



# **Setting Up and Managing ENS (IBM InfoSphere Global Name Management Enterprise Name Search) Version 6.0**

Prepared by: Richard Strangfeld

## **Edition**

This edition applies to Version 6.0 IBM InfoSphere Global Name Management (product number 5724-Q20) and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 2013, 2017.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

## Table of Contents

<b>Table of Contents .....</b>	<b>1</b>
<b>1 Introduction.....</b>	<b>5</b>
1.1 In a Nutshell – Setting up an ENS system .....	5
1.2 ENS Architecture – Overview .....	6
1.3 Platforms and Prerequisites .....	10
1.4 Local vs. Distributed Installation .....	10
1.5 Profiles.....	11
1.6 Requirements for hosts in a distributed installation .....	12
<b>2 Before installing.....</b>	<b>13</b>
2.1 In a Nutshell .....	13
2.2 Database Installation.....	13
2.3 Database Creation .....	13
2.4 Common file paths.....	14
2.5 Defining a host alias.....	14
2.6 Adjusting System Limits .....	15
<b>3 Installing ENS .....</b>	<b>16</b>
3.1 In a Nutshell .....	16
3.2 Starting the ENS 6.0 Installer.....	16
3.3 Title Screen.....	16
3.4 Welcome Screen .....	17
3.5 Installation Environment .....	17
3.6 Copyright Screen.....	18
3.7 Installation Type Screen (UNIX only) .....	18
3.8 Installation Location Screen (Local Version) .....	19
3.9 Installation Location Screen (Distributed Version).....	19
3.10 ENS Security Screen.....	20
3.11 WebSphere Security Screen .....	21
3.12 Type of Database Screen.....	21
3.13 JDBC Driver Screen (DB2 version) .....	22
3.14 Database Information Screen (DB2 version).....	23
3.15 Database Information Screen (Oracle version) .....	24
3.16 Database Update Options Screen .....	25
3.17 Create Server Profile Options Screen.....	26
3.18 Server Profile Information Screen.....	26
3.19 Pre-Installation Summary Screen.....	28
3.20 Installing ENS Progress Screen.....	28
3.21 Install Complete Screen .....	29
3.22 Adjusting Database Settings for Good ENS and NameLoader Performance .....	29
<b>4 Managing Profiles with the ENS Configuration Utility.....</b>	<b>31</b>
4.1 In a Nutshell .....	31
4.2 Using ensclu to add a profile .....	31
4.3 Using ensclu to remove a profile .....	33
<b>5 Adding Names with the ENS NameLoader .....</b>	<b>36</b>

- 5.1 In a Nutshell – a Simple Example .....36
- 5.2 The CSV Data File .....37
  - 5.2.1 Data fields in a name file (normal format) ..... 37
  - 5.2.2 Specifying Category Explicitly..... 38
  - 5.2.3 Specifying Culture Explicitly..... 39
  - 5.2.4 Extended File Format – Specifying Script Type Explicitly..... 40
  - 5.2.5 Use UTF-8 Encoding ..... 41
  - 5.2.6 Duplicate External Ids ..... 41
- 5.3 The NameLoader Configuration File .....41
- 5.4 Name Lists.....42
- 5.5 Permissions .....43
- 5.6 Removing a Name List.....43
- 5.7 Clearing All Names .....44
- 5.8 Revision Tracking.....44
- 5.9 Command-Line Arguments for the NameLoader.....45
- 5.10 NameLoader Performance.....46
- 5.11 Updating Database Statistics for Better Performance .....47
- 5.12 NameLoader configuration file tuning (advanced topic) .....47
  - 5.12.1 Queue sizes ..... 48
  - 5.12.2 Threads for analysis and writing ..... 48
  - 5.12.3 Database batch size..... 48
  - 5.12.4 Fine tuning ..... 49
- 6 Starting and Stopping ENS Servers .....50**
- 7 Adding Hosts in a Distributed ENS Installation .....51**
  - 7.1 In a nutshell.....51
  - 7.2 The first host in a distributed installation .....51
  - 7.3 Additional hosts: Meeting Prerequisites .....52
  - 7.4 Additional hosts: Starting ENS.....52
  - 7.5 Additional hosts: Configuring.....52
- 8 Managing Components Using the Console GUI.....54**
  - 8.1 In a Nutshell .....54
  - 8.2 Browsing to the Console GUI .....54
  - 8.3 The Console GUI: Cell Configuration Tab .....55
    - 8.3.1 Choosing which servers will have searchers..... 56
    - 8.3.2 Choosing which servers will have dispatchers..... 56
    - 8.3.3 Choosing a distribution strategy..... 56
    - 8.3.4 Applying your configuration..... 58
  - 8.4 The Console GUI: Server Status Tab .....59
  - 8.5 Starting the cell.....59
  - 8.6 When the cell is running.....60
  - 8.7 Stopping the cell.....61
  - 8.8 Starting and stopping components .....61
    - 8.8.1 The Auto Start Flag for Components..... 62
    - 8.8.2 Stopping a Component ..... 63
    - 8.8.3 Starting a Component..... 65
  - 8.9 Mirroring a Searcher .....65
  - 8.10 Mirroring a Server .....67
  - 8.11 Unconfiguring a Searcher .....69

8.12	Other information in the ENS Console.....	70
8.12.1	URL for the Search GUI.....	70
8.12.2	The product version .....	71
8.12.3	Number of names. ....	71
<b>9</b>	<b>Managing Users, Passwords, and Roles .....</b>	<b>72</b>
9.1	In a Nutshell .....	72
9.2	Liberty Security .....	72
9.3	Supported Roles in ENS .....	72
9.4	Fine-grained control at the name list level.....	73
9.5	The users.xml file.....	74
9.5.1	Adding a new user.....	74
9.5.2	Changing your own password.....	74
9.5.3	Resetting a password.....	75
9.5.4	Specifying permissions (group membership) for a user.....	75
9.5.5	Deleting a user.....	77
<b>10</b>	<b>Searching for Names .....</b>	<b>78</b>
<b>11</b>	<b>Uninstalling ENS.....</b>	<b>80</b>
<b>12</b>	<b>Advanced Topics.....</b>	<b>81</b>
12.1	NameWorks Configuration File .....	81
12.2	Using SSL .....	81
<b>13</b>	<b>What's New in ENS Version 6 .....</b>	<b>82</b>
13.1	Multiple names can have the same external id.....	82
13.2	Support for Polish, Portuguese, and Turkish cultures.....	82
13.3	Organization names can have cultures .....	82
13.4	Explicit Script Type and Extended File Format .....	82
13.5	NameLoader allows commas within names .....	83
13.6	Support for DB2 10.5 and Oracle 12c .....	83
13.7	WebSphere Liberty .....	83

## 1 Introduction

IBM InfoSphere Global Name Management Enterprise Name Search, or ENS, is software that allows sophisticated name searching based on Global Name Management NameWorks to be done at the enterprise level. It uses proven IBM middleware and database components to:

- Perform efficient NameWorks-based searches in large lists of names.
- Distribute those names across multiple servers to allow larger lists, higher performance, and reliability through redundancy.
- Make it easy to manage those name lists and servers.

This document tells how to install, set up, and manage an ENS system. It describes use of the ENS installer, `enscu` (ENS configuration utility) and the ENS console to set up and manage an installation of ENS on one or more host machines.

This document is *not* about using ENS to search for names. For that, see the document titled “Searching for Names with ENS”.

### 1.1 In a Nutshell – Setting up an ENS system

Setting up the server side of an ENS system involves the following steps:

1. Prerequisites: installation type, database, folders, host alias (section 2).
2. Install ENS, creating the first profile and adjusting database settings (section 3 and notably 3.22).
3. If needed, define additional profiles using `enscu` (section 4).
4. Use the ENS NameLoader to add initial names (section 5).
5. Start ENS servers on host machines (section 6).
6. Use the browser-based ENS console GUI (section 8) to:
  - a. Configure the cell (specify which components to run on each server), and
  - b. Start the cell.

**Figure 1 - Steps to set up ENS**

This gives you a running ENS installation where people can search for names.

People using the system to search for names don’t need to do all this. Once you’ve set up and started an ENS system, those users just point their browser (or application) to the search web application that your system provides, log in, and start searching as in section 10.

You may find that you can set up a simple one-server local installation by following just the “In a Nutshell” instructions at the beginning of each section in this document. The other subsections provide more detail when you need it.

After setting up your ENS system, you may want to do the following from time to time to manage it:

1. Add and remove names using nameLoader and/or web services (section 5)
2. Important: After adding names, update database statistics to optimize performance (section 5.11)
3. Add/delete/modify users and permissions by editing the users.xml file. Users can modify passwords with a GUI screen. (section 9).
4. Use the ENS console GUI to manage components on servers (section 8).
5. Start servers on new hosts and/or stop them on old ones (section 7).
6. To adjust capacity, use encsu to add, delete, and modify profiles (section 4).

### 1.2 ENS Architecture – Overview

The major parts of ENS are:

**Cell** – All of the servers in your ENS system, running on one or more host machines.

**Profile** – A named configuration for a logical ENS server to be run on one or more host machines. A given host can run multiple profiles to utilize multiple processors.

**Server** – A container running on a host machine based on configuration details defined in some profile. This may contain a dispatcher, searcher, console, and dashboard. Typically it contains just some of these.

**Dispatcher** – Component that receives search requests from web clients, delegates them to one or more searchers, aggregates the results, and returns them to the client.

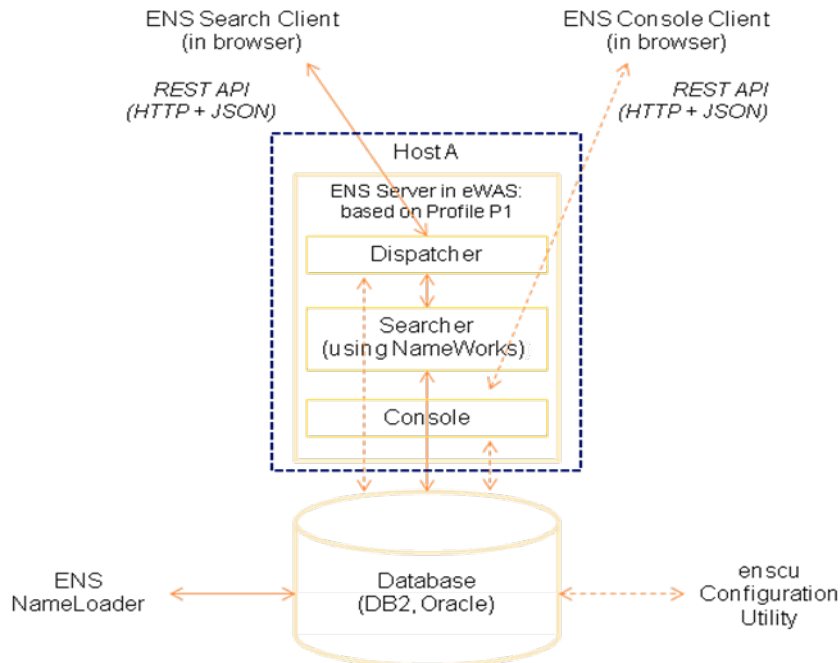
**Searcher** – Component wrapping an instance of NameWorks holding names. It takes requests from a dispatcher, searches for names, and returns results to the dispatcher.

**Console** – Component for monitoring the ENS system and controlling its components.

**Dashboard** – A read-only version of the console. Often omitted.

**NameLoader** – Reads names from a file, analyzes, and loads them into the database.

**Web Clients** – Run in browsers to provide GUIs for searching and for the console.



**Figure 2- Simple one-server local ENS system**

Figure 2 above shows these components as arranged in a simple one-server system. This system has a single host with a single server containing three components: a dispatcher, a searcher, and a console.

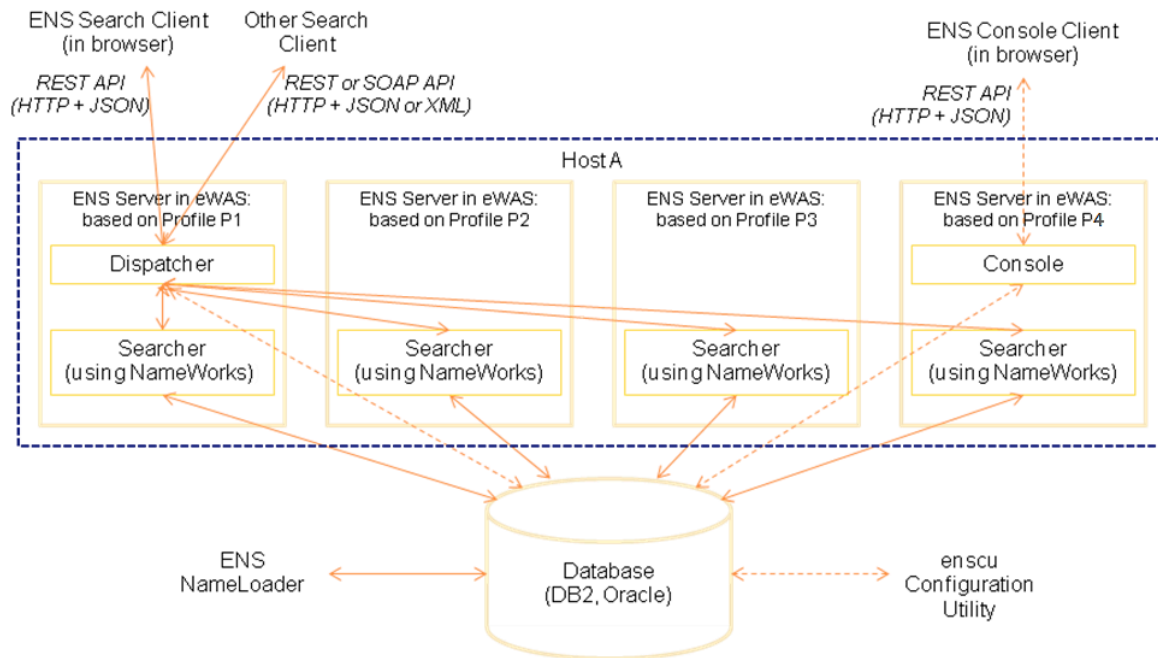
In that system, a **search GUI client** in the browser sends web service requests to the dispatcher, which passes them on to the searcher. The searcher uses NameWorks to search and match names, then uses information in the database to map search results back to original forms of the names. The searcher passes these results back to the dispatcher which returns them to the client where they are displayed.

A **console GUI client** sends web service requests to the server-side console component to let an administrative user monitor and control the system and its components.

To load names into the system, the **NameLoader** program reads names from csv files. It uses NameWorks to analyze them; then stores information in the database about name lists, source names read, search names to use internally in searchers, and the relationships among them.

## Setting Up and Managing ENS

You can increase capacity and take advantage of multiple processors even in a one-host ENS installation by defining multiple profiles using the `enscu` utility. Figure 3 below shows such an installation. Each profile is a separate instance of WebSphere Liberty, acting as a separate logical server. Typically one such server has a dispatcher, and all have searchers. Each searcher has its own GNM NameWorks instance holding some subset of the names to be searched.



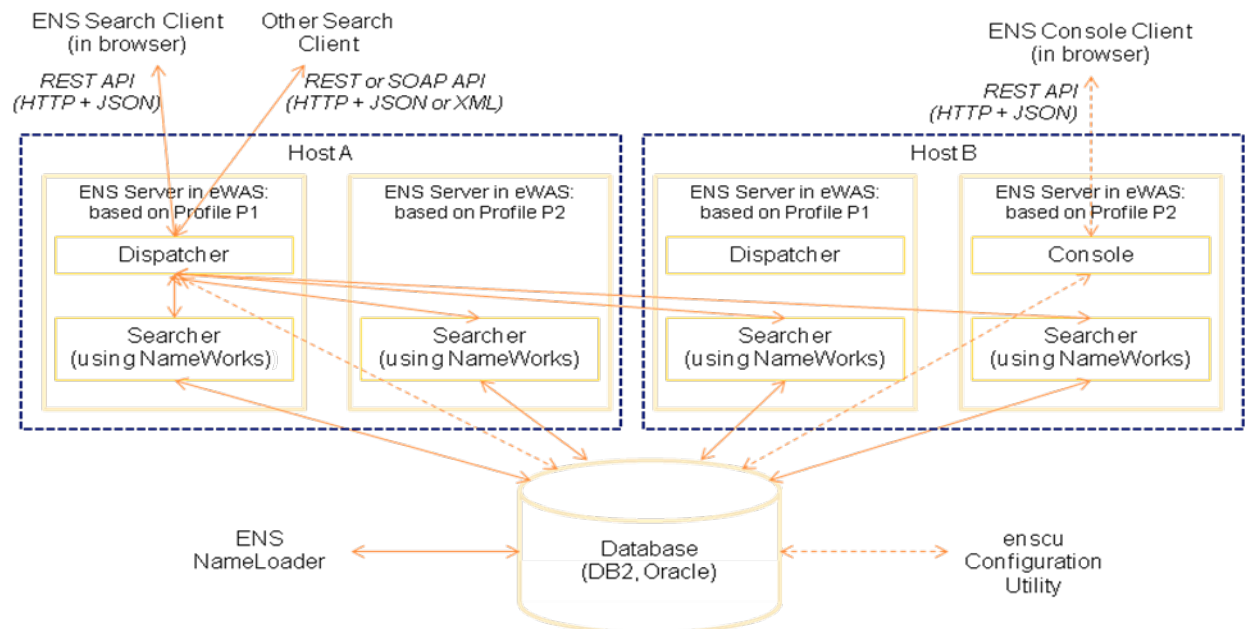
**Figure 3 – Local ENS installation (single host) with multiple profiles**

When a dispatcher receives a search request, it delegates the searching to one or more searchers. It merges results from those searchers and returns them to the client.



## Setting Up and Managing ENS

To scale further, you can use a distributed ENS system running on multiple hosts. Figure 4 shows such a system with multiple hosts and multiple profiles. The components and their behavior is the same as in the simpler system, but here the names are apportioned among multiple searchers on different servers. As before, when a dispatcher receives a search request, it delegates the searching to one or more searchers. It merges their results before returning them to the client. It merges their results before returning them to the client.



**Figure 4 - Distributed ENS system - multiple hosts, two profiles**

### 1.3 Platforms and Prerequisites

ENS runs on the following server platforms, all 64-bit:

Operating System	Architecture	ENS Installation Type
AIX 7.1	pSeries	local or distributed
RedHat 6	xSeries	local or distributed
RedHat 7	xSeries	local or distributed
RedHat 7 for Power	pSeries	local or distributed
RedHat 7 for Z	zSeries	local or distributed
Ubuntu 14	xSeries	local or distributed
Ubuntu 15 for Power	pSeries	local or distributed
Windows 2008 R2, 2012 R2	xSeries	local only
Windows 7	xSeries	local only (for demos)

**Table 1 - Supported platforms for ENS**

You must provide your own database when using ENS. Currently this can be DB2 (version 10.5 or later) or Oracle (12c or later). The database instance need not be on the machine where you are installing ENS; only its client software needs to be available locally.

The client GUIs for searching and the console have been tested in Firefox (ESR 45) and Internet Explorer (11), though they also run in Chrome.

### 1.4 Local vs. Distributed Installation

ENS can be installed in two different ways:

- A *local* installation, where all server-side components run on a single host machine. This is most useful for smaller systems and for demos.
- A *distributed* installation, with components running on multiple machines. This has advantages for capacity, redundancy, and performance. A typical production installation uses this approach.

Even a single-host local installation can have more than one logical server running on it. All of the following combinations are possible:

Host and Server Options	Example	Comments
1. Local: One physical host machine with one logical server.	Host A runs profile P1.	For small systems and demos.
2. Local: One host with several logical servers.	Host A runs profiles P1 and P2.	Takes better advantage of multiple processors.
3. Distributed: N hosts, with one logical server each.	Host A runs profile P1. Host B runs profile P1. Host C runs profile P1.	Spreads the load between hosts for better performance, capacity, and redundancy.
4. Distributed: N hosts, with several logical servers each.	Host A runs profiles P1 and P2. Host B runs profiles P1 and P2. Host C runs profiles P1 and P2.	Spreads the load as above, and takes better advantage of multiple processors.

**Table 2 - Host and server options**

Note that in any case, *clients* don't have to be on the same machine. People using the search web application or other search clients can do so from other machines.

### 1.5 Profiles

An ENS *profile* definition is a named set of configuration information that you create, to define the ports that a server will use.

