Reference Manual

Sun™ ONE Directory Server

Version 5.2

816-6699-10
June 2003

Contents

About This Reference Manual

Sun™ ONE Directory Server 5.2 is a powerful and scalable distributed directory server based on the industry-standard Lightweight Directory Access Protocol (LDAP). Sun ONE Directory Server software is part of the Sun Open Net Environment (Sun ONE), Sun's standards-based software vision, architecture, platform, and expertise for building and deploying Services On Demand.

Sun ONE Directory Server is the cornerstone for building a centralized and distributed data repository that can be used in your intranet, over your extranet with your trading partners, or over the public Internet to reach your customers.

Purpose of This Reference Manual

Most Directory Server administrative tasks can be performed through the Sun ONE Server Console, the graphical user interface provided with Sun ONE Directory Server. For information on using the Sun ONE Server Console, see *Sun ONE Server Console Server Management Guide*, and for details of how to use the console to manage the Directory Server in particular, see the *Sun ONE Directory Server Administration Guide*.

This reference manual deals with the other methods of managing the Directory Server, namely altering the server configuration attributes via the command line and using the command-line utilities.

The reference manual provides comprehensive information on the command-line utilities and scripts provided with Sun ONE Directory Server, configuration attributes, file formats, schemas, and error and connection codes.

For experienced users of Sun ONE Directory Server and the previous documentation set, note that this reference manual combines the *Configuration, Command and File Reference* and the *Schema Reference* of previous releases.

Contents of This Reference Manual

This reference manual contains the following sections:

Part 1 - Command-Line Utilities and Scripts

Command-Line Utilities

Directory Server comes with a set of configurable command-line utilities that you can use to search and modify entries in the directory and administer the server. Chapter 1, “Command-Line Utilities” describes these command-line utilities and contains information on where the utilities are stored and how to access them. In addition to these command-line utilities, Directory Server also provides `ns-slapd` and `slapd.exe` command-line utilities for performing directory operations as described in Appendix B, “ns-slapd and slapd.exe Command-Line Utilities.”

Command-Line Scripts

In addition to command-line utilities, several non-configurable scripts are provided with the Directory Server that make it quick and easy to perform routine server administration tasks from the command line. Chapter 2, “Command-Line Scripts” lists the most frequently used scripts and contains information on where the scripts are stored and how to access them.

Part 2 - Server Configuration

Core Server Configuration

The format and method for storing configuration information for Sun ONE Directory Server 5.2 mark a significant change from previous versions of the Directory Server. A full explanation of these changes and a listing for all server attributes can be found in Chapter 3, “Core Server Configuration” and Chapter 5, “Plug-In Implemented Server Functionality.”

Core Server Configuration Attributes

This chapter provides an alphabetical reference of all the attributes involved in configuring and monitoring the core server functionality.

Plug-in Implemented Server Functionality Reference

This chapter serves as a plug-in implemented server functionality reference and includes an alphabetical list of attributes common to all plug-ins, attributes allowed by certain plug-ins, database plug-in attributes, and retro changelog plug-in attributes.

Migration From Earlier Versions

In version 4.x of Directory Server, all configuration parameters were stored in text files. However, in Sun ONE Directory Server 5.2, configuration attributes are stored as LDAP configuration entries in a `dse.ldif` file. The mapping of configuration parameters in Directory Server 4.1, 4.11, and 4.12 to the corresponding configuration entries and attributes in Sun ONE Directory Server 5.2 is described in Chapter 6, “Migration From Earlier Versions.”

Part 3 - File Reference

Server Instance File Reference

This chapter provides an overview of the files and configuration information stored in each instance of Directory Server. This assists administrators in understanding the changes or absence of changes in the course of directory activity. In terms of security, such an overview can help administrators to detect errors and intrusion as they know what kind of changes to expect and what should be considered abnormal behavior.

Access Log and Connection Code Reference

Monitoring allows you to detect and remedy failures and, when done proactively, to anticipate and resolve potential problems before they result in failure or poor performance. This chapter provides the information you need to understand the structure and content of the logs, thereby enabling you to monitor your directory more effectively.

Part 4 - Directory Server Schema

About Schema

This chapter provides an overview of some of the basic concepts of the directory schema, and lists the files in which the schema is described. It describes object classes, attributes and Object Identifiers (OIDs), and briefly discusses extending server schema and schema checking.

Object Class Reference

This chapter contains an alphabetical list of the object classes accepted by the default schema. It provides a definition of each object class, and lists its required and allowed attributes. The object classes listed in this chapter are available for you to use to support your own information in the Directory Server. Object classes that are used by the Directory Server or other Sun ONE products for internal operations are not documented here.

Attribute Reference

This chapter contains an alphabetic list of the standard attributes. It provides a definition of each attribute, and gives the attribute syntax and OID.

Operational Attributes, Special Attributes, and Special Object Classes

This chapter describes the operational attributes used by the directory server. Operational attributes may be available for use on every entry in the directory, regardless of whether they are defined for the object class of the entry. This chapter also describes certain special attributes and object classes that are used by the server.

Appendices

Error Codes

This appendix provides an extensive list of the error messages generated by Sun ONE Directory Server. While this list is not exhaustive, the information presented in this chapter will serve as a a good starting point for common problems.

Using the ns-slapd and slapd.exe Command-Line Utilities

This appendix looks at the `ns-slapd` (UNIX) and `slapd.exe` (Windows) command-line utilities that can also be used to perform the same tasks as the scripts and utilities described previously.

Directory Internationalization

Directory Server allows you to store, manage, and search for entries and their associated attributes in a number of different languages. This appendix provides information on the locales and language types supported by Sun ONE Directory Server.

LDAP URLs

One way to express an LDAP query is to use a URL to specify the directory server host machine and the DN or filter for the search. This appendix provides information on the components of an LDAP URL and on escaping unsafe characters. It also provides several LDAP URL examples.

LDAP Data Interchange Format

Sun ONE Directory Server uses the LDAP Data Interchange Format (LDIF) to describe a directory and directory entries in text format. This appendix provides information on the LDIF file format, specifying directory entries using LDIF, defining directories using LDIF, and storing information in multiple languages.

Prerequisite Reading

Before using this manual we strongly recommend that you read the online release notes to obtain the latest information about new features and enhancements in this release of Sun ONE Directory Server. The release notes can be found at

```
http://docs.sun.com/db/doc/816-6703-10
```

This reference manual does not describe many of the basic directory and architectural concepts that you need to successfully design, implement, and administer your directory service.

To familiarize yourself with basic directory concepts, refer to the *Sun ONE Directory Server Getting Started Guide*. This guide provides you with enough information to install Sun ONE Directory Server for evaluation purposes. Complete installation and configuration information is provided in the *Sun ONE Directory Server Installation and Tuning Guide.*

To plan, implement, and administer your directory, refer to the *Sun ONE Directory Server Deployment Guide* and the *Sun ONE Directory Server Administration Guide.*

Typographical Conventions

This section explains the typographical conventions used in this book.

`Monospaced font` - This typeface is used for literal text, such as the names of attributes and object classes when they appear in text. It is also used for URLs, filenames and examples.

Italic font - This typeface is used for emphasis, for new terms, and for text that you must substitute for actual values, such as placeholders in path names.

The greater-than symbol (>) is used as a separator when naming an item in a menu or sub-menu. For example, Object > New > User means that you should select the User item in the New sub-menu of the Object menu.

NOTE Notes, Cautions and Tips highlight important conditions or limitations. Be sure to read this information before continuing.

Default Paths and Filenames

All path and filename examples in the Sun ONE Directory Server product documentation are one of the following two forms:

- *ServerRoot/...* - The *ServerRoot* is the location of the Sun ONE Directory Server product. This path contains the shared binary files of Directory Server, Sun ONE Administration Server, and command line tools.

  The actual *ServerRoot* path depends on your platform, your installation, and your configuration. The default path depends on the product platform and packaging as shown in Table 1.

- *ServerRoot*`/slapd-`*serverID*`/...` - The *serverID* is the name of the Directory Server instance that you defined during installation or configuration. This path contains database and configuration files that are specific to the given instance.

NOTE Paths specified in this manual use the forward slash format of UNIX and commands are specified without file extensions. If you are using a Windows version of Sun ONE Directory Server, use the equivalent backslash format. Executable files on Windows systems generally have the same names with the `.exe` or `.bat` extension.

Table 1 Default *ServerRoot* Paths

Product Installation	*ServerRoot* Path
Solaris 9[1]	`/var/mps/serverroot` - After configuration, this directory contains links to the following locations: • `/etc/ds/v5.2` (static configuration files) • `/usr/admserv/mps/admin` (Sun ONE Administration Server binaries) • `/usr/admserv/mps/console` (Server Console binaries) • `/usr/ds/v5.2` (Directory Server binaries)
Compressed Archive Installation on Solaris and Other Unix Systems	`/var/Sun/mps`
Zip Installation on Windows Systems	`C:\Program Files\Sun\MPS`

1. If you are working on the Solaris Operating Environment and are unsure which version of the Sun ONE Directory Server software is installed, check for the existence a key package such as `SUNWdsvu` using the `pkginfo` command. For example: `pkginfo | grep SUNWdsvu`.

Directory Server instances are located under *ServerRoot*`/slapd-`*serverID*`/`, where *serverID* represents the server identifier given to the instance on creation. For example, if you gave the name `dirserv` to your Directory Server, then the actual path would appear as shown in Table 2. If you have created a Directory Server instance in a different location, adapt the path accordingly.

Table 2 Default Example `dirserv` Instance Locations

Product Installation	Instance Location
Solaris 9	`/var/mps/serverroot/slapd-dirserv`

Table 2 Default Example `dirserv` Instance Locations *(Continued)*

Product Installation	Instance Location
Compressed Archive Installation on Solaris and Other Unix Systems	`/usr/Sun/mps/slapd-dirserv`
Zip Installation on Windows Systems	`C:\Program Files\Sun\MPS\slapd-dirserv`

Downloading Directory Server Tools

Some supported platforms provide native tools for accessing Directory Server. More tools for testing and maintaining LDAP directory servers, download the Sun ONE Directory Server Resource Kit (DSRK). This software is available at the following location:

```
http://wwws.sun.com/software/download/
```

Installation instructions and reference documentation for the DSRK tools is available in the *Sun ONE Directory Server Resource Kit Tools Reference*.

For developing directory client applications, you may also download the Sun ONE LDAP SDK for C and the Sun ONE LDAP SDK for Java from the same location.

Additionally, Java Naming and Directory Interface (JNDI) technology supports accessing the Directory Server using LDAP and DSML v2 from Java applications. Information about JNDI is available from:

```
http://java.sun.com/products/jndi/
```

The JNDI Tutorial contains detailed descriptions and examples of how to use JNDI. It is available at:

```
http://java.sun.com/products/jndi/tutorial/
```

Suggested Reading

Sun ONE Directory Server product documentation includes the following documents delivered in both HTML and PDF:

- *Sun ONE Directory Server Getting Started Guide* - Provides a quick look at many key features of Directory Server 5.2.

- *Sun ONE Directory Server Deployment Guide* - Explains how to plan directory topology, data structure, security, and monitoring, and discusses example deployments.
- *Sun ONE Directory Server Installation and Tuning Guide* - Covers installation and upgrade procedures, and provides tips for optimizing Directory Server performance.
- *Sun ONE Directory Server Administration Guide* - Gives the procedures for using the console and command-line to manage your directory contents and configure every feature of Directory Server.
- *Sun ONE Directory Server Reference Manual* - Details the Directory Server configuration parameters, commands, files, error messages, and schema.
- *Sun ONE Directory Server Plug-In API Programming Guide* - Demonstrates how to develop Directory Server plug-ins.
- *Sun ONE Directory Server Plug-In API Reference* - Details the data structures and functions of the Directory Server plug-in API.
- *Sun ONE Server Console Server Management Guide* - Discusses how to manage servers using the Sun ONE Administration Server and Java based console.
- *Sun ONE Directory Server Resource Kit Tools Reference* - Covers installation and features of the Sun ONE Directory Server Resource Kit, including many useful tools.

Other useful information can be found on the following Web sites:

- Product documentation online: `http://docs.sun.com/coll/S1_DirectoryServer_52`
- Sun software: `http://wwws.sun.com/software/`
- Sun ONE Services: `http://www.sun.com/service/sunps/sunone/`
- Sun Support Services: `http://www.sun.com/service/support/`
- Sun ONE for Developers: `http://sunonedev.sun.com/`
- Training: `http://suned.sun.com/`

NOTE Sun Microsystems Inc., is not responsible for the availability of third-party Web sites mentioned in this document. Sun Microsystems Inc. does not endorse and is not responsible or liable for any content, advertising, products, or other material on or available from such sites or resources. Sun Microsystems Inc. will not be responsible or liable for any damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through any such sites or resources.

Part 1

Command-Line Utilities and Scripts

This part contains reference information on the command-line utilities and scripts provided with Sun ONE Directory Server 5.2. Note that the LDAP utilities are provided with the Directory Server Resource Kit (DSRK) and are documented in the *Directory Server Resource Kit Tools Reference*.
This part includes the following chapters:

- Command-Line Utilities
- Command-Line Scripts

Chapter 1

Command-Line Utilities

This chapter contains reference information on the command-line utilities provided with Sun ONE Directory Server 5.2. Note that the LDAP utilities are provided with the Directory Server Resource Kit (DSRK) and are documented in the *Directory Server Resource Kit Tools Reference*.

This chapter is divided into the following sections:

- Finding and Executing Command-Line Utilities
- Command-Line Utilities Quick Reference
- LDIF Command-Line Utilities
- Replication Monitoring Tools
- Other Tools

Finding and Executing Command-Line Utilities

All the command-line utilities, apart from `ldif` and `pwdhash`, are located in the following directory:

ServerRoot`/shared/bin`

The `ldif` and `pwdhash` command-line utilities are located in the following directory:

ServerRoot`/bin/slapd/server`

CAUTION To execute the command-line utilities, you must change to the directory in which they are stored. Although it is possible to set command-path and library-path variables to execute the utilities, this is *not* recommended procedure. You run the risk of disrupting the correct execution of other utilities and of compromising the security of the system, particularly when you have more than one server version installed.

The same procedure also applies to the Perl scripts provided with Directory Server. For further information on these and other scripts, see Chapter 2, "Command-Line Scripts."

Command-Line Utilities Quick Reference

Table 1-1 describes the command-line utilities that enable you to format LDIF files. Note that the LDAP utilities (`ldapsearch`, `ldapmodify`, `ldapdelete` and `ldapcompare`) are provided with the Directory Server Resource Kit (DSRK) and are documented in the *Directory Server Resource Kit Tools Reference*.

Table 1-1 LDIF Command-Line Utilities

Command-Line Utility	Description
`ldif`	Formats input by adding base 64 encoding to make it suitable as an attribute value for inclusion in an LDIF file.
`fildif`	Enables you to create a filtered version of an LDIF input file.

Table 1-2 describes the command-line utilities that enable you to manage and monitor replication:

Table 1-2 Replication Monitoring Tools

Command-Line Utility	Description
`insync`	Indicates the synchronization state between a master replica and one or more consumer replicas.
`entrycmp`	Compares the attributes and values of the same entry on two different servers.
`repldisc`	Enables you to “discover” a replication topology, constructing a graph of all known servers and displaying a matrix describing the topology.

Table 1-3 describes the other command-line utilities provided with Sun ONE Directory Server:

Table 1-3 Other Command-Line Tools

Command-Line Utility	Description
`pwdhash`	Prints the encrypted form of a password using one of the server's encryption algorithms.

LDIF Command-Line Utilities

ldif

The `ldif` command-line utility formats input by adding base 64 encoding to make it suitable for inclusion in an LDIF file. This makes it easy to include binary data, such as JPEG images, along with other textual attribute values. In an LDIF file, base 64 encoded attribute values are indicated by a `::` after the attribute name, for example:

`jpegPhoto::` *encoded data*

In addition to binary data, other values that must be base 64 encoded include:

- any value that begins with a semicolon (;) or a space
- any value that contains non-ASCII data, including newlines

The `ldif` command-line utility takes any input and formats it with the correct line continuation and appropriate attribute information.

To undo base 64 encodings in LDIF files, you can use the `ldifxform` utility in the Directory Server Resource Kit (DSRK), with the `-c nob64` option. Note, however, that the resulting file may not be reparsable as LDIF. For more information on the tools provided with the DSRK, see the *Directory Server Resource Kit Tools Reference.*

Syntax

The `ldif` command has the following format:

Solaris packages
`/usr/sbin/directoryserver ldif [-b] [`*attrtypes*`]`

Other platforms
ServerRoot`/bin/slapd/server ldif [-b] [`*attrtypes*`]`

Options

Option	Meaning
`-b`	Specifies that the `ldif` utility should interpret the entire input as a single binary value. If `-b` is not present, each line is considered to be a separate input value. As an alternative to the `-b` option, you can you can use the `:<` URL specifier notation, which is in fact simpler to use. For example: `jpegphoto:< file:///tmp/myphoto.jpg` Although the official notation requires three `///`, the use of one `/` is tolerated.

fildif

This utility enables you to create a filtered version of any LDIF input file. `fildif` does not require the directory server to be running.

`fildif` takes a configuration file as an input parameter. This configuration file must conform to the configuration rules of the Filtering Service included as part of Sun ONE Directory Server, and must contain the specific set and element entries that define these rules. The configuration rules can be defined using the Sun ONE Server Console or at the command line. For more information on the Filtering Service and how it is configured, see Chapter 8, “Managing Replication” in the *Sun ONE Directory Server Administration Guide*.

Sun ONE Directory Server allows you to configure the following filtering rules:

1. Filter in a list of attributes that must be included in an entry.
2. Filter out a list of attributes that must be excluded from an entry.

A filtering service configuration is accessed through a *pointer entry*. The pointer entry is provided to `fildif` with the `-b` parameter. A *pointer attribute* within this entry (provided by the `-a` parameter) determines the RDN of the filtering service configuration entry to be used for the filtering.

Syntax

The `fildif` command has the following format:

`fildif -i` *input_file* `[-f] [-o` *output_file*`] [-c` *config_file*`] -b` *pointer_entry* `[-a` *pointer_attr*`]`

Options

Option	Meaning
`-i`	The input LDIF file whose contents will be filtered. This parameter is mandatory.
`-f`	Forces `fildif` to overwrite the contents of the specified output file, if it exists.
`-o`	The output LDIF file in which the filtered results will be stored. If no output file is specified, the default output file is `./output.ldif`.
`-c`	The configuration file in which the filtering configuration is stored.
`-b`	The pointer entry. This parameter is mandatory and specifies the DN of the entry that will be used as the filtering service configuration entry point. The entry specified by this DN must exist in the configuration file specified by the `-c` parameter.
`-a`	The attribute that will be used inside the pointer entry to point to a particular filtering service configuration definition. If this parameter is not present, the default `partialReplConfiguration` is used.

Exit Status

The following exit values are returned:

`0` Successful completion

`1` An error occurred

On error, verbose error messages are output to standard output.

Example

```
# fildif -i data.ldif -o filt_data.ldif -f -c config_fildif.ldif -b
"cn=conf_20,cn=sets,cn=filtering service,cn=features,cn=config" -a
ds5PartialReplConfiguration
```

For more information, see the `fildif(1M)` manual page.

Replication Monitoring Tools

The replication monitoring tools enable you to monitor replication between servers, and to assist in identifying the cause of replication problems.

The following replication monitoring tools are provided:

- `insync`
- `entrycmp`
- `repldisc`

Before describing how these tools work, it is important that you understand the following general replication information.

A Replication Update Vector (RUV) is maintained on each replica. The RUV identifies each master replica within the topology, its Replica ID, and the latest change on each master, expressed as a Change Sequence Number (CSN). A CSN identifies each change made to a master server. A CSN consists of a timestamp, a sequence number, the master Replica ID, and a subsequence number.

The machine on which you are running the `insync` and `entrycmp` tools must be able to reach all the specified hosts. If the hosts are unreachable due to a firewall, VPN, or other network setup reasons, you will encounter difficulties using these tools. For the same reason, you should ensure that all the servers are up and running before attempting to use the replication monitoring tools.

The replication monitoring tools connect to the server(s) via LDAP and rely on access to `cn=config` to obtain the replication status. The user of these tools *must* therefore have read access to the data under `cn=config`. This should be taken into account particularly when replication is configured over SSL.

The following section describes the usage of the replication monitoring tools and provides an explanation for each of the options. Certain options are common to all the tools, and are discussed at the beginning of the section.

Common Replication Monitoring Tool Options

The following table describes the options that are common to all the replication monitoring tools.

Table 1-4 Common Replication Monitoring Tool Options

Option	Meaning
HostSpec	*HostSpec* is defined as: [bindDN[:[password]]@]*host*[:port] For example `"cn=directory manager":mypword@myServer:5201`
ServerSpec	The server specification. This can be: -s / -S *HostSpec* [-c / -C *HostSpec* -c / -C *HostSpec* ...] or -c / -C *HostSpec* [-s / -S *HostSpec* -s / -S *HostSpec* ...] where `-s` is the supplier replica and `-c` is the consumer replica. You can specify any number of supplier and consumer replicas in this list. If you are using SSL, use `-S` and `-C` in the server specification. In addition, if you are using client authentication, *HostSpec* specifies the certificate name and key password, rather than the bindDN and password. Note: If no `-c` option is specified, the `-s` *HostSpec* may refer to any server, either a consumer or a supplier.
-D	The distinguished name with which to bind to the server. This parameter is optional if the server is configured to support anonymous access. If a DN is specified in the ServerSpec, this overrides the `-D` option.
-w	The password associated with the distinguished name specified by the -D option. If a password is specified in the ServerSpec, this overrides the `-w` option.
-n	Specifies that the tools should not run in interactive mode. Running in interactive mode allows you to re-enter the bindDN, password and host and port, if the tool encounters a bind error.
-p	The TCP port used by the Directory Server. The default port is 389. If a port is specified in the ServerSpec, this overrides the `-p` option.
-j	If specifying the default password at the command line poses a security risk, the password can be stored in a file. The `-j` option specifies this file.

CAUTION When identifying hosts, you must use either symbolic names or IP addresses for *all* hosts. Using a combination of the two can cause problems.

SSL Options

You can use the following options to specify that the replication monitoring tools use LDAPS when communicating with the Directory Server. You also use these options if you want to use certificate-based authentication. These options are valid only when LDAPS has been turned on and configured. For more information on certificate-based authentication and how to create a certificate database for use with LDAP clients, see Chapter 11, "Managing SSL" in the *Sun ONE Directory Server Administration Guide*.

You must specify the Directory Server's encrypted port when you use the SSL options:

Table 1-5 Common SSL Options

Option	Meaning
`-e`	The default SSL port.
`-J`	This option has the same function as the -j option, for the key password.
`-K`	Specifies the location of the key database used for certificate-based client authentication.
`-N`	Specifies the certificate name to use for certificate-based client authentication. For example, `-N Server-Cert`. If this option is specified, the `-W` option is required.
`-P`	Specifies the location of the certificate database.
`-W`	Specifies the password for the certificate database identified by the `-P` option. For example, `-W serverpassword`.

CAUTION When running the replication monitoring tools over SSL, the directory server on which you are running the tools must have a copy of all the certificates used by the other servers in the topology.

insync

The insync tool indicates the synchronization state between a master replica and one or more consumer replicas. insync compares the RUVs of replicas and displays the time difference or delay (in seconds) between the servers.

Syntax

insync [-D *binddn*] [-w *password*] [-n] [-d] [-t] [-p *port*] [-e *SSL port*] [-j *file*] [-J *file*] [-W *keypasswd*] [-K *keydbpath*] [-N *certname*] [-P *certdbpath*] [-b *ReplicaRoot*] *ServerSpec* [interval]

Note that the *ServerSpec* option includes the -s and -c options, and is described in Table 1-4 on page 37.

Options

In addition to the Common Replication Monitoring Tool Options and Common SSL Options described previously, insync takes the following options:

Table 1-6 insync Specific Options

Option	Meaning
-d	Prints the date of the last change recorded on the master. Using the -d option twice (-d -d) prints the time difference (in days, minutes, and seconds) between the time of the last change and the current time.
-b	The suffix (replica root) that has been specified for replication. If -b is not specified, the delay for all suffixes is printed.
interval	The amount of time (in seconds) after which the synchronization query will start again (in an infinite loop). If no interval is specified, the synchronization query will run only once.
-t	Prints the mode of transport (SSL or CLEAR).

NOTE If a delay of -1 is returned, insync was unable to obtain any replication information. This may indicate that a total update has just been run, that no changes have been sent to the supplier, or that the Replication Agreement is disabled. The corresponding warning is output in each of these cases.

Examples

1. Specifying one supplier, one consumer, and a repetition interval of 30 seconds. Note that the delay changes to 2, indicating that the consumer is 2 seconds behind the supplier at this point.

```
# insync -s "cn=directory manager:password@portugal:1389" -c
"cn=directory manager:password@france:2389" 30

ReplicaDn               Consumer     Supplier       Delay
l=Europe,o=example.com  france:2389  portugal:1389  0
l=States,o=example.com  france:2389  portugal:1389  0
l=Europe,o=example.com  france:2389  portugal:1389  2
l=States,o=example.com  france:2389  portugal:1389  2
l=Europe,o=example.com  france:2389  portugal:1389  0
l=States,o=example.com  france:2389  portugal:1389  0
```

2. Requesting the date of the last change and restricting the output data to the DN `o=example.com`:

```
# insync -s "cn=directory manager:password@portugal:1389" -b o=example.com -d

ReplicaDn               Consumer     Supplier       Delay  Last Update
l=Europe,o=example.com  france:2389  portugal:1389  0      05/12/2002 16:05:08
l=States,o=example.com  france:2389  portugal:1389  0      05/12/2002 16:05:08
```

3. Using certificate-based authentication

```
# insync -n -K /ServerRoot/alias/slapd-S1-key3.db
-P /ServerRoot/alias/slapd-S1-cert7.db -W password -N
"MyCertificate" -S "portugal:24211" -C "france:24213"
```

For more information, see the `insync(1M)` manual page.

entrycmp

The `entrycmp` tool compares the same entry on two or more different servers. An entry is retrieved from the master and the entry's `nsuniqueid` is used to retrieve the same entry from a specified consumer. All the attributes and values of the two entries are compared. If they are identical, the entries are considered to be the same.

Syntax

`entrycmp [-D` *binddn*`] [-w` *password*`] [-n] [-p` *port*`] [-e` *SSL port*`] [-j` *file*`] [-J` *file*`]`
`[-W` *keypasswd*`] [-K` *keydbpath*`] [-N` *certname*`] [-P` *certdbpath*`]` *ServerSpec entry DN*

Note that the *ServerSpec* option includes the `-s` and `-c` options, and is described in Table 1-4 on page 37.

Options

In addition to the Common Replication Monitoring Tool Options and Common SSL Options described previously, `entrycmp` takes the following option:

Table 1-7 entrycmp Specific Options

Option	Meaning
entry DN	Specifies the DN of the entry that you wish to compare.

Examples

1. Basic example

```
# entrycmp -s "cn=directory manager:password@portugal:1389" -c
"cn=directory manager:password@france:2389"
"ou=people,dc=example,dc=com"

entrycmp: france:2389 - entries match
```

2. SSL example

```
# entrycmp -n -K /ServerRoot/alias/slapd-S1-key3.db
-P /ServerRoot/alias/slapd-S1-cert7.db -W password -N "MyCertificate"
-S "portugal:24211" -C "france:24213" "ou=people,dc=example,dc=com"
```

For more information, see the `entrycmp(1M)` manual page.

NOTE Operational attributes are not taken into account when comparing entries.

repldisc

The `repldisc` utility enables you to "discover" a replication topology. Topology discovery starts with one server and constructs a graph of all known servers (using the RUVs and Replication Agreements). `repldisc` then prints an adjacency matrix describing the topology.

Note that the *HostSpec* option includes the `-c` option, and is described in Table 1-4 on page 37.

Syntax

`repldisc [-D` *binddn*`] [-w` *password*`] [-n] [-a] [-t] [-p` *port*`] [-e` *SSL port*`] [-j` *file*`] [-J` *file*`] [-W` *keypasswd*`] [-K` *keydbpath*`] [-N` *certname*`] [-P` *certdbpath*`] [-b` *ReplicaRoot*`] -s/-S` *HostSpec*

NOTE `repldisc` takes the host specification from the replication agreement, unless otherwise specified at the command line.

Options

In addition to the Common Replication Monitoring Tool Options and Common SSL Options described previously, `repldisc` takes the following options:

Table 1-8 repldisc Specific Options

Option	Meaning
-a	Specifies that only the arcs between pairs of connected hosts are printed. For more information, see the examples that follow. Note: If the total line length of the output exceeds 80 characters, symbolic host names are used, accompanied by a legend. Otherwise, the full host name is printed. Using the `-a` option ensures that symbolic host names are not used.
-b	The suffix (replica root) that has been specified for replication. If `-b` is not specified, the topology for all suffixes is printed.
-t	If used with the `-a` option, this option prints the mode of transport (`SSL` or `CLEAR`).

Examples

1. `repldisc` output in a single master replication scenario.

```
# repldisc -D "cn=directory manager" -w mypword -b o=rtest -s
myserver:1389
```

```
Topology for suffix: o=rtest

Legend:
^ : Host on row sends to host on column.
v : Host on row receives from host on column.
x : Host on row and host on column are in MM mode.
H1 : france.example.com:1389
H2 : spain:1389
H3 : portugal:389


   | H1 | H2 | H3 |
===+===============
H1 |    | ^  |    |
---+---------------
H2 | v  |    | ^  |
---+---------------
H3 |    | v  |    |
---+---------------
```

2. The same example as above, but using the `-a` and `-t` options.

```
# repldisc -D "cn=directory manager" -w mypword -b o=rtest -s
myserver:1389 -a -t

Topology for suffix: o=rtest

Legend:
The direction of the replication is indicated with arrows.
Single-master: suppliers appear on left, consumers on right (->).
Multi-master : servers are shown linked by a double arrow (<->).

france.example.com:1389 -> spain:1389 CLEAR
spain:1389 -> portugal:389 CLEAR
```

3. SSL example

```
# repldisc -n -K /ServerRoot/alias/slapd-S1-key3.db
  -P /ServerRoot/alias/slapd-S1-cert7.db -W password -N
  "MyCertificate" -S "portugal:24211" -a -t

Topology for suffix: o=rtest

Legend:
The direction of the replication is indicated with arrows.
Single-master: suppliers appear on left, consumers on right (->).
Multi-master : servers are shown linked by a double arrow (<->).
```

```
spain:24210 -> portugal:24211 SSL
```

For more information, see the `repldisc(1M)` manual page.

Other Tools

pwdhash

The `pwdhash` command prints the encrypted form of a password using one of the server's encryption algorithms. If a user cannot log in, you can use this command to compare the user's password to the password stored in the directory.

Syntax

`pwdhash -D instance_dir [-H] [-c` *`comparepwd`* `| -s` *`scheme`*`] password...`

Options

`pwdhash` takes the following options:

Table 1-9 `pwdhash` Options

Option	Meaning
`-c`	Specifies the encrypted password to be compared with. The result of the comparison is either `OK` or `doesn't match`.
`-s`	Generates the encrypted passwords according to the scheme's algorithm. The available schemes are `SSHA`, `SHA`, `CRYPT` and `CLEAR`.
`-D`	The instance directory.
`-H`	Specifies that the passwords are hex-encoded.
password	The clear password/s from which the encrypted form should be generated (or against which the password in the directory should be compared).

Example

```
# pwdhash -D serverRoot/slapd-serverID -s SSHA myPassword
# {SSHA}mtHyZSHfhOZ4FHmvQe09FQjvLZpnW1wbmw05cw==
```

Chapter 2

Command-Line Scripts

This chapter provides information on the scripts you can use to back up and restore your database. These scripts are a shortcut to executing the `ns-slapd` interface commands, documented in Appendix B, “ns-slapd and slapd.exe Command-Line Utilities.”

This chapter is divided into the following sections:

- Command-Line Scripts Quick Reference
- Shell and Batch Scripts
- Perl Scripts

Command-Line Scripts Quick Reference

All scripts and commands, with the exception of `admin_ip.pl` and `schema_push.pl`, can be accessed in the following way:

Solaris Packages	Using the `/usr/sbin/directoryserver` *name* command
Other platforms	As the */ServerRoot*`/slapd-`*serverID*`/`*name* script or `.bat` file

The `admin_ip.pl` script is located in the */ServerRoot*`/shared/bin` folder. The `schema_push.pl` script is located in */ServerRoot*`/slapd-`*serverID*`/` and must be run from this location on all platforms.

Refer to the tables below for the name and purpose of each script.

When scripts request either a directory name or a file name, always provide the absolute path. The scripts assume that you want to use the `dse.ldif` file located in:

/ServerRoot`/slapd-`*serverID*`/config`

CAUTION To execute the Perl Scripts, you must change to the directory in which the command-line utilities are stored. Although it is possible to set command-path and library-path variables to execute these scripts, this is *not* the recommended procedure. You run the risk of disrupting the correct execution of other scripts and utilities and of compromising the security of the system, particularly when you have more than one server version installed.

The same procedure applies to the `ldapsearch`, `ldapmodify`, `ldapdelete`, `ldapcompare` and `ldif` command-line utilities. For more information on these command-line utilities, see the *Directory Server Resource Kit Tools Reference*.

Note also that when you are running the Perl scripts on Windows machines, the path environment variable must contain the Perl executable (`perl.exe`) file. You must therefore run the scripts from the following directory on Windows:

`\`*ServerRoot*`\bin\slapd\admin\bin`

The following table lists command-line script *names*, which are also option *names* for the `/usr/sbin/directoryserver` command (Solaris packages.)

Table 2-1 Commonly Used Command-Line Shell and Batch Scripts

Command Name	Description
`bak2db`	Restores the database from the most recent archived backup.
`db2bak`	Creates a backup of the current database contents.
`db2ldif`	Exports the contents of the database to LDIF.
`getpwenc`	Prints the encrypted form of a password using one of the server's encryption algorithms. If a user cannot log in, you can use this script to compare the user's password to the password stored in the directory. This command name is not an option for the `directoryserver` command for Solaris packages.
`ldif2db`	Imports LDIF files to the database.
	Runs the `slapd` (Windows) or `ns-slapd` (UNIX) command-line utility with the `ldif2db` keyword. By default, the script first saves and then merges any existing configuration tree (`o=NetscapeRoot`) with any files to be imported.

Table 2-1 Commonly Used Command-Line Shell and Batch Scripts *(Continued)*

Command Name	Description
`ldif2ldap`	Performs an import operation over LDAP to the Directory Server.
`monitor`	Retrieves performance monitoring information using the `ldapsearch` command-line utility.
`restart-slapd`	Restarts Directory Server. Use the following command on the Solaris 9 platform: `/usr/sbin/directoryserver restart`.
`restoreconfig`	Restores by default the most recently saved Admin Server configuration to NetscapeRoot partition.
`saveconfig`	Saves Admin Server configuration, stored in the NetscapeRoot suffix, to *ServerRoot* / slapd-*serverID* / confbak
`start-slapd`	Starts Directory Server. Use the following command on the Solaris 9 platform: `/usr/sbin/directoryserver start`.
`stop-slapd`	Stops Directory Server. Use the following command on the Solaris 9 platform: `/usr/sbin/directoryserver stop`.
`suffix2instance`	Maps a suffix to a backend name.
`vlvindex`	Creates and generates virtual list view (VLV) indexes, also called browsing indexes.

The following table gives the names of Perl scripts and the equivalent command for Solaris packages.

Table 2-2 Commonly Used Command-Line Perl Scripts

Perl Script	Description
`admin_ip.pl`	Changes the IP address in the local Administration Server configuration file and in the configuration directory.
`bak2db.pl` `directoryserver bak2db-task`	Restores the database from the most recent archived backup.
`db2bak.pl` `directoryserver db2bak-task`	Creates a backup of the current database contents
`db2index.pl` `directoryserver db2index-task`	Creates and regenerates indexes for attributes that are present in the database configuration as index attributes.
`db2ldif.pl` `directoryserver db2ldif-task`	Exports the contents of the database to LDIF.

Table 2-2 Commonly Used Command-Line Perl Scripts

Perl Script	Description
`ldif2db.pl` `directoryserver ldif2db-task`	Imports LDIF files to database. Runs the `slapd` (Windows) or `ns-slapd` (UNIX) command-line utility with the `ldif2db` keyword. By default, the script first saves and then merges any existing configuration tree (`o=NetscapeRoot`), with any files to be imported.
`migrateInstance5` (not available on the Solaris 9 platform)	Migrates a 4.x or 5.0 version of the Directory Server to the 5.2 version, converting the configuration files to LDIF format. Located in: */ServerRoot*`/bin/slapd/admin/bin`
`ns-accountstatus.pl` `directoryserver account-status`	Provides account status information to establish whether an entry or group of entries is locked or not.
`ns-activate.pl` `directoryserver account-activate`	Activates an entry or a group of entries by unlocking it (them).
`ns-inactivate.pl` `directoryserver account-inactivate`	Inactivates an entry or a group of entries.
`schema_push.pl`	Updates the timestamp used by replication to ensure that manual schema modifications are replicated. Located in: */ServerRoot*`/slapd-`*ServerID*`/`

Shell and Batch Scripts

Some of the shell and batch scripts can be executed while the server is running. Others require that the server is stopped. The description of each script below indicates whether the server must be stopped, or if it can continue to run while you execute the script. When a Shell or Batch script has a Perl equivalent, a cross-reference to the section describing the equivalent Perl script is provided.

bak2db (Restore Database From Backup)

Restores the database from the most recent archived backup. To run this script the server must be stopped.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver bak2db` *backup_directory*
Other platforms	`bak2db` *backup_directory*

For information on the equivalent Perl script, see “bak2db.pl (Restore Database From Backup),” on page 61. For more information on restoring databases, see Chapter 3, “Populating Directory Contents” in the *Sun ONE Directory Server Administration Guide*.

db2bak (Create Backup of Database)

Creates a backup of the current database contents. This script can be executed while the server is running.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver db2bak [`*backup_directory*`]`
Other platforms	`db2bak [`*backup_directory*`]`

For information on the equivalent Perl script, refer to “db2bak.pl (Create Backup of Database),” on page 62.

db2ldif (Export Database Contents to LDIF)

Exports the contents of the database to LDIF. This script can be executed while the server is still running.

For information on the equivalent Perl script, refer to “db2ldif.pl (Export Database Contents to LDIF),” on page 64.

For the shell and batch scripts, the script runs the `slapd` (Windows) or `ns-slapd` (UNIX) command-line utility with the `ldif2db` keyword.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver db2ldif` *options*
Other platforms	`db2ldif` *options*
options	`{-n` *backend_instance*`}* | {-s` *includesuffix*`}* [{-x` *excludesuffix*`}*] [-r] [-C] [-u] [-U] [-m] [-M] [-a` *outputfile*`] [-1] [-N]`

Options

Option	Meaning
`-a`	File name of the output LDIF file.
`-n`	Instance to be exported.
`-s`	Suffix(es) to be included. If used in conjunction with the `-n` option, this option specifies the subtree(s) to be included.
`-x`	Suffix(es) to be excluded.
`-m`	Minimal base64 encoding.
`-M`	Use of several files for storing the output LDIF, with each *instance* stored in *instance_outfile* (where *outfile* is the file name specified for `-a` option).
`-r`	Export replica.
`-u`	Request that the unique id is not exported.

`-C`	Only the main db file is used.
`-N`	Specifies that entry IDs are not to be included in the LDIF output. The entry IDs are necessary only if the `db2ldif` output is to be used as input to `db2index`.
`-U`	Request that the output LDIF is not folded.
`-1`	For reasons of backward compatibility, delete the first line of the LDIF file, that gives the version of the LDIF standard.

NOTES

`db2ldif -r` cannot be used if another `slapd` process is running, because replication writes the RUV entry into the database during export. To export the database while a `slapd` process is running, use `db2ldif.pl -r` instead.

You must specify either the `-n` or the `-s` option (or both).

The output LDIF will be stored in one file by default. Should you want to specify the use of several files, then use the option `-M`.

getpwenc (Print Encrypted Password)

Prints the encrypted form of a password using one of the server's encryption algorithms. If a user cannot log in, you can use this script to compare the user's password to the password stored in the directory.

Syntax

Platform	Syntax
Solaris Packages	*/serverRoot/*`slapd-`*serverID*`/getpwenc` *storagescheme clearpassword*
Other platforms	`getpwenc` *storagescheme clearpassword*

Options

There are no options for this script.

For more information on the different storage schemes such as `SSHA`, `SHA`, `CRYPT`, and `CLEAR`, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

ldif2db (Import)

Runs the `slapd` (Windows) or `ns-slapd` (UNIX) command-line utility with the `ldif2db` keyword. To run this script the server must be stopped.

For information on the equivalent Perl script, see "ldif2db.pl (Import)," on page 65.

NOTES

1. `ldif2db` supports LDIF version 1 specifications. You can load an attribute using the `:<` URL specifier notation. For example:

   ```
   jpegphoto:< file:///tmp/myphoto.jpg
   ```

   Although the official notation requires three `///`, the use of one `/` is tolerated. For more information on the LDIF format, see Appendix E, "LDAP Data Interchange Format".

2. The default behavior of a read-write replica that has been initialized either online or offline from a backup or an LDIF file, is NOT to accept client update requests. The replica will remain in read-only mode and refer any updated operations to other suppliers in the topology until the administrator does one of the following:

- changes the duration of the read-only mode default period using the `ds5referralDelayAfterInit` attribute
- manually resets the server to read-write mode using the `ds5BeginReplicaAcceptUpdates` attribute (once the replica has completely converged with the other suppliers in the topology)

The second option is advised because it does not present non-convergence risks. For more information, refer to Chapter 8, "Managing Replication" in the *Sun ONE Directory Server Administration Guide*.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver ldif2db` *options*
Other platforms	`ldif2db` *options*
options	`-n` *backend_instance* `|` `{-s` *includesuffix*`}*` `[{-x` *excludesuffix*`}*] {-i` *ldif-file*`}* [-O] [-Y` *keydb-pwd*`] [-y` *keydb-pwd-file*`]`

Options

Option	Meaning
`-n`	Instance to be imported. Ensure that you specify an instance that corresponds to the suffix contained by the LDIF file. Otherwise the data contained by the database is deleted and the import fails.
`-s`	Suffix(es) to be included. If used in conjunction with the `-n` option, this option specifies the subtree(s) to be included.
`-i`	File name of the input ldif file(s). When you import multiple files, they are imported in the order in which you specify them on the command line.
`-x`	Suffix(es) to be included.
`-O`	Request that only the core db is created without attribute indexes.
`-Y`	Specifies the password for the key database (used for certificate-based client authentication).
`-y`	Specifies the file in which the password for the key database is held (used for certificate-based client authentication).

NOTE You must specify either the `-n` or the `-s` option (or both).

ldif2ldap (Perform Import Operation Over LDAP)

Performs an import operation over LDAP to the Directory Server. To run this script the server must be running.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver ldif2ldap -D` *rootDN* `-w` *password* `-f` *filename*
Other platforms	`ldif2ldap -D` *rootDN* `-w` *password* `-f` *filename*

Options

Option	Meaning
`-D`	User DN with root permissions, such as Directory Manager.
`-w`	Password associated with the user DN.
`-f`	File name of the file to be imported. When you import multiple files, they are imported in the order in which you specify them on the command line.

monitor (Retrieve Monitoring Information)

Retrieves performance monitoring information using the `ldapsearch` command-line utility.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver monitor`
Other platforms	`monitor`

Options

There are no options for this script.

For more information on the `ldapsearch` command-line utility, see Chapter 1, “Command-Line Utilities.”

restart-slapd (Restart Directory Server)

Restarts Directory Server.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver restart`
Other platforms	`restart-slapd`

Options

There are no options for this script.

Exit Status

0: Server restarted successfully.

1: Server could not be started.

2: Server restarted successfully but was already stopped.

3: Server could not be stopped.

restoreconfig (Restore Administration Server Configuration)

By default, restores the most recently saved Administration Server configuration information to the `NetscapeRoot` suffix under the following directory:

/ServerRoot`/slapd-`*serverID*`/config`

To restore the Administration Server configuration:

1. Stop Directory Server
2. Run the `restoreconfig` script
3. Restart Directory Server

4. Restart the Administration Server for the changes to be taken into account.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver restoreconfig`
Other platforms	`restoreconfig`

Options

There are no options for this script.

saveconfig (Save Administration Server Configuration)

Saves the Administration Server configuration information to the following directory

/ServerRoot/slapd-*serverID*/confbak

:This script will run only if the server is running.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver saveconfig`
Other platforms	`saveconfig`

Options

There are no options for this script.

start-slapd (Start Directory Server)

Starts Directory Server.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver start`
Other platforms	`start-slapd`

Options

There are no options for this script.

Exit Status

0: Server started successfully.

1: Server could not be started.

2: Server was already started.

stop-slapd (Stop Directory Server)

Stops Directory Server.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver stop`
Other platforms	`stop-slapd`

Options

There are no options for this script.

Exit Status

0: Server stopped successfully.

1: Server could not be stopped.

2: Server was already stopped.

suffix2instance (Map Suffix to Backend Name)

Maps a suffix to a backend name.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver suffix2instance {-s` *suffix*`}`
Other platforms	`suffix2instance {-s` *suffix*`}`

Options

Option	Meaning
`-s`	The suffix to be mapped to the backend.

vlvindex (Create Virtual List View (VLV) Indexes)

To run the `vlvindex` script, the server must be stopped. The `vlvindex` script creates virtual list view (VLV) indexes, known in the Directory Server console as Browsing Indexes. VLV indexes introduce flexibility in the way you view search results. Using VLV indexes, you can organize search results alphabetically or in reverse alphabetical order, and you can scroll through the list of results. VLV index configuration must already exist prior to running this script.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver vlvindex` *options*
Other platforms	`vlvindex` *options*
options	`[-d` *debug_level*`] [-n` *backend_instance*`] [-s` *suffix*`] [-T` *VLVTag*`]`

Options

Option	Meaning
-d	Specifies the debug level to use during index creation. Debug levels are defined in “nsslapd-errorlog-level (Error Log Level),” on page 104.
-n	Name of the database containing the entries to index.
-s	Name of the suffix containing the entries to index.
-T	VLV index identifier to use to create VLV indexes. You can use the console to specify VLV index identifier for each database supporting your directory tree, as described in the *Sun ONE Directory Server Administration Guide*. You can define additional VLV tags by creating them in LDIF, and adding them to Directory Server’s configuration, as described in the *Sun ONE Directory Server Administration Guide*. In any case, we recommend you use the dn of the entry for which you want to accelerate the search sorting.

NOTE You must specify either the -n or the -s option.

Perl Scripts

admin_ip.pl (Change IP Address)

When your system's IP address changes, you must update the local Administration Server configuration file and the configuration directory. If you do not enter the new IP address in these locations, you will not be able to start the Administration Server.

A Perl script is provided to help you update these two configurations. The script changes the IP address for an instance of Administration Server in both the `local.conf` file and the configuration directory. The script is called `admin_ip.pl` and is located in the `serverRoot/shared/bin` folder.

Usage

To run `admin_ip.pl`, follow the instructions for UNIX or Windows systems as appropriate:

On UNIX Systems

In the *serverRoot*`/shared/bin` folder, type the following

`admin_ip.pl` *Directory_Manager_DN Directory_Manager_password old_IP new_IP* [*port*]

The old IP address is saved in a file called `local.conf.old`.

On Windows

From the command line go to the *serverRoot*`/shared/bin` folder and type the following:

`../../install/perl admin_ip.pl` *Directory_Manager_DN Directory_Manager_password old_IP new_IP* [*port*]

The old IP address is saved in a file called `local.conf.old`.

bak2db.pl (Restore Database From Backup)

The perl script `bak2db.pl` creates an entry in the directory that launches this dynamic task. An entry is generated based upon the values you provide for each option.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver bak2db-task` *options*
Other platforms	`bak2db.pl` *options*
options	`[-v] -D` *rootDN* `{-w` *password* `| -w - | -j` *filename* `}` `-a` *backup_directory* `[-t` *databasetype*`]`

Options

Option	Meaning
`-D`	User DN with root permissions, such as Directory Manager. The default is the DN of the directory manager, which is read from the `nsslapd-root` attribute under `cn=config`.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the -w option must not be specified.
`-a`	Directory of the backup files.
`-v`	Verbose mode.
`-t`	Database type. Currently, `ldbm` is the only possible type and the default value.

db2bak.pl (Create Backup of Database)

The perl script `db2bak.pl` creates an entry in the directory that launches this dynamic task. An entry is generated based upon the values you provide for each option.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver db2bak-task` *options*
Other platforms	`db2bak.pl` *options*
options	`[-v] -D` *rootDN* `{-w` *password* `| -w - | -j` *filename* `}` `-a` *backup_directory* `[-t` *databasetype*`]`

Options

Option	Meaning
`-D`	User DN with root permissions, such as Directory Manager. The default is the DN of the directory manager, which is read from the `nsslapd-root` attribute under `cn=config`.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the -w option must not be specified.
`-a`	Directory where the backup files will be stored. By default it is under *ServerRoot*`/slapd-`*serverID*`/bak`. The backup file is named according to the year-month-day-hour format (`YYYY_MM_DD_hh_mm_ss`).
`-v`	Verbose mode.
`-t`	Database type. Currently, `ldbm` is the only possible type and the default value.

db2index.pl (Create and Generate Indexes)

Creates and generates the new set of indexes to be maintained following the modification of indexing entries in the `cn=config` configuration file. Note that indexes are generated only for those attributes that are present in the database configuration as index attributes.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver db2index-task` *options*
Other platforms	`db2bindex.pl` *options*
options	`[-v] -D` *rootDN* `{-w` *password* `| -w - | -j` *filename* `}` `-n` *backend_instance* `[-t` *attributeName*`]`

Options

Option	Meaning
`-D`	User DN with root permissions, such as Directory Manager.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the -w option must not be specified.
`-n`	Instance to be indexed.
`-t`	Name of the attribute to be indexed. If omitted, all indexes defined for that instance are generated.
`-v`	Verbose mode.

NOTE This perl script `db2index.pl` creates an entry in the directory that launches this dynamic task. An entry is generated based upon the values you provide for each option.

db2ldif.pl (Export Database Contents to LDIF)

Exports the contents of the database to LDIF. This Perl script creates an entry in the directory that launches this dynamic task. The entry is generated based upon the values you provide for each option. The * indicates that multiple occurrences are allowed.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver db2ldif-task` *options*
Other platforms	`db2ldif.pl` *options*
options	`[-v] -D` *rootDN* `{-w` *password* `| -w - | -j` *filename* `}` `{-n` *backend_instance*`}* |` `{-s` *includesuffix*`}* [{-x` *excludesuffix*`}*]` `[-a` *outfile*`] [-N] [-r] [-C] [-u] [-U] [-m] [-o] [-1] [M]`

CAUTION To run this script the server must be running and either `-n` *backend_instance* or `-s` *includesuffix* is required.

This perl script `bak2db.pl` creates an entry in the directory that launches this dynamic task. An entry is generated based upon the values you provide for each option.

Options

Option	Meaning
`-D`	User DN with root permissions, such as Directory Manager.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the `-w` option must not be specified.
`-n`	Instance to be exported.

`-s`	Suffix(es) to be included. If used in conjunction with the `-n` option, this option specifies the subtree(s) to be included.
`-a`	File name of the output LDIF file.
`-x`	Suffix(es) to be excluded.
`-m`	Minimal base64 encoding.
`-o`	Output LDIF to be stored in one file by default with each *instance* stored in *instance_outfile.*
`-r`	Export replica.
`-u`	Request that the unique id is not exported.
`-C`	Only the main db file is used.
`-N`	Suppress printing sequential number.
`-U`	Request that the output LDIF is not folded.
`-v`	Verbose mode.
`-1`	For the purposes of backward compatibility, delete the first line of the LDIF file that gives the version of the LDIF standard.
`-M`	Output LDIF is stored in multiple files.

ldif2db.pl (Import)

To run this Perl script, the server must be running. This script creates an entry in the directory that launches this dynamic task. The entry is generated based upon the values you provide for each option.

Syntax

Platform	**Syntax**
Solaris Packages	`directoryserver ldif2db` *options*
Other platforms	`ldif2db.pl` *options*
options	`[-v] -D` *rootDN* `{-w` *password* `| -w - | -j` *filename* `} -n` *backend_instance* `| {-s` *includesuffix*`}* [{-x` *excludesuffix*`}*] [-O] [-c] [-g` *string*`] [-G` *namespace_id*`] {-i` *filename*`}*`

Options

Option	Meaning
`-D`	User DN with root permissions, such as Directory Manager.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the `-w` option must not be specified.
`-n`	Instance to be imported.
`-s`	Suffix(es) to be included. If used in conjunction with the `-n` option, this option specifies the subtree(s) to be included.
`-i`	File name of the input LDIF file(s). When you import multiple files, they are imported in the order in which you specify them on the command line.
`-x`	Suffix(es) to be excluded.
`-O`	Request that only the core database is created without attribute indexes.
`-c`	Merge chunk size.
`-g` *string*	Generation of a unique ID. Type `none` for no unique ID to be generated and `deterministic` for the generated unique ID to be name-based. By default a time based unique ID is generated. If you use the `deterministic` generation to have a name-based unique ID, you can also specify the namespace you want the server to use as follows: `-g deterministic namespace_id` where `namespace_id` is a string of characters in the following format 00-xxxxxxxx-xxxxxxxx-xxxxxxxx-xxxxxxxx Use this option if you want to import the same LDIF file into two different directory servers, and if you want the contents of both directories to have the same set of unique IDs. If unique IDs already exist in the LDIF file you are importing, then the existing IDs are imported to the server regardless of the options you have specified.
`-G` *namespace_id*	Generates a namespace ID as a name-based unique ID. This is the same as specifying the `-g deterministic` option.
`-v`	Verbose mode.

migrateInstance5 (Migrate to Directory Server 5.x)

The `migrateInstance5` Perl script (note that this is a Perl script despite the fact that it does not have the `.pl` extension) migrates a 4.x Directory Server to Directory Server 5.x. It can also be used to upgrade from Directory Server 5.0 or 5.1 to Directory Server 5.2.

When you run this script, it migrates the configuration files or configuration entries, database instances and schema with minimum manual intervention. The `migrateInstance5` script calls on the `migrateTo5` script, which then executes the migration.

For complete information on the configuration parameters and attributes that are migrated, see Chapter 6, “Migration From Earlier Versions.”

Before performing the migration, check that the user-defined variables contain the following associated values, where *ServerRoot* is the path to where Sun ONE Directory Server 5.2 is installed:

$PERL5LIB	*ServerRoot*`/bin/slapd/admin/bin`
PATH	*ServerRoot*`/bin/slapd/admin/bin`

Syntax

`migrateInstance5 -D` *rootDN* `{-w` *password* `| -w - | -j` *filename* `}`
`-n` *backend_instance* `-p` *port* `-o` *4.xInstancePath* `-n` *5.xInstancePath* `[-t] [-L]`

Options

Option	Meaning
`-D`	Directory Server 5.2 userDN with root permissions, such as Directory Manager.
`-w`	Password associated with the Directory Server 5.2 user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the `-w` option must not be specified.
`-p`	Directory Server 5.2 port.

`-o` *4.xInstancePath*	Netscape Directory Server 4.x.
`-n` *5.0InstancePath*	Path for the new Directory Server 5.0 instance.
`-t`	Trace level. The trace level is set to 0 by default with a valid range of 0 to 3.
-L	File in which to log the migration report. By default the migration report is stored under *ServerRoot*`/slapd-`*serverID*`/logs/Migration_`*ddmmyyy_hhmmss*`.log` A sample log might contain: `/`*ServerRoot*`/slapd-`*serverID*`/logs/Migration_20022003_153604.log` for a log created on 20 February 2003 at 15.36:04.

ns-accountstatus.pl (Establish Account Status)

Provides account status information to establish whether an entry or group of entries is inactivated or not.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver account-status` *options*
Other platforms	`ns-accountstatus.pl` *options*
options	`[-D` *rootDN*`] {-w` *password* `| -w -	-j` *filename* `}` `[-h` *host*`] [-p` *port*`] -I` *DN*

Options

Option	Meaning
`-D`	Directory Server 5.2 userDN with root permissions, such as Directory Manager.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.

`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the `-w` option must not be specified.
`-p`	Directory Server 5.2 port. The default value is the Directory Server LDAP port, specified at installation time.
`-h`	Host name of Directory Server 5.2. The default value is the full host name of the machine on which Directory Server is installed.
`-I` *DN*	Entry DN or role DN whose status is required.

ns-activate.pl (Activate an Entry or Group of Entries)

Activates an entry or group of entries.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver account-activate` *options*
Other platforms	`ns-activate.pl` *options*
options	`[-D` *rootDN*`] {-w` *password* `| -w - | -j` *filename* `}` `[-h` *host*`] [-p` *port*`] -I` *DN*

Options

Option	Meaning
`-D`	Directory Server 5.2 userDN with root permissions, such as Directory Manager.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the `-w` option must not be specified.
`-p`	Directory Server 5.2 port. The default value is the Directory Server LDAP port, specified at installation time.

`-h`	Host name of Directory Server 5.2. The default value is the full host name of the machine on which Directory Server is installed.
`-I` *DN*	Entry DN or role DN to activate.

ns-inactivate.pl (Inactivate an Entry or Group of Entries)

Inactivates, and thus locks, an entry or group of entries.

Syntax

Platform	Syntax
Solaris Packages	`directoryserver account-inactivate` *options*
Other platforms	`ns-inactivate.pl` *options*
options	`[-D` *rootDN*`] {-w` *password* `| -w -	-j` *filename* `}` `[-h` *host*`] [-p` *port*`] -I` *DN*

Options

Option	Meaning
`-D`	Directory Server 5.2 userDN with root permissions, such as Directory Manager.
`-w`	Password associated with the user DN. If you do not specify this option, anonymous access is used. If you specify `-w -`, the utility prompts for the password. If either `-w` option is specified, the `-j` option must not be specified. For example, `-w diner892`.
`-j`	Specifies the file from which the bind password is read. Used for simple authentication. If this option is specified, the `-w` option must not be specified.
`-p`	Directory Server 5.2 port. The default value is the Directory Server LDAP port, specified at installation time.
`-h`	Host name of Directory Server 5.2. The default value is the full host name of the machine on which Directory Server is installed.
`-I` *DN*	Entry DN or role DN to inactivate.

schema_push.pl

When schema modifications are made manually (by editing the `.ldif` files directly), this script should be run to update the modification time used by replication. This ensures that the modified schema are replicated to the consumers. Once the script has been run, you must restart the server to trigger the schema replication.

Syntax

/ServerRoot`/slapd-`*serverID*`/schema_push.pl`

Part 2

Server Configuration

This part provides reference information on configuring Directory Server and contains a comprehensive list of the standard configuration attributes and the plug-in configuration attributes. This part also contains migration information. It includes the following chapters:

- Core Server Configuration
- Core Server Configuration Attributes
- Plug-In Implemented Server Functionality
- Migration From Earlier Versions

Chapter 3

Core Server Configuration

The configuration information for Sun ONE Directory Server 5.2 is stored as LDAP entries within the directory itself. Therefore, changes to the server configuration must be implemented through the use of the server rather than by simply editing configuration files. The principal advantage of this method of configuration storage is that it allows a directory administrator to reconfigure the server via LDAP while it is still running, and avoids having to shut it down.

This chapter provides details of how the configuration is organized, and how to alter it. An alphabetical reference for all attributes is provided in Chapter 4, “Core Server Configuration Attributes.

