupId+"\\\"");
    if(rc == RC_SUCCESS){
        var query = scaccess.restricting_query;
        restriction = parse(query);
    }
    return restriction;
}

/**
 * convert the restrict query to solr query string
 * @param sql query string in restrict query.
 * @returns {String}
 */
function parse(sql){
    sql = replaceAND(sql);
    sql = replaceOR(sql);
    sql = replaceISIN(sql);
    sql = replaceNotEqual(sql);
    sql = replaceEqual(sql);
}

```

```
        sql = replaceEmbrace(sql);
        sql = convertToSolrFields(sql);
        sql = replaceQuote(sql);
        return sql;
    }

/**
 * convert 'and' to 'AND'
 * @param str
 * @returns
 */
function replaceAND(str){
    return str.replace(/\band\b/g, 'AND');
}

/**
 * convert 'or' to 'OR'
 * @param str
 * @returns
 */
function replaceOR(str){
    return str.replace(/\bor\b/g, 'OR');
}

/**
 * convert 'isin' to ':'
 * @param str
 * @returns
 */
function replaceISIN(str){
    return str.replace(/\s*isin\s*/, ':');
}

/**
 * convert 'a~b' to 'NOT a:b'
 * @param str
 * @returns
 */
function replaceNotEqual(str) {
    return str.replace(/\w+\s*~=\s*\w+/g, function(word){
        word = word.replace(/\s+/g, '');
        return 'NOT '+ word.replace(/~/, ':');
    });
}

/**
 * convert 'a=b' to 'a:b'
 * @param str
```

```

    * @returns
    */
function replaceEqual(str) {
    return str.replace(/=/g, ':');
}

/**
 * convert to solr fields
 * 'a.b.c.d' => 'abcd'
 */
function convertToSolrFields(str) {
    return str.replace(/[\.](\w+)/g, "$1");
}

/**
 * convert '{a,b}' to '(a, b)', add a whitespace after comma to adjust solr
query format
 * @param str
 * @returns
 */
function replaceEmbrace(str) {
    str = str.replace(/{/g, '(');
    str = str.replace(/}/g, ')');
    str = str.replace(/,/g, ', ');
    return str;
}

/**
 * replace double/single quote in given string
 * @param str
 * @returns
 */
function replaceQuote(str) {
    str = str.replace(/"'/g, '');
    return str;
}

```

Differences between Solr and K2 Search Security Scripts

Service Manager applications versions earlier than 9.30 support the K2 search engine. Instructions on how to enable Mandanten security for the Knowledge Management module that uses the K2 search engine can be found in this knowledge document:

<https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM436754>

For Knowledge Management that uses the Solr search engine, you should follow the instructions in this document. Pay attention to the following differences:

- For the Solr search engine, you no longer need to change the file names for individual libraries, because each knowledgebase record now already contains the record type information, which the search security script will parse as search configuration.
- For the Solr search engine, each search security script already contains the security control defined in Mandanten security groups/ Mandanten security restriction queries. The operator in each query will be parsed based on the mappings listed in the following table to match the Solr query.

K2	Solr
field=value	field:value
field~value	NOT field:value
field='value'	field:value
field.fieldext:value	fieldfieldext:value
field isin {value0,value1}	field:(value0, value1) Note: There is a whitespace after the comma.
field=value or field=value1	field:value OR field:value1

Chapter 9: Searching the Knowledgebases

When searching the knowledgebases, you can perform:

- A **simple search** where you search for a text string;
- An **advanced search** where you can provide several search parameters. In the advanced search you can specify the knowledgebases to search and the document categories to search. You can also specify a set of filtering parameters, such as exact phrase and creation date.
- A search within the search results after doing an initial search or advanced search.

Each of the knowledgebases has different fields that are indexed for searching, so when you search a knowledgebase, provide search parameters that match the fields in the knowledgebase. For example, the knowledge articles have a title and author field. When you view an incident, the out-of-box system displays the incident number, incident description, and solution for closed incidents.

The out-of-box system includes five separate knowledgebases that can be searched collectively or separately, depending upon what information you are searching for. In addition to making your search more efficient by specifying a knowledgebase, it is also best to search with a limited number of document categories. When you search, your log-in profile determines what information you can search and view.

You can use the Knowledgebase Maintenance feature to add additional knowledgebases for searching. These knowledgebases are created from a table in Service Manager (sclib) or by using web crawling to browse and index an external web site (weplib).

For more information, see the Service Manager help.

Send Documentation Feedback

If you have comments about this document, you can [contact the documentation team](#) by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Solr Search Engine Guide (Service Manager 9.41)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to ovdoc-ITSM@hp.com.

We appreciate your feedback!