An *ENS server* is logically an ENS profile running on a host machine. Under the covers, it is an instance of a WebSphere Liberty profile running on that machine, using the ports and other information specified in some ENS profile definition, and hosting ENS components in an ENS web application. The installer sets up the files that are needed for that Liberty profile and web application.

In Table 2 above, Host A running profile P1 constitutes one ENS server, and Host A running profile P2 is another. If you want to run more than one server on some host machine (as in cases 2 and 4), they must each use a different profile, so that they each use different ports.

When you run the installer to set up a *local* installation, you are defining one profile and installing it on a host. In cases 1 and 2 above, you use the installer to define profile P1 and install it on host A. The configuration files and libraries used by ENS are placed in the file system for host A. In case 2 where you want multiple profiles, you run the ENS Configuration Utility (*enscu*) after installation as in section 4.2 to define the second profile P2.

In a *distributed* system (cases 3 and 4 above), each profile definition is used by servers on different hosts. Since the configuration information defining a particular profile is held in a shared file system, the servers all reference a single definition of that profile.

Thus when you run the installer for a distributed installation, you are defining the first profile in shared files on your network. This makes it available for use in any host that meets certain requirements described below. You run the installer on host A, but what you're setting up is not really specific to host A; it can be used by numerous hosts as long as they meet the requirements.

So in cases 3 and 4, you use the installer to define profile P1 for use on multiple hosts. Then in case 4 where you want additional profiles, you run the *enscu* utility after installation (see section 4.2) to define P2 and any other profiles.

Once you have defined profiles, then on any hosts that meet the requirements below, you can start up ENS servers that use these profiles (see section 6).

### 1.6 Requirements for hosts in a distributed installation

In a distributed installation, several things must be true about each host machine being used:

1. Each host must have the same operating system, processor, architecture, and database client software.
2. Each host must have the same file paths defined for three things:
  - a. The database driver location.
  - b. A networked location that all machines can reach. This is a shared folder location where all hosts can find the same configuration information and libraries. The installer populates this folder.
  - c. A local location, not in shared file storage, where each machine can store logs and other host-specific information.
3. In the `/etc/hosts` file on each host, an alias (named “enshost” by convention) must be defined that points to that host (see section 2.5).

Note that in a distributed installation you don't have to decide in advance what host machines you're going to use in your installation. You just run the installer on one host having paths as above, and the installer sets up profile information and scripts in the shared folder.

After installation, you can add more host machines to the system by just ensuring that they meet the above requirements regarding machine configuration, file paths, and host alias, and then running the start profile commands (see section 6) on each one to start the wanted profiles. See section 7 for more on this.

## 2 Before installing

### 2.1 In a Nutshell

Before installing ENS, you need to have the following:

1. A proper installation of DB2 10.5+ or Oracle 12c, with client libraries available, and an environment variable named DB2DIR or ORACLE\_HOME pointing to them.
2. A database instance with empty schema (for DB2), or an empty user/schema (for Oracle).
3. For a distributed installation, file paths set up in a way that will be the same on each host machine for the JDBC libraries, for a shared space for common files used on different hosts, and for a local space for logs and other per-host files.
4. For a distributed installation, a host alias (normally “enshost”) defined in /etc/hosts pointing to the host itself.
5. If running more than 5 profiles on a given Linux host, you may need to adjust system limits on maximum number of processors and open files per user.

These are all described in more detail below.

### 2.2 Database Installation

Before installing ENS, you must have a proper installation of your database (DB2 10.5+ or Oracle 12c+), with the appropriate environment variable (typically DB2DIR<sup>1</sup> or ORACLE\_HOME) pointing to the installation location. The actual database may be on a remote machine, but its client libraries must be available on the host where you are running. Database installation is outside the scope of this document.

### 2.3 Database Creation

The ENS installer asks for details about an existing database to be used by ENS. Prior to installing ENS you'll need to create a database (DB2) or user/schema (Oracle).

- If using DB2: The database instance must exist before you run the installer. You may choose to create one specifically for ENS, or you may want to use an existing database. Typically, this decision is made in consultation with your DBA.

---

<sup>1</sup> The ENS installer looks first for an environment variable named DB2DIR to tell it the installed location of DB2 or its client libraries. On Microsoft Windows, some versions of DB2 instead set variables named DB2\_HOME or DB2PATH. The ENS installer looks for these if DB2DIR is not found. If none are found, it assumes that DB2 is not present.

You also have the option (recommended, but ask your DBA) of specifying a schema that the installer is to create within the database. If you do, the schema must be initially empty or nonexistent.

- If using Oracle: The database user / schema must exist before you run the installer, and must initially be empty.

**Important:** With either database, make sure that the database or schema instance that you create uses UTF-8 character encoding. This will allow it to properly store and handle the non-Roman characters used in some languages.

### 2.4 Common file paths

In a distributed installation, make sure that each host you are using has the common file paths mentioned in section 1.6. Specifically,

1. The path for the JDBC driver must be the same on each host that you use.<sup>2</sup>
2. You provide some common networked folder for shared ENS configuration and library files. That location must be reached by the same path on each host that you use.
3. You provide some local folder for ENS logs and other per-host files. This needs to exist on each host that you use, with the same path on each one.

These do not apply in a local installation.

### 2.5 Defining a host alias

In a distributed installation, this manual step is required before installing ENS. Each host machine in the distributed installation must have an alias (typically named “enshost”) defined in its hosts file to refer to the machine’s own IP address. This allows the same ENS scripts to run unmodified on each machine, and to operate on the right machine in each case.

Typically you define such an alias in Unix by adding an entry to the `/etc/hosts` file. Editing this file normally requires root permission. As an example, if your machine has IP address 9.35.149.122, the line in the hosts file would be:

```
9.35.149.122  enshost
```

In a local installation, this alias is not required, but if you are installing a local demo version on a laptop machine that you plan to use on different networks (where it may be assigned different IP addresses), it’s a good idea to use such an alias there too.

---

<sup>2</sup> Each host in your ENS environment must have the same database driver located at the same path. It is suggested that this be specified via a dynamic link to a shared location. If all of the machines point to the same location, you can make changes to the database driver location without having to make changes on every machine.

Define `enshost` in your hosts file as below, and use it as your host name in sections 3.8 and 3.14. Then you can update the `enshost` definition in the hosts file when your laptop moves to a different network and is assigned a different IP address. For Windows, the hosts file is typically `c:/windows/System32/drivers/etc/hosts`. You can edit it by running Notepad in administrator mode, as follows:

1. Click Start > All Programs > Accessories to display a list of Windows accessory programs
2. Right-click Notepad and choose “Run as administrator” from the popup menu.
3. Open `c:\windows\System32\drivers\etc\hosts`, and add or update a line like the above to define `enshost` as an alias for your IP address. Save and close.

### 2.6 Adjusting System Limits

If you are planning to run more than a few profiles on a given Linux or AIX host, you may need to adjust the configured system limits on the number of open files and processes allowed per user.

Some Linux versions default the limit on number of processes (and threads) per user to 2048, but Red Hat Linux 6 and later has a default of 1024. An ENS profile is a web application running in an instance of WebSphere Liberty. That instance can use 100 or more threads. Eight ENS profiles running on a host could exceed the limit, leading to java out of memory dumps and error messages about “Failed to create a thread: retval -1073741830, errno 11 received”. This issue is documented at <http://www.ibm.com/support/docview.wss?uid=swg21648497>.

As shown there, the solution is increase the system limit on the number of threads per user. See <http://www.ibm.com/support/docview.wss?uid=swg21469413> for more detailed instructions. Note that for some RHEL versions the command “`ulimit -n unlimited`” is not allowed and leaves the limit in place; in those cases a high number should be specified rather than “unlimited”.

This step doesn’t have to happen before installation time, but if you leave it out and then try to run too many ENS profiles on a machine you’ll run into the limits.

Similarly, in an ENS installation with numerous profiles you may need to adjust the system limits on the number of open files.

## 3 Installing ENS

### 3.1 In a Nutshell

You install ENS by running the ENS installer program included in the GNM 6.0 GA distribution. In a distributed installation, you only run the installer on the first host in the system. Installation does not need to be repeated on each host used.

The ENS installer program installs ENS and all of the components that it uses, apart from DB2 or Oracle. This includes Java, WebSphere Liberty (used internally by ENS servers), and the libraries and other components comprising NameWorks.

It places files in the file system (local for a local installation, shared folders for a distributed one), and initializes tables and other details in the database.

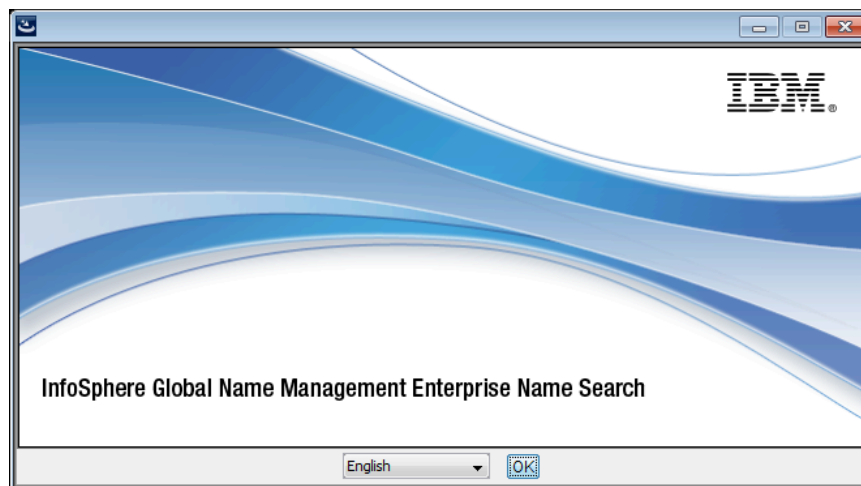
### 3.2 Starting the ENS 6.0 Installer

The installer program is in the /Disk1/InstData/VM directory on the product media for the ENS 6.0 release, and is named `install.exe` or `install.bin`.

When you run the install program, you see an Install Anywhere user interface which prompts you for inputs. This is a graphical UI on a system with Microsoft Windows or X Windows, and text-based elsewhere<sup>3</sup>. The information collected is similar in both cases. Some details vary depending on your platform, choice of database, and choice of local versus distributed installation.

The panels shown are as follows.

### 3.3 Title Screen



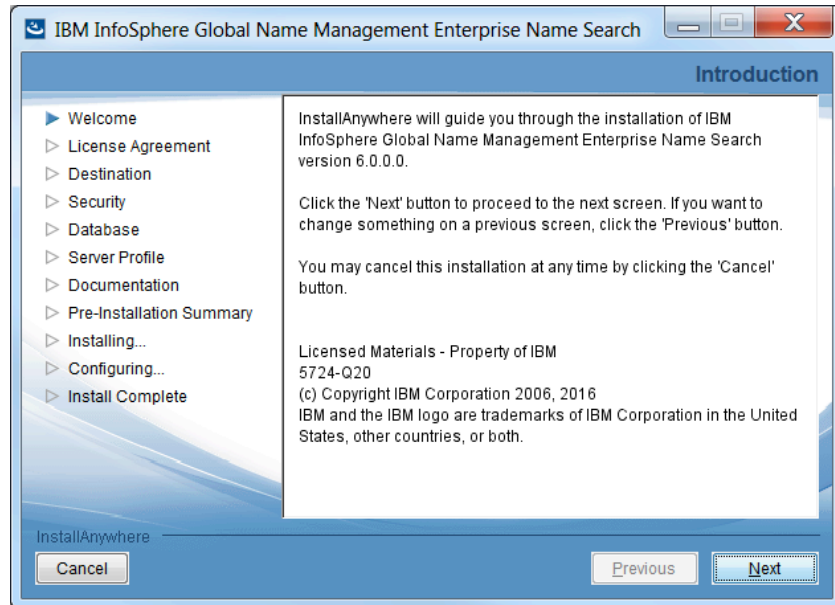
---

<sup>3</sup> You can also force the text-based interface by adding the command-line argument "`-i console`".



Assuming English is your preferred language, just click OK.

### 3.4 Welcome Screen



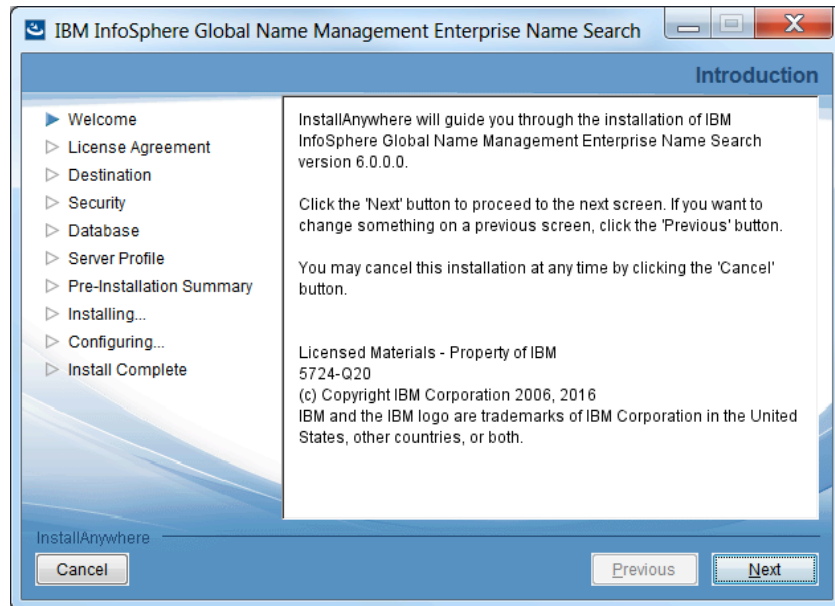
Click Next to continue.

### 3.5 Installation Environment



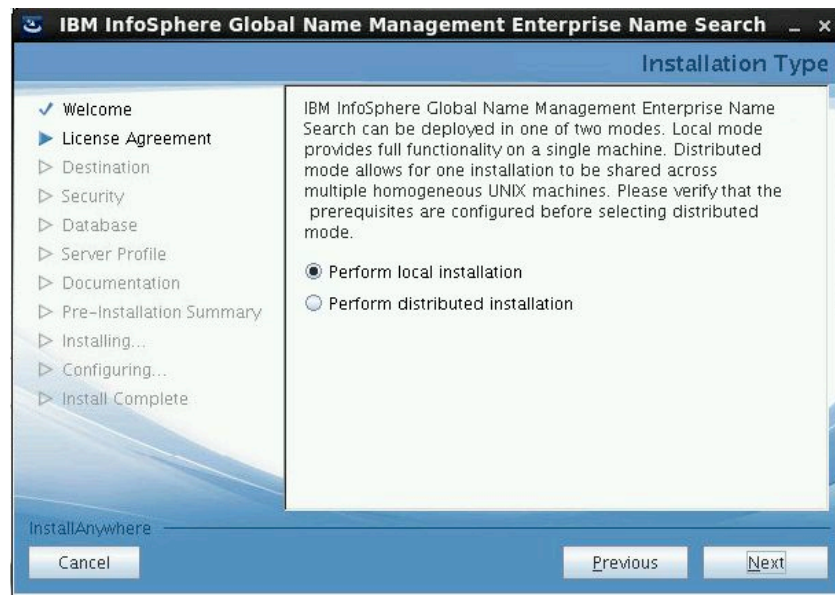
This screen determines the license under which you are installing ENS. If installing for test or evaluation, choose "Non-Production". In most situations, you will choose "Production". Then click Next.

### 3.6 Copyright Screen



If you accept the terms, select the appropriate radio button and click Next.

### 3.7 Installation Type Screen (UNIX only)

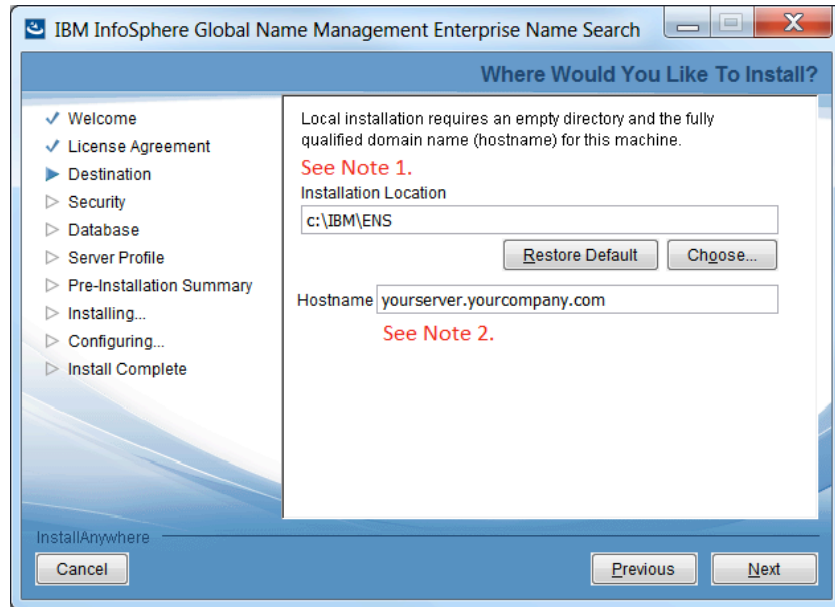


This screen is only displayed when running the installer on a UNIX platform. It lets the user choose the installation type, as described in section 1.4. This can be:

- local (to be run on only one host machine), or
- distributed, where the installation and configuration files are placed in shared network folders, and used by servers running on different hosts

### 3.8 Installation Location Screen (Local Version)

If you choose local installation, the next screen looks like this:



Specify a location path and the host name for the current machine. We'll refer to this location as <ensroot> elsewhere in this document.

**Note 1:** In a Windows environment, due to an installer issue, you are strongly encouraged to choose an installation location path that does not include spaces, such as "C:\IBM\ENS".

The installer tries to guess your host name, but you should check what it finds. Make sure that the name shown is a legitimate fully qualified DNS domain name reachable from elsewhere, not just a local network name.

Make sure as well that the name does not contain underscores, and does not begin or end with a hyphen.

**Note 2:** If you defined a host alias in step 2.5, type that alias (e.g. enshost) here in the Hostname field.

### 3.9 Installation Location Screen (Distributed Version)

For distributed installation, the screen looks like this:

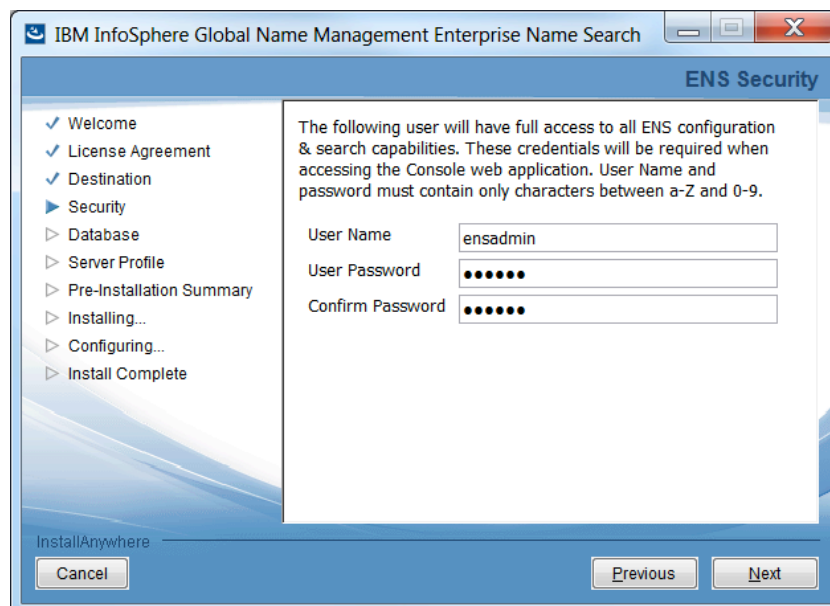


It prompts for two file-system locations you set up before running the installer (see section 2.4).

- A networked folder that will be shared among all hosts using this installation, using the same path on all hosts to refer to the same shared folder, and
- A folder that is local to each host using this installation, using the same path on all hosts to refer to the host-specific local folder

This screen also asks the user for the host alias name (recommended name enshost) you defined in /etc/hosts on each machine as in section 2.5.

### 3.10 ENS Security Screen



On this screen, you specify a user name and password that an administrator will use within ENS to log in to the ENS console and search GUIs. During installation you just define a single administrative user. After installation you can add and manage additional users as in section 9.