This chapter is divided into the following sections:

- Server Configuration Overview
- Accessing and Modifying Server Configuration

Server Configuration Overview

When you install the Sun ONE Directory Server 5.2, its default configuration is stored as a series of LDAP entries within the directory, under the subtree `cn=config`. When the server is started, the contents of the `cn=config` subtree are read from a file in LDIF format: `dse.ldif`. This `dse.ldif` file contains all of the server configuration information. It is worth noting that the latest version of this file is called `dse.ldif`, the version prior to the last modification is called `dse.ldif.bak`, and the latest file with which the server successfully started is called `dse.ldif.startOK`. Many of the features of Sun ONE Directory Server 5.2 are designed as discrete modules that plug into the core server. The details of the internal configuration for each plug-in are contained in separate entries under `cn=plugins,cn=config`. For example, the configuration of the Telephone Syntax plug-in is contained in the entry:

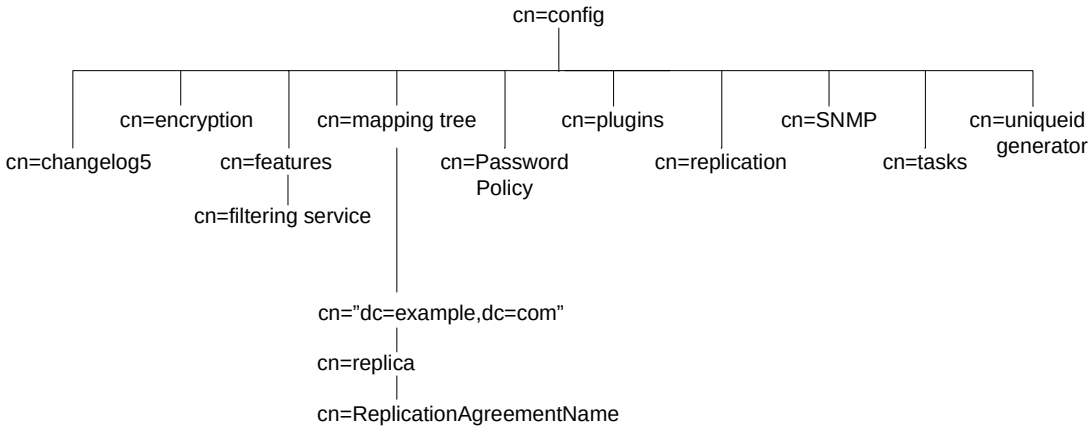
`cn=Telephone Syntax,cn=plugins,cn=config`

Similarly, database-specific configuration is stored under:

`cn=ldbm database,cn=plugins,cn=config and cn=chaining database,cn=plugins,cn=config`

Figure 3-1 shows how the configuration data fits within the `cn=config` Directory Information Tree.

Figure 3-1 Configuration Data Under cn=config



This overview is divided into the following sections:

- LDIF Configuration Files - Location
- Schema Configuration Files - Location
- How the Server Configuration is Organized
- Migration of Pre-Directory Server 5.x Configuration Files to LDIF Format

LDIF Configuration Files - Location

The Directory Server configuration data is automatically output to files in LDIF format that are located in the following directory by default:

ServerRoot`/slapd-`*serverID*`/config`

In this chapter, all examples use `myServer` for the server identifier where appropriate.

Schema Configuration Files - Location

Schema configuration is also stored in LDIF format and these files are located in the following directory:

ServerRoot`/slapd-`*serverID*`/config/schema`

For a full list of the LDIF configuration files that are supplied with Directory Server, see Table 4-7 on page 185.

How the Server Configuration is Organized

The `dse.ldif` file contains all configuration information including directory specific entries created by the directory at server startup, and directory specific entries related to the database, also created by the directory at server startup. The file includes the Root DSE (named by `""`) and the entire contents of `cn=config`. When the server generates the `dse.ldif` file, it lists the entries in hierarchical order. It does so in the order that the entries appear in the directory under `cn=config`.

This section provides an overview of configuration attributes, plug-in functionality configuration, database configuration, and index configuration.

Configuration Attributes

Within a configuration entry, each attribute is represented as an attribute name. The value of the attribute corresponds to the attribute's configuration.

The following example shows part of the `dse.ldif` file for a Directory Server and indicates, amongst other things, that schema checking has been turned *on*. This is represented by the attribute `nsslapd-schemacheck`, which takes the value `on`.

Code Example 3-1 Extract of `dse.ldif` File

```
dn: cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsslapdConfig
nsslapd-accesslog-logging-enabled: on
nsslapd-enquote-sup-oc: on
nsslapd-localhost: myServer.example.com
nsslapd-errorlog: ServerRoot/slapd-myServer/logs/errors
nsslapd-schemacheck: on
nsslapd-port: 389
nsslapd-localuser: nobody
...
```

Configuration of Plug-in Functionality

The configuration for each part of Directory Server plug-in functionality has its own separate entry and set of attributes under the subtree `cn=plugins,cn=config`. The following example shows the configuration entry for a plug-in, in this case the Telephone Syntax plug-in.

Code Example 3-2 Configuration Entry for Telephone Syntax Plug-in

```
dn: cn=Telephone Syntax,cn=plugins,cn=config
objectclass: top
objectclass: nsSlapdPlugin
objectclass: ds-signedPlugin
objectclass: extensibleObject
cn: Telephone Syntax
nsslapd-pluginPath: ServerRoot/lib/syntax-plug-in.so
nsslapd-pluginInitfunc: tel_init
nsslapd-pluginType: syntax
nsslapd-pluginEnabled: on
...
```

Some of these attributes are common to all plug-ins and some may be particular to a specific plug-in. You can check which attributes are currently being used by a plug-in by performing an `ldapsearch` on the `cn=config` subtree.

For a list of plug-ins supported by Sun ONE Directory Server 5.2, general plug-in configuration information, the plug-in configuration attribute reference, and a list of plug-ins requiring the server to be restarted see Chapter 5, “Plug-In Implemented Server Functionality.”

Configuration of Databases

The `cn=NetscapeRoot` and `cn=UserRoot` subtrees contain configuration data for the databases containing the `o=NetscapeRoot` and `o=UserRoot` suffixes respectively. The `cn=NetscapeRoot` subtree contains the configuration data used by the Sun ONE Administration Server for authentication and all actions that cannot be performed through LDAP (such as start / stop). The `cn=UserRoot` subtree contains all the configuration data for the first user-defined database created during server installation. The `cn=UserRoot` subtree is called `UserRoot` by default. However, this is not hard-coded, and, given the fact that there will be multiple database instances, this name will be changed and defined by the user when new databases are added.

Configuration of Indexes

Configuration information for indexing is stored as entries in the Directory Server under the three following information tree nodes:

- `cn=index,cn=NetscapeRoot,cn=ldbm database,cn=plugins,cn=config`
- `cn=index,cn=UserRoot,cn=ldbm database,cn=plugins,cn=config`
- `cn=default indexes,cn=config,cn=ldbm database, cn=plugins,cn=config`

For more information regarding indexes in general, see the *Sun ONE Directory Server Administration Guide* and for information regarding the index configuration attributes, see “Default Index Attributes” on page 238. The attributes are presented here because this node is the first to appear in the representation of the configuration attributes based on the `cn=config` information tree.

Migration of Pre-Directory Server 5.x Configuration Files to LDIF Format

Sun ONE Directory Server 5.2 recognizes configuration files that are in LDIF format only, which means that the `slapd.conf` and `slapd.ldbm.conf` configuration files from 4.x versions of Directory Server must be converted to LDIF format. Directory Server 4.x configurations can be migrated to the new LDIF format using the `migrateInstance5` tool. For information on the attributes that are migrated with this tool, see Chapter 6, “Migration From Earlier Versions.”

Accessing and Modifying Server Configuration

This section discusses access control for configuration entries and describes the various ways in which the server configuration can be viewed and modified. It also covers restrictions on the types of modification that can be made and discusses attributes that require the server to be restarted for changes to take effect. This section has been divided into the following parts:

- Access Control for Configuration Entries
- Changing Configuration Attributes

Access Control for Configuration Entries

When the Directory Server is installed, a default set of Access Control Instructions (ACIs) is implemented for all entries under `cn=config`. Code Example 3-3 shows an example of these default ACIs.

Code Example 3-3 Default ACIs in `dse.ldif`

```
aci: (targetattr = "*")(version 3.0; acl "Configuration Adminstrators Group";
 allow (all)
 groupdn = "ldap:///cn=Configuration Administrators,ou=Groups,
 ou=TopologyManagement, o=NetscapeRoot";)
aci: (targetattr = "*")(version 3.0; acl "Configuration  Adminstrators";
 allow (all) userdn =
 "ldap:///uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeRoot";)
aci: (targetattr = "*")(version 3.0; acl "Local Directory Adminstrators Group";
 allow (all)
 groupdn = "ldap:///ou=Directory Administrators, dc=example,dc=com";)
aci: (targetattr = "*")(version 3.0; acl "SIE Group"; allow(all) groupdn =
 "ldap:///cn=slapd-myServer, cn=Netscape Directory Server, cn=Server Group,
 cn=myServer.example.com, dc=example,dc=com, o=NetscapeRoot";)
```

These default ACIs allow all LDAP operations to be carried out on all configuration attributes by the following users:

- Members of the Configuration Administrators Group
- The user acting as the Administrator, who has the uid `admin` that can be configured at installation time
- Members of the local Directory Administrators Group
- The local Directory Administrator (root DN)
- The SIE (Server Instance Entry) Group that is usually assigned using the Set Access Permissions from the main topology view in the main console.

For more information, see Chapter 6, “Managing Access Control” in the *Sun ONE Directory Server Administration Guide*.

Changing Configuration Attributes

You can view and change server attribute values in one of three ways: by using LDAP through Sun ONE Server Console, by performing `ldapsearch` and `ldapmodify` commands, or by manually editing the `dse.ldif` file.

NOTE If you edit the `dse.ldif` file, you must stop the server beforehand, otherwise your changes will be lost. Editing the `dse.ldif` file is recommended only for changes to attributes which cannot be altered dynamically. For further information, see "Configuration Changes Requiring Server Restart," on page 188.

The following sections describe how to modify entries using LDAP (both via the Sun ONE Server Console and over the command line), the restrictions to modifying entries, the restrictions to modifying attributes, and the configuration changes requiring restart.

Modifying Configuration Entries Using LDAP

The configuration entries in the directory can be searched and modified using LDAP, either via the Sun ONE Server Console or by performing `ldapsearch` and `ldapmodify` operations in the same way as other directory entries. The advantage of using LDAP to modify entries is that you can make the changes while the server is running. You must remember to specify the port number when modifying configuration entries as the server is not necessarily running on port 389. For further information see Chapter 2, "Creating Directory Entries" in the *Sun ONE Directory Server Administration Guide*. However, certain changes do require the server to be restarted before they are taken into account. For further information, see "Configuration Changes Requiring Server Restart," on page 188.

NOTE As with any set of configuration files, care should be taken when changing or deleting nodes in the `cn=config` subtree, as this risks affecting Sun ONE Directory Server functionality.

The entire configuration, including attributes that always take default values, can be viewed by performing an `ldapsearch` operation on the `cn=config` subtree:

```
ldapsearch -D bindDN -w password -p port -b cn=config objectclass=*
```

where *bindDN* is the DN chosen for the Directory Manager when the server was installed and *password* is the password chosen for Directory Manager. For more information on using `ldapsearch` see Chapter 1, "Command-Line Utilities."

Previously we saw an example of the configuration entry for the Telephone Syntax plug-in where the plug-in was enabled. If you want to disable this feature you can use the following series of commands to implement this change.

Code Example 3-4 Disabling the Telephone Syntax Plug-in

```
ldapmodify -D bindDN -w password -p port
dn: cn=Telephone Syntax,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: off
```

Restrictions to Modifying Configuration Entries

Certain restrictions apply when modifying server entries:

- The `dse.ldif cn=monitor` entry and its child entries are read-only and cannot be modified.

Restrictions to Modifying Configuration Attributes

Certain restrictions apply when modifying server attributes:

- If an attribute is added to `cn=config`, the server will ignore it.
- If an invalid value is entered for an attribute, the server will ignore it.
- Since `ldapdelete` is used for deleting entire entries, you should use `ldapmodify` if you want to remove an attribute from an entry.

Configuration Changes Requiring Server Restart

Some configuration attributes cannot be altered dynamically while the server is running. In these cases the server needs to be shut down and restarted for the changes to take effect. The modifications should be made either through the Directory Server console or by manually editing the `dse.ldif` file. Table 4-8 under Configuration Quick Reference Tables in the following chapter contains a list of these attributes.

Core Server Configuration Attributes

This chapter provides an alphabetical reference of the attributes used to configure and monitore core server functionality. It is divided into the following sections:

- Core Server Configuration Attributes Reference
- Monitoring Attributes
- Configuration Quick Reference Tables

Core Server Configuration Attributes Reference

This section guides you through all the core server functionality configuration attributes. For server functionality implemented via plug-ins, see the section “Plug-In Implemented Server Functionality,” on page 191. For implementing your own server functionality, contact Sun ONE Professional Services.

For information on where to find the server configuration and how to change it, see “Server Configuration Overview,” on page 75 and “Accessing and Modifying Server Configuration,” on page 80.

The configuration information that is stored in the `dse.ldif` file is organized as an information tree under the general configuration entry `cn=config`. This information tree is illustrated in Figure 3-1 on page 76.

This section describes the configuration tree nodes within this information tree, and is divided into the following subsections:

- cn=config
- cn=changelog5
- cn=encryption
- cn=features

- cn=mapping tree
- cn=Password Policy
- cn=replica
- cn=ReplicationAgreementName
- cn=replication
- cn=SNMP
- cn=tasks
- cn=uniqueid generator

The cn=plugins node is covered in Chapter 5, “Plug-In Implemented Server Functionality.” Attributes are arranged alphabetically and a full description is provided for each, giving the DN of its directory entry, its default value, the valid range of values, and an example of its use.

CAUTION Some of the entries and attributes described in this chapter may change in future releases of the product.

cn=config

General configuration entries are stored under the cn=config entry. The cn=config entry is an instance of the nsslapdConfig object class, which in turn inherits from the extensibleObject object class. For attributes to be taken into account by the server, both of these object classes (in addition to the top object class) must be present in the entry. General configuration entries are presented in this section.

ds-start-tls-enabled (Enable startTLS)

Enables startTLS (Windows installations only). startTLS facilitates dynamic changing to a secured connection. To enable startTLS, security must also be enabled (by setting the nsslapd-security attribute to on).

Because startTLS has a performance impact on Windows installations, it is disabled by default and should only be enabled if required.

Property	Value
Entry DN	cn=config

Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`ds-start-tls-enabled: off`

nsslapd-accesscontrol (Enable Access Control)

Turns access control on and off. If this attribute has a value `off`, any valid bind attempt (including an anonymous bind) results in full access to all information stored in the Directory Server.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	on
Syntax	DirectoryString
Example	`nsslapd-accesscontrol: off`

nsslapd-accesslog (Access Log)

Specifies the path and filename of the log used to record each database access. The following information is recorded in the log file by default:

- IP address of the client machine that accessed the database
- operations performed (for example, search, add, modify)
- result of the access (for example, the number of entries returned)

For more information on turning access logging off, see Chapter 12, “Managing Log Files” in the *Sun ONE Directory Server Administration Guide.*

For access logging to be enabled, this attribute must have a valid path and file name and the `nsslapd-accesslog-logging-enabled` configuration attribute must be switched to on. Table 4-1 lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of access logging.

Table 4-1 Possible Value Combinations of Access Log Attributes

Attribute Pair	Value Pair	Logging Status
`nsslapd-accesslog-logging-enabled` `nsslapd-accesslog`	`on` empty string	Disabled
`nsslapd-accesslog-logging-enabled` `nsslapd-accesslog`	`on` *filename*	Enabled
`nsslapd-accesslog-logging-enabled` `nsslapd-accesslog`	`off` empty string	Disabled
`nsslapd-accesslog-logging-enabled` `nsslapd-accesslog`	`off` *filename*	Disabled

Property	Value
Entry DN	`cn=config`
Valid Range	Any valid filename.
Default Value	*ServerRoot*`/slapd-serverID/logs/access`
Syntax	DirectoryString
Example	`nsslapd-accesslog: /`*ServerRoot*`/slapd-`*serverID*`/logs/access`

nsslapd-accesslog-level

Controls what is logged to the access log.

Property	Value
Entry DN	`cn=config`
Valid Range	0—No access logging 4—Logging for internal access operations 256—Logging for access to an entry 512—Logging for access to an entry and referrals 131072—Precise timing of operation duration. This gives microsecond resolution for the Elapsed Time item in the access log. These values can be added together to provide you with the exact type of logging you require, for example, 516 (4 + 512) to obtain internal access operation, entry access, and referral logging.
Default Value	`256`
Syntax	Integer
Example	`nsslapd-accesslog-level: 256`

nsslapd-accesslog-list

This read-only attribute cannot be set. It provides a list of access log files used in access log rotation.

Property	Value
Entry DN	`cn=config`
Valid Range	N/A
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-accesslog-list:accesslog2,accesslog3`

nsslapd-accesslog-logbuffering (Log Buffering)

When set to `off`, the server writes all access log entries directly to disk.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nsslapd-accesslog-logbuffering: off`

nsslapd-accesslog-logexpirationtime (Access Log Expiration Time)

Specifies the maximum age that a log file is allowed to reach before it is deleted. This attribute supplies only the number of units. The units are provided by the `nsslapd-accesslog-logexpirationtimeunit` attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	`1`
Syntax	Integer

Example	`nsslapd-accesslog-logexpirationtime: 2`

nsslapd-accesslog-logexpirationtimeunit (Access Log Expiration Time Unit)

Specifies the unit for the `nsslapd-accesslog-logexpirationtime` attribute. If the unit is unknown by the server, the log will never expire.

Property	Value
Entry DN	`cn=config`
Valid Range	month | week | day
Default Value	`month`
Syntax	DirectoryString
Example	`nsslapd-accesslog-logexpirationtimeunit: week`

nsslapd-accesslog-logging-enabled (Access Log Enable Logging)

Disables and enables access log logging, but only in conjunction with the `nsslapd-accesslog` attribute that specifies the path and filename of the log used to record each database access.

For access logging to be enabled, this attribute must be switched to on and the `nsslapd-accesslog` configuration attribute must have a valid path and filename. Table 4-1 on page 88 lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of access logging.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nsslapd-accesslog-logging-enabled: off`

nsslapd-accesslog-logmaxdiskspace (Access Log Maximum Disk Space)

Specifies the maximum amount of disk space in megabytes that the access logs are allowed to consume. If this value is exceeded, the oldest access log is deleted.

When setting the maximum disk space, consider the total number of log files that can be created due to log file rotation. Also, remember that there are 3 different log files (access log, audit log, and error log) maintained by the Directory Server, each of which will consume disk space. Compare these considerations to the total amount of disk space that you want to be used by the access log.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647)
Default Value	`500` (A value of -1 means that the disk space allowed to the access log is unlimited in size).
Syntax	Integer
Example	`nsslapd-accesslog-logmaxdiskspace: 200`

nsslapd-accesslog-logminfreediskspace (Access Log Minimum Free Disk Space)

Specifies the minimum allowed free disk space in megabytes. When the amount of free disk space falls below the value specified by this attribute, the oldest access log is deleted until enough disk space is freed to satisfy this attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	`5`
Syntax	Integer
Example	`nsslapd-accesslog-logminfreediskspace: 4`

nsslapd-accesslog-logrotationtime (Access Log Rotation Time)

Specifies the time between access log file rotations. The access log will be rotated when this time interval is up, regardless of the current size of the access log. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the `nsslapd-accesslog-logrotationtimeunit` attribute.

For performance reasons, it is not recommended that you specify no log rotation as the log will grow indefinitely. However, there are two ways to specify no log rotation. Either set the `nsslapd-accesslog-maxlogsperdir` attribute value to 1 or the `nsslapd-accesslog-logrotationtime` attribute to -1. The server checks the `nsslapd-accesslog-maxlogsperdir` attribute first and if this attribute value is larger than 1, the server then checks the `nsslapd-accesslog-logrotationtime` attribute. See “nsslapd-accesslog-maxlogsperdir (Access Log Maximum Number of Log Files)” on page 94 for more information.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the time between access log file rotation is unlimited.
Default Value	`1`
Syntax	Integer
Example	`nsslapd-accesslog-logrotationtime: 100`

nsslapd-accesslog-logrotationtimeunit (Access Log Rotation Time Unit)

Specifies the units for the `nsslapd-accesslog-logrotationtime` attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	month | week | day | hour | minute
Default Value	`day`
Syntax	DirectoryString
Example	`nsslapd-accesslog-logrotationtimeunit: week`

nsslapd-accesslog-maxlogsize (Access Log Maximum Log Size)

Specifies the maximum access log size in megabytes. When this value is reached, the access log is rotated. That is, the server starts writing log information to a new log file. If you set the `nsslapd-accesslog-maxlogsperdir` attribute to 1, the server ignores this attribute.

When setting a maximum log size, consider the total number of log files that can be created due to log file rotation. Also, remember that there are 3 different log files (access log, audit log, and error log) maintained by the Directory Server, each of which will consume disk space. Compare these considerations to the total amount of disk space that you want to be used by the access log.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means the log file is unlimited in size.
Default Value	`100`
Syntax	Integer
Example	`nsslapd-accesslog-maxlogsize: 100`

nsslapd-accesslog-maxlogsperdir (Access Log Maximum Number of Log Files)

Specifies the total number of access logs that can be contained in the directory where the access log is stored. If you are using log file rotation, each time the access log is rotated, a new log file is created. When the number of files contained in the access log directory exceeds the value stored on this attribute, the oldest version of the log file is deleted. For performance reasons, it is not recommended that you set this value to 1, as the server will not rotate the log and it will grow indefinitely.

If the value for this attribute is higher than 1, then you need to check the `nsslapd-accesslog-logrotationtime` attribute to establish whether or not log rotation is specified. If the `nsslapd-accesslog-logrotationtime` attribute has a value of -1, there is no log rotation. For more information, see “nsslapd-accesslog-logrotationtime (Access Log Rotation Time)” on page 93.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)

Default Value	`10`
Syntax	Integer
Example	`nsslapd-accesslog-maxlogsperdir: 10`

nsslapd-attribute-name-exceptions

Allows non-standard characters in attribute names to be used for backward compatibility with older servers.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-attribute-name-exceptions: on`

nsslapd-auditlog (Audit Log)

Specifies the pathname and filename of the log used to record changes made to each database.

Property	Value
Entry DN	`cn=config`
Valid Range	Any valid filename
Default Value	*ServerRoot*`/slapd-`*serverID*`/logs/audit`
Syntax	DirectoryString
Example	`nsslapd-auditlog: /`*ServerRoot*`/slapd-`*serverID*`/logs/audit`

For audit logging to be enabled, this attribute must have a valid path and file name and the `nsslapd-auditlog-logging-enabled` configuration attribute must be switched to on. Table 4-2 lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of audit logging.

Table 4-2 Possible Value Combinations of Audit Log Attributes

Attribute Pair	Value Pair	Logging Status
`nsslapd-auditlog-logging-enabled` `nsslapd-auditlog`	`on` empty string	Disabled
`nsslapd-auditlog-logging-enabled` `nsslapd-auditlog`	`on` *filename*	Enabled
`nsslapd-auditlog-logging-enabled` `nsslapd-auditlog`	`off` empty string	Disabled
`nsslapd-auditlog-logging-enabled` `nsslapd-auditlog`	`off` *filename*	Disabled

nsslapd-auditlog-list

Provides a list of audit log files.

Property	Value
Entry DN	`cn=config`
Valid Range	N/A
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-auditlog-list: auditlog2,auditlog3`

nsslapd-auditlog-logexpirationtime (Audit Log Expiration Time)

Specifies the maximum age that a log file can be before it is deleted. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the `nsslapd-auditlog-logexpirationtimeunit` attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	`1`
Syntax	Integer
Example	`nsslapd-auditlog-logexpirationtime: 1`

nsslapd-auditlog-logexpirationtimeunit (Audit Log Expiration Time Unit)

Specifies the units for the `nsslapd-auditlog-logexpirationtime` attribute. If the unit is unknown by the server, the log will never expire.

Property	Value
Entry DN	`cn=config`
Valid Range	month | week | day
Default Value	`month`
Syntax	DirectoryString
Example	`nsslapd-auditlog-logexpirationtimeunit: day`

nsslapd-auditlog-logging-enabled (Audit Log Enable Logging)

Turns audit logging on and off.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`

Syntax	DirectoryString
Example	`nsslapd-auditlog-logging-enabled: off`

For audit logging to be enabled this attribute must be switched to on and the `nsslapd-auditlog` configuration attribute must have a valid path and file name. Table 4-2 on page 96 lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of audit logging.

nsslapd-auditlog-logmaxdiskspace (Audit Log Maximum Disk Space)

Specifies the maximum amount of disk space in megabytes that the audit logs are allowed to consume. If this value is exceeded, the oldest audit log is deleted.

When setting a maximum disk space, consider the total number of log files that can be created due to log file rotation. Also, remember that there are three different log files (access log, audit log, and error log) maintained by the Directory Server, each of which will consume disk space. Compare these considerations with the total amount of disk space that you want to be used by the audit log.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the disk space allowed for the audit log is unlimited in size.
Default Value	`100`
Syntax	Integer
Example	`nsslapd-auditlog-logmaxdiskspace: 500`

nsslapd-auditlog-logminfreediskspace (Audit Log Minimum Free Disk Space)

Specifies the minimum permissible free disk space in megabytes. When the amount of free disk space falls below the value specified on this attribute, the oldest audit log is deleted until enough disk space is freed to satisfy this attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	`5`
Syntax	Integer
Example	`nsslapd-auditlog-logminfreediskspace: 3`

nsslapd-auditlog-logrotationtime (Audit Log Rotation Time)

Specifies the time between audit log file rotations. The audit log will be rotated when this time interval is up, regardless of the current size of the audit log. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the `nsslapd-auditlog-logrotationtimeunit` attribute. If you set the `nsslapd-auditlog-maxlogsperdir` attribute to 1, the server ignores this attribute.

For performance reasons, it is not recommended that you specify no log rotation, as the log will grow indefinitely. However, there are two ways to specify no log rotation. Either set the `nsslapd-auditlog-maxlogsperdir` attribute value to 1 or the `nsslapd-auditlog-logrotationtime` attribute to -1. The server checks the `nsslapd-auditlog-maxlogsperdir` attribute first and if this attribute value is larger than 1, the server checks the `nsslapd-auditlog-logrotationtime` attribute. See “nsslapd-auditlog-maxlogsperdir (Audit Log Maximum Number of Log Files)” on page 101 for more information.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the time between audit log file rotations is unlimited.
Default Value	`1`
Syntax	Integer

Example	`nsslapd-auditlog-logrotationtime: 100`

nsslapd-auditlog-logrotationtimeunit (Audit Log Rotation Time Unit)

Specifies the units for the `nsslapd-auditlog-logrotationtime` attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	month | week | day | hour | minute
Default Value	`week`
Syntax	DirectoryString
Example	`nsslapd-auditlog-logrotationtimeunit: day`

nsslapd-auditlog-maxlogsize (Audit Log Maximum Log Size)

Specifies the maximum audit log size in megabytes. When this value is reached, the audit log is rotated. That is, the server starts writing log information to a new log file. If you set `nsslapd-auditlog-maxlogsperdir` to `1`, the server ignores this attribute.

When setting a maximum log size, consider the total number of log files that can be created due to log file rotation. Also remember that there are 3 different log files (access log, audit log, and error log) maintained by the Directory Server, each of which will consume disk space. Compare these considerations to the total amount of disk space that you want to be used by the audit log.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647) where a value of -1 means the log file is unlimited in size.
Default Value	`100`
Syntax	Integer
Example	`nsslapd-auditlog-maxlogsize: 50`

nsslapd-auditlog-maxlogsperdir (Audit Log Maximum Number of Log Files)

Specifies the total number of audit logs that can be contained in the directory where the audit log is stored. If you are using log file rotation, then each time the audit log is rotated, a new log file is created. When the number of files contained in the audit log directory exceeds the value stored on this attribute, the oldest version of the log file is deleted. The default is 1 log. If you accept this default, the server will not rotate the log and it will grow indefinitely.

If the value for this attribute is higher than 1, you need to check the `nsslapd-auditlog-logrotationtime` attribute to establish whether or not log rotation is specified. If the `nsslapd-auditlog-logrotationtime` attribute has a value of -1, then there is no log rotation. See “nsslapd-auditlog-logrotationtime (Audit Log Rotation Time)” on page 99 for more information.

Property	Value
Entry DN	`cn=config`
Valid range	1 to the maximum 32 bit integer value (2147483647)
Default value	`1`
Syntax	Integer
Example	`nsslapd-auditlog-maxlogsperdir: 10`

nsslapd-certmap-basedn (Certificate Map Search Base)

This attribute can be used when client authentication is performed using SSL certificates in order to avoid limitation of the security subsystem certificate mapping, configured in `certmap.conf`. Depending on the `certmap.conf` configuration, the certificate mapping may be done using a directory subtree search based at the root DN. Note that if the search is based at the root DN, then the `nsslapd-certmap-basedn` attribute may force the search to be based at some entry other than the root. For further information, see Chapter 11, “Implementing Security” in the *Sun ONE Directory Server Administration Guide.*

Property	Value
Entry DN	`cn=config`
Valid Range	The DN of an entry in the directory
Default Value	`N/A`

Syntax	DN
Example	`nsslapd-certmap-basedn: ou=people,dc=example,dc=com`

nsslapd-config

This read-only attribute is the config DN.

Property	Value
Entry DN	`cn=config`
Valid Range	Any valid config DN.
Default Value	N/A
Syntax	DirectoryString
Example	`nsslapd-config:cn=config`

nsslapd-ds4-compatible-schema

Makes the schema in `cn=schema` compatible with 4.x versions of Directory Server.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-ds4-compatible-schema: off`

nsslapd-enquote-sup-oc (Enable Superior Object Class Enquoting)

Controls whether the quoting in the `objectclasses` attributes contained in the `cn=schema` entry conforms to the quoting specified by internet draft RFC 2252. By default, the Directory Server places single quotes around the superior object class identified on the `objectclasses` attributes contained in `cn=schema`. RFC 2252 indicates that this value should not be quoted.

That is, the Directory Server publishes `objectclasses` attributes in the `cn=schema` entry as follows:

```
objectclasses: ( 2.5.6.6 NAME 'person' DESC 'Standard ObjectClass'
SUP 'top' MUST ( objectclass $ sn $ cn ) MAY ( aci $ description $
seealso $ telephonenumber $ userpassword ) )
```

However, RFC 2252 indicates that this attribute should be published as follows:

```
objectclasses: ( 2.5.6.6 NAME 'person' DESC 'Standard ObjectClass'
SUP top MUST ( objectclass $ sn $ cn ) MAY ( aci $ description $
seealso $ telephonenumber $ userpassword ) )
```

Notice the absence of single quotes around the word `top`.

Turning this attribute on means that the Directory Server Resource Kit LDAP Clients will no longer function, as they require the schema as defined in RFC 2252.

Turning this attribute off causes the Directory Server to conform to RFC 2252, but doing so may interfere with some earlier LDAP clients. Specifically, any client written using the Sun ONE LDAP SDK for Java 4.x will no longer be able to correctly read and modify schema. This includes the 4.x version of the Sun ONE Server Console. Please note that turning this attribute on or off does not affect the 5.x Sun ONE Server Console.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-enquote-sup-oc: off`

nsslapd-errorlog (Error Log)

Specifies the pathname and filename of the log used to record error messages generated by the Directory Server. These messages can describe error conditions, but more often they contain informative conditions such as:

- server startup and shutdown times
- port number the server uses

This log contains varying amounts of information depending on the current setting of the Log Level attribute. See “nsslapd-errorlog-level (Error Log Level),” on page 104 for more information.

Property	Value
Entry DN	`cn=config`
Valid Range	Any valid filename
Default Value	*ServerRoot*`/slapd-`*serverID*`/logs/error`
Syntax	DirectoryString
Example	`nsslapd-errorlog: /`*ServerRoot*`/slapd-`*serverID*`/logs/error`

For error logging to be enabled, this attribute must have a valid path and file name and the `nsslapd-errorlog-logging-enabled` configuration attribute must be switched to on. Table 4-3 lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of error logging.

Table 4-3 Possible Value Combinations of Error Log Attributes

Attribute Pair	Value Pair	Logging Status
`nsslapd-errorlog-logging-enabled`	`on`	Disabled
`nsslapd-errorlog`	empty string	
`nsslapd-errorlog-logging-enabled`	`on`	Enabled
`nsslapd-errorlog`	*filename*	
`nsslapd-errorlog-logging-enabled`	`off`	Disabled
`nsslapd-errorlog`	empty string	
`nsslapd-errorlog-logging-enabled`	`off`	Disabled
`nsslapd-errorlog`	*filename*	

nsslapd-errorlog-level (Error Log Level)

Specifies the level of logging to be used by the Directory Server.

NOTE This attribute has been deprecated in Directory Server 5.2. It is still supported for backward compatibility but has been replaced by the nsslapd-infolog-area (Information Log Area) and nsslapd-infolog-level (Information Log Level) attributes.

nsslapd-errorlog-list (Error Log List)

This read-only attribute provides a list of error log files.

Property	Value
Entry DN	`cn=config`
Valid Range	N/A
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-errorlog-list:errorlog2,errorlog3`

nsslapd-errorlog-logexpirationtime (Error Log Expiration Time)

Specifies the maximum age that a log file is allowed to reach before it is deleted. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the `nsslapd-errorlog-logexpirationtimeunit` attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	`1`
Syntax	Integer
Example	`nsslapd-errorlog-logexpirationtime: 1`

nsslapd-errorlog-logexpirationtimeunit (Error Log Expiration Time Unit)

Specifies the units for the `nsslapd-errorlog-logexpirationtime` attribute. If the unit is unknown by the server, the log will never expire.

Property	Value
Entry DN	`cn=config`
Valid Range	month | week | day
Default Value	`month`
Syntax	DirectoryString
Example	`nsslapd-errorlog-logexpirationtimeunit: week`

nsslapd-errorlog-logging-enabled (Enable Error Logging)

Turns error logging on and off.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nsslapd-errorlog-logging-enabled: on`

nsslapd-errorlog-logmaxdiskspace (Error Log Maximum Disk Space)

Specifies the maximum amount of disk space in megabytes that the error logs are allowed to consume. If this value is exceeded, the oldest error log is deleted.

When setting a maximum disk space, consider the total number of log files that can be created due to log file rotation. Also, remember that there are 3 different log files (access log, audit log, and error log) maintained by the Directory Server, each of which will consume disk space. Compare these considerations to the total amount of disk space that you want to be used by the error log.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the disk space allowed to the error log is unlimited in size.
Default Value	`100`
Syntax	Integer
Example	`nsslapd-errorlog-logmaxdiskspace: 500`

nsslapd-errorlog-logminfreediskspace (Error Log Minimum Free Disk Space)

Specifies the minimum allowed free disk space in megabytes. When the amount of free disk space falls below the value specified on this attribute, the oldest error log is deleted until enough disk space is freed to satisfy this attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	`5`
Syntax	Integer
Example	`nsslapd-errorlog-logminfreediskspace: 5`

nsslapd-errorlog-logrotationtime (Error Log Rotation Time)

Specifies the time between error log file rotations. The error log will be rotated when this time interval is up, regardless of the current size of the error log. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the `nsslapd-errorlog-logrotationtimeunit` attribute.

For performance reasons, it is not recommended that you specify no log rotation as the log will grow indefinitely. However, there are two ways to specify no log rotation. Either set the `nsslapd-errorlog-maxlogsperdir` attribute value to 1 or the `nsslapd-errorlog-logrotationtime` attribute to -1. The server checks the `nsslapd-errorlog-maxlogsperdir` attribute first and if this attribute value is larger than 1, the server then checks the `nsslapd-errorlog-logrotationtime` attribute. See "nsslapd-errorlog-maxlogsperdir (Maximum Number of Error Log Files)" on page 109 for more information.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647), where a value of `-1` means that the time between error log file rotation is unlimited).
Default Value	`1`
Syntax	Integer
Example	`nsslapd-errorlog-logrotationtime: 100`

nsslapd-errorlog-logrotationtimeunit (Error Log Rotation Time Unit)

Specifies the units for `nsslapd-errorlog-logrotationtime` (Error Log Rotation Time). If the unit is unknown by the server, the log will never expire.

Property	Value
Entry DN	`cn=config`
Valid Range	month | week | day | hour | minute
Default Value	`week`
Syntax	DirectoryString
Example	`nsslapd-errorlog-logrotationtimeunit: day`

nsslapd-errorlog-maxlogsize (Maximum Error Log Size)

Specifies the maximum error log size in megabytes. When this value is reached, the error log is rotated. That is, the server starts writing log information to a new log file. If you set `nsslapd-errorlog-maxlogsperdir` to 1, the server ignores this attribute.

When setting a maximum log size, consider the total number of log files that can be created due to log file rotation. Also, remember that there are 3 different log files (access log, audit log, and error log) maintained by the Directory Server, each of which will consume disk space. Compare these considerations to the total amount of disk space that you want to be used by the error log.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 | 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means the log file is unlimited in size.
Default Value	`100`
Syntax	Integer
Example	`nsslapd-errorlog-maxlogsize: 100`

nsslapd-errorlog-maxlogsperdir (Maximum Number of Error Log Files)

Specifies the total number of error logs that can be contained in the directory where the error log is stored. If you are using log file rotation, then each time the error log is rotated, a new log file is created. When the number of files contained in the error log directory exceeds the value stored on this attribute, the oldest version of the log file is deleted. If this attribute is set to `1`, the server will not rotate the log and it will grow indefinitely.

If the value for this attribute is higher than 1, then you need to check the `nsslapd-errorlog-logrotationtime` attribute to establish whether or not log rotation is specified. If the `nsslapd-errorlog-logrotationtime` attribute has a value of -1 then there is no log rotation. See “nsslapd-errorlog-logrotationtime (Error Log Rotation Time)” on page 108 for more information.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647)

Default Value	2
Syntax	Integer
Example	`nsslapd-errorlog-maxlogsperdir: 10`

nsslapd-groupevalnestlevel

Specifies the number of levels of nesting that the access control system will perform for group evaluation.

Property	Value
Entry DN	`cn=config`
Valid Range	0 to the maximum 64-bit integer value
Default Value	0
Syntax	Integer
Example	`nsslapd-groupevalnestlevel:5`

nsslapd-hash-filters

Enables experimental code that attempts to speed up filter comparisons by using a hash. This attribute would be used if search tune in the database instance is set to include the VLV_INDEX flag.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-hash-filters: off`

nsslapd-idletimeout (Idle Timeout)

Specifies the amount of time in seconds after which an idle LDAP client connection is closed by the server. A value of 0 indicates that the server will never close idle connections.

Property	Value
Entry DN	`cn=config`
Valid Range	0 to the maximum 32 bit integer value (2147483647)
Default Value	`0`
Syntax	Integer
Example	`nsslapd-IdleTimeout: 0`

nsslapd-infolog-area (Information Log Area)

Specifies the component for which logging information should be provided. Each component is identified as an area, whose value is a decimal translation of the hex values in `slapi-plugin.h`.

The log area is additive; for example, to enable logging on Search filter processing (32) and Config file processing (64), you would set this attribute to 96 (32+64).

If you are writing plug-ins for the Directory Server, refer to the *Sun ONE Directory Server Plug-In API Programming Guide* for more information on using this attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	1 = Trace function calls. Logs a message when the server enters and exits a function. 2 = Debug packet handling 4 = Heavy trace output debugging 8 = Connection management 16 = Print out packets sent/received 32 = Search filter processing 64 = Config file processing 128 = Access control list processing 2048 = Log entry parsing debugging 4096 = Housekeeping thread debugging 8192 = Replication debugging 16384 = Default logging area, used for critical errors and other messages that are always written to the error log, for example server startup messages. Messages at this level are always included in the error log regardless of the `nsslapd-infolog-level` setting. 32768 = Database cache debugging. 65536 = Server plug-in debugging. An entry is written to the log file when a server plug-in calls `slapi_log_error_ex`.
Default Value	`0`
Syntax	Integer
Example	`nsslapd-infolog-area: 0`

nsslapd-infolog-level (Information Log Level)

Specifies the level of logging information that should be returned for the server component defined by the `nsslapd-infolog-area` attribute. A value of `0` means that only default logging information is returned for the selected area. Setting this attribute to `1` enables additional logging information to be returned for the selected area.

Property	Value
Entry DN	`cn=config`
Valid Range	0 | 1
Default Value	`0`
Syntax	Integer
Example	`nsslapd-infolog-level: 0`

nsslapd-instancedir (Instance Directory)

Specifies the full path to the directory where this server instance is installed. The hostname is the default *serverID* given at installation time.

Property	Value
Entry DN	`cn=config`
Valid Range	Any valid file path.
Default Value	*ServerRoot*`/slapd-`*serverID*
Syntax	DirectoryString
Example	`nsslapd-instancedir: /`*ServerRoot*`/slapd-mySer`ver

nsslapd-ioblocktimeout (IO Block Time Out)

Specifies the amount of time in milliseconds after which the connection to a stalled LDAP client is closed. An LDAP client is considered to be stalled when it has not made any I/O progress for read or write operations.

Property	Value
Entry DN	`cn=config`
Valid Range	0 to the maximum 32 bit integer value (2147483647) in ticks
Default Value	`1800000`
Syntax	Integer
Example	`nsslapd-ioblocktimeout: 1800000`

nsslapd-lastmod (Track Modification Time)

Specifies whether the Directory Server maintains the modification attributes for Directory Server entries. These attributes include:

- `modifiersname`—The distinguished name of the person who last modified the entry.
- `modifytimestamp`—The timestamp, in GMT format, for when the entry was last modified.
- `creatorsname`—The distinguished name of the person who initially created the entry.
- `createtimestamp`—The timestamp for when the entry was created in GMT format.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nsslapd-lastmod: off`

nsslapd-listenhost (Listen to IP Address)

Allows multiple Directory Server instances to run on a multihomed machine (or makes it possible to limit listening to one interface of a multihomed machine). Provide the hostname which corresponds to the IP interface you want to specify as a value for this attribute. Directory Server will only respond to requests sent to the interface that corresponds to the hostname provided on this attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	Any hostname.
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-listenhost:` *host_name*

nsslapd-localhost (Local Host)

This read-only attribute specifies the host machine on which the Directory Server runs.

Property	Value
Entry DN	`cn=config`
Valid Range	Any fully qualified hostname.
Default Value	Hostname of installed machine.
Syntax	DirectoryString
Example	`nsslapd-localhost:myServer.example.com`

nsslapd-localuser (Local User)

UNIX and Linux installations only. Specifies the user under which the Directory Server runs. The group under which the user runs is derived from this attribute, by examining the groups that the user is a member of. Should the user change, all the files in the installation directory must be owned by this user.

Property	Value
Entry DN	`cn=config`
Valid Range	Any valid user on the local UNIX machine.
Default Value	To run as the same user who started the Directory Server.
Syntax	DirectoryString
Example	`nsslapd-localuser: nobody`

nsslapd-maxbersize (Maximum Message Size)

Defines the maximum size in bytes allowed for an incoming message. This limits the size of LDAP requests that can be handled by the Directory Server. Limiting the size of requests prevents some kinds of denial of service attacks.

The limit applies to the total size of the LDAP request. For example, if the request is to add an entry, and the entry in the request is larger than two megabytes, then the add request is denied. Care should be taken when changing this attribute and we recommend contacting Sun ONE Professional Services before doing so.

Property	Value
Entry DN	`cn=config`
Valid Range	0 - 2GB (*2,147,483,647 bytes)* where a value of 0 indicates that the default value should be used.
Default Value	`2097152`
Syntax	Integer
Example	`nsslapd-maxbersize: 2097152`

nsslapd-maxconnections (Maximum Number of Connections)

This attribute limits the number of simultaneous connections the server can manage. The value of this attribute is not set by default. If it is not set manually, its implicit value is the maximum number of file descriptors a process can open on the system.

You can use this attribute to limit the amount of memory used by Directory Server. Directory Server allocates *n**512 bytes of data, where *n* is equal to the value of `nsslapd-maxconnections`, if set, or to the maximum number of file descriptors a process can open on the system.

For example, on Solaris 9 systems, the maximum number of file descriptors is 64000. If `nsslapd-maxconnections` is not set, Directory Server will allocate 35MB of data, which may cause problems for some deployments. Setting `nsslapd-maxconnections` to a suitable value can help to alleviate this problem.

Property	Value
Entry DN	`cn=config`
Valid Range	`nsslapd-reservedescriptors` +1 to `maxdescriptors`. If the `maxdescriptors` attribute is not set, the maximum value of `nsslapd-maxconnections` is the maximum number of file descriptors a process can open on the system.
Default Value	`N/A`
Syntax	Integer
Example	`nsslapd-maxconnections: 4096`

nsslapd-maxdescriptors (Maximum File Descriptors)

Not applicable to directory installations on Windows and AIX.

This attribute sets the maximum, platform-dependent number of file descriptors that the Directory Server will try to use. A file descriptor is used whenever a client connects to the server. It is also used for some server activities such as index maintenance. The number of available file descriptors for TCP/IP connections is the total for the `nsslapd-maxdescriptors` attribute minus the number of file descriptors used by the server for non-client connections, such as index management and managing replication, as specified in the `nsslapd-reservedescriptors` attribute (see "nsslapd-reservedescriptors (Reserved File Descriptors)" on page 123.)

The number that you specify here should not be greater than the total number of file descriptors that your operating system allows the `ns-slapd` process to use. This number will differ depending on your operating system. Some operating systems allow you to configure the number of file descriptors available to a process. See your operating system documentation for details on file descriptor limits and configuration. It is worth noting that the included `idsktune` program can be used to suggest changes to the system kernel or TCP/IP tuning attributes, including increasing the number of file descriptors if necessary. You should consider increasing the value on this attribute if the Directory Server is refusing connections because it is out of file descriptors. When this occurs, the following message is written to the Directory Server's error log file:

```
Not listening for new connections -- too many fds open
```

NOTE UNIX shells usually have configurable limits on the number of file descriptors. See your operating system documentation for further information regarding `limit` and `ulimit` as these limits can often cause problems.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to 65535
Default Value	`1024`
Syntax	Integer
Example	`nsslapd-maxdescriptors: 1024`

nsslapd-maxpsearch (Maximum Persistent Searches)

Defines the maximum number of persistent searches that can be performed on the Directory Server. The persistent search mechanism provides an active channel through which entries that change (and information about the changes that occur) can be communicated. Because each persistent search operation uses one thread, limiting the number of simultaneous persistent searches prevents certain kinds of denial of service attacks.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to maximum threadnumber
Default Value	`30`
Syntax	Integer
Example	`nsslapd-maxpsearch: 30`

nsslapd-maxthreadsperconn (Maximum Threads Per Connection)

Defines the maximum number of threads that a connection should use. For normal operations where a client binds and performs only one or two operations before unbinding, you should use the default value. For situations where a client binds and simultaneously issues many requests, you should increase this value to allow each connection enough resources to perform all the operations.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to maximum threadnumber
Default Value	`5`
Syntax	Integer
Example	`nsslapd-maxthreadsperconn: 5`

nsslapd-nagle

When the value of this attribute is `off`, the TCP_NODELAY option is set so that LDAP responses (such as entries or result messages) are sent back to a client immediately. When the attribute is turned on, default TCP behavior applies. That is, the sending of data is delayed, in the hope that this will enable additional data to be grouped into one packet of the underlying network MTU size (typically 1500 bytes for Ethernet).

Property	Value
Entry DN	`cn=config`
Valid range	on | off
Default value	`off`
Syntax	DirectoryString
Example	`nsslapd-nagle: off`

nsslapd-plugin

This multi-valued, read-only attribute lists the syntaxes and matching rules loaded by the server.

nsslapd-port (Port Number)

TCP/IP port number used for LDAP communications. If you want to run SSL/TLS over this port, you can do so through the Start TLS extended operation. This selected port must be unique on the host system; make sure no other application is attempting to use the same port number. On UNIX systems, specifying a port number of less than 1024 requires the Directory Server to run as root.

If you are changing the port number for a configuration directory, you must also update the corresponding Server Instance Entry in the configuration directory. Please note that you need to restart the server for the port number change to be taken into account.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to 65535
Default Value	`389`
Syntax	Integer

Example	`nsslapd-port: 389`

nsslapd-privatenamespaces

Contains the list of the private naming contexts `cn=config`, `cn=schema`,and `cn=monitor`.

Property	Value
Entry DN	`cn=config`
Valid Range	cn=config, cn=schema ,and cn=monitor
Default Value	`N/A`
Syntax	DirectoryString
Example	`nsslapd-privatenamespaces: cn=config`

nsslapd-readonly (Read Only)

Specifies whether the whole server is in read-only mode, meaning that neither data in the database(s) nor configuration information can be modified. Any attempt to modify a database in read-only mode returns an error indicating that the server is unwilling to perform the operation.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-readonly: off`

nsslapd-referral (Referral)

This multi-valued attribute specifies the LDAP URL(s) to be returned by the suffix, when the server receives a request for an entry not belonging to the local tree, that is, an entry whose suffix does not match the value specified on any of the suffix attributes. For example, suppose the database contains only the entries:

`ou=People, dc=example,dc=com`

but the request is for:

`ou=Groups, dc=example,dc=com`

In this case, the referral is returned so the client may contact the corresponding directory for the requested entry. Although only one referral is allowed per Directory Server instance, this referral can have multiple values.

NOTE If you want to use SSL and TLS communications, the Referral attribute should be in the following form:

`ldaps://`*serverHost*

Start TLS does not support referrals.

For more information on managing referrals, see "Setting Referrals" in Chapter 2 of the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=config`
Valid Range	Valid LDAP URL in the following format: `ldap://`*serverHost*
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-referral: ldap://alternate.example.com`

nsslapd-referralmode (Referral Mode)

When set, this attribute will send back the referral for *any* request on *any* suffix.

Property	Value
Entry DN	`cn=config`
Valid Range	Valid LDAP URL in the following format: `ldap://`*serverHost*
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-referralmode: ldap://backup.example.com`

nsslapd-reservedescriptors (Reserved File Descriptors)

Not applicable to directory installations on Windows and AIX.

This read-only attribute specifies the number of file descriptors that Directory Server reserves for managing non-client connections, such as index management and managing replication. The number of file descriptors that the server reserves for this purpose subtracts from the total number of file descriptors available for servicing LDAP client connections (see "nsslapd-maxdescriptors (Maximum File Descriptors)" on page 118).

Most installations of Directory Server should never need to change this attribute. However, consider increasing the value on this attribute if all of the following are true:

- The server is replicating to a large number of consumer servers (more than 10) and / or the server is maintaining a large number of index files (more than 30).
- The server is servicing a large number of LDAP connections.
- You get error messages reporting that the server is unable to open file descriptors (the actual error message will differ depending on the operation that the server is attempting to perform), but these error messages are NOT related to managing client LDAP connections.

Increasing the value on this attribute may result in more LDAP clients being unable to access your directory. Therefore, when you increase the value on this attribute, increase the value on the `nsslapd-maxdescriptors` attribute also. Note that you may not be able to increase the `nsslapd-maxdescriptors` value if your server is already using the maximum number of file descriptors that your operating system allows a process to use (see your operating system documentation for details). If this is the case, then reduce the load on your server by causing LDAP clients to search alternative directory replicas.

To assist you in computing the number of file descriptors you set for this attribute, we suggest you use the following formula:

`nsslapd-reservedescriptor =`
`20 + (`*NumBackends* `* 4) +` *NumGlobalIndexes* `+` *ReplicationDescriptors* `+` *ChainingBackendDescriptors* `+` *PTADescriptors* `+` *SSLDescriptors*

where the terms are given in the following table:

Table 4-4 Terms for Computing the Value of `nsslapd-reservedescriptor`

Term	Definition
NumldbmBackends	Number of ldbm databases.

Table 4-4 Terms for Computing the Value of `nsslapd-reservedescriptor`

Term	Definition
NumGlobalIndexes	Total number of configured indexes for all databases including system indexes. By default, there are 8 system indexes and 17 additional indexes per database.
ReplicationDescriptors	*NumSupplierReplicas* + 8 Where *NumSupplierReplicas* is number of replicas in the server that can act as a supplier (hub or master).
ChainingBackendDescriptors	*NumChainingBackends* * `nsOperationConnectionsLimit` Where `nsOperationConnectionsLimit` is defined in the chained suffix configuration and 10 by default.
PTADescriptors	3 if PTA is configured, 0 if PTA is not configured.
SSLDescriptors	5 (4 files + 1 listen socket) if SSL is configured, 0 if SSL is not configured.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to 65535
Default Value	`64`
Syntax	Integer
Example	`nsslapd-reservedescriptors: 64`

nsslapd-return-exact-case (Return Exact Case)

Returns the exact case of attribute names, as defined in the schema.

Attribute names are case-insensitive by default. However, when an attribute is returned by the Directory Server (as the result of a search operation) some client applications require attribute names to match the case of the attribute as it is listed in the schema. Other client applications require attribute names to be returned in lower case (the default behavior in Directory Server 4.x).

`nsslapd-return-exact-case` is enabled by default. You should disable this attribute if you have legacy clients that expect attribute names to be returned in lower case (for backward compatibility with Directory Server 4.x). You must stop and restart the server for changes to this attribute to be taken into account.

Note that if the attribute name is specified in the search, it is returned in the case in which it is specified, regardless of the value of `nsslapd-return-exact-case`.

For example, the following search command

```
ldapsearch -b "cn=config" -s base objectclass=* "PassWordMinAGe"
```

returns the attribute as "`PassWordMinAGe=0`", whether `nsslapd-return-exact-case` is set to `on` or `off`.

If `nsslapd-return-exact-case` is set to `on`, the following search command

```
ldapsearch -b "cn=config" -s base objectclass=*
```

returns the attribute as "`passwordMinAge=0`", which is how this attribute is defined in the schema.

If `nsslapd-return-exact-case` is set to `off`, the same search command

```
ldapsearch -b "cn=config" -s base objectclass=*
```

returns the attribute as "`passwordminage=0`" (in lower case).

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	on
Syntax	DirectoryString
Example	`nsslapd-return-exact-case: on`

nsslapd-rootdn (Manager DN)

Specifies the distinguished name of an entry that is not subject to access control restrictions, administrative limit restrictions for operations on the directory or resource limits in general. The attributes `nsslapd-sizelimit`, `nsslapd-timelimit`, and `nsslapd-schemacheck` do not apply to this DN either.

For information on changing the Root DN, see Chapter 2, "Creating Directory Entries" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=config`
Valid Range	Any valid distinguished name
Default Value	N/A
Syntax	DN
Example	`nsslapd-rootdn: cn=Directory Manager`

nsslapd-rootpw (Root Password)

Allows you to specify the password associated with the `"Manager DN"`. When you provide the root password, it will be encrypted according to the encryption method you selected for "nsslapd-rootpwstoragescheme (Root Password Storage Scheme)" on page 127. When viewed from the server console, this attribute shows the value:`*****` When viewed from the `dse.ldif` file, this attribute shows the encryption method followed by the encrypted string of the password. Please note that the example below is what you *view*, *not* what you type.

CAUTION If you configure a root DN at server installation time, you must also provide a root password. However, it is possible for the root password to be deleted from `dse.ldif` by direct editing of the file. In this situation, the root DN can only obtain the same access to your directory as you allow for anonymous access. Always make sure that a root password is defined in `dse.ldif` when a root DN is configured for your database.

Property	Value
Entry DN	`cn=config`

Valid Range	Any valid password encrypted by any one of the encryption methods that are described in "passwordStorageScheme (Password Storage Scheme)," on page 170.
Default Value	N / A
Syntax	DirectoryString `{encryption_method} encrypted_Password`
Example	`nsslapd-rootpw: {SSHA}9Eko69APCJfF`

nsslapd-rootpwstoragescheme (Root Password Storage Scheme)

Available only from the server console. This attribute indicates the encryption method used for the root password.

Property	Value
Entry DN	`cn=config`
Valid Range	Any encryption method as described in "passwordStorageScheme (Password Storage Scheme)" on page 170.
Default Value	SSHA
Syntax	DirectoryString
Example	`nsslapd-rootpwstoragescheme: SSHA`

nsslapd-schema-repl-useronly

This attribute allows you to have greater control over the schema that is replicated. The attribute is `off` by default, implying that the entire schema is replicated. If the attribute is set to on, only schema with an X-ORIGIN of `user-defined` is replicated. This setting greatly improves the performance of schema replication.

If you are replicating from a 5.2 Directory Server to a 5.1 server, you *must* set this attribute to on. Otherwise the 5.2 schema will be pushed to the 5.1 server and the 5.1 server will be unable to restart, due to duplicate objects.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`

Syntax	DirectoryString
Example	`nsslapd-schema-repl-useronly: off`

nsslapd-schemacheck (Schema Checking)

Specifies whether the database schema will be enforced during entry insertion or modification. When this attribute has a value of on, Directory Server will not check the schema of existing entries until they are modified. The database schema defines the type of information allowed in the database. You can extend the default schema using the `objectclasses` and attribute types. For information on how to extend your schema using Directory Server console, see Chapter 9, "Extending the Directory Schema" in the *Sun ONE Directory Server Administration Guide.*

NOTE Schema checking works by default when database modifications are made using an LDAP client, such as `ldapmodify`, the Directory Server console, or when importing a database from LDIF using `ldif2db` (`directoryserver ldif2db` on Solaris 9 platforms).

If you turn schema checking off, you will have to verify manually that your entries conform to the schema. If schema checking is turned on, the server sends an error message to inform you of the entries that do not match the schema. Make sure that the attributes and object classes you create in your LDIF statements are both spelled correctly and identified in `dse.ldif`. You will need to create a file in LDIF format in the schema directory or add the elements to `99user.ldif`.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	on
Syntax	DirectoryString
Example	`nsslapd-schemacheck: on`

nsslapd-securelistenhost

Allows multiple Directory Server instances to run on a multihomed machine, using secure SSL / TLS connections (or makes it possible to limit listening to one interface of a multihomed machine). Provide the hostname which corresponds to the IP interface you want to specify as a value for this attribute. Directory Server will only respond to requests sent to the interface that corresponds to the hostname provided on this attribute.

Property	Value
Entry DN	`cn=config`
Valid Range	Any secure hostname.
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-securelistenhost:`*secure_host_name*

nsslapd-securePort (Encrypted Port Number)

TCP / IP port number used for SSL / TLS communications. This selected port must be unique on the host system; make sure no other application is attempting to use the same port number. For UNIX systems, specifying a port number of less than 1024 requires that Directory Server runs as root.

The default value 636 is only used if the server has been configured with a private key and a certificate; otherwise it does not listen on this port.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to 65535
Default Value	`636`
Syntax	Integer
Example	`nsslapd-securePort: 636`

nsslapd-security (Security)

Enables the use of security features (SSL / TLS and attribute encryption) in Directory Server. If you require secure connections, or the use of the attribute encryption feature, this attribute should be set to on.

Property	Value
Entry DN	`cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-security: off`

nsslapd-sizelimit (Size Limit)

Specifies the maximum number of entries to return from a search operation. If this limit is reached, `ns-slapd` returns any entries it has located that match the search request, as well as an exceeded size limit error.

When no limit is set, `ns-slapd` will return every matching entry to the client regardless of the number found. To set a no limit value whereby the Directory Server will wait indefinitely for the search to complete, specify a value of -1 for this attribute in the `dse.ldif` file.

This limit applies to everyone regardless of their organization.

Property	Value
Entry DN	`cn=config`
Valid Range	-1 to the maximum 32 bit integer value (2147483647)
Default Value	`2000`
Syntax	Integer
Example	`nsslapd-sizelimit: 2000`

nsslapd-threadnumber (Thread Number)

Defines the number of operation threads that the Directory Server will create during startup. The `nsslapd-threadnumber` value should be increased if you have many directory clients performing time-consuming operations such as add or modify. This ensures that there are other threads available for servicing short-lived operations such as simple searches.

Property	Value
Entry DN	`cn=config`
Valid Range	1 to the number of threads supported by your system
Default Value	`30`
Syntax	Integer
Example	`nsslapd-threadnumber: 60`

nsslapd-timelimit (Time Limit)

Specifies the maximum number of seconds allocated for a search request. If this limit is reached, Directory Server returns any entries it has located that match the search request, as well as an exceeded time limit error.

When no limit is set, `ns-slapd` will return every matching entry to the client regardless of the time it takes. To set a no limit value whereby Directory Server will wait indefinitely for the search to complete, specify a value of -1 for this attribute in the `dse.ldif` file. A value of zero (0) causes no time to be allowed for searches. The smallest time limit is 1 second.

Property	Value
Entry DN	`cn=config`
Valid range	-1 to the maximum 32 bit integer value (2147483647) in seconds
Default value	`3600`
Syntax	Integer
Example	`nsslapd-timelimit: 3600`

nsslapd-versionstring (Version String)

Specifies the server version number.