### 3.11 WebSphere Security Screen



On this screen, you specify a password that will be used in some rare circumstances to manage the instance(s) of WebSphere Liberty installed and used by ENS.

### 3.12 Type of Database Screen



Assuming you have a proper installation of the DB2 client or Oracle client software, and that the installer finds the DB2DIR or ORACLE\_HOME environment variable from that installation, this screen lists one or both databases. Choose one.

### 3.13 JDBC Driver Screen (DB2 version)



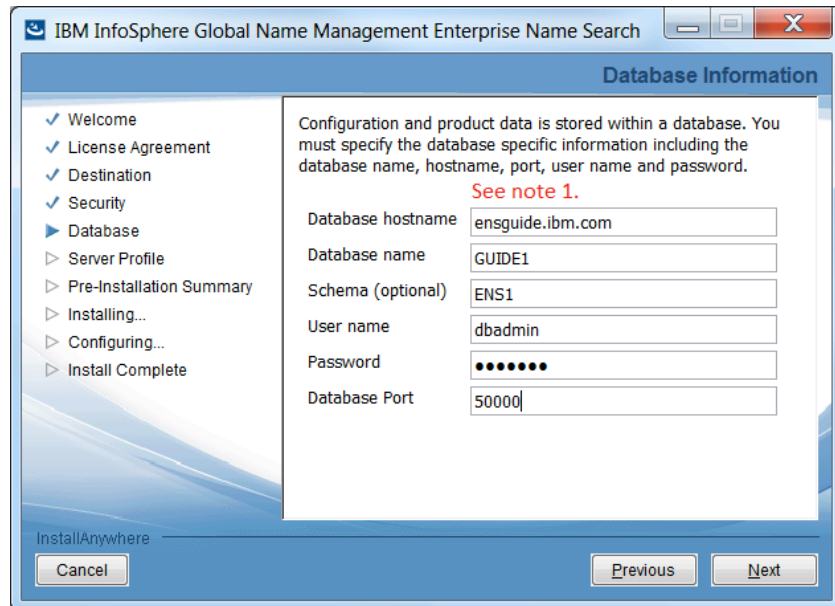
Here you specify the location of the JDBC drivers for your database. Normally the installer determines this location from your DB2DIR or ORACLE\_HOME environment variable, and you can use the default value shown. If the default value is blank, this may indicate a problem with your database installation.

With DB2 on Windows, the path shown is based on the DB2DIR environment variable, or possibly an environment variable named DB2\_HOME or DB2PATH as discussed in section 2.2.

On a Windows machine, make sure that the path shown here follows 8.3 file naming conventions and does not contain spaces. For example, if your DB2 drivers are at `c:\Program Files\ibm\SQLLIB\java`, change that to

```
c:\progra~1\ibm\SQLLIB\java
```

### 3.14 Database Information Screen (DB2 version)



**DB2 version:** On this screen you specify details used to access your database, if using DB2. All but the schema name are details about the existing database instance: the host name of the machine where it is located, the database name, the user name and password for accessing it, and the port at which it is available. The schema name is for a new schema that the installer will create in your database. This is optional but is generally recommended, especially if using a shared database.

See your DBA (or whoever is responsible for your database) for the information required on this screen, and for advice on use and naming of a schema. If you specify a schema name, make sure it is legal in DB2.<sup>4</sup>

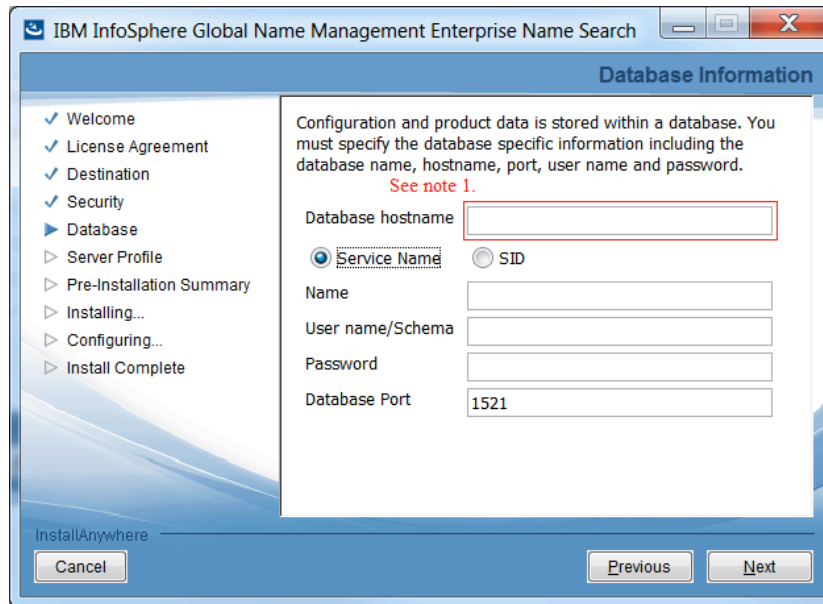
When you click Next, the installer tests the information you provided to make sure that it can connect to the database. If successful, the next screen is the Database Update Options Screen.

Note 1: Local installation only: if your database is on the same host where you are installing and you defined a host alias in section 2.5, you can use that alias (e.g. enshost) in the hostname field here.

---

<sup>4</sup> DB2 schema naming rules are at [https://www.ibm.com/support/knowledgecenter/SSEPGG\\_10.5.0/com.ibm.db2.luw.admin.dboj.doc/doc/c0007246.html](https://www.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.admin.dboj.doc/doc/c0007246.html). As a quick guideline, keep names to 8 characters or less, avoid names that start with "SYS" or "SESSION", and avoid spaces and special characters. "ENS1" in the example is a good schema name.

### 3.15 Database Information Screen (Oracle version)



**Oracle version:** On this screen you specify details used to access your database, if using Oracle.

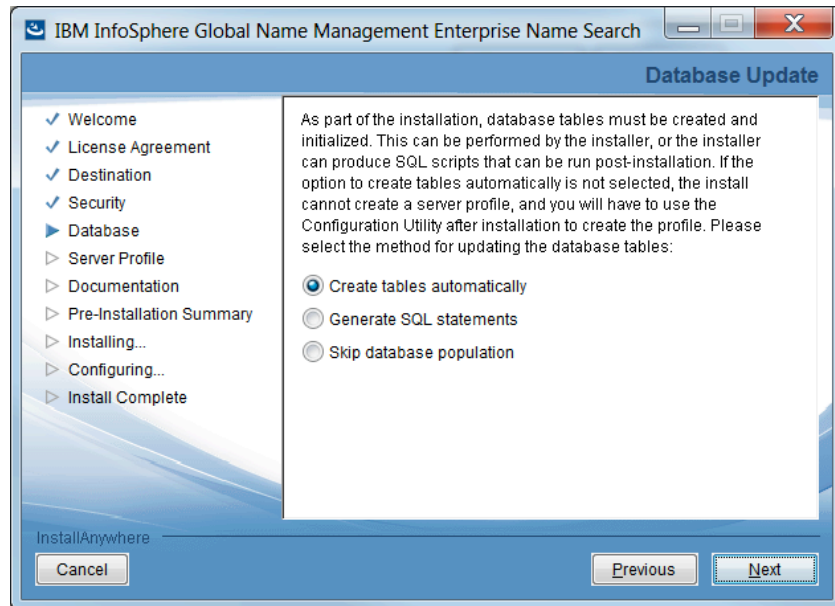
Everything shown here is information about a database instance that you created before running the installer. This includes the host name of the machine where it is located, the SID or service name identifying a particular database instance, the user name/schema that you set up before running the ENS installer, the password, and the port. See your DBA (or whoever is responsible for your database) for this information.

When you click Next, the installer tests the information you provided to make sure that it can connect to the database. If successful, the next screen is the Database Update Options Screen.

Note 1: For local installation only: if your database is on the same host where you are installing and you defined a host alias in section 2.5, you can use that alias (e.g. enshost) in the hostname field here.



### 3.16 Database Update Options Screen



This screen asks whether the installer should do all of the database initialization work now, or save SQL statements in a file for you to use later.

- The first option is the simplest: to have the installer do the work now. For a demo scenario or a simple local installation, this is a good choice.
- If you are working with a DBA who wants to see and approve changes to the database, you can choose the second option. This will generate a file containing SQL statements for your DBA to review and apply. In that case you'll be able to continue with the installation but you won't be able to let the installer create the profile. You'll have to create the profile in a separate post-installation step using the ENS configuration utility, described later in this document.
- The third option, to skip database population, is for advanced users in special situations.

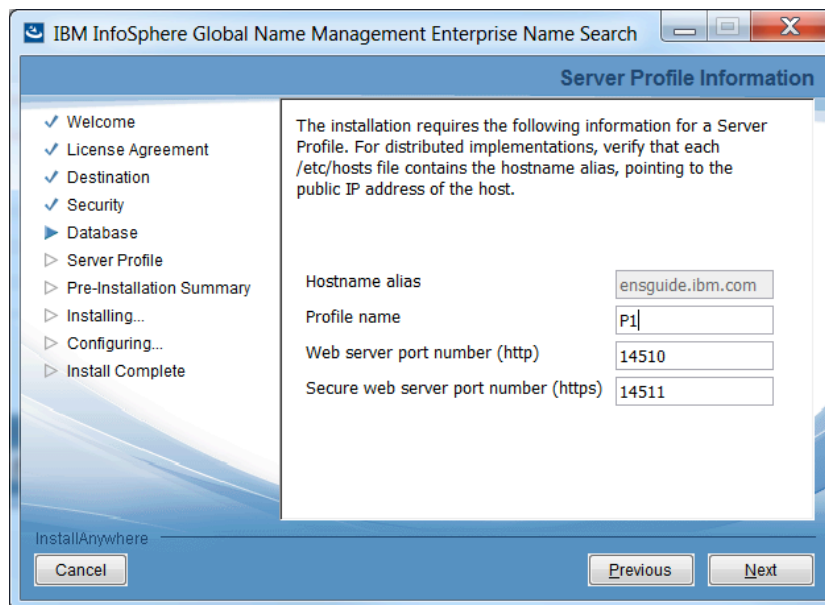
### 3.17 Create Server Profile Options Screen



Similarly, the installer asks whether it should create the server profile now or let you do it later with a separate utility.

- Assuming you let the installer initialize your database on the previous screen, the strong recommendation is to let it create the profile now.
- Alternatively you can choose to use the ENS configuration utility later, after installation.

### 3.18 Server Profile Information Screen



On this screen you specify a name for the initial profile that your ENS system will have and the ports that it will use.

In a local installation, the first item shown is the hostname that you specified in section 3.8. In a distributed installation, it is the hostname alias (typically “enshost”) that you defined in `/etc/hosts` before running the installer and then specified in section 3.9.

The second item on this screen is the profile name. Each profile in your ENS installation has a unique name.<sup>5</sup> This name is used to identify profiles when you are managing them in the ENS console GUI or the configuration utility. Internally it also becomes the name of a WebSphere Liberty profile. This name should be short and should avoid special characters and spaces. The name “P1” shown in the example is a good choice. (Then you can use P2 for a second profile, etc.)

The remaining items are port numbers for HTTP and HTTPS. Choose port numbers to avoid conflicts with other profiles and with other applications on the same machine. One suggested scheme is to use the default values (14510 and 14511) for the first profile, then use 14521 and 14522 for the second profile, and so on.

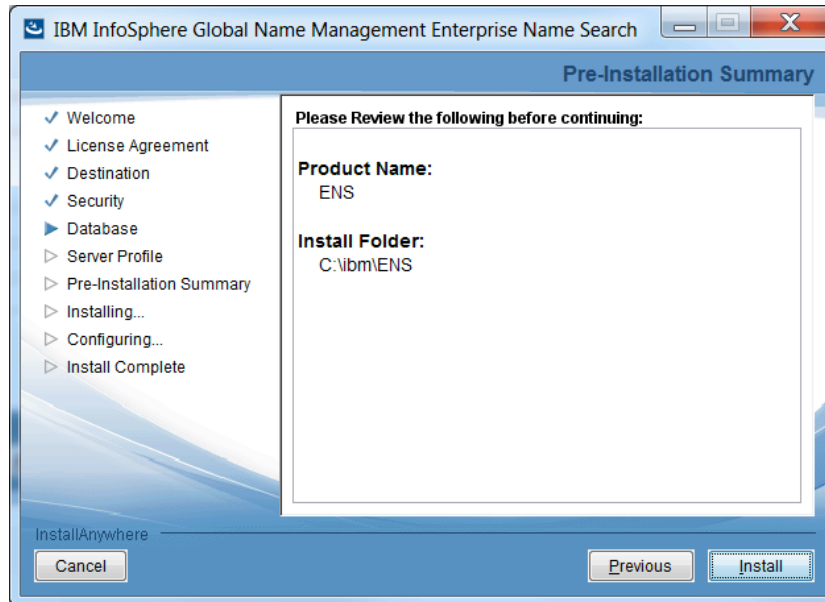
The installer verifies that the ports you specify are not in use by any program currently running on the host where you are running the installer. It doesn’t know about ports used by programs that are not currently running.

In a distributed installation where you’re defining a profile that will run on several different hosts, you need to choose port numbers that are free from conflict on all of the hosts where this profile will run. The installer can check for current conflicts on the host where the installer itself is running, but it doesn’t know about any other hosts you intend to use. It’s up to you to avoid conflicts on all of them.

---

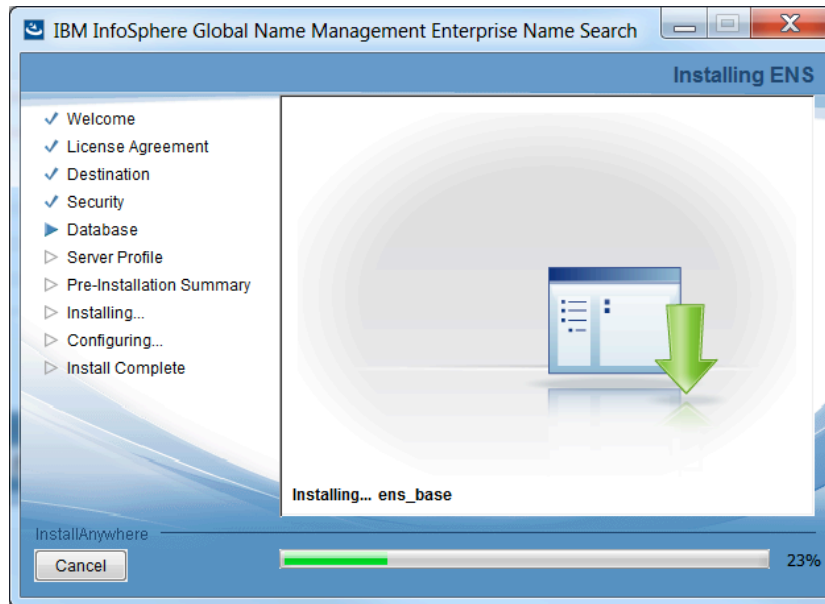
<sup>5</sup> Recall that here in the ENS installer we’re only creating the first profile; we’ll use the ENS configuration utility (`enscu`) after installation to create any others.

### 3.19 Pre-Installation Summary Screen



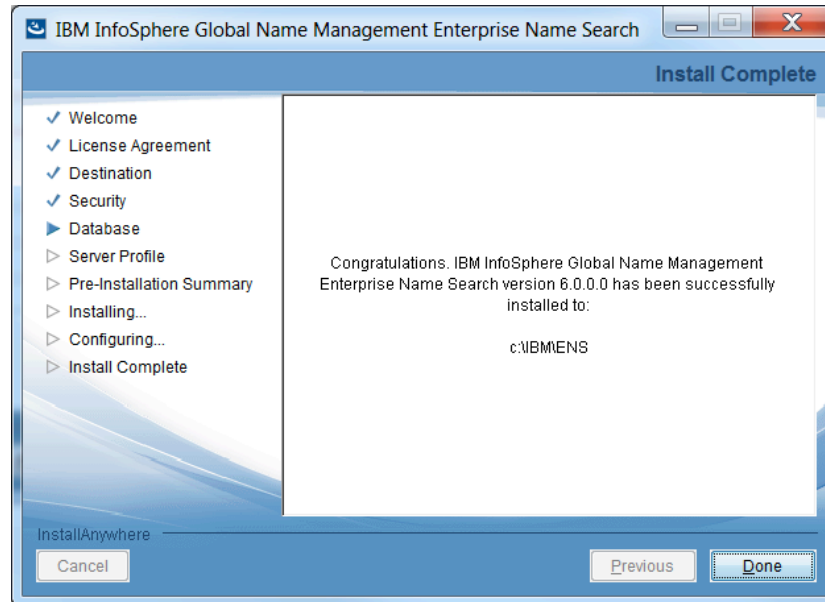
This screen just summarizes what is being installed, and in which folder. When you click “Install”, the installer makes actual changes to files and the database.

### 3.20 Installing ENS Progress Screen



This screen shows progress and activities during the installation.

### 3.21 Install Complete Screen



When you see this screen, your installation has succeeded. Assuming you chose to have the installer create database tables and a profile, you now have a runnable ENS system.

NOTE: Before proceeding, it is strongly recommended for performance reasons that, after creating database tables, you adjust database settings as in the next section.

Once you have done this, you can try out ENS by adding some names as in section 5, starting servers as in section 6, configuring and starting the cell as in section 8, and searching as in section 10.

### 3.22 Adjusting Database Settings for Good ENS and NameLoader Performance

ENS uses the database to persist names in their original source form and in an analyzed “search name” form, with a m:n mapping between the two forms. Significant database use occurs at several times:

- When adding source and search names to the database using NameLoader,
- At searcher startup time, when loading search names from the database into GNM NameWorks inside the searchers.
- During a name search, when mapping from found search names back to source name forms.

In each of these situations, good performance depends on appropriate database settings. Database settings should be adjusted as soon as the database tables are created, typically as the next step after ENS installation.

## Setting Up and Managing ENS

This is usually best done by a DBA, but if you are on your own with your own database in a demo situation, the following DB2 settings changes can help NameLoader and other ENS performance substantially.

This example assumes a database and schema named “ensdemo” and “ens” respectively.

```
db2 update db cfg for ensdemo using LOGFILSIZ 200000
db2 update db cfg for ensdemo using LOGPRIMARY 3
db2 connect to ensdemo
db2 alter table ens.ens_search_name append on
db2 alter table ens.ens_source_name append on
db2 alter table ens.ens_search_source_name append on
db2set DB2_APPENDERS_PER_PAGE=1
db2set DB2_LOGGER_NON_BUFFERED_IO=ON
db2set DB2_INLIST_TO_NLJN=yes

db2stop    or    db2stop force    (again, assuming you're not sharing the database)
db2start
```

Make sure that your DB2 database does not have the setting `DB2_KEEPTABLELOCK=CONNECTION`. This is not compatible with ENS, and can cause deadlocks during server startup in a multi-server ENS installation.

If you are using Oracle, it is recommended that you have `query_rewrite_enabled=true` and `query_rewrite_integrity=trusted` to allow ENS to drop and create functional indexes as needed when configuring cells.

## 4 Managing Profiles with the ENS Configuration Utility

### 4.1 In a Nutshell

A simple ENS system (whether local or distributed) has just one profile<sup>6</sup>, usually created by the installer. If that's all you want, you can skip this section.

Even if you intend to add profiles, you can feel free to run your ENS installation first as a single-profile system with some sample names to try it out.

When you're ready to add additional profiles, you do so using the ENS Configuration Utility program (`enscu`). On the command line in the `<ensroot>/bin` folder, type:

```
enscu
```

Then you follow prompts to add a profile, specifying a profile name and its port numbers. Based on this, the program creates a WebSphere Liberty profile and modifies entries in the ENS database to describe it. More details follow.

### 4.2 Using `enscu` to add a profile

The `enscu` program has a text-based user interface. It displays prompts in the console, and waits for you to type responses.

The following transcript shows an `enscu` session that adds profile P2. The user's inputs are highlighted *like this*.

```
C:\projects\ensguide\bin>enscu
```

```
=====
```

```
Main Menu
```

```
-----
```

```
Choose a screen to edit related configuration properties.
```

```
Screen 3 is [Not Applicable] if this configuration utility cannot connect to the Enterprise Name Search database, or if the Enterprise Name Search database has already been initialized.
```

```
Screen 4 will be [Not Applicable] if this configuration utility cannot connect to the Enterprise Name Search database, or if the Enterprise Name Search database has not yet been initialized. In such case, use screens 2 and/or 3 to set up the Enterprise Name Search database.
```

- 1- Introduction / Instructions
- 2- Database Properties Configuration

---

<sup>6</sup> See section 1.5 for an explanation of profiles.

## Setting Up and Managing ENS

[Not Applicable]- Database Content Initialization

4- Profiles

5- Exit

=====  
Enterprise Name Server Profiles List  
-----

This screen lists existing server profiles. Select a profile to view or modify configuration details or to delete the profile. Select the last option to add a server profile.

- 1- P1
- 2- Create new profile

ENTER NUMBER, "back", or "exit": **2**      << user chooses to create a new profile

=====  
Profile Configuration  
-----

Use this screen to create an Enterprise Name Search profile.

Server Profile Name: **P2**      << new profile name and  
Web server port number (http): **14520**      << ports, different from  
Secure web server port number (https): **14521**      << those in other profiles

You entered the following values for the new profile:

Server Profile Name: P2  
Web server port number (http): 14520  
Secure web server port number (https): 14521

Do you want to create a new profile with those values?

"Y" to create profile / "N" to edit above values / "back" to return to

Profiles Screen: **y**      << user confirms

Creating a new server profile. This process might take more than 1 minute to complete.

Processing.....

The profile has been successfully created.

Related changes have been made to all previously existing profiles.

In order for the applied changes to take effect, you must restart all servers.      << See note below

Press ENTER to continue:      << user presses ENTER

=====  
Enterprise Name Server Profiles List  
-----



## Setting Up and Managing ENS

This screen lists existing server profiles. Select a profile to view or modify configuration details or to delete the profile. Select the last option to add a server profile.

- 1- P1
- 2- P2
- 3- Create new profile

```
ENTER NUMBER, "back", or "exit": exit          << user types "exit"
Cleaning up...
Program exited
```

```
C:\projects\ensguide\bin>
```

Adding a new profile is usually done as part of initial setup after installation, as part of setting up enough logical servers to provide the capacity you need. It is not a frequent operation in an established system, but it can be done as needed. If you add a new profile in a running system, you'll need to restart servers (running other profiles) in that system as in section 6, so that they can detect and communicate with new servers based on the new profile.

If you add a new profile in a system where you have already done cell configuration (section 8.3), you'll probably also want to redo the cell configuration, typically designating that new servers running the new profile should run searchers. You'll want to start those servers (see section 6) so that they are known to the system before you run cell configuration.

You should only create one ENS profile at a time, rather than running multiple copies of `enscu` concurrently. This is because one step in profile creation is the generation of an SSL encryption key between the new profile and each existing profile. These keys are used for SSL/HTTPS secure communication between profiles, when that feature is enabled.

Creating a large number of profiles can take some time because of the additional encryption keys involved. For example, if you create 15 profiles, creation of the 15th profile requires creation of 14 encryption keys. Typical ENS installations do not involve such large numbers of profiles.

### 4.3 Using `enscu` to remove a profile

Removing a profile is not a frequent operation. You do it if you have decided that you want a different configuration with fewer profiles, e.g. to leave more processors free for other purposes.

Before removing a profile, you should first make sure that there are no servers running (on any host, if you have a distributed installation) using that profile. To stop such servers, use stop commands as in section 6. The ENS console (section 8) is helpful in seeing what servers are running on which hosts.

## Setting Up and Managing ENS

In this example, we'll remove profile P2. We'll first cd to <ensroot>/bin and run the stop-P2 script, repeating this step on each host where a P2 server is running.

```
C:\projects\ensguide\bin>stop-P2
```

After that, we'll run ensclu (just once, not once per host) to remove the profile, as follows.

```
C:\projects\ensguide\bin>ensclu
```

```
=====
```

Main Menu

-----

Choose a screen to edit related configuration properties.

Screen 3 is [Not Applicable] if this configuration utility cannot connect to the Enterprise Name Search database, or if the Enterprise Name Search database has already been initialized.

Screens 4, 5, and 6 will be [Not Applicable] if this configuration utility cannot connect to the Enterprise Name Search database, or if the Enterprise Name Search database has not yet been initialized. In such case, use screens 2 and/or 3 to set up the Enterprise Name Search database.

- 1- Introduction / Instructions
- 2- Database Properties Configuration
- [Not Applicable]- Database Content Initialization
- 4- Profiles
- 5- Exit

```
ENTER NUMBER or "exit": 4                    << User chooses to work with profiles
```

```
=====
```

Enterprise Name Server Profiles List

-----

This screen lists existing server profiles. Select a profile to view or modify configuration details or to delete the profile. Select the last option to add a server profile.

- 1- P1
- 2- P2
- 3- Create new profile

```
ENTER NUMBER, "back", or "exit": 2            << User selects profile P2
```

```
=====
```

Profile Configuration

-----

Use this screen to modify an existing Enterprise Name Search profile.

## Setting Up and Managing ENS

```
Server Profile Name: p2
1- Web server port number (http): 14520
2- Secure web server port number (https): 14521
[Not Applicable]- Revert to original values
[Not Applicable]- Apply changes
5- Delete
ENTER NUMBER, "back", or "exit": 5 << User chooses to delete
Are you sure you want to delete this profile (Y/N)? y << User confirms
```

```
Before continuing, you must stop all servers started from the profile that is
about to be deleted. << Make sure you do this.
Press ENTER to continue: << User presses ENTER
```

```
Deleting the profile. This process might take more than 1 minute to complete.
Processing.....
```

The profile has been successfully deleted.

```
If you have already used the Enterprise Name Search Console's Cell
Configuration screen to assign searchers to servers, you may now need to
revise that configuration. << True. See note below.
Press ENTER to continue: << User presses ENTER
```

```
=====
Enterprise Name Server Profiles List
-----
```

This screen lists existing server profiles. Select a profile to view or modify configuration details or to delete the profile. Select the last option to add a server profile.

- 1- P1
- 2- Create new profile

```
ENTER NUMBER, "back", or "exit": exit << User types "exit"
Cleaning up...
Program exited
```

```
C:\ projects\ensguide\bin>
```

When you delete a profile, any server using that profile on any hosts no longer exists. If you were using a cell configuration that relied on some such server to provide a searcher handling some portion of your names, that configuration is no longer valid.

In other words, if you delete a profile from a system where you have done cell configuration (as in section 8.3), you'll need to redo that cell configuration with the new smaller set of servers.

## 5 Adding Names with the ENS NameLoader

The best way to add a list of names to your ENS system is with the ENS nameLoader program. This is a command-line program found in the <ensroot>/bin folder. It reads names from a csv (comma-separated-value) file that you provide, and stores them in ENS for searching. Each time you run nameLoader, it populates (or adds to) one name list, taking names from one input file.

After adding any significant number of names, you should update database statistics to optimize database performance as in section 5.11. Also be sure that after installation you adjusted database settings as in .

### 5.1 In a Nutshell – a Simple Example

As a simple example, let's say we want to add a list of customers to our ENS system. We'll run nameLoader and give it a file containing the names. That file might look like this:

```
Abitbol,Miro,615984
Aboulhosn,Majed,404820
Abitbol Bensimon,Maurice,422711
Aboulnaga,Fayza,757735
Aboumrاد,Hacib,853451
Aboumrاد Padua,Alfredo,875315
...
```

Let's say that this data is in a file named c:\data\customers.csv and we want to populate a CUSTOMERS list in our ENS system. We can run nameLoader by going to our <ensroot>/bin folder and running a command like this:

```
nameLoader -nlc "CUSTOMERS" -nld "Our customers" -in c:\data\customers.csv -dbu
<userid> -dbp <password>
```

The arguments specify the name and description for the name list we are creating, the name of the input file with names, and the database credentials to be used<sup>7</sup>.

When we run this command, nameLoader adds or updates the CUSTOMER name list in our ENS installation. It stores names in the database, both in their original source form and in a search-name form that is the result of NameWorks' analysis and parsing. Later when we start up ENS servers with searcher components, those searchers load these search names for use in search queries.

---

<sup>7</sup> If you prefer not to include credentials on the command line for security reasons, you can omit them there. NameLoader will prompt for them and mask your inputs.

This simple example is enough to get you started using nameLoader, but read on for more information about the data file, configuration files, and other aspects of the ENS nameLoader.

## 5.2 The CSV Data File

In the simple example above, each line in the data file had a surname, given name, and external id. Several more fields are possible, as follows:

### 5.2.1 Data fields in a name file (normal format)

In the normal file format, the data file contains the following comma-separated fields:

	Field	Notes
1	Surname, full personal name, or organization name	For a parsed personal name, the surname goes here. For a single-field personal name, the full name (including given name and surname) goes here. For an organization, the full org name goes here.
2	Given name	Only used for parsed personal name. Otherwise blank.
3	External id	Some ID that is meaningful to the customer. This is often an id in some database external to ENS. ENS treats it as an opaque identifier. Starting in ENS version 6, ids do not need to be unique. See section 5.2.5.
4	Surname culture (or full name or organization name culture)	Optional. A surname (or full name) culture enum value from Table 4. Knowing the culture of a name can help NameWorks searches use culture-appropriate name matching rules. Usually this field is left blank and ENS attempts to determine the culture using NameWorks analysis. For personal names this analysis considers the script type and the actual content of the name. For organization names it considers only the script type. If you already know the culture of a name and want to override NameWorks' determination, you can specify it explicitly here.
5	Given name culture	Optional, used for personal names only. Same notes as for surname culture.
6	Category	Optional. This can be: The letter "p" for a personal name or The letter "o" for an organization name. or The digit "0" for a personal name, or The digit "1" for an organization name. If not specified: <ul style="list-style-type: none"> <li>• If the name has both surname and given name parts, it is assumed to be personal.</li> <li>• If doCategorize is true in loader.config (see Table 7), NameWorks analysis attempts to determine a category.</li> <li>• Otherwise it assumes the name is personal.</li> </ul>

**Table 3 - Data Fields in a Name File – Standard Format**

The simple example had personal names with separate surname and given name:

```
Smith, John, 1234567
```

Alternatively, the given name and surname may be combined in a single field like this (note the double comma):

```
John Smith,,1234567
```

You can add a name containing a comma by putting it in quotes. Typically this is done for a full name in the surname field (before the first comma).

```
"Monsters, Inc.",,b5432,,o
```

```
"John Smith, Esq",,a1001
```

As part of its processing of incoming names, nameLoader uses GNM NameWorks facilities to analyze the name, determining the following:

- The category (personal versus organization name):
  - If an input line has an explicit category, that category is used.
  - Otherwise if the name has text in both the surname and given name fields, it is assumed to be personal.
  - Otherwise, if doCategorize is true in the loader configuration file, NameWorks attempts to determine category by examining the name.
  - Otherwise, the name is assumed to be personal.
- For a personal name, the culture for the given and surname based on the name's script and content.
- For an organization name, the culture for the name based on the name's script.
- For a personal name, the best parse(s) identifying the given name and surname parts. This is particularly helpful where the file provides a name as a single field, but even when the incoming name is already parsed as a surname and given names, you can opt to have nameLoader find and include alternative parses that it feels are more plausible.

### 5.2.2 Specifying Category Explicitly

Where you know that a name is for an organization, it's a good idea to indicate that explicitly in the line in the input file.

```
Madison Dodge,,x12345,,o
```

Since this is an organization name, we leave the given name and the two culture fields empty, but terminate them with commas.

If you already know the category on all your names, you can improve search results by marking the category explicitly in the sixth field for any organization names, and

setting doCategorize to false in your loader configuration file as in section 5.3. This prevents unusual personal names being miscategorized as organization names.

### 5.2.3 Specifying Culture Explicitly

NameWorks uses the culture of a stored name to adjust the matching rules used in a search, so that it can apply culture-specific rules when appropriate.

Normally it's best to let NameWorks determine the cultures of the given name and surname as part of its analysis, but if you happen to know the culture of a name you can specify it as part of your data. The culture you specify overrides the one determined by NameWorks.

This might happen, for example, if you know you are adding a number of Korean names and one of them is "June Park". This name could be Korean/Korean or Anglo/Anglo. NameWorks has no way of knowing which culture is correct. If you do know, you can specify a surname culture and given name culture in your input file, like this:

```
Park, June, x12345, 5, 5
```

The values that can be used for the culture fields are as follows:

Code	Meaning	Notes
0	Ambiguous	
1	Anglo	
2	Arabic	
3	Chinese	
4	Hispanic	
5	Korean	
6	Russian	
7	French	
8	German	
9	Thai	
10	Indonesian	
11	Yoruban	
12	Farsi	
13	Pakistani	
14	Indian	
15	Japanese	
16	Afghan	
17	Vietnamese	
18	Polish	
19	Portuguese	
20	Turkish	
38	SouthwestAsian	Non-specific combination of Arabic, Farsi, Indian Pakistani, and Turkish
39	European	Non-specific combination of Anglo, French, German, Hispanic, and Portuguese
40	Han	Non-specific combination of Chinese, Japanese, and Korean
41	Custom culture 1	If defined in your NameWorks configuration
42	Custom culture 2	If defined in your NameWorks configuration

...	...	...
60	Custom culture 20	If defined in your NameWorks configuration

**Table 4 - Culture code values for the csv file**

#### 5.2.4 Extended File Format – Specifying Script Type Explicitly

The usual file format for NameLoader is as described in section 5.2.1. An alternative extended format is available for customers who need to specify script type explicitly for certain Japanese organization names. This is not common; most customers can use the normal format.<sup>8</sup>

To use extended file format, include the line

`extendedInputFormat=true`

in your loader configuration file.

In an input file using extended format, a line for a personal name has these fields:

	Field	Notes
1	External id	Some ID that is meaningful to the customer. This is often an id in some database external to ENS. ENS treats it as an opaque identifier. Starting in ENS version 6, ids do not need to be unique. See section 5.2.5.
2	Category	The letter “P” for a personal name
3	ScriptType	Not used for personal names. This should be left blank.
4	Surname or full personal name	For a parsed personal name, the surname goes here. For a single-field personal name, the full name (including given name and surname) goes here.
5	Given name	Only used for parsed personal name. Otherwise blank.
6	Surname culture	An optional culture enumeration value from the chart in Table 4 for the surname. If not provided, ENS determines a surname culture using NameWorks analysis and uses it to select culture-appropriate name matching rules. If you specify a culture here, it overrides any automatically determined one.
7	Given name culture	Optional culture for the given name. See notes above.

**Table 5 - Data fields in an extended format name file –personal name line**

A line for an organization name has these fields:

	Field	Notes
1	External id	As above, some ID that is meaningful to the customer.
2	Category	The letter “O” for an organization name
3	ScriptType	Optional explicit scriptType hint, almost always left blank. Currently the only intended use for this field is to distinguish Hanzi from Kanji organization-name input by specifying CHINESESCRIPT or JAPANESESCRIPT.

<sup>8</sup> You only need to provide explicit script type in one rare case: Japanese organization names that are written entirely with Chinese characters (that is, with no Kana characters). The script type for such a name cannot automatically be recognized as Japanese. It will be assumed to be CHINESESCRIPT unless you explicitly indicate JAPANESESCRIPT.

If your data has such names, you should use extended format so that you can show explicit script type where needed. Otherwise -- and far more often -- you can use the normal format.



		Other script types are determined automatically from the name text.
4	Organization name	The organization name
5	Culture	As above, an optional culture enumeration value for the name, from the chart in Table 4. If not provided, NameWorks attempts to determine a culture from the script type of the incoming name. (Note that for organizations, NameWorks does not determine culture from the name itself, just from the script used.) If you specify a culture here, it overrides any automatically determined one.

**Table 6 - Data fields in an extended format name file –organization name line**

### 5.2.5 Use UTF-8 Encoding

If your input file contains non-Roman characters, or even Roman letters with accent marks, be sure that the file uses UTF-8 encoding. A byte order mark (BOM) is not required at the start of the file. If one is present it will be ignored.

### 5.2.6 Duplicate External Ids

Starting with version 6, ENS allows multiple names with the same external id in the same name list. This means you can have both “John Wayne” and “Marion Mitchell Morrison” as source names with external id DUKE1. One does not replace the other. The deleteName web service deletes all names having the specified external id.

## 5.3 The NameLoader Configuration File

NameLoader uses a configuration file to control numerous details of its operation.

By default, the nameLoader script in your <ensroot>/bin folder uses the <ensroot>/data/loader.config file. That file contains information inserted by the ENS installer including:

- The JDBC driver class
- The URL and schema for the database
- The path of a NameWorks configuration file to be used by the NameWorks instance that analyzes incoming names

Beyond that, the loader.config file contains several parameters that control the parsing and processing of raw names, and several that can be adjusted to optimize performance. That file is annotated with comments that explain most parameters. A few additional notes follow.

Parameter Name	Default Value	Notes
doFullName	true	If this is true, a single-field source name (with a surname but no given name) is parsed into a given name and surname on loading. If it is false, it is treated as an intended parse having a blank given name, though in this case NameWorks may additionally provide an alternate parse.
parseThreshold	65	This controls use of alternate parses on source

		names.
doCategorize	true	This controls whether single-field source names with no category (person/org) are categorized by NameWorks on loading. If this is false, such names are assumed to be personal.  As noted in section 5.2.2, if you already know the category of each name and have explicitly marked category for organization names in the input data, or if your file contains only personal names, you can set this to false. This will prevent miscategorization of unusual personal names that look like organization names, and thereby improve search results.
nameListManagerRole	manager101	Role that users must be in to manage the loaded name list using the addName and deleteName web services. See section 5.5 for more information on permissions. You can specify/override this information on the command line with the -managerRole or -mr argument.
nameListSearcherRole	searcher101	Role that users must be in to search the loaded name list using the search web services or the ENS search GUI. Again, see section 5.5. You can specify/override this information on the command line with the -searcherRole or -sr argument
extendedInputFormat	false	Set to true to use extended input format as in section 5.2.4

**Table 7 - Special loader.config parameters**

The simplest way to use nameLoader is to always use the default loader.config file, and specify the name list name, description, and roles on the nameLoader command line for each particular run.

Another approach is to make different copies of the default loader.config file for different name lists that you commonly load, and specify the name, description, and roles in those files. Then your nameLoader command line arguments will just specify the configuration file (possibly preceded by credentials).

## 5.4 Name Lists

ENS lets you define multiple name lists. For example, an airline’s ENS system might have one name list for passengers, another for employees, another for no-fly, and so on. When searching for names, a user can choose to search in all name lists<sup>9</sup> or only in specific ones.

---

<sup>9</sup> Really this means “all name lists searchable by this user”; read on for more information.

Each time you run `nameLoader`, it adds names to one name list. You specify the name list with the `-nlc` command-line argument.

- If you specify a name list that does not yet exist in your ENS data, `nameLoader` adds a new name list and populates it based on data in the csv file.
- If you specify a name list that already exists, `nameLoader` just adds or updates names in that list, based on data in the csv file.

### 5.5 Permissions

Access to name lists is controlled using ENS's role-based security. When you add a new name list to the system using `nameLoader`, you can specify two roles for that list:

- The role that a user must have in order to search the list.
- The role that a user must have in order to manage the list (adding or removing names).

As described in section 9, the `users.xml` file used by the WebSphere Liberty instances in ENS lets you assign roles to users. By default,<sup>10</sup> the roles available in ENS include “searcher101” through “searcher120” and “manager101” through “manager120”. There are also more generic groups “searchers\_all\_lists”, “managers\_all\_lists”, and the more powerful “admin”.

When you add a CUSTOMERS list, you might specify that only users with role “searcher101” are allowed to search it, and only users with role “manager110” are allowed to modify it.<sup>11</sup>

When you use ENS to search, you only deal with name lists that you are allowed to search, based on the role(s) for which your username is authorized. In the ENS search GUI, these name lists are listed in a checkbox list (see Figure 27) where you can select particular name lists to search. By default, your search includes all of the name lists that your roles allow you to search.

### 5.6 Removing a Name List

To remove a name list from your ENS system, run `nameLoader` with the `-clear` option, like this:

```
nameLoader -clear -nlc BAD_CUSTOMERS -dbu <userid> -dbp <password>
```

where `<userid>` and `<password>` are database credentials.

---

<sup>10</sup> If you need to use a wider range of roles, you can extend the configuration to support up to searcher1-searcher500 and manager1-manager500 by modifying two files. See section 9.4 for more.