Property	Value
Entry DN	`cn=config`
Valid range	Any valid server version number.
Default value	N / A
Syntax	DirectoryString
Example	`nsslapd-versionstring:SunONE-Directory/5.2`

cn=changelog5

Multi-master replication changelog configuration entries are stored under the `cn=changelog5` entry. The replication changelog behaves much like a database. The `cn=changelog5,cn=config` entry is an instance of the `extensibleObject` object class. For attributes to be taken into account by the server, this object class (and the `top` object class) must be present in the entry.

It is worth noting that two different types of change logs are maintained by Sun ONE Directory Server 5.2. The first type, which is stored here and referred to as *changelog*, is used by multi-master replication; the second change log, which is actually a plug-in and referred to as *retro changelog*, is intended for use by Sun ONE Meta Directory. See “Retro Changelog Plug-In” on page 208 of Chapter 5, “Plug-In Implemented Server Functionality” for further information regarding the Retro Changelog Plug-in. Multi-master replication changelog attributes are presented in this section.

nsslapd-cachesize (Cache Size)

Specifies the replication changelog cache size, in terms of the number of entries it can hold. Note that it is simpler to limit the cache by memory size only (see the `nsslapd-cachememsize` attribute). If you attempt to set a value that is not an integer or is too big for a 64-bit unsigned integer (32-bit unsigned integer for 32-bit installations), you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

Property	Value
Entry DN	`cn=changelog5,cn=config`
Valid Range	1 to 2,147,483,647 (or -1 which means unlimited) entries
Default Value	`-1`
Syntax	Integer
Example	`nsslapd-cachesize: -1`

nsslapd-cachememsize (Cache Memory Size)

Specifies the changelog cache size, in terms of the available memory space. Limiting cachesize in terms of memory occupied is the simplest method. If automatic cache resizing is activated, this attribute is overridden. If you attempt to set a value that is not an integer or is too big for a 64-bit unsigned integer (32-bit unsigned integer for 32-bit installations), you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

Property	Value
Entry DN	`cn=changelog5,cn=config`
Valid Range	200KB to 2^{64} Bytes (200KB to 2^{32} Bytes for 32-bit installations)
Default Value	`10 485 760` (10Mb)
Syntax	Integer
Example	`nsslapd-cachememsize:10`

nsslapd-changelogdir (Changelog Directory)

This required attribute specifies the name of the directory in which the change log database will be created. Whenever a change log configuration entry is created it must contain a valid directory or the operation will be rejected. The GUI proposes by default that this database be stored under:

ServerRoot/slapd-*serverID*/changelogdb

NOTE For performance reasons, it is recommended that you store this database on a different physical disk.

Property	Value
Entry DN	cn=changelog5,cn=config
Valid Range	Any valid path to the directory storing the change log
Default Value	None
Syntax	DirectoryString
Example	nsslapd-changelogdir: /usr/myhome/slapd-local/changelogdb

nsslapd-changelogmaxage (Max Changelog Age)

Specifies the maximum age of any entry in the change log. The change log contains a record for each directory modification and is used when synchronizing consumer servers. Each record contains a timestamp. Any record with a timestamp that is older than the value specified in this attribute will be removed. If this attribute is absent, there is no age limit on change log records. For information on the change log, see “nsslapd-changelogdir.”

Property	Value
Entry DN	cn=changelog5,cn=config
Valid Range	0 (meaning that entries are not removed according to their age) to maximum integer (2147483647)
Default Value	0

Syntax	DirectoryString *IntegerAgeID* where AgeID is “s” for seconds, “m” for minutes, “h” for hours, “d” for days, or “w” for weeks.
Example	`nsslapd-changelogmaxage: 30d`

nsslapd-changelogmaxentries (Max Changelog Records)

Specifies the maximum number of records the change log may contain. If this attribute is absent, there is no maximum number of records the change log can contain. For information on the change log, see “nsslapd-changelogdir (Changelog Directory),” on page 134.

Property	Value
Entry DN	`cn=changelog5,cn=config`
Valid Range	0 (meaning that the only maximum limit is the disk size) to maximum integer (2147483647)
Default Value	`0`
Syntax	Integer
Example	`nsslapd-changelogmaxentries: 5000`

cn=encryption

Encryption related attributes are stored under the `cn=encryption,cn=config` entry. This entry is an instance of the `nsEncryptionConfig` object class. For encryption related attributes to be taken into account by the server, this object class (in addition to the `top` object class) must be present in the entry. Encryption configuration attributes are presented in this section.

nsSSLSessionTimeout

Specifies the lifetime duration of an SSL session for both SSLv2 and SSLv3. The minimum timeout value is 5 seconds and if you enter a value below this, it is automatically replaced by 5 seconds. Values outside the valid ranges are replaced by the default value of 100 seconds (SSLv2).

Property	Value

Entry DN	`cn=encryption,cn=config`
Valid Range	(SSLv2) 5 seconds to 100 seconds (SSLv3) 5 seconds to 24 hours
Default Value	`0` (which translates to 100 seconds if you are running SSLv2 and 24 hours if you are running SSLv3).
Syntax	Integer
Example	`nsSSLSessionTimeout: 5`

nsSSLClientAuth

In an SSL connection, this attribute specifies whether a client certificate is `allowed`, `required`, or should not be sent (`off`) to the SSL server.

Property	Value
Entry DN	`cn=encryption,cn=config`
Valid Range	off | allowed | required
Default Value	`allowed`
Syntax	DirectoryString
Example	`nsSSLClientAuth: allowed`

nsSSLServerAuth

Specifies the action that the SSL client should take on the server certificate sent by the SSL server in an SSL connection.

Property	Value
Entry DN	`cn=encryption,cn=config`
Valid Range	weak - make no attempt to verify whether the server certificate is from a trusted certificate authority cert - verify whether the server certificate is from a trusted certificate authority cncheck - verify whether the server certificate is from a trusted certificate authority *and* verify the DN contained in the server certificate (to avoid man-in-the middle attacks on the server)

Default Value	`cert`
Syntax	DirectoryString
Example	`nsSSLServerAuth: cert`

nsSSL2 (SSL 2)

Supports SSL version 2.

Property	Value
Entry DN	`cn=encryption,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsSSL2: on`

nsSSL3 (SSL 3)

Supports SSL version 3.

Property	Value
Entry DN	`cn=encryption,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsSSL3: on`

nsSSL3ciphers

This multi-valued attribute specifies the set of encryption ciphers the Directory Server will use during SSL communications. For more information on the ciphers supported by the Directory Server, see Chapter 11, “Managing SSL”, in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	cn=encryption,cn=config

Valid Range	For domestic versions, any combination of the following:
	
	For SSLv3
	`rsa_null_md5`
	`rsa_rc4_128_md5`
	`rsa_rc4_40_md5`
	`rsa_rc2_40_md5`
	`rsa_des_sha`
	`rsa_fips_des_sha`
	`rsa_3des_sha`
	`rsa_fips_3des_sha`
	
	For TLS
	`tls_rsa_export1024_with_rc4_56_sha`
	`tls_rsa_export1024_with_des_cbc_sha`

Default Value	N / A
Syntax	DirectoryString + symbol to enable or - symbol to disable followed by the cipher(s). It is important to note that blank spaces are not allowed in the list of ciphers. To enable all ciphers (except `rsa_null_md5` which must be specifically called) you can specify `+all`.
Example	`nsslapd-SSL3ciphers:` `+RSA_NULL_MD5,+RC4_56_SHA,-RC4_56_SHA`

If you are using the Directory Server Console to set the cipher preferences, the values on the SSL 3.0 tab of the Cipher Preference dialog box correspond to the following:

Table 4-5 SSLv3 Ciphers

Cipher in Console	Corresponding SSLv3 Cipher
None	rsa_null_md5
RC4	rsa_rc4_128_md5
RC4 (Export)	rsa_rc4_40_md5
RC2(Export)	rsa_rc2_40_md5
DES	rsa_des_sha
DES (FIPS)	rsa_fips_des_sha
Triple-DES	rsa_3des_sha
Triple-DES (FIPS)	rsa_fips_3des_sha

If you are using the Directory Server Console to set the cipher preferences, the values on the TLS tab of the Cipher Preference dialog box correspond to the following:

Table 4-6 TLS Ciphers

Cipher in Console	Corresponding TLS Cipher
RC4 (Export)	tls_rsa_export1024_with_rc4_56_sha
DES (Export)	tls_rsa_export1024_with_des_cbc_sha

cn=features

The cn=features,cn=config entry is an instance of the nsContainer object class. Configuration attributes for the filtering service (used by the partial replication feature) are stored here, under the cn=filtering service,cn=features,cn=config entry. The filtering service subtree contains two nodes: cn=sets and cn=elements.rlo

cn=elements contains all defined filtering units. A filtering unit is the minimum filtering concept that the filtering service can understand in a particular subtree.

cn=sets contains combinations and unions of the filtering units under cn=elements to extend the filtering definition.

For more information on the filtering service, see the *Sun ONE Directory Server Administration Guide.*

cn=mapping tree

Configuration attributes for suffixes and replication are stored under `cn=mapping tree,cn=config`. Configuration attributes related to suffixes are found under the suffix subentry

`cn="`*suffixName*`",cn=mapping tree,cn=config.`

Replication configuration attributes are stored under

`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config.`

Replication agreement attributes are stored under

`cn=`*replicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree, cn=config.`

Suffix Configuration Attributes Under cn="*suffixName*"

Suffix configuration attributes are stored under the `cn="suffixName"` entry, for example `cn="dc=example,dc=com"`. This entry is an instance of the `nsMappingTree` object class, which inherits from the `extensibleObject` object class. For suffix configuration attributes to be taken into account by the server, these object classes (in addition to the `top` object class) must be present in the entry. Suffix configuration attributes are presented in this section.

nsslapd-backend

Gives the name of the suffix or chained suffix used to process requests. This attribute can be multi-valued if you are using a custom distribution plug-in, with one suffix name per value. In this case, you must also specify the `nsslapd-distribution-plugin` and `nsslapd-distribution-funct` attributes.

This attribute is required when the value of the `nsslapd-state` attribute is set to `backend` or `referral on update`.

Property	Value
Entry DN	`cn="`*suffixName*`",cn=mapping tree,cn=config`

Valid Range	Any valid partition name.
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-backend: NetscapeRoot`

nsslapd-distribution-plugin

Specifies the full path and filename of the shared library for the custom distribution plugin. This attribute is required along with `nsslapd-distribution-funct` when you have specified more than one suffix in the `nsslapd-backend` attribute.

Contact Sun ONE Professional Services for information on how to create distribution logic for your directory server.

Property	Value
Entry DN	`cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	The full path and filename of the plug-in library.
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-distribution-plugin:` *ServerRoot*`/plugins/custom/myDistrib.so`

NOTE Once you have distributed entries, you cannot redistribute them. The following restrictions apply:

- You cannot change your distribution function once you have deployed entry distribution.
- You cannot use the LDAP `modrDN` or `ldapmodify` commands to change an entry if that would cause them to be distributed into a different database.
- You cannot replicate databases that are distributed over multiple databases.

Violating these restrictions prevents Sun ONE Directory Server from correctly locating and returning entries.

nsslapd-distribution-funct

Specifies the name of your distribution function within the library named by nsslapd-distribution-plugin. This attribute is required along with nsslapd-distribution-plugin when you have specified more than one database in the nsslapd-backend attribute.

Contact Sun ONE Professional Services for information on how to create distribution logic for your directory server.

Property	Value
Entry DN	`cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	The name of the distribution function.
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-distribution-funct: alphabeticalDistrib`

nsslapd-referral

Lists the servers to which updates are referred. This attribute can be multi-valued, with one server per value. This attribute is required when the value of the `nsslapd-state` attribute is set to `referral`.

Property	Value
Entry DN	`cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Any valid LDAP URL.
Default Value	Defined by the Replication Agreement.
Syntax	DirectoryString
Example	`nsslapd-referral: ldap://myServer.example.com:389`

nsslapd-state

Determines how the suffix handles operations.

Property	Value
Entry DN	`cn="`*suffixName*`",cn=mapping tree,cn=config`

Valid Range	Backend = the backend (database) is used to process all operations. Disabled = the database is not available for processing operations. The server returns a “No such search object” error in response to requests made by client applications. Referral = a referral is returned for requests made to this suffix. Referral on update = the database is used for all operations except update requests, which receive a referral.
Default Value	`backend`
Syntax	DirectoryString
Example	`nsslapd-state: backend`

Replication Attributes Under cn=replica, cn="*suffixName*",cn=mapping tree,cn=config

Replication configuration attributes are stored under

`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config.`

The `cn=replica` entry is an instance of the `nsDS5Replica` object class. For replication configuration attributes to be taken into account by the server, this object class (in addition to the `top` object class) must be present in the entry. Replication configuration attributes are presented in this section. For further information regarding replication, see Chapter 8, “Managing Replication” in the *Sun ONE Directory Server Administration Guide*.

cn

This attribute is used to name the replica. Once it has been set, it cannot be modified.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Any valid suffix name.
Default Value	`cn=replica`
Syntax	DirectoryString
Example	`cn: "cn=replica"`

nsDS5Flags

This attribute enables you to specify replica properties you have previously defined in flags. At present only two flags exist. One enables you to specify whether changes are logged. The second enables you to overwrite automatic referrals.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Changelog activation 0 = no changes are logged 1 = changes are logged Overwriting referrals 0 = automatic referrals are not overwritten 1 = automatic referrals are overwritten
Default Value	`0` (no changes are logged) `0` (automatic referrals are not overwritten)
Syntax	Integer
Example	`nsDS5Flags: 0`

nsDS5ReplicaBindDN

This multi-valued attribute specifies the DN to use when binding. The value can either be the DN of the local entry on the consumer server or, in the case of an SSL connection, the certificate identity associated with the same DN.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Any valid DN.
Default Value	`cn=replication manager, cn=replication,cn=config`
Syntax	DirectoryString
Example	`nsDS5ReplicaBindDN: cn=replication manager, cn=replication,cn=config`

nsDS5ReplicaChangeCount (Replica Change Count)

This read-only attribute informs you of the total number of entries in the change log (whether they still remain to be replicated or not). When the change log is purged, only the entries that are still to be replicated are left. See "nsDS5ReplicaPurgeDelay," on page 146 and "nsDS5ReplicaTombstonePurgeInterval," on page 147 for more information regarding purge operation properties.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	-1 to maximum integer (2147483647)
Default Value	N / A
Syntax	Integer
Example	`nsDS5ReplicaChangeCount: 675`

nsDS5ReplicaId (Replica ID)

Specifies the unique ID for masters in a given replication environment. Consumer services always have the same replica id : 65535.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	1 to 65534
Default Value	N / A
Syntax	Integer
Example	`nsDS5ReplicaId: 1`

nsDS5ReplicaLegacyConsumer

If this attribute is absent or has a value of `false`, then the replica is not a legacy consumer.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`

Valid Range	true | false
Default Value	`false`
Syntax	DirectoryString
Example	`nsDS5ReplicaLegacyConsumer: false`

nsDS5ReplicaName

This read-only attribute specifies the name of the replica with a unique identifier for internal operations. This unique identifier is allocated by the server when the replica is created. This attribute is for internal use only.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	N / A
Default Value	N / A
Syntax	DirectoryString (a UID identifies the replica)
Example	`nsDS5ReplicaName: 66a2b699-1dd211b2-807fa9c3-a58714648`

nsDS5ReplicaPurgeDelay

Specifies the period of time in seconds after which internal purge operations will be performed on the change log. When setting this attribute, ensure that the purge delay is longer than the longest replication cycle in your replication policy, to avoid incurring conflict resolution problems and server divergence.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	0 (keep forever) to maximum integer (2147483647)
Default Value	`604800` (1 week : 60x60x24x7)
Syntax	Integer
Example	`nsDS5ReplicaPurgeDelay: 604800`

nsDS5ReplicaReferral

This multi-valued attribute specifies the user-defined referrals. This should be defined on a consumer only. User referrals are only returned when a client attempts to modify data on a read-only consumer.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Any valid LDAP URL.
Default Value	N / A
Syntax	DirectoryString
Example	`nsDS5ReplicaReferral: ldap://ldap.aceindustry.com`

nsDS5ReplicaRoot

Specifies the DN at the root of a replicated area. This attribute must have the same value as the suffix of the database being replicated. It cannot be modified.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Suffix of the database being replicated.
Default Value	N / A
Syntax	DirectoryString
Example	`nsDS5ReplicaRoot: "dc=example,dc=com"`

nsDS5ReplicaTombstonePurgeInterval

Specifies the time interval in seconds between purge operation cycles. When setting this attribute, bear in mind that the purge operation is time consuming.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	0 to maximum integer (2147483647) in seconds
Default Value	`3600` (1 hour)

Syntax	Integer
Example	`nsDS5ReplicaTombstonePurgeInterval: 3600`

nsDS5ReplicaType

Defines the type of replication relationship that exists between this replica and the others.

Property	Value
Entry DN	`cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	0 = unknown 1 = primary (not yet used) 2 = consumer (read-only) 3 = consumer / supplier (updateable)
Default Value	N / A
Syntax	Integer
Example	`nsDS5ReplicaType: 2`

Replication Attributes Under cn=ReplicationAgreementName,cn=replica, cn="*suffixName*", cn=mapping tree,cn=config

The replication attributes that concern the replication agreement are stored under

cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*",cn=mapping tree,cn=config.

The cn=*ReplicationAgreementName* entry is an instance of the nsDS5ReplicationAgreement object class. For replication agreement configuration attributes to be taken into account by the server, this object class (in addition to the top object class) must be present in the entry. Replication agreements are configured only on supplier replicas. The replication agreement configuration attributes are presented in this section.

cn

This attribute defines the replication agreement name. Once this attribute has been set it cannot be modified.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	Any valid suffix name.
Default Value	cn=replica
Syntax	DirectoryString
Example	cn: "cn=ReplicationAgreement1"

description

Free form text description of the replication agreement. This attribute can be modified.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	Any string.

Default Value	N / A
Syntax	DirectoryString
Example	`description: Replication Agreement between Server A and Server B.`

ds5AgreementEnable

Specifies whether a replication agreement is enabled or disabled.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	on | off
Default Value	on
Syntax	DirectoryString
Example	`ds5agreementEnable: on`

ds5BeginReplicaAcceptUpdates

Enables you to specify that the replica should accept client updates instead of referring them.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	stop | start
Default Value	N / A
Syntax	DirectoryString
Example	`ds5BeginReplicaAcceptUpdates: start`

ds5ReferralDelayAfterInit

Enables you to specify the delay after which a recently initialized replica will start accepting client updates instead of referring them.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	0 to any 64-bit integer (seconds)
Default Value	0 (infinite)
Syntax	DirectoryString
Example	`ds5ReferralDelayAfterInit: 100`

ds5ReplicaAutomaticInit

An On / Off flag that enables a consumer that is out of sync to be reinitialized automatically.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Any string.
Default Value	N / A
Syntax	DirectoryString
Example	`ds5ReplicaAutomaticInit: on`

ds5ReplicaChangesSentDuringLastUpdate

This read-only attribute specifies the number of entries that were replicated in the last update session.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	N / A

Default Value	N / A
Syntax	Integer
Example	`ds5ReplicaChangesSentDuringLastUpdate: 0`

ds5ReplicaPendingChanges

This read-only attribute lists the changes not yet sent to the specified consumer. The attribute must be specifically requested in an `ldapsearch` operation. If the `ds5agreementEnable` attribute is set to `off`, this information is returned in an `ldapsearch` operation on `ds5ReplicaPendingChanges`.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	N / A.
Default Value	N / A
Syntax	DirectoryString
Example	`ds5ReplicaPendingChanges: DEL`

ds5ReplicaPendingChangesCount

This read-only attribute provides the number of changes not yet sent to the specified consumer. The attribute must be specifically requested in an `ldapsearch` operation. If the `ds5agreementEnable` attribute is set to `off`, this information is returned in an `ldapsearch` operation on `ds5ReplicaPendingChangesCount`.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	N / A
Default Value	N / A
Syntax	Integer
Example	`ds5ReplicaPendingChangesCount: 2`

ds5ReplicaTransportCompressionLevel

Available on Solaris and Linux platforms only, this attribute specifies the level of compression used in transporting updates to a consumer.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	0-3 0 = No compression 1 = Default Zlib compression (Zlib numeric value = -1) 2 = Best speed (Zlib numeric value = 1) 3 = Best compression (Zlib numeric value = 9)
Default Value	`0`
Syntax	Integer
Example	`ds5ReplicaTransportCompressionLevel: 0`

ds5ReplicaTransportGroupSize

The number of updates (for an incremental update) or entries (for a total update) that the supplier will group together before sending the changes to the consumer.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	`0 to 100`
Default Value	`1`
Syntax	Integer
Example	`ds5ReplicaTransportGroupSize: 1`

ds5ReplicaTransportWindowSize

The number of updates (for an incremental update) or entries (for a total update) that the supplier will send before waiting for a reply from the consumer.

Property	Value

Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	1 to 1000
Default Value	10
Syntax	Integer
Example	ds5ReplicaTransportWindowSize: 10

filterSPConfChecksum

The checksum for partial replication configuration.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="suffixName", cn=mapping tree,cn=config
Valid Range	Any string.
Default Value	N / A
Syntax	DirectoryString
Example	filterspconfchecksum:

filterSPConfDefinition

This single-valued attribute may contain any AND or OR combination of any number of Configuration Elements entries located in the configuration directory. The value of this attribute must conform to the following syntax:

```
filterSPConfDefinition: SUBSET(1) || SUBSET(2) || .... || SUBSET(N)
```

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	Any string.
Default Value	N / A
Syntax	DirectoryString
Example	filterSPConfDefinition: (people_subtree && any && include_cn_sn) || (group_subtree && include_cn)

filterSPConfEnabled

Activates or deactivates a specified Configuration Set without the need to remove the complete definition. If this attribute is set to `off`, clients are unable to use the Configuration Set to apply any kind of filtering.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	on | off
Default Value	on
Syntax	DirectoryString
Example	`filterspconfenabled: on`

filterSPFrcAttr

If the `filterSPType` attribute is set to `fractional_include`, this attribute contains the list of attributes to be included for replication.

If the `filterSPType` attribute is set to `fractional_exclude`, this attribute contains the list of attributes to be excluded for replication.

filterSPType

Specifies the type of partial replication.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	`fractional_include` : specifies a global list of included attributes that will be applied unconditionally to all entries in the database. `fractional_exclude` : specifies a global list of excluded attributes that will be applied to all entries in the database.
Default Value	N / A
Syntax	DirectoryString
Example	`filterSPType: fractional_include`

nsDS5BeginReplicaRefresh

Allows you to initialize a replica. This attribute is absent by default. However, if you add this attribute with a value of `start`, the server reinitializes the replica and removes the attribute value.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	stop | start
Default Value	N/A
Syntax	DirectoryString
Example	`nsDS5BeginReplicaRefresh: start`

nsDS5ReplicaBindDN

Specifies the DN to use when binding. The value of this attribute must be the same as the one in `cn=replica` on the consumer replica. A default DN of `"cn=replication manager"` is created when you set up a replication agreement. This can be modified. This attribute may be empty if certificate-based authentication is used.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Any valid DN.
Default Value	`cn=replication manager,cn=replication,cn=config`
Syntax	DirectoryString
Example	`nsDS5ReplicaBindDN: cn=replication manager, cn=replication,cn=config`

nsDS5ReplicaBindMethod

Specifies the method to use for binding. This attribute can be modified.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	SIMPLE (This bind method requires a DN and password) SSLCLIENTAUTH
Default Value	`SIMPLE`
Syntax	DirectoryString
Example	`nsDS5ReplicaBindMethod: SIMPLE`

nsDS5ReplicaChangesSentSinceStartup

This read-only attribute provides you with the number of changes sent to this replica since the server started.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	`0 to maximum integer (2147483647)`
Default Value	N / A
Syntax	Integer
Example	`nsDS5ReplicaChangesSentSinceStartup:` 647

nsDS5ReplicaCredentials

Specifies the credentials for the bind DN (specified in the `nsDS5ReplicaBindDN` attribute) on the remote server containing the consumer replica. The value for this attribute can be modified. When certificate-based authentication is used, this attribute may not have a value. The example below is what you view, not what you type.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	Any valid password that will be encrypted using the DES reversible password encryption schema.
Default Value	N / A
Syntax	DirectoryString `{DES}` *encrypted_password*
Example	`nsDS5ReplicaCredentials: {DES} 9Eko69APCJfFReplica`

nsDS5ReplicaHost

Specifies the hostname for the remote server containing the consumer replica. Once this attribute has been set it cannot be modified.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	Any valid host server name.
Default Value	N / A
Syntax	DirectoryString
Example	`nsDS5ReplicaHost: MyServer`

nsDS5ReplicaLastInitEnd

This optional, read-only attribute states when the initialization of the consumer replica ended.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`
Valid Range	N / A
Default Value	N / A
Syntax	GeneralizedTime
Example	`nsDS5ReplicaLastInitEnd: YYYYMMDDhhmmssZ (19711223113229)`

nsDS5ReplicaLastInitStart

This optional, read-only attribute states when the initialization of the consumer replica started.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`",cn=mapping tree,cn=config`

Valid Range	N / A
Default Value	N / A
Syntax	GeneralizedTime
Example	`nsDS5ReplicaLastInitStart: YYYYMMDDhhmmssZ (20000902160000)`

nsDS5ReplicaLastInitStatus

This optional, read-only attribute provides status for the initialization of the consumer.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	0 (Consumer Initialization Succeeded) followed by any other status message.
Default Value	N / A
Syntax	String
Example	`nsDS5ReplicaLastUpdateStatus: 0 Consumer Initialization Succeeded`

nsDS5ReplicaLastUpdateEnd

This read-only attribute states when the most recent replication schedule update ended.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	0 (Consumer Initialization succeeded.)
Default Value	N / A
Syntax	GeneralizedTime
Example	`nsDS5ReplicaLastUpdateEnd: YYYYMMDDhhmmssZ (20000902160000)`

nsDS5ReplicaLastUpdateStart

This read-only attribute states when the most recent replication schedule update started.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	N / A
Default Value	N / A
Syntax	GeneralizedTime
Example	nsDS5ReplicaLastUpdateStart: YYYYMMDDhhmmssZ (20000902160000)

nsDS5ReplicaLastUpdateStatus

This read-only attribute provides the status for the most recent replication schedule updates.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	0 (no replication sessions started) followed by any other error or status message.
Default Value	N / A
Syntax	DirectoryString
Example	nsDS5ReplicaLastUpdateStatus: 0 replica acquired successfully

nsDS5ReplicaPort

Specifies the port number for the remote server containing the replica. Once this attribute has been set, it cannot be modified.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	Port number for the remote server containing the replica.
Default Value	N / A
Syntax	Integer
Example	`nsDS5ReplicaPort:` 389

nsDS5ReplicaRoot

Specifies the DN at the root of a replicated area. This attribute must have the same value as the suffix of the database being replicated. It cannot be modified.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	Suffix of the database being replicated.
Default Value	N / A
Syntax	DirectoryString
Example	`nsDS5ReplicaRoot: "dc=example,dc=com"`

nsDS5ReplicaTimeout

This allowed attribute specifies the number of seconds outbound LDAP operations will wait for a response from the remote replica before timing out and failing. If you see "`Warning: timed out waiting`" messages in the error log file, then you should increase the value of this attribute.

You can find out the amount of time the operation actually lasted by examining the access log on the remote machine. You can then set the `nsDS5ReplicaTimout` attribute accordingly to optimize performance.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	0 to maximum integer value (2147483647) in seconds
Default Value	600
Syntax	Integer
Example	nsDS5ReplicaTimeout: 600

nsDS5ReplicaTransportInfo

Specifies the type of transport used for transporting data to and from the replica. The attribute values can either be SSL, which means that the connection is established over SSL, or LDAP, which means that regular LDAP connections are used. If this attribute is absent, regular LDAP connections are used. This attribute cannot be modified once set.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	SSL | LDAP
Default Value	LDAP
Syntax	DirectoryString
Example	nsDS5ReplicaTransportInfo: LDAP

nsDS5ReplicaUpdateInProgress

This read-only attribute states whether or not a replication schedule update is in progress.

Property	Value
Entry DN	cn=*ReplicationAgreementName*,cn=replica,cn="*suffixName*", cn=mapping tree,cn=config
Valid Range	true | false
Default Value	N/A

Syntax	DirectoryString
Example	`nsDS5ReplicaUpdateInProgress:true`

nsDS5ReplicaUpdateSchedule

This multi-valued attribute specifies the replication schedule. It can be modified.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	Time schedule presented as XXXX-YYYY 0123456 where XXXX is the starting hour, YYYY is the finishing hour and the numbers 0123456 are the days of the week starting with Sunday.
Default Value	`0000-2359 0123456` (all the time)
Syntax	Integer
Example	`nsDS5ReplicaUpdateSchedule: 0000-2359 0123456`

nsDS50ruv

This attribute is responsible for managing the internal state of the replica via the replication update vector. It is always present and must not be changed.

partialReplConfiguration

Specifies the partial replication configuration entry point, as defined in the Replication Agreement. The value of this attribute is the RDN of the `cn=sets, cn=filtering service,cn=features,cn=config` entry, which stores the filtering information required by the partial replication module.

Property	Value
Entry DN	`cn=`*ReplicationAgreementName*`,cn=replica,cn="`*suffixName*`", cn=mapping tree,cn=config`
Valid Range	Any valid DN
Default Value	`cn=sets, cn=filtering service,cn=features,cn=config`
Syntax	DN
Example	`partialReplConfiguration: include_people_cn`

cn=Password Policy

Configurable password policy attributes are stored under `cn=Password Policy,cn=config`. For a description of the operational or state attributes related to password policy, refer to “Operational Attributes,” on page 477.

Configurable password attributes fall into one of the following categories:

- attributes that determine the password policy itself
- attributes that determine the account lockout policy

NOTE In previous versions of Directory Server, configurable password policy attributes were stored directly under `cn=config`.

Password Policy Attributes

The following attributes determine the password policy.

passwordChange (Password Change)

Indicates whether users may change their passwords. If this attribute is not present, a value of `on` is assumed (users can change their passwords).

For more information on password policies, see Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	`on` | `off`
Default Value	`on`
Syntax	DirectoryString
Example	`passwordChange: on`

passwordCheckSyntax (Check Password Syntax)

Indicates whether the password syntax will be checked before the password is saved. The password syntax checking mechanism checks that the password meets the password minimum length requirement and that the string does not contain any "trivial" words, such as the user's name or user ID or any attribute value stored in the `uid`, `cn`, `sn`, `givenName`, `ou`, or `mail` attributes of the user's directory entry.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`passwordCheckSyntax: off`

passwordExp (Password Expiration)

Indicates whether user passwords will expire after a given number of seconds. By default, user passwords do not expire. If password expiration is enabled, you can set the number of seconds after which the password will expire using the `passwordMaxAge` attribute.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`passwordExp: on`

passwordExpireWithoutWarning (Password Expire Without Warning)

Indicates whether a password can expire regardless of whether the user was warned about the expiration date.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`passwordExpireWithoutWarning: on`

passwordInHistory (Number of Passwords to Remember)

Indicates the number of passwords the Directory Server stores in history. Passwords that are stored in history cannot be reused by users. The password history feature is disabled by default (the `passwordInHistory` attribute has a value of `0`). This implies that the Directory Server does not store any old passwords and users can reuse passwords.

To prevent users from rapidly cycling through a number of passwords, use the `passwordMinAge` attribute.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	0 to 24 passwords
Default Value	`0`
Syntax	Integer
Example	`passwordInHistory: 6`

passwordMaxAge (Password Maximum Age)

Indicates the number of seconds after which user passwords will expire. To use this attribute, you must enable password expiration using the `passwordExp` attribute.

For more information on password policies, see Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	`8640000` (100 days)
Syntax	Integer
Example	`passwordMaxAge: 100`

passwordMinAge (Password Minimum Age)

Specifies the number of seconds that must elapse between password modifications. Use this attribute in conjunction with the `passwordInHistory` attribute to prevent users from quickly cycling through passwords so that they can use their old password again. A value of zero (0) indicates that the user can change the password immediately.

For more information on password policies, see Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	0 to 2147472000 seconds (24,855 days)
Default Value	`0`
Syntax	Integer
Example	`passwordMinAge: 86400`

passwordMinLength (Password Minimum Length)

Specifies the minimum number of characters that must be used in a password. Syntax checking is performed against this attribute, if the `passwordCheckSyntax` attribute is set to `on`.

For more information on password policies, see Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	2 to 512 characters
Default Value	6
Syntax	Integer
Example	`passwordMinLength: 6`

passwordMustChange (Password Must Change)

Indicates whether users must change their passwords when they first bind to the Directory Server, or when the password has been reset by the administrator. If this attribute is set to on, users are required to change their passwords.

For users to be able to change their passwords, the `passwordChange` attribute must also be set to on.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`passwordMustChange: off`

passwordRootDNMayBypassModsChecks

Allows the root DN to modify passwords, even if the modification violates the password policy.

When this attribute is set to on, the Directory Manager can make modifications to passwords that violate the password policy. This allows exceptions to the password policy, and can be used, for example, in the case of applications that reset passwords to the same default value. If the Directory Manager changes a password and the server detects that the new password violates the minimum length or the password history, a warning is logged, but the modification proceeds.

This attribute is set to `off` by default, which means that the server rejects password modifications by the Directory Manager if they violate the password policy.

For more information on password policies, see Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`passwordRootdnMayBypassModsChecks: off`

passwordStorageScheme (Password Storage Scheme)

Specifies the algorithm used to encrypt Directory Server passwords. The default password storage scheme is the Salted Secure Hash Algorithm (SSHA).

The following encryption types are supported by Directory Server 5.2:

- SSHA (Salted Secure Hash Algorithm) is the recommended method as it is the most secure.
- SHA (Secure Hash Algorithm). This is the method supported by 4.x Directory Servers.
- CRYPT is the UNIX crypt algorithm. It is provided for compatibility with UNIX passwords.

If this attribute is set to `CLEAR`, passwords are not encrypted and appear in plain text.

You can modify how the Directory Server stores password attributes by writing your own password storage scheme plug-in. For more information see Chapter 11, "Writing Password Storage Scheme Plug-Ins" in the *Sun ONE Directory Server Plug-In API Programming Guide*.

NOTE You can no longer choose to encrypt passwords using the NS-MTA-MD5 password storage scheme. The storage scheme is still present but only for backward compatibility.

For more information on password policies, see Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide.*

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid range	Any of the following password storage schema: SSHA | SHA | CRYPT | CLEAR
Default value	SSHA
Syntax	DirectoryString
Example	`passwordStorageScheme: SSHA`

passwordWarning (Send Warning)

Specifies the number of seconds before a user’s password expires, that a warning is sent. The user will receive a password expiration warning on attempting to authenticate to the directory. Depending on the LDAP client, the user may also be prompted to change their password at the time the warning is sent.

If this attribute is not present, or if the value of the attribute is `0`, no warning messages are sent. For password expiration to be enabled, the `passwordExp` attribute must be set to on.

For more information on password policies, see Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide.*

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	`86400` (1 day)
Syntax	Integer
Example	`passwordWarning: 86400`

Account Lockout Attributes

The following attributes determine the account lockout policy.

passwordLockout (Account Lockout)

Enables the account lockout mechanism. If this attribute is set to `on`, users will be locked out of the directory (for the length of time specified in the `passwordLockoutDuration` attribute) once the maximum number of consecutive failed bind attempts has been reached. The maximum number of consecutive bind attempts is specified by the `passwordMaxFailure` attribute.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`passwordLockout: off`

passwordLockoutDuration (Lockout Duration)

If the account lockout feature is enabled (`passwordLockout` is set to `on`), this attribute specifies the length of time (in seconds) during which users will be locked out of the directory. The account is locked when the maximum number of consecutive failed bind attempts (specified by `passwordMaxFailure`) has been reached.

If this attribute is not present, or if it is set to `0`, the account will remain locked until it is reset by the administrator.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	`3600`

Syntax	Integer
Example	`passwordLockoutDuration: 3600`

passwordMaxFailure (Maximum Password Failures)

If the account lockout feature is enabled (`passwordLockout` is set to `on`), this attribute specifies the number of consecutive failed bind attempts after which a user will be locked out of the directory. Each time an invalid password is sent from the user's account, the password failure counter is incremented. The value of this counter is stored in the operational attribute, `passwordRetryCount`.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	1 to 32767
Default Value	3
Syntax	Integer
Example	`passwordMaxFailure: 3`

passwordResetFailureCount (Reset Password Failure Counter)

Each time an invalid password is sent from the user's account, the password failure counter is incremented. The value of this counter is stored in the operational attribute, `passwordRetryCount`. This attribute specifies the length of time (in seconds) after which `passwordRetryCount` is reset to `0` (even if no successful authentication occurs).

If `passwordResetFailureCount` is set to `0`, the failure counter is reset only when a successful bind occurs.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=Password Policy,cn=config`
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds

Default Value	600
Syntax	Integer
Example	passwordResetFailureCount: 600

passwordUnlock (Unlock Account)

If the account lockout mechanism is enabled, (passwordLockout is set to on), this attribute specifies whether user accounts will be unlocked after a period of time. The period of time is specified in the passwordLockoutDuration attribute.

If passwordUnlock is set to on and the value of the passwordMaxFailure attribute has been reached, the account will be unlocked after the number of seconds specified in the passwordLockoutDuration attribute. However, if passwordUnlock is set to off, and the value of the passwordMaxFailure attribute has been reached, the account will remain locked until the administrator resets it.

For more information on password policies, see Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	cn=Password Policy,cn=config
Valid Range	on | off
Default Value	on
Syntax	DirectoryString
Example	passwordUnlock: off

cn=replication

A default replication bind DN (cn=replication manager) is created when you set up a replication agreement. This can be modified.

When configuring legacy replication, configuration attributes are stored under this cn=replication,cn=config node, which serves as a placeholder.

cn=SNMP

SNMP configuration attributes are stored under `cn=SNMP,cn=config`. The `cn=SNMP` entry is an instance of the `nsSNMP` object class. For SNMP configuration attributes to be taken into account by the server, this object class (in addition to the `top` object class) must be present in the entry. SNMP configuration attributes are presented in this section.

nssnmpenabled

Specifies whether SNMP is enabled or not.

Property	Value
Entry DN	`cn=SNMP,cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nssnmpenabled: off`

nssnmporganization

Specifies the organization to which the Directory Server belongs.

Property	Value
Entry DN	`cn=SNMP,cn=config`
Valid Range	Organization name
Default Value	N / A
Syntax	DirectoryString
Example	`nssnmporganization: Sun ONE`

nssnmplocation

Specifies the location within the company or organization where the Directory Server resides.

Property	Value
Entry DN	cn=SNMP,cn=config
Valid Range	Location
Default Value	N / A
Syntax	DirectoryString
Example	`nssnmplocation: B14`

nssnmpcontact

Specifies the E-mail address of the person responsible for maintaining the Directory Server.

Property	Value
Entry DN	cn=SNMP,cn=config
Valid Range	Contact E-mail address
Default Value	N / A
Syntax	DirectoryString
Example	`nssnmpcontact: ITdept@example.com`

nssnmpdescription

Provides a unique description of the Directory Server instance.

Property	Value
Entry DN	cn=SNMP,cn=config
Valid Range	Description
Default Value	N / A
Syntax	DirectoryString
Example	`nssnmpdescription: Employee directory instance`

nssnmpmasterhost

This *required* attribute specifies the hostname of the machine on which the master agent is installed. For UNIX only.

Property	Value
Entry DN	`cn=SNMP,cn=config`
Valid Range	Machine hostname or local host.
Default Value	`localhost`
Syntax	DirectoryString
Example	`nssnmpmasterhost: localhost`

nssnmpmasterport

Specifies the port number used to communicate with the master agent. For UNIX only.

Property	Value
Entry DN	`cn=SNMP,cn=config`
Valid Range	Operating System dependent port number. Refer to your Operating System documentation for further information.
Default Value	`199`
Syntax	Integer
Example	`nssnmpmasterport: 199`

cn=tasks

No specific configuration attributes.

cn=uniqueid generator

The uniqueid generator configuration attributes are stored under `cn=uniqueid generator,cn=config`. The `cn=uniqueid generator` entry is an instance of the `extensibleObject` object class. For uniqueid generator configuration attributes to be taken into account by the server, this object class (in addition to the `top` object class) must be present in the entry. Uniqueid generator configuration attributes are presented in this section.

nsState

This attribute stores information on the state of the clock. It is intended for internal use only, to ensure that the server cannot generate a change sequence number (CSN) inferior to existing ones required for detecting backward clock errors. Do not edit this attribute.

Property	Value
Entry DN	`cn=uniqueid generator,cn=config`
Valid Range	N / A
Default Value	N / A
Syntax	DirectoryString
Example	`nsstate:AbId0c3oMIDUntiLCyYNGgAAAAAAAAAA`

Monitoring Attributes

Read-only monitoring information is stored under the `cn=monitor` entry.

cn=monitor

The `cn=monitor` entry is an instance of the `extensibleObject` object class. For `cn=monitor` configuration attributes to be taken into account by the server, this object class (in addition to the `top` object class) must be present in the entry. The `cn=monitor` read-only attributes are presented in this section.

backendMonitorDN

DN for each Directory Server backend.

For further database monitoring information, see “Database Monitoring Attributes” on page 231, “Database Performance Attributes” on page 236, “Database Monitoring Attributes Under cn=NetscapeRoot” on page 240, and “Chained Suffix Monitoring Attributes” on page 256.

bytesSent

Number of bytes sent by Directory Server.

cache-avail-bytes

The number of bytes available for caching.

connection

List of open connections given in the following format:

```
connection=31:20010201164808Z:45:45::cn=directory manager:LDAP
```

where `31` is the connection number, `20010201164808Z` is the date the connection was opened, `45` is the number of operations received, `45` is the number of completed operations, and `cn=directory manager` is the bind DN.

connectionPeak

Maximum number of simultaneous connections since server startup.

currentConnections

Number of current Directory Server connections.

currentTime

Current time usually given in Greenwich Mean Time (indicated by GeneralizedTime syntax `Z` notation, for example `20010202131102Z`).

dTableSize

Size of the Directory Server descriptor table.

entriesSent

Number of entries sent by Directory Server.

nbackEnds

Number of Directory Server backends.

opsCompleted

Number of Directory Server operations completed.

opsInitiated

Number of Directory Server operations initiated.

request-que-backlog

The number of requests waiting to be processed by a thread. Each request received by the server is accepted, then placed in a queue until a thread is available to process it. The queue backlog should always be small, (`0` or close to `0`). If the queue backlog is large, use the `nsslapd-threadnumber` attribute to increase the number of threads available in the server.

This attribute applies to UNIX and Linux only.

readWaiters

Number of connections where some requests are pending and not currently being serviced by a thread in Directory Server.

startTime

Directory Server start time.

threads

Number of operation threads Directory Server creates during startup. This attribute can be set using the `nsslapd-threadnumber (Thread Number)` attribute under `cn=config`. The `nsslapd-threadnumber` attribute is not present in the `dse.ldif` file by default, but can be added.

totalConnections

Total number of Directory Server connections.

version

Directory Server version and build number.

cn=disk,cn=monitor

The cn=disk entry enables you to monitor disk conditions over LDAP. This entry is an instance of the extensibleObject object class. A cn=*disknumber*,cn=disk,cn=monitor entry exists for each disk. The following disk monitoring attributes appear under each of these individual disk entries.

disk-dir

Specifies the pathname of a directory used by the server on disk. Where several database instances reside on the same disk or an instance refers to several directories on the same disk, the short pathname is displayed. The disk numbering is arbitrary.

disk-free

Indicates the amount of free disk space available to the server, in MB.

NOTE The disk space available to the server process may be less than the total free disk space. For example, on some platforms a process that is not running as superuser may not have all the free disk space available to it.

disk-state

Indicates the state of the disk, based on the available free space and on the thresholds set for disk low and disk full (with the configuration parameters nsslapd-disk-low-threshold and nsslapd-disk-full-threshold). Possible values are normal, low, and full.

cn=counters,cn=monitor

This entry holds counter information for the various subtree entry counter plug-ins, if they are enabled. For more information on these plug-ins, see “Subtree Entry Counter Plug-Ins” on page 209.

cn=snmp,cn=monitor

The `cn=snmp` entry enables you to monitor Directory Server access, operations, and errors. This entry is an instance of the `extensibleObject` object class.

addentryops

The number of add operations serviced by this directory since server startup.

anonymousbinds

The number of anonymous binds to the directory since server startup.

bindsecurityerrors

The number of bind requests that have been rejected by the directory due to authentication failures or invalid credentials since server startup.

bytesrecv

The number of bytes received by this directory since server startup.

bytessent

The number of bytes sent to clients by this directory since server startup.

cacheentries

The number of entries cached in the directory.

cachehits

The number of operations serviced from the locally held cache since application startup.

chainings

The number of chaining operations returned by this directory in response to client requests since server startup.

compareops

The number of compare operations serviced by this directory since server startup.

connections

The number of current open connections.

connectionseq

The number of connections handled by the directory since server startup.

copyentries

The number of directory entries for which this directory contains a consumer copy. The value of this object will always be 0 (as no updates are currently performed).

entriesreturned

The number of entries returned by this directory in response to client requests since server startup.

errors

The number of requests that could not be serviced due to errors (other than security or referral errors). Errors include name errors, update errors, attribute errors, and service errors. Partially serviced requests are not counted as errors.

inops

The number of operations forwarded to this directory from another directory since server startup.

listops

The number of list operations serviced by this directory since server startup. The value of this object will always be 0 because LDAP implements list operations indirectly via the search operation.

masterentries

The number of directory entries for which this directory contains the master entry. The value of this object will always be 0 (as no updates are currently performed).

modifyentryops

The number of modify operations serviced by this directory since server startup.

modifyrdnops

The number of modify RDN operations serviced by this directory since server startup.

onelevelsearchops

The number of one-level search operations serviced by this directory since server startup.

readops

The number of read operations serviced by this directory since application start. The value of this object will always be 0 because LDAP implements read operations indirectly via the search operation.

referrals

The number of referrals returned by this directory in response to client requests since server startup.

referralsreturned

The number of referrals returned by this directory in response to client requests since server startup.

removeentryops

The number of delete operations serviced by this directory since server startup.

searchops

The total number of search operations serviced by this directory since server startup.

securityerrors

The number of operations forwarded to this directory that did not meet security requirements.

simpleauthbinds

The number of binds to the directory that were established using a simple authentication method (such as password protection) since server startup.

slavehits

The number of operations that were serviced from locally held replications (shadow entries). The value of this object will always be 0.

strongauthbinds

The number of binds to the directory that were established using a strong authentication method (such as SSL or an SASL mechanism like Kerberos) since server startup.

unauthbinds

The number of unauthenticated binds to the directory since server startup.

wholesubtreesearchops

The number of whole subtree search operations serviced by this directory since server startup.

Configuration Quick Reference Tables

This section provides quick reference tables for LDIF configuration files supplied with the Directory Server, object classes and schema used in server configuration, and attributes requiring server restart.

LDIF Configuration Files

Table 4-7 on page 185 lists all the configuration files that are supplied with the Directory Server, including those for the schema of other Sun ONE and legacy servers. Each file is preceded by a number that indicates the order in which they should be loaded (in ascending numerical and then alphabetical order). See “LDIF Configuration Files - Location” on page 76 for information on where these files are stored.

Table 4-7 Directory Server Configuration LDIF Files

Configuration Filename	Purpose
`dse.ldif`	Contains front-end Directory Specific Entries created by the directory at server startup. These include the Root DSE (`""`), and the contents of `cn=config` and `cn=monitor`.

Table 4-7 Directory Server Configuration LDIF Files *(Continued)*

Configuration Filename	Purpose
`00core.ldif`	Contains LDAPv3 standard operational schema, such as "subschemaSubentry," the LDAPv3 standard user and organization schema defined in RFC 2256 (based on X.520 / X.521), inetOrgPerson and other widely-used attributes, and the operational attributes used by Sun ONE Directory Server 5.2 configuration. Modifying this file will cause interoperability problems. User defined attributes should be added using Sun ONE Server Console.
`05rfc2247.ldif`	Schema from RFC 2247 and related pilot schema: "Using Domains in LDAP / X500 Distinguished Names."
`05rfc2927.ldif`	Schema from RFC 2927: "MIME Directory Profile for LDAP Schema." Contains the `ldapSchemas` operational attribute required for the attribute to show up in the subschema subentry.
`11rfc2307.ldif`	Schema from RFC 2307: "An Approach for Using LDAP as a Network Information Service."
`20subscriber.ldif`	Contains new schema elements and the Nortel subscriber interoperability specification. Also contains the `adminRole` and `memberOf` attributes and `inetAdmin` object class previously stored in 50ns-delegated-admin.ldif file.
`25java-object.ldif`	Schema from RFC 2713: "Schema for Representing Java(tm) Objects in an LDAP Directory."
`28pilot.ldif`	Contains pilot directory schema from FRC 1274 that is no longer recommended for new deployments. Please note that future RFCs that succeed RFC 1274 may deprecate some or all of `28pilot.ldif` attribute types and classes.
`30ns-common.ldif`	Schema that contains objects classes and attributes common to the Sun ONE Server Console framework.
`50ns-admin.ldif`	Schema used by Sun ONE Administration Services.
`50ns-calendar.ldif`	Schema used by Sun ONE Calendar Server.
`50ns-certificate.ldif`	Schema for Sun ONE Certificate Management System.

Table 4-7 Directory Server Configuration LDIF Files *(Continued)*

Configuration Filename	Purpose
`50ns-compass.ldif`	Schema used by Netscape Compass Server to define personal interest profiles.
`50ns-delegated-admin.ldif`	Schema used by Delegated Administrator 4.5.
`50ns-directory.ldif`	Contains additional configuration schema used by Directory Server 4.12 and earlier versions of the directory, which is no longer applicable to Sun ONE Directory Server 5.2. This schema is required for replicating between Directory Server 4.12 and Sun ONE Directory Server 5.2.
`50ns-legacy.ldif`	Legacy schema used by Sun ONE Administration Server for legacy servers.
`50ns-mail.ldif`	Schema used by Sun ONE Messaging Server to define mail users and mail groups.
`50ns-mcd-browser.ldif`	Schema used by Mission Control Desktop to hold browser client preferences.
`50ns-mcd-config.ldif`	Schema used by Mission Control Desktop to set MCD "config()" preferences.
`50ns-mcd-li.ldif`	Schema used by Mission Control Desktop to define location independence.
`50ns-mcd-mail.ldif`	Schema used by Mission Control Desktop to hold mail client and messenger security preferences.
`50ns-media.ldif`	Schema used for Media Server.
`50ns-mlm.ldif`	Schema used by Messaging Server 4.0 for mailing list management.
`50ns-msg.ldif`	Schema used for Web Mail.
`50ns-netshare.ldif`	Schema used for Netshare.
`50ns-news.ldif`	Schema used for Collabra Server to hold news group preferences.
`50ns-proxy.ldif`	Schema used for Sun ONE Proxy Server.
`50ns-value.ldif`	Schema for Sun ONE servers' *value item* schema.
`50ns-wcal.ldif`	Schema for Sun ONE Web Calendaring.
`50ns-web.ldif`	Schema for Sun ONE Web Server.

Table 4-7 Directory Server Configuration LDIF Files *(Continued)*

Configuration Filename	Purpose
`99user.ldif`	User-defined schema maintained by Directory Server replication consumers that contains the attributes and object classes from the suppliers.

Configuration Changes Requiring Server Restart

Table 4-8 lists the configuration attributes that cannot take effect dynamically, while the server is still running. After modifying these parameters through the console or the `ldapmodify` command, the server must be stopped and restarted for them to take effect. The table lists the configuration attributes concerned, with their full DNs, and provides a brief description of their functions.

Table 4-8 Configuration Changes Requiring Server Restart

Configuration Attribute	Action Requiring Restart
cn=*suffixName*,cn=ldbm database,cn=plugins,cn=config: nsslapd-cachesize	Modifying the cachesize attribute.
cn=config,cn=ldbm database,cn=plugins,cn=config: nsslapd-dbcachesize	Modifying the dbcachesize attribute.
cn=config,cn=ldbm database,cn=plugins,cn=config: nsslapd-dbncache	Modifying the database cache.
cn=changelog5,cn=config:nsslapd-changelogdir	Modifying the change log directory.
cn=changelog5,cn=config:nsslapd-changelogmaxage	Modifying the maximum age limit of the change log.
cn=changelog5,cn=config:nsslapd-changelogmaxentries	Modifying the maximum number of entries supported by the change log.
cn=changelog5,cn=config:nsslapd-changelogsuffix	Modifying the change log suffix.
cn=config:nsslapd-port	Changing the port number.
cn=config:nsslapd-secureport	Changing the secure port number.
cn=changelog5,cn=config:nsslapd-db*	Modifying any of the changelog database parameters.
cn=encryption,cn=config:nsssl2	Enabling or disabling SSL Version 2 for Directory Server.
cn=encryption,cn=config:nsssl3	Enabling or disabling SSL Version 3 for Directory Server.

Table 4-8 Configuration Changes Requiring Server Restart *(Continued)*

Configuration Attribute	Action Requiring Restart
cn=encryption,cn=config:nssslclientauth	Enabling or disabling client authentication.
cn=encryption,cn=config:nssslsessiontimeout	Changing the lifetime of an SSL session.

Chapter 5

Plug-In Implemented Server Functionality

This chapter serves as a plug-in implemented server functionality reference and is divided into the following sections:

- Plug-In Overview
- Server Plug-In Functionality Reference
- Attributes Common to All Plug-Ins
- Attributes Allowed by Certain Plug-Ins
- Database Plug-In Attributes
- Chained Suffix Plug-In Attributes
- Frontend Plug-In Attributes
- Retro Changelog Plug-In Attributes
- Subtree Entry Counter Plug-In Attributes

Plug-In Overview

The configuration for each part of Directory Server plug-in functionality has its own separate entry and set of attributes under the subtree `cn=plugins,cn=config`. A second look at Code Example 3-2 (configuration entry for the Telephone Syntax plug-in) described in Chapter 3, “Core Server Configuration” shows some of the plug-in configuration attributes:

```
dn: cn=Telephone Syntax,cn=plugins,cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsSlapdPlugin
cn: Telephone Syntax
nsslapd-pluginPath: ServerRoot/lib/syntax-plugin.so
nsslapd-pluginInitfunc: tel_init
nsslapd-pluginType: syntax
nsslapd-pluginEnabled: on
```

Some of these attributes are common to all plug-ins while others may be particular to a specific plug-in. You can check which attributes are currently being used by a given plug-in by performing an `ldapsearch` on the `cn=config` subtree.

Object Classes for Plug-In Configuration

All plug-ins are instances of the `nsSlapdPlugin` object class, which in turn inherits from the `extensibleObject` object class. For plug-in configuration attributes to be taken into account by the server, both of these object classes (in addition to the `top` object class) must be present in the entry as shown in the following example:

```
dn:cn=ACL Plugin,cn=plugins,cn=config
objectclass:top
objectclass:extensibleObject
objectclass:nsSlapdPlugin
```

Server Plug-In Functionality Reference

The following tables provide an overview of the plug-ins provided with Sun ONE Directory Server 5.2, along with their configurable options, configurable arguments, default setting, dependencies, general performance related information, and further reading. These tables will enable you to compare plug-in performance gains and costs and choose the optimal settings for your deployment. A reference to additional information on the plug-ins is provided where this is available.

7-Bit Check Plug-In

Plug-In Name	7-Bit Check (NS7bitAttr)
DN of Config Entry	`cn=7-bit check,cn=plugins,cn=config`
Description	Checks certain attributes are 7-bit clean.
Configurable Options	on | off
Default Setting	`on`
Configurable Arguments	List of attributes (`uid mail userpassword`) followed by "," and then suffix(es) on which the check is to occur.
Dependencies	None
Performance Related Information	None
Further Information	If your Directory Server uses non-ASCII characters, for example, Japanese, turn this plug-in off.

ACL Plug-In

Plug-In Name	ACL Plugin
DN of Config Entry	`cn=ACL Plugin,cn=plugins,cn=config`
Description	ACL access check plug-in
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 6, "Managing Access Control" in the *Sun ONE Directory Server Administration Guide*.

ACL Preoperation Plug-In

Plug-In Name	ACL preoperation
DN of Config Entry	`cn=ACL preoperation,cn=plugins,cn=config`
Description	ACL access check plug-in.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	Database
Performance Related Information	It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 6, "Managing Access Control" in the *Sun ONE Directory Server Administration Guide*.

Binary Syntax Plug-In

Plug-In Name	Binary Syntax
DN of Config Entry	`cn=Binary Syntax,cn=plugins,cn=config`
Description	Syntax for handling binary data.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

Boolean Syntax Plug-In

Plug-In Name	Boolean Syntax
DN of Config Entry	`cn=Boolean Syntax,cn=plugins,cn=config`
Description	Syntax for handling booleans.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

Case Exact String Syntax Plug-In

Plug-In Name	Case Exact String Syntax
DN of Config Entry	`cn=Case Exact String Syntax,cn=plugins,cn=config`
Description	Syntax for handling case-sensitive strings.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

Case Ignore String Syntax Plug-In

Plug-In Name	Case Ignore String Syntax
DN of Config Entry	`cn=Case Ignore String Syntax,cn=plugins,cn=config`
Description	Syntax for handling case-insensitive strings.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

Chaining Database Plug-In

Plug-In Name	Chaining Database
DN of Config Entry	`cn=Chaining database,cn=plugins,cn=config`
Description	Syntax for handling DNs.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	"Creating Chained Suffixes" in Chapter 3 of the *Sun ONE Directory Server Administration Guide.*

Class of Service Plug-In

Plug-In Name	Class of Service
DN of Config Entry	`cn=Class of Service,cn=plugins,cn=config`
Description	Allows for sharing of attributes between entries.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 5, "Advanced Entry Management" in the *Sun ONE Directory Server Administration Guide.*

Country String Syntax Plug-In

Plug-In Name	Country String Syntax
DN of Config Entry	`cn=Country String Syntax,cn=plugins,cn=config`
Description	Syntax for handling countries.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

Distinguished Name Syntax Plug-In

Plug-In Name	Distinguished Name Syntax
DN of Config Entry	`cn=Distinguished Name Syntax,cn=plugins,cn=config`
Description	Syntax for handling DNs.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

DSML Frontend Syntax Plug-In

Plug-In Name	Frontends
DN of Config Entry	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Description	Enables you to access the directory using DSMLv2 over SOAP / HTTP.
Configurable Options	on | off
Default Setting	`off`
Configurable Arguments	ds-hdsml-soapschemalocation ds-hdsml-dsmlschemalocation
Dependencies	None
Performance Related Information	None

Generalized Time Syntax Plug-In

Plug-In Name	Generalized Time Syntax
DN of Config Entry	`cn=Generalized Time Syntax,cn=plugins,cn=config`
Description	Syntax for dealing with dates, times, and time zones.
Configurable Options	on | off
Default Setting	`on`
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	The Generalized Time String consists of the following: four digit year, two digit month (for example, 01 for January), two digit day, two digit hour, two digit minute, two digit second, an optional decimal part of a second and a time zone indication. We strongly recommend that you use the Z time zone indication (Greenwich Mean Time.)

Integer Syntax Plug-In

Plug-In Name	Integer Syntax
DN of Config Entry	`cn=Integer Syntax,cn=plugins,cn=config`
Description	Syntax for handling integers.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

Internationalization Plug-In

Plug-In Name	Internationalization Plugin
DN of Config Entry	`cn=Internationalization Plugin,cn=plugins,cn=config`
Description	Syntax for handling DNs.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None. In contrast to previous versions of Directory Server, the collation orders and locales used by the internationalization plug-in are now stored in the `dse.ldif` file.
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	See Appendix C, “Directory Internationalization.”

ldbm Database Plug-In

Plug-In Name	ldbm database plug-in
DN of Config Entry	`cn=ldbm database plug-in,cn=plugins,cn=config`
Description	Implements local databases.
Configurable Options	N/A
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	See "Database Plug-In Attributes" on page 217 for further information on database configuration. It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 2, "Creating Your Directory Tree" in the *Sun ONE Directory Server Administration Guide*.

Legacy Replication Plug-In

Plug-In Name	Legacy Replication plug-in
DN of Config Entry	`cn=Legacy Replication plug-in,cn=plugins,cn=config`
Description	Enables Sun ONE Directory Server 5.2 to be a consumer of a 4.x supplier.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None.
Dependencies	database
Performance Related Information	None
Further Information	This plug-in can be disabled if the server is not (and never will be) a consumer of a 4.x server. See Chapter 8, "Managing Replication" in the *Sun ONE Directory Server Administration Guide* for more information.

Multimaster Replication Plug-In

Plug-In Name	Multimaster Replication Plugin
DN of Config Entry	`cn=Multimaster Replication plugin,cn=plugins,cn=config`
Description	Enables replication between two 5.x Directory Servers.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	database
Performance Related Information	N/A
Further Information	You can turn this plug-in off if you have only one server, which will never replicate. See Chapter 8, "Managing Replication" in the *Sun ONE Directory Server Administration Guide* for more information.

Octet String Syntax Plug-In

Plug-In Name	Octet String Syntax
DN of Config Entry	`cn=Octet String Syntax,cn=plugins,cn=config`
Description	Syntax for handling octet strings.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

CLEAR Password Storage Plug-In

Plug-In Name	CLEAR
DN of Config Entry	`cn=CLEAR,cn=Password Storage Schemes,cn=plugins,cn=config`
Description	CLEAR password storage scheme used for password encryption.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide.*

CRYPT Password Storage Plug-In

Plug-In Name	CRYPT
DN of Config Entry	`cn=CRYPT,cn=Password Storage Schemes,cn=plugins,cn=config`
Description	CRYPT password storage scheme used for password encryption.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide.*

NS-MTA-MD5 Password Storage Scheme Plug-In

Plug-In Name	NS-MTA-MD5
DN of Config Entry	`cn=NS-MTA-MD5,cn=Password Storage Schemes,cn=plugins,cn=config`
Description	NS-MTA-MD5 password storage scheme for password encryption.
Configurable Options	on | off
Default Setting	`on`
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	You can no longer choose to encrypt passwords using the NS-MTA-MD5 password storage scheme. The storage scheme is still present, but for backward compatibility only (the data in your directory still contains passwords encrypted with the NS-MTA-MD5 password storage scheme.) See Chapter 7, “User Account Management” in the *Sun ONE Directory Server Administration Guide*.

SHA Password Storage Scheme Plug-In

Plug-In Name	SHA
DN of Config Entry	`cn=SHA,cn=Password Storage Schemes,cn=plugins,cn=config`
Description	SHA password storage scheme for password encryption.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	If there are no passwords encrypted using the SHA password storage scheme, you may turn this plug-in off. If you want to encrypt your password with the SHA password storage scheme, we recommend that you choose SSHA instead, as SSHA is a far more secure option.
Further Information	Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

SSHA Password Storage Scheme Plug-In

Plug-In Name	SSHA
DN of Config Entry	`cn=SSHA,cn=Password Storage Schemes,cn=plugins,cn=config`
Description	SSHA password storage scheme for password encryption.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 7, "User Account Management" in the *Sun ONE Directory Server Administration Guide*.

Postal Address String Syntax Plug-In

Plug-In Name	Postal Address Syntax
DN of Config Entry	`cn=Postal Address Syntax,cn=plugins,cn=config`
Description	Syntax used for handling postal addresses.
Configurable Options	on | off
Default Setting	`on`
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

PTA Plug-In

Plug-In Name	Pass Through Authentication
DN of Config Entry	`cn=Pass Through Authentication,cn=plugins,cn=config`
Description	Enables pass-through authentication, the mechanism that allows one directory to consult another to authenticate bind requests.
Configurable Options	on | off
Default Setting	`off`
Configurable Arguments	The LDAP URL to the configuration directory. `nsslapd-pluginarg0: ldap://config.example.com/o=NetscapeRoot`
Dependencies	None
Further Information	Chapter 14, “Using the Pass-Through Authentication Plug-in” in the *Sun ONE Directory Server Administration Guide*. Note that the PTA plug-in is not listed in Directory Server console or in the `dse.ldif` file if you use the same server instance for your user directory and your configuration directory

Referential Integrity Postoperation Plug-In

Plug-In Name	Referential Integrity Postoperation
DN of Config Entry	`cn=Referential Integrity Postoperation,cn=plugins,cn=config`
Description	Enables the server to ensure referential integrity.
Configurable Options	All configuration and on | off
Default Setting	`off`
Configurable Arguments	When enabled, the post operation Referential Integrity plug-in performs integrity updates on the `member`, `uniquemember`, `owner` and `seeAlso` attributes immediately after a delete or rename operation. You can reconfigure the plug-in to perform integrity checks on all other attributes. The following arguments are configurable: 1. Check for referential integrity `-1` = no check for referential integrity `0` = check for referential integrity is performed immediately `positive integer` = request for referential integrity is queued and processed at a later stage. This positive integer serves as a wake-up call for the thread to process the request, at intervals corresponding to the integer specified. 2. Log file for storing the change, for example */ServerRoot/*`logs/referint` 3. All the additional attribute names you want to be checked for referential integrity.
Dependencies	Database
Limitations	Observe the following limitations when you use the referential integrity plug-in in a multi-master replication environment: • Enable the referential integrity plug-in on all servers containing master replicas • Enable the referential integrity plug-in with the same configuration on every master
Further Information	See “Maintaining Referential Integrity” in Chapter 2 of the *Sun ONE Directory Server Administration Guide*.

Retro Changelog Plug-In

Plug-In Name	Retro Changelog Plugin
DN of Config Entry	`cn=Retro Changelog Plugin,cn=plugins,cn=config`
Description	Used by LDAP clients for maintaining application compatibility with Directory Server 4.x versions. Maintains a log of all changes occurring in the Directory Server. The Retro Changelog offers the same functionality as the changelog in the 4.x versions of Directory Server.
Configurable Options	on | off
Default Setting	`off`
Configurable Arguments	See "Retro Changelog Plug-In Attributes," on page 263 for further information on the two configuration attributes for this plug-in.
Dependencies	None
Performance Related Information	May slow down Directory Server performance.
Further Information	Chapter 8, "Managing Replication" in the *Sun ONE Directory Server Administration Guide*.

Roles Plug-In

Plug-In Name	Roles Plugin
DN of Config Entry	`cn=Roles Plugin,cn=plugins,cn=config`
Description	Enables the use of roles in Directory Server.
Configurable Options	on | off
Default Setting	`on`
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.
Further Information	Chapter 5, "Advanced Entry Management" in the *Sun ONE Directory Server Administration Guide*.

State Change Plug-In

Plug-In Name	State Change Plugin
DN of Config Entry	`cn=State Change Plugin,cn=plugins,cn=config`
Description	State change notification service plug-in.
Configurable Options	on | off
Default Setting	`off`
Configurable Arguments	None
Dependencies	None

Subtree Entry Counter Plug-Ins

Plug-In Name	**Subtree Entry Counter For *ObjectClass***
DN of Config Entry	`cn=Subtree Entry Counter for` *ObjectClass*`,cn=plugins,cn=config`
Description	Maintain a count of entries with a particular object class. The following plug-ins are provided: - Subtree entry counter for departments in domains - Subtree entry counter for domains within a domain - Subtree entry counter for mail lists - Subtree entry counter for nested departments - Subtree entry counter for total domains - Subtree entry counter for users
Configurable Options	on | off
Default Setting	`off`
Configurable Arguments	None
Dependencies	None
Performance Related Information	These plug-ins are provided for use with Messaging Server only, and are disabled by default. It is recommended that you leave these plug-ins disabled unless your Messaging Server requires them.

Telephone Syntax Plug-In

Plug-In Name	Telephone Syntax
DN of Config Entry	`cn=Telephone Syntax,cn=plugins,cn=config`
Description	Syntax for handling telephone numbers.
Configurable Options	on | off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

UID Uniqueness Plug-In

Plug-In Name	UID Uniqueness
DN of Config Entry	`cn=UID Uniqueness,cn=plugins,cn=config`
Description	Checks that the values of specified attributes are unique each time a modification occurs on an entry.
Configurable Options	on | off
Default Setting	`off`

Configurable Arguments	Enter the following arguments:
	`uid`
	`"DN"`
	`"DN"...`
	to check for UID attribute uniqueness in all listed subtrees.
	However, enter the following arguments:
	`attribute="uid"`
	`MarkerObjectclass = "ObjectClassName"`
	and optionally
	`requiredObjectClass = "ObjectClassName"`
	to check for UID attribute uniqueness when adding or updating entries with the `requiredObjectClass`, starting from the parent entry containing the `ObjectClass` as defined by the `MarkerObjectClass` attribute.

Dependencies	N / A
Performance Related Information	Sun ONE Directory Server 5.2 provides the UID Uniqueness plug-in by default. To ensure unique values for other attributes, you can create instances of the UID Uniqueness plug-in for those attributes. The UID Uniqueness plug-in may slow down Directory Server performance.
Further Information	Chapter 15, “Using the UID Uniqueness Plug-in” in the *Sun ONE Directory Server Administration Guide*.

URI Plug-In

Plug-In Name	URI Syntax
DN of Config Entry	`cn=URI Syntax,cn=plugins,cn=config`
Description	Syntax for handling URIs (Unique Resource Identifiers) including URLs (Unique Resource Locators.)
Configurable Options	on | off
Default Setting	`on`
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. It is recommended that you leave this plug-in running at all times.

Attributes Common to All Plug-Ins

This list provides a brief attribute description, the Entry DN, valid range, default value, syntax, and an example for each attribute.

nsslapd-pluginPath

Specifies the full path to the plug-in.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	Any valid path
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-pluginPath: /`*ServerRoot*`/lib/uid-plugin.so`

nsslapd-pluginInitfunc

Specifies the plug-in function to be initiated.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	Any valid plug-in function.
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-pluginInitfunc: NS7bitAttr_Init`

nsslapd-pluginType

Specifies the plug-in type. See “nsslapd-plugin-depends-on-type” on page 216 for further information.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	Any valid plug-in type.
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-pluginType: preoperation`

nsslapd-pluginEnabled

Specifies whether or not the plug-in is enabled. This attribute can be changed over protocol, but will only take effect when the server is next restarted.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nsslapd-pluginEnabled: on`

nsslapd-pluginId

Specifies the plug-in ID.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	Any valid plug-in ID.
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-pluginId: chaining database`

nsslapd-pluginVersion

Specifies the plug-in version.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	Any valid plug-in version.
Default Value	Product version
Syntax	DirectoryString
Example	`nsslapd-pluginVersion:` *5.0b1*

nsslapd-pluginVendor

Specifies the vendor of the plug-in.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	Any approved plug-in vendor.
Default Value	Sun Microsystems, Inc.
Syntax	DirectoryString
Example	`nsslapd-pluginVendor: Sun Microsystems, Inc.`

nsslapd-pluginDescription

Provides a description of the plug-in.

Property	Value
Entry DN	`cn=`*plug-inName*`,cn=plugins,cn=config`
Valid Range	N / A
Default Value	None
Syntax	DirectoryString
Example	`nsslapd-pluginDescription: acl access check plug-in`

Attributes Allowed by Certain Plug-Ins

nsslapd-plugin-depends-on-type

Multi-valued attribute, used to ensure that plug-ins are called by the server in the correct order. Takes a value that corresponds to the `type` of a plug-in, contained in the attribute `nsslapd-pluginType` (see “nsslapd-pluginType” on page 214.) All plug-ins whose type value matches one of the values in the following valid range will be started by the server prior to this plug-in. The following example shows that the database plug-in will be started prior to the postoperation Referential Integrity plug-in.

Property	Value
Entry DN	`cn=referential integrity postoperation,cn=plugins,cn=config`
Valid Range	Database
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-plugin-depends-on-type: database`

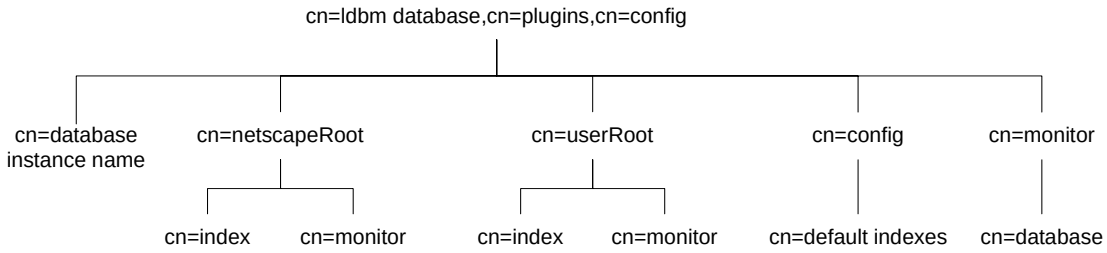
nsslapd-plugin-depends-on-named

Multi-valued attribute, used to ensure that plug-ins are called by the server in the correct order. Takes a value that corresponds to the `cn` value of a plug-in. The plug-in whose `cn` value matches one of the values below it will be started by the server prior to this plug-in. If the plug-in does not exist, the server will fail to start. The following example shows that the Class of Service plug-in will be started prior to the postoperation Referential Integrity plug-in. If the Class of Service plug-in does not exist, the server will fail to start.

Property	Value
Entry DN	`cn=referential integrity postoperation,cn=plugins,cn=config`
Valid Range	Class of Service
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-plugin-depends-on-named: Class of Service`

Database Plug-In Attributes

The database plug-in is also organized in an information tree as shown in the following diagram:



All plug-in technology used by the database instances is stored in the `cn=ldbm database` plug-in node. This section presents the additional attribute information for each of the nodes in bold in the `cn=ldbm database,cn=plugins,cn=config` information tree.

Database Configuration Attributes

Global configuration attributes common to all database instances are stored in the `cn=config,cn=ldbm database,cn=plugins,cn=config` tree node.

nsLookthroughLimit

This performance-related attribute specifies the maximum number of entries that Directory Server will check when examining candidate entries in response to a search request. If you bind as the directory manager DN, `unlimited` is set by default and overrides any other settings you may specify here.

Binder based resource limits work for this limit, which means that if a value for the operational attribute `nsLookThroughlimit` is present in the entry used to bind, the default limit is overridden. If you attempt to set a value that is not a number or is too big for a 64-bit signed integer, you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	-1 to the maximum number of entries (where -1 is unlimited)
Default Value	`5000`
Syntax	Integer
Example	`nsLookthroughLimit: 5000`

nsslapd-allidsthreshold

This performance-related attribute is present by default. It specifies the number of entry IDs that can be maintained for an index key, before the server sets the All IDs token and stops maintaining a list of IDs for that specific key. If you attempt to set a value that is not a number or is too big for a 64-bit signed integer, you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

However, as tuning this attribute is a complex task and can severely degrade performance, it is advisable to keep the default value. For a more detailed explanation of the All IDs Threshold see Chapter 10,"Managing Indexes" in the *Sun ONE Directory Server Administration Guide.*

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	100 to the maximum 64-bit integer value entry IDs
Default Value	`4000`
Syntax	Integer
Example	`nsslapd-allidsthreshold: 4000`

nsslapd-cache-autosize

This performance tuning related attribute is turned off by default. It specifies the percentage of free memory to use for all the combined caches. For example, if the value is set to 80, then 80 percent of the remaining free memory is claimed for the cache. If you plan to run other servers on the machine, then the value will be lower. Setting the value to 0 turns off the cache autosizing and uses the normal `nsslapd-cachememsize` and `nsslapd-dbcachesize` attributes.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	0 (turns cache autosizing off) to 100
Default Value	0
Syntax	Integer
Example	`nsslapd-cache-autosize: 80`

nsslapd-cache-autosize-split

This performance-related attribute specifies the percentage of cache space to allocate to the database cache. For example, setting this to “60” would give the database cache 60 percent of the cache space and divide the remaining 40 percent between the backend entry caches. That is, if there were 2 databases, each of them would receive 20 percent. This attribute applies only when the `nsslapd-cache-autosize` attribute has a value of `0`.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	
Default Value	66 (this will not necessarily optimize your operations.)
Syntax	Integer
Example	`nsslapd-cache-autosize-split: 66`

nsslapd-dbcachesize

This performance tuning related attribute specifies database cache size. Note that this is neither the index cache nor the entry cache. If you activate automatic cache resizing, you override this attribute, by replacing these values with its own guessed values at a later stage of the server startup.

If you attempt to set a value that is not a number or is too big for a 32-bit signed integer, you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	500KB to 4GB for 32-bit platforms and 500KB to 2^64-1 for 64-bit platforms
Default Value	`10 MB`
Syntax	Integer
Example	`nsslapd-dbcachesize: 10 MB`

NOTE On HP-UX only the maximum value for the `nsslapd-dbcachesize` attribute is 1GB, due to a PA-RISC hardware limitation that prevents memory-mapped files from crossing quadrant boundaries.

On Solaris platforms, the actual cache used may be significantly higher than what is specified in the `nsslapd-cachememsize` and and `nsslapd-dbcachesize` attributes. It is therefore recommended that you do not specify a total cache size of more than 1GB for 32-bit servers.

nsslapd-db-checkpoint-interval

The amount of time in seconds after which the Directory Server sends a checkpoint entry to the database transaction log. The database transaction log contains a sequential listing of all recent database operations and is used for database recovery only. A checkpoint entry indicates which database operations have been physically written to the directory database. The checkpoint entries are used to determine where in the database transaction log to begin recovery after a system failure. The `nsslapd-db-checkpoint-interval` attribute is absent from `dse.ldif`. To change the checkpoint interval, you add the attribute to `dse.ldif`. This attribute can be dynamically modified using `ldapmodify`. For further information on modifying this attribute, see the section on “Transaction Logging” in the *Sun ONE Directory Server Installation and Tuning Guide.*

This attribute is provided only for system modification / diagnostics and should be changed only with the guidance of Sun ONE engineering staff and Sun ONE Professional Services. Inconsistent settings of this attribute and other configuration attributes may cause the Directory Server to be unstable.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	10 to 300 seconds
Default Value	`60`
Syntax	Integer
Example	`nsslapd-db-checkpoint-interval: 120`

nsslapd-db-circular-logging

Specifies circular logging for the transaction log files. If this attribute is switched off, old transaction log files are not removed, and are kept renamed as old log transaction files. Turning circular logging off can severely degrade server performance. It should therefore only be modified with the guidance of Sun ONE engineering staff and Sun ONE Professional Services.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	on or off
Default Value	`on`
Syntax	DirectoryString
Example	`nsslapd-db-circular-logging: on`

nsslapd-db-durable-transactions

Indicates whether database transaction log entries are immediately written to the disk. The database transaction log contains a sequential listing of all recent database operations and is used for database recovery only.

With durable transactions enabled, every directory change is physically recorded in the log file and is therefore able to be recovered in the event of a system failure. However, the durable transactions feature may also slow down the performance of the Directory Server. With durable transactions disabled, all transactions are logically written to the database transaction log but may not be physically written to disk immediately. If there is a system failure before a directory change is physically written to disk, that change is not recoverable.

NOTE In previous versions of Directory Server, this attribute could not be modified dynamically. In Directory Server 5.2, this attribute can be modified dynamically using `ldapmodify`, without stopping the server.

For more information on database transaction logging, see Chapter 12, "Managing Log Files" in the *Sun ONE Directory Server Administration Guide.*

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nsslapd-db-durable-transactions: on`

nsslapd-db-home-directory

UNIX only. Used to fix a situation on UNIX platforms where the operating system endlessly flushes pages. This flushing can be so excessive that performance of the entire system is severely degraded.

This situation will occur only for certain combinations of the database cache size, the size of physical memory, and kernel tuning attributes. In particular, this situation should not occur if the database cache size is less than 100mb.

For example, if your Solaris host seems excessively slow and your database cache size is around 100mb or more, then you can use the `iostat` utility to diagnose the problem. Use `iostat` to monitor the activity of the disk where the Directory Server's database files are stored. If all of the following conditions are true:

- the disk is heavily used (more than 1mb per second of data transfer)
- there is a long service time (more than 100ms)
- there is mostly write activity

then you should use the `nsslapd-db-home-directory` attribute to specify a subdirectory of a tempfs type file system.

NOTE The directory referenced by the `nsslapd-db-home-directory` attribute must be a subdirectory of a file system of type tempfs (such as `/tmp`).

If you have multiple Directory Servers on the same machine, their `nsslapd-db-home-directory` attributes must be configured with different directories. Failure to do so will result in the databases for both directories becoming corrupted.

Finally, use of this attribute causes internal Directory Server database files to be moved to the directory referenced by the attribute. It is possible, but unlikely, that the server will no longer start after the files have been moved because not enough memory can be committed. This is a symptom of an overly large database cache size being configured for your server. If this happens, reduce the size of your database cache size to a value where the server will start again.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Any valid directory name in a tempfs file system, such as `/tmp`.
Default Value	N/A
Syntax	DirectoryString
Example	`nsslapd-db-home-directory: /tmp/slapd-phonebook`

nsslapd-db-idl-divisor

Specifies the index block size in terms of the number of blocks per database page. The block size is calculated by dividing the database page size by the value of this attribute. A value of 1 makes the block size exactly equal to the page size. The default value of 0 sets the block size to the page size minus an estimated allowance for internal database overhead. Before modifying the value of this attribute export all databases using the `db2ldif` script. Once the modification has been made, reload the databases using the `ldif2db` script.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	0 to 8
Default Value	`0`
Syntax	Integer
Example	`nsslapd-db-idl-divisor: 2`

nsslapd-db-locks

Specifies the number of locks that can be used by the database. Increase the value of this attribute if you observe the following error:

```
libdb: Lock table is out of available locks
```

The current number of locks being used, the number of locks configured, and the maximum number of locks reached during the life of the process can be checked using the attributes `nsslapd-db-current-locks`, `nsslapd-db-configured-locks`, and `nsslapd-db-max-locks` respectively, under the entry `cn=database,cn=monitor,cn=ldbm dabatase,cn=plugins,cn=config`.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	1 to maximum integer
Default Value	20000
Syntax	Integer
Example	`nsslapd-db-locks: 20000`

nsslapd-db-logbuf-size

Specifies the log information buffer size. Log information is stored in memory until the buffer fills up or the transaction commit forces the buffer to be written to disk. Larger buffer sizes can significantly increase throughput in the presence of long running transactions, highly concurrent applications, or transactions producing large amounts of data. The `nsslapd-db-logbuf-size` attribute is only valid if the `nsslapd-db-durable-transaction` attribute is set to on.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	32768 bytes to maximum integer (limited to available memory)
Default Value	32768
Syntax	Integer
Example	`nsslapd-db-logbuf-size: 32768`

nsslapd-db-logdirectory

The path to the directory containing the database transaction log. The database transaction log contains a sequential listing of all recent database operations and is used for database recovery only. By default, the database transaction log is stored in the same directory as the directory entries themselves:

/ServerRoot`/slapd-`*serverID*`/db`

For fault-tolerance and performance reasons, you may want to move this log file to another physical disk. The `nsslapd-db-logdirectory` attribute is absent from `dse.ldif`. To change the location of the database transaction log, add the attribute to `dse.ldif`. For more information on database transaction logging, see Chapter 12, “Managing Log Files” in the *Sun ONE Directory Server Administration Guide*.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Any valid path and directory name.
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-db-logdirectory: /logs/txnlog`

nsslapd-db-logfile-size

Specifies the maximum size of a single file in the log in bytes. By default, or if the value is set to 0, a maximum size of 10 MB is used. The maximum size is an unsigned 4-byte value. The value of this attribute can have significant impact on performance, as it can be tuned to avoid extensive log switching in the event of heavy entries.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	0 to unsigned 4-byte integer
Default Value	`10 (MB)`
Syntax	Integer
Example	`nsslapd-db-logfile-size: 10`

nsslapd-db-page-size

Specifies the size of the pages used to hold items in the database in bytes. The minimum size is 512 bytes and the maximum size is 64K bytes. If the page size is not explicitly set, Directory Server defaults to a page size of 8K bytes. Changing this default value can have significant performance impact. If the page size is too small, it results in extensive page splitting and copying, whereas if the page size is too large, it can waste disk space.

NOTE Before modifying the value of this attribute, export all databases using the `db2ldif` script. Once the modification has been made, reload the databases using the `ldif2db` script.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	512 bytes to 64 KB
Default Value	`8 (KB)`
Syntax	Integer
Example	`nsslapd-db-page-size: 8`

nsslapd-db-transaction-batch-val

Specifies how many transactions will be batched before being committed. You can use this attribute to improve update performance when full transaction durability is not required. This attribute can be dynamically modified using `ldapmodify`.

If you do not define this attribute or set it to a value of 0, transaction batching will be turned off and it will be impossible to make remote modifications to this attribute via LDAP. However, setting this attribute to a value greater than 0 causes the server to delay committing transactions until the number of queued transactions is equal to the attribute value. A value greater than 0 also allows you to modify this attribute remotely via LDAP. A value of 1 for this attribute allows you to modify the attribute setting remotely via LDAP, but results in no batching behavior. A value of 1 at server startup is therefore useful for maintaining normal durability, while also allowing transaction batching to be turned on and off remotely when desired. Bear in mind that the value you choose for this attribute may require you to modify the `nsslapd-db-logbuf-size` attribute to ensure sufficient log buffer size for accommodating your batched transactions.

NOTE The `nsslapd-db-transaction-batch-val` attribute is only valid if the `nsslapd-db-durable-transaction` attribute is set to on.

For more information on database transaction logging, see Chapter 12, “Managing Log Files” in the *Sun ONE Directory Server Administration Guide.*

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	0 to 30
Default Value	`0` (or turned off)
Syntax	Integer
Example	`nsslapd-db-transaction-batch-val: 5`

nsslapd-db-tx-max

Specifies the maximum number of concurrent transactions that can be handled by the database. Increase the value of this attribute if you observe the following error:

```
Serious Error---Failed in dblayer_txn_begin, err=12 (Not enough
space)
```

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	1 to maximum integer
Default Value	200
Syntax	Integer
Example	`nsslapd-db-tx-max: 200`

nsslapd-dbncache

This attribute allows you to split the `ldbm` cache into equally sized separate pieces of memory. It is possible to specify caches that are large enough so that they cannot be allocated contiguously on some architectures. For example, some releases of Solaris limit the amount of memory that may be allocated contiguously by a process. If `nsslapd-dbncache` is 0 or 1, the cache will be allocated contiguously in memory. If it is greater than 1, the cache will be broken up into `ncache` equally sized separate pieces of memory.

This attribute is provided only for system modification / diagnostics and should be changed only with the guidance of Sun ONE engineering staff and Sun ONE Professional Services. Inconsistent settings of this attribute and other configuration attributes may cause the Directory Server to be unstable.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Positive integer or 0
Default Value	`0`
Syntax	Integer
Example	`nsslapd-dbncache: 0`

nsslapd-import-cachesize

This performance tuning related attribute determines the size of the database cache used in the bulk import process. By setting this attribute value so that the maximum available system physical memory is used for the database cache during bulk importing, you can optimize bulk import speed. If you attempt to set a value that is not a number or is too big for a 32-bit signed integer, you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

NOTE A cache is created for each load that occurs. For example, if the user sets the `nsslapd-import-cachesize` attribute to 1GB, then 1GB is used when loading one database, 2GB is used when loading 2 databases, etc.

Ensure that you have sufficient physical memory to prevent swapping from occurring, as this results in performance degradation.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	500KB to 4GB for 32-bit platforms and 500KB to 2^64-1 for 64-bit platforms
Default Value	`20 (MB)`
Syntax	Integer
Example	`nsslapd-import-cachesize: 20`

nsslapd-mode

Specifies the permissions used for newly created index files.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Any four-digit octal number. However, mode 0600 is recommended. This allows read and write access for the owner of the index files (which is the user that `ns-slapd` runs as), and no access for other users.
Default Value	`0600`

Syntax	Integer
Example	`nsslapd-mode: 0600`

nsslapd-exclude-from-export

Specifies a list of attributes that will be excluded when the database is exported.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	N / A
Default Value	`entrydn entryid dncomp parentid numSubordinates`
Syntax	DirectoryString
Example	`nsslapd-exclude-from-export: entrydn entryid`

nsslapd-disk-low-threshold

Specifies the “low” free space on the disk (in MB). When the available free space on any one of the disks used by a database instance falls below the value specified by this attribute, protocol updates on that instance are permitted only by the directory manager.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	0 to unsigned 4-byte integer
Default Value	`100`
Syntax	Integer
Example	`nsslapd-disk-low-threshold: 100`

nsslapd-disk-full-threshold

When the minimum free space on the disk (in MB). When the available free space on any one of the disks used by a database instance falls below the value specified by this attribute, no updates are permitted and the server returns an LDAP_UNWILLING_TO_PERFORM error. Updates are allowed again as soon as free space rises above the threshold.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	0 to unsigned 4-byte integer
Default Value	`10`
Syntax	Integer
Example	`nsslapd-disk-full-threshold: 10`

Database Monitoring Attributes

Table 5-1 lists the global read-only attributes containing database statistics for monitoring activity on databases. These attributes are stored under `cn=monitor,cn=ldbm database,cn=plugins,cn=config`. For more information on these monitoring read-only entries see Chapter 12, “Managing Log Files” in the *Sun ONE Directory Server Administration Guide*.

Table 5-1 Database Monitoring Attributes

Attribute	Description
dbcachehits	Requested pages found in the database.
dbcachetries	Total requested pages found in the database cache.
dbcachehitratio	Percentage of requested pages found in the database cache (hits / tries).
dbcachepagein	Pages read into the database cache.
dbcachepageout	Pages written from the database cache to the backing file.
dbcacheroevict	Clean pages forced from the cache.
dbcacherwevict	Dirty pages forced from the cache.

Database Configuration Attributes Under cn=NetscapeRoot and cn=UserRoot

The cn=NetscapeRoot and cn=UserRoot subtrees contain configuration data for the databases containing the o=NetscapeRoot and o="suffixname" suffixes, respectively. The cn=NetscapeRoot subtree contains the configuration data used by the Sun ONE Administration Server for authentication and all actions that cannot be performed through LDAP (such as start/stop). The cn=UserRoot subtree contains all the configuration data for the user-defined database. The cn=UserRoot subtree is called UserRoot by default. However, this is not hard-coded, and, given the fact that there will be multiple database instances, this name will be changed and defined by the user when new databases are added.

The following attributes are common to both the cn=NetscapeRoot,cn=ldbm database,cn=plugins,cn=config and cn=UserRoot,cn=ldbm database,cn=plugins,cn=config subtrees.

nsslapd-cachesize

This performance tuning related attribute specifies the cache size in terms of the entries it can hold. However, it is worth noting that it is simpler to limit by memory size only (see nsslapd-cachememsize attribute). If you attempt to set a value that is not a number or is too big for a 32-bit signed integer, you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

Property	Value
Entry DN	cn=*suffixName*,cn=ldbm database,cn=plugins,cn=config
Valid Range	1 to 2,147,483,647 (or -1 which means limitless) entries
Default Value	-1
Syntax	Integer
Example	nsslapd-cachesize: -1

nsslapd-cachememsize

This performance tuning related attribute specifies the cache size in terms of available memory space. Limiting cachesize in terms of memory occupied is the simplest method. By activating automatic cache resizing, you override this attribute, replacing these values with its own guessed values at a later stage of the server startup. If you attempt to set a value that is not a number or is too big for a 32-bit signed integer, you will receive an LDAP_UNWILLING_TO_PERFORM error message with additional error information explaining the problem.

Property	Value
Entry DN	`cn=`*`suffixName`*`,cn=ldbm database,cn=plugins,cn=config`
Valid Range	200KB to 4GB
Default Value	`10 485 760` (10Mb)
Syntax	Integer
Example	`nsslapd-cachememsize:10`

nsslapd-directory

Specifies the absolute path to the database instance. If the database instance is created manually, this attribute must be included. It is set by default in the Sun ONE Server Console and can be modified. Once the database instance has been created, do not modify this path as any changes risk preventing the server from accessing data.

Property	Value
Entry DN	`cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Any valid absolute path to the database instance.
Default Value	N / A
Syntax	DirectoryString
Example	`nsslapd-directory: /`*`ServerRoot`*`/slapd-`*`serverID`*`/db`

nsslapd-readonly

Specifies read only permission. When this attribute is set to on, directory entries can be viewed but cannot be modified. This is useful, for example, when you are performing a backup of the directory.

Property	Value
Entry DN	`cn=`*`suffixName`*`,cn=ldbm database,cn=plugins,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-readonly: off`

nsslapd-require-index

When switched to on, this attribute allows you to refuse non-indexed or allids searches. This performance related attribute avoids saturating the server with erroneous searches.

Property	Value
Entry DN	`cn=`*`suffixName`*`,cn=ldbm database,cn=plugins,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsslapd-require-index: off`

nsslapd-suffix

Specifies the chained suffix. This is a single-valued attribute as each database instance can have only one suffix. Previously, it was possible to have more than one suffix on a single database instance but this is no longer the case. Any changes made to this attribute after the entry has been created take effect only after you restart the server containing the chained suffix.

Property	Value
Entry DN	`cn=`*`suffixName`*`,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Any valid DN
Default Value	*N/A*
Syntax	DirectoryString
Example	`nsslapd-suffix: o=Netscaperoot`

Database Performance Attributes

Table 5-2 lists the read-only database performance attributes. These attributes are stored under `cn=database,cn=monitor,cn=ldbm database, cn=plugins,cn=config`. All of the values for these attributes are 32-bit integers.

Table 5-2 Database Performance Attributes

Attribute	Description
nsslapd-db-abort-rate	Number of transactions that have been aborted.
nsslapd-db-active-txns	Number of transactions that are currently active (used by the database.)
nsslapd-db-cache-hit	Requested pages found in the cache.
nsslapd-db-cache-region-wait-rate	Number of times that a thread of control was forced to wait before obtaining the region lock.
nsslapd-db-cache-size-bytes	Total cache size in bytes.
nsslapd-db-cache-try	Total cache lookups.
nsslapd-db-clean-pages	Clean pages currently in the cache.
nsslapd-db-commit-rate	Number of transactions that have been committed.
nsslapd-db-configured-locks	Configured number of locks.
nsslapd-db-configured-txns	Configured number of transactions.
nsslapd-db-current-locks	Number of locks currently used by the database.
nsslapd-db-deadlock-rate	Number of deadlocks detected.
nsslapd-db-dirty-pages	Dirty pages currently in the cache.
nsslapd-db-hash-buckets	Number of hash buckets in buffer hash table.
nsslapd-db-hash-elements-examine-rate	Total number of hash elements traversed during hash table lookups.
nsslapd-db-hash-search-rate	Total number of buffer hash table lookups.
nsslapd-db-lock-conflicts	Total number of locks not immediately available due to conflicts.
nsslapd-db-lockers	Number of current lockers.
nsslapd-db-lock-region-wait-rate	Number of times that a thread of control was forced to wait before obtaining the region lock.
nsslapd-db-lock-request-rate	Total number of locks requested.
nsslapd-db-log-bytes-since-checkpoint	Number of bytes written to this log since the last checkpoint.

Table 5-2 Database Performance Attributes

Attribute	Description
nsslapd-db-log-flush-commit	The number of log flushes that contained a transaction commit record.
nsslapd-db-log-flush-count	The number of times the log has been flushed to disk.
nsslapd-db-log-max-commit-per-flush	The maximum number of commits contained in a single log flush.
nsslapd-db-log-min-commit-per-flush	The minimum number of commits contained in a single log flush that contained a commit.
nsslapd-db-log-region-wait-rate	Number of times that a thread of control was forced to wait before obtaining the region lock.
nsslapd-db-log-write-count	The number of times the log has been written to disk.
nsslapd-db-log-write-count-fill	The number of times the log has been written to disk because the in-memory log record cache filled up.
nsslapd-db-log-write-rate	Number of bytes written to the log since the last checkpoint.
nsslapd-db-longest-chain-length	Longest chain ever encountered in buffer hash table lookups.
nsslapd-db-max-locks	Maximum number of locks used by the database since the last startup.
nsslapd-db-max-txns	Maximum number of transactions used since the last startup.
nsslapd-db-page-create-rate	Pages created in the cache.
nsslapd-db-page-read-rate	Pages read into the cache.
nsslapd-db-page-ro-evict-rate	Clean pages forced from the cache.
nsslapd-db-page-rw-evict-rate	Dirty pages forced from the cache.
nsslapd-db-pages-in-use	All pages, clean or dirty, currently in use.
nsslapd-db-page-trickle-rate	Dirty pages written using the memp_trickle interface.
nsslapd-db-page-write-rate	Pages read into the cache.
nsslapd-db-txn-region-wait-rate	Number of times that a thread of control was force to wait before obtaining the region lock.

Default Index Attributes

The set of default indexes is stored under `cn=default indexes,cn=config,cn=ldbm database,cn=plugins,cn=config`. Default indexes are configured per backend in order to optimize Directory Server functionality for the majority of deployments.

All indexes, except system-essential ones, can be removed, but care should be taken not to cause unnecessary disruptions. This section presents four required indexing attributes and one optional indexing attribute. For further information on indexes see Chapter 10, "Managing Indexes" in the *Sun ONE Directory Server Administration Guide*.

nsSystemIndex

This mandatory attribute specifies whether the index is a system index, that is, an index that is vital for Directory Server operations. If this attribute has a value of `true`, it is system essential. System indexes must not be removed as this will seriously disrupt server functionality.

Property	Value
Entry DN	`cn=default indexes,cn=config,cn=ldbm database,cn=plugins,cn=config`
Valid Range	`true | false`
Default Value	N / A
Syntax	DirectoryString
Example	`nssystemindex: true`

nsIndexType

This optional, multi-valued attribute specifies the types of index used in Directory Server operations and the values of the attributes to be indexed. Each index type must be entered on a separate line.

Property	Value
Entry DN	`cn=default indexes,cn=config,cn=ldbm database,cn=plugins,cn=config`

Valid Range	pres = presence index eq = equality index approx = approximate index sub = substring index matching rule= international index index browse = browsing index
Default Value	N / A
Syntax	DirectoryString
Example	`nsindextype: eq`

nsMatchingRule

This optional, multi-valued attribute specifies the collation order object identifier (OID) required for the Directory Server to operate international indexing.

Property	Value
Entry DN	`cn=default indexes,cn=monitor,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Any valid collation order object identifier (OID)
Default Value	None
Syntax	DirectoryString
Example	`cn: 1.3.6.1.4.1.42.2.27.9.4.23.1` (For Bulgarian)

cn

Provides the name of the attribute to be indexed.

Property	Value
Entry DN	`cn=default indexes,cn=monitor,cn=ldbm database,cn=plugins,cn=config`
Valid Range	Any valid index cn.
Default Value	None
Syntax	DirectoryString
Example	`cn: aci`

description

This optional attribute provides a free-hand text description of what the index actually performs.

Property	Value
Entry DN	`cn=default indexes,cn=monitor,cn=ldbm database,cn=plugins,cn=config`
Valid Range	N/A
Default Value	None
Syntax	DirectoryString
Example	`description: substring index`

Database Monitoring Attributes Under cn=NetscapeRoot

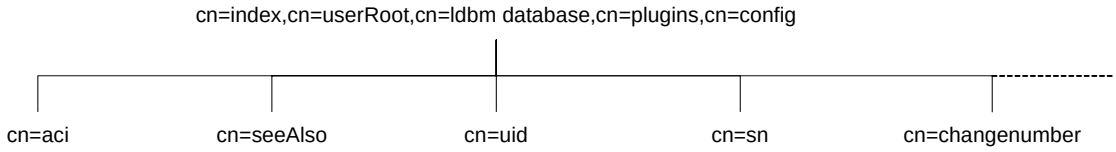
Table lists the global, read-only entries for monitoring activity on the NetscapeRoot database, stored under `cn=monitor,cn=Netscaperoot,cn=ldbm database, cn=plugins,cn=config`. These attributes contain database statistics and are provided for each file that makes up your database. For further information see Chapter 12, "Managing Log Files" in the *Sun ONE Directory Server Administration Guide*.

Table 5-3 Database Monitoring Attributes Under cn=NetscapeRoot

Attribute	Description
dbfilename-number	This attribute indicates the name of the file and provides a sequential integer identifier (starting at 0) for the file. All associated statistics for the file are given the same numerical identifier.
dbfilecachehit	Number of times that a search requiring data from this file was performed and data successfully obtained from the cache.
dbfilecachemiss	Number of times that a search requiring data from this file was performed and that the data could not be obtained from the cache.
dbfilepagein	Number of pages brought to the cache from this file.
dbfilepageout	Number of pages for this file written from cache to disk.

Database Index Attributes Under cn=NetscapeRoot and cn=UserRoot

In addition to the set of default indexes that are stored under `cn=default indexes,cn=config,cn=ldbm database,cn=plugins,cn=config`, custom indexes can be created for `o=Netscaperoot`, `o=UserRoot`, and manually created databases. These custom indexes are stored under the `cn=index,cn=NetscapeRoot,cn=ldbm database,cn=plugins,cn=config` and `cn=index,cn=UserRoot,cn=ldbm database,cn=plugins,cn=config` entries, respectively. Each indexed attribute represents a subentry under the above `cn=config` information tree nodes, as shown in the following figure:



For example, the index file for the `aci` attribute under `o=UserRoot` will appear in the Directory Server as follows:

```
dn:cn=aci,cn=index,cn=UserRoot,cn=ldbm
database,cn=plugins,cn=confi
objectclass:top
objectclass:nsIndex
cn=aci
nssystemindex:true
nsindextype:pres
```

Note that the `aci` attribute is an operational attribute and is not returned in a search unless you explicitly request it.

For details on the five possible indexing attributes, see the section “Default Index Attributes,” on page 238. For further information about indexes see Chapter 10, “Managing Indexes” in the *Sun ONE Directory Server Administration Guide*.

VLV Index Object Classes

A VLV (virtual list view) index provides fast searches against a known result set and sort ordering. To do this, the object class `vlvSearch` is needed to define the VLV search, and the object class `vlvIndex` is needed to order the search. VLV index object classes are stored under `cn=`*`MCCsuffixName`*`, cn=userRoot, cn=ldbm database,cn=plugins,cn=config`.

vlvIndex

Used to define the sort criteria of a Virtual List View index. Each VLV index specification defines the sort order to be imposed on the result set defined in the VLV search entry. A set of VLV index entries may appear below the VLV search entry. The `cn (commonName)` attribute is used as the naming component for the entry.

Property	Value
Entry DN	`cn=`*`MCCsuffixName`*`, cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Superior Class	top
OID	2.16.840.1.113730.3.2.42
Required Attributes	`cn, objectClass, vlvSort`
Allowed Attributes	`vlvEnabled, vlvUses`

vlvSearch

Used to define a VLV search. Specifies the entry result set to be VLV indexed.

Property	Value
Entry DN	`cn=`*`MCCsuffixName`*`, cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Superior Class	top
OID	2.16.840.1.113730.3.2.38
Required Attributes	`cn, objectClass, vlvBase, vlvFilter, vlvScope`
Allowed Attributes	`multiLineDescription`

VLV Index Attributes

VLV Index Attributes are stored under `cn=`*MCCsuffixName*`,cn=userRoot, cn=ldbm database,cn=plugins,cn=config`.

vlvBase

Defines the base DN of a VLV search.

Property	Value
Entry DN	`cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Valid Range	N / A
Default Value	N / A
Syntax	DN
Example	`vlvBase:o=example.com`

vlvEnabled

Used by the server to signal whether the index is available or unavailable. When VLV indexes are created offline, new `vlvSearch` entries are enabled when the indexes are rebuilt. VLV indexes can also be created while the server is running in read-only mode. This attribute is read-only and single-valued.

Property	Value
Entry DN	`cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Valid Range	0
Default Value	N / A
Syntax	Integer
Example	`vlvEnabled:0`

vlvFilter

Defines the filter for a VLV search.

Property	Value
Entry DN	`cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Valid Range	
Default Value	N / A
Syntax	IA5String
Example	`vlvFilter:(uid>=r)`

vlvScope

Defines the scope of a VLV search.

Property	Value
Entry DN	`cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Valid Range	0=base search 1=one level search 2=subtree search
Default Value	N / A
Syntax	Integer
Example	`vlvScope:1`

vlvSort

Defines the sort specification for a VLV search. Consists of a list of comma-delimited attribute names. A minus sign is used to denote a reverse sort. The example below will result in a sort by uid, then by reverse common name.

Property	Value
Entry DN	`cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Valid Range	N / A
Default Value	N / A
Syntax	DirectoryString
Example	`vlvSort:uid, -cn`

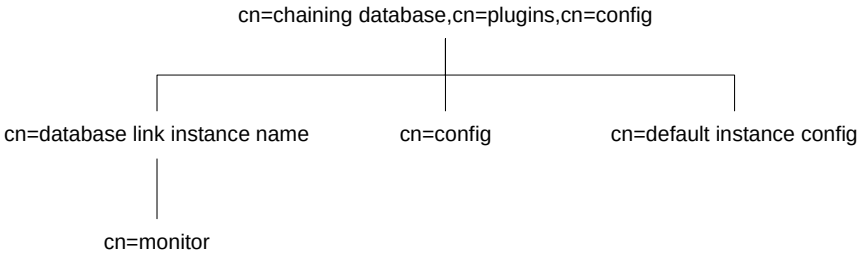
vlvUses

This read-only attribute displays the number of times the VLV index was used. This number resets after a restart of the server.

Property	Value
Entry DN	`cn=userRoot, cn=ldbm database, cn=plugins, cn=config`
Valid Range	1-x
Default Value	N / A
Syntax	Integer
Example	`vlvUses:7`

Chained Suffix Plug-In Attributes

The chained suffix plug-in is organized in an information tree as shown below:



All plug-in technology used by the chained suffix instances is stored in the `cn=chaining database` plug-in node. This section presents the additional attribute information for the three nodes marked in bold in the `cn=chaining database,cn=plugins,cn=config` information tree. For more information on the chaining backend, refer to "Creating Chained Suffixes" in Chapter 3 of the *Sun ONE Directory Server Administration Guide*.

Chained Suffix Attributes

Global chained suffix configuration attributes common to all instances are stored under `cn=config,cn=chaining database,cn=plugins,cn=config`.

nsActiveChainingComponents

Lists the components using chaining. A component is any functional unit in the server. The value of this attribute overrides the value in the global configuration attribute. To disable chaining on a particular database instance, use the value `None`.

This attribute also allows you to alter the components used to chain. By default, no components are allowed to chain. For this reason, this attribute does not appear in a list of `cn=config,cn=chaining database,cn=config` attributes, as LDAP considers empty attributes to be non-existent.

Property	Value
Entry DN	`cn=config,cn=chaining database,cn=plugins,cn=config`
Valid Range	Any valid component entry.
Default Value	None
Syntax	DirectoryString
Example	`nsActiveChainingComponents: cn=uid uniqueness,cn=plugins,cn=config`

nsMaxResponseDelay

This error detection, performance related attribute specifies the maximum period of time it can take a remote server to respond to an LDAP operation request made by a chained suffix before an error is suspected. Once this delay period has been met, the chained suffix tests the connection with the remote server.

Property	Value
Entry DN	`cn=config,cn=chaining database,cn=plugins,cn=config`
Valid Range	Any valid delay period in seconds.
Default Value	60 seconds
Syntax	Integer
Example	`nsMaxResponseDelay: 60`

nsMaxTestResponseDelay

This error detection, performance related attribute specifies the duration of the test issued by the chained suffix to check whether the remote server is responding. If a response from the remote server is not returned within this period, the chained suffix assumes the remote server is down and the connection is not used for subsequent operations.

Property	Value
Entry DN	`cn=config,cn=chaining database,cn=plugins,cn=config`
Valid Range	Any valid delay period in seconds.
Default Value	15 seconds
Syntax	Integer
Example	`nsMaxTestResponseDelay: 15`

nsTransmittedControls

This attribute, which can be both a global (and thus dynamic) configuration or an instance (`cn=chained suffix instance,cn=chaining database, cn=plugins,cn=config`) configuration attribute, allows you to alter the controls that the chained suffix forwards. The following controls are forwarded by default:

- Managed DSA, object identifier: 2.16.840.1.113730.3.4.2.
- Virtual list view (VLV), object identifier: 2.16.840.1.113730.3.4.9
- Server side sorting, object identifier: 1.2.840.113556.1.4.473

Property	Value
Entry DN	`cn=config,cn=chaining database,cn=plugins,cn=config`
Valid Range	Any valid OID or the above listed controls forwarded by the chained suffix.
Default Value	None
Syntax	Integer
Example	`nsTransmittedControls: 1.2.840.113556.1.4.473`

Default Instance Chained Suffix Attributes

Default instance chained suffix attributes are stored under `cn=default instance config,cn=chaining database,cn=plugins,cn=config`.

nsAbandonedSearchCheckInterval

The number of seconds that pass before the server checks for abandoned operations.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	0 to 2147483647 seconds
Default Value	`2`
Syntax	Integer
Example	`nsabandonedsearchcheckinterval: 10`

nsBindConnectionsLimit

Maximum number of TCP connections the chained suffix establishes with the remote server.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	1 to 50 connections
Default Value	`3`
Syntax	Integer
Example	`nsbindconnectionslimit: 3`

nsBindRetryLimit

Number of times a chained suffix attempts to bind with the remote server if the initial bind attempt is unsuccessful. A value of 0 here indicates that the chained suffix will only attempt to bind once only.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	0 to 5
Default Value	3
Syntax	Integer
Example	`nsbindretrylimit: 3`

nsBindTimeout

Period of time before the bind attempt times out. There is no real Valid Range for this attribute, except reasonable patience limits.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	0 to 60 seconds
Default Value	15
Syntax	Integer
Example	`nsbindtimeout:15`

nsCheckLocalACI

Reserved for advanced use only. Controls whether ACIs are evaluated on the chained suffix as well as the remote data server. Changes to this attribute only take effect once the server has been restarted.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`

Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nschecklocalaci: on`

nsConcurrentBindLimit

The maximum number of concurrent bind operations per TCP connection.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	1 to 25 binds
Default Value	`10`
Syntax	Integer
Example	`nsconcurrentbindlimit:10`

nsConcurrentOperationsLimit

The maximum number of concurrent operations allowed.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	1 to 50 operations
Default Value	`50`
Syntax	Integer
Example	`nsconcurrentoperationslimit: 50`

nsConnectionLife

Specifies the connection lifetime. You can keep connections between the chained suffix and the remote server open for an unspecified time, or you can close them after a specific period of time. Keeping the connections open is faster, but uses more resources. When the value is 0 and a list of failover servers is provided in the `nsFarmServerURL` attribute, the "main" server is never contacted after failover to the alternate server.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	0 to limitless seconds (where 0 means forever)
Default Value	`0`
Syntax	Integer
Example	`nsconnectionlife: 0`

nsOperationConnectionsLimit

Maximum number of LDAP connections the chained suffix establishes with the remote server.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	1 to 20 connections
Default Value	`10`
Syntax	Integer
Example	`nsoperationconnectionslimit:10`

nsProxiedAuthorization

Reserved for advanced use only, this attribute permits you to disable proxied authorization. A value of `off` means that proxied authorization is disabled.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	on | off
Default Value	`on`
Syntax	DirectoryString
Example	`nsproxiedauthorization: on`

nsReferralOnScopedSearch

Controls whether or not referrals are returned by scoped searches. This attribute allows you to optimize your directory, because returning referrals in response to scoped searches is more efficient.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	on | off
Default Value	`off`
Syntax	DirectoryString
Example	`nsreferralonscopedsearch: off`

nsslapd-sizelimit

Specifies the size limit of an entry for the chained suffix, in bytes.

Property	Value
Entry DN	`cn=default instance config,cn=chaining database, cn=plugins,cn=config`
Valid Range	-1 (no limit) to 2147483647 entries

Default Value	2000
Syntax	Integer
Example	nsslapd-sizelimit: 2000

nsslapd-timelimit

Specifies the default search time limit for the chained suffix.

Property	Value
Entry DN	cn=default instance config,cn=chaining database, cn=plugins,cn=config
Valid Range	-1 to 2147483647 seconds
Default Value	3600
Syntax	Integer
Example	nsslpad-timelimit: 3600

Instance-Specific Chained Suffix Attributes

Instance-specific chained suffix attributes are stored under cn=*chained suffix instance name*,cn=chaining database,cn=plugins,cn=config.

nsFarmServerURL

The LDAP URL of the remote server. A *farm server* is contains data in one or more databases. This attribute can contain optional servers for failover, separated by spaces. For cascading chaining, this URL can point to another chained suffix.

Property	Value
Entry DN	cn=*chained suffix instance name*,cn=chaining database,cn=plugins,cn=config
Valid Range	Any valid remote server LDAP URL.
Default Value	N / A
Syntax	DirectoryString
Example	nsFarmServerURL: ldap://epdiote.example.com:alternate_server:3333

nsMultiplexorBindDN

DN of the administrative entry used to communicate with the remote server. The *multiplexor* is the server that contains the chained suffix and communicates with the farm server. This bind DN cannot be the Directory Manager. If this attribute is not specified, the chained suffix binds as anonymous.

Property	Value
Entry DN	`cn=`*chained suffix instance name*`,cn=chaining database,cn=plugins,cn=config`
Valid Range	N / A
Default Value	DN of the multiplexor.
Syntax	DirectoryString
Example	`nsMultiplexorBindDN: cn=proxy manager`

nsMultiplexorCredentials

Password for the administrative user, in plain text. If no password is provided, users can bind as anonymous. The password is encrypted in the configuration file. Please note that the example below is what you *view*, *not* what you type.

Property	Value
Entry DN	`cn=`*chained suffix instance name*`,cn=chaining database,cn=plugins,cn=config`
Valid Range	Any valid password (that is encrypted using the DES reversible password encryption schema.)
Default Value	N / A
Syntax	DirectoryString
Example	`nsMultiplexorCredentials: {DES} 9Eko69APCJfF`

nshoplimit

Specifies the maximum number of times a suffix is allowed to chain, that is, the number of times a request can be forwarded from one chained suffix to another.

Property	Value
Entry DN	`cn=`*chained suffix instance name*`,cn=chaining database,cn=plugins,cn=config`
Valid Range	1 to an appropriate upper limit for your deployment.
Default Value	10
Syntax	Integer
Example	`nsHopLimit: 3`

Chained Suffix Monitoring Attributes

Table 5-4 lists the chained suffix attributes used for monitoring activity on instances. These attributes are stored under `cn=monitor,cn=`*database instance name*`,cn=chaining database,cn=plugins,cn=config`.

Table 5-4 Chained Suffix Monitoring Attributes

Attribute	Description
nsAddCount	Number of add operations received.
nsDeleteCount	Number of delete operations received.
nsModifyCount	Number of modify operations received.
nsRenameCount	Number of rename operations received.
nsSearchBaseCount	Number of base level searches received.
nsSearchOneLevelCount	Number of one-level searches received.
nsSearchSubtreeCount	Number of subtree searches received.
nsAbandonCount	Number of abandon operations received.
nsBindCount	Number of bind requests received.
nsUnbindCount	Number of unbinds received.
nsCompareCount	Number of compare operations received.
nsOperationConnectionCount	Number of open connections for normal operations.
nsBindConnectionCount	Number of open connections for bind operations.

Frontend Plug-In Attributes

The frontend plug-in enables you to access directory data by methods other than LDAP. Sun ONE Directory Server 5.2 provides a DSML frontend plug-in that enables access using DSMLv2 over HTTP / SOAP. Attributes for the DSML frontend plug-in are stored under `cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`.

ds-hdsml-clientauthmethod

Defines how the server will identify a client on a secure (SSL) connection.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	`clientCertOnly`: the server uses the credentials from the client certificate to identify the client. `httpBasicOnly`: the server uses the credentials from the HTTP authorization header to identify the client. `clientCertFirst`: the server attempts to use the client certificate credentials to identify the client. If there are no client certificate credentials, credentials from the HTTP authorization header are used.
Default Value	`clientCertFirst`
Syntax	DirectoryString
Example	`ds-hdsml-clientauthmethod: clientCertFirst`

ds-hdsml-dsmlschemalocation

The path to the DSMLv2 schema. This is generated automatically and should not be changed.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	Any valid path to the directory storing the DSML schema.
Default Value	*ServerRoot* / lib / DSMLv2.xsd
Syntax	DirectoryString
Example	`ds-hdsml-dsmlschemalocation: /var/ds5/slapd-myServer/lib/DSMLv2.xsd`

ds-hdsml-iobuffersize

The size of the buffer in which the DSML request is stored.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	1 to an appropriate upper limit for your deployment, with a maximum of 2147483647 (2^{31}-1). The value must be a multiple of 256.
Default Value	`8192`
Syntax	Integer
Example	`ds-hdsml-buffersize: 8192`

ds-hdsml-poolmaxsize

The maximum size of the pool of parsers.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	1 to an appropriate upper limit for your deployment, with a maximum of 2147483647 (2^{31}-1).
Default Value	`10`
Syntax	Integer
Example	`ds-hdsml-poolmaxsize: 10`

ds-hdsml-poolsize

The minimum (and default) size of the pool of parsers

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	1 to an appropriate upper limit for your deployment, with a maximum of 2147483647 (2^{31}-1).
Default Value	`5`
Syntax	Integer

Example	`ds-hdsml-poolsize: 5`

ds-hdsml-port

The HTTP port used for DSML communications. The selected port must be unique on the host system; make sure no other application is attempting to use the same port number. On UNIX systems, specifying a port number of less than 1024 requires the Directory Server to run as root.

Note that you must restart the server for a port number change to be taken into account.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	1-65535
Default Value	`80`
Syntax	Integer
Example	`ds-hdsml-port: 8080`

ds-hdsml-requestmaxsize

The maximum size of a DSML request. If the request is larger than this value, the server responds with the error message `REQUEST_ENTITY_TOO_LARGE` and closes the connection to prevent the client from continuing the request.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	1-2147483647 (2^{31}-1)
Default Value	`32768`
Syntax	Integer
Example	`ds-hdsml-requestmaxsize: 32768`

ds-hdsml-responsemsgsize

The maximum size of a server response to a DSML request (or a fraction of the maximum response size in the case of intermediate search responses).

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	1-2147483647 (2^{31}-1)
Default Value	`65536`
Syntax	Integer
Example	`ds-hdsml-responsemsgsize: 65536`

ds-hdsml-rooturl

The root URL that will be used in a DSML request.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	Any valid URL.
Default Value	`/dsml`
Syntax	DirectoryString
Example	`ds-hdsml-rooturl: /dsml`

ds-hdsml-secureport

The port number used for secure DSML communications (over SSL). The selected port must be unique on the host system; make sure no other application is attempting to use the same port number. On UNIX systems, specifying a port number of less than 1024 requires the Directory Server to run as root. Note that you must restart the server for a port number change to be taken into account.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	1-65535
Default Value	None
Syntax	Integer
Example	`ds-hdsml-secureport: 1443`

ds-hdsml-soapschemalocation

The path to the SOAP schema. This is generated automatically and should not be changed.

Property	Value
Entry DN	`cn=DSMLv2-SOAP-HTTP,cn=frontends,cn=plugins,cn=config`
Valid Range	Any valid path to the directory storing the SOAP schema.
Default Value	`/`*ServerRoot*`/lib/soap-env.xsd`
Syntax	DirectoryString
Example	`ds-hdsml-soapschemalocation: /var/ds5/slapd-myServer/lib/soap-eng.xsd`

Implementation of the DSMLv2 Standard

The complete DSMLv2 specification and supporting documentation can be found at:

http://www.oasis-open.org/committees/dsml/docs/DSMLv2.xsd and

http://www.oasis-open.org/committees/dsml/docs/DSMLv2.doc

The Sun ONE Directory Server implementation of this specification is complete, with the following restrictions:

- *Bindings*

  DSMLv2 defines two normative bindings: a SOAP request/response binding and a file binding that serves as the DSMLv2 analog of LDIF. Sun ONE Directory Server supports the SOAP request/response binding.

- *Modify DN*

  Sun ONE Directory Server supports the DSML `modDNRequest` and `modDNResponse` operations. Changing of a DN is supported; however, moving an entry to a different part of the directory tree is not supported.

- *Abandon Request*

  Sun ONE Directory Server does not support the `abandonRequest` operation, since this operation is of no use over HTTP.

- *Search Operations*

  Some DSML clients incorrectly send an equality match with value "*" when a presence match is intended. The directory server will return zero results from these misformatted queries. You can detect these incorrect clients by searching for the characters =\2a in the access log.