<sup>11</sup> Actually, beyond the specific numbered roles, users in the generic “searchers\_all\_lists” group can search any name list. Users in the generic “managers\_all\_lists” group can search and modify any name list. Users with the “admin” role can do both.

This removes BAD\_CUSTOMERS from the database's table of name lists, removes all associated source names from its table of source names, and removes all records linking those source names to search names.<sup>12</sup>

This does not necessarily remove all search names that were associated with those source names, because those search names may be linked to other source names from other lists.

### 5.7 Clearing All Names

To clear all names and name lists from your ENS database, run nameLoader with the `-clearAll` option, like this:

```
nameLoader -clearAll -dbu <userid> -dbp <password>
```

where `<userid>` and `<password>` are database credentials.

This clears the database tables containing name lists, source names, search names, and the associations between source and search names. Effectively it clears all names and name lists from the system. This is usually much faster than clearing name lists individually.

Because this operation clears all names, NameLoader displays a confirmation prompt before proceeding with the operation:

```
This will clear ALL names and ALL namelists from the database. Are you sure you want to proceed?[n]
```

Type "Y" to proceed, or "N" to cancel. You can suppress this check by adding the line

```
quiet=true
```

in your loader.config file.

The `clearAll` operation is not allowed if the cell is active, since active searchers and dispatchers could be expecting names to be present. Before doing a `clearAll` operation, you can either stop the cell using the ENS console web application as in section 8.7, or stop all running profiles (as in section 6) and allow several minutes for the cell to time out.

### 5.8 Revision Tracking

The `nameLoader` program adds source names and search names to tables in the ENS database. The search names (or some partitioned subset of them) are then loaded into searchers' NameWorks instances when those searchers start up.

---

<sup>12</sup> Note that this does not remove all of the related search names from the search name table, because a particular search name may be linked to more multiple source names from different name lists. Orphaned search names may remain in the system but do not cause false hits in ENS results because, since they are not linked to source names, they do not appear in ENS results.

One pattern for using nameLoader is to load names before starting the system, then start things up. Searchers then find all of the names that have been loaded.

You can also run nameLoader after searchers are running. In that case, there is a mechanism for tracking revisions, to let searchers find out about added names that they need to pick up.

When you run nameLoader, it checks to see whether the cell is currently active. If so it automatically uses the revision tracking mechanism that lets any running searchers automatically pick up their new names. You don't need to do anything to make this happen.

Optionally you can override this behavior and force revision tracking to be on or off, using the revisionTracking parameter in the loader.config file:

- If this is AUTO (the default), the behavior is as described above. In an active cell, the system keeps track of revisions and added names so that currently active searchers find out about the names they need. In an inactive cell, it does not, since searchers will automatically pick up all names when they start.
- If this is FALSE, revision tracking records are not generated. This makes the nameLoader run faster, but searchers will not pick up the new names until the next time they are restarted.
- If it is TRUE, revision tracking records are generated whether the cell is active or not. This setting is seldom if ever needed.

You can also suppress revision tracking with the -noRevTrack argument to the nameLoader command line, like this:

```
nameLoader -noRevTrack -nlc "CUSTOMERS" -nld "Our customers" -in  
c:\data\customers.csv -dbu <userid> -dbp <password>
```

Even if you suppress the revision tracking mechanism, searchers will still pick up the new names the next time they are started or restarted. If your system has redundant searchers, you can stop and restart them one at a time without interrupting overall service.

### 5.9 Command-Line Arguments for the NameLoader

The general form of the nameLoader command line is:

```
nameLoader <action> <options> <configurationFileName>
```

The <action>, if specified, is one of the following:

Action	Notes
-load	Adds/updates the specified name list and its names. This is the default action.

-clear	Removes the specified name list and its names
-clearAll	Clears all name lists and their names

**Table 8 - NameLoader actions**

If no action is specified, -load is assumed.

The options can be any of the following:

Option	Example	Notes
-nlc <name> -nameListCode <name>	-nlc CUSTOMERS -nameListCode CUSTOMERS	Name of the name list to be added/updated. Required when loading or clearing.
-nld <desc> -nameListDescription <desc>	-nld "All customers"	Description for the name list.
-in <file> -inFileName <file>	-in /data/cust.csv	Input file containing names. Required when loading.
-sr <role> -searcherRole <role>	-sr searcher101	Role that a user must have to search this list. By default this may be searcher101 to searcher120.
-mr <role> -managerRole <role>	-mr manager101	Role that a user must have to add and remove names. By default this may be manager101 to manager120
-dbu <username> -dbUserId <username>	-dbu mydbuserid	User name for accessing the database. If not specified, nameLoader will prompt for this in the console.
-dbp <password> -dbPassword <password>	-dbp mydbpassword	Password for accessing the database. If not specified, nameLoader will prompt for this in the console.
-revtrack -revisionTracking -norevtrack -noRevisionTracking	-norevtrack	Overrides revision tracking settings (see section 5.8). Typically the default settings are best.
-debug -trace -verbose	-debug	Several options that provide additional diagnostic information. Use at most one of these.
-bt -batchTracing	-batchTracing	Provides additional diagnostic information.
-usage	-usage	Displays usage information

**Table 9 - NameLoader options**

### 5.10 NameLoader Performance

NameLoader performance depends directly on database performance. Two things are very important for proper NameLoader performance:

- Adjusting database settings after ENS installation (or after first creating database tables with ensclu) as described in section 3.22
- Updating statistics as in section 5.11 after loading some names, and probably repeating at several larger sizes.

These two steps can affect NameLoader performance dramatically (by a factor of 40 or more).

### 5.11 Updating Database Statistics for Better Performance

Important: Once you use NameLoader to load some names into ENS tables in DB2, you should use its RUNSTATS command to update statistics about the tables that hold names. This lets DB2 optimize the way it accesses those tables. This can make a dramatic difference in ENS performance.

You would do this after loading some names, and can often even do it while NameLoader is running. You may find that it pays to run statistics after adding 200,000 names, and again at some larger sizes (say after adding a million names and again after 5 million, etc).

This is normally done by a DBA. If you are on your own (say in a demo system), typical DB2 commands for doing this, assuming schema name "ENS", would be:

```
runstats on table ens.ens_search_name with distribution and detailed indexes all
runstats on table ens.ens_source_name with distribution and detailed indexes all
runstats on table ens.ens_search_source_name with distribution and detailed indexes all
commit
flush package cache dynamic
```

Oracle has a similar command. More information on it is available at:

[http://docs.oracle.com/cd/B19306\\_01/appdev.102/b14258/d\\_stats.htm#i103646](http://docs.oracle.com/cd/B19306_01/appdev.102/b14258/d_stats.htm#i103646) .

Note that some systems are configured to automatically run statistics at intervals to optimize their operation. If your system does this at a time when your ENS tables happen to be empty of names, it may effectively undo the effect of your earlier manual statistics run. In that case, you may need to rerun statistics again with names loaded, as above.

### 5.12 NameLoader configuration file tuning (advanced topic)

While database settings and statistics are the most important consideration in NameLoader performance, some adjustments to the loader configuration file can also be of benefit. This is an advanced topic most relevant to large installations. The default settings are often fine.

NameLoader reads source names from a file, queues them for analysis by multiple analyzer threads, and then queues the combination of source and analyzed search names for writing to the database. When writing to the database, it handles names in batches for efficiency.

You can specify the size of the queues, the number of threads used for name analysis and for writing to the database, and the batch size for database operations.

The best values for your environment depend on the machine's number of logical CPUs and speed, the database performance and connectivity, the number of instances of NameLoader being run concurrently (if any), and the amount of duplication of source names having the same id in the input.

### 5.12.1 Queue sizes

```
threadQueueSize=500
```

This is the maximum queue size both for the analysis queue and the writer queue. A larger size here helps keep the writer queue populated so that writer threads handle efficient batches, and not just a few names at a time.

Increase this number if you have a large database batch size and multiple writer threads.

### 5.12.2 Threads for analysis and writing

```
numThreadsAnalyzer=6  
numThreadsWriter=2
```

Typically the number of threads assigned to analyzer and writer use should add up to the total number of logical CPUs that you have available, with a large majority going to the analyzer.

On a machine with 4 logical CPUs and good DB performance and connectivity, 3 analyzer and 1 writer thread are probably best.

On a machine with 8 logical CPUs and good database performance and connectivity, you might choose to have 6 analyzer threads and 2 writer threads (the default), or even 7 analyzer and 1 writer thread. If database performance or connectivity are not as good, you may need to add writer threads.

On a machine with 16 logical CPUs available, you might start with 13 analyzer and 3 writer threads.

You can monitor the queue sizes reported as NameLoader runs, and use the displayed information to fine tune your settings as discussed in section 5.12.4 below.

### 5.12.3 Database batch size

```
dbBatchSize=64
```

NameLoader writes names to the database in batches for efficiency. The dbBatchSize parameter sets the maximum size for a batch.

If you only have one or two writer threads, you can set this high (say 240 or more).

If you have multiple writer threads (or multiple NameLoader instances) and are encountering numerous collisions/retries between threads because the input data contains many duplicate names, smaller batch sizes and/or fewer writer threads will help.



### 5.12.4 Fine tuning

NameLoader displays a trace line after every ten thousand names are processed. You can watch the snapshot write queue size displayed in these trace lines and make adjustments as follows:

- If the write queue size stays near the threadQueueSize configured here, try adding a writer thread and removing an analyzer one. You shouldn't normally have many writer threads though. Also make sure that you have adjusted database settings as in section 3.22, and that you have run statistics with a significant number of names loaded as in section 5.11.
- If the write queue size is often below 100 and you have multiple writer threads, you may be able to remove a writer thread and add an analyzer one.
- If the write queue size fluctuates widely, make sure the threadQueueSize parameter that controls its maximum is set high enough that you don't frequently have inefficient small batches due to the queue not having many names to be processed.

If the timestamps on the trace lines show significant slowing as the NameLoader run progresses, this usually means you need to rerun database statistics as in section 5.11. If things suddenly slow down as sizes get large, you may also want to make sure that the database machine has adequate memory.

## 6 Starting and Stopping ENS Servers

You start an ENS server on a particular host by going to an ENS bin folder on that host, and running a start script.

The bin folder is <ensroot>/bin, where <ensroot> is:

- In a local installation, the folder that you specified when running the ENS installer.
- In a distributed installation, the shared location in the network that you specified when running the ENS installer.

In a local Windows installation, the start script for profile P1 would be like this:

```
c:\projects\ensguide\bin>start-P1.bat
```

The name of the start script is based on the profile name. If your profile is named “P1”, the start script is named `start-P1.bat` on a Windows system or `start-P1.sh` on UNIX.

In either case, when you `cd` to the appropriate bin directory and run the start script, the WebSphere Liberty instance for that server starts up, and initializes its ENS web application.

Having done that, you can browse to the console GUI as in section 8.2. In a new installation or after starting new servers (based on new hosts or profiles), you’ll use the console to configure the cell (section 8.3) and then start the cell (8.5). Later, you’ll typically just start the cell (8.5) after starting its servers.

To stop an ENS server, you `cd` to the same bin directory and run a stop script. In our example, the script would be named `stop-P1.bat` on Windows or `stop-P1.sh` on UNIX.

## 7 Adding Hosts in a Distributed ENS Installation

### 7.1 In a nutshell

Adding more hosts to a distributed ENS installation is trivial. No installation step is required. All you have to do is:

- Make sure that the new hosts meet the prerequisites described in section 2.4 and 2.5, and
- Start the wanted profile(s) on the new hosts as in section 6. The new servers join the cell automatically, and will appear in the console when it is refreshed.

Each profile that you run on a new host (that is, each host+profile combination) is a new ENS server. Once you start it up, you probably want to give it a searcher holding some portion of your name collection, so it can do useful work. There are two ways to do this.

**Reconfigure the cell:** After you add hosts and start servers, you may choose to reconfigure the cell as in section 8.3. This lets you completely redo the partitioning of names, determining how they are distributed among searchers. This wholesale partitioning is easy but it can only be done in a stopped cell, since searchers need to reload their names afterward.

**Mirror an existing searcher:** Alternatively, if you just want some new server to replace or augment an old one, you may choose to mirror an existing searcher to the new server as in section 8.9. The searcher on the new server will have the same partition assignment as the existing one.

You might mirror an existing searcher to provide additional redundancy and throughput, or to make it possible for the old server to be shut down, e.g. for maintenance. Mirroring a searcher this way does not require shutting down the cell, since it does not affect the distribution of names among other searchers.

### 7.2 The first host in a distributed installation

In a distributed ENS installation, the same configuration information and scripts are used for ENS servers on multiple host machines. On each such host, ENS gets configuration information from the same shared file locations, accessible via the same path on each machine. The files there contain the Liberty installation and profile, as well as ENS configuration files and scripts.

You only need to run the installer once, on the first host that you're using. That sets up the shared files. Having done this on the first host, you can start a server on that host by:

1. Opening a shell on that host and cd'ing to the <ensroot>/<profile name>/bin folder.
2. Running the start script for that profile

The result is a running ENS server. If you have multiple profiles, you CD to the other <profile name> folders and run their start scripts too.

### 7.3 Additional hosts: Meeting Prerequisites

No further installation is needed to run the same profile on other hosts. You don't have to copy files or run an installer. All you need to do is to make sure that:

1. Each new host has the same architecture and OS as the original host,
2. They are configured with the same shared and local file paths as described in section 2.4, and
3. They have the same host alias defined (pointing to their own host IP addresses) as in section 2.5.

### 7.4 Additional hosts: Starting ENS

If a new host meets these prerequisites for some distributed ENS cell, then starting ENS on that new host is the same as starting it on the original host. You start it by:

1. Opening a shell on the new host and cd'ing to the <ensroot>/<profile name>/bin folder.
2. Running the start script for that profile

The result is another running ENS server in the cell. If you have multiple profiles, you can then cd to the other <profile name> folders and run their start scripts too.

### 7.5 Additional hosts: Configuring

Once you add additional servers to a cell, you'll need to decide what they're going to do. In other words, you'll need to decide which components are to run on the new servers and how names are to be partitioned among the searchers that now exist in the cell.

One option is to completely reconfigure the cell as in section 8.3. This lets you redo the partitioning of names, determining how they are distributed among searchers. For example, after adding a fifth server to a cell, you could go from a four-way partitioning to a five-way partitioning of the names. Such a wholesale change is easy

to do, but it can only be done in a stopped cell<sup>13</sup>, since it changes what names are to be held in each searcher. When the cell is restarted, the searchers automatically load names based on their new partition assignments.

Alternatively, if you are just adding a new server to replace or augment an old one and you want to keep the system running, you may just choose to mirror some existing searcher to the new server as in section 8.9. The new server's searcher will then have the same partition assignment as the existing one. You might do this to provide additional redundancy and throughput, or to make it possible for the old server to be shut down e.g. for maintenance.

You can even add a new server to a cell without assigning it any components initially, just to have it available as a spare. Then when you do want to swap out some existing server, you can mirror components from the existing server to the new one.

---

<sup>13</sup> A “stopped cell” here means one where you have run scripts to start the servers (that is, the Liberty instances) as in section 6, but you have not yet started the cell using the ENS console as in section 8.5. (Or you have stopped the cell using the console as in section 8.7.)

## 8 Managing Components Using the Console GUI

Once you have started the ENS servers as in section 6, you can use the browser-based ENS console GUI to manage the components.

### 8.1 In a Nutshell

Briefly, here's what you need to do to configure and start the cell:

1. Be sure you've started your ENS server(s) as in section 6.
2. In your browser, go to the console GUI at <http://<host>:<port>/ens/console>, using the host and HTTP port from sections 3.8 and 3.18.
3. Log in using the ENS credentials you specified in section 3.10.
4. Select the "Cell Configuration" tab and:
  - a. Select a redundancy strategy. The simplest choice is "none".
  - b. Check checkboxes indicating where you want component to run. One simple arrangement is to check one box in the "Dispatcher" column and all the boxes in the "Searcher" column. Don't worry about dashboards.
  - c. Click the "Apply Changes" button.
5. Select the "Server Status" tab, and:
  - a. Click the "Start Cell" button at the top.
  - b. Watch the status of the components in the console's status grid. When the dispatcher and searcher components you specified have "active" status rather than "initializing", you are ready to search.

**Figure 5 - Configuring and starting the cell, in a nutshell**

The following sections discuss this in more detail.

### 8.2 Browsing to the Console GUI

We'll assume that you've done the installation and setup steps in sections 2-4, and started your ENS servers as in section 6. Now, in your browser, go to the console GUI at <http://<host>:<port>/ens/console/> . It's simplest to use the host and port from sections 3.8 and 3.18.<sup>14</sup>

Log in using the credentials you specified in section 3.10 (or of some other user with admin privileges). You'll see the ENS console GUI pictured in the next two sections.

---

<sup>14</sup> In fact, the console can run on any server in your cell, but only one instance of the console will run at a time. If you browse to the console URL on a server in your cell when no console is yet running in the cell, the system will start one up on the server you chose. If a console is already running on some other server in the cell, the system will redirect your browser to it.

### 8.3 The Console GUI: Cell Configuration Tab

The ENS console GUI has two main tabs:

- Server status – shows status of servers and the components running on them
- Cell configuration – designates which components are to run on each server

Most of the time you'll use the server status tab, but when first configuring you will want to select the second tab, Cell Configuration. It looks like this:

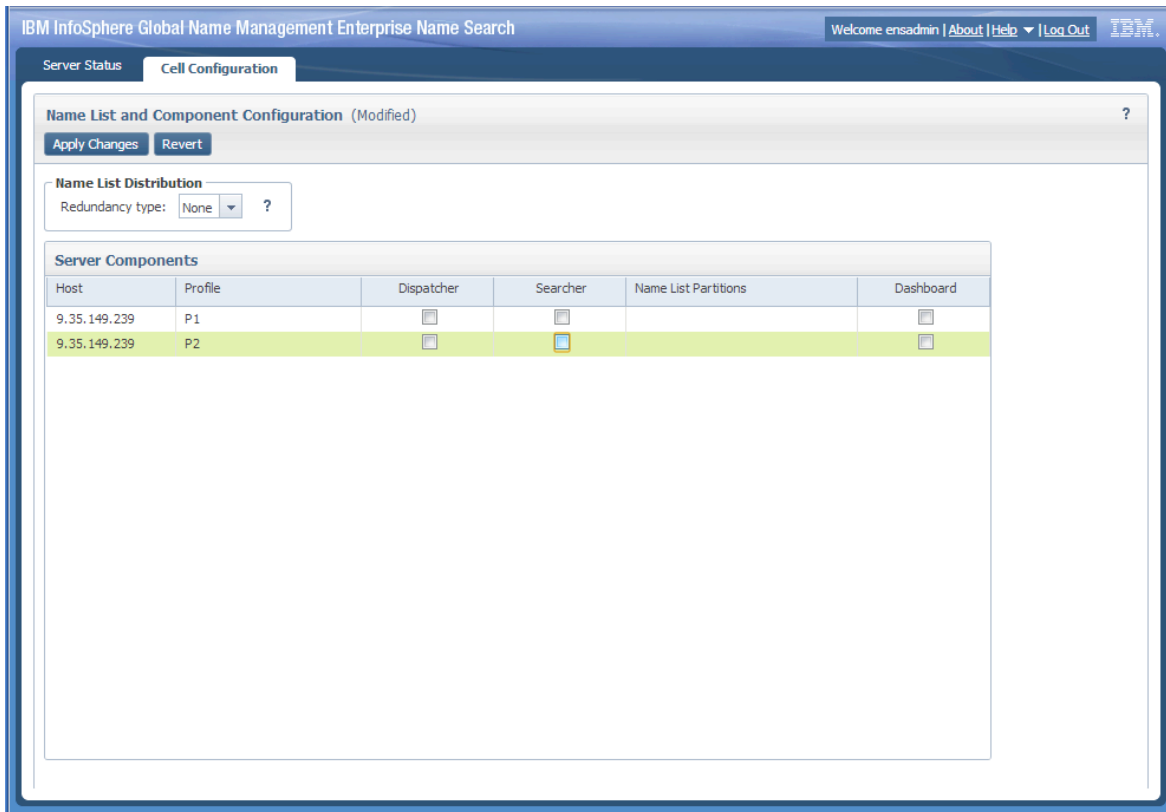


Figure 6 - ENS console - cell configuration tab

Cell configuration means telling your ENS system two things:

- Which components should run on each server, and
- How names should be distributed across searchers.

The main thing on this tab is a grid with one row for each server in the cell. Each row has checkboxes to designate which components should normally run on that server.