Content of the HTTP Header

Sun ONE Directory Server supports only the HTTP `POST` operation. The following example shows the minimum fields required to send a DSML request to the server over HTTP:

```
POST /dsml HTTP/1.1
content-length: 450
HOST: hostMachine
SOAPAction: ""
Content-Type: text/xml
Connection: close
```

The `Connection` field is optional. In HTTP 1.0, the default value of this field is `close`. In HTTP 1.1, however, the default value is `keep-alive`. It is therefore recommended that you include this field with a value of `close` in your last request if you are using HTTP 1.1, to accelerate the dialog.

Additional fields may be included in the HTTP header. If they are supported by Directory Server, their values will override the defaults. If the fields are not supported, the request will not be rejected by the server but the fields will be ignored.

Retro Changelog Plug-In Attributes

Two different types of changelogs are maintained by Sun ONE Directory Server 5.2. The first type, referred to as *changelog*, is used by multi-master replication and the second changelog, which is in fact a plug-in referred to as *retro changelog*, is intended for use by LDAP clients for maintaining application compatibility with Directory Server 4.x versions.

This Retro Changelog plug-in is used to record modifications made to a supplier server. When the supplier server's directory is modified, an entry is written to the Retro Changelog that contains:

- A number that uniquely identifies the modification. This number is sequential with respect to other entries in the change log.
- The modification action; that is, exactly how the directory was modified.

It is through the Retro Changelog plug-in that you access the changes performed to the Directory Server using searches to "`cn=changelog,cn=config`" file.

nsslapd-changelogdir

This attribute specifies the name of the directory in which the changelog database is created the first time the plug-in is run. By default the database is stored with all the other databases under:

ServerRoot`/slapd-`*serverID*`/db/changelog`

NOTE For performance reasons you will probably want to store this database on a different physical disk.

Property	Value
Entry DN	`cn=Retro Changelog Plugin,cn=plugins,cn=config`
Valid Range	Any valid path to the directory.
Default Value	None

Syntax	DirectoryString
Example	`nsslapd-changelogdir: /var/slapd-serverID/changelog`

nsslapd-changelogmaxage (Max Changelog Age)

Specifies the maximum age of any entry in the change log. The change log contains a record for each directory modification and is used when synchronizing consumer servers. Each record contains a timestamp. Any record with a timestamp that is older than the value specified in this attribute will be removed. If this attribute is absent, there is no age limit on change log records, which is the default behavior as this attribute is not present by default.

Property	Value
Entry DN	`cn=Retro Changelog Plugin,cn=plugins,cn=config`
Valid Range	0 (meaning that entries are not removed according to their age) to the maximum 32 bit integer value (2147483647).
Default Value	0
Syntax	DirectoryString *IntegerAgeID* where AgeID is “s” for seconds, “m” for minutes, “h” for hours, “d” for days, or “w” for weeks.
Example	`nsslapd-changelogmaxage: 30d`

nsslapd-changelogmaxentries (Max Changelog Entries)

Specifies the maximum number of entries in the change log. The change log contains a record for each directory modification and is used when synchronizing consumer servers.

Property	Value
Entry DN	`cn=Retro Changelog Plugin,cn=plugins,cn=config`
Valid Range	0 (no limit to the number of entries) to the maximum 32 bit integer value (2147483647).
Default Value	0
Syntax	Integer
Example	`nsslapd-changelogmaxentries: 0`

Subtree Entry Counter Plug-In Attributes

The subtree entry counter plug-ins maintain a count of entries with a particular object class. The counter attributes are listed in Table 5-5.

Table 5-5 Subtree Entry Counter Plug-In Attributes

Attribute	Definition
nsNumDepts	Either the number of departments within a domain, or the number of departments within a department (nested departments), depending on the dn of the entry.
nsNumDomains	Either the number of total domains, or the number of domains within a domain (nested domains), depending on the dn of the entry.
nsNumMailLists	Number of mail lists.

Chapter 6

Migration From Earlier Versions

This chapter is intended to provide a reference of the information migrated by the `migrateInstance5` script. It describes which attributes are migrated automatically by the migration script, and which ones must be set manually.

In the case of migration from a 4.x Directory Server to a 5.2 Directory Server, it also describes the mapping of configuration parameters to configuration attributes and configuration entries in the new Directory Server.

For information on how to run the `migrateInstance5` script, refer to the *Sun ONE Directory Server Installation and Tuning Guide.*

Migrating From Directory Server 4.x to 5.2

In the Directory Server 4.x architecture, all configuration parameters were stored in text files. In Sun ONE Directory Server 5.x, all configuration attributes are stored in LDAP configuration entries in the `dse.ldif` file.

This section describes the mapping of configuration parameters in Directory Server 4.x to the corresponding LDAP configuration entries and attributes in Sun ONE Directory Server 5.2.

Server Attributes

In Directory Server 4.x, configuration parameters are stored in the `slapd.conf` file under the `/usr/netscape/server4/slapd-`*serverID* directory.

The corresponding configuration attributes in Sun ONE Directory Server 5.2 are stored in the `cn=config` entry. Table 6-1 shows the mapping of Directory Server 4.x configuration parameters to Directory Server 5.2 configuration attributes.

Table 6-1 Mapping of Legacy Server Parameters to Configuration Attributes

Legacy Configuration Parameter	Sun ONE Directory Server Configuration Attribute
accesscontrol	nsslapd-accesscontrol
error-logging-enabled	nsslapd-error-logging-enabled
audit-logging-enabled	nsslapd-audit-logging-enabled
logbuffering	nsslapd-accesslog-buffering
accesslog-logexpirationtime	nsslapd-accesslog-logexpirationtime
accesslog-logexpirationtimeunit	nsslapd-accesslog-logexpirationtimeunit
accesslog-maxlogdiskspace	nsslapd-accesslog-logmaxdiskspace
accesslog-minfreediskspace	nsslapd-accesslog-minfreediskspace
accesslog-logrotationtime	nsslapd-accesslog-logrotationtime
accesslog-logrotationtimeunit	nsslapd-accesslog-logrotationtimeunit
accesslog-maxlogsize	nsslapd-accesslog-maxlogsize
accesslog-MaxNumOfLogsPerDir	nsslapd-accesslog-maxlogsperdir
auditlog-logexpirationtime	nsslapd-auditlog-logexpirationtime
auditlog-logexpirationtimeunit	nsslapd-auditlog-logexpirationtimeunit
auditlog-maxlogdiskspace	nsslapd-auditlog-logmaxdiskspace
auditlog-minfreediskspace	nsslapd-auditlog-minfreediskspace
auditlog-logrotationtime	nsslapd-auditlog-logrotationtime
auditlog-logrotationtimeunit	nsslapd-auditlog-logrotationtimeunit
auditlog-maxlogsize	nsslapd-auditlog-maxlogsize
auditlog-MaxNumOfLogsPerDir	nsslapd-auditlog-maxlogsperdir
certmap-basedn	nsslapd-certmap-basedn
enquote_sup_oc	nsslapd-enquote_sup_oc
loglevel	nsslapd-error-loglevel
errorlog-logexpirationtime	nsslapd-errorlog-logexpirationtime
errorlog-logexpirationtimeunit	nsslapd-errorlog-logexpirationtimeunit
errorlog-maxlogdiskspace	nsslapd-errorlog-logmaxdiskspace
errorlog-minfreediskspace	nsslapd-errorlog-logminfreediskspace
errorlog-logrotationtime	nsslapd-errorlog-logrotationtime

Table 6-1 Mapping of Legacy Server Parameters to Configuration Attributes *(Continued)*

Legacy Configuration Parameter	Sun ONE Directory Server Configuration Attribute
errorlog-logrotationtimeunit	nsslapd-errorlog-logrotationtimeunit
errorlog-maxlogsize	nsslapd-errorlog-maxlogsize
errorlog-maxlogsperdir	nsslapd-errorlog-maxlogsperdir
idletimeout	nsslapd-idletimeout
ioblocktimeout	nsslapd-ioblocktimeout
lastmod	nsslapd-lastmod
listenhost	nsslapd-listenhost
maxdescriptors	nsslapd-maxdescriptors
(No equivalent)	nsslapd-depends-on-named
(No equivalent)	nsslapd-depends-on-type
referral	nsslapd-referral
reservedescriptors	nsslapd-reservedescriptors
rootpwstoragescheme	nsslapd-rootpwstoragescheme
schemacheck	nsslapd-schemacheck
secure-port	nsslapd-securePort
security	nsslapd-security
sizelimit	nsslapd-sizelimit
SSL3ciphers	nsslapd-SSL3ciphers
timelimit	nsslapd-timelimit
pw_change	passwordChange
pw_syntax	passwordCheckSyntax
pw_exp	passwordExp
pw_history	passwordHistory
pw_inhistory	passwordinHistory
pw_lockout	passwordLockout
pw_lockduration	passwordLockoutDuration
pw_maxage	passwordMaxAge
pw_maxfailure	passwordMaxFailure
pw_minage	passwordMinAge

Table 6-1 Mapping of Legacy Server Parameters to Configuration Attributes *(Continued)*

Legacy Configuration Parameter	Sun ONE Directory Server Configuration Attribute
pw_minlength	passwordMinLength
pw_must_change	passwordMustChange
pw_reset_failurecount	passwordResetFailureCount
pw_storagescheme	passwordStorageScheme
pw_unlock	passwordUnlock
pw_warning	passwordWarning
localhost	nsslapd-localhost
localuser	nsslapd-localuser
port	nsslapd-port
rootdn	nsslapd-rootdn
rootpw	nsslapd-rootpw
accesslog	nsslapd-accesslog
accesslog-level	nsslapd-accesslog-level
auditfile	nsslapd-auditlog
errorlog	nsslapd-errorlog
instancedir	nsslapd-instancedir
maxbersize	nsslapd-maxbersize
nagle	nsslapd-nagle
result_tweak	nsslapd-result_tweak
return_exact_case	nsslapd-return_exact_case
threadnumber	nsslapd-threadnumber
maxthreadsperconn	nsslapd-maxthreadsperconn

Database Attributes

In Directory Server 4.x, database parameters are stored in the `slapd.ldbm.conf` file under the `/usr/netscape/server4/slapd-serverID` directory.

Because one instance of Sun ONE Directory Server 5.x can manage several databases, the corresponding attributes in Sun ONE Directory Server 5.x are stored in a general entry for all databases (`cn=config,cn=ldbm database,cn=plugins,cn=config`), or in an entry specific to a particular database, of the form

`cn=`*database instance name*`,cn=ldbm database,cn=config`

Table 6-2 shows the mapping of general database configuration parameters between Directory Server 4.x and Directory Server 5.2.

Table 6-2 Mapping of General Legacy Database Parameters to Configuration Attributes

Legacy Configuration Parameter	Sun ONE Directory Server Configuration Attribute
`allidsthreshold`	`nsslapd-allidsthreshold`
`lookthroughlimit`	`nsslapd-lookthroughlimit`
`mode`	`nsslapd-mode`
`database`	OBSOLETE (used to specify database type)

Table 6-3 shows the mapping of database-specific parameters between Directory Server 4.x and Directory Server 5.2.

Table 6-3 Mapping of Database-Specific Legacy Parameters to Configuration Attributes

Legacy Configuration Parameter	Sun ONE Directory Server Configuration Attribute
`cachesize`	`nsslapd-cachesize`
`readonly`	`nsslapd-readonly`
`directory`	`nsslapd-directory`

Not all parameters are migrated by the `migrateInstance5` script. Table 6-4 indicates the Directory Server 4.x parameters that are not migrated automatically, and why automatic migration is not done in each case.

Table 6-4 Legacy Parameters Not Migrated by the Migration Script

Legacy Configuration Parameter	Directory Server 5.2 Configuration Attribute	Reason
localhost	nsslapd-localhost	Already configured.
port	nsslapd-port	Configured manually during installation.
rootdn	nsslapd-rootdn	Configured manually during installation.
rootpw	nsslapd-rootpw	Configured manually during installation.
accesslog	nsslapd-accesslog	Set up automatically. Pathname of the database access log.
accessloglevel	nsslapd-accesslog-level	Reserved for future use. Do not use, change or remove. Doing so may have unpredictable results.
auditfile	nsslapd-auditlog	Set up automatically. Pathname of the log used to record changes made to the database.
errorlog	nsslapd-errorlog	Set up automatically. Pathname of the log used to record error msgs generated by Directory Server.
instancedir	nsslapd-instancedir	Set up during installation.
result_tweak	nsslapd-result_tweak	Reserved for future use. Do not use, change or remove. Doing so may have unpredictable results.
directory	nsslapd-directory	Set up during installation.
database	(No equivalent)	OBSOLETE (used to specify database type)

Table 6-5 indicates the parameters that are migrated but are potentially problematic. You are advised to check their values in the new installation:

Table 6-5 Legacy Parameters Migrated by the Migration Script

Legacy Configuration Parameter	Directory Server 5.2 Configuration Attribute
`maxbersize`	`nsslapd-maxbersize`
`maxthreadsperconn`	`nsslapd-maxthreadsperconn`
`nagle`	`nsslapd-nagle`
`return_exact_case`	`nsslapd-return_exact_case`
`threadnumber`	`nsslapd-threadnumber`

Upgrading From Directory Server 5.0 or 5.1 to 5.2

In Directory Server 5.0, 5.1, and 5.2, configuration information is stored in the same way. This section explains which configuration attributes are automatically migrated by the `migrateInstance5` script, and which ones are not. Attributes which are not automatically migrated are either configured during the installation process for the new Directory Server, or need to be configured manually for security reasons after the initial setup.

General Server Configuration Attributes

The following list provides the configuration attributes stored in the `cn=config` entry that are automatically migrated when you run the `migrateInstance5` script:

- nsslapd-accesscontrol
- nsslapd-errorlog-logging-enabled
- nsslapd-accesslog-logging-enabled
- nsslapd-auditlog-logging-enabled
- nsslapd-accesslog-level
- nsslapd-accesslog-logbuffering
- nsslapd-accesslog-logexpirationtime
- nsslapd-accesslog-logexpirationtimeunit

- nsslapd-accesslog-logmaxdiskspace
- nsslapd-accesslog-logminfreediskspace
- nsslapd-accesslog-logrotationtime
- nsslapd-accesslog-logrotationtimeunit
- nsslapd-accesslog-maxlogsize
- nsslapd-accesslog-maxlogsperdir
- nsslapd-attribute_name_exceptions
- nsslapd-auditlog-logexpirationtime
- nsslapd-auditlog-logexpirationtimeunit
- nsslapd-auditlog-logmaxdiskspace
- nsslapd-auditlog-logminfreediskspace
- nsslapd-auditlog-logrotationtime
- nsslapd-auditlog-logrotationtimeunit
- nsslapd-auditlog-maxlogsize
- nsslapd-auditlog-maxlogsperdir
- nsslapd-certmap-basedn
- nsslapd-ds4-compatible-schema
- nsslapd-enquote_sup_oc
- nsslapd-errorlog-level
- nsslapd-errorlog-logexpirationtime
- nsslapd-errorlog-logexpirationtimeunit
- nsslapd-errorlog-logmaxdiskspace
- nsslapd-errorlog-logminfreediskspace
- nsslapd-errorlog-logrotationtime
- nsslapd-errorlog-logrotationtimeunit
- nsslapd-errorlog-maxlogsize
- nsslapd-errorlog-maxlogsperdir

- nsslapd-groupevalnestlevel
- nsslapd-idletimeout
- nsslapd-ioblocktimeout
- nsslapd-lastmod
- nsslapd-listenhost
- nsslapd-maxdescriptors (Not applicable on NT and AIX platforms)
- nsslapd-nagle
- nsslapd-readonly
- nsslapd-referralmode
- nsslapd-plugin-depends-on-name
- nsslapd-plugin-depends-on-type
- nsslapd-referral
- nsslapd-reservedescriptors (Not applicable on NT and AIX platforms)
- nsslapd-rootpwstoragescheme
- nsslapd-schemacheck
- nsslapd-securePort
- nsslapd-security
- nsslapd-sizelimit
- nsslapd-SSL3ciphers
- nsslapd-timelimit

NOTE The attribute `nsslapd-errorlog-level` has been deprecated in Sun ONE Directory Server 5.2. It is still supported for backward compatibility but has been replaced by the nsslapd-infolog-area (Information Log Area) and nsslapd-infolog-level (Information Log Level) attributes.

Table 6-6 lists the configuration attributes stored in the `cn=config` entry that are *not* automatically migrated when you run the `migrateInstance5` script. Attributes that are not automatically migrated are either configured during the installation process for the new Directory Server, or need to be configured manually. The reason for not migrating an attribute is stated in the table.

Table 6-6 Attributes in cn=config Not Migrated

Attribute Name	Reason for not Migrating Automatically
nsslapd-localhost	Already set up.
nsslapd-localuser	Configured during the installation process.
nsslapd-port	Configured during the installation process.
nsslapd-rootdn	Configured during the installation process.
nsslapd-rootpw	Configured during the installation process.
nsslapd-accesslog	Path name to the log that records database access. It is set up during installation.
nsslapd-accesslog-list	Read-only attribute.
nsslapd-auditlog	Path name to the log that records changes made to the directory database. It is set up during installation.
nsslapd-accesslog-level	Read-only attribute.
nsslapd-errorlog	Path name to the log that records error messages generated by Directory Server. It is set up during installation.
nsslapd-errorlog-list	Read-only attribute.
nsslapd-instancedir	Configured during the installation process.
nsslapd-maxbersize	Do not change the value of this attribute unless told to do so by Sun ONE technical staff.
nsslapd-plug-in	
nsslapd-result-tweak	Reserved for future use. Do not change or remove.
nsslapd-return-exact-case	Do not modify unless you have legacy client applications that can check the case of attribute names in results returned from the server.
nsslapd-threadnumber	This attribute is not available from the Directory Server Console.
nsslapd-maxthreadsperconn	This attribute corresponds to a system parameter.

Password Policy Attributes

The attributes that determine the password policy are stored in the entry `cn=Password Policy,cn=config`. Note that the location of these attributes has changed. In previous versions of Directory Server, they were located directly under `cn=config`. The following list provides the password policy attributes that are automatically migrated when you run the `migrateInstance5` script:

- passwordChange
- passwordCheckSyntax
- passwordExp
- passwordExpireWithoutWarning
- passwordInHistory
- passwordLockout
- passwordLockoutDuration
- passwordMaxAge
- passwordMaxFailure
- passwordMinAge
- passwordMinLength
- passwordMustChange
- passwordResetFailureCount
- passwordStorageScheme
- passwordUnlock
- passwordWarning

Database Attributes

All general database configuration attributes are automatically migrated. These attributes are stored in the entry `cn=config,cn=ldbm database, cn=plugins,cn=config`, and are as follows:

- nsslapd-allidthreshold
- nsslapd-lookthroughlimit

- nsslapd-mode
- nsslapd-dbcachesize
- nsslapd-cache-autosize
- nsslapd-cache-autosize-split
- nsslapd-db-transaction-logging

Database-specific attributes are stored in entries of the form `cn=`*database instance name*`,cn=ldbm database,cn=config`. The following list provides the attributes that are migrated:

- nsslapd-cachesize
- nsslapd-cachememsize
- nsslapd-readonly
- nsslapd-require-index

Table 6-7 lists the attributes that are *not* migrated automatically and indicates why this is the case:

Table 6-7 Database-Specific Attributes Not Migrated

Attribute Name	Reason For Not Migrating Automatically
nsslapd-directory	Set up automatically during installation.
nsslapd-db-logdirectory	Set up automatically during installation.
nsslapd-db-checkpoint-interval	This attribute is provided only for system modification / diagnostics and should be changed only under guidance from Sun ONE technical staff. Inconsistent settings of this attribute might cause Directory Server crashes.
nsslapd-db-durable-transactions	This attribute is provided only for system modification / diagnostics and should be changed only under guidance from Sun ONE technical staff. Inconsistent settings of this attribute might cause Directory Server crashes.
nsslapd-db-home-directory	If you have several directory servers running on the same machine, the value of this attribute must be different for each instance of the directory server. Therefore, it needs to be configured manually.

Chained Suffix Attributes

All chained suffix configuration attributes are migrated automatically. The following configuration attributes are common to all chained suffixes. These attributes are stored in the entry `cn=config,cn=chaining database, cn=plugins,cn=config`.

- nsActivechainingComponents
- nsTransmittedControls

The following configuration attributes apply to a default instance of a chained suffix. These attributes are stored in the entry `cn=default instance config, cn=chaining database,cn=plugins,cn=config`.

- nsAbandonedSearchCheckInterval
- nsBindConnectionsLimit
- nsBindTimeout
- nsBindRetryLimit
- nsHopLimit
- nsmaxresponsedelay
- nsmaxtestresponsedelay
- nsCheckLocalACI
- nsConcurrentBindLimit
- nsConcurrentOperationsLimit
- nsConnectionLife
- nsOperationConnectionslimit
- nsProxiedAuthorization
- nsReferralOnScopedSearch
- nsslapd-sizelimit
- nsslapd-timelimit

SNMP Attributes

All SNMP configuration attributes are automatically migrated. These attributes are stored in the entry `cn=SNMP,cn=config`, and are as follows:

- nssnmpenabled
- nssnmporganization
- nssnmplocation
- nssnmpcontact
- nssnmpdescription
- nssnmpmasterhost
- nssnmpmasterport

Part 3

File Reference

This part provides reference information on the files that are installed with Directory Server and on the log files. It includes the following chapters:

- Server Instance Files
- Access Logs and Connection Codes

Chapter 7

Server Instance Files

This chapter provides an overview of the files stored under *ServerRoot*/slapd-*serverID*. Having an overview of the files and configuration information stored in each instance of Directory Server will help you understand the file changes or absence of file changes that occur in the course of directory activity. It will also help you to detect errors and intrusion, by indicating what kind of changes to expect, and as a result, what changes are considered abnormal.

Overview of Directory Server Files

Directory Server files and command-line scripts are stored under *ServerRoot*/slapd-*serverID*, where *serverID* is the server identifier. The only exception is the migrateInstance5 script, which is stored under *ServerRoot*/bin/slapd/admin/bin.

A summary of the files installed in a typical directory installation is provided in Appendix A of the *Sun ONE Directory Server Installation and Tuning Guide*.

To reflect the directory structure under *ServerRoot*/slapd-*serverID*, this chapter is divided into the following sections:

- Backup Files
- Configuration Files
- Database Files
- ldif Files
- Lock Files
- Log Files

Each section describes the file type and contents.

Backup Files

Each Directory Server instance contains the following three directories for storing backup related files:

- `bak` - the default directory in which database backups (created with the `db2bak` script) are placed. The `bak` directory contains one directory for each database backup, the name of which corresponds to the time and date of the backup, for example `2002_12_13_174524`. This directory holds the backup copy of the database. Note that you can specify an alternative location for the database backups if you do not want them to be stored in the default `bak` directory. See "db2bak (Create Backup of Database)," on page 49 for more information.
- `confbak` - the default directory in which the Admin Server configuration is stored, (and from which the configuration is read) when the `saveconfig` and `restoreconfig` scripts are used. See "saveconfig (Save Administration Server Configuration)" and "restoreconfig (Restore Administration Server Configuration)," on page 55 for more information.
- `conf_bk` - contains a backup copy of the `dse.ldif` configuration file from the time of installation. This copy can be used for comparison with the current configuration file, should problems arise.

Configuration Files

Each Directory Server instance contains the following directory for storing configuration files:

- `config` - contains the configuration files as explained in "Server Configuration Overview" on page 75.

The `dse.ldif` file is a configuration file for each directory instance, whereas the admin server configuration (everything under `o=NetscapeRoot`) is only in the configuration directory. The configuration directory is usually the first directory that was installed, or may be a completely separate instance.

For small deployments, it is possible to install configuration, user and other directories on the same directory instance. For larger deployments, consider placing the configuration directory in its own instance. Refer to the *Sun ONE Server Console Server Management Guide* for information on the appropriate location of configuration, user and group data.

Database Files

Each Directory Server instance contains the db directory for storing all the database files. The following list shows the sample contents of the db directory at installation.

```
DBVERSION       __db.002        __db.005
NetscapeRoot/   __db.003        log.0000017
__db.001        __db.004        userRoot/
```

- db.00x files - used internally by the database. These files should not be moved, deleted, or modified in any way.
- log.xxxxxxxxxx files - store the transaction logs per database.
- DBVERSION - stores the version of the database.
- NetscapeRoot - this directory stores the o=NetscapeRoot database created by default during a typical installation. This branch of the directory stores admin server configuration information. The same configuration directory can be used to store the admin server configuration information for all directory instances. Refer to the *Sun ONE Server Console Server Management Guide* for information on the appropriate location of configuration, user and group data.
- userRoot - this directory stores the user-defined suffix (user-defined databases) created during a typical installation, for example dc=example,dc=com.

The following list shows the sample contents of the NetscapeRoot directory:

```
DBVERSION                       NetscapeRoot_nsUniqueId.db3
NetscapeRoot_aci.db3            NetscapeRoot_numsubordinates.db3
NetscapeRoot_ancestorid.db3     NetscapeRoot_objectclass.db3
NetscapeRoot_cn.db3             NetscapeRoot_parentid.db3
NetscapeRoot_entrydn.db3        NetscapeRoot_sn.db3
NetscapeRoot_givenName.db3      NetscapeRoot_uid.db3
NetscapeRoot_id2entry.db3       NetscapeRoot_uniquemember.db3
```

NOTE To ensure that database filenames are unique across suffixes, the files are prefixed with the suffix name. So, for the NetscapeRoot suffix in the above example, all the filenames in the directory start with NetscapeRoot_.

The NetscapeRoot and userRoot subdirectories contain a file of the format *suffix_index_name*.db3 for every index currently defined in the database (where *index_name* is the name of the attribute being indexed). In addition to these *suffix_index_name*.db3 files, the subdirectories contain a file named *suffix*_id2entry.db3. This file contains the actual directory database entries. All other database files can be recreated from this one, if necessary.

ldif Files

Each Directory Server instance contains the ldif directory for storing ldif related files. The following list shows the default contents of the ldif directory.

```
European.ldif
Example.ldif
Example-roles.ldif
Example-Plugin.ldif
identityMapping_Examples.ldif
```

The following list describes the contents of each of the ldif files:

- European.ldif - contains European character samples.
- Example.ldif - a sample ldif file.
- Example-roles.ldif - a sample ldif file similar to Example.ldif except that it uses roles and class of service instead of groups for setting access control and resource limits for Directory Administrators
- Example-Plugin.ldif - a sample ldif file to be used with the examples provided in the *Sun ONE Directory Server Plug-In API Programming Guide.*
- identityMapping_Examples.ldif - a sample identity mapping configuration file. For more information on identity mapping, refer to the *Sun ONE Directory Server Administration Guide.*

Lock Files

On UNIX installations, each Directory Server instance contains a `locks` directory for storing `lock` related files. The following list shows the sample contents of the `locks` directory.

```
exports/
imports/
server/
```

The lock mechanisms stored under the subdirectories `exports`, `imports`, and `server` prevent simultaneous operations from conflicting with each other. The lock mechanisms allow one server instance to run at a time, with possible multiple export jobs. They also permit only one `ldif2db` import operation (or one `directoryserver ldif2db` operation for Solaris 9) at a time. This means that no export and `slapd` server operations can be run during an import.

This restriction does not apply to the `ldif2db.pl` script (`directoryserver ldif2db-task` for Solaris 9), since you can run multiple `ldif2db.pl` operations at any time.

Log Files

Each Directory Server instance contains a `logs` directory for storing log related files. The following list shows a sample of the `logs` directory contents.

```
access                audit.rotationinfo   pid
access.rotationinfo   errors               slapd.stats
audit                 errors.rotationinfo
```

- The content of the `access`, `audit`, and `errors` log files is dependent on the log configuration.
- The `slapd.stats` file is a memory-mapped file that cannot be read in an editor. It contains data collected by the Directory Server SNMP data collection component. This data is read by the SNMP subagent in response to SNMP attribute queries and is communicated to the SNMP master agent responsible for handling Directory Server SNMP requests.
- The `pid` is the `slapd` process identifier.

Access logs and their content are described in detail in Chapter 8, “Access Logs and Connection Codes”.

Chapter 8

Access Logs and Connection Codes

Sun ONE Directory Server 5.2 provides you with logs to help you monitor directory activity. Monitoring allows you to detect and remedy failures and, when done proactively, to anticipate and resolve potential problems before they result in failure or poor performance. To monitor your directory effectively, you need to understand the structure and content of the logs.

This chapter includes the following sections:

- Access Log Content
- Common Connection Codes
- LDAP Result Codes

For information on the error codes returned in log files, see Appendix A, “Error Codes.”

If you require further assistance in the investigation of your access log reports, please contact Sun ONE Technical Support:

`http://www.sun.com/service/sunone/index.html`

Access Log Content

The Sun ONE Directory Server 5.2 access log contains detailed information about client connections to the directory. A connection is a sequence of requests from the same client with the following structure:

- Connection record that gives the connection index and the IP address of the client
- Bind record
- Bind result record
- Sequence of operation request / operation result pairs of records (or individual records in the case of connection, closed, and abandon records)
- Unbind record
- Closed record

The access log files are located in the directory *ServerRoot*/slapd-*serverID*/logs. Each line of a log file begins with a timestamp [20/Aug/2002:11:39:51 -0700], where -0700 indicates the time difference in relation to GMT. The format of the timestamp may vary depending on the platform you are using. Apart from the connection, closed, and abandon records that appear individually, all records appear in pairs, consisting of a request for service record followed by a result record. These two records frequently appear on adjacent lines but this is not always the case.

This section presents the different levels of access logging available with Sun ONE Directory Server 5.2, then describes the default access logging content and ends with a description of the additional access logging level content. This section is divided into the following parts:

- “Access Logging Levels,” on page 291
- “Default Access Logging Content,” on page 291
- “Access Log Content for Additional Access Logging Levels,” on page 298

Access Logging Levels

Different levels of access logging exist. By changing the value of the `nsslapd-accesslog-level` configuration attribute, you can select the exact type of logging you require. The default level of logging is level 256 which logs access to an entry but you can choose from the following logging levels, combining more than one level to suit your needs:

0=No access logging

4=Logging for internal access operations

256=Logging for access to an entry

512=Logging for access to an entry and referrals

131072=Precise timing of operation duration. This gives microsecond resolution for the Elapsed Time item in the access log.

For example, if you want to log internal access operations, entry access, and referrals, you would set a value of 516 (512+4) in the `nsslapd-accesslog-level` configuration attribute. For further information on other access log configuration attributes, see Chapter 4, “Core Server Configuration Attributes.”

Default Access Logging Content

This section describes the access log content in detail, based on the default access logging level extract in Code Example 8-1.

Code Example 8-1 Access Log Extract with Default Access Logging Level (Level 256)

```
[22/Oct/2002:12:05:04 +0200] conn=25 op=-1 msgId=-1 - fd=32
slot=32 LDAP connection from 127.0.0.1 to 127.0.0.1
[22/Oct/2002:12:05:04 +0200] conn=25 op=0 msgId=1 - BIND
dn="cn=Directory Manager" method=128 version=3
[22/Oct/2002:12:05:04 +0200] conn=25 op=0 msgId=1 - RESULT err=0
tag=97 nentries=0 etime=0 dn="cn=directory manager"
[22/Oct/2002:12:07:19 +0200] conn=25 op=1 msgId=2 - ADD
dn="cn=Simon Campbell,ou=People,dc=Example,dc=COM"
[22/Oct/2002:12:07:20 +0200] conn=25 op=1 msgId=2 - RESULT err=0
tag=105 nentries=0 etime=1
[22/Oct/2002:12:07:26 +0200] conn=25 op=2 msgId=3 - UNBIND
[22/Oct/2002:12:07:26 +0200] conn=25 op=2 msgId=-1 - closing (3
ops still in progress) - U1
[22/Oct/2002:12:07:27 +0200] conn=25 op=-1 msgId=-1 - closed.
[22/Oct/2002:12:09:43 +0200] conn=26 op=-1 msgId=-1 - fd=32
slot=32 HTTP connection from 129.157.192.74 to 129.157.192.74
```

Code Example 8-1 Access Log Extract with Default Access Logging Level (Level 256)

```
[22/Oct/2002:12:05:04 +0200] conn=25 op=-1 msgId=-1 - fd=32
slot=32 LDAP connection from 127.0.0.1 to 127.0.0.1
[22/Oct/2002:12:09:45 +0200] conn=26 op=0 msgId=0 - DSML Batch
Request requestID=""
[22/Oct/2002:12:09:45 +0200] conn=26 op=2 msgId=1 - DSML Modify
requestID="" (parent msgId="0")
[22/Oct/2002:12:09:45 +0200] conn=26 op=2 msgId=1 - MOD
dn="cn=Simon Campbell,ou=People,dc=Example,dc=COM"
[22/Oct/2002:12:09:45 +0200] conn=26 op=2 msgId=1 - RESULT err=0
tag=103 nentries=0 etime=0
[22/Oct/2002:12:09:45 +0200] conn=26 op=0 msgId=-1 -
protocol=HTTP host="Foo" remlog="-" uname="-" date="[Tue, 22 Oct
2002 10:09:46 GMT]" request="POST /dsml HTTP/1.1" status="200 OK"
length=565
[22/Oct/2002:12:09:45 +0200] conn=26 op=0 msgId=-1 - closing (3
ops still in progress) -    (HTTP closure.)
[22/Oct/2002:12:09:46 +0200] conn=26 op=-1 msgId=-1 - closed.
[22/Oct/2002:12:11:01 +0200] conn=27 op=-1 msgId=-1 - fd=32
slot=32 LDAP connection from 127.0.0.1 to 127.0.0.1
[22/Oct/2002:12:11:01 +0200] conn=27 op=0 msgId=1 - BIND
dn="cn=Directory Manager" method=128 version=3
[22/Oct/2002:12:11:01 +0200] conn=27 op=0 msgId=1 - RESULT err=0
tag=97 nentries=0 etime=0 dn="cn=directory manager"
[22/Oct/2002:12:11:01 +0200] conn=27 op=1 msgId=2 - SRCH
base="dc=Example,dc=COM" scope=2 filter="(uid=scampbell)"
attrs=ALL
[22/Oct/2002:12:11:01 +0200] conn=27 op=1 msgId=2 - RESULT err=0
tag=101 nentries=1 etime=0
[22/Oct/2002:12:11:01 +0200] conn=27 op=2 msgId=3 - UNBIND
[22/Oct/2002:12:11:01 +0200] conn=27 op=2 msgId=-1 - closing (3
ops still in progress) - U1
[22/Oct/2002:12:11:02 +0200] conn=27 op=-1 msgId=-1 - closed.
```

Connection Number

Every external request is listed with an incremental connection number (`conn=25`, `conn=26`, and `conn=27` in the preceding example), starting at `conn=0` immediately after server startup. In this example, `conn=25` represents an LDAP add operation, `conn=26` is a DSML add operation and `conn=27` is an LDAP search operation.

Internal LDAP requests are not recorded in the access log by default. To activate the logging of internal access operations, specify an access logging level of 4 in the `nsslapd-accesslog-level` configuration attribute.

File Descriptor

Every connection from an external LDAP client to Directory Server requires a file descriptor, or socket descriptor, from the operating system (`fd=32` in the preceding example). `fd=32` indicates that file descriptor number 32 was used from the total pool of available file descriptors.

Slot Number

The slot number (`slot=32` in the preceding example), has the same meaning as file descriptor. It is a legacy section of the access log and can be ignored.

Operation Number

In processing an external request, Directory Server performs the required series of operations. For a specific connection, all operation request and operation result pairs are given incremental operation numbers beginning with `op=0` to identify the distinct operations being performed. In Code Example 8-1 on page 291, `op=0` is given for the bind operation request and result pair, then `op=1` for the LDAP search request and result pair, and so on. Should you see `op=-1` in the access log, it generally means that the LDAP request for this connection was not issued by an external LDAP client, but instead initiated internally.

Method Type

The method number, in this case `method=128`, indicates which LDAPv3 bind method was used by the client. There are three possible bind method values:

`0` = no authentication

`128` = simple bind with user password

`sasl`= SASL bind using external authentication mechanism

Version Number

The version number, in this case `version=3`, indicates the LDAP version number (either LDAPv2 or LDAPv3) that the LDAP client used to communicate with the LDAP server.

Error Number

The error number, in this case `err=0`, provides the LDAP result code returned from the LDAP operation performed. The LDAP error number `0` means that the operation was successful. For a more comprehensive list of LDAP result codes see “LDAP Result Codes,” on page 301.

Tag Number

The tag number, in this case `tag=97`, indicates the type of result returned, which is almost always a reflection of the type of operation performed. The tags used are the BER tags from the LDAP protocol. Commonly used tags include:

`tag=97` for a result from a client bind operation

`tag=100` indicates the actual entry for which you were searching

`tag=101` for a result from a search operation

`tag=103` for a result from a modify operation

`tag=105` for a result from an add operation

`tag=107` for a result from a delete operation

`tag=109` for a result from a moddn operation

`tag=111` for a result from a compare operation

`tag=115` indicates a search reference when the entry you perform your search on holds a referral to the entry you require. Search references are expressed in terms of a referral.

`tag=120` for a result from an extended operation

NOTE `tag=100` and `tag=115` are not result tags as such. It is unlikely that you will see them in your access log.

Number of Entries

The number of entries, in this case `nentries=0`, indicates the number of entries that were found matching the LDAP client's request.

Elapsed Time

Elapsed time, in this case `etime=1000`, indicates the amount of time (in seconds) that it took Directory Server to perform the LDAP operation. An `etime` value of 0 means that the operation actually took milliseconds to perform. If you want to have microsecond resolution for this item in the access log, enter a value of 131328 (256+131072) in the `nsslapd-accesslog-level` configuration attribute.

LDAP Request Type

The LDAP request type indicates the type of LDAP request being issued by the LDAP client. Possible values are:

`SRCH`=search

`MOD`=modify

`DEL`=delete

`ADD`=add

`MODDN`=moddn

`EXT`=extended operation

`ABANDON`=abandon operation

LDAP Response Type

The LDAP response type indicates the LDAP response being issued by the LDAP client. Possible values are:

`RESULT`=result

`ENTRY`=entry

`REFERRAL`=referral or search reference

Unindexed Search Indicator

The unindexed search indicator, `notes=U`, indicates that the search performed was unindexed, which means that the database itself had to be directly searched instead of the index file. Unindexed searches occur either when the All IDs Threshold was reached within the index file used for the search, when no index file existed, or when the index file was not configured in the way required by the search.

NOTE An unindexed search indicator is often accompanied by a large `etime` value, as unindexed searches are generally more time consuming.

Extended Operation OID

An extended operation OID, in this case either `EXT oid="2.16.840.1.113730.3.5.3"` or `EXT oid="2.16.840.1.113730.3.5.5"`, provides the OID of the extended operation being performed. Table 8-1 on page 296 provides the list of the LDAPv3 extended operations that are supported by Sun ONE Directory Server 5.2, and their OIDs.

Table 8-1 LDAPv3 Extended Operations Supported by Sun ONE Directory Server 5.2

Extended Operation Name	Description	OID
Sun ONE Directory Server 5.x Start Replication Request	Sent by a replication initiator to indicate that a replication session is requested.	2.16.840.1.113730.3.5.3
Sun ONE Directory Server 5.x Replication Response	Sent by a replication responder in response to a Start Replication Request Extended Operation or an End Replication Request Extended Operation.	2.16.840.1.113730.3.5.4
Sun ONE Directory Server 5.x End Replication Request	Sent to indicate that a replication session is to be terminated.	2.16.840.1.113730.3.5.5
Sun ONE Directory Server 5.x Replication Entry Request	Carries an entry, along with its state information (`csn` and `UniqueIdentifier`), and is used to perform a replica initialization.	2.16.840.1.113730.3.5.6
Sun ONE Directory Server 5.x Bulk Import Start	Sent by the client to request a bulk import together with the suffix being imported to *and* sent by the server to indicate that the bulk import may begin.	2.16.840.1.113730.3.5.7
Sun ONE Directory Server 5.x Bulk Import Finished	Sent by the client to signal the end of a bulk import *and* sent by the server to acknowledge it.	2.16.840.1.113730.3.5.8

Change Sequence Number

The change sequence number, in this case `csn=3b4c8cfb000000030000`, is the replication change sequence number, indicating that replication is enabled on this particular naming context.

Abandon Message

The abandon message, in this case, [06/Aug/2002:11:39:52 -0700] conn=12 op=2 ABANDON targetop=1 msgid=2 nentries=0 etime=0, indicates that an operation has been aborted, where nentries=0 indicates the number of entries sent before the operation was aborted, etime=0 indicates how much time (in seconds) had elapsed, and targetop=1 corresponds to an operation value from a previously initiated operation (that appears earlier in the access log).

There are two possible log ABANDON messages depending on whether the message ID succeeds in locating which operation was to be aborted or not. If the message ID succeeds in locating the operation (the targetop) then the log will read as above. However, if the message ID does not succeed in locating the operation or if the operation had already finished prior to the ABANDON request being sent, then the log will read as follows:

```
[06/Aug/2002:11:39:52 -0700] conn=12 op=2 ABANDON targetop=NOTFOUND
msgid=2
```

where targetop=NOTFOUND indicates that the operation to be aborted was either an unknown operation or already complete.

Message ID

The message ID, in this case msgid=2, is the LDAP operation identifier, as generated by the LDAP SDK client. The message ID may have a different value to the Sun ONE Directory Server Operation Number, but identifies the same operation. The message ID is used in the context of an ABANDON operation and tells the user which client operation is being abandoned.

NOTE The Sun ONE Directory Server operation number starts counting at 0. In the majority of LDAP SDK / client implementations the message ID number starts counting at 1. This explains why the message ID is frequently equal to the Sun ONE Directory Server operation number plus 1.

SASL Multi-Stage Bind Logging

Sun ONE Directory Server 5.2 logging for multi-stage binds is now more explicit. Each stage in the bind process is logged and, where appropriate, the progress statement SASL bind in progress is included.

NOTE The authenticated DN (the DN used for access control decisions) is logged in the BIND result line and not in the bind request line:

```
[06/Aug/2002:11:39:55 -0700] conn=14 op=1 RESULT err=0
tag=97 nentries=0 etime=0
dn="uid=coulbeck,dc=example,dc=com"
```

For SASL binds, the DN value displayed in the BIND request line is not used by the server and is, therefore, not relevant. However, given that the authenticated DN is the DN which, for SASL binds, must be used for audit purposes, it is essential that this be clearly logged. Having this authenticated DN logged in the BIND result line avoids any confusion as to which DN is which.

Access Log Content for Additional Access Logging Levels

This section presents the additional access logging levels available in the Sun ONE Directory Server 5.2 access log.

In Code Example 8-2 on page 298, access logging level 512, which logs access to entries and referrals, is enabled. In this extract, 6 entries and 1 referral are returned in response to the search request in bold.

Code Example 8-2 Access Log Extract with Entry Access and Referral Logging Level (Level 512)

```
06/Aug/2002:16:43:02 +0200] conn=306 fd=60 slot=60 connection from 127.0.0.1 to
127.0.0.1
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 SRCH base="dc=example,dc=com"
scope=2 filter="(description=*)" attrs=ALL
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY dn="ou=Special
Users,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=PD
Managers,ou=groups,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY dn="ou=Sun ONE
Servers,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 REFERRAL
```

Code Example 8-2 Access Log Extract with Entry Access and Referral Logging Level (Level 512) *(Continued)*

```
06/Aug/2002:16:43:02 +0200] conn=306 fd=60 slot=60 connection from 127.0.0.1 to
127.0.0.1
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 SRCH base="dc=example,dc=com"
scope=2 filter="(description=*)" attrs=ALL
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY
dn="cn=Accounting Managers,ou=groups,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=HR
Managers,ou=groups,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=QA
Managers,ou=groups,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=PD
Managers,ou=groups,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 ENTRY dn="ou=Sun ONE
Servers,dc=example,dc=com"
[06/Aug/2002:16:43:02 +0200] conn=306 op=0 REFERRAL
```

In Code Example 8-3 on page 299, access logging level 4, which logs internal operations, is enabled.

Code Example 8-3 Access Log Extract with Internal Access Operations Level (Level 4)

```
[06/Aug/2002:16:45:46 +0200] conn=Internal op=-1 SRCH
base="cn=\22dc=example,dc=com\22,cn=mapping
tree,cn=config"scope=0
filter="objectclass=nsMappingTree"attrs="nsslapd-referral"
options=persistent
06/Aug/2002:16:45:46 +0200] conn=Internal op=-1 RESULT err=0
tag=48 nentries=1etime=0
 [06/Aug/2002:16:45:46 +0200] conn=Internal op=-1 SRCH
base="cn=\22dc=example,dc=com\22,cn=mapping tree,cn=config"
scope=0 filter="objectclass=nsMappingTree" attrs="nsslapd-state"
[06/Aug/2002:16:45:46 +0200] conn=Internal op=-1 RESULT err=0
tag=48 nentries=1etime=0
```

Access log level 4 enables logging for internal operations which log the details of the search being performed, and the search base, scope, filter, and requested search attributes.

Connection Description

The connection description, in this case `conn=Internal`, indicates that the connection is an internal connection. The operation number `op=-1` indicates that the operation was initiated internally.

Options Description

The options description, in this case `options=persistent`, indicates that a persistent search is being performed. Persistent searches can be used as a form of monitoring. They can be configured to return changes to given configurations when changes occur.

NOTE The Sun ONE Directory Server 5.2 access log distinguishes between persistent and regular searches. Some earlier Directory Server releases did not make this distinction.

In Code Example 8-4 on page 300, both access logging level 512 and 4 are enabled, which results in both internal access operations, as well as entry access and referrals being logged.

Code Example 8-4 Access Log Extract with Internal Access Operation, Entry Access and Referral Logging Levels (Levels 4+512)

```
[06/Aug/2002:16:45:46 +0200] conn=Internal op=-1 ENTRY
dn="cn=\22dc=example,dc=com\22, cn=mapping tree, cn=config"
[06/Aug/2002:16:45:46 +0200] conn=Internal op=-1 ENTRY
dn="cn=\22dc=example,dc=com\22, cn=mapping tree, cn=config"
```

Common Connection Codes

A connection code is a code that is added to the `closed` log message to provide additional information related to the connection closure. Common connection codes include:

A1=Client aborts the connection.

B1=Corrupt BER tag encountered or BER tag is longer than the `nsslapd-maxbersize` attribute value. For further information about this configuration attribute, see “nsslapd-maxbersize (Maximum Message Size),” on page 116.

If BER tags, which encapsulate data being sent over the wire, are corrupt when they are received, a `B1` connection code is logged to the access log. BER tags can be corrupted due to physical layer network problems or bad LDAP client operations, such as an LDAP client aborting before receiving all request results.

B2=BER tag is longer than the `nsslapd-maxbersize` attribute value. For further information about this configuration attribute, see “nsslapd-maxbersize (Maximum Message Size),” on page 116.

B3=Corrupt BER tag encountered.

B4=Server failed to flush data response back to client.

P2=Closed or corrupt connection has been detected.

T1=Client does not receive a result within the specified idletimeout period.

T2=Server closed connection after ioblocktimeout period was exceeded.

U1= Connection closed by server after client sends an UNBIND request. The server will always close the connection when it sees an UNBIND request.

LDAP Result Codes

LDAP has a set of operation result codes with which you should be familiar. The following result codes may be generated by the LDAP server:

Table 8-2 LDAP Server Result Codes

Result Code	Meaning
0	Success
1	Operations error
2	Protocol error
3	Timelimit exceeded
4	Sizelimit exceeded
5	Compare false
6	Compare true
7	Authentication method not supported
8	Strong authentication required

Table 8-2 LDAP Server Result Codes *(Continued)*

Result Code	Meaning
9	Partial results and referral received
10	Referral received
11	Administrative limit exceeded
12	Unavailable critical extension
13	Confidentiality required
14	SASL bind in progress
16	No such attribute
17	Undefined attribute type
18	Inappropriate matching
19	Constraint violation
20	Type or value exists
21	Invalid syntax
32	No such object
33	Alias problem
34	Invalid DN syntax
35	Object is a leaf
36	Alias dereferencing problem
48	Inappropriate authentication
49	Invalid credentials
50	Insufficient access
51	Server is busy
52	Server is unavailable
53	Server is unwilling to perform
54	Loop detected
64	Naming violation
65	Object class violation
66	Operation not permitted on a non-leaf entry

Table 8-2 LDAP Server Result Codes *(Continued)*

Result Code	Meaning
67	Operation not permitted on a RDN
68	Entry already exists
69	Cannot modify object class
70	Results too large
71	Affects multiple servers
76	Virtual list view error

The following result codes may be generated by LDAP clients:

Table 8-3 LDAP Client Result Codes

Result Code	Meaning
80	Unknown error
81	Cannot contact LDAP server
82	Local error
83	Encoding error
84	Decoding error
85	Timed out
86	Unknown authentication method
87	Bad search filter
88	User cancelled operation
89	Bad parameter to an LDAP routine
90	Out of memory
91	Cannot connect to the LDAP server
92	Not supported by this version of LDAP
93	Requested LDAP control not found
94	No results returned
95	Additional results to return

Table 8-3 LDAP Client Result Codes *(Continued)*

Result Code	Meaning
96	Client detected loop
97	Referral hop limit exceeded

Part 4

Directory Server Schema

This part provides reference information on the directory schema, and contains a comprehensive list of the object classes, attributes, and operational attributes. It includes the following chapters:

- About Schema
- Object Class Reference
- Attribute Reference
- Operational Attributes

Chapter 9

About Schema

This chapter provides an overview of some of the basic concepts of the directory schema, and lists the files in which the schema is described. It describes object classes, attributes, and Object Identifiers (OIDs), and briefly discusses extending server schema and schema checking.

Schema Definition

The directory schema is a set of rules that defines how data can be stored in the directory. The data is stored in the form of directory entries. Each entry is a set of attributes and their values. Each entry must have an object class. The object class specifies the kind of object the entry describes and defines the set of attributes it contains. The schema defines the type of entries allowed, their attribute structure and the syntax of the attributes. The schema can be modified and extended if it does not meet your requirements.

To find detailed information about object classes, attributes, and how the Directory Server uses the schema, refer to the *Sun ONE Directory Server Deployment Guide*.

Object Classes

In LDAP, an object class defines the set of attributes that can be used to define an entry. The LDAP standard provides some basic types of object classes, including:

- Groups, including unordered lists of individual objects or groups of objects.
- Locations, such as the country name and description.
- Organizations.
- People.

- Devices.

Object classes may be subdivided into three types:

- Structural: indicates the attributes that the entry may have and where each entry may occur in the DIT. This object class represents the corresponding real world object. Entries must belong to a structural object class, so most object classes are structural object classes.
- Auxiliary: indicates the attributes that the entry may have. An auxiliary object class does not represent a real world object, but represents additional attributes that can be associated with a structural object class to supplement its specification. Each entry may belong to only a single structural object class, but may belong to zero or more auxiliary object classes.
- Abstract: defined only as a superclass or template for other (structural) object classes. An abstract object class is a way of collecting a set of attributes that will be common to a set of structural object classes, so that these classes may be derived as subclasses of the abstract class rather than being defined from scratch. An entry may not belong to an abstract object class.

NOTE Directory Server currently does not distinguish between structural and auxiliary object classes.

Required and Allowed Attributes

Every object class includes a number of required attributes and allowed attributes. Required attributes *must* be present in entries using the object class. All entries require the `objectClass` attribute, which defines the object classes assigned to the entry.

Allowed attributes *may* be present in entries using the object class.

Example: Object Class = person

Required Attributes

```
objectClass
cn (common name)
sn (surname)
```

Allowed Attributes

```
description
seeAlso
telephoneNumber
userPassword
```

Object Class Inheritance

Each entry must be assigned to one structural object class. All object classes inherit from the `top` object class. They can also inherit from other object classes. The server's object class structure determines the list of required and allowed attributes for a particular entry. For example, a `person` entry is usually defined with the following object class structure:

```
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgperson
```

In this structure, the `inetOrgperson` inherits from the `organizationalPerson` and `person` object classes. Therefore, when you assign the `inetOrgperson` object class to an entry, it automatically inherits the required and allowed attributes from the superior object class.

Note that object class inheritance is dependent on the order in which the object classes appear in the schema `.ldif` files. The order in which object classes appear in the `.ldif` file must be consistent with the object class hierarchy, otherwise the server will not start. An object class that inherits from another object class must therefore appear *after* this object class in the schema `.ldif` file.

Attributes

Directory data is represented as attribute-value pairs. Any piece of information in the directory is associated with a descriptive attribute.

For instance, the `commonName`, or `cn`, attribute is used to store a person's name. A person named Barbara (Babs) Jensen can be represented in the directory as

```
cn: Babs Jensen
```

Each person entered in the directory can be defined by the collection of attributes in the `inetOrgperson` object class. Other attributes used to define this entry could include:

```
givenname: Barbara
surname: Jensen
mail: bjensen@example.com
```

Attribute Syntax

Each attribute has a syntax definition that describes the type of information provided by the attribute.

Attribute syntax is used by the Directory Server to perform sorting and pattern matching.

Table 9-1 lists the different syntax methods that can be applied to attributes, and gives an OID and a definition for each syntax method.

Table 9-1 Attribute Syntax

Syntax and OID	Definition
Binary 1.3.6.1.4.1.1466.115.121.1.5	Indicates that values for this attribute are treated as binary data, and cannot be matched.
Boolean 1.3.6.1.4.1.1466.115.121.1.7	Indicates that this attribute has one of only two values: `True` or `False`.
Country String 1.3.6.1.4.1.1466.115.121.1.11	Indicates that values for this attribute are limited to exactly two printable string characters, representing the ISO code of a country for example `fr`.
DN 1.3.6.1.4.1.1466.115.121.1.12	Indicates that values for this attribute are DNs (distinguished names).
DirectoryString 1.3.6.1.4.1.1466.115.121.1.15	Indicates that values for this attribute are UTF-8 encoded characters, and are treated as case insensitive.
GeneralizedTime 1.3.6.1.4.1.1466.115.121.1.24	Indicates that values for this attribute are encoded as printable strings. The time zone must be specified. It is strongly recommended to use GMT.
IA5String 1.3.6.1.4.1.1466.115.121.1.26	Indicates that values for this attribute must contain only ASCII characters, and are treated as case sensitive.
INTEGER 1.3.6.1.4.1.1466.115.121.1.27	Indicates that valid values for this attribute are numbers.
OctetString 1.3.6.1.4.1.1466.115.121.1.40	Same behavior as binary.

Table 9-1 Attribute Syntax *(Continued)*

Syntax and OID	Definition
Postal Address 1.3.6.1.4.1.1466.115.121.1.41	Indicates that values for this attribute are encoded as *dstring[$ dstring]** where each *dstring* component is encoded as a value with DirectoryString syntax. Backslashes and dollar characters within *dstring* must be quoted, so that they will not be mistaken for line delimiters. Many servers limit the postal address to 6 lines of up to thirty characters. For example: `1234 Main St.$Anytown, TX 12345$USA`
TelephoneNumber 1.3.6.1.4.1.1466.115.121.1.50	Indicates that values for this attribute are in the form of telephone numbers. It is recommended to use telephone numbers in international form.
URI 1.3.6.1.4.1.4401.1.1.1	Indicates that the values for this attribute are in the form of a URL, introduced by a string such as `http://`, `https://`, `ftp`, `LDAP`. The URI has the same behavior as IA5String. See RFC 2396.

Single-Valued and Multi-Valued Attributes

By default, most attributes are multi-valued. This means that an entry can contain the same attribute with multiple values. For example, `cn`, `tel` and `objectClass` are all attributes that can have more than one value. Attributes that are single-valued (only one instance of the attribute can be specified) are noted as such. For example, `uidNumber` can have only one possible value.

Schema Supported by Directory Server 5.2

The schema provided with Sun ONE Directory Server 5.2 is described in a set of files stored in the following directory:

ServerRoot`/slapd-`*serverID*`/config/schema`

You can modify the schema by creating new object classes and attributes. These modifications are stored in a file called `99user.ldif`. You should not modify the standard files provided with the Directory Server, because you run the risk of breaking compatibility with other Sun ONE products, or of causing interoperability problems with directory servers from vendors other than Sun ONE.

For more information about how the Directory Server stores information and suggestions for planning directory schema, refer to the *Sun ONE Directory Server Deployment Guide*.

The following tables list the schema files that are provided with Sun ONE Directory Server. Table 9-2 lists the schema files that are used by the Directory Server.

Table 9-2 Schema Files Used by Directory Server

Schema Filename	Purpose
`00core.ldif`	Recommended core schema from the X.500 and LDAP standards (RFCs), and schema used by the Directory Server itself.
`05rfc2247.ldif`	Schema from RFC 2247 and related pilot schema 'Using Domains in LDAP / X.500 Distinguished Names."
`05rfc2927.ldif`	Schema from RFC 2927 "MIME Directory Profile for LDAP Schema."
`11rfc2307.ldif`	Schema from RFC 2307 "An Approach for Using LDAP as a Network Information Service."
`20subscriber.ldif`	Common schema elements for Sun ONE-Nortel subscriber interoperability.
`25java-object.ldif`	Schema from RFC 2713 "Schema for Representing Java™ Objects in an LDAP Directory."
`28pilot.ldif`	Schema from the pilot RFCs, especially RFC 1274, that is no longer recommended for use in new deployments.
`30ns-common.ldif`	Common Sun ONE schema.
`50ns-admin.ldif`	Schema used by Sun ONE Administration Services.
`50ns-directory.ldif`	Additional schema used by Directory Server 4.x.
`50ns-value.ldif`	Sun ONE servers "value item" schema.
`99user.ldif`	Customer modifications to the schema.

Table 9-3 lists the schema files that are used by other Sun ONE products.

Table 9-3 Schema Files Used by Other Sun ONE Products

Schema Filenames	Purpose
`50iplanet-servicemgt.ldif`	Sun ONE service management schema elements.

Table 9-3 Schema Files Used by Other Sun ONE Products *(Continued)*

Schema Filenames	Purpose
`50ns-calendar.ldif`	Sun ONE Calendar Server schema.
`50ns-certificate.ldif`	Schema for Sun ONE Certificate Management System.
`50ns-compass.ldif`	Schema for the Netscape Compass Server.
`50ns-delegated-admin.ldif`	Schema for Sun ONE Delegated Administrator 4.5.
`50ns-legacy.ldif`	Legacy Netscape Schema.
`50ns-mail.ldif`	Schema for Sun ONE Messaging Server.
`50ns-mcd-browser.ldif`	Schema for Netscape Mission Control Desktop - Browser.
`50ns-mcd-config.ldif`	Schema for Netscape Mission Control Desktop - Configuration.
`50ns-mcd-li.ldif`	Schema for Netscape Mission Control Desktop - Location Independence.
`50ns-mcd-mail.ldif`	Schema for Netscape Mission Control Desktop - Mail.
`50ns-media.ldif`	Schema for Netscape Media Server.
`50ns-mlm.ldif`	Schema for Sun ONE Mailing List Manager.
`50ns-msg.ldif`	Schema for Sun ONE Web Mail.
`50ns-netshare.ldif`	Schema for Sun ONE Netshare.
`50ns-news.ldif`	Schema for Sun ONE Collabra Server.
`50ns-proxy.ldif`	Schema for Sun ONE Proxy Server.
`50ns-wcal.ldif`	Schema for Sun ONE Web Calendaring.
`50ns-web.ldif`	Schema for Sun ONE Web Server.

Object Identifiers (OIDs)

Object identifiers (OIDs) are assigned to all attributes and object classes to conform to the LDAP and X.500 standards. An OID is a sequence of integers, typically written as a dot-separated string. When no OID is specified, the Directory Server automatically uses *ObjectClass_name*-oid and *attribute_name*-oid.

Sun ONE Directory Server uses Sun based OIDs. Previous versions of Directory Server used Netscape based OIDs.

Sun ONE-defined attributes and object classes using the Sun base have the base OID of `1.3.6.1.4.1.42.2.27.9`.

Sun ONE-defined attributes and object classes using the Netscape base have the base OID of `2.16.840.1.113730.3`

For more information about OIDs, or to request a prefix for your enterprise, please go to the IANA (Internet Assigned Number Authority) website at `http://www.iana.org/`.