To configure the cell, you check checkboxes for some set of dispatchers and searchers, optionally adjust the name list distribution details at the top, and click Apply Changes.

You can only do cell configuration while the cell is stopped as in section 8.7. This is because most changes to cell configuration involve substantial changes in which names need to be loaded on which searcher. Going from a four-way partitioning to a

five-way partitioning would mean that every searcher needed to load a different set of names from what it had before.

### 8.3.1 Choosing which servers will have searchers

The number of searchers that you choose to have, together with the redundancy strategy that you choose, will determine how many partitions the search names are divided into.

Typically you'll check the searcher checkboxes for every row, so that every server has a searcher. This spreads the names out so as to take advantage of all of your servers, and loads fewer names on each searcher for better performance.

The main consideration in distributing names among searchers is to avoid exceeding physical memory when the searcher loads its names into NameWorks, since the resulting swapping during searching would significantly impact performance.

If you have more servers than you need, you may want to hold one or two in reserve as spares, with no searcher initially assigned to them. That way, later when the system is running, if you have some need to restart some searcher (or its server or host), or to take it offline, you can first mirror its names to the spare as in section 8.9.

### 8.3.2 Choosing which servers will have dispatchers

In a small system, you might choose to have only one dispatcher. Or you could have more than one, for added capacity and redundancy. You can always start up another dispatcher later, in a running system; you're not restricted to the set that you choose at cell configuration time.

Note that each dispatcher has a separate URL. Client applications or people using the ENS search GUI will need to be given the URL for the dispatcher that they are using. There is no automatic redirection from one dispatcher to another within ENS. Something like this might be done external to ENS using a reverse proxy, but this is not supported as part of ENS.

### 8.3.3 Choosing a distribution strategy

The "Name List Distribution" section above the main grid lets you choose how names will be distributed among the searchers you designate. There are three main options:

#### 8.3.3.1 No redundancy

With this option, if you select checkboxes for six searchers, the search names will be divided into six partitions, one for each searcher. Each partition is fully loaded onto one searcher. Each search name is loaded into only one searcher, so there is no redundancy. With six searchers and no redundancy, there would be six partitions which the system might assign as follows:



Searcher A: Partition 0  
Searcher B: Partition 1  
Searcher C: Partition 2  
Searcher D: Partition 3  
Searcher E: Partition 4  
Searcher F: Partition 5

In this case, the dispatcher has to delegate all search requests to all searchers.

### **8.3.3.2 Mirrored**

With this option, if you select checkboxes for six searchers and a mirror count of two, the system divides the names into three partitions, so that each partition (with all its names) is fully loaded onto two searchers. The resulting redundancy helps reliability, since if one of those searchers (or its server or host) goes down, the other can act as a backup.

Searcher A: Partition 0  
Searcher B: Partition 1  
Searcher C: Partition 2  
Searcher D: Partition 0  
Searcher E: Partition 1  
Searcher F: Partition 2

The dispatcher knows which partitions are on which searchers, and it chooses an appropriate set of searchers to ensure coverage of all partitions. It does not issue unnecessary redundant search requests. However, in the above example, if it has chosen searcher B as part of some search and does not get a response, it automatically fails over to searcher E.

### **8.3.3.3 Overlapping**

With this option, the system automatically chooses a partitioning and an assignment of partitions to searchers that ensures some redundancy.

Overlapping is best illustrated with a simple example. If you choose overlapping in a system with six searchers, the system will make six partitions and might assign them as follows:

Searcher A: Partitions 0 and 1  
Searcher B: Partitions 1 and 2  
Searcher C: Partitions 2 and 3  
Searcher D: Partitions 3 and 4  
Searcher E: Partitions 4 and 5  
Searcher F: Partitions 5 and 0

In this scheme, every name is loaded into more than one searcher. As before, the dispatcher knows the arrangement and chooses an appropriate set of searchers. Again, it fails over automatically from one searcher to another if necessary.

Mirroring is usually a more effective redundancy strategy than overlapping.

### 8.3.4 Applying your configuration

Once you have checked one or more dispatchers and searchers and chosen a redundancy type, click the “Apply Changes” button to make the change. The console will update the “Name List Partitions” column based on the searcher count and redundancy details you chose.

The next step is to start the cell, in the console’s “Server Status” tab.

### 8.4 The Console GUI: Server Status Tab

The Server Status tab has a main grid showing all the servers in the cell, one per row. Figure 7 shows a cell with two active servers.

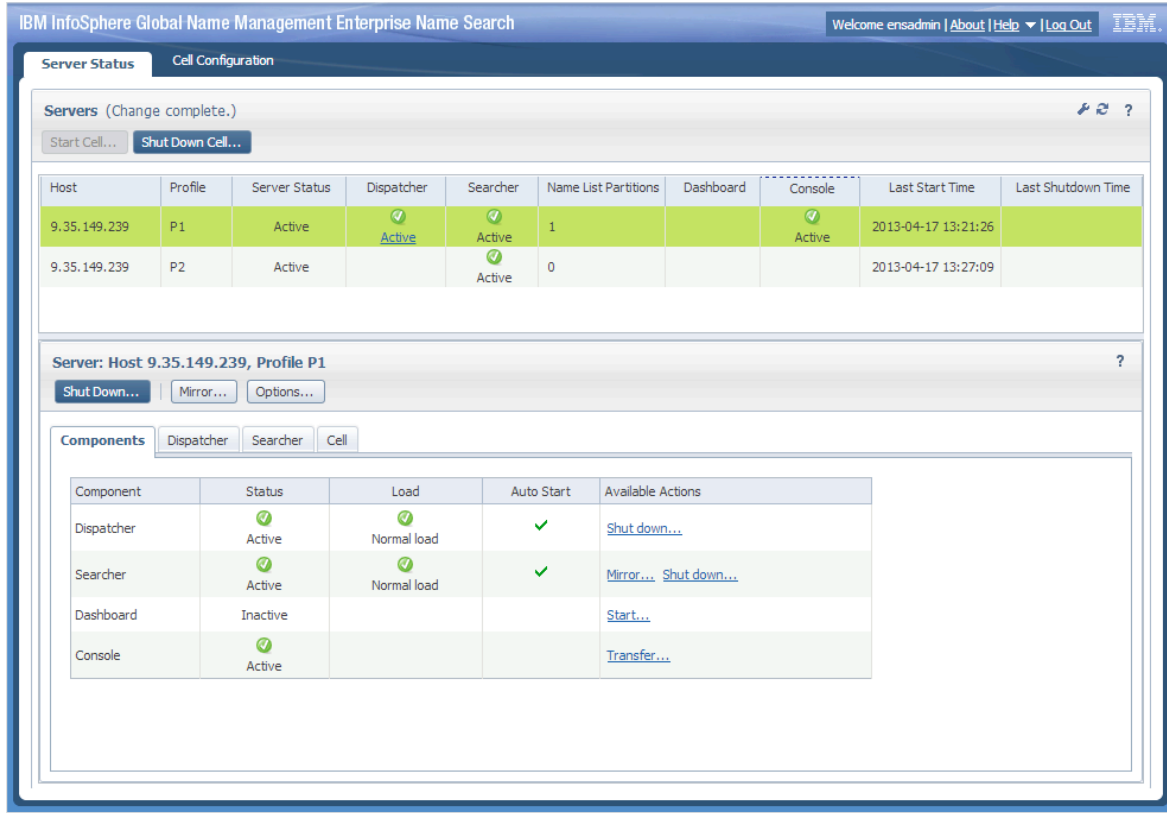


Figure 7 - ENS Console - server status tab

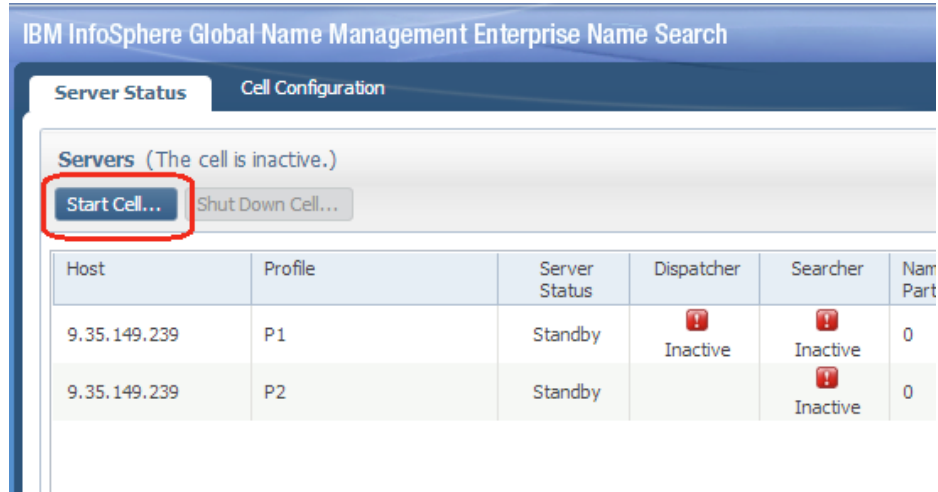
Columns here show the host IP and profile name, the server status, the status of any components (dispatcher, searcher, dashboard and console), the name list partitions assigned to the searcher if any, and the server’s last start and/or shutdown time.

When you select a row in the main grid, the bottom half of the screen shows details on the selected server. Its main tab displays component status, with buttons and links for making changes. Other tabs at the bottom show details on the dispatcher, searcher, and cell.

If no main grid is visible on this tab and only a panel saying “Select a server in the servers table at the top panel to see server details here” is seen, the browser window may need to be made taller.

### 8.5 Starting the cell

Once you have configured the cell as in section 8.3 (and applied changes as in 8.3.4), you start the cell by clicking the Start Cell button at the top of the Server Status tab.



**Figure 8 - The Start Cell button**

This starts dispatchers and searchers on any servers where they have been designated for automatic start; normally this means the same set of components on servers that you specified in cell configuration.

Typically, dispatchers start very quickly but searchers take some time to start up, because each searcher contains an instance of NameWorks that needs to initialize and load its subset of your system’s names (based on the partitions assigned to that searcher) before it is ready for use.

The search names to be loaded come from the database (as earlier populated using the nameLoader or the addName service). The time required for starting depends on the number of names to be loaded, as well as the speed of the machine, network, and database.

While components are still initializing, the console GUI refreshes itself every 10 seconds. You see a status of “Initializing” with a spinning icon for any component that is still initializing, and the status summary at the top shows that some needed components are not yet active.

### 8.6 When the cell is running

Once the cell has completed startup, you should see “Active” status for all the components that you designated in cell configuration. The status summary at the top should say that the cell is active.

Normally the ENS console GUI shows a snapshot of the system. It does not change until you refresh it, either by refreshing your browser or by clicking the refresh icon at the right-side above the status grid.

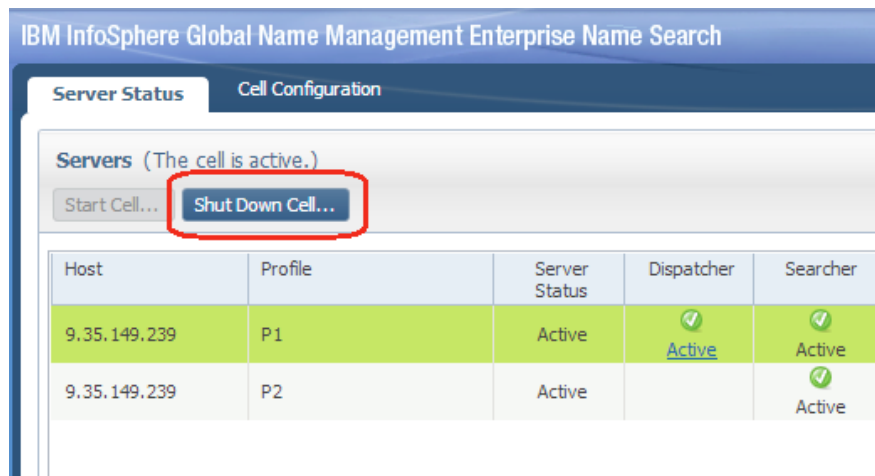
An exception is when you are starting or stopping the cell or some of its components. When the console GUI detects that any component is in the “Initializing” or

“Terminating” state, it automatically refreshes itself at 10-second intervals. Once components finish initializing or terminating, this frequent refreshing stops.

You can choose to have the console continue to refresh itself automatically at intervals by clicking the gear icon (near the refresh icon), choosing automatic refresh, and choosing some time interval.

### 8.7 Stopping the cell

If you want to stop the cell, you can just click the “Shut Down Cell” button on the left side above the main status grid. This will shut down all components on all servers in the cell.



**Figure 9 - The Shut Down Cell button**

Note that this does not stop the servers themselves. Internally, an ENS server is a WebSphere Liberty instance with some Liberty profile running the ENS web application. That continues to run unless you stop the server using the stop server scripts described in section 6.

### 8.8 Starting and stopping components

In a running cell, you can start and stop components using the buttons and links in the bottom half of the console’s Server Status tab.

The “Shut Down” button stops all of the components on the selected server. The “Start” button starts any components on that server that have been designated for auto start. Only one of these buttons is displayed at a time, depending on what components are active.

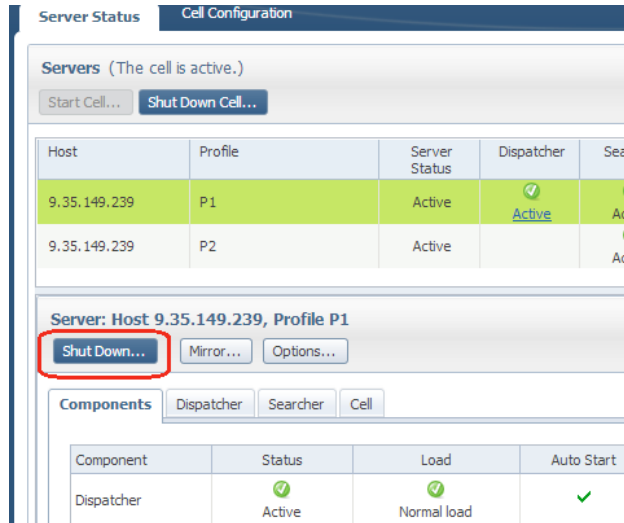


Figure 10 - Shut down button for a server

### 8.8.1 The Auto Start Flag for Components

Cell configuration has two effects: it determines a partitioning of the search names and assigns partitions to selected searchers, and it flags selected components to be automatically started when the cell is started. That flag on a component is called its "Auto Start" flag. The detail grid on the components tab has a column showing these flags with check marks.

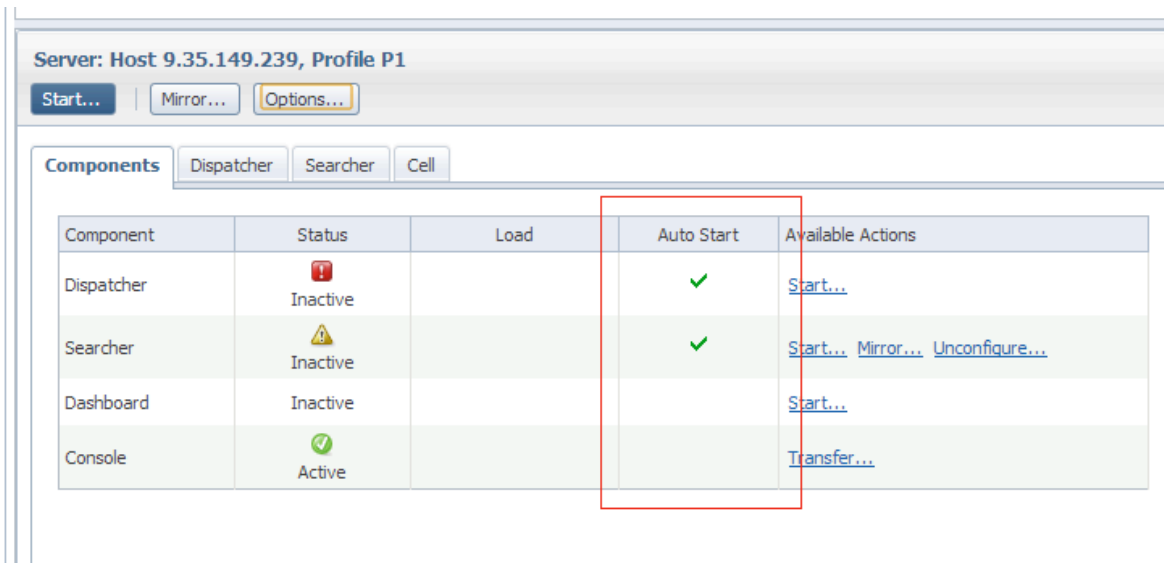
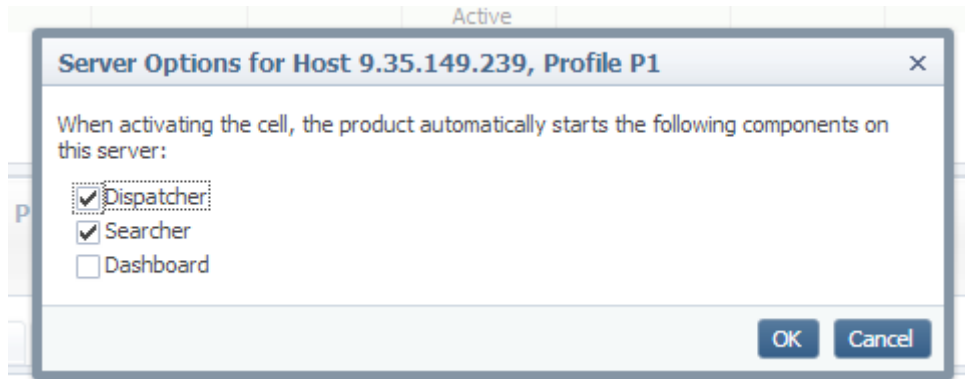


Figure 11 - Auto start flags

You can change the auto start flags for a server's components with the "Options" button in the lower panel of the console. This displays a dialog indicating which components automatically start on cell activation:

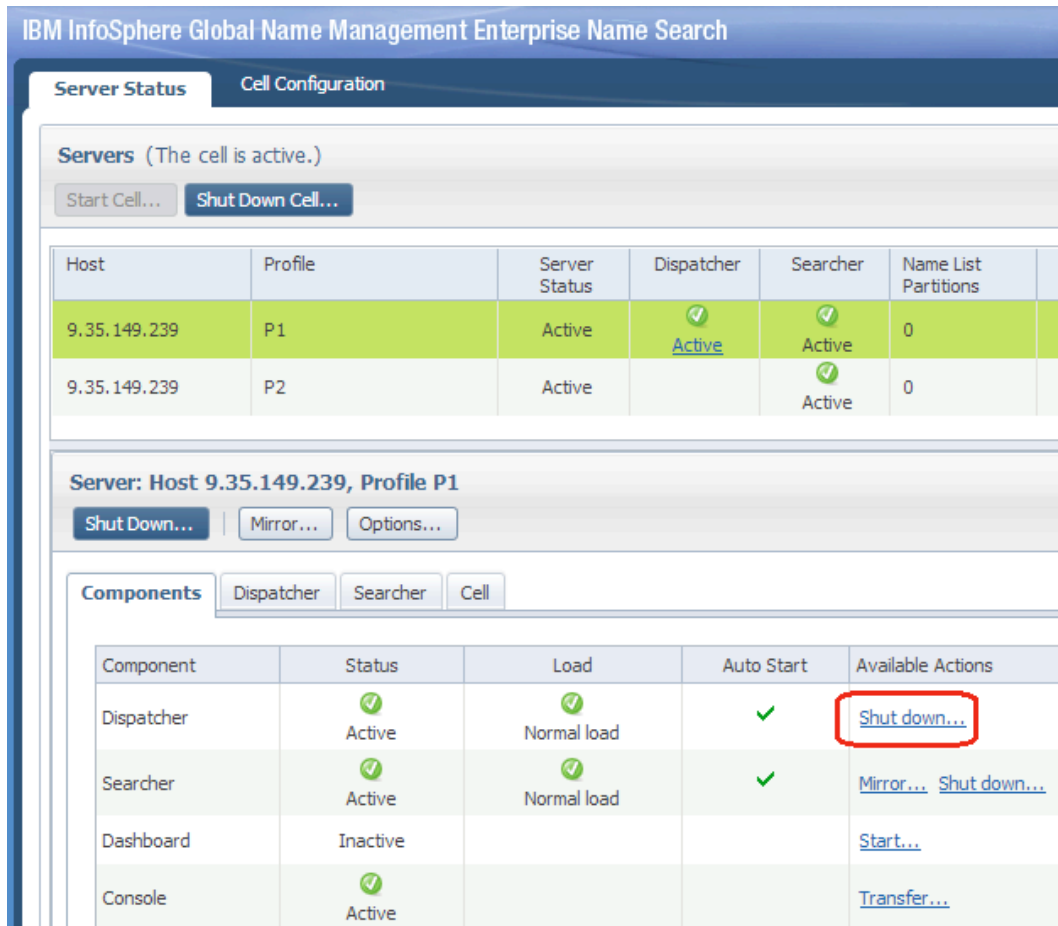


**Figure 12 - Server Options dialog**

Note that you cannot set Auto Start to true for a searcher that was not assigned partitions during cell configuration. You can't start a searcher with no names assigned. To add another searcher that wasn't included in your cell configuration, you'll either need to redo cell configuration to assign names to that searcher, or copy another searcher's partition assignment in a mirror operation as in section 8.9.