Extending Server Schema

The Directory Server schema includes hundreds of object classes and attributes that can be used to meet most of your requirements. This schema can be extended with new object classes and attributes that meet evolving requirements for the directory service in the enterprise.

When adding new attributes to the schema, a new object class should be created to contain them (adding a new attribute to an existing object class can compromise the Directory Server's compatibility with existing LDAP clients that rely on the standard LDAP schema and may cause difficulties when upgrading the server).

For more information about extending server schema, refer to the *Sun ONE Directory Server Deployment Guide.*

Schema Checking

You should run Directory Server with schema checking turned on.

The schema checking capability of Sun ONE Directory Server checks entries when you add them to the directory or when you modify them, to verify that:

- Object classes and attributes in the entry are defined in the directory schema
- Attributes required for an object class are contained in the entry
- Only attributes allowed by the object class are contained in the entry

Schema checking also occurs when importing a database using LDIF. For more information, refer to the *Sun ONE Directory Server Administration Guide.*

NOTE In the current version of Sun ONE Directory Server, schema checking does *not* enforce the validity of values with respect to their syntax. This functionality is slated for a future version of the product.

Chapter 10

Object Class Reference

This chapter contains an alphabetical list of the object classes accepted by the default schema. It provides a definition of each object class, and lists its Required and Allowed Attributes. If an object class inherits attributes from other object classes, the inherited attributes are shown in italics. An object class that inherits from another object class must appear *after* this object class in the schema `.ldif` file, otherwise the server will not start.

This chapter distinguishes between *structural*, and *auxiliary*, and *abstract* object classes. All directory entries are instances of structural object classes. Structural object classes represent real world objects, such as people, buildings, or countries. Auxiliary object classes allow you to extend object class definitions for specific entries. Abstract object classes are defined purely as a superclasses or templates for other (structural) object classes. Object classes listed here can be considered structural, unless otherwise indicated.

The object classes listed in this chapter are available to support your own information in the Directory Server. Object classes that are used by the Directory Server or other Sun ONE products for internal operations are not documented here. For information about these internal object classes, refer to Chapter 5, “Plug-In Implemented Server Functionality.”

NOTES

1. The schema provided with Sun ONE Directory Server differs from that specified in RFC 2256 with regard to the `groupOfNames` and `groupOfUniqueNames` object classes. In the schema provided, the `member` and `uniquemember` attribute types are optional, while RFC 2256 specifies that at least one value for these types must be present in the respective object class.
2. The LDAP RFCs (and X.500 standards) allow for an object class to have more than one superior. This behavior is not currently supported by Directory Server.

account

Definition

Used to define entries representing computer accounts.

This object class is defined in RFC 1274.

Superior Class

top

OID

0.9.2342.19200300.100.4.5

Required Attributes

Attribute	Description
objectClass	Defines the object class for the entry.
uid (userID)	Identifies the account's user ID.

Allowed Attributes

Attribute	Description
description	Text description of the entry.
host	Hostname of the computer on which the account resides.
l (localityName)	Place in which the account is located.
o (organizationName)	Organization to which the account belongs.
ou (organizationUnitName)	Organizational unit to which the account belongs.
seeAlso	DN to information relevant to the account.

alias

Definition

Abstract object class, used to point to other entries in the directory tree.

Note that alias dereferencing is not supported in Sun ONE Directory Server.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.1

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
aliasedObjectName	Distinguished name of the entry for which this entry is an alias.

bootableDevice

Definition

Auxiliary object class that specifies a device with boot parameters.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.12

Allowed Attributes

Attribute	Description
bootFile	The name of the boot image.
bootParameter	Boot parameters.

changeLogEntry

Definition

Internal object class, used to represent changes made to the directory server. You can configure Sun ONE Directory Server 5.2 to maintain a change log that is compatible with the change log implemented in Directory Server 4.x, 5.0, and 5.1 by enabling the Retro Changelog plug-in. Each entry in the change log has the object class `changeLogEntry`. This object class is defined in the Changelog Internet Draft.

Superior Class

top

OID

2.16.840.1.113730.3.2.1

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
changeNumber	Number assigned arbitrarily to the changelog.
changeTime	The time at which a change took place.
changeType	The type of change performed on an entry.
targetDn	The distinguished name of an entry added, modified, or deleted on a supplier server.

Allowed Attributes

Attribute	Description
changes	Changes made to Directory Server.
deleteOldRdn	A flag that defines whether the old Relative Distinguished Name (RDN) of the entry should be kept as a distinguished attribute of the entry, or deleted.
newRdn	New RDN of an entry that is the target of a modRDN or modDN operation.
newSuperior	Name of the entry that becomes the immediate superior of the existing entry, when processing a modDN operation.

cosClassicDefinition

Definition

Identifies the template entry using both the template entry's DN (as specified in the `cosTemplateDn` attribute) and the value of one of the target entry's attributes (as specified in the `cosSpecifier` attribute).

This object class is defined in Sun ONE Directory Server.

Superior Class

cosSuperDefinition

OID

2.16.840.1.113730.3.2.100

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cosAttribute	Provides the name of the attribute for which you want to generate a value. You can specify more than one cosAttribute value.

Allowed Attributes

Attribute	Description
cn (commonName)	Common name of the entry.
cosSpecifier	Specifies the attribute value used by a classic CoS, which, along with the template entry's DN, identifies the template entry.
cosTemplateDn	Provides the DN of the template entry associated with the CoS definition.
description	Text description of the entry.

cosDefinition

Definition

Defines the Class of Service you are using. This object class is supported for compatibility with the Directory Server 4.1 CoS Plugin. It will be deprecated in a future Directory Server release.

This object class is defined in Sun ONE Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.84

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
aci	Evaluates what rights are granted or denied when the directory receives an LDAP request from a client.
cn (commonName)	Common name of the entry.
cosAttribute	Provides the name of the attribute for which you want to generate a value. You can specify more than one `cosAttribute` value.
cosSpecifier	Specifies the attribute value used by a classic CoS, which, along with the template entry's DN, identifies the template entry.
cosTargetTree	Determines the subtree of the DIT to which the CoS schema applies. This attribute is single-valued. Using multiple values will have a negative performance impact.
cosTemplateDn	Provides the DN of the template entry associated with the CoS definition.
uid (userID)	Identifies the user id.

cosIndirectDefinition

Definition

Identifies the template entry using the value of one of the target entry's attributes. The attribute of the target entry is specified in the `cosIndirectSpecifier` attribute.

This object class is defined in Sun ONE Directory Server.

Superior Class

cosSuperDefinition

OID

2.16.840.1.113730.3.2.102

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cosAttribute	Provides the name of the attribute for which you want to generate a value. You can specify more than one `cosAttribute` value.

Allowed Attributes

Attribute	Description
cn (commonName)	Common name of the entry.
cosIndirectSpecifier	Specifies the attribute value used by an indirect CoS to identify the template entry.
description	Text description of the entry.

cosPointerDefinition

Definition

Identifies the template entry associated with the CoS definition using the template entry's DN value. The DN of the template entry is specified in the `cosTemplateDn` attribute.

This object class is defined in Sun ONE Directory Server.

Superior Class

cosSuperDefinition

OID

2.16.840.1.113730.3.2.101

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cosAttribute	Provides the name of the attribute for which you want to generate a value. You can specify more than one `cosAttribute` value.

Allowed Attributes

Attribute	Description
cn (commonName)	Common name of the entry.
cosTemplateDn	Provides the DN of the template entry associated with the CoS definition.
description	Text description of the entry.

cosSuperDefinition

Definition

All CoS definition object classes inherit from the `cosSuperDefinition` object class.

This object class is defined in Sun ONE Directory Server.

Superior Class

ldapSubEntry

OID

2.16.840.1.113730.3.2.99

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cosAttribute	Provides the name of the attribute for which you want to generate a value. You can specify more than one `cosAttribute` value.

Allowed Attributes

Attribute	Description
cn (commonName)	Common name of the entry.
description	Text description of the entry.

cosTemplate

Definition

Contains a list of the shared attribute values.

This object class is defined in Sun ONE Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.128

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
cn (commonName)	Common name of the entry.
cosPriority	Specifies which template provides the attribute value, when CoS templates compete to provide an attribute value.

country

Definition

Used to define entries that represent countries.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.2

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
c (countryName)	Contains the two-character code representing country names in the directory (as defined in ISO-3166.)

Allowed Attributes

Attribute	Description
description	Text description of the country.
searchGuide	Specifies information for suggested search criteria when using the entry as the base object in the directory tree for a search operation (Distinguished Name).

dcObject

Definition

This auxiliary object class defines a domain component, such as a network domain that is associated with the entry. This object class is defined as auxiliary because it is commonly used in combination with another object class, such as `organization`, `organizationUnit`, or `locality`. For example:

```
dn: ou=Engineering,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
objectClass: dcObject
ou: Engineering
dc: eng
```

This object class is defined in RFC 2247.

NOTE Suffixes often contain the `dc` attribute, such as `dc=example,dc=com` in the example above. Suffixes use the `dc` attribute to suggest that the directory they represent is associated with a certain domain. However, the suffix is a string associated with a database and is not related to the `dcObject` object class.

Superior Class

top

OID

1.3.6.1.4.1.1466.344

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
dc (domainComponent)	One component of a domain name.

See Also

`domain`

device

Definition

Used to store information about network devices, such as printers, in the directory.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.14

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The common name of the series.

Allowed Attributes

Attribute	Description
description	Text description of the device.
l (localityName)	Place in which the device is located.
o (organizationName)	Organization to which the device belongs.
ou (organizationUnitName)	Organizational unit to which the device belongs.
owner	Distinguished name of the person responsible for the device.
seeAlso	DN to information relevant to the device.
serialNumber	Serial number of the device.

document

Definition

Used to define entries that represent documents in the directory.

This object class is defined in RFC 1274.

Superior Class

pilotObject

OID

0.9.2342.19200300.100.4.6

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
documentIdentifier	Unique identifier for a document.

Allowed Attributes

Attribute	Description
abstract	Abstract of the document.
audio	Stores a sound file in binary format.
authorCn	Author's common or given name.
authorSn	Author's surname.
cn (commonName)	Common name of the document.
description	Text description of the document.
ditRedirect	Distinguished name to use as a redirect for the entry.
documentAuthor	Distinguished name of the document author.
documentLocation	Location of the original document.
documentPublisher	Person or organization that published the document.
documentStore	Not defined.
documentTitle	The document's title.
documentVersion	The document's version number.

info	Information about the object.
jpegPhoto	Photo in jpeg format.
keyWords	Keywords that describe the document.
l (localityName)	Place in which the document is located.
lastModifiedBy	Distinguished name of the last user to modify the document.
lastModifiedTime	Last time the document was modified.
manager	Distinguished name of the object's manager.
o (organizationName)	Organization to which the document belongs.
obsoletedByDocument	Distinguished name of a document that obsoletes this document.
obsoletesDocument	Distinguished name of a document that is obsoleted by this document.
ou (organizationUnitName)	Organizational unit to which the document belongs.
photo	Photo of the document, in binary form.
seeAlso	DN to information relevant to the document.
subject	Subject of the document.
uniqueIdentifier	Specific item used to distinguish between two entries when a distinguished name has been reused.
updatedByDocument	Distinguished name of a document that is an updated version of this document.
updatesDocument	Distinguished name of a document for which this document is an updated version.

documentSeries

Definition

Used to define an entry that represents a series of documents.

This object class is defined in RFC 1274.

Superior Class

top

OID

0.9.2342.19200300.100.4.9

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The common name of the series.

Allowed Attributes

Attribute	Description
description	Text description of the series.
l (localityName)	Place in which the series is located.
o (organizationName)	Organization to which the series belongs.
ou (organizationUnitName)	Organizational unit to which the series belongs.
seeAlso	DN to information relevant to the series.
telephoneNumber	Telephone number of the person responsible for the series.

domain

Definition

Used to represent Internet Domains (for example, `example.com`). The `domainComponent` attribute should be used for naming entries of this object class.

The `domain` object class can only be used with an entry that does not correspond to an organization, organizational unit, or other type of object for which an object class has been defined. The domain object class requires that the `domainComponent` attribute be present, and allows several other attributes to be present in the entry. These allowed attributes are used to describe the object represented by the domain, and may also be useful when searching.

This object class is defined in RFC 2247.

Superior Class

top

OID

0.9.2342.19200300.100.4.13

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
dc (domainComponent)	One component of a domain name.

Allowed Attributes

Attribute	Description
associatedName	Entry in the organizational directory tree associated with a DNS domain.
businessCategory	Type of business in which this domain is engaged.
description	Text description of the domain.
destinationIndicator	Country and city associated with the entry needed to provide Public Telegram Service.
fax (facsimileTelephoneNumber)	Domain's fax number.
internationaliSDNNumber	Domain's ISDN number.

l (localityName)	Place in which the domain is located.
o (organizationName)	Organization to which the domain belongs.
physicalDeliveryOfficeName	Location where physical deliveries can be made.
postOfficeBox	Domain's post office box.
postalAddress	Domain's mailing address.
postalCode	The postal code for this address (such as a United States zip code).
preferredDeliveryMethod	Domain's preferred method of contact or delivery.
registeredAddress	Postal address suitable for reception of expedited documents, where the recipient must verify delivery.
searchGuide	Specifies information for suggested search criteria when using the entry as the base object in the directory tree for a search operation.
seeAlso	DN to information relevant to the domain.
st (stateOrProvinceName)	State or province in which the domain is located.
street (streetAddress)	Street address in which the domain is located.
telephoneNumber	Domain's telephone number.
telexNumber	Identifier for a domain's teletex terminal.
telexNumber	Domain's telex number.
userPassword	Password with which the entry can bind to the directory.
x121Address	X.121 address of the domain.

See Also

`dcObject`

domainRelatedObject

Definition

Used to define entries that represent DNS/NRS domains that are “equivalent” to an X.500 domain, for example, an organization or organizational unit.

This object class is defined in RFC 1274.

Superior Class

top

OID

0.9.2342.19200300.100.4.17

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
associatedDomain	Specifies a DNS domain associated with an object in the directory tree.

dSA

Definition

Used to define entries representing Directory Server Agents.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.13

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The Directory Server Agent's common name.
presentationAddress	Contains an OSI presentation address for the entry.

Allowed Attributes

Attribute	Description
description	Text description of the series.
knowledgeInformation	This attribute is no longer used.
l (localityName)	Place in which the series is located.
o (organizationName)	Organization to which the series belongs.
ou (organizationUnitName)	Organizational unit to which the series belongs.
seeAlso	DN to information relevant to the series.
supportedApplicationContext	This attribute contains the identifiers of OSI application contexts.

extensibleObject

Definition

Auxiliary object class which, when present in an entry, permits the entry to optionally hold any attribute. The allowed attribute list of this class is implicitly the set of all attributes known to the server.

This object class is defined in RFC 2252.

Superior Class

top

OID

1.3.6.1.4.1.1466.101.120.111

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

All attributes known to the server.

friendlyCountry

Definition

Used to define country entries in the directory tree. This object class is used to allow more user-friendly country names than those allowed by the country object class.

This object class is defined in RFC 1274.

Superior Class

country

OID

0.9.2342.19200300.100.4.18

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
co (friendlyCountryName)	Stores the name of a country.
c (countryName)	Contains the two-character code representing country names in the directory (as defined in ISO-3166).

Allowed Attributes

Attribute	Description
description	Text description of the country.
searchGuide	Specifies information for suggested search criteria when using the entry as the base object in the directory tree for a search operation.

groupOfCertificates

Definition

Used to describe a set of X.509 certificates. Any certificate that matches one of the `memberCertificateDescription` values is considered a member of the group.

This object class is defined in Sun ONE Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.31

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The group's common name.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which the group is engaged.
description	Text description of the group's purpose.
memberCertificateDescription	Values used to determine if a particular certificate is a member of this group.
o (organizationName)	Organization to which the group of certificates belongs.
ou (organizationUnitName)	Organizational unit to which the group belongs.
owner	Distinguished name of the person responsible for the group.
seeAlso	DN to information relevant to the group.

groupOfNames

Definition

Used to define entries for a group of names.

NOTE The definition in Sun ONE Directory Server differs from the standard definition. In the standard definition, `member` is a required attribute. In Directory Server `member` is an allowed attribute. Directory Server therefore allows a group to have no member.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.9

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The group's common name.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which the group is engaged.
description	Text description of the group's purpose.
member	Distinguished name of a group member.
o (organizationName)	Organization to which the group belongs.
ou (organizationUnitName)	Organizational unit to which the group belongs.
owner	Distinguished name of the person responsible for the group.
seeAlso	DN to information relevant to the group.

groupOfUniqueNames

Definition

Used to define entries for a group of unique names.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.17

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The group's common name.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which the group is engaged.
description	Text description of the group's purpose.
o (organizationName)	Organization to which the group belongs.
ou (organizationUnitName)	Organizational unit to which the group belongs.
owner	Distinguished name of the person responsible for the group.
seeAlso	DN to information relevant to the group.
uniqueMember	Distinguished name of a unique group member, optionally followed by a hash (#) and a unique identifier label.

groupOfURLs

Definition

An auxiliary object class of `groupOfUniqueNames` or `groupOfNames`. The group consists of a list of labeled URLs.

This object class is defined in Sun ONE Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.33

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The group's common name.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which the group is engaged.
description	Text description of the group's purpose.
memberURL	URL associated with each member of the group.
o (organizationName)	Organization to which the group belongs.
ou (organizationUnitName)	Organizational unit to which the group belongs.
owner	Distinguished name of the person responsible for the group.
seeAlso	DN to information relevant to the group.

ieee802Device

Definition

Auxiliary object class, specifying a device with a MAC address.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.11

Allowed Attributes

Attribute	Description
macAddress	The MAC address of the device.

inetOrgPerson

Definition

Used to define entries representing people in an organization's enterprise network.

This object class is defined in RFC 2798.

Superior Class

organizationalPerson

OID

2.16.840.1.113730.3.2.2

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The person's common name.
sn (surname)	The person's surname, or last name.

Allowed Attributes

Attribute	Description
audio	Stores a sound file in binary format.
businessCategory	Type of business in which the person is engaged.
carLicense	The license plate number of the person's vehicle.
departmentNumber	Department for which the person works.
description	Text description of the person.
destinationIndicator	Country and city associated with the entry needed to provide Public Telegram Service.
displayName	Preferred name of a person to be used when displaying entries.
employeeNumber	The person's employee number.
employeeType	The person's type of employment (for example, full time).
fax (facsimileTelephoneNumber)	The person's fax number.

givenName	The person's given, or first, name.
homePhone	The person's home phone number.
homePostalAddress	The person's home mailing address.
initials	The person's initials.
internationaliSDNNumber	The person's ISDN number.
jpegPhoto	Photo in JPEG format.
l (localityName)	Place in which the person is located.
labeledURI	Universal Resource Identifier that is relevant to the person.
mail	The person's email address.
manager	Distinguished name of the person's manager.
mobile	The person's mobile phone number.
o (organizationName)	Organization to which the person belongs.
ou (organizationUnitName)	Organizational unit to which the person belongs.
pager (pagerTelephoneNumber)	The person's pager number.
photo	Photo of the person, in binary form.
physicalDeliveryOfficeName	Location where physical deliveries can be made to the person.
postOfficeBox	The person's post office box.
postalAddress	The person's mailing address.
postalCode	The postal code for this address (such as a United States zip code).
preferredDeliveryMethod	The person's preferred method of contact or delivery.
preferredLanguage	The person's preferred written or spoken language.
registeredAddress	Postal address suitable for reception of courier documents, where the recipient must verify delivery.
roomNumber	The room number in which the person is located.
secretary	Distinguished name of the person's secretary or administrative assistant.
seeAlso	DN to information relevant to the person.
st (stateOrProvinceName)	State or province in which the person is located.

street (streetAddress)	Street address at which the person is located.
telephoneNumber	The person's telephone number.
telexNumber	Identifier for the person's teletex terminal.
telexNumber	The person's telex number.
title	The person's job title.
uid (userID)	Identifies the person's user id (usually the logon ID).
userCertificate	Stores a user's certificate in clear text (not used).
userPassword	Password with which the entry can bind to the directory.
userSMIMECertificate	Stores a user's certificate in binary form. Used by Netscape Communicator for S/MIME.
x121Address	X.121 address of the person.
x500UniqueIdentifier	Reserved.

ipHost

Definition

Auxiliary object class, specifying an abstraction of a host, an IP device. The distinguished value of the cn attribute denotes the canonical name of the host.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.6

Required Attributes

Attribute	Description
cn (commonName)	The common name of the host.
ipHostNumber	The IP address, expressed as a dotted decimal.

Allowed Attributes

Attribute	Description
description	Text description of the host.
manager	Distinguished name of the object's manager.

ipNetwork

Definition

Auxiliary object class, specifying an abstraction of a host, an IP device. The distinguished value of the cn attribute denotes the canonical name of the host.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.7

Required Attributes

Attribute	Description
cn (commonName)	The common name of the host.
ipHostNumber	The IP address, expressed as a dotted decimal.

Allowed Attributes

Attribute	Description
description	Text description of the host.
manager	Distinguished name of the object's manager.

ipProtocol

Definition

Abstraction of an IP protocol. This object class maps a protocol number to one or more names. The distinguished value of the cn attribute denotes the protocol's canonical name.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.4

Required Attributes

Attribute	Description
cn (commonName)	The common name of the protocol.
ipProtocolNumber	The IP protocol number.

Allowed Attributes

Attribute	Description
description	Text description of the host.

ipService

Definition

Abstraction an Internet Protocol service. This object class maps an IP port and protocol (such as tcp or udp) to one or more names. The distinguished value of the cn attribute denotes the service's canonical name.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.3

Required Attributes

Attribute	Description
cn (commonName)	The common name of the protocol.
ipServicePort	The IP service port number.
ipServiceProtocol	The IP service protocol.

Allowed Attributes

Attribute	Description
description	Text description of the host.

javaContainer

Definition
Represents a container for a Java object.

This object class is defined in RFC 2713.

Superior Class
top

OID
1.3.6.1.4.1.42.2.27.4.2.1

Required Attributes

Attribute	Description
cn (commonName)	The common name of the protocol.

javaMarshalledObject

Definition
Auxiliary object class that represents a Java marshalled object. It must be mixed with a structural object class.

This object class is defined in RFC 2713.

Superior Class
javaObject

OID
1.3.6.1.4.1.42.2.27.4.2.8

Required Attributes

Attribute	Description
javaSerializedData	The serialized form of a Java object.

javaNamingReference

Definition

Auxiliary object class that represents a JNDI reference. It must be mixed in with a structural object class.

This object class is defined in RFC 2713.

Superior Class

javaObject

OID

1.3.6.1.4.1.42.2.27.4.2.7

Allowed Attributes

Attribute	Description
javaFactory	The fully qualified class name of the object factory.
javaReferenceAddress	The sequence of addresses of a JNDI reference.

javaObject

Definition

Abstract object class that represents a Java object.

This object class is defined in RFC 2713.

Superior Class

top

OID

1.3.6.1.4.1.42.2.27.4.2.4

Required Attributes

Attribute	Description
javaClassName	The fully qualified name of the Java object's distinguished class or interface.

Allowed Attributes

Attribute	Description
description	Text description of the host.
javaClassNames	The Java object's fully qualified class or interface names.
javaCodebase	The Java class definition's locations.
javaDoc	A pointer to the Java documentation for the class.

javaSerializedObject

Definition

Auxiliary object class that represents a Java serialized object. It must be mixed in with a structural object class.

This object class is defined in RFC 2713.

Superior Class

javaObject

OID

1.3.6.1.4.1.42.2.27.4.2.5

Required Attributes

Attribute	Description
javaSerializedData	The serialized form of a Java object.

labeledURIObject

Definition

Auxiliary object class that can be added to existing directory objects to allow for inclusion of URI values. This approach does not preclude including the labeledURI attribute type directly in other object classes as appropriate.

This object class is defined in RFC 2079.

Superior Class

top

OID

1.3.6.1.4.1.250.3.15

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
labeledURI	Universal Resource Identifier that is relevant to the entry.

ldapSubentry

Definition

This structural object class may be used to indicate operations and management related entries in the directory, called LDAP Subentries.

This object class is defined in the LDAP Subentry Internet Draft.

Superior Class

top

OID

2.16.840.1.113719.2.142.6.1.1

Allowed Attributes

Attribute	Description
cn (commonName)	Identifies the name of the subentry.

locality

Definition

Used to define entries that represent localities or geographic areas.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.3

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
description	Text description of the locality.
l (localityName)	Place in which the entry is located.
searchGuide	Specifies information for a suggested search criteria when using the entry as the base object in the directory tree for a search operation.
seeAlso	DN to information relevant to the locality.
st (stateOrProvinceName)	State or province to which the locality belongs.
street (streetAddress)	Street address associated with the locality.

newPilotPerson

Definition

Used as a subclass of person, to allow the use of a number of additional attributes to be assigned to entries of the person object class. Inherits cn and sn from the person object class.

This object class is defined in Internet White Pages Pilot.

Superior Class

person

OID

0.9.2342.19200300.100.4.4

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The person's common name.
sn (surname)	The person's surname, or last name.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which this person is engaged.
description	Text description of the person.
drink (favoriteDrink)	The person's favorite drink.
homePhone	The person's home phone number.
homePostalAddress	The person's home mailing address.
janetMailbox	The person's email address, intended for the convenience of UK users unfamiliar with rfc822 mail addresses.
mail	The person's email address.
mailPreferenceOption	Indicates a preference for inclusion of the person's name on mailing lists (electronic or physical). Not valid in Messaging Server 4.0.

mobile	The person's mobile phone number.
organizationalStatus	The person's type of employment (for example, full time).
otherMailbox	Values for electronic mailbox types other than X.400 and rfc822.
pager (pagerTelephoneNumber)	The person's pager number.
personalSignature	The person's signature file.
personalTitle	The person's personal title.
preferredDeliveryMethod	The person's preferred method of contact or delivery.
roomNumber	The person's room number.
secretary	Distinguished name of the person's secretary or administrative assistant.
seeAlso	DN to information relevant to the person.
telephoneNumber	The person's telephone number.
textEncodedORAddress	The person's text-encoded Originator/Recipient (X.400) address.
uid (userID)	Identifies the person's user id (usually the logon ID).
userClass	Category of user.
userPassword	Password with which the entry can bind to the directory.

nisMap

Definition

A generic abstraction of a NIS map.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.9

Required Attributes

Attribute	Description
nisMapName	The name of the NIS map.

Allowed Attributes

Attribute	Description
description	Text description of the NIS map.

nisNetgroup

Definition

An abstraction of a netgroup. May refer to other netgroups.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.8

Required Attributes

Attribute	Description
cn (commonName)	The common name of the netgroup.

Allowed Attributes

Attribute	Description
description	Text description of the netgroup.
nisNetgroupTriple	Defines a NIS netgroup with the syntax "hostname","username","domainname".
memberNisNetgroup	The name of the netgroup.

nisObject

Definition

Defines an entry in a NIS map.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.10

Required Attributes

Attribute	Description
cn (commonName)	The common name of the entry.
nisMapEntry	The NIS map entry ID.
nisMapName	The name of the NIS map.

Allowed Attributes

Attribute	Description
description	Text description of the locality.

nsComplexRoleDefinition

Definition

Any role that is not a simple role is, by definition, a complex role.

This object class is defined in Sun ONE Directory Server.

Superior Class

nsRoleDefinition

OID

2.16.840.1.113730.3.2.95

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
cn (commonName)	The entry's common name.
description	Text description of the entry.

nsFilteredRoleDefinition

Definition

Specifies assignment of entries to the role, depending upon the attributes contained by each entry.

This object class is defined in Sun ONE Directory Server.

Superior Class

nsComplexRoleDefinition

OID

2.16.840.1.113730.3.2.97

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
nsRoleFilter	Specifies the filter assigned to an entry.

Allowed Attributes

Attribute	Description
cn (commonName)	The entry's common name.
description	Text description of the entry.

nsLicenseUser

Definition

Used to track licenses for servers that are licensed on a per-client basis. `nsLicenseUser` is intended to be used with the `inetOrgPerson` object class. You can manage the contents of this object class through the Users and Groups area of the Administration Server.

This object class is defined in Sun ONE Administration Services.

Superior Class

top

OID

2.16.840.1.113730.3.2.7

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
nsLicensedFor	Specifies a license.
nsLicenseEndTime	Specifies an end time for a license.
nsLicenseStartTime	Specifies a start time for a license.

nsManagedRoleDefinition

Definition

Specifies assignment of a role to an explicit, enumerated list of members.

This object class is defined in Sun ONE Directory Server.

Superior Class

nsSimpleRoleDefinition

OID

2.16.840.1.113730.3.2.96

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
cn (commonName)	The entry's common name.
description	Text description of the entry.

nsNestedRoleDefinition

Definition

Specifies containment of one or more roles of any type within the role.

This object class is defined in Sun ONE Directory Server.

Superior Class

nsComplexRoleDefinition

OID

1.3.6.1.4.1.42.2.27.9.2.9

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
nsRoleDN	Specifies the roles assigned to an entry.

Allowed Attributes

Attribute	Description
cn (commonName)	The entry's common name.
description	Text description of the entry.
nsRoleScopeDN	Defines the scope of the role entry.

nsRoleDefinition

Definition

All role definition object classes inherit from the `nsRoleDefinition` object class.

This object class is defined in Sun ONE Directory Server.

Superior Class

ldapSubEntry

OID

2.16.840.1.113730.3.2.93

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
cn (commonName)	The entry's common name.
description	Text description of the entry.

nsSimpleRoleDefinition

Definition

Roles containing this object class are called simple roles because they have a deliberately limited flexibility, which makes it easy to:

- Enumerate the members of a role.
- Determine whether a given entry possesses a particular role.
- Enumerate all the roles possessed by a given entry.
- Assign a particular role to a given entry.
- Remove a particular role from a given entry.

This object class is defined in Sun ONE Directory Server.

Superior Class

nsRoleDefinition

OID

2.16.840.1.113730.3.2.94

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
cn (commonName)	The entry's common name.
description	Text description of the entry.

oncRpc

Definition

An abstraction of an Open Network Computing (ONC) Remote Procedure Call (RPC) binding. This class maps an ONC RPC number to a name. The distinguished value of the cn attribute denotes the RPC service's canonical name.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.5

Required Attributes

Attribute	Description
cn (commonName)	The entry's common name.
oncRpcNumber	The ONC RPC number.

Allowed Attributes

Attribute	Description
description	Text description of the entry.

organization

Definition

Used to define entries that represent organizations. An organization is generally assumed to be a large, relatively static grouping within a larger corporation or enterprise.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.4

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
o (organizationName)	The name of the organization.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which the organization is engaged.
description	Text description of the organization.
destinationIndicator	Country and city associated with the entry needed to provide Public Telegram Service.
fax (facsimileTelephoneNumber)	The organization's fax number.
internationaliSDNNumber	The organization's ISDN number.
l (localityName)	Place in which the organization is located.
physicalDeliveryOfficeName	Location where physical deliveries can be made to the organization.
postalAddress	The organization's mailing address.
postalCode	The postal code for this address (such as a United States zip code).

postOfficeBox	The organization's post office box.
preferredDeliveryMethod	The organization's preferred method of contact or delivery.
registeredAddress	Postal address suitable for reception of expedited documents, where the recipient must verify delivery.
searchGuide	Specifies information for suggested search criteria when using the entry as the base object in the directory tree for a search operation.
seeAlso	DN to information relevant to the organization.
st (stateOrProvinceName)	State or province in which the organization is located.
street (streetAddress)	Street address at which the organization is located.
telephoneNumber	The organization's telephone number.
telexNumber	Identifier for the organization's teletex terminal.
telexNumber	The organization's telex number.
userPassword	Password with which the entry can bind to the directory.
x121Address	X.121 address of the organization.

organizationalPerson

Definition

Used to define entries for people employed by or associated with an organization.

This object class is defined in RFC 2256.

Superior Class

person

OID

2.5.6.7

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The person's common name.
sn (surname)	The person's surname, or last name.

Allowed Attributes

Attribute	Description
description	Text description of the person.
destinationIndicator	Country and city associated with the person needed to provide Public Telegram Service.
fax (facsimileTelephoneNumber)	The person's fax number.
internationaliSDNNumber	The person's ISDN number.
l (localityName)	Place in which the person is located.
ou (organizationUnitName)	Organizational unit to which the person belongs.
physicalDeliveryOfficeName	Location where physical deliveries can be made to this person.
postalAddress	The person's mailing address.
postalCode	The postal code for this address (such as a United States zip code).

postOfficeBox	The person's post office box.
preferredDeliveryMethod	The person's preferred method of contact or delivery.
registeredAddress	Postal address suitable for reception of expedited documents, where the recipient must verify delivery.
seeAlso	DN to information relevant to the person.
st (stateOrProvinceName)	State or province in which the person is located.
street (streetAddress)	Street address at which the person is located.
telephoneNumber	The person's telephone number.
telexNumber	Identifier for the person's teletex terminal.
telexNumber	The person's telex number.
title	The person's job title.
userPassword	Password with which the entry can bind to the directory.
x121Address	X.121 address of the person.

organizationalRole

Definition

Used to define entries that represent roles held by people within an organization.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.8

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The role's common name.

Allowed Attributes

Attribute	Description
description	Text description of the role.
destinationIndicator	Country and city associated with the entry needed to provide Public Telegram Service.
fax (facsimileTelephoneNumber)	Fax number of the person in the role.
internationaliSDNNumber	ISDN number of the person in the role.
l (localityName)	Place in which the person in the role is located.
ou (organizationUnitName)	Organizational unit to which the person in the role belongs.
physicalDeliveryOfficeName	Location where physical deliveries can be made to the person in the role.
postalAddress	The mailing address for the person in the role.
postalCode	The postal code for this address (such as a United States zip code).
postOfficeBox	The post office box for the person in the role.

preferredDeliveryMethod	Preferred method of contact or delivery of the person in the role.
registeredAddress	Postal address suitable for reception of expedited documents, where the recipient must verify delivery.
roleOccupant	Distinguished name of the person in the role.
seeAlso	DN to information relevant to the person in the role.
st (stateOrProvinceName)	State or province in which the person in the role is located.
street (streetAddress)	Street address at which the person in the role is located.
telephoneNumber	Telephone number of the person in the role.
telexNumber	Identifier for the teletex terminal of the person in the role.
telexNumber	Telex number of the person in the role.
x121Address	X.121 address of the person in the role.

organizationalUnit

Definition

Used to define entries that represent organizational units. An organizational unit is generally assumed to be a relatively static grouping within a larger organization.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.5

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
ou (organizationUnitName)	The name of the organizational unit.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which the organizational unit is engaged.
description	Text description of the organizational unit.
destinationIndicator	Country and city associated with the organizational unit needed to provide Public Telegram Service.
fax (facsimileTelephoneNumber)	The organizational unit's fax number.
internationaliSDNNumber	The organizational unit's ISDN number.
l (localityName)	Place in which the organizational unit is located.
physicalDeliveryOfficeName	Location where physical deliveries can be made to the organizational unit.
postalAddress	The organizational unit's mailing address.
postalCode	The postal code for this address (such as a United States zip code).

postOfficeBox	The organizational unit's post office box.
preferredDeliveryMethod	The organizational unit's preferred method of contact or delivery.
registeredAddress	Postal address suitable for reception of expedited documents, where the recipient must verify delivery.
searchGuide	Specifies information for suggested search criteria when using the entry as the base object in the directory tree for a search operation.
seeAlso	DN to information relevant to the organizational unit.
st (stateOrProvinceName)	State or province in which the organizational unit is located.
street (streetAddress)	Street address at which the organizational unit is located.
telephoneNumber	The organizational unit's telephone number.
telexNumber	Identifier for the organizational unit's teletex terminal.
telexNumber	The organizational unit's telex number.
userPassword	Password with which the entry can bind to the directory.
x121Address	X.121 address of the organizational unit.

passwordPolicy

Definition

Defines a password policy entry.

This object class is defined in Sun ONE Directory Server.

Superior Class

top

OID

1.3.6.1.4.1.42.2.27.9.2.6

Required Attributes

Attribute	Description
cn (commonName)	The common name of the password policy.

Allowed Attributes

Attribute	Description
description	Text description of the password policy.
passwordChange	Indicates whether users may change their passwords.
passwordCheckSyntax	Indicates whether the password syntax will be checked before the password is saved.
passwordExp	Indicates whether user passwords will expire after a given number of seconds.
passwordExpireWithoutWarning	Indicates whether a password can expire regardless of whether the user was warned about the expiration date.
passwordInHistory	Indicates the number of passwords the Directory Server stores in history.
passwordLockout	Enables the account lockout mechanism.
passwordLockoutDuration	Specifies the length of time (in seconds) during which users will be locked out of the directory.
passwordMaxAge	Indicates the number of seconds after which user passwords will expire.

passwordMaxFailure	Specifies the number of consecutive failed bind attempts after which a user will be locked out of the directory.
passwordMinAge	Specifies the number of seconds that must elapse between password modifications.
passwordMinLength	Specifies the minimum number of characters that must be used in a password.
passwordMustChange	Indicates whether users must change their passwords when they first bind to the Directory Server, or when the password has been reset by the administrator.
passwordResetFailureCount	Specifies the length of time (in seconds) after which the password failure is reset to `0`.
passwordStorageScheme	Specifies the algorithm used to encrypt Directory Server passwords.
passwordUnlock	Specifies whether user accounts will be unlocked after a period of time.
passwordWarning	Specifies the number of seconds before a user's password expires that the user will receive a password expiration warning on attempting to authenticate to the directory.

person

Definition

Used to define entries that generically represent people. This object class is the base class for the organizationalPerson object class.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.6

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The person's common name.
sn (surname)	The person's surname, or last name.

Allowed Attributes

Attribute	Description
description	Text description of the person.
seeAlso	DN to information relevant to the person.
telephoneNumber	The person's telephone number.
userPassword	Password with which the entry can bind to the directory.

pilotObject

Definition

Used as a subclass to allow additional attributes to be assigned to entries of all other object classes.

This object class is defined in RFC 1274.

Superior Class

top

OID

0.9.2342.19200300.100.4.3

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
audio	Stores a sound file in binary format.
ditRedirect	Distinguished name to use as a redirect for the entry.
info	Information about the object.
jpegPhoto	Photo in jpeg format.
lastModifiedBy	Distinguished name of the last user to modify the object.
lastModifiedTime	Last time the object was modified.
manager	Distinguished name of the object's manager.
photo	Photo of the object.
uniqueIdentifier	Specific item used to distinguish between two entries when a distinguished name has been reused.

pilotOrganization

Definition

Used as a subclass to allow additional attributes to be assigned to organization and organizationalUnit object class entries.

This object class is defined in RFC 1274.

Superior Class

top

OID

0.9.2342.19200300.100.4.20

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
o (organizationName)	Organization to which the entry belongs.
ou (organizationUnitName)	Organizational unit to which the entry belongs.

Allowed Attributes

Attribute	Description
buildingName	Name of the building in which the entry is located.
businessCategory	Type of business in which the entry is engaged.
description	Text description of the entry.
destinationIndicator	Country and city associated with the pilot organization needed to provide Public Telegram Service.
fax (facsimileTelephoneNumber)	The pilot organization's fax number.
internationaliSDNNumber	The pilot organization's ISDN number.
l (localityName)	Place in which the pilot organization is located.
physicalDeliveryOfficeName	Location where physical deliveries can be made to the pilot organization.
postalAddress	The pilot organization's mailing address.

postalCode	The postal code for this address (such as a United States zip code).
postOfficeBox	The pilot organization's post office box.
preferredDeliveryMethod	The pilot organization's preferred method of contact or delivery.
registeredAddress	Postal address suitable for reception of expedited documents, where the recipient must verify delivery.
searchGuide	Specifies information for suggested search criteria when using the entry as the base object in the directory tree for a search operation.
seeAlso	DN to information relevant to the pilot organization.
st (stateOrProvinceName)	State or province in which the pilot organization is located.
street (streetAddress)	Street address at which the pilot organization is located.
telephoneNumber	The pilot organization's telephone number.
telexNumber	Identifier for the pilot organization's teletex terminal.
telexNumber	The pilot organization's telex number.
userPassword	Password with which the entry can bind to the directory.
x121Address	X.121 address of the pilot organization.

posixAccount

Definition

Auxiliary object class.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.0

Required Attributes

Attribute	Description
cn (commonName)	The common name of the account.
gidNumber	Group ID number.
homeDirectory	Home directory of the account.
uid (userID)	The userid of the account.
uidNumber	UNIX only. Related to the `/etc/shadow` file, this attribute specifies the login ID of the account.

Allowed Attributes

Attribute	Description
description	A human-readable description of the account.
gecos	The default GECOS.
loginShell	The path to the login shell.
userPassword	The entry's password and encryption method.

posixGroup

Definition
Structural object class.

This object class is defined in RFC 2307.

Superior Class
top

OID
1.3.6.1.1.1.2.2

Required Attributes

Attribute	Description
cn (commonName)	The common name of the group.
gidNumber	Group ID number.

Allowed Attributes

Attribute	Description
description	A human-readable description of the group.
memberUid	The member userid.
userPassword	The entry's password and encryption method.

referral

Definition

Used to represent a subordinate reference information in the directory. These referral objects hold one or more URIs contained in values of the `ref` attribute type and are used to generate protocol referrals and continuations.

This object class is defined in RFC 3296.

Superior Class

top

OID

2.16.840.1.113730.3.2.6

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
ref	The referral URI.

NOTE To use this object class, you must either make it a subclass, or use it with the `extensibleObject` object class. This ensures that you have an attribute for naming the entry.

residentialPerson

Definition

Used by the directory server to contain a person's residential information.

This object class is defined in RFC 2256.

Superior Class

person

OID

2.5.6.10

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	The person's common name.
l (localityName)	Place in which the person resides.
sn (surname)	The person's surname, or last name.

Allowed Attributes

Attribute	Description
businessCategory	Type of business in which the person is engaged.
description	Text description of the person.
destinationIndicator	Country and city associated with the entry needed to provide Public Telegram Service.
fax (facsimileTelephoneNumber)	The person's fax number.
internationaliSDNNumber	The person's ISDN number.
physicalDeliveryOfficeName	Location where physical deliveries can be made to the person.
postalAddress	The person's business mailing address.
postalCode	The postal code for this address (such as a United States zip code).

postOfficeBox	The person's business post office box.
preferredDeliveryMethod	The person's preferred method of contact or delivery.
registeredAddress	Postal address suitable for reception of expedited documents, where the recipient must verify delivery.
seeAlso	DN to information relevant to the person.
st (stateOrProvinceName)	State or province in which the person resides.
street (streetAddress)	Street address at which the person is located.
telephoneNumber	The person's telephone number.
telexNumber	Identifier for the person's teletex terminal.
telexNumber	The person's telex number.
userPassword	Password with which the entry can bind to the directory.
x121Address	X.121 address of the person.

RFC822LocalPart

Definition

Used to define entries that represent the local part of RFC822 mail addresses. The directory treats this part of an RFC822 address as a domain.

This object class is defined in Internet directory pilot.

Superior Class

domain

OID

0.9.2342.19200300.100.4.14

Allowed Attributes

Attribute	Description
cn (commonName)	The local part's common name.
sn (surname)	The entry's surname, or last name.

room

Definition

Used to store information in the directory about a room.

This object class is defined in RFC 1274.

Superior Class

top

OID

0.9.2342.19200300.100.4.7

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
cn (commonName)	Common name of the room.

Allowed Attributes

Attribute	Description
description	Text description of the room.
roomNumber	The room's number.
seeAlso	DN to information relevant to the room.
telephoneNumber	The room's telephone number.

shadowAccount

Definition

Auxiliary object class applicable to UNIX systems only. Related to the `/etc/shadow` file.

This object class is defined in RFC 2307.

Superior Class

top

OID

1.3.6.1.1.1.2.1

Required Attributes

Attribute	Description
uid (userID)	The entry's userid (usually the logon ID).

Allowed Attributes

Attribute	Description
description	Text description of the account.
shadowExpire	An absolute date specifying when the login may no longer be used.
shadowFlag	Reserved for future use.
shadowInactive	Number of days of inacitivity allowed for the specified user.
shadowLastChange	Number of days between January 1, 1970, and the date that the password was last modified.
shadowMax	Maximum number of days the password is valid.
shadowMin	Minimum number of days required between password changes.
shadowWarning	Number of days before the password expires that the user is warned.
userPassword	Password with which the entry can bind to the directory.

simpleSecurityObject

Definition

Used to allow an entry to contain the `userPassword` attribute when an entry's principal object classes do not allow `userPassword` as an attribute type. Reserved for future use.

This object class is defined in RFC 1274.

Superior Class

top

OID

0.9.2342.19200300.100.4.19

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
userPassword	Password with which the entry can bind to the directory.

strongAuthenticationUser

Definition

Auxiliary object class, used to store a user's certificate entry in the directory. This object class is used with other object classes, such as the `person` and `organization` object classes.

This object class is defined in RFC 2256.

Superior Class

top

OID

2.5.6.15

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.
userCertificate	Stores a user's certificate, usually in binary form.

subschema

Definition

Internal object class. An auxiliary object class subentry used to administer the subschema for the subschema administrative area. It holds the operational attributes representing the policy parameters used to express the subschema.

This object class is defined in RFC 2252.

Superior Class

top

OID

2.5.20.1

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Allowed Attributes

Attribute	Description
attributeTypes	Attribute types used within a subschema.
dITContentRules	Defines the DIT content rules in force within a subschema.
dITStructureRules	Defines the DIT structure rules in force within a subschema.
matchingRules	Defines the matching rules used within a subschema.
matchingRuleUse	Indicates the attribute types to which a matching rule applies in a subschema.
nameForms	Defines the name forms used in a subschema.
objectClasses	Defines the object classes used in a subschema.

top

Definition

Abstract object class, that defines the root of the object class hierarchy.

This object class is defined in RFC 2256.

Superior Class

N / A

OID

2.5.6.0

Required Attributes

Attribute	Description
objectClass	Defines the object classes for the entry.

Chapter 11

Attribute Reference

This chapter contains an alphabetic list of the standard attributes. It provides a definition of each attribute, the attribute syntax and the OID.

abstract

Definition
Provides an abstract of a document entry.

This attribute is defined in Internet White Pages Pilot.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.102.1.9

aliasedObjectName

Definition
Used by the directory server to identify alias entries in the directory. Contains the distinguished name of the entry for which it is an alias.

Note that alias dereferencing is not supported in Sun ONE Directory Server.

For example:

```
aliasedObjectName: cn=jdoe, dc=example, dc=com
```

This attribute is defined in RFC 2256.

Syntax
DN, single-valued.

OID
2.5.4.1

associatedDomain

Definition
Specifies a DNS domain associated with an object in the directory tree. For example, the entry in the directory tree with a distinguished name `c=US, o=example Corporation` might be associated to the domain `example.com`. Note that all domains should be represented in rfc822 order.

For example:

```
associatedDomain: example.com
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.37

associatedName

Definition
Specifies an entry in the organizational directory tree associated with a DNS domain.

For example:

```
associatedName: c=us
```

This attribute is defined in RFC 1274.

Syntax
DN, multi-valued.

OID
0.9.2342.19200300.100.1.38

audio

Definition
Contains a sound file in binary format. The attribute uses a u-law encoded sound file.

For example:

```
audio:: AAAAAA==
```

This attribute is defined in RFC 1274.

Syntax
Binary, multi-valued.

OID
0.9.2342.19200300.100.1.55

authorCn

Definition
Contains the common name of the author of a document entry.

For example:

```
authorCn: Kacey
```

This attribute is defined in Internet White Pages Pilot.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.102.1.11

authorSn

Definition
Contains the surname of the author of a document entry.

For example:

```
authorSn: Doe
```

This attribute is defined in Internet White Pages Pilot.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.102.1.12

authorityRevocationList

Definition
Contains a list of CA certificates that have been revoked. This attribute is to be stored and requested in the binary form, as `authorityRevocationList;binary`.

For example:

```
authorityRevocationList;binary:: AAAAAA==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.38

bootFile

Definition
The name of the boot image.

For example:

```
bootFile: mach
```

This attribute is defined in RFC 2307.

Syntax
String, multi-valued.

OID
1.3.6.1.1.1.1.24

bootParameter

Definition
Specified boot parameters.

For example:

```
bootParameter: root=fs:/nfsroot/peg
bootParameter: swap=fs:/nfsswap/peg
bootParameter: dump=fs:/nfsdump/peg
```

This attribute is defined in RFC 2307.

Syntax
bootParameterSyntax

OID
1.3.6.1.1.1.1.23

buildingName

Definition
Defines the building name associated with the entry.

For example:

```
buildingName: B14
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.48

businessCategory

Definition
Identifies the type of business in which the entry is engaged. This should be a broad generalization such as is made at the corporate division level.

For example:

```
businessCategory: Engineering
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.15

c (countryName)

Definition
Contains the two-character code representing country names, as defined in ISO-3166.

For example:

```
countryName: IE
```

or

```
c: IE
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, single-valued.

OID
2.5.4.6

CACertificate

Definition
Contains the CA's certificate. This attribute is to be stored and requested in the binary form, as `CACertificate;binary`.

For example:

```
CACertificate;binary:: AAAAAA==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.37

carLicense

Definition
Identifies the entry's automobile license plate number.

For example:

```
carLicense: 4MCS389
```

This attribute is defined in RFC 2798.

Syntax
DirectoryString, multi-valued.

OID
2.16.840.1.113730.3.1.1

certificateRevocationList

Definition
Contains a list of revoked user certificates. This attribute is to be stored and requested in the binary form, as `certificateRevocationList;binary`.

For example:

```
certificateRevocationList;binary:: AAAAAA==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.39

changes

Description
For add and modify operations, contains the changes made to the entry, in LDIF format.

This attribute is defined in Changelog Internet Draft.

Syntax
Binary, multi-valued.

OID
2.16.840.1.113730.3.1.8

changeLog

Description
The distinguished name of the entry that contains the set of entries comprising the server change log.

This attribute is defined in Changelog Internet Draft.

Syntax
DN, multi-valued.

OID
2.16.840.1.113730.3.1.35

changeNumber

Description
This single-valued attribute is always present. It contains an integer that uniquely identifies each change made to a directory entry. This number is related to the order in which the change occurred. The higher the number, the later the change.

This attribute is defined in the Changelog Internet Draft.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.5

changeTime

Description
Defines a time, in a YYMMDDHHMMSS format, when the entry was added.

This attribute is defined in the Changelog Internet Draft.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.77

changeType

Description
Specifies the type of LDAP operation. This attribute can have one of the following values: add, delete, modify, or modrdn.

For example:

```
changeType: modify
```

This attribute is defined in the Changelog Internet Draft.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.7

cn (commonName)

Definition
Identifies the name of an object in the directory. When the object corresponds to a person, the cn is typically the person's full name.

When identifying the entry's common name or full name:

```
commonName: Bill Anderson
```

or

```
cn: Bill Anderson
```

When in reference to LDAPReplica or LDAPServer object classes:

```
commonName: replicater.example.com:17430/dc%3Dexample%2Cdc%3Dcom
```

or

```
cn: replicater.example.com:17430/dc%3Dexample%2Cdc%3Dcom
```

This attribute is defined in RFC 2256.

Syntax

DirectoryString, multi-valued.

OID

2.5.4.3

co (friendlyCountryName)

Definition

Contains the name of a country. Often, the country attribute is used to describe a two-character code for a country, and the friendlyCountryName attribute is used to describe the actual country name.

For example:

```
friendlyCountryName: Ireland
```

or

```
co: Ireland
```

This attribute is defined in RFC 1274.

Syntax

DirectoryString, multi-valued.

OID

0.9.2342.19200300.100.1.43

cosAttribute

Description

Provides the name of the attribute for which you want to generate a value. You can specify more than one `cosAttribute` value. This attribute is used by all types of CoS definition entries.

The `cosAttribute` attribute allows two qualifiers following the name of the CoS attribute. The *override* qualifier has one of the following values:

- `default` (or no qualifier) - Indicates that the server does not override a real attribute value stored in the entry when it has the same type as the virtual attribute.
- `override` - Indicates that the server always returns the value generated by the CoS, even when there is a value stored with the entry.
- `operational` - Indicates that the attribute will only be returned if it is explicitly requested in the search. Operational attributes do not need to pass a schema check in order to be returned. It also has the same behavior as the override qualifier.

The merge qualifier is either absent or given with the following value:

- `merge-schemes` - Allows the virtual CoS attribute to be multivalued, either from multiple templates or multiple CoS definitions. For more information, see the section “Managing CoS From the Command Line” in Chapter 5 of the *Sun ONE Directory Server Administration Guide*.

This attribute is defined in Sun ONE Directory Server.

Syntax

Directory String, multi-valued.

OID

2.16.840.1.113730.3.1.550

cosIndirectSpecifier

Description

Specifies the attribute values used by an indirect CoS to identify the template entry.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.577

cosPriority

Definition
Specifies which template provides the attribute value, when CoS templates compete to provide an attribute value. This attribute represents the global priority of a particular template. A priority of zero is the highest priority.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.569

cosSpecifier

Description
Specifies the attribute value used by a classic CoS, which, along with the template entry's DN, identifies the template entry.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.551

cosTargetTree

Definition
Determines the subtree of the DIT to which the CoS schema applies. The values for this attribute for the schema and for multiple CoS schema may overlap their target trees in an arbitrary fashion.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.552

cosTemplateDn

Definition
Points to the entry that contains the CoS template.

This attribute is defined in Sun ONE Directory Server.

Syntax
Distinguished Name, single-valued.

OID
2.16.840.1.113730.3.1.553

crossCertificatePair

Definition
This attribute contains a pair of cross signed certificates. It is to be stored and requested in the binary form, as `crossCertificatePair;binary`.

For example:

```
crossCertificatePair;binary:: AAAAAA==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.40

dc (domainComponent)

Definition
Specifies one component of a domain name.

For example:

```
domainComponent: example
```

or

```
dc: example
```

This attribute is defined in RFC 2247.

Syntax
DirectoryString, single-valued.

OID
0.9.2342.19200300.100.1.25

deleteOldRdn

Description
In the case of `modrdn` operations, specifies whether the old RDN was deleted.

This attribute is defined in Changelog Internet Draft.

Syntax
Boolean, multi-valued.

OID
2.16.840.1.113730.3.1.10

deltaRevocationList

Definition
This attribute contains the *delta revocation list*, a list of newly revoked certificates. It is stored and requested in the binary form, as `deltaRevocationList;binary`.

For example:

```
deltaRevocationList;binary:: AAAAAA==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.53

departmentNumber

Definition
Identifies the entry's department number.

For example:

```
departmentNumber: 2604
```

This attribute is defined in RFC 2798.

Syntax
DirectoryString, multi-valued.

OID
2.16.840.1.113730.3.1.2

description

Definition
Provides a human-readable description of the object. For people and organizations this often includes their role or work assignment.

For example:

```
description: Quality control inspector for the ME2873 product line
```

This attribute is defined in RFC 2256.

Syntax

DirectoryString, multi-valued.

OID

2.5.4.13

destinationIndicator

Definition

The country and city associated with the entry needed to provide Public Telegram Service. Generally used in conjunction with `registeredAddress`.

For example:

```
destinationIndicator: Stow, Ohio, USA
```

This attribute is defined in RFC 2256.

Syntax

DirectoryString, multi-valued.

OID

2.5.4.27

displayName

Definition

Preferred name of a person to be used when displaying entries. Especially useful in displaying a preferred name for an entry within a one-line summary list. Since other attribute types, such as `cn`, are multi-valued, they cannot be used to display a preferred name.

For example:

```
displayName: Michigan Smith
```

This attribute is defined in RFC 2798.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.241

ditRedirect

Definition
Used to indicate that the object described by one entry now has a newer entry in the directory tree. This attribute may be used when an individual's place of work changes, and the individual acquires a new organizational DN.

For example:

```
ditRedirect: cn=jdoe, dc=example, dc=com
```

This attribute is defined in RFC 1274.

Syntax
DN

OID
0.9.2342.19200300.100.1.54

dmdName

Definition
The value of this attribute specifies a directory management domain (DMD), the administrative authority that operates the directory server.

For example:

```
dmdName: example.com
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.54

dn (distinguishedName)

Definition
Defines the distinguished name (dn) for the entry. Note that the dn is not always a mandatory attribute in an entry.

For example:

```
dn: cn=Jane Doe, ou=Quality Control, dc=example, dc=com
```

This attribute is defined in RFC 2256.

Syntax
DN

OID
2.5.4.49

dNSRecord

Definition
Specifies DNS resource records, including type A (Address), type MX (Mail Exchange), type NS (Name Server), and type SOA (Start Of Authority) resource records.

For example:

```
dNSRecord: IN NS ns.uu.net
```

This attribute is defined in Internet directory pilot.

Syntax
IA5String, multi-valued.

OID
0.9.2342.19200300.100.1.26

documentAuthor

Definition
Contains the distinguished name of the author of a document entry.

For example:

```
documentAuthor: cn=John Doe, dc=example, dc=com
```

This attribute is defined in RFC 1274.

Syntax
DN, multi-valued.

OID
0.9.2342.19200300.100.1.14

documentIdentifier

Definition
Specifies a unique identifier for a document.

For example:

```
documentIdentifier: L3204REV1
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.11

documentLocation

Definition
Defines the location of the original copy of a document entry.

For example:

```
documentLocation: Department Library
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.15

documentPublisher

Definition
The person and/or organization that published a document.

For example:

```
documentPublisher: Southeastern Publishing
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, single-valued.

OID
0.9.2342.19200300.100.1.56

documentStore

Definition
Defines the place in which a document is stored. This attribute is defined in the Internet White Pages Pilot.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.102.1.10

documentTitle

Definition
Contains the title of a document entry.

For example:

```
documentTitle: Directory Administrator's Guide
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.12

documentVersion

Definition
Defines the version of a document entry.

For example:

```
documentVersion: 1.1
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.13

drink (favoriteDrink)

Definition
Describes the favorite drink of a person entry.

For example:

```
drink: gin
```

or

```
favoriteDrink: gin
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.5

dSAQuality

Definition
Specifies the purported quality of a DSA. This attribute allows a DSA manager to indicate the expected level of availability of the DSA.

For example:

```
dSAQuality: high
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, single-valued.

OID
0.9.2342.19200300.100.1.49

employeeNumber

Definition
Identifies the entry's employee number.

For example:

```
employeeNumber: 3440
```

This attribute is defined in RFC 2798.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.3

employeeType

Definition
Identifies the entry's type of employment.

For example:

```
employeeType: Full time
```

This attribute is defined in RFC 2798.

Syntax
DirectoryString, multi-valued.

OID
2.16.840.1.113730.3.1.4

enhancedSearchGuide

Definition
Used by X.500 clients when constructing search filters.

For example:

```
enhancedSearchGuide: (uid=mhughes)
```

This attribute is defined in RFC 2798.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.47

fax (facsimileTelephoneNumber)

Definition
Identifies the fax number at which the entry can be reached. Abbreviation: fax.

For example:

```
facsimileTelephoneNumber: 415-555-1212
```

or:

```
fax: 415-555-1212
```

This attribute is defined in RFC 2256.

Syntax
TelephoneNumber, multi-valued.

OID
2.5.4.23

gecos

Definition
The default GECOS.

This attribute is defined in RFC 2307.

Syntax
String, single-valued.

OID
1.3.6.1.1.1.1.2

generationQualifier

Definition
Contains the generation Qualifier part of the name, typically appearing in the suffix.

For example:

```
generationQualifier: Jr
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.44

gidNumber

Definition
Group ID number.

For example:

```
gidNumber: 162035
```

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.1

givenName

Definition
Identifies the entry's given name, usually a person's first name.

For example:

```
givenName: Hecuba
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.42

homeDirectory

Definition
The home directory of the account.

For example:

```
homeDirectory: /home/bsmith
```

This attribute is defined in RFC 2307.

Syntax
String, single-valued.

OID
1.3.6.1.1.1.1.3

homePhone

Definition
Identifies the entry's home phone number.

For example:

```
homePhone: 415-555-1212
```

This attribute is defined in RFC 1274.

Syntax
TelephoneNumber, multi-valued.

OID
0.9.2342.19200300.100.1.20

homePostalAddress

Definition
Identifies the entry's home mailing address. This field is intended to include multiple lines, but each line within the entry should be separated by a dollar sign ($). To represent an actual dollar sign ($) or backslash (\) within this text, use the escaped hex values \24 and \5c respectively.

To identify an entry's home mailing address:

```
homePostalAddress: 1234 Ridgeway Drive$Santa Clara, CA$99555
```

Additionally, to represent the string:

```
The dollar ($) value can be found
in the c:\cost file.
```

provide the string:

```
The dollar (\24) value can be found$in the c:\5ccost file.
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.39

host

Definition
Defines the hostname of a computer.