### 8.8.2 Stopping a Component

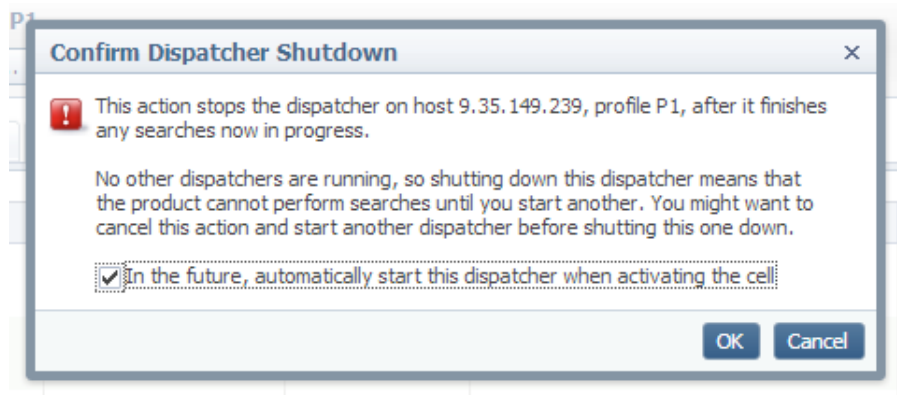
To stop one component – say a dispatcher – you select its server in the console's top grid, and then click the "Shut down" link for that component in the detail grid at the bottom.



**Figure 13 - Shut down link for a component**

Clicking this link displays a confirmation dialog. The message in that dialog depends on the importance of the component selected.

In Figure 13, we're choosing to shut down the only active dispatcher in the cell. The dialog tells us that this would make the system unable to perform searches:



**Figure 14 - Confirmation dialog when shutting down a needed component**



You'd probably want to start a different dispatcher – and make sure that search clients were using its URL instead of this one's – before shutting this dispatcher down.

In the same example from Figure 13, if you click the Shut Down link for the searcher, you get a different message. This is because that is one of two redundant searchers with the same assigned partition. Either one can be shut down without stopping the system's ability to search. The confirmation dialog here says:

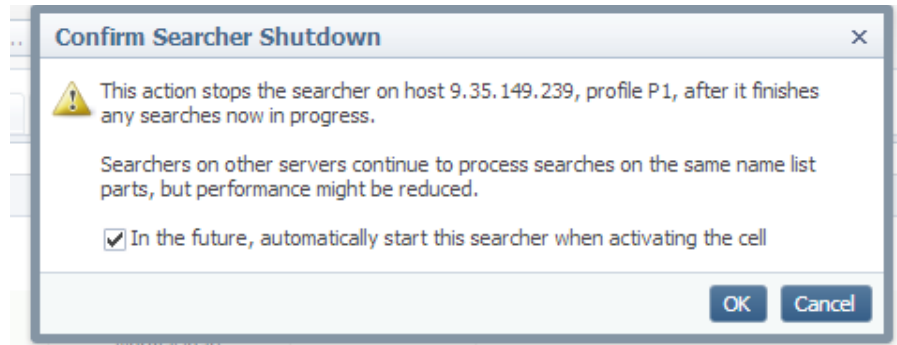


Figure 15 - Confirmation dialog when shutting down a redundant component

### 8.8.3 Starting a Component

Starting an inactive component is similar: you click the Start link in the detail grid. Now the confirmation dialog looks like this:

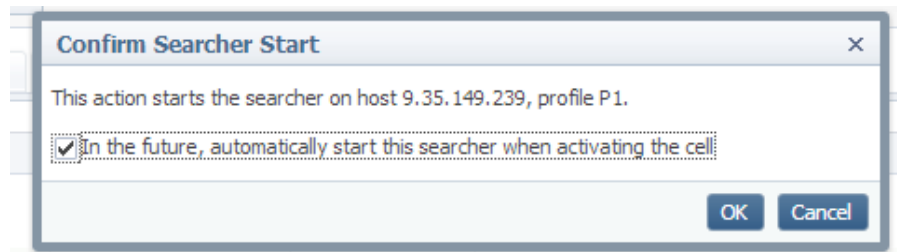


Figure 16 - Confirmation dialog when starting a component

The checkbox in these dialogs lets you adjust the Auto Start dialog for the component you are starting or stopping. That way you can say “Stop this component, and don't start it automatically in the future” or “Stop this component, but just for now. Continue to start it automatically in the future”.

## 8.9 Mirroring a Searcher

In an active cell, you can't add a new searcher in a way that changes the distribution of names among existing searchers. For that, you need to stop the cell and use cell configuration, so that the searchers load their new names when you restart the cell.

However, you still can add a new searcher in an active cell in a way that doesn't change the existing distribution of names. You do this by assigning the new searcher

the same name list partition(s) as some existing searcher. This is called mirroring the existing searcher to the new one.

You might do this to add a new searcher in order to share the load with a busy searcher, or to provide added redundancy for fault tolerance.<sup>15</sup> Or you might do it to configure a spare searcher as a replacement for one that you are planning to restart, or whose server or host you need to shut down for some reason.

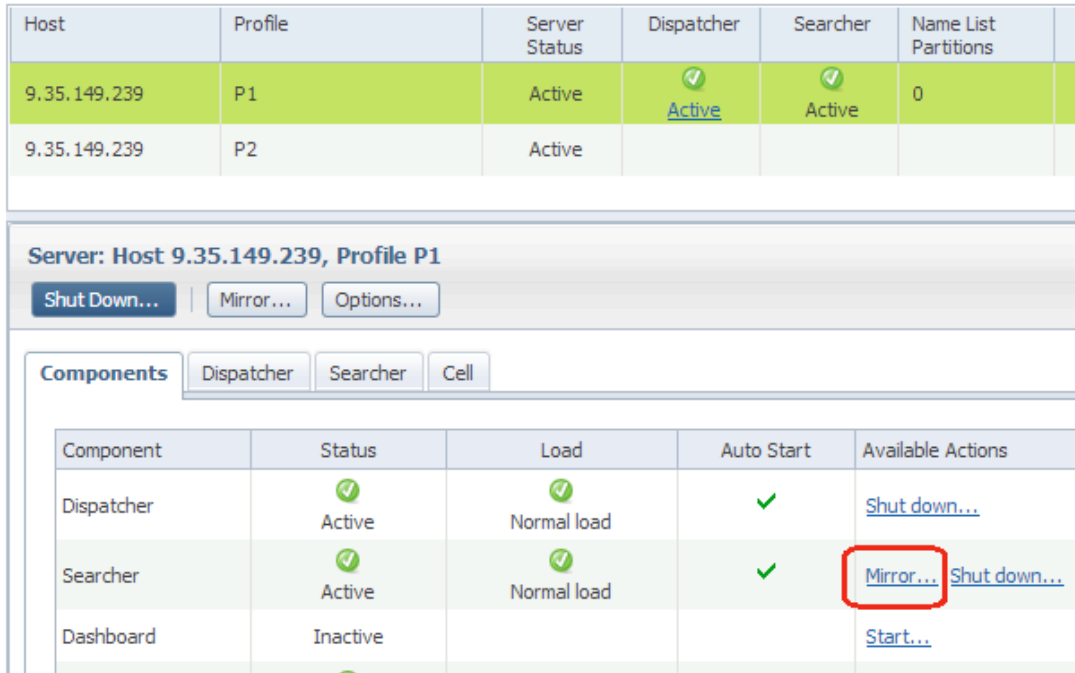
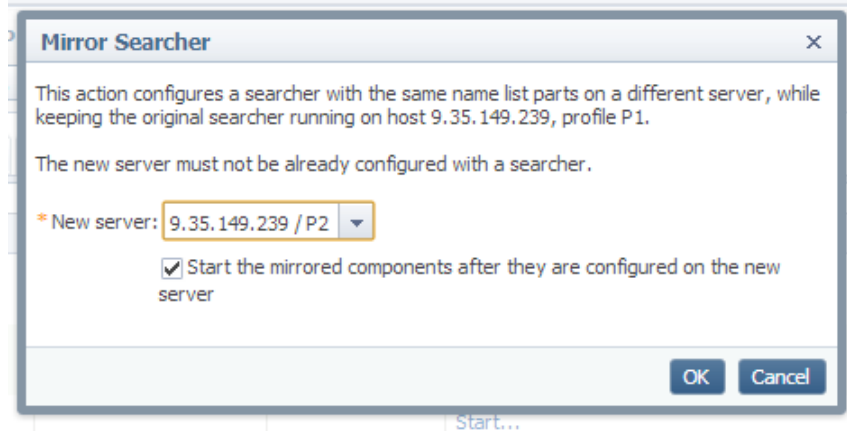


Figure 17 - The "Mirror" link for an existing searcher

In this example, we mirror an existing searcher on server A (9.35.149.239/P1) to a new searcher on server B (9.35.149.239/P2), so that searcher B has the same name lists partitions as searcher A.

1. Select server A in the main grid in the console, and click the "Mirror" link on the Searcher row in its components detail grid.
2. In the confirmation dialog, select the host and profile combination for server B. Note that this server must not already have a searcher configured.

<sup>15</sup> Mirroring servers this way rather than at cell configuration time lets you choose which specific server will hold the same partitions as some other server. In particular, it lets you make sure that the redundant copies are located on different physical machines whenever possible. If your goal is high availability and/or fault tolerance, this is a good idea.



**Figure 18 - Confirmation dialog for mirroring a searcher**

- The result is that server B now has a searcher configured with the same partition(s) as the one on server A, as seen below. Since we left the checkbox checked in this confirmation dialog, the searcher is started right away.

Servers (The searcher is initializing...)

Start Cell... Shut Down Cell...

Host	Profile	Server Status	Dispatcher	Searcher	Name List Partitions
9.35.149.239	P1	Active	Active	Active	0
9.35.149.239	P2	Active		Initializing	0

**Figure 19 - After mirroring a searcher**

- Once the searcher finishes initializing, it is displayed as “Active”. At that point we could shut down the searcher in server A.

### 8.10 Mirroring a Server

Mirroring a server means mirroring server A’s searcher to server B as in the previous section, but also setting auto start flags for server B’s dispatcher (and dashboard if applicable) to match those on server A.

The process is almost identical to mirroring the searcher, but instead of clicking the “Mirror” link in the searcher row of the detail grid, you click the “Mirror” button in the header of the detail section, as in Figure 20:

## Setting Up and Managing ENS

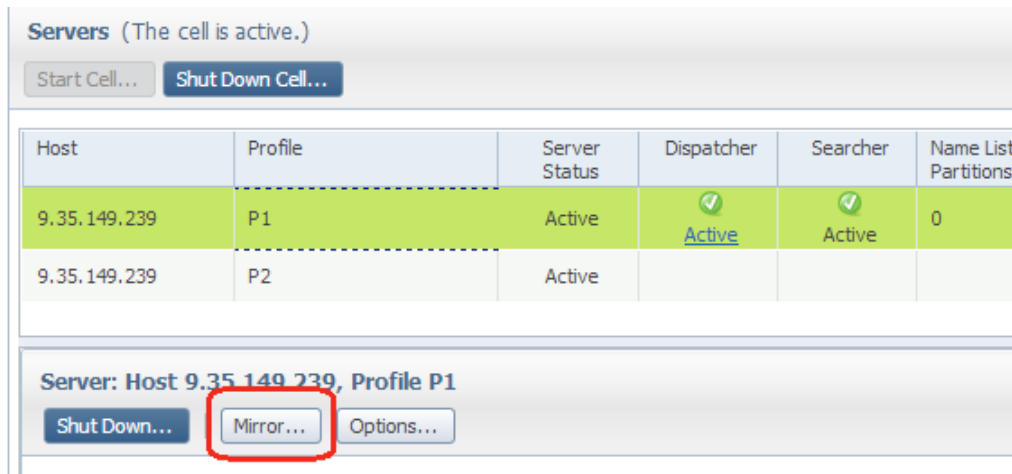


Figure 20 - Button for mirroring a server

The dialog displayed is only slightly different:

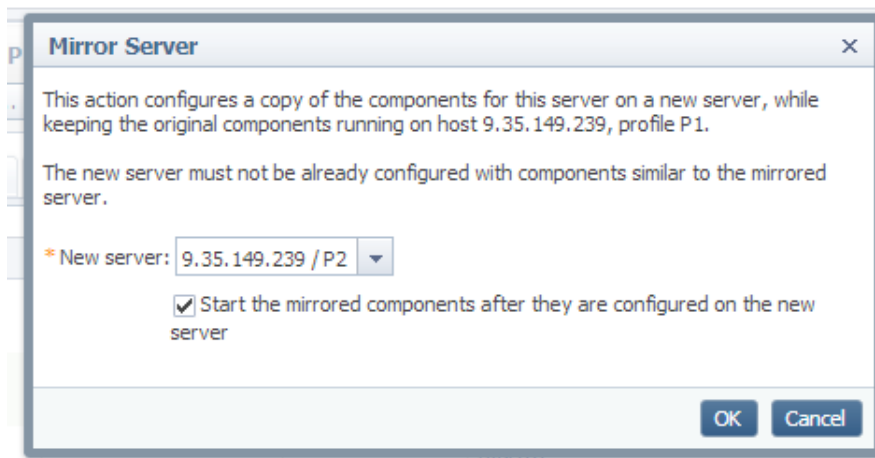


Figure 21 - Confirmation dialog for mirroring a server

The end result is that, along with a new searcher, server B also has a new dispatcher:

Host	Profile	Server Status	Dispatcher	Searcher	Name List Partitions
9.35.149.239	P1	Active	Active	Active	0
9.35.149.239	P2	Active	Active	Initializing	0

Figure 22 - After mirroring a server

### 8.11 Unconfiguring a Searcher

After mirroring the searcher from server A to server B, you may want to unconfigure the one on server A so that it can be a spare for future mirroring. Unconfiguring a searcher takes away its assigned name list partitions and clears its auto start flag.

Note that once you do this, you won't be able to start the searcher on server A until you either assign it new partitions using cell configuration, or mirror some other searcher to server A.

To unconfigure the searcher on server A:

1. Select server A in the servers grid.
2. Make sure its searcher is not active. If necessary, shut it down as in 8.8.2.
3. Make sure that this is not a needed searcher. For example, if searcher A is the only one in the cell handling partition 2, then unconfiguring it will leave the system unable to search until you stop the cell and run cell configuration again. (In fact, the console checks for this, but you should too.)
4. When the searcher finishes shutdown and becomes inactive, click the "Unconfigure" link on the searcher row in the components detail grid:

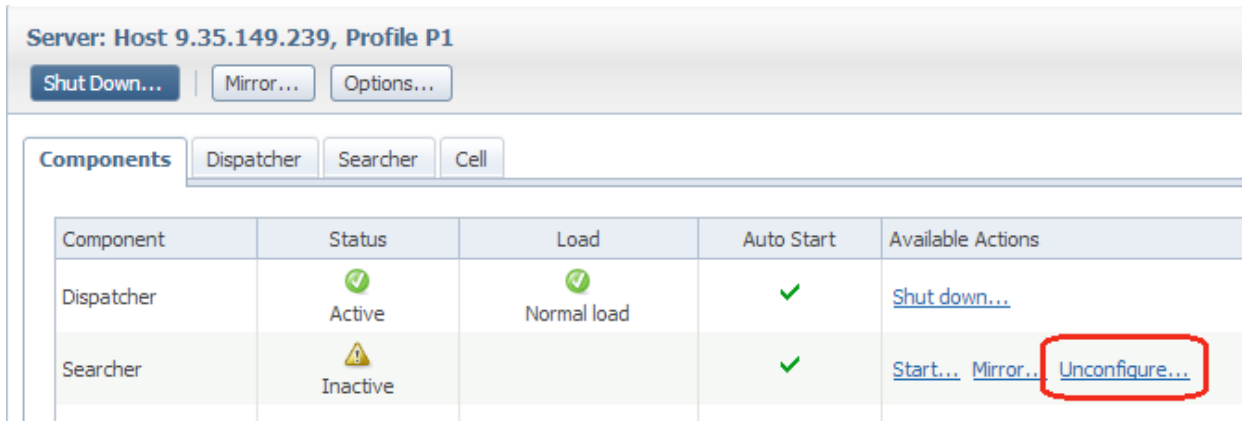


Figure 23 - "Unconfigure" link for a searcher

5. In the confirmation dialog, click OK.

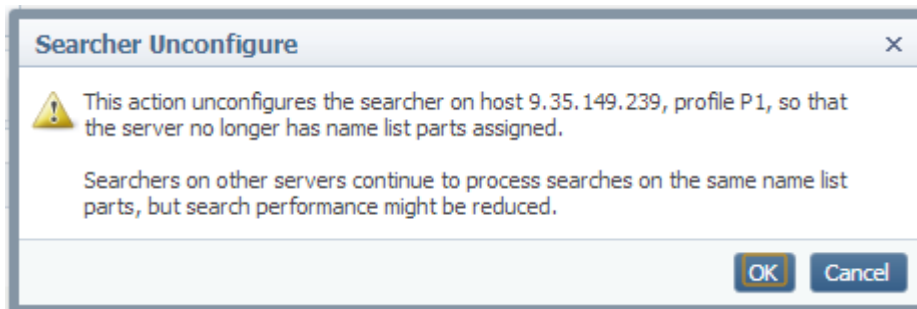


Figure 24 - Confirmation dialog for unconfiguring a searcher

6. The result is that server A no longer has a searcher configured.

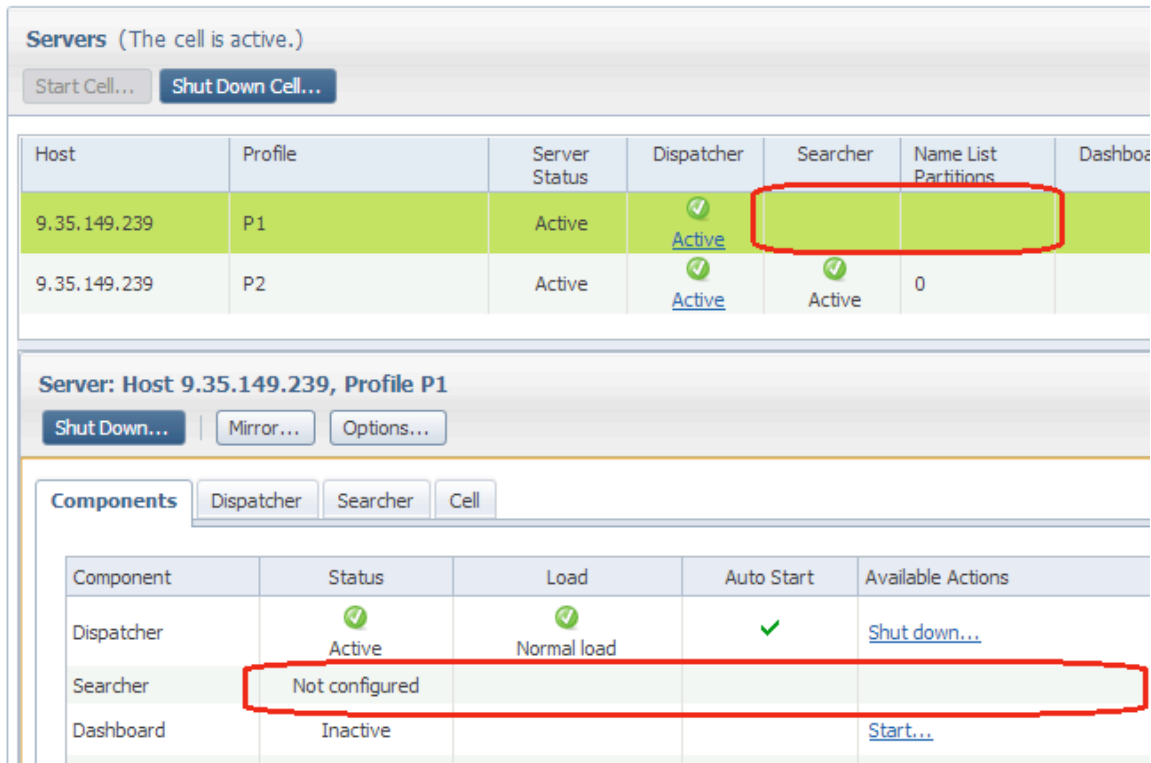


Figure 25 - After unconfiguring a searcher

### 8.12 Other information in the ENS Console

The server detail section of the console has four tabs. So far we've only discussed the Components tab. The other three have information about the state and performance of the selected server's dispatcher and searcher (if any), and about the cell as a whole.

#### 8.12.1 URL for the Search GUI

The console's dispatcher subtab includes the URL of the ENS search GUI for the selected dispatcher. This is something that an administrator may need to provide to users who want to search for names. Clicking the URL also opens the GUI in a new tab.

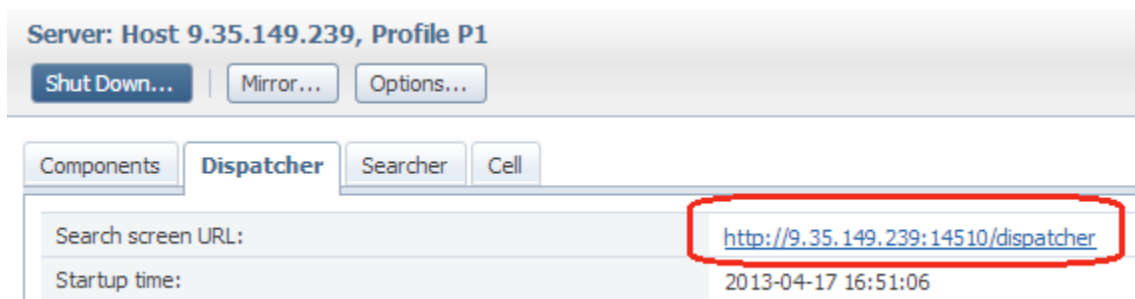


Figure 26 - URL for search GUI shown in console

### 8.12.2 The product version

This is shown on the cell tab, as well as in the About dialog.

### 8.12.3 Number of names.

The cell tab shows the number of source names and search names in the system. In a system with a very large number of names, these are estimates since an exact count of all names in such a system can be expensive.

## 9 Managing Users, Passwords, and Roles

### 9.1 In a Nutshell

For a quick start, you can just use the administrative userid and password that you created in the installer in section 3.10, and not worry about other userids or permissions. The administrative user can use all ENS console and search operations, and can manage and search any name list in the system.

For a production system, depending on requirements, you may want to define multiple userids with different permissions. The following sections describe this.

### 9.2 Liberty Security

ENS incorporates the WebSphere Liberty application server to host its web components, and relies on WebSphere Liberty's security implementation for authentication and authorization.

By default, Liberty uses a simple file-based store for information on users, groups, and passwords. In ENS, the file containing this information is found at `<ensroot>/ibm-home/wlp/users.xml`. You can add, remove, and manage users by editing this file. This can be done while the system is running.

Users can change their own ENS passwords on a change-password GUI screen.

### 9.3 Supported Roles in ENS

The ENS installer creates an initial administrative user with a name and password that you provide at installation time. This user is a member of the "admins" group, which gives it the ability to do all ENS operations.

Depending on your needs, this one user definition may be sufficient. You might also want to define multiple users but give them all admin rights.

If you need a more complex security model, you can define multiple users having different rights. If needed, these rights can be set up in a fine-grained way to allow users in particular groups/roles to have access to particular name lists, as described in the next section.

In all cases, you define and manage these users and permissions by adding and modifying `<user>` and `<member>` elements in the `users.xml` file.



ENS supports the following groups to control access to operations and to name lists.

Group	Rights
admins	<ul style="list-style-type: none"> <li>• Use all console operations to view and change the system state.</li> <li>• Search and manage any name list.</li> <li>• Use the dashboard.</li> </ul>
console_users	<ul style="list-style-type: none"> <li>• Use all console operations to view and change the system state.</li> <li>• Use the dashboard.</li> </ul>
dashboard_users	<ul style="list-style-type: none"> <li>• Use the dashboard to view the system state.</li> </ul>
managers_all_lists	<ul style="list-style-type: none"> <li>• Search and manage (add/delete from) any name list.</li> </ul>
managersNNN* (e.g. managers101)	<ul style="list-style-type: none"> <li>• Search and manage name lists for which this manager role was specified when creating the list in the NameLoader. Used for fine-grained control as in section 9.4.</li> </ul>
searchers_all_lists	<ul style="list-style-type: none"> <li>• Search any name list.</li> </ul>
searchersNNN* (e.g. searchers101)	<ul style="list-style-type: none"> <li>• Search name lists for which this searcher role was specified when creating the list in the NameLoader. Used for fine-grained control.</li> </ul>

**Table 10 - ENS Groups**

#### 9.4 Fine-grained control at the name list level

The security model for ENS allows the option of fine-grained control where you can grant particular users search access to some name lists but not to others.

As described in section 5.5, when you add a name list using NameLoader, you provide the name of a searcher role (e.g. “searcher101”) for the list. Any user in the associated group (“searchers101”) is allowed to search for names in that list.

Similarly, you specify the name of a manager role (e.g. “managers101”) for that list. Any user in the associated group (“managers101”) is allowed to add and delete names in that list via web services.

When you add a new user, you do so by adding elements to the users.xml file to define the user and to indicate to which groups that user belongs.

Note that the names of groups as seen in users.xml are pluralized (“searchers101”) but the corresponding role names seen in NameLoader are singular (“searcher101”).

These fine-grained roles are optional. You don’t have to use them if you find that the generic “searchers\_all\_lists” and “managers\_all\_lists” groups are sufficient, though you still need to specify the roles when adding names with NameLoader.

By default the roles available are searcher101-searcher120 and manager101-manager120. If you need a wider range of roles, up to searcher1-searcher500 and manager1-manager500, you can define them by adding more <group> elements in the users.xml file and <security-role> elements in applications.xml. Both files are found in the <ensroot>/ibm-home/wlp folder. See the comments in those files for more information.

### 9.5 The users.xml file

The ENS installer writes information about the initial admin user to the users.xml file, found at <ensroot>/ibm-home/wlp/users.xml . After that, you can edit that file by hand to add, delete, and manage permissions of users.

#### 9.5.1 Adding a new user

The users.xml file has a section near the top with the comment “List users here”:

```
<!-- List users here -->
<user name="ensadmin" password="{aes}AICwWs/XAgx70bztj0s09zkS31vubq0rERfNM/o81K"/>
<user name="ens-app-user" password="{aes}ACA5snq8qKUydWlrc9+jK0HgvjSyy/e0DIgT+5Usw"/>
```

You can add a user by adding a line in this section. Do not remove or modify the two entries for the admin user and the internal ens-app-user created at installation time.

When you add a new user, you can give it some plain-text initial password like this:

```
<user name="smith" password="changeMe"/>
```

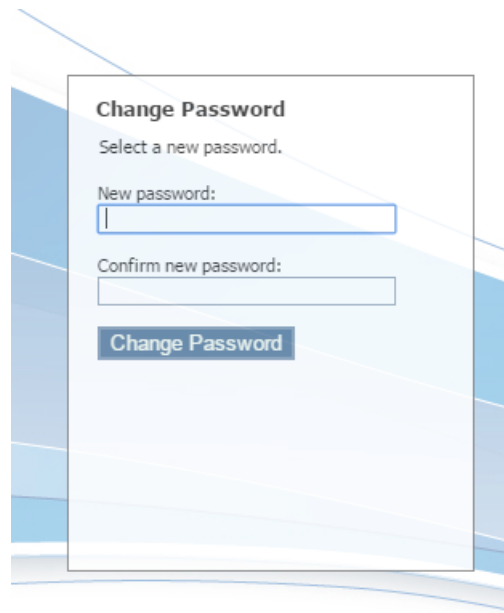
The user name and password must contain only letters (a-z and A-Z) and digits (0-9).

In addition to adding the <user> element, you’ll need to list the user in one or more groups as in section 9.5.3 below to specify which operations the new user can perform.

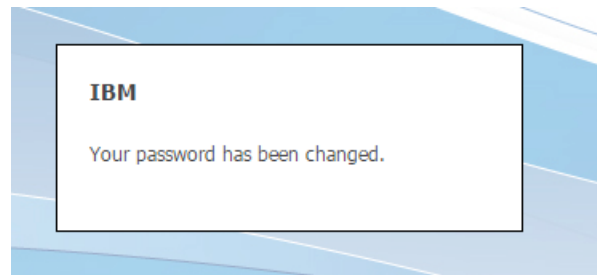
After you add a user entry, the user should change their password as in the next section.

#### 9.5.2 Changing your own password

Individual users can change their own passwords by browsing (in a running system) to <http://port:host/changePassword.jsp> . This displays first a login page, then a change-password form like this:



On successfully changing their password, the user sees this:



and the users.xml file is updated to contain an encrypted password like this:

```
<user name="smith" password="{aes}AJ0Ve9aQURVS+1XWC0Md0ImIXpnGtVPNHj8YrYuaqZ0Qs"/>
```

### 9.5.3 Resetting a password

As an administrator, you can change a user's password back to a known value by changing their `<user>` element in users.xml back to a simple plain-text password:

```
<user name="smith" password="changeMeAgain"/>
```

After you do this, the user should change their password with the form as in section 9.5.2 so that it is encrypted again.

### 9.5.4 Specifying permissions (group membership) for a user

Below the list of `<user>` entries, the user.xml file contains a number of `<group>` elements like the following.

```
<!--  
The admins group can do anything. This effectively implies all other groups.  
It is not necessary to list a member of this group in the other groups.
```

## Setting Up and Managing ENS

```
-->
<group name="admins">
  <member name="ens-app-user"/>
  <member name="ensadmin"/>
</group>

<!--
  The console_users group can use the console to see and manage system state.
  This implies dashboard_users too; it is not necessary to list the same user in both groups.
-->
<group name="console_users">
</group>

<!--
  The dashboard_users group can use the dashboard to see system state.
-->
<group name="dashboard_users">
</group>

<!--
  The managers_all_lists group can add/delete/get/search in all lists.
  This implies searchers_all_lists too; it is not necessary to list the same user in both groups.
  It also implies all numbered managerXXX and searcherXXX groups; it is not necessary to
  list a user from this group in any of the numbered manager or searcher groups.
-->
<group name="managers_all_lists">
</group>

<!--
  The searchers_all_lists group can search in all lists.
  This implies all numbered searcherXXX groups; it is not necessary to list a user from
  this group in any of the numbered searcher groups.
-->
<group name="searchers_all_lists">
</group>

<!--
  Numbered searcher and manager groups to support fine-grained control of access to name lists.
  By default only searchers101 through 120 and managers101 through 120 are defined here,
  but you can add more, up to searchers1 through searchers500 and managers1 through managers500,
  if wanted. If you add them here, you also need to add them in the applications.xml file.
-->
<group name="searchers101"></group>
<group name="searchers102"></group>
<group name="searchers103"></group>
...
<group name="searchers120"></group>
<group name="managers101"></group>
<group name="managers102"></group>
<group name="managers103"></group>
<group name="managers104"></group>
...
<group name="managers120"></group>
```

You add a user to a group by inserting a `<member>` element in some group, like this:

```
<group name="searchers_all_lists">
  <member name="jdoe"/>
</group>
```

WebSphere Liberty reads the `users.xml` file at profile startup time, but also watches for changes to that file while running. This means you can edit `users.xml` to add/delete/modify users in a running system.

If you make a mistake and save the users.xml file with an invalid element, WebSphere Liberty typically logs an error message in the console.log file in <ensroot>/<profile name>/logs/console.log. Usually the error message indicates the line in the file where the error was detected.

### 9.5.5 Deleting a user

You delete a user by deleting their <user> element and deleting any <member> elements with the same name.

## 10 Searching for Names

This is described in much more detail in a separate “Using ENS to Search for Names” document, but here’s a quick look at searching for names in a running ENS system. For this example, we assume that:

- You’ve done the steps listed in Figure 1 (detailed in sections 2-9) to set up and start your ENS system, including configuring and starting the cell.
- You specified host “myhost.mycompany.com” and the default web service port 14510, in (sections 3.8 and 3.18) respectively, and
- You used nameLoader to add name lists called “EMPLOYEES” and “GOOD\_CUSTOMERS” in section 5.

Now browse to the search GUI at <http://myhost.com:14510/ens/dispatcher>. First you’ll see a login screen where (for now) you can provide the ENS admin credentials that you specified in section 3.10.

On successful login, you see the search GUI screen, like this:

Figure 27 - Search GUI before Searching

## Setting Up and Managing ENS

When you type a given name and surname into the search form and click the Search button at the top, ENS searches for names and displays the results:

The screenshot shows the IBM InfoSphere Global Name Management Enterprise Name Search interface. The top bar includes the title "IBM InfoSphere Global Name Management Enterprise Name Search" and navigation links: "Welcome ensadmin | About | Help | Log Out" and the IBM logo.

**Name Search Criteria**

Buttons: Search, Clear

Name type: Personal Name (Fields)

Given name: Jose Culture (optional): [dropdown] ?

Surname: Martin Culture (optional): [dropdown] ?

Name lists:  All  Specified:

- EMPLOYEES
- GOOD\_CUSTOMERS

Advanced:

Search strategy: [Default strategy]

Search category: All names

Minimum name score (0-100):  Default  Specified: [spinners]

Maximum results returned:  Default  Specified: [spinners]

Include searches on alternate parses of this name

**Search Results**

Parse Details: Given name "JOSE" (Hispanic), Surname "MARTIN" (European) ?

Names Found (4) ?

1-4 of 4 Page 1 of 1 Items per page: 20

Name Score	Surname	Given Name	SN Culture	GN Culture	Name List	Total Count
75	ALLENDES MARIN	JOSE IGNACIO	Hispanic	Hispanic	GOOD_CUSTOMERS (1)	1
74	ALVARADO MARTINEZ	JOSE CORONADO	Hispanic	Hispanic	GOOD_CUSTOMERS (1)	1
72*	ALVARADO MARTINEZ	JOSE CORONADO	Hispanic	Hispanic	GOOD_CUSTOMERS (1)	1
71	ACUNA MARQUIN	JOSE A	Hispanic	Hispanic	GOOD_CUSTOMERS (1)	1

\* - Name score based on alternate parse

**Figure 28 - Search GUI with Results**

The search form lets you specify the name that you're searching for in several ways, and provides a number of settings that influence how the search is done.

You can click on the triangle on the "Parse Details" header to see how NameWorks parsed the name that you were searching for.

The results grid shows a separate row for each distinct form of source name that was found, initially sorted by descending score. The last two columns show how many occurrences were found in each name list, and the total number of occurrences found. Clicking on the number in the "Total Count" column displays a dialog with details on the names found, including their external ids.

See the "Using ENS to Search for Names" document for more information.

## 11 Uninstalling ENS

The ENS installer makes changes to the file system in the <ensroot> folder that you specify in section 3.8, and in the database within the schema that you specify in section 3.14 or 3.15.

The simplest way to uninstall ENS is to remove that folder (as well as the files in the “local” ENS folder for each host in a distributed installation), and to delete the contents of that schema.



## 12 Advanced Topics

### 12.1 NameWorks Configuration File

The main configuration files and related data files used by the NameWorks instances in ENS are found at <ensroot>/data.

In that folder, the relevant files are:

- `ens.config` – the main NameWorks configuration file.
- `datalist_template.config` – a template file providing paths for data files used by NameWorks.

By default, the NameLoader program uses `ens.config` by itself.

The NameWorks instance within an ENS searcher uses a combination of `ens.config` and material from `datalist_template.config`, in a temporary configuration file that is created at searcher startup time.

Configuring NameWorks by modifying its configuration files is outside the scope of this document; see the NameWorks documentation for more on this.

### 12.2 Using SSL

By default, ENS uses HTTP for communication between client and server, between dispatchers and searchers, and between the console and other components.

You can change this communication to use SSL by modifying one record in the `ENS_CONFIG` table in your database. That record is the one with `PARAM_NAME` equal to “https.enabled”. If you change its `VALUE_TEXT` to “true” instead of “false”, leaving everything else the same, and then restart all of your ENS servers, they will use SSL.

## 13 What's New in ENS Version 6

### 13.1 Multiple names can have the same external id

Prior to version 6, ENS required that names be unique within a name list. Version 6 removes this restriction and allows multiple names with the same external id. This means you can have both “John Wayne” and “Marion Mitchell Morrison” with external id DUKE1.

The old “replaceOnDuplicate” and “duplicateLogging” flags are now obsolete.

### 13.2 Support for Polish, Portuguese, and Turkish cultures

Version 6 of GNM and ENS include support for these three new cultures, with rules added to recognize them in analysis of names and to improve search results.

### 13.3 Organization names can have cultures

ENS and GNM version 6 allow organization names to have cultures to help improve match results when searching. Previously this was only supported for personal names.

Both in NameLoader and in the addName web service, when you provide an organization name as input you can now include an explicit culture field. This is also true in the query name in a search.

If you don't provide an explicit culture but the input name data is in a non-Latin script, the system will try to infer a culture from the script type. It does this for names in Arabic, Chinese, Cyrillic, Indian, Japanese, and Korean script. Culture cannot be inferred from script type for Latin script names because many cultures use that script. It is not inferred for Greek script names because GNM does not have specific support for Greek culture in name matching.

Note that, unlike with personal names, GNM does not analyze the actual content of an organization name to try to determine a culture. Only the script type is considered.

In other words, for example, GNM does not see the word “Banco” in an organization name and infer Hispanic culture, but if it sees an organization name written in Korean script like “우리 은행”, it infers Korean culture.

### 13.4 Explicit Script Type and Extended File Format

Most script types can be recognized automatically, but ENS and GNM cannot automatically distinguish Hanzi from Kanji in input names. If you are providing either kind of name, you can include an explicit script type field to differentiate them. To do this in NameLoader, use the new extended file format described in section 5.2.4.

### 13.5 NameLoader allows commas within names

ENS 6.0 allows name fields that contain commas. To include a name field that contains commas in a field in the csv file, put it in quotes:

```
"Monsters, Inc",,org101,,1  
"John Smith, Esquire",,abc123
```

Between opening and closing quotes, commas are treated as part of the field and not as field separators.

### 13.6 Support for DB2 10.5 and Oracle 12c

ENS version 6 runs with DB2 10.5 and Oracle 12c. It takes advantage of DB2 10.5's support of functional indexes. In Oracle, it allows service names (as well as SIDs) when specifying database details.

### 13.7 WebSphere Liberty

Internally, ENS 6.0 now uses WebSphere Liberty as its application server for hosting web components, instead of the earlier Embedded WebSphere. This has several implications for administrators and users:

There is no longer a `wpswd` utility for managing users, roles, and passwords. Instead, an administrator manages these by editing the `users.xml` file as in section 9.5. Users can change their own passwords in a running system as in section 9.5.2.

You no longer need to use the `enscu` utility to install an updated ENS ear file if one is provided by IBM support. Instead you just copy the new ear file to `<ens root>/ibm-home/wlp/apps`.

You no longer need to use the `enscu` utility to change logging settings if directed by IBM support. Instead you edit the `<ens root>/ibm-home/wlp/logging.xml` file, and then uncomment a line in `<ens root>/wlp/usr/servers/<profileName>/server.xml` to enable its use.

The URLs for the ENS GUI and web services have changed. The GUI URLs now have paths starting with `"/ens"`, as in:

<http://host:port/ens/console/>  
<http://host:port/ens/dispatcher/>

The web service urls now have paths starting with `"/ws"`, as in:

<http://host:port/ws/resources/wsdl/EnterpriseNameSearcher.wsdl>  
<http://host:port/ws/dispatcher/api/rest/search/matches>