For example:

```
host: myServer
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.9

houseIdentifier

Definition
Identifies a building in a location.

For example:

```
houseIdentifier: B105
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.51

info

Definition
Specifies any general information pertinent to an object. It is recommended that specific usage of this attribute type is avoided, and that specific requirements are met by other (possibly additional) attribute types.

For example:

```
info: not valid
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.4

initials

Definition
Identifies the entry's initials. Does not identify the entry's surname.

For example:

```
initials: BFA
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.43

internationaliSDNNumber

Definition
Contains the ISDN number of the entry. This is in the internationally agreed format for ISDN addresses given in CCITT Rec. E. 164.

For example:

```
internationaliSDNNumber: +SO 812467
```

This attribute is defined in RFC 2256.

Syntax
IA5String, multi-valued.

OID
2.5.4.25

ipHostNumber

Definition
IP address, expressed as a dotted decimal, omitting leading zeros.

For example:

```
ipHostNumber: 10.0.0.1
```

This attribute is defined in RFC 2307.

Syntax
IA5String{128}

OID
1.3.6.1.1.1.1.19

ipNetmaskNumber

Definition
IP netmask, expressed as a dotted decimal, omitting leading zeros.

For example:

```
ipNetmaskNumber: 255.255.255.0
```

This attribute is defined in RFC 2307.

Syntax
IA5String{128}, single-valued.

OID
1.3.6.1.1.1.1.21

ipNetworkNumber

Definition
IP network, expressed as a dotted decimal, omitting leading zeros.

For example:

```
ipNetworkNumber: 192.168
```

This attribute is defined in RFC 2307.

Syntax
IA5String{128}, single-valued.

OID
1.3.6.1.1.1.1.20

ipProtocolNumber

Definition
The IP protocol number. This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.17

ipServicePort

Definition
The IP service port number. This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.15

ipServiceProtocol

Definition
The IP service protocol.

For example:

```
ipServiceProtocol: tcp
ipServiceProtocol: udp
```

This attribute is defined in RFC 2307.

Syntax
String, multi-valued.

OID
1.3.6.1.1.1.1.16

janetMailbox

Definition
Specifies an email address. This attribute is intended for the convenience of UK users unfamiliar with rfc822 mail addresses. Entries using this attribute must also include an `rfc822Mailbox` attribute.

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.46

javaClassName

Definition
Stores the fully qualified name of the Java object's distinguished class or interface.

For example:

```
javaClassName: java.lang.String
```

This attribute is defined in RFC 2713.

Syntax
Directory String, single-valued.

OID
1.3.6.1.4.1.42.2.27.4.1.6

javaClassNames

Definition
Stores the Java object's fully qualified class or interface names. It is a multivalued attribute. When more than one value is present, each is the name of a class or interface, or ancestor class or interface, of this object.

This attribute is defined in RFC 2713.

Syntax
Directory String, multi-valued.

OID
1.3.6.1.4.1.42.2.27.4.1.13

javaCodebase

Definition
Stores the Java class definition's locations. It specifies the locations from which to load the class definition for the class specified by the `javaClassName` attribute. If this attribute contains more than one value, each value is an independent codebase.

This attribute is defined in RFC 2713.

Syntax
IA5String, multi-valued.

OID
1.3.6.1.4.1.42.2.27.4.1.7

javaDoc

Definition
This attribute stores a pointer to the Java documentation for the class. Its value is a URL.

For example:

```
javaDoc:
http://java.sun.com/products/jdk/1.2/docs/api/java/lang/String.html
```

This attribute is defined in RFC 2713.

Syntax
IA5String, multi-valued.

OID
1.3.6.1.4.1.42.2.27.4.1.12

javaFactory

Definition
Stores the fully qualified class name of the object factory that can be used to create an instance of the object identified by the `javaClassName` attribute.

For example:

```
javaFactory: com.example.jndi.ExampleObjectFactory
```

This attribute is defined in RFC 2713.

Syntax
String, multi-valued.

OID
1.3.6.1.4.1.42.2.27.4.1.10

javaReferenceAddress

Definition
Represents the sequence of addresses of a JNDI reference. Each of its values represents one address, a Java object of type `javax.naming.RefAddr`. Its value is a concatenation of the address type and address contents, preceded by a sequence number.

For example:

```
ipServiceProtocol:   #0#TypeA#ValA
                     #1#TypeB#ValB
                     #2#TypeC##rO0ABXNyABpq
```

This attribute is defined in RFC 2713.

Syntax
Directory String, multi-valued.

OID
1.3.6.1.4.1.42.2.27.4.1.11

javaSerializedData

Definition
Stores the serialized form of a Java object.

This attribute is defined in RFC 2713.

Syntax
Octet String, single-valued.

OID
1.3.6.1.4.1.42.2.27.4.1.8

jpegPhoto

Definition
Contains a JPEG photo of the entry.

For example:

```
jpegPhoto:: AAAAAA==
```

This attribute is defined in RFC 2798.

Syntax
Binary, multi-valued.

OID
0.9.2342.19200300.100.1.60

keyWords

Definition
Contains keywords for the entry.

For example:

```
keyWords: directory LDAP X.500
```

This attribute is defined in Internet White Pages Pilot.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.102.1.7

knowledgeInformation

Definition
This attribute is no longer used.

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.2

l (localityName)

Definition
Identifies the county, city, or other geographical area in which the entry is located or with which it is in some other way associated.

For example:

```
localityName: Santa Clara
```

or

```
l: Santa Clara
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.7

labeledURI

Definition
Specifies a Uniform Resource Identifier (URI) that is relevant in some way to the entry. Values placed in the attribute should consist of a URI (currently only URLs are supported) optionally followed by one or more space characters and a label.

For example:

```
labeledURI: http://home.sun.com
```

```
labeledURI: http://home.sun.com Sun website
```

This attribute is defined in RFC 2079.

Syntax
IA5String, multi-valued.

OID
1.3.6.1.4.1.250.1.57

lastModifiedBy

Definition
Specifies the distinguished name of the last user to modify the associated entry.

For example:

```
lastModifiedBy: cn=Jane Doe,ou=Quality Control,dc=example,dc=com
```

This attribute is defined in RFC 1274.

Syntax
DN, single-valued.

OID
0.9.2342.19200300.100.1.24

lastModifiedTime

Definition
Defines the last time, in UTC format, that a change was made to the entry.

For example:

```
lastModifiedTime: Thursday, 22-Sep-93 14:15:00 GMT
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, single-valued.

OID
0.9.2342.19200300.100.1.23

loginShell

Definition
The path to the login shell.

For example:

```
loginShell: /bin/csh
```

This attribute is defined in RFC 2307.

Syntax
IA5String, single-valued.

OID
1.3.6.1.1.1.1.4

macAddress

Definition
The MAC address in maximal, colon separated hex notation, eg. 00:00:92:90:ee:e2.

For example:

```
macAddress: 00:00:92:90:ee:e2
```

This attribute is defined in RFC 2307.

Syntax
String, multi-valued.

OID
1.3.6.1.1.1.1.22

mail

Definition
Identifies a user's primary email address (the email address retrieved and displayed by "white-pages" lookup applications).

For example:

```
mail: banderson@example.com
```

This attribute is defined in RFC 1274.

Syntax

DirectoryString, single-valued.

OID

0.9.2342.19200300.100.1.3

mailPreferenceOption

Definition

Not used in Messaging Server 4.0.

Indicates a preference for the inclusion of user names on mailing lists (electronic or physical). Accepted values include:

- 0: user doesn't want to be included in mailing lists.
- 1: user consents to be added to any mailing list.
- 2: user only wants to be added to mailing lists that the list provider views as relevant to the user's professional interests.

The absence of this attribute for a person should be interpreted as if the attribute were present with the value `no-list-inclusion`. This attribute should be interpreted by anyone using the directory to derive mailing lists, and its value respected.

For example:

```
mailPreferenceOption:0
```

This attribute is defined in RFC 1274.

Syntax

Integer, single-valued.

OID

0.9.2342.19200300.100.1.47

manager

Definition
Identifies the distinguished name of the entry's manager.

For example:

```
manager:cn=Jane Doe, ou=Quality Control, dc=example, dc=com
```

This attribute is defined in RFC 1274.

Syntax
DN, multi-valued.

OID
0.9.2342.19200300.100.1.10

member

Definition
Identifies the distinguished names for each member of the group.

For example:

```
member: cn=John Doe, dc=example, dc=com
```

This attribute is defined in RFC 2256.

Syntax
DN, multi-valued.

OID
2.5.4.31

memberCertificateDescription

Definition

A multi-valued attribute, for which each value is a description, a pattern, or a filter matching the subject DN of a certificate (usually certificates used for SSL client authentication).

memberCertificateDescription matches any certificate that contains a subject DN with the same AVAs as the description. The description may contain multiple "ou=" AVAs. A matching DN must contain those same "ou=" AVAs, in the same order, although it may contain other AVAs (including other "ou=" AVAs) interspersed. For any other attribute type (not ou), there should be at most one AVA of that type in the description. If there are several, all but the last are ignored.

A matching DN must contain that same AVA, but no other AVA of the same type nearer the root (later, syntactically).

AVAs are considered the same if they contain the same attribute description (case-insensitive comparison) and the same attribute value (case-insensitive comparison, leading and trailing whitespace ignored, and consecutive whitespace characters treated as a single SP).

In order to be considered a member of a group with the following memberCertificateDescription, a certificate would need to include ou=x, ou=A, and o=example, but not o=company.

```
memberCertificateDescription: {ou=x, ou=A, o=company, o=example}
```

In order to match the group's requirements, a certificate's subject DNs must contain the same ou attribute types in the same order as defined in the memberCertificateDescription attribute.

This attribute is defined in Sun ONE Directory Server.

Syntax

IA5String, multi-valued.

OID

2.16.840.1.113730.3.1.199

memberNisNetgroup

Definition
The name of a netgroup. This attribute is defined in RFC 2307.

Syntax
IA5String, multi-valued.

OID
1.3.6.1.1.1.1.13

memberUid

Definition
The user id of the member. This attribute is defined in RFC 2307.

Syntax
IA5String, multi-valued.

OID
1.3.6.1.1.1.1.12

memberURL

Definition
Identifies a URL associated with each member of a group. Any type of labeled URL can be used.

For example:

```
memberURL: ldap:///cn=jdoe,dc=example,dc=com
```

This attribute is defined in Sun ONE Directory Server.

Syntax
IA5String, multi-valued.

OID
2.16.840.1.113730.3.1.198

mobile

Definition
Identifies the entry's mobile or cellular phone number. Abbreviation: mobile.

For example:

```
mobileTelephoneNumber: 415-555-4321
```

```
mobile: 415-555-4321
```

This attribute is defined in RFC 1274.

Syntax
TelephoneNumber, multi-valued.

OID
0.9.2342.19200300.100.1.41

multiLineDescription

Definition
Provides descriptive text for a mail user. When represented in LDIF format, each line should be separated by a dollar sign ($). The Directory Server expects 0 or 1 occurrences of this attribute per mail account.

For example:

```
multiLineDescription: Account Administrator and$directory manager.
```

To represent an actual dollar sign ($) or backslash (\) within this text, use the escaped hex values \24 and \5c respectively. For example, to represent the string:

```
The dollar ($) value can be found in the c:\cost file.
```

provide the string:

```
The dollar (\24) value can be found$in the c:\5ccost file.
```

This attribute is defined in Internet White Pages Pilot.

Syntax
DirectoryString, multi-valued.

OID
1.3.6.1.4.1.250.1.2

name

Definition

Identifies the attribute supertype from which string attribute types used for naming may be formed. It is unlikely that values of this type will occur in an entry. LDAP server implementations that do not support attribute subtyping do not need to recognize this attribute in requests. Client implementations should not assume that LDAP servers are capable of performing attribute subtyping.

This attribute is defined in RFC 2256.

Syntax

DirectoryString, multi-valued.

OID

2.5.4.41

newRdn

Description

In the case of `modrdn` operations, specifies the new RDN of the entry.

This attribute is defined in Changelog Internet Draft.

Syntax

DN, single-valued.

OID

2.16.840.1.113730.3.1.9

newSuperior

Description
In the case of `modrdn` operations, specifies the `newSuperior` attribute of the entry.

This attribute is defined in Changelog Internet Draft.

Syntax
DN, single-valued.

OID
2.16.840.1.113730.3.1.11

nisMapEntry

Definition
The NIS map entry ID.

This attribute is defined in RFC 2307.

Syntax
IA5String{1024}, single-valued

OID
1.3.6.1.1.1.1.27

nisMapName

Definition
The name of the NIS map. This attribute is defined in RFC 2307.

Syntax
String, multi-valued.

OID
1.3.6.1.1.1.1.26

nisNetgroupTriple

Definition
Defines a NIS netgroup with the syntax "hostname","username","domainname".

For example:

```
nisNetgroupTriple: (myserver,jsmith,example.com)
```

This attribute is defined in RFC 2307.

Syntax
nisNetgroupTripleSyntax

OID
1.3.6.1.1.1.1.14

nsLicensedFor

Definition
Identifies the server the user is licensed to use. The Administration Server expects each `nsLicenseUser` entry to contain zero or more instances of this attribute. Valid keywords for this attribute are currently:

- mail: the user is a licensed client of the Messaging Server.
- new: the user is a licensed client of the Collabra Server.
- slapd: the user is a licensed client of the Directory Server.
- cal: the user is a licensed client of the Calendar Server.

For example:

```
nsLicensedFor: slapd
```

This attribute is defined in Sun ONE Administration Services.

Syntax
DirectoryString, multi-valued.

OID
2.16.840.1.113730.3.1.36

nsRoleScopeDn

Definition
Determines the scope of a role entry. If this attribute is not present, the scope of the role is defined by the LDAPsubentry. Otherwise, the scope is the union of the scope defined by the LDAPsubentry and the scope defined in this attribute.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
1.3.6.1.4.1.1466.115.121.1.12

o (organizationName)

Definition
Identifies the name of the organization.

For example:

```
organizationName: example, Inc.
```

or

```
o: example, Inc
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.10

objectClass

Definition
Specifies the object classes of the object. Must include the object.

For example:

```
objectClass: person
```

This attribute is defined in RFC 2256.

Syntax
IA5String, multi-valued.

OID
2.5.4.0

obsoletedByDocument

Definition
Contains the distinguished name of a document that obsoletes the document entry.

For example:

```
obsoletedbyDocument: cn=Document Version 2, ou=Document Library,
 dc=example, dc=com
```

This attribute is defined in Internet White Pages Pilot.

Syntax
DN, multi-valued.

OID
0.9.2342.19200300.102.1.4

obsoletesDocument

Definition
Contains the distinguished name of a document that is obsoleted by the document entry.

For example:

```
obsoletesDocument: cn=Document Version 1, ou=Document Library,
 dc=example, dc=com
```

This attribute is defined in Internet White Pages Pilot.

Syntax
DN, multi-valued.

OID
0.9.2342.19200300.102.1.3

oncRpcNumber

Definition
The Open Network Computing (ONC) Remote Procedure Call (RPC) number.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.18

organizationalStatus

Definition
Specifies a category by which a person is often referred to in an organization.

For example:

```
organizationalStatus: researcher
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.45

otherMailbox

Definition
Specifies values for electronic mailbox types other than X.400 and rfc822.

For example:

```
otherMailbox: Telemail: x378: Joe
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.22

ou (organizationUnitName)

Definition
Identifies the name of an organizational unit.

For example:

```
organizationUnitName: Marketing
```

or

```
ou: Marketing
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.11

owner

Definition
Identifies the distinguished name of the person responsible for the entry.

For example:

```
owner: cn=Babs Jensen, dc=example, dc=com
```

This attribute is defined in RFC 2256.

Syntax
DN, multi-valued.

OID
2.5.4.32

pager (pagerTelephoneNumber)

Definition
Identifies the entry's pager phone number.

For example:

```
pagerTelephoneNumber: 415-555-6789
```

or

```
pager: 415-555-6789
```

This attribute is defined in RFC 1274.

Syntax
TelephoneNumber, multi-valued.

OID
0.9.2342.19200300.100.1.42

passwordChange

Definition
Indicates whether users may change their passwords.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.102

passwordCheckSyntax

Definition
Indicates whether the password syntax will be checked before the password is saved.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.103

passwordExp

Definition
Indicates whether user passwords will expire after a specified number of seconds.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.98

passwordInHistory

Definition
Indicates the number of passwords the Directory Server stores in history.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.101

passwordLockout

Definition
Enables the account lockout mechanism.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.105

passwordLockoutDuration

Definition
Specifies the length of time (in seconds) during which users will be locked out of the directory.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.109

passwordMaxAge

Definition
Indicates the number of seconds after which user passwords will expire.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.97

passwordMaxFailure

Definition
Specifies the number of consecutive failed bind attempts after which a user will be locked out of the directory.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.106

passwordMinAge

Definition
Specifies the number of seconds that must elapse between password modifications.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.222

passwordMinLength

Definition
Specifies the minimum number of characters that must be used in a password.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.99

passwordMustChange

Definition
Indicates whether users must change their passwords when they first bind to the Directory Server, or when the password has been reset by the administrator.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.220

passwordResetFailureCount

Definition
Specifies the length of time (in seconds) after which the password failure is reset to `0`.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.223

passwordStorageScheme

Definition
Specifies the algorithm used to encrypt Directory Server passwords.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.221

passwordUnlock

Definition
Specifies whether user accounts will be unlocked after a period of time.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.108

passwordWarning

Definition
Specifies the number of seconds before a user's password expires that the user will receive a password expiration warning on attempting to authenticate to the directory.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.104

personalSignature

Definition
A signature file, in binary format, for the entry.

For example:

```
personalSignature:: AAAAAA==
```

This attribute is defined in RFC 1274.

Syntax
Binary, multi-valued.

OID
0.9.2342.19200300.100.1.53

personalTitle

Definition
Specifies a personal title for a person. Examples of personal titles are “Ms,” “Dr,” “Prof,” and “Rev.”

For example:

```
personalTitle: Mr
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.40

photo

Definition
Contains a photo, in binary form, of the entry.

For example:

```
photo:: AAAAAA==
```

This attribute is defined in RFC 1274.

Syntax
Binary, multi-valued.

OID
0.9.2342.19200300.100.1.7

physicalDeliveryOfficeName

Definition
Identifies the name of the city or village in which a physical delivery office is located.

For example:

```
physicalDeliveryOfficeName: Santa Clara
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.19

postalAddress

Definition
Identifies the entry's mailing address. This field is intended to include multiple lines. When represented in LDIF format, each line should be separated by a dollar sign ($).

For example:

```
postalAddress: P.O. Box 3541$Santa Clara, CA$99555
```

To represent an actual dollar sign ($) or backslash (\) within the text, use the escaped hex values \24 and \5c respectively.

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.16

postalCode

Definition
Identifies the entry's zip code in the United States.

For example:

```
postalCode: 44224
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.17

postOfficeBox

Definition
Specifies a postal mailing address.

For example:

```
postOfficeBox: P.O. Box 1234
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.18

preferredDeliveryMethod

Definition
Identifies the entry's preferred contact or delivery method.

For example:

```
preferredDeliveryMethod: telephone
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, single-valued.

OID
2.5.4.28

preferredLanguage

Definition
Defines a person's preferred written or spoken language. The value for this attribute should conform to the syntax for HTTP Accept-Language header values.

For example:

```
preferredLanguage: en-us
```

This attribute is defined in RFC 2798.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.39

presentationAddress

Definition
Contains an OSI presentation address for the entry. The presentation address consists of an OSI Network Address and up to three selectors, one each for use by the transport, session, and presentation entities.

For example:

```
presentationAddress: TELEX+00726322+RFC-1006+02+130.59.2.1
```

This attribute is defined in RFC 2256.

Syntax
IA5String, single-valued.

OID
2.5.4.29

protocolInformation

Definition
Used in conjunction with the presentationAddress attribute to provide additional information to the OSI network service.

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.48

ref

Description
Used in LDAPv3 to support smart referrals. Contains an LDAP URL in the format:

```
ldap://<servername>:<portnumber>/<dn>
```

The port number is optional.

For example:

```
ref: ldap://server.example.com:389/ou=People, o=example.com
```

Note that DN special characters must be escaped. For example:

```
ref: ldap://server.example.com:389/ou=People, o=example%Inc
```

This attribute is defined in RFC 3296.

Syntax
IA5String, multi-valued.

OID
2.16.840.1.113730.3.1.34

registeredAddress

Definition
This attribute contains a postal address for receiving telegrams or expedited documents. The recipient's signature is usually required on delivery.

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.26

roleOccupant

Definition
Contains the distinguished name of the person acting in the role defined in the `organizationalRole` entry.

For example:

```
roleOccupant: uid=jdoe, dc=example, dc=com
```

This attribute is defined in RFC 2256.

Syntax
DN, multi-valued.

OID
2.5.4.33

roomNumber

Definition
Specifies the room number of an object. Note that the commonName attribute should be used for naming room objects.

For example:

```
roomNumber: 230
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.6

searchGuide

Definition
Specifies information for a suggested search criteria when using the entry as the base object in the directory tree for a search operation. When constructing search filters, use `enhancedSearchGuide` instead.

This attribute is defined in RFC 2256.

Syntax
IA5String, multi-valued.

OID
2.5.4.14

secretary

Definition
Identifies the entry's secretary or administrative assistant.

For example:

```
secretary: cn=John Doe, dc=example, dc=com
```

This attribute is defined in RFC 1274.

Syntax
DN, multi-valued.

OID
0.9.2342.19200300.100.1.21

seeAlso

Definition
Identifies another directory server entry that may contain information related to this entry.

For example:

```
seeAlso: cn=Quality Control Inspectors,ou=manufacturing,
 dc=example, dc=com
```

This attribute is defined in RFC 2256.

Syntax
DN, multi-valued.

OID
2.5.4.34

serialNumber

Definition
Specifies the serial number of a device.

For example:

```
serialNumber: 555-1234-AZ
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.5

shadowExpire

Definition
UNIX systems only. Related to the `/etc/shadow` file, this attribute contains an absolute date specifying when the login may no longer be used.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.10

shadowFlag

Definition
UNIX systems only. Related to the `/etc/shadow` file, this attribute is currently not used and is reserved for future use.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.11

shadowInactive

Definition
UNIX systems only. Related to the `/etc/shadow` file, this attribute specifies the number of days of inacitivity allowed for the specified user.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.9

shadowLastChange

Definition
UNIX systems only. Related to the `/etc/shadow` file, this attribute specifies number of days between January 1, 1970, and the date that the password was last modified.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.5

shadowMax

Definition
UNIX systems only. Related to the `/etc/shadow` file, this attribute specifies the maximum number of days the password is valid.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.7

shadowMin

Definition
UNIX systems only. Related to the `/etc/shadow` file, this attribute specifies the minimum number of days required between password changes.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.6

shadowWarning

Definition
UNIX systems only. Related to the `/etc/shadow` file, this attribute specifies the number of days before the password expires that the user is warned.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.8

singleLevelQuality

Definition
Specifies the purported data quality at the level immediately below in the DIT.

This attribute is defined in RFC 1274.

Syntax
DirectoryString, single-valued.

OID
0.9.2342.19200300.100.1.50

sn (surname)

Definition
Identifies the entry's surname, also referred to as last name or family name.

For example:

```
surname: Anderson
```

or

```
sn: Anderson
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.4

st (stateOrProvinceName)

Definition
Identifies the state or province in which the entry resides. Abbreviation: st.

For example:

```
stateOrProvinceName: California
```

or

```
st: California
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.8

street (streetAddress)

Definition
Identifies the entry's house number and street name.

For example:

```
streetAddress: 1234 Ridgeway Drive
```

or

```
street: 1234 Ridgeway Drive
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.9

subject

Definition
Contains information about the subject matter of the document entry.

For example:

```
subject: employee option grants
```

This attribute is defined in Internet White Pages Pilot.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.102.1.8

subtreeMaximumQuality

Definition
Specifies the purported maximum data quality for a DIT subtree.

This attribute is defined in RFC 1274.

Syntax
DirectoryString, single-valued.

OID
0.9.2342.19200300.100.1.52

subtreeMinimumQuality

Definition
Specifies the purported minimum data quality for a DIT subtree.

This attribute is defined in RFC 1274.

Syntax
DirectoryString, single-valued.

OID
0.9.2342.19200300.100.1.51

supportedAlgorithms

Definition
This attribute is to be stored and requested in the binary form, as `supportedAlgorithms;binary`.

For example:

```
supportedAlgorithms;binary: AAAAAA==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.52

supportedApplicationContext

Definition
This attribute contains the identifiers of OSI application contexts.

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.30

targetDn

Description
Contains the DN of the entry that was affected by the LDAP operation. In the case of a `modrdn` operation, the `targetDn` attribute contains the DN of the entry before it was modified or moved.

This attribute is defined in Changelog Internet Draft.

Syntax
DN, multi-valued.

OID
2.16.840.1.113730.3.1.6

telephoneNumber

Definition
Identifies the entry's phone number.

For example:

```
telephoneNumber: 415-555-2233
```

This attribute is defined in RFC 2256.

Syntax
TelephoneNumber, multi-valued.

OID
2.5.4.20

telexNumber

Definition
Defines the telex number of the entry. The format of the telex number is as follows:

```
actual-number "$" country "$" answerback
```

where:

- actual-number: the syntactic representation of the number portion of the TELEX number being encoded.
- country: the TELEX country code.
- answerback: the answerback code of a TELEX terminal.

This attribute is defined in RFC 2256.

Syntax

DirectoryString, multi-valued.

OID

2.5.4.21

textEncodedORAddress

Definition

Defines the text-encoded Originator / Recipient (X.400) address of the entry as defined in RFC987.

For example:

```
textEncodedORAddress: /S=doe/OU=eng/O=example/ADMD=telemail/C=us/
```

This attribute is defined in RFC 1274.

Syntax

DirectoryString, multi-valued.

OID

0.9.2342.19200300.100.1.2

title

Definition

Identifies the title of a person in the organization.

For example:

```
title: Senior QC Inspector
```

This attribute is defined in RFC 2256.

Syntax
DirectoryString, multi-valued.

OID
2.5.4.12

uid (userID)

Definition
Identifies the entry's userid (usually the logon ID). Abbreviation: uid.

For example:

```
userid: banderson
```

or

```
uid: banderson
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.1

uidNumber

Definition
UNIX only. Related to the `/etc/shadow` file, this attribute specifies the user's login ID.

This attribute is defined in RFC 2307.

Syntax
Integer, single-valued.

OID
1.3.6.1.1.1.1.0

uniqueIdentifier

Definition
Identifies a specific item used to distinguish between two entries when a distinguished name has been reused. This attribute is intended to detect an instance of a reference to a distinguished name that has been deleted. This attribute is assigned by the server.

For example:

```
uniqueIdentifier: 17B
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.44

uniqueMember

Definition
Identifies a group of names associated with an entry where each name was given a uniqueIdentifier to ensure its uniqueness. A value for the uniqueMember attribute is a DN followed by the uniqueIdentifier.

For example:

```
uniqueMember: cn=John Doe, dc=example, dc=com 17
```

This attribute is defined in RFC 2256.

Syntax
DN, multi-valued.

OID
2.5.4.50

updatedByDocument

Definition

Contains the distinguished name of a document that is an updated version of the document entry.

For example:

```
updatedByDocument: cn=Document Version 2, ou=Document Library,
 dc=example, dc=com
```

This attribute is defined in Internet White Pages Pilot.

Syntax

DN, multi-valued.

OID

0.9.2342.19200300.102.1.6

updatesDocument

Definition

Contains the distinguished name of a document for which this document is an updated version.

For example:

```
updatesDocument: cn=Document Version 1, ou=Document Library,
 dc=example, dc=com
```

This attribute is defined in Internet White Pages Pilot.

Syntax

DN, multi-valued.

OID

0.9.2342.19200300.102.1.5

userCertificate

Definition
This attribute contains a certificate. It is to be stored and requested in the binary form, as `userCertificate;binary`.

For example:

```
userCertificate;binary:: AAAAAA==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.36

userClass

Definition
Specifies a category of computer user. The semantics of this attribute are arbitrary. The organizationalStatus attribute makes no distinction between computer users and others users and may be more applicable.

For example:

```
userClass: intern
```

This attribute is defined in RFC 1274.

Syntax
DirectoryString, multi-valued.

OID
0.9.2342.19200300.100.1.8

userPassword

Definition
Identifies the entry's password and encryption method in the following format:

```
{encryption method}encrypted password
```

Transfer of clear text passwords is strongly discouraged where the underlying transport service cannot guarantee confidentiality. Transfer of clear text may result in disclosure of the password to unauthorized parties.

For example:

```
userPassword: {ssha}9LsFG7RT+dFnPErwSfxDlaQTn6dbIFGklMNFRr==
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.35

userPKCS12

Definition
This attribute provides a format for the exchange of personal identity information. The attribute is to be stored and requested in binary form, as `userPKCS12;binary`. The attribute values are PFX PDUs stored as binary data.

This attribute is defined in RFC 2798.

Syntax
Binary, multi-valued.

OID
2.16.840.1.113730.3.1.216

userSMIMECertificate

Definition
Used by Netscape Communicator for S/MIME. This attribute is to be stored and requested in the binary form, as `userSMIMECertificate;binary`.

For example:

```
userSMIMECertificate;binary:: AAAAAA==
```

This attribute is defined in RFC 2798.

Syntax
Binary, multi-valued.

OID
2.16.840.1.113730.3.1.40

x121Address

Definition
Defines the X.121 address of a person.

This attribute is defined in RFC 2256.

Syntax
IA5String, multi-valued.

OID
2.5.4.24

x500UniqueIdentifier

Definition
Reserved for future use. A binary method of identification useful for differentiating objects when a distinguished name has been reused.

For example:

```
x500UniqueIdentifier: 17B
```

This attribute is defined in RFC 2256.

Syntax
Binary, multi-valued.

OID
2.5.4.45

Chapter 12

Operational Attributes

This chapter describes the operational attributes used by the Directory Server. Operational attributes may be available for use on every entry in the directory, regardless of whether they are defined for the object class of the entry. Operational attributes are returned in an `ldapsearch` operation only if they are specifically requested.

accountUnlockTime

Definition
Indicates the exact time after which a user can attempt to bind to the directory (after an account lockout). This attribute is used only when the password policy is enabled.

This attribute is defined in Sun ONE Directory Server.

Syntax
GeneralizedTime, single-valued.

OID
2.16.840.1.113730.3.1.95

aci

Definition
Used by the directory server to evaluate what rights are granted or denied when it receives an LDAP request from a client. Note that this is an operational attribute. It is not returned in a search unless you explicitly request it.

This attribute is defined in Sun ONE Directory Server.

Syntax
IA5String, multi-valued.

OID
2.16.840.1.113730.3.1.55

attributeTypes

Definition
Multi-valued attribute that specifies the attribute types used within a subschema. Each value describes a single attribute.

This attribute is defined in RFC 2252.

Syntax
Attribute types syntax, multi-valued.

OID
2.5.21.5

copiedFrom

Definition
Used by read-only replica to recognize master data source. Contains a reference to the server that holds the master data. Note that this attribute is only used for legacy replication. It is not used for multi-master replication.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.613

copyingFrom

Definition
Used by read-only replica to recognize master data source while replication is in progress. Contains a reference to the server that holds the master data. Note that this attribute is only used for legacy replication. It is not used for multi-master replication.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.614

dITContentRules

Definition
Multi-valued attribute that defines the DIT content rules in force within a subschema. Each value defines one DIT content rule. Each value is tagged by the object identifier of the structural object class to which it pertains.

Note that Sun ONE Directory Server does not support or use this attribute.

This attribute is defined in RFC 2252.

Syntax
DIT content rules syntax, multi-valued.

OID
2.5.21.2

dITStructureRules

Definition
Multi-valued attribute that defines the DIT structure rules in force within a subschema. Each value defines one DIT structure rule.

Note that Sun ONE Directory Server does not support or use this attribute.

This attribute is defined in RFC 2252.

Syntax
DIT structure rules syntax, multi-valued.

OID
2.5.21.1

ds-pluginDigest

Definition
The configuration digest of a signed plug-in. (The plug-in entry DN, ID, version, type, init function, and vendor are hashed together to create the configuration digest.)

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
1.3.6.1.4.1.42.2.27.9.1.57

ds-pluginSignature

Definition
The configuration signature of a signed plug-in.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
1.3.6.1.4.1.42.2.27.9.1.7

ds5PartialReplConsumerFlagged

Definition
Specifies that a consumer will receive partial replication updates.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
1.3.6.1.4.1.42.2.27.9.1.23

ldapSyntaxes

Definition
This attribute identifies the syntaxes implemented, with each value corresponding to one syntax.

This attribute is defined in RFC 2252.

Syntax
LDAP Syntaxes syntax, multi-valued.

OID
1.3.6.1.4.1.1466.101.120.16

matchingRules

Definition
Multi-valued attribute that defines the matching rules used within a subschema. Each value defines one matching rule.

This attribute is defined in RFC 2252.

Syntax
Matching rule syntax, multi-valued.

OID
2.5.21.4

matchingRuleUse

Definition
Used to indicate the attribute types to which a matching rule applies in a subschema.

This attribute is defined in RFC 2252.

Syntax
Matching rule syntax, multi-valued.

OID
2.5.21.8

nameForms

Definition
Multi-valued attribute that defines the name forms used in a subschema. Each value defines one name form.

Note that Sun ONE Directory Server does not support or use this attribute.

This attribute is defined in RFC 2252.

Syntax
Name form syntax, multi-valued.

OID
2.5.21.7

namingContexts

Definition
Corresponds to a naming context the server is mastering or shadowing. When the directory server does not master any information (for example, it is an LDAP gateway to a public X.500 directory), this attribute is absent. When the directory server believes it contains the entire directory, the attribute has a single value, and that value is the empty string (indicating the null DN of the root).This attribute permits a client contacting a server to choose suitable base objects for searching.

This attribute is defined in RFC 2252.

Syntax
DN, multi-valued.

OID
1.3.6.1.4.1.1466.101.120.5

nsds5replconflict

Definition
This attribute is a conflict marker attribute. It is included on entries that have a change conflict that cannot be resolved automatically by the replication process.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, multi-valued.

OID
2.16.840.1.113730.3.1.973

nsRole

Definition
This attribute is a computed attribute that is not stored with the entry itself. It identifies which roles an entry belongs to.

This attribute is defined in Sun ONE Directory Server.

Syntax
DN, multi-valued.

OID
2.16.840.1.113730.3.1.574

nsRoleDN

Definition
This attribute contains the distinguished name of each managed role to which the entry belongs. Membership of a managed role is conferred upon an entry by adding the role's DN to the entry's `nsRoleDN` attribute.

This attribute is not to be confused with the generated `nsRole` attribute that contains the DN of *all* roles to which the entry belongs, as computed by the Directory Server. Use `nsRoleDN` to set managed role membership, and use `nsRole` to evaluate role membership.

For example:

```
dn: cn=staff,ou=People,dc=example,dc=com
objectclass: LDAPsubentry
objectclass: nsRoleDefinition
objectclass: nsSimpleRoleDefinition
objectclass: nsManagedRoleDefinition
```

```
dn: uid=bjensen,ou=People,dc=example,dc=com
objectclass: top
objectclass: person
sn: Jensen
cn: Babs Jensen
uid: bjensen
nsroledn: cn=staff,ou=People,dc=example,dc=com
```

A nested role specifies containment of one or more roles of any type. In that case, nsRoleDN defines the DN of the contained roles.

For example:

```
dn: cn=everybody,o=SunONE,o=example.com
objectclass: LDAPsubentry
objectclass: nsRoleDefinition
objectclass: nsComplexRoleDefinition
objectclass: nsNestedRoleDefinition
nsroledn: cn=manager,ou=People,dc=example,dc=com
nsroledn: cn=staff,ou=People,dc=example,dc=com
```

This attribute is defined in Sun ONE Directory Server.

Syntax

DN, multi-valued.

OID

2.16.840.1.113730.3.1.575

numSubordinates

Description

Indicates how many immediate subordinates an entry has.

For example, `numSubordinates=0` in a leaf entry.

This attribute is defined in numSubordinates Internet Draft.

Syntax

Integer, single-valued.

OID

1.3.1.1.4.1.453.16.2.103

objectClasses

Definition
Multi-valued attribute that defines the object classes used in a subschema. Each value defines one object class.

This attribute is defined in RFC 2252.

Syntax
Object classes syntax, multi-valued.

OID
2.5.21.6

passwordAllowChangeTime

Definition
Indicates the exact time after which the user can change their password.

This attribute is defined in Sun ONE Directory Server.

Syntax
GeneralizedTime, single-valued.

OID
2.16.840.1.113730.3.1.214

passwordExpirationTime

Definition
Indicates the exact time after which the user's password expires.

This attribute is defined in Sun ONE Directory Server.

Syntax
GeneralizedTime, single-valued.

OID
2.16.840.1.113730.3.1.91

passwordExpWarned

Definition
Indicates that a password expiration warning has been sent to the user.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
2.16.840.1.113730.3.1.92

passwordHistory

Definition
Contains the history of the user's previous passwords.

This attribute is defined in Sun ONE Directory Server.

Syntax
Binary, multi-valued.

OID
2.16.840.1.113730.3.1.96

passwordPolicySubentry

Definition
The DN of an LDAPsubentry containing the password policy attributes that will be applied to a user entry.

This attribute is defined in Sun ONE Directory Server.

Syntax
DirectoryString, single-valued.

OID
1.3.6.1.4.1.42.2.27.9.1.30

passwordRetryCount

Definition
Counts the number of consecutive failed attempts at entering the correct password.

This attribute is defined in Sun ONE Directory Server.

Syntax
Integer, single-valued.

OID
2.16.840.1.113730.3.1.93

retryCountResetTime

Definition
Specifies the exact time after which the passwordRetryCount is reset.

This attribute is defined in Sun ONE Directory Server.

Syntax
GeneralizedTime, single-valued.

OID
2.16.840.1.113730.3.1.94

subschemaSubentry

Definition
DN of the entry that contains schema information for this entry. This attribute is present for every entry in the directory.

For example:

```
subschemaSubentry: cn=schema
```

This attribute is defined in RFC 2252.

Syntax
DN, single-valued.

OID
2.5.18.10

supportedControl

Definition
The values of this attribute are the object identifiers (OIDs) that identify the controls supported by the server. When the server does not support controls, this attribute is absent.

This attribute is defined in RFC 2252.

Syntax
OID, multi-valued.

OID
1.3.6.1.4.1.1466.101.120.13

supportedExtension

Definition
The values of this attribute are the object identifiers (OIDs) that identify the supported extended operations supported by the server. When the server does not support extensions, this attribute is absent.

This attribute is defined in RFC 2252.

Syntax
OID, multi-valued.

OID
1.3.6.1.4.1.1466.101.120.7

supportedLDAPVersion

Definition
Identifies the versions of the LDAP protocol implemented by the server. This attribute is defined in RFC 2252.

Syntax
Integer, multi-valued.

OID
1.3.6.1.4.1.1466.101.120.15

supportedSASLMechanisms

Definition
Identifies the names of supported SASL mechanisms supported by the server. When the server does not support SASL attributes, this attribute is absent. This attribute is defined in RFC 2252.

Syntax
DirectoryString, multi-valued.

OID
1.3.6.1.4.1.1466.101.120.14

vendorName

Definition
Represents the name of the LDAP server implementer. This attribute must not be used by client applications to gather information related to supported features of the LDAP implementation.

For example:

```
vendorName: Sun Microsystems, Inc.
```

This attribute is defined in RFC 3045.

Syntax
DirectoryString, single-valued.

OID
1.3.6.1.1.4

vendorVersion

Definition

Represents the version of the LDAP server implementation. This attribute must not be used by client applications to gather information related to supported features of the LDAP implementation.

For example:

```
vendorVersion: v5.2
```

This attribute is defined in RFC 3045.

Syntax

DirectoryString, single-valued.

OID

1.3.6.1.1.5

Part 5

Appendices

This part includes the following appendices:

- Error Codes
- ns-slapd and slapd.exe Command-Line Utilities
- Directory Internationalization
- LDAP URLs
- LDAP Data Interchange Format

Appendix A

Error Codes

This appendix provides an extensive list of the error messages generated by Sun ONE Directory Server. While this list is not exhaustive, the information presented in this chapter will serve as a good starting point for common problems.

Common Error Codes

The following table describes the error codes displayed in the error log and the appropriate action to take should these errors occur.

Errors are defined according to their severity:

- *Error* - The error is severe. Immediate action should be taken to avoid the loss or corruption of directory data.
- *Warning* - Action should be taken at some stage to prevent a severe error occurring in the future.
- *Info* - An informative message, usually describing server activity. No action is necessary.

In this release, only the severe *Error* codes are documented. If you require further assistance in diagnosing errors, please contact Sun ONE Technical Support: `http://www.sun.com/service/sunone/software/index.html`

NOTES In the case of internal errors, plug-in writers should check their parameters to `slapi` functions first.

When using the error log for debugging, increase the log level progressively until the debugging data you need becomes evident in the log. Do not enable error logging for all Directory Server components at once, especially on a production system, to avoid severely impacting performance.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4104	Error	No backend has been defined to do the import.	The server cannot detect a backend to do the import. This is an internal error and should not occur under normal circumstances.	Contact Sun ONE Technical Support.
4105	Error	Bulk import not supported by this backend.	The backend will not accept wire import. This is an internal error and should not occur under normal circumstances.	Contact Sun ONE Technical Support.
4107	Error	Ignoring extremely large value for configuration attribute *attribute_name*.	The value of the specified configuration attribute is too large.	Change the value of the specified configuration attribute. See the attribute description in this Reference for the acceptable value range.
4108	Error	The given file *filename* could not be accessed.	The server is unable to obtain any information on the specified configuration file.	Check that the file exists and that it has the appropriate access rights.
4109	Error	The given file *filename* could not be opened for reading.	The server is unable to open the specified configuration file.	Check that the file exists and that it has the appropriate access rights.
4110	Error	Could only read *value* of *value* bytes from configuration file *filename*.	The server is unable to read the specified configuration file.	Check that the file exists and that it has the appropriate access rights.
4111	Error	The default password storage scheme SSHA could not be read or was not found in the file *filename*. It is mandatory. Server exiting.	The mandatory password storage scheme Salted Secure Hashing Algorithm (SSHA) could not be retrieved from the configuration file.	Check that the password storage scheme SSHA exists in the configuration file. If it is not present, add it.
4112	Error	Skipping plugin *plugin* - no valid signature.	The specified plug-in does not have a valid signature.	Provide a valid signature for the plug-in or disable the plug-in.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4112	Error	Unable to load plugin *plugin_name*.	An error occurred while loading configuration information for the specified plug-in.	Check that the configuration information for the specified plug-in is accurate. For more information, it may be useful to turn debugging on for `SLAPI_DEBUG_PLUGIN`. Change the configuration information as required and restart the server.
4119	Error	No password storage scheme plugins defined in the configuration.	No encoding scheme was found in the configuration file. Under normal circumstances, this error will not occur, because the server cannot start if the mandatory scheme SSHA is not present in the configuration file.	Add a password storage scheme plug-in to the configuration file and restart the server.
4120	Error	Invalid scheme to hash password: *scheme*. Valid values are: *scheme values*.	The tag (algorithm) specified to hash the password is not defined in the configuration file.	Add a password storage scheme to the configuration file, or change the specified scheme, and restart the server.
4121	Error	Invalid scheme: *scheme*. No password storage scheme loaded.	The tag (algorithm) specified to hash the password is defined but the server is unable to retrieve the associated information.	Check the password storage scheme configuration and its installation and restart the server.
4122	Error	The configuration files in *directory* directory could not be read or were not found. Please refer to the error log or output for more information.	An error occurred reading the configuration files. The specific cause for the error is logged in the log files.	See the log files for more information.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4123	Error	The configuration file `dse.ldif` in directory *directory* could not be read or was not found. Please refer to the error log or output for more information.	An error occurred reading the `dse.ldif` configuration file. The specific cause for the error is logged in the log files.	See the log files for more information.
4124	Error	Unknown attribute *attribute_name* will be ignored	An attempt was made to set an unknown attribute in the configuration file.	Check and correct the attribute name.
4125	Error	The configuration file *filename* was not restored from backup.	The configuration file backup has failed. The reason for the failed backup is provided in the error message.	Correct the error and back up the configuration file manually.
4126	Error	Failed to create lock. Cannot register supported SASL mechanism. Server exiting.	This indicates a resource problem on the machine.	Restart the server.
4127	Error	Failed to create lock. Cannot register supported extended operations. Server exiting.	This indicates a resource problem on the machine.	Restart the server.
4128	Error	Could not load configuration file *filename*.	An error occurred when attempting to load the specified configuration file.	Check that the configuration file exists and that it has the appropriate access permissions. See the error log for more details.
4129	Error	Bad configuration file. Edit the configuration file to correct the reported problems and then restart the server. Server exiting.	There is an error in the configuration file. Details of the error are reported in the error log.	Edit the configuration file to correct the reported problems and restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4130	Error	Cannot copy DSE file *filename* to *path*.	Several possible causes (file system full, incorrect permissions, etc.). Details of the error are reported in the error log.	Check that the configuration file exists and that it has the appropriate access permissions.
4131	Error	The entry *entry_name* in file *filename* is invalid.	The server cannot read the specified entry. Details of the error are provided in the error message.	Check that the entry is valid and change as necessary.
4132	Error	Cannot parse dse entry *entry_name*.	The server cannot parse the specified entry. There is an error in the LDIF syntax of the entry.	Check that the entry is valid and change as necessary.
4133	Error	Cannot write temporary DSE file *filename*.	System error (file system full, incorrect permissions, etc.)	Check the log file for more information and restart the server.
4134	Error	Cannot backup DSE file *filename*.	The server cannot write to the specified DSE file.	Check the specified path and ensure that you have the appropriate write permissions.
4135	Error	Cannot rename temporary DSE file *filename*.	The server cannot rename the specified DSE file.	Check the specified path and ensure that you have the appropriate write permissions.
4136	Error	Invalid plugin action *plugin_name*.	The configuration file contains an invalid value for the specified plug-in.	Check the value in the configuration file and set a valid value.
4137	Error	Attempting to delete a child entry whose existence is unknown to the parent. Deletion attempt ignored.	An attempt was made to delete a child entry for which there was no subcount on the parent.	This error should not occur under normal circumstances.
4138	Error	Failed to start *plugin_name* plug-in.	Plug-in dependencies have not been configured correctly.	Check that the dependencies are valid and that they are enabled.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4139	Error	Failed to resolve plug-in dependencies.	An error occurred while resolving dependencies (usually the consequence of an earlier problem - disabled plug-in, etc.)	Check that the dependencies are valid and that they are enabled.
4140	Error	Could not load symbol *symbol_name* from library *library_name* for plug-in *plugin_name*.	This may be due to: 1. Incorrect configuration of the plug-in entry in the `dse.ldif` file. 2. The library is missing or in the wrong location. 3. The expected symbol corresponding to the `init` function could not be found in the library.	1. Check the plug-in configuration in the `dse.ldif` file. 2. Check that the library path and the `init` function name are correct.
4152	Error	Unknown plugin type *type*.	A plug-in configuration entry does not have a recognized plug-in type.	Check the configuration and correct the specified plug-in entry.
4153	Error	Only one instance allowed for plugin type *type*.	Multiple plug-ins of the specified type have been defined in the configuration. Only a single plug-in of that type is allowed.	Correct the configuration so that there is only a single plug-in of the specified type.
4158	Error	UNBIND	Invalid unbind PDU. This is an error in the client code.	Correct the error in the client code.
4159	Error	Bad controls in the UNBIND.	Invalid controls in an unbind PDU. The control is marked as critical and is unknown to the server or the control is badly encoded. This is an error in the client code.	The client should not require critical controls on unbind. Correct the error in the client code.
4162	Error	ldapu_get_cert_subject_dn_fails	The server is unable to obtain the subject in the client certificate.	Check the message in the error log for more information.
4163	Error	ldapu_get_cert_issuer_dn_fails	The server is unable to obtain the certificate issuer of the client certificate.	Check the message in the error log for more information.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4164	Error	Bad BER decoding of an attribute value assertion.	An error occurred during the decoding of an attribute value assertion. The format of the attribute value assertion is incorrect.	Check the client application making the request.
4165	Error	BER decoding: found *id* instead of *id* for MessageId.	The MessageID tag was not found in the LDAP request.	The request is invalid. Check the application that created the request.
4166	Error	BER decoding: ber_peek_tag returns no Operation tag.	An error occurred while decoding the operation tag.	The request is invalid. Check the application that created the request.
4167	Error	Load library error.	An error occurred while loading the dynamic library. This may be because the library does not exist, the library requires another library that does not exist, or the library could not resolve a symbol.	Check that the library exists and is accessible.
4177	Error	Could not open lockfile *filename* in write mode.	The specified lock file could not be opened.	Check that the lock file exists and is accessible.
4178	Error	Could not open file *filename* in mode *mode*.	The specified file could not be opened.	Check that the file exists and is accessible.
4191	Error	Failed to change user and group identity to that of *user*.	The server was unable to change the user and group identity to the specified user.	Check the user privileges and correct.
4612	Error	Unable to start slapd because it is already running as process *process*.	Unable to start slapd because it is already running.	Stop the running server instance before launching a new server.
4613	Error	Unable to start slapd because the process *process* is importing the database	Unable to start slapd because a process is currently importing the database.	Stop the running import process instance before launching a new server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4614	Error	Unable to run db2ldif with the -r flag because the database is being used by another slapd process.	Unable to run `db2ldif` with the `-r` flag because the database is being used by another slapd process.	If the other process is not an import process, run `db2ldif.pl -r` instead. If it is an import process, stop the running import process before launching `db2ldif`.
4615	Error	Unable to run db2ldif because the process *process* is importing the database	Unable to run `db2ldif` because a process is currently importing the database.	Stop the running import process before launching `db2ldif`.
4616	Error	Unable to run `db2bak` because the process *process* is importing the database	Unable to run `db2bak` because a process is importing the database.	Stop the running import process before launching `db2bak`.
4617	Error	Unable to import the database because it is being used by another slapd process	Unable to import the database because it is being used by another slapd process.	Stop the running slapd process before importing.
4618	Error	Unable to create an index because the database is being used by another slapd process	Unable to create an index because the database is being used by another slapd process.	Stop the running slapd process before creating indexes.
4623	Error	Pathname *path* too long.	When trying to convert the absolute path, it was discovered that the pathname is too long.	Change the relative path or the absolute path base so that the sum of their length is lower than the maximum allowed length.
4625	Error	Cannot determine current directory.	When trying to convert the absolute path, the server was unable to determin the current directory.	Contact Sun ONE Technical Support.
4626	Error	slapi_add_internal: add_values for type *type* failed.	Internal error when converting from a set of modifications to an entry.	Contact Sun ONE Technical Support.
4627	Error	Unable to test the database because it is being used by another slapd process	Unable to test the database because it is being used by another slapd process.	Stop the running process and retry.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4629	Error	Unable to create directory.	System error - the directory could not be created.	Check that your file system is valid and retry.
4631	Error	ref_adjust: referrals suppressed (could not get target DN operation or scope from pblock).	Referrals have been suppressed. The server was unable to obtain the target DN and operation structure.	Contact Sun ONE Technical Support.
4633	Error	Suffix to be imported contains encrypted attributes.	No password for the key database has been supplied within the arguments configured for this suffix. The password is required to retrieve the key and proceed with encryption.	Use the `-Y` *pwd* or `-y` *pwd-file* arguments when executing the `ldif2db` command.
4634	Error	Security initialisation for attribute encryption failed.	The security initialisation required by the attribute encryption feature failed.	Make sure that the password supplied is correct and that the password file syntax is correct. Check that SSL has been configured correctly (cert file ciphers.)
4737	Error	Security Initialization failed: unable to read configuration from *dn*.	Security initialization failed. The server was unable to read the configuration from the specified configuration DN.	Check that the configuration DN is valid and retry.
4738	Error	Security Initialization: Failed to retrieve SSL configuration attribute nscertfile from *filename*	Security initialization error. The server was unable to retrieve the SSL configuration attribute `nscertfile`.	Check that the value of the `nscertfile` attribute is correct and retry.
4739	Error	Security Initialization: Failed to retrieve SSL configuration information (error *error*): nskeyfile: *filename* nscertfile: *filename*	Security initialization error. The server was unable to retrieve one of the SSL configuration attributes, `nscertfile` or `nskeyfile`.	Check that the value of the `nscertfile` and `nskeyfile` attributes are correct and retry.
4740	Error	Security Initialization: NSS initialization failed (error *error*): path: *path* certdb prefix: *prefix* keydb prefix: *prefix*.	Security initialization error. NSS initialization failed.	Check the NSS configuration and retry.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4741	Error	Security Initialization: NSS initialization failed (error *error*)	Security initialization error. NSS initialization failed.	Contact Sun ONE Technical Support.
4742	Error	Security Initialization: Failed to retrieve SSL configuration information (error *error*): nssslSessionTimeout: variable	Security initialization error. The server was unable to retrieve the SSL configuration attribute `nssslSessionTimeout`.	Check that the value of the `nssslSessionTimeout` attribute is correct and retry.
4744	Error	Security Initialization: Unable to get token for variable cipher family (error *error*)	Security initialization error. The server was unable to obtain the required token (from the `nsssltoken` attribute).	Check that the `nsssltoken` attribute is present in the cipher family entry, and that it has a valid value.
4745	Error	Security Initialization: Unable to find slot for variable cipher family (error *error*)	Security initialization error. The server was unable to find the required slot.	Make sure that the security token (external or internal) is accessible to the server.
4746	Error	slapd_get_tmp_dir mkdir(variable) Error: *error*	System error. The server was unable to create a *temp* directory.	Check that the current user has sufficient access rights to create the *temp* directory and retry.
4747	Error	Security Initialization: Unable to set SSL export policy (error *error*)	Security initialization error. The server was unable to set the SSL export policy.	Contact Sun ONE Technical Support.
4748	Error		Security initialization error. The server was unable to set SSL cipher preference information.	1. Check the syntax of the ciphers in the configuration. 2. Make sure that all the ciphers are supported by the server.
4749	Error	Security Initialization: Failed to import NSPR fd into SSL (error *error*)	Security initialization error. The server was unable to import the NSPR file descriptor into SSL.	Contact Sun ONE Technical Support.
4750	Error	Security Initialization: Unable to get internal slot (error *error*)	Security initialization error. The server was unable to obtain the internal slot?	Contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4751	Error	Security Initialization: Unable to authenticate (error *error*)	Security initialization error. The server was unable to authenticate.	Contact Sun ONE Technical Support.
4756	Error	None of the ciphers are valid.	The ciphers are invalid.	Check the ciphers and retry.
4757	Error	Config of SSL session cache failed: out of disk space! Make more room in the temp directory and try again.	The configuration of the SSL session cache failed, due to a disk space problem.	Free up some room in the */tmp* directory directory and retry.
4758	Error	Config of SSL session cache failed (error *error*).	The configuration of the SSL session cache failed.	Contact Sun ONE Technical Support.
4759	Error	Security Initialization: Failed to enable security on the imported socket (error *error*)	Security initialization error. The server could not enable security on the imported socket.	Contact Sun ONE Technical Support.
4760	Error	Security Initialization: Failed to enable SSLv3 on the imported socket (error *error*)	Security initialization error. The server could not enable SSLv3 on the imported socket.	Contact Sun ONE Technical Support.
4761	Error	Security Initialization: Failed to enable TLS on the imported socket (error *error*)	Security initialization error. The server could not enable TLS on the imported socket.	Contact Sun ONE Technical Support.
4766	Error	Encryption alias not configured.	The encryption alias has not been configured.	Contact Sun ONE Technical Support.
4769	Error	Failed to set SSL client ready for client authentication: certificate db: *database* returned code *return_code* (error *error*)	The server was unable to set the SSL client ready for client authentication.	Check that the certificate and key databases are accessible to the server (acting as an SSL client).
4772	Error	SSL client authentication cannot be used (no password) (error *error*)	SSL client authentication cannot be used because a password has not been defined.	Make sure that the server receives the password for the security token, using a `pin.txt` file option with the `start-slapd` command.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4773	Error	ldapssl_enable_ clientauth (*variable*) (error *error*)	SSL error - the server cannot enable client authentication.	Check that the password given to the server is correct.
4774	Error	ldap_simple_bind_s (*variable*) (error *error*)	Simple bind over SSL failed. The password may be incorrect.	Check that the password for the DN is correct.
4775	Error	ldap_sasl_bind(““LDAP _SASL_EXTERNAL) (error *error*)	The bind attempt failed with the SASL EXTERNAL method. The server was unable to find any external credentials.	Make sure that the client's certificate is received by the server before the bind attempt.
4776	Error	sasl *error* message	SASL error. The details of the error are logged in the error log.	Check the error log for more information.
4779	Error	Security initialization: Unable to create PinObj (error *error*.)	Security initialization error. The server was unable to create the pin object.	Make sure that the server receives the password for the security token, using a `pin.txt` file option with the `start-slapd` command.
4780	Error	Security Initialization: Unable to authenticate to slot for *variable* cipher family (error *error*)	Security initialization error. The server was unable to authenticate to the required slot.	The password entered was incorrect. Check the correct password and retry.
4781	Error	SSL is misconfigured. Client authentication is enabled but no certificate authority is trusted for SSL client authentication.	The server is configured to allow or require client authentication for SSL. The database contains no CA certificates marked as trusted for issuing client certificates. The server cannot perform SSL client authentication.	Install one or more CA certificates using the console. Ensure that the trust attributes of CA certificates installed with `certutil` include the T trust attribute.
4782	Error	Failed to create context for cipher operation.	NSS context creation failed.	Ensure that a valid certificate is available so that the key may be generated.
4785	Error	Cipher operation failed.	The server was unable to accomplish the cipher operation.	It is likely that the context is incorrect. Restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4786	Error	Crypto mechanism not supported by this server.	The cryptography mechanism is invalid or unsupported.	Generate a symmetric key for the cryptography mechanism or choose a supported mechanism.
4793	Error	Failed to generate symmetric key.	The server was unable to generate the symmetric key.	Check that a security token is available to the server (as a certificate.)
4795	Error	Failed to map key generation parameters into crypto operation ones.	The server was unable to map the key generation mechanism to the cryptography mechanism.	Restart the server.
4796	Error	Unable to retrieve private key for certificate.	The server was unable to retrieve a private key from the certificate.	Ensure that the certificate has been imported into the database with both its private and public keys. (This is usually performed as part of the process beginning with a certificate request.)
4797	Error	Signature failed.	The signature required for attribute encryption failed.	Restart the server.
4798	Error	Key database password was rejected.	The password for the key database has been rejected.	Enter a new password and retry.
4799	Error	Couldn't read key database password.	The server was unable to find the key database password. No password was provided, or the password syntax was incorrect.	Enter a non-null password or ensure that a valid password file, containing a valid password, is supplied.
4800	Error	No key db password was specified.	No key database password was specified (either explicitly or via a password file.)	Supply a valid password or the path to a valid password file.
4801	Error	Unable to read key password file from *directory*.	The server was unable to read the key database password from the password file.	Check the password file access rights and ensure that the file is of a reasonable size.
4802	Error	Bad password file syntax: missing ':' preceeding password.	The syntax of the password file is incorrect. The ":" is missing.	Supply a password file with the correct syntax.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4803	Error	Bad token identifier: *token*.	The token identifier in the password file does not match the open token.	Supply a token identifier that is consistent with the `nsSSLToken` attribute value in the configuration.
4865	Error	Detected virtual attribute loop in get on entry *entry* attribute *attribute*.	A loop was detected while retrieving the virtual attributes of an entry.	Check the virtual attributes configured for this entry and break the loop.
4866	Error	Out of memory to duplicate a type name.	There is insufficient memory for the server to allocate a service provider for the virtual attributes map insert.	Make more memory available to the server and restart the server.
4867	Error	Detected virtual attribute loop in compare on entry *entry* attribute *attribute*.	The server detected a virtual attribute loop when comparing virtual attribute service providers.	Check the virtual attributes configured for this entry and break the loop.
4868	Error	Out of memory to allocate a service provider.	There is insufficient memory for the server to allocate a service provider for the virtual attributes register.	Make more memory available to the server and restart the server.
4869	Error	Out of memory to allocate a service provider handle.	There is insufficient memory for the server to allocate a service provider handle.	Make more memory available to the server and restart the server.
4870	Error	Out of memory to create a map for virtual attributes.	There is insufficient memory for the server to allocate a map for virtual attributes.	Make more memory available to the server and restart the server.
4871	Error	Out of memory to create a new hash table.	There is insufficient memory for the server to allocate a new hash table for virtual attributes.	Make more memory available to the server and restart the server.
4872	Error	Failed to create a new lock for virtual attributes map insert.	The server was unable to create a new lock for virtual attribute map creation. This is probably due to a memory error.	Make more memory available to the server and restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
4994	Error	Multiple backend instances are specified.	More than one backend instance has been specified for the attempted task.	Contact Sun ONE Technical Support.
4995	Error	Cannot perform an import with pre-V3 backend plugin.	You are a version of the backend plug-in API that is no longer supported and cannot perform the database import.	Upgrade to a newer version of the backend plug-in API (at least version 3), recompile, and add the import functionality.
4996	Error	No ldif2db function defined for backend *backend*	No `ldif2db` function is defined for this backend. This kind of database is unable to perform an import.	Use a backend that has the import functionality.
4997	Error	Unable to allocate new task for import.	The server is unable to allocated a new task for the import. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
4998	Error	Cannot export - backend not found.	The database could not be exported because the specified backend could not be found.	Check the configuration file and make sure that the correct database and suffix are specified.
4999	Error	ldbm2ldif: backend backend export failed (*error*)	The `db2ldif` function failed when attempting to export the database.	See the error log for more information and contact Sun ONE Technical Support.
5000	Error	No backend instance names are specified.	The database could not be exported because no backend instance names were specified.	Contact Sun ONE Technical Support.
5003	Error	Cannot perform an import with pre-V3 backend plugin.	You are a using version of the backend plug-in API that is no longer supported and cannot perform the database import.	Upgrade to a newer version of the backend plug-in API (at least version 3), recompile, and add the import functionality.
5004	Error	No ldif2db function defined for backend *backend*	No `ldif2db` function is defined for this backend. This kind of database is unable to perform an import.	Use a backend that has the import functionality.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5005	Error	Unable to allocate new task.	The server is unable to allocated a new task for the export. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5006	Error	Unable to create ldbm2ldif thread for export.	The server is unable to create a thread for the export. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5007	Error	db2archive function failed when trying to backup (error *error*)	The `db2archive` function failed when attempting to backup.	See the error log for more information and contact Sun ONE Technical Support.
5008	Error	Unable to process backup when no db2archive function defined	The database could not be backed up because the `db2archive` function was not defined.	None - this type of database cannot be backed up.
5009	Error	Cannot perform a backup with pre-V3 backend plugin variable	You are a using version of the backend plug-in API that is no longer supported and cannot perform the database backup.	Upgrade to a newer version of the backend plug-in API (at least version 3), recompile, and add the backup functionality.
5010	Error	Unable to allocate new task for backup.	The server is unable to allocated a new task for the backup. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5011	Error	Unable to create backup thread.	The server is unable to create a backup thread. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5012	Error	Restore failed (error *error*)	The restore process failed.	See the error log for more information and contact Sun ONE Technical Support.
5014	Error	Cannot perform a restore with pre-V3 backend plugin variable	You are using a version of the backend plug-in API that is no longer supported and cannot perform the database restore.	Upgrade to a newer version of the backend plug-in API (at least version 3), recompile, and add the restore functionality.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5015	Error	Unable to allocate new task for restore.	The server is unable to allocated a new task for the restore. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5016	Error	Unable to create restore thread for restore.	The server is unable to create a restore thread. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5017	Error	db2index function failed when trying to restore (error *error*)	The `db2index` function failed when attempting to restore the database.	See the error log for more information and contact Sun ONE Technical Support.
5019	Error	No db2index function defined for backend *backend*.	The database could not be indexed because no db2index function was defined for the backend.	Contact Sun ONE Technical Support.
5020	Error	Unable to allocate new task for index.	The server is unable to allocated a new task for the index. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5021	Error	Unable to create index thread.	The server is unable to create an index thread. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5023	Error	Cannot create task node (error *error*)	The server is unable to create a task node.	See the error log for more information and contact Sun ONE Technical Support.
5024	Error	Unable to create global tasks lock.	The server is unable to create a global tasks lock. This is usually due to a resource problem.	Free up resources on the machine and restart the server.
5025	Error	Cannot import. Lookup instance name by suffixes failed.	The database could not be imported because the server was unable to locate the instance name for the specified suffix.	Check that the suffix is specified correctly in the configuration.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5026	Error	Cannot import. Could not find database for suffix.	The database could not be imported because the server was unable to locate the database for the specified suffix.	Check that the database and the suffix are specified correctly in the configuration.
5027	Error	Cannot import. Backend not found.	The database could not be imported because the server was unable to locate the specified backend.	Check that the database and the suffix are specified correctly in the configuration.
5028	Error	Cannot import - lookup instance names by suffix failed.	The database could not be imported due to a problem with the suffix configuration.	Check that the suffix is specified correctly in the configuration.
5029	Error	Could not find database for suffix.	The database could not be exported because it could not be found.	Check that the database and the suffix are specified correctly in the configuration.
5030	Error	No archive2db function defined.	The database could not be restored because the `archive2db` function was not defined.	None - this type of database cannot be restored.
5031	Error	Cannot index - backend not found.	The server cannot index the database because the specified backend was not found.	Contact Sun ONE Technical Support.
5034	Error	Incompatible options nsExportReplica=true and dsDecryptAttrs=false: cannot dump replica with encrypted attributes.	An export has been called with incompatible options nsExportReplica=true and dsDecryptAttrs=false. It is not possible to dump a replica with encrypted attributes.	Avoid using both options at the same time. Ensure that attributes are decrypted (i.e. dsDecryptAttrs=true) if you want to export the database for replication purposes.
5121	Error	reslimit_init: slapi_register_object_ extension() failed.	The server cannot register an object extension (during resource limit initialization).	Contact Sun ONE Technical Support.
5122	Error	PR_NewRWLock() failed for reslimit.	System error - the server cannot create a new lock for the resource limit.	Contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5123	Error	*error*: Resource limit initialization failed.	Resource limit initialization failed. This is likely to be a resource issue.	Check the error message in the log file and contact Sun ONE Technical Support.
5124	Error	*error*: slapi_get_object_ extension() returned NULL	The server could not obtain the object extension (for the resource limit).	Contact Sun ONE Technical Support.
5126	Error	*error*: parameter error (*attribute* already registered)	A parameter error occurred when registering a new resource to be tracked. The LDAP attribute type that can be consulted in the bound entry to determine the limit's value is already registered.	Check that the attribute provided is registered only once.
5127	Error	*error*: parameter error	A parameter error occurred when registering a new resource to be tracked.	1. Check that the type is SLAPI_RESLIMIT_TYPE_ INT 2. Check that `attrname` is an LDAP attribute type that can be consulted in the bound entry to determine the limit's value.
5127	Error	*error*: parameter error	Internal error. When retrieving the integer limit associated with a connection and a resource, a parameter with a NULL value was found.	Contact Sun ONE Technical Support.
5128	Error	error: unknown handle *handle*	Parameter error. The handle used to identify a resource is unknown.	Contact Sun ONE Technical Support.
5129	Error	Cannot malloc bytes.	An attempt is being made to allocate 0 or a negative number of bytes. This is likely to be a software issue.	Contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5130	Error	malloc of *bytes* bytes failed; errno *error*.	Memory allocation has failed. This is probably because of a lack of available memory.	Increase the virtual memory available to your server, or reduce the size of the server's maximum entries in cache (`cachesize`) or maximum database cache size (`dbcachesize`) parameters.
5132	Error	realloc of *bytes* bytes failed; errno *error*.	Memory reallocation has failed. This is probably because of a lack of available memory.	Increase the virtual memory available to your server, or reduce the size of the server's maximum entries in cache (`cachesize`) or maximum database cache size (`dbcachesize`) parameters.
5135	Error	calloc of *bytes* bytes failed; errno *error*.	Memory c-allocation has failed. This is probably because of a lack of available memory.	Increase the virtual memory available to your server, or reduce the size of the server's maximum entries in cache (`cachesize`) or maximum database cache size (`dbcachesize`) parameters.
5136	Error	strdup of *chars* chars failed; errno *error*.	String duplication has failed. This is probably because of a lack of available memory.	Increase the virtual memory available to your server, or reduce the size of the server's maximum entries in cache (`cachesize`) or maximum database cache size (`dbcachesize`) parameters.
5137	Error	ber_bvdup of *bytes* bytes failed; errno *error*.	BER value duplication has failed. This is probably because of a lack of available memory.	Increase the virtual memory available to your server, or reduce the size of the server's maximum entries in cache (`cachesize`) or maximum database cache size (`dbcachesize`) parameters.
5249	Error	The entry *entry* in the configfile *filename* was empty or could not be parsed.	An entry in the configuration file was empty or could not be parsed.	Check the entry syntax in the configuration file.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5250	Error	Invalid value	The specified configuration attribute in the dse.ldif file has no value or the value is invalid.	Check that the value of the attribute under cn=config in the dse.ldif file is either `on` or `off`.
5251	Error	Cannot set error log *filename*.	The error log filename could not be set, either because the filename was NULL or the path was invalid.	Check that the value of the attribute `nsslapd-errorlog` under cn=config in the dse.ldif file is valid, and that the path exists.
5252	Error	Undefined value for errorlog level.	The error log level could not be set because its value is undefined.	Check that the value of the attribute `nsslapd-errorlog-level` under cn=config in the dse.ldif file is set, and is correct.
5253	Error	Bad value for nsslapd-maxdescriptors.	The request to set the maximum number of file descriptors has failed. The value is either NULL, or out of the permitted range [1..max] where max is the maximum number of file descriptors that can be created by a process.	Check that the value of the attribute `nsslapd-maxdescriptors` in the dse.ldif file is not higher than the RLIMIT_NOFILE parameter, and is not lower than 1.
5254	Error	Ignoring *attribute* (since -d *option* was given on the command line) nsslapd-errorlog-level.	The attribute nsslapd-errorlog-level in the configuration file has been ignored, because the -d option was specified at the command line.	Do not specify the -d option at the command line if you want the value of this attribute in the configuration file to be taken into account.
5255	Error	The plugin entry *entry* in the configfile *filename* was invalid.	Failed to load the specified plug-in because the configuration entry of the plug-in in the dse.ldif file is invalid.	Check and correct the faulty configuration entry in the dse.ldif file.
5385	Error	Convert LDIF entry into LDAP entry fast method. Error: entry has no dn.	While attempting to convert an LDIF entry to an LDAP entry, the server found that the entry has no DN.	Check the entry and make sure that it has a DN.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5390	Error	str2entry_dupcheck: entry has no dn.	While attempting to convert a string entry to an LDAP entry, the server found that the entry has no DN.	Check the entry and make sure that it has a DN.
5392	Error	Error occurs while removing attribute values. Possible existing duplicate value for attribute type *attribute* found in entry *entry*.	An error occurred while attempting to remove attribute values. This may be due to a duplicate attribute value.	Check the attribute values being removed.
5393	Error	str2entry_dupcheck: unexpected failure constructing the value tree.	The server failed to add a value to the value tree.	Check the error log for more information.
5395	Error	Attribute 'nscpEntryWSI' can only be computed by root user.	The attribute nscpEntryWSI cannot be computed by a user who is not the Directory Manager.	Check the client application making the request. The client must bind as root to be able to compute this attribute.
5505	Error	Registration of extension failed.	A plugin has attempted to register a new extension to an object type, but the object type is in use, by at least one object.	Correct the plugin code.
5641	Error	Could not find parent node for entry *entry*. Node parent is defaulting to root node.	The parent node for the current mapping tree node could not be located.	Check the `nsslapd-parent-suffix` attribute of the entry in the configuration file (`dse.ldif`).
5642	Error	Node *node* is either a 'backend' or 'referral on update' node therefore it must define a backend (attribute 'nsslapd-backend').	The new mapping tree node is either a “backend” or “referral on update” node but has no backend defined.	Check the `nsslapd-backend` attribute of the entry in the configuration file (`dse.ldif`).
5643	Error	Node *node* is either a 'referral' or 'referral on update' node therefore it must define a referral (attribute 'nsslapd-referral').	The new mapping tree node is either a “referral” or “referral on update” node but has no referral defined.	Check the `nsslapd-referral` attribute of the entry in the configuration file (`dse.ldif`).

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5644	Error	Cannot load distribution plugin lib *library* for node *node*.	The distribution plugin could not be loaded.	Check the error log for more information. The dynamic library may not be present, may be inaccessible, or may be using another library that is not present.
5645	Error	Node *node* wants to define a distribution plugin but either nsslapd-distribution-plugin' or 'nsslapd-distribution-funct' attribute is missing in the configuration file (dse.ldif).	The entry is missing either the distribution plugin or the distribution function name.	Check the nsslapd-distribution-plugin and nsslapd-distribution-func attributes in the configuration file (`dse.ldif`).
5648	Error	Could not create mapping tree node for entry *entry*.	The mapping tree node could not be created.	Check the error log for evidence of the failure, otherwise not contact Sun ONE Technical Support.
5659	Error	Cannot find distribution function *function* in distribution plugin lib *library* for node *node*.	The distribution function in the plugin library could not be located.	Check the error log for more information. The dynamic library may not be present, may be inaccessible, or may be using another library that is not present.
5890	Error	No schema files were found in the directory *directory_name*.	No schema files are present in the schema directory.	Restore the default schema files from a backup or CD image.
5893	Error	Entry *entry* required attribute *objectclass* is missing.	The specified entry was added without an objectclass attribute.	Check the application that added the entry.
5894	Error	Entry *entry* has unknown objectclass.	The entry was added or modified with an unknown objectclass.	Check the application that added or modified the entry.
5895	Error	Entry *entry* single-valued attribute has multiple values.	The entry that was added or modified is invalid. A single-valued attribute has multiple values.	Check the application that added or modified the entry.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
5896	Error	Entry *entry* attribute *attribute* required by objectclass *objectclass* is missing.	The entry that was added or modified is missing a required attribute.	Check the application that added or modified the entry.
5897	Error	Entry *entry* attribute *attribute* is not allowed.	The entry that was added or modified contains an invalid attribute.	Check the application that added or modified the entry.
5900	Error	Missing value for objectClasses attribute.	While parsing the schema ldif file, no value was specified for the objectClasses attribute.	Check the schema ldif file or the schema modification request.
8194	Error	Replication session aborted for agreement *agreement_name* because consumer replica is disabled.	The consumer has returned a disabled error, that is, it is not in a state in which it can receive replication updates.	Enable the consumer replica. It may also be necessary to reinitialize the consumer.
8195	Error	Pending changes: error *value*.	Looping through the changelog failed.	Ensure that replication is working correctly (using the `insync` utility and checking the replication agreement object). Check the error code in the error log for more information.
8196	Error	Bad Window size value for agreement *agreement_name*.	The value of the `ds5ReplicaTransportWindowSize` attribute is invalid.	Check the `dse.ldif` file or the LDAP entry defining the Replication Agreement. Check the modification operation attempted on the replication agreement.
8197	Error	Bad Group size value for agreement *agreement_name*.	The value of the `ds5ReplicaTransportGroupSize` attribute is invalid.	Check the `dse.ldif` file or the LDAP entry defining the Replication Agreement. Check the modifications attempted on the replication agreement.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8198	Error	Bad Compression Level value for agreement *agreement_name*.	The value of the `ds5ReplicaTransportCompressionLevel` attribute is invalid.	Check the `dse.ldif` file or the LDAP entry defining the Replication Agreement. Check the modifications attempted on the replication agreement.
8199	Error	Modification of *attribute_name* attribute is not allowed - agreement *agreement_name*.	The user is not permitted to modify the specified replication agreement attribute.	Check the `dse.ldif` file or the LDAP entry defining the Replication Agreement. Check the modifications attempted on the replication agreement.
8200	Error	Failed to update flag to force 5.1 Replication protocol for agreement *agreement_name*.	The replication agreement is being stopped.	Wait until the agreement has been stopped and retry.
8202	Error	Unknown replication agreement	A replication agreement with the specified DN could not be found.	Check the specified DN and all replication agreements. Check that the error is not in the client application.
8204	Error	Refusing to update partial replication checksum for agreement *agreement_name* permission denied.	The server received an update operation that is permitted for internal operations only.	Check the client that sent the forbidden update operation.
8212	Error	Failed to update replication schedule for agreement *agreement_name*.	1. The replication schedule format is invalid. 2. The replication agreement is stopping.	1. Check the client application. 2. Wait until the agreement has stopped and try again.
8213	Error	Failed to update Partial Replication Configuration for agreement *agreement_name*. The agreement needs to be disabled first.	An attempt was made to change the configuration for partial replication, on an enabled replication agreement	To change the partial replication configuration, disable the replication agreement first.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8215	Error	Partial replication not started for agreement *agreement_name*.	Partial replication has not been started.	Check the configuration of this replication agreement (specifically partial configuration entries). Start the partial replication feature for this agreement in the console.
8216	Error	Partial replication pointed to by this *entry* has been modified. Please update the current configuration on this supplier or re-initialize consumer accordingly.	The partial replication configuration has been modified.	Update the current configuration on the supplier, or reinitialize the consumer.
8218	Error	Replication protocol v5.0 not supported for *consumer*.	The latest replication protocol (v5.0) is not supported for this consumer.	Check the version of Directory Server running on the specified consumer.
8219	Error	Could not parse update vector for replica *replica_name*. The replica must be reinitialized.	The server was unable to parse the update vector for the specified replica.	Check that the consumer sent the replica update vector (RUV) during the start request.
8220	Error	Too much time skew between replicas for [*consumer*:*port*]	The time difference between the specified replicas is too great for replication to work correctly.	Ensure that the supplier and consumer machines have the same time and date. The use of the Network Time Protocol (NTP) is recommended.
8221	Error	Failed and requires administrator action.	A fatal error occurred during an incremental update. Replication on this consumer will be disabled.	Check the error log on the consumer for more information. Restart replication by updating the replication agreement and reinitializing updates.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8225	Error	Replica_write_partial_ repl_checksum: failed to update partial repl checksum with value *value* for replica *replica*. LDAP error.	An error occurred while writing an attribute value in the replica entry. Although harmless while the server is up and running, this error may lead to a replication malfunction the next time the server is restarted. The error occurs when the value of an important replication configuration attribute cannot be stored persistently in the `dse.ldif` file.	Stop the server inmediately and check the `cn=replica` entry for this suffix (in the `dse.ldif` file.) If the attribute `dsfilterspconfig checksum` is present in the entry, set its value to the value included in the error log. If the attribute `dsfilterspconfig checksum` is not present in the entry, add it and set its value to the value included in the error log. Restart the server.
8226	Error	replica_write_last_init_ time: failed to update last init timestamp with value *value* for replica *replica*. LDAP error.	An error occurred while writing an attribute value in the replica entry. Although harmless while the server is up and running, this error may lead to a replication malfunction the next time the server is restarted. The error occurs when the value of an important replication configuration attribute cannot be stored persistently in the `dse.ldif` file.	Stop the server inmediately and check the `cn=replica` entry for this suffix (in the `dse.ldif` file.) If the attribute `lastInitTimeStamp` is present in the entry, set its value to the value included in the error log. If the attribute `lastInitTimeStamp` is not present in the entry, add it and set its value to the value included in the error log. Restart the server.
8227	Error	Unable to read user schema.	The server was unable to access to its own internal schema entry.	Stop and restart the server. If this does not solve the problem, contact Sun ONE Technical Support.
8228	Error	Bind error for agreement: *.agreement*.	A replication protocol bind error has occurred.	Check that the consumer is up and running.
8229	Error	Failed to start a total update session.	The server was unable to start a total replication update session.	Check that the consumer is up and running.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8230	Error	Failed to create directory for changelog *changelog* error *error*.	The pathname is invalid, or there is unsufficient access to create the changelog directory.	Check that the path is valid and that there are sufficient access rights to create the directory.
8232	Error	Removal of changelog file *filename* failed.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8234	Error	Changelog is not initialized.	The changelog is not initialized, or an attempt has been made to configure the changelog cleanup parameters, when the changelog service is not started.	Ensure that the changelog service has been enabled.
8236	Error	Failed to open changelog.	This is probably due to a database or file access problem.	Enable the replication logs and retry the operation to see if additional reasons are output to the error log.
8241	Error	Change record has an invalid data version	A change record in the database has an invalid version number.	1. Disable and re-enable replication for this database. 2. Reinitialize the server. 3. Contact Sun ONE Technical Support.
8242	Error	Change record has an invalid operation type.	There is an invalid change record in the changelog.	Ordinarily, this error should not occur. If it does, the changelog is likely to be corrupted. In this case, reset the changelog for this database by reloading the data or disabling/enabling replication. If this does not solve the problem, contact Sun ONE Technical Support.
8243	Error	Failed to begin transaction for trimming DB error.	A database error occurred while the transaction was starting. This is likely to be a resource problem.	Check DB error and take action based on the error code. In other words, refer to the database errors guide.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8244	Error	Failed to abort transaction for trimming DB error.	A database error occurred while the transaction was being aborted. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8245	Error	Failed to commit transaction for trimming DB error.	A database error occurred while the transaction was being committed. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8246	Error	Failed to begin transaction for writing changelog *changelog* RUV DB error.	A database error occurred while the transaction was starting. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8247	Error	Failed to abort transaction for writing changelog *changelog* RUV DB error.	A database error occurred. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8248	Error	Failed to commit transaction for writing changelog *changelog* RUV DB error.	A database error occurred while the transaction was being aborted. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8249	Error	Writing the changelog *changelog* RUV in the file *filename* failed DB error.	A database error occurred while the transaction was being committed. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8250	Error	Failed to begin transaction for writing change count entry DB error.	A database error occurred. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8251	Error	Failed to abort transaction for writing change count entry DB error.	A database error occurred. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8252	Error	Failed to commit transaction for writing change count entry DB error.	A database error occurred. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8253	Error	Failed to write change count entry to the file *filename* DB error.	A database error occurred. This is likely to be a resource problem.	Check the corresponding database error code, and take action according to the database problem.
8256	Error	Failed to begin transaction for writing change operation DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8257	Error	Failed to abort transaction for writing change operation DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8258	Error	Failed to commit transaction for writing change operation DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8259	Error	Failed to write change operation with CSN *number*. DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8260	Error	Failed to create cursor for retrieving first change DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8261	Error	Failed to retrieve first change DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8262	Error	Failed to retrieve the next change DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8263	Error	Failed to delete the current change DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8264	Error	Failed to position in db at CSN *number*. DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8266	Error	Failed to open changelog file for replica *replica*. DB error.	An internal database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8267	Error	Failed to retrieve change count from changelog for replica *replica*.	The server was unable to retrieve the number of entries in the changelog.	Enable replication logging and check the specific replication error code for more information.
8268	Error	Failed to close changelog file *filename*. DB error.	A database error occurred.	Check the corresponding database error code, and take action according to the database problem.
8271	Error	Consumer replica *replica_name* has an invalid RUV.	The RUV returned by the consumer could not be parsed or caused a problem.	Check the consumer configuration. It may be necessary to reinitialize the consumer.
8272	Error	Replication session aborted for agreement *agreement_name* because consumer replica is disabled.	The consumer returned a disabled error, that is, it is not in a state to receive replication updates.	Enable the consumer replica. It may also be necessary to reinitialize the consumer.
8276	Error	Failed to start Replication Session for suffix *suffix_name*.	The replica is still being configured. The replication session cannot be accepted yet.	Wait until the configuration is complete and restart replication on the supplier.
8277	Error	Failed to start Replication Session for suffix *suffix_name*.	The replication session cannot be accepted because no replica has been defined for the suffix.	Check that the supplier replication agreement is correct. Enable replication on the consumer.
8278	Error	Failed to start Replication Session for suffix *suffix_name*.	The consumer is configured as a legacy replica and can therefore not accept multimaster replication.	Correct the replication topology.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8279	Error	Failed to start Replication Session for suffix *suffix_name*.	The consumer is denying the right to replicate	Check that the replication identity is properly defined and matches the one that the supplier is using.
8281	Error	Failed to start Replication Session for suffix *suffix_name*.	The consumer is not yet initialized and can therefore not accept changes.	Initialize the consumer, either online or offline.
8282	Error	Failed to start Replication Session for suffix *suffix_name*.	The consumer appears to have the same replicaId as the supplier (both are masters).	Disable and re-enable replication, providing a different ReplicaID for one of the servers.
8283	Error	Failed to start Replication Session for suffix *suffix_name*.	The consumer replica is already busy with a replication session.	Wait and try later. If this error persists, restart the server.
8284	Error	Failed to start Replication Session for suffix *suffix_name*.	The consumer server is a master and can therefore not accept a partial replica.	Make the consumer a read-only server, or unconfigure partial replication in the replication agreement.
8287	Error	Bad Group Packet size value for agreement *agreement_name*.	The value of the attribute `ds5ReplicaTransportGrpPktSize` is invalid.	Check the `dse.ldif` file or the LDAP entry defining the replication agreement. Check the modifications attempted on the replication agreement.
8288	Error	Bad Concurrency Level value for agreement *agreement_name*.	Value of attribute `ds5ReplicaTransportConcurrencyLevel` is invalid.	Check the `dse.ldif` file or the LDAP entry defining the replication agreement. Check the modifications attempted on the replication agreement.
8292	Error	Total update of a consumer *consumer* with an empty database is not allowed.	Consumer initialization has been requested but the supplier database is empty.	Load data onto the supplier before attempting to initialize the consumer with that supplier.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8293	Error	A fatal problem occurred on the consumer side: *consumer* with error *error*.	A fatal problem has occurred on the remote consumer.	Check the error log on the consumer for more information. Once the problem has been solved, you will need to update the replication agreement and reinitiate updates.
8294	Error	_cl5TrimFile: Removing changelog file *filename* as it belongs to an unexisting replica.	The changelog file contains data changes from a replica that has been unconfigured.	No action is necessary - this is an informational message.
8302	Error	Decoding replicate entry failed.	A protocol error occurred. The entry was incorrectly encoded.	Check the error code and contact Sun ONE Technical Support.
8303	Error	Failed with error code *error*.	Schema replication failed locally on the consumer.	Check error code and contact Sun ONE Technical Support.
8307	Error	Failed to import database entry.	An internal error occurred while adding an entry to the import queue, or while acknowledging the entry to the supplier.	Check the error log for a disk space problem and reinitialize the database. If the problem persists, contact Sun ONE Technical Support.
8308	Error	Invalid change_operation: entry_UUID *entry* CSN *CSN_value*.	A badly formed change was received.	Contact Sun ONE Technical Support.
8311	Error	Unexpected operation sequence number *value* (expecting *value*).	An internal error occurred in the sequencing of replicated operations.	Contact Sun ONE Technical Support.
8312	Error	Replay of pending changes failed returning.	The replicated change could not be applied on this consumer.	Check the error code. A delete operation may generate a return code of 32 - this error code is harmless (a dependency of changes between several masters). If the error persists, contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
8318	Error	[S] Bind failed with response: *error_code*.	Authentication failed. This may be due to an invalid host:port, an invalid identity, or the fact that the consumer is down.	Check the error code and fix the replication agreement. It may be necessary to restart the consumer.
8319	Error	[S] Start Failed with response: *error_code*.	Replication was unable to start. This is likely to be caused by an error in the replication configuration.	Check the error log for more information. Also check the error logs on the consumers.
8320	Error	[S] End Failed with response: *error_code*.	Replication was unable to end. This may be because a network outage has occurred, the consumer is down, or the consumer has already dropped the connection.	Check the error log for more information. Also check the error logs on the consumers.
12289	Error	PR_Accept() failed error variable (variable)	The TCP port to which you are attempting to bind is already in use.	1. Restart the server, using a different port. 2. Stop the application bound to that port and restart the server.
12290	Error	PR_GetIPNodeByName( ) failed errno variable (variable)	There is an error in the naming service configuration.	Add `listen host` (variable) to the naming service.
12291	Error	No port to listen on.	The LDAP port is missing from the configuration.	Add an LDAP port to the configuration file or use the command line.
12292	Error	Unable to create time thread (variable - variable) - shutting down.	System error, probably due to a resource problem.	Free up resources on the machine and restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
12293	Error	Too many open file descriptors - not listening on new connection.q	There is an error in the configuration file. See the `reservedfd` attribute.	Increase the maximum number of file descriptors (in the configuration file) by increasing the value of `nsslapd-maxdescriptors` .Otherwise, check the Directory configuration and reduce the resource usage (number of threads, and number of backends, for example.)
12294	Error	Not enough descriptors to accept any additional connections.	There are insufficient file descriptors to accept new connections. This may be because: 1. the value of the `maxdescriptors` attribute is too small 2. the hard limit on descriptors is too small 3. the value of the `reservedescriptors` attribute is too large	Increase the number of file descriptors available to the `slapd` process. The error log displays the number of file descriptors currently available to the `slapd` process, and the number of descriptors reserved for internal `slapd` use. The total number of file descriptors available to the process must be greater than variable
12295	Error	Cannot initialize lock. The server is terminating	Probably due to a resource problem on the system.	Restart the directory server.
12296	Error	Cannot create lock. The server is terminating.	Probably due to a resource problem on the system.	Restart the directory server.
12297	Error	Cannot create condvar. The server is terminating.	Probably due to a resource problem on the system.	Restart the directory server.
12298	Error	PR_SetNetAddr(PR_IpAddrAny) failed errno	Internal error.	Contact Sun ONE Technical Support.
12299	Error	PR_EnumerateHostEnt() failed.	There is an error in the naming service configuration.	Add the `listen host` variable to the naming service. See your operating system documentation for more information.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
12300	Error	gethostname *host* failed error *error* (variable).	There is an error in the naming service configuration.	Add the `listen host` variable to the naming service. See your operating system documentation for more information.
12301	Error	NSS Initialization failed.	The server was unable to initialize the security library.	Contact Sun ONE Technical Support.
12302	Error	Shutting down due to possible conflicts with other `slapd` processes.	More than one directory server is running.	Stop the directory servers that should not be running.
12304	Error	Shutting down due to inability to find user in system account database.	The server was unable to locate the specified user in the system account database.	Add the user to the system account database and restart the server.
12308	Error	ber encoding failed.	This is an internal error, most likely to be related to a memory allocation problem.	Increase the virtual memory of the machine and restart the directory server.
12318	Error	Call to _base64Decode fails.	An error occurred during the base64 encoding of a value. This is an internal error with no specific cause. It may be due to a resource problem.	Report the error to your administrator.
12319	Error	connection_push_back_ data has failed.	The request has been aborted due to an internal error.	Please contact Sun ONE Technical Support.
12320	Error	Invalid arguments: entry.	Configuration error. The server failed to obtain the frontend configuration entry.	Correct the frontend configuration entry and restart the server.
12321	Error	Failure during frontend sanity check.	Configuration error. The server failed the frontend sanity check.	Correct the frontend declaration and restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
12322	Error	Start parse of DSML operation fails, operation aborted.	Internal error occurred during the call to DsmlParser_startParse. This error has no specific cause but may be related to a resource problem.	Report the error to your administrator.
12323	Error	Could not store worker context in Batch operation.	This is an internal error with no specific cause. It may be related to a resource problem.	Report the error to your administrator.
12324	Error	Can't register HTTP port *port*.	Internal error. The server failed to register the HTTP port.	Check that the specified port is not currently in use and restart the server.
12325	Error	Can't register HTTPS port *port*.	Internal error. The server failed to register the HTTPS port.	Check that the specified port is not currently in use and restart the server.
12326	Error	Max size *value* of parser pool is lower than current size *value*.	Configuration error: the maximum size of the parser pool is lower than the current size.	In the `dse.ldif` file, check that the value of the `ds-hdsml-poolsize` attribute is lower than the value of the `ds-hdsml-maxpoolsize` attribute.
12327	Error	Cannot create XMLCh to UTF8 Transcoder.	An error occurred while trying to create an instance of a UTF8 transcoder. This is an internal error with no specific cause. It may be related to a resource problem.	Report the error to your administrator.
12328	Error	Can't initialize DSML Worker.	Internal error. The server failed during the initialization of the DSML worker.	Please contact Sun ONE Technical Support.
12329	Error	Extra datacopy failed.	A request has not been processed due to a connection closure.	Check the connection and retry.
12330	Error	Operation Key creation for HTTP context failed.	An internal memory management error has occurred.	Please contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
12332	Error	HTTP / DSML frontend initialization failed.	Initialization error. The server failed to set the plug-in functions.	Correct the frontend configuration and restart the server.
12333	Error	HTTP frontend instance creation failed.	Internal error. The server failed to instantiate the frontend plug-in.	Please contact Sun ONE Technical Support.
12334	Error	Unknown internal error has been raised.	Unknown internal error.	Please contact Sun ONE Technical Support.
12335	Error	Error with config attribute *attribute*.	Configuration error. A configuration attribute is invalid.	Correct the specified attribute and restart the server.
12336	Error	Invalid attribute syntax.	Configuration error. The syntax of a configuration attribute is invalid.	Correct the syntax of the specified attribute and restart the server.
12337	Error	System I / O error.	Internal I / O error.	Please contact Sun ONE Technical Support.
12338	Error	Memory allocation error.	System error, probably due to insufficient resources (lack of memory).	Please contact Sun ONE Technical Support.
12339	Error	Memory usage error.	Memory management system error.	Please contact Sun ONE Technical Support.
12340	Error	DSML schema location is not defined.	Configuration error: DSML schema location is not defined. Under normal circumstances, the default value of the DSML schema location is hardcoded. However, this default value can be overridden in the `dse.ldif` file.	Correct the value of the ds-hdsml-schemalocation attribute in the `dse.ldif` file, or remove this attribute from the file.
12341	Error	DSML schema URN is not defined.	Configuration error: DSML schema URN is not defined. Under normal circumstances, the default value of the DSML schema URN is hardcoded. However, this default value can be overridden in the `dse.ldif` file.	Correct the value of the `ds-hdsml-urn` attribute in the `dse.ldif` file, or remove this attribute from the file.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
12342	Error	SOAP schema location is not defined.	Configuration error. Under normal circumstances, the default value of the SOAP schema location is hardcoded. If this error occurs, there is an internal problem.	Report the error to your administrator.
12343	Error	SOAP schema URN is not defined.	Configuration error. Under normal circumstances, the default value of the SOAP schema URN is hardcoded. If this error occurs, there is an internal problem.	Report the error to your administrator.
12344	Error	Lock for concurrent access to _freeList does not exist.	Internal error: a lock for concurrent access to the specified list is missing. The lock should have been defined previously.	Report the error to your administrator.
12345	Error	No more parser in the pool, operation aborted.	Internal error that occurs when the pool of parsers is empty and cannot be extended (all the parsers are in use).	Increase the value of the maximum pool size, specified by the `ds-hdsml-poolmaxsize` attribute in the `dse.ldif` file.
12346	Error	Bad Dsml request - *SOAP fault code.*	An error occurred during the call to `DsmlParser_getNextRequest`.	None - a SOAP fault is returned to the client with the reason for the failure.
12347	Error	Error with secure identity method.	Configuration error. The secure identity method configuration parameter is invalid.	Correct this parameter and restart the server. Possible values for the secure identity method parameter are: clientCertOnly clientCertFirst httpBasicOnly

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
12348	Error	Exception raised when calling XMLString::transcode.	An exception was raised when calling `XMLString::transcode`. This is an internal error with no specific cause. It may be due to a resource issue.	Report the error to your administrator.
12352	Error	Bad Dsml request - *SOAP error message*.	A SOAP / DSML error occurred during a call to DSMLParser_startParse.	None - a SOAP / DSML error message is returned to the client with the reason for the failure.
12353	Error	Parse of fake request fails *error*.	This error occurs when a bad request is submitted to the parser. It should not occur in the case of the valid fake request. The DSML / SOAP schema URN and / or location may be invalid.	Check the error log for more information. If the schema URN and / or location are invalid, check the following attributes in the dse.ldif file: ds-hdsml-dsmlurn ds-hdsml-dsmlschemalocation
12354	Error	Parse of fake request fails.	This error occurs when a bad request is submitted to the parser. It should not occur in the case of the valid fake request. Cause unknown.	Please contact Sun ONE Technical Support.
12355	Error	The XML schema file *filename* is missing.	Configuration error: an XML schema is missing.	Insert the missing schema in the specified location and restart the server.
12356	Error	SOAPAction header is missing.	The client must provide a SOAPAction header. If it is absent, the request is rejected.	Provide a SOAPAction header, the contents of which may be set to any value (including an empty value), for example: SOAPAction: SOAPAction: ““ SOAPAction: “batchRequest”
12362	Error	PR_Bind() on address *host* port *port* failed.	It is likely that the port number configured for this server requires that the server be run as root.	Restart the server using a port that does not required root access or start the server as a user with root access.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20490	Error	Database recovery process FAILED. The database is not recoverable.	Database recovery has failed.	This is a serious database error. Please contact Sun ONE Technical Support.
20492	Error	Failed to create thread (NSPR error).	The Netscape Portable Runtime (NSPR) was unable to create one or more threads. This may be due to insufficient resources.	1. Check that there is sufficient available memory and that a sufficient number of threads per process has been set up in the operating system configuration. 2. Check the error code that appears in the log against the NSPR error codes (see `http://www.myServer.org/projects/nspr/reference/html/prerr.html`).
20494	Error	Instance *instance_name* does not have the expected version *version_number*.	An attempt was made to open a database with a different database version. This is probably a migration issue.	Re-export the database from the old server and re-import it to the new server.
20499	Error	`dblayer_instance_start_fail`: backend *instance_name* has no IDs left. Database must be rebuilt.	The internal NEXTID counter has reached the limit.	Rebuild the database.
20501	Error	Serious failure in `dblayer_txn_begin`. Err=*value*.	The database has reported an error. If the printed value is positive, this is a system error. If the printed value is negative, the database has not been recognized or must be recovered.	This is a serious database error. Please contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20502	Error	Serious failure in `dblayer_txn_commit`. Err=*value*.	The database has reported an error. If the printed value is positive, this is a system error. If the printed value is negative, the database has not been recognized or must be recovered.	This is a serious database error. Please contact Sun ONE Technical Support
20503	Error	Serious failure in `dblayer_txn_abort`. Err=*value*.	The database has reported an error. If the printed value is positive, this is a system error. If the printed value is negative, the database has not been recognized or must be recovered.	This is a serious database error. Please contact Sun ONE Technical Support
20504	Error	Serious failure in deadlock detect (aborted at *address*). Err=*value*.	The database has reported an error. If the printed value is positive, this is a system error. If the printed value is negative, the database has not been recognized or must be recovered.	This is a serious database error. Please contact Sun ONE Technical Support
20505	Error	Serious failure during database checkpointing. Err=*value*.	The database has reported an error other than an inability to write pages to the disk immediately. If the printed value is positive, this is a system error. If the printed value is negative, the database has not been recognized or must be recovered.	This is a serious database error. Please contact Sun ONE Technical Support
20506	Error	Serious failure during trickle. Err=*value*.	The database has reported an error. If the printed value is positive, this is a system error. If the printed value is negative, the database has not been recognized or must be recovered.	This is a serious database error. Please contact Sun ONE Technical Support

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20507	Error	Failed to create guardian file. Database corruption possible.	This is a file system error. The server was unable to create the required guardian file.	Check that the user specified at installation has the appropriate permissions to write to the database directory.
20508	Error	Database database is corrupt and being marked unavailable. Either re-import or delete the database.	The database is corrupt. This is most likely to be the result of a previously aborted database import.	Reimport or delete the database.
20512	Error	Failed to write guardian file. Database corruption possible.	This is a file system error. The server was unable to write to or close the guardian file.	Check that the user specified at installation has the appropriate permissions to write to the database directory. Ensure that the file system is not full.
20513	Error	Failed to delete guardian file. Database corruption possible.	This is a file system error. The server was unable to delete the guardian file.	Check that the user specified at installation has the appropriate permissions to write to the database directory.
20737	Error	ldbm backend instance: nextid not initialized.	This is a software problem.	Please contact Sun ONE Technical Support.
20738	Error	ldbm backend instance: FATAL ERROR: backend name has no IDs left. DATABASE MUST BE REBUILT.	The limit for the database internal identifier has been reached. This is probably due to several adds and deletes being performed on the local database.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20739	Error	ldbm backend instance: WARNING: backend *backend_name* may run out of IDs.	The limit for the database internal identifier is close to being reached. This is probably due to several adds and deletes being performed on the local database	If the limit has been reached, rebuild the database, using `db2ldif`, then `ldif2db`.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20740	Error	Numsubordinates assertion failure.	The database is not coherent. There is a child entry that is unknown to the parent entry and the `numsubordinates` attribute is absent in the parent entry.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20745	Error	ldbm_back_seq : id2entry err *error*.	An entry could not be located during an ldbm_back_seq operation. The database is incoherent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20746	Error	ldbm_back_seq: could not open index file for attribute *attribute*.	An index file could not be located during an ldbm_back_seq operation. The database is incoherent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20747	Error	compare_entries db err *error_number* while loading entry *entry*.	Certain entries were deleted while the server was attempting to sort them. This is probably due to a VLV or SORT control in a search.	Create a VLV index to avoid “on the fly” sorting.
20748	Error	start : Resource limit registration failed.	The local database could not be started because the limit subsystem did not allow it to register.	Check the resource limit configuration and restart the server.
20749	Error	start : Failed to init database err=*error*.	The local database could not be started because the underlying database component did not start.	Check that the database configuration is correct, and that there is enough disk space available.
20750	Error	start : Failed to start databases err=*error*.	The local database instances could not be started.	Check that the database configuration is correct, and that there is enough disk space available.
20751	Error	Database version mismatch (expecting *version* but found *version* in directory *directory*.)	The binary code for one version of Directory Server was started on a database with a different version.	Check the versions and ensure that the same binary and database versions are used.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20752	Error	VLV : can't get index file *file* (err *error*).	The server could not locate the file used for the virtual list view (VLV) index during an update. The database is inconsistent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20753	Error	`vlv_build_idl`: can't follow db cursor (err *error*).	The database is incoherent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20754	Error	`nomem`: wants *value* key *value* data.	The system is out of memory	Check the configuration.
20755	Error	VLV : can't get index file *file* (err *error*).	The server could not locate the file used for virtual list view (VLV) indexes. The database is inconsistent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20756	Error	VLV : couldn't get cursor (err *error*).	The server could not locate a cursor used for virtual list view (VLV) indexes. The database is inconsistent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20757	Error	vlv_filter_candidates: Candidate *id* not found err=*error*.	The server could not locate an entry that is present in the virtual list view (VLV) index. The database is inconsistent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20758	Error	vlv_trim_candidates_ byvalue: Candidate ID *id* not found err *error*.	The server could not locate an entry that is referenced in a virtual list view (VLV) index. The database is inconsistent.	Rebuild the database, using `db2ldif`, then `ldif2db`.
20759	Error	vlv find index: err *error*.	The server could not locate an index used in virtual list view (VLV).	Check the VLV configuration.
20760	Error	Couldn't generate valid filename from Virtual List View Index Name name. Need some alphabetical characters.	An LDAP client attempted to create a virtual list view (VLV) index with an invalid name. This should not harm the directory server.	Change the LDAP client so that it uses a valid name.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20761	Error	Add: maximum ID reached cannot add entry to backend *backend*.	The limit for the database internal identifier has been reached. This is probably because several adds and deletes have been performed on the local database.	Regenerate the database using `ldif2db` and `db2ldif`.
20762	Error	Add: attempt to index *entry* failed.	The server was unable to index the entry being added.	Check the previous errors in the log for additional information.
20763	Error	Retry count exceeded in add.	The acceptable number of add retry counts was exceeded without success. Another operation may be ongoing, resulting in a conflict when trying to access that part of the database.	Wait until other operations have ended and retry the add operation.
20764	Error	Line *line_number*: Fatal Error: Failed to initialize attribute structuring.	The server was unable to initialize the attribute structure. This is probably a memory error.	Check the available memory.
20765	Error	Attempt to delete a non-tombstone entry *entry*.	An attempt was made to delete an entry that was not a tombstone entry.	Please contact Sun ONE Technical Support.
20766	Error	Attempt to tombstone again a tombstone entry *entry*.	An attempt was made to tombstone an entry that is already a tombstone entry.	Please contact Sun ONE Technical Support.
20768	Error	Retry count exceeded in delete.	The acceptable number of delete retry counts was exceeded without success. Another operation may be ongoing, resulting in a conflict when trying to access that part of the database.	Wait until other operations have ended and retry the delete operation.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20772	Error	Retry count exceeded in modify.	The acceptable number of modify retry counts was exceeded without success. Another operation may be ongoing, resulting in a conflict when trying to access that part of the database.	Wait until other operations have ended and retry the modify operation.
20773	Error	Retry count exceeded in modrdn.	The acceptable number of modrdn retry counts was exceeded without success. Another operation may be ongoing, resulting in a conflict when trying to access that part of the database.	Wait until other operations have ended and retry the modrdn operation.
20774	Error	modrdn: could not add new value to index err=*error*	The server was unable to add a new value to the index.	Check the error log for more information and contact Sun ONE Technical Support.
20775	Error	Database error *error*.	A database error occurred while trying to build the list of possible candidate entries. The index files may be corrupt.	Re-index and try again.
20776	Error	Null referral in *entry*.	The candidate entry has a NULL referral.	Update the referral in the entry or remove the `ref` attribute.
20777	Error	Filter bypass error on entry *entry*.	The server failed to bypass the filter test.	Please contact Sun ONE Technical Support.
20778	Error	Unable to add config entries to the DSE.	The server was unable to add configuration entries to the DSE.	Ensure that there is no inconsistency within the entries.
20779	Error	ERROR: ldbm plugin unable to read cn=config.	The configuration information under `cn=config` could not be read.	Please contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20780	Error	ERROR: ldbm plugin unable to read attribute nsslapd-instancedir from cn=config.	The `nsslapd-instancedir` attribute under `cn=config` could not be read. The attribute may be missing.	Ensure that the `nsslapd-instancedir` attribute is present and has an appropriate value.
20786	Error	Invalid value for *attribute*. Must be between 0 and 100.	An invalid value was provided for the `nsslapd-db-trickle-percentage` attribute. The value should be between 0 and 100.	Check and correct the value provided for the `nsslapd-db-trickle-percentage` attribute
20787	Error	*Attribute* can't be modified while the server is running.	An attempt was made to modify a configuration attribute while the server was running. This attribute cannot be changed online.	Stop the server before modifying the attribute.
20788	Error	Value *value* for attribute *attribute* is not a number.	The attribute value must be numerical.	Ensure that the attribute has a numerical value.
20789	Error	Value *value* for attribute *attribute* is greater than the maximum *value*.	The value specified for the attribute is greater than the maximum permitted.	Ensure that the attribute value is smaller than or equal to the maximum value.
20790	Error	Value *value* for attribute *attribute* is less than the minimum *value*.	The value specified for the attribute is smaller than the minimum permitted.	Ensure that the attribute value is greater than or equal to the minimum value.
20791	Error	Value *value* for attribute *attribute* is outside the range of representable values.	The value specified for the attribute is outside the permissible range.	Ensure that the attribute value is within the representable range.
20792	Error	Could not set instance config attr *attribute* to *value*.	The server failed to set the instance configuration attribute.	Ensure that both the syntax and the value of the attribute are correct.
20793	Error	Could not retrieve ldbm config info from DSE.	The server was unable to access the ldbm configuration in the DSE.	Check that the `dse.ldif` file has not been corrupted and restart the server.
20795	Error	ldbm: instance instance does not exist!	The specified instance was not found because no such instance exists.	Verify that the instance name is correct and corresponds to an existing instance.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
20796	Error	ldbm: instance is in the middle of a task. Cancel the task or wait for it to finish then try again.	The specified instance is currently processing a task.	Cancel the current task or wait for it to finish and retry.
20797	Error	ldbm: modify attempted to change the root suffix of a backend (which is not allowed).	An attempt was made to change the suffix associated with an ldbm database.	Do not modify the `nsslapd-suffix` attribute of an existing instance.
20806	Error	System info mismatch (expecting *variable* but found *variable* in directory *directory_name*).	The system information from the backend's `DBVERSION` file did not match the server information.	Edit the backend's `DBVERSION` file to match the server information.
20807	Error	Failed to read server system information	The server was unable to obtain the system information. This is possibly a permissions or NSPR compilation issue.	Check that the user specified at installation has the appropriate permissions.
20994	Error	Disk full under *variable*.	The available space on a disk used by the Directory Server has dropped below the value of the `disk-full-threshold` attribute.	Increase the available disk space.
20996	Error	Cannot parse entry from database for id *id* string =*variable*.	Database corruption.	Restore the database from a backup.
20997	Error	Inconsistent database: entrydn for *entry* refers to id *id* missing from `id2entry`.	Database corruption.	Restore the database from a backup.
21005	Error	Could not open index *index* for update.	An attribute index is configured but the corresponding database index file could not be opened.	Check whether the file exists and/or rebuild it using `db2index`.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
21006	Error	Could not open index *index* for range query.	An attribute index has been configured but the corresponding database index file could not be opened.	Check whether the file exists and / or rebuild it using `db2index`.
21008	Error	Backend initialization failed: could not allocate a lock.	Insufficient system resources.	Check the available memory.
21009	Error	Backend initialization failed: could not allocate a condition variable.	Insufficient system resources.	Check the available memory.
21010	Error	Backend initialization failed: could not set plugin functions.	Insufficient system resources.	Check the available memory.
21011	Error	Backend initialization failed on instance *instance*: could not allocate a lock.	Insufficient system resources.	Check the available memory.
21012	Error	Backend initialization failed on instance *instance*: could not allocate a condition variable.	Insufficient system resources.	Check the available memory.
21016	Error	Failed to create ancestorid index.	An index could not be created on the disk.	Check the error log for previous messages that should isolate the problem.
21017	Error	Incomplete parentid index suspected (*value* extra keys in ancestorid)	Database corruption.	Rebuild the parentid index or restore the database from a backup.
21018	Error	Entry cache initialization failed: could not allocate lock.	Insufficient system resources.	Check the system free memory.
21022	Error	*variable* is configured to use more than the available physical memory.	The cachesize as defined in the configuration file exceeds database limits.	Lower the value of the `cachesize` attribute in the configuration file.
21023	Error	Index *index* is inconsistent.	Database corruption.	Rebuild the affected index or restore the database from a backup.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
21249	Error	Failed to encrypt some attribute inside the entry *entry* before writing it to the database.	The server was unable to encrypt the specified attribute inside the entry.	Check the attribute encryption configuration.
21250	Error	Failed to decrypt some attribute inside the entry *entry* when when reading it from the database.	The server was unable to decrypt the specified attribute inside the entry.	Check the attribute encryption configuration.
21251	Error	Encrypted value's prefix doesn't match the corresponding algorithm *algorithm* in the attribute encryption configuration.	The value is already encrypted or does not match the algorithm specified in the configuration.	Check that the attribute encryption configuration is correct.
21252	Error	Server didn't find plug-in for algorithm *algorithm*.	The server was unable to locate the plug-in for the specified algorithm.	Enable the encryption plug-in.
21253	Error	Failed to encrypt index keys.	The server was unable to encrypt the specified values.	Check that the values are not already encrypted and that the cipher with which they are being encrypted match the configuration settings.
21254	Error	Attribute encryption: failed to *encrypt* / *decrypt* attribute *attribute* with algorithm *algorithm*.	The server was unable to encrypt / decrypt the attribute's values. The attribute may already be encrypted with an incorrect algorithm or the algorithm plug-in may be missing.	Check for inconsistencies in the attribute encryption configuration.
21255	Error	Encryption plugin (*plugin*): failed to encrypt.	An error occurred during the plug-in's encryption function.	Check the plug-in traces. Ensure that the plug-in itself has not been corrupted.
21256	Error	Encryption plugin (*plugin*): failed to decrypt.	An error occurred during the plug-in's decryption function.	Check the plug-in traces. Ensure that the plug-in itself has not been corrupted.
24577	Error	Bulk import process failed: state=*state*, error code=*error*.	The bulk import has been aborted.	Ensure that the bulk import is started or previously suspended before attempting an update or restart.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
28673	Error	filter_sp_replace_or_add _checksum: failed to update *attribute* attribute from *entry* entry; LDAP error - *errnum*.	The attribute `filterspconfchecksum` could not be updated with a new value.	1. Check whether the attribute already exists in the entry. 2. Check whether the attribute is present in the `dse.ldif` file.
32769	Error	Unable to allocate memory. Cannot start Roles plugin.	There is not enough memory to register the roles plug-in into the service provider broker.	Restart the server.
32770	Error	Unable to allocate memory. Cannot start Roles plugin.	There is not enough memory to register the `nsrole` attribute.	Restart the server.
32771	Error	Unable to allocate memory. Cannot create Roles cache.	This error indicates a resource problem on the machine.	Restart the server.
32772	Error	Lock creation failed. Cannot create Roles cache.	This error indicates a resource problem on the machine.	Restart the server.
32773	Error	Conditional variable creation failed. Cannot create Roles cache.	This error indicates a resource problem on the machine.	Restart the server.
32774	Error	Thread creation failed. Cannot create Roles cache.	This error indicates a resource problem on the machine.	Restart the server.
32775	Error	Failed to get objectclass from *entry*.	The specified entry does not contain an objectclass.	Check the entry and add the required objectclass.
32776	Error	Unsupported operation *operation*.	An unknown operation has been performed on the server and is triggering a role cache update.	Check that the specified operation is valid.
32778	Error	Maximum number of nested roles exceeded (max *value* current *value*). Not retrieving roles from entry *entry*. Probable circular definition.	The maximum number of nested roles has been exceeded. This is probably due to a circular role definition.	Check the role definitions. The maximum number of nested roles permitted is defined by `MAX_NESTED_ROLES`.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
32779	Error	Nested role *entry* does not exist.	The entry corresponding to the DN does not exist.	Check the role definition.
32780	Error	Cannot initialize Roles plugin.	The server is unable to update the pblock parameters.	Restart the server.
32781	Error	Unknown role type *type*.	The role type is unknown. Valid role types are : `managed`, `filtered`, or `nested`.	Check the role definition and amend the type as necessary.
33025	Error	Could not allocate PB.	Internal error, probably due to insufficient available memory.	Free up some memory. If the error continues, please contact Sun ONE Technical Support.
33026	Error	Internal PBG error.	Internal error.	Please contact Sun ONE Technical Support.
33027	Error	Internal search error in Attribute Uniqueness plugin.	Internal error.	Please contact Sun ONE Technical Support.
33028	Error	Internal PB error.	Internal error.	Please contact Sun ONE Technical Support.
33029	Error	Could not find plugin argument number.	Memory corruption or invalid configuration.	Check the plug-in configuration. If it is valid, please contact Sun ONE Technical Support.
33030	Error	Could not find plugin arguments.	Memory corruption or invalid configuration.	Check the plug-in configuration. If it is valid, please contact Sun ONE Technical Support.
33031	Error	Could not find a valid argument.	Configuration error.	Check the plug-in configuration parameters in the `dse.ldif` file. Make sure that the syntax and values are correct.
33032	Error	ADD / MOD / MODRDN: unable to get replication flag.	Internal error.	Please contact Sun ONE Technical Support.
33033	Error	ADD / MOD / MODRDN: unable to get target DN.	Internal error.	Please contact Sun ONE Technical Support.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
33034	Error	Unable to get entry data.	Internal error.	Please contact Sun ONE Technical Support.
33035	Error	Could not get MODIFY data.	Internal error.	Please contact Sun ONE Technical Support.
33036	Error	Error while retrieving mod values.	Internal error.	Please contact Sun ONE Technical Support.
33037	Error	Unable to get new superior DN.	The new superior DN does not exist.	Check the validity of the intended operation.
33038	Error	Unable to get new DN.	The new rdn is invalid or is not correctly specified.	Check the validity of the intended operation.
33039	Error	Unable to allocate a new entry.	Internal error.	Please contact Sun ONE Technical Support.
33040	Error	ADD parameter untagged: *error*.	Configuration error.	Check the plug-in configuration parameters in the `dse.ldif` file. Make sure that the syntax and values are correct.
33041	Error	ADD result *result*.	An error occurred during an internal search while performing an ADD operation.	Ensure that the database is not corrupt and contact Sun ONE Technical Support.
33042	Error	MODIFY result *result*.	An error occurred during an internal search while performing a MOD operation.	Ensure that the database is not corrupt and contact Sun ONE Technical Support.
33043	Error	MODRDN bad rdn value=*value*.	Internal error.	Please contact Sun ONE Technical Support.
33044	Error	MODRDN result *result*	An error occurred during an internal search while performing a MODRDN operation.	Ensure that the database is not corrupt and contact Sun ONE Technical Support.
33045	Error	NSUniqueAttr_Init Error: *error*	Configuration error.	Check the plug-in configuration parameters in the `dse.ldif` file.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
33793	Error	cos_cache_init: cannot create mutexes	The server was unable to allocate mutexes for the CoS plug-in. This is probably due to a memory problem.	Free up resources on the machine and restart the server.
33794	Error	cos_cache_init: cannot register as service provider	The server was unable to register a virtual attribute service provider.	Free up resources on the machine and restart the server.
33795	Error	cos_cache_init: PR_CreateThread failed	The server was unable to create a CoS thread.	Free up resources on the machine and restart the server.
33796	Error	cos_cache_create: failed to cache the schema	The server was unable to create the CoS schema cache.	1. Free up resources on the machine. 2. “Touch” a CoS definition to retrigger CoS cache building. 3. Restart the server.
33797	Error	cos_cache_create: failed to index cache	The server was unable to index the CoS cache.	1. Free up resources on the machine. 2. “Touch” a CoS definition to retrigger CoS cache building. 3. Restart the server.
33798	Error	COS memory allocation failure: variable	The server was unable to allocate memory for the CoS cache.	1. Free up resources on the machine. 2. “Touch” a CoS definition to retrigger CoS cache building. 3. Restart the server.
33799	Error	cos_cache_build_ definition_list: failed to find suffixes in the rootDSE.	The server was unable to read the suffix list from the rootDSE entry.	Restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
33801	Error	COS Definition error *error*	There is an error in the definition of the specified CoS.	Check and correct the CoS definition. Note that a definition cannot supply its own specifier. The DN of the CoS template may be incorrect.
33802	Error	cos_cache_add_dn_ tmpls: could not cache cos template *variable*	The server was unable to add the specified template to the CoS cache.	1. Free up resources on the machine. 2. "Touch" a CoS definition to retrigger CoS cache building. 3. Restart the server.
33803	Error	cos_cache_query_atr: failed to get entry dn	The server was unable to locate the dn of the target entry during a search operation. This error should not occur under normal circumstances.	1. Retry the search operation. 2. Restart the server.
33804	Error	COS failed to get objectclass from entry (*entry*)	The server was unable to locate the objectClass of the target entry during a search or update operation. This error should not occur under normal circumstances.	1. Retry the search or update operation. 2. Restart the server.
33806	Error	cos_start: failed to initialise	The server was unable to start the CoS plug-in. This is probably due to a memory problem.	1. Check the CoS plug-in configuration in the `dse.ldif` file. 2. Check the CoS definitions and templates. 3. Check the error log for a more specific error message. 4. Restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
33807	Error	cos_init: failed to register plugin	The server was unable to register the CoS plug-in. This is probably due to a memory problem.	1. Check the CoS plug-in configuration in the `dse.ldif` file. 2. Check the error log for a more specific error message. 3. Restart the server.
33808	Error	COS Definition error (no DN)	There is an error in the definition of the specified CoS.	Check and correct the CoS definition.
33809	Error	cos_cache_change_notify : failed to get dn of changed entry	The server was unable to obtain the dn of the target entry during an update operation. This error should not occur under normal circumstances.	1. Retry the update operation. 2. Restart the server.
34817	Error	ACL library initialization failed.	The server is unable to initialize the ACL plug-in. This is usually an indication of memory problems.	1. Check the ACL plug-in configuration in the `dse.ldif` file. 2. Check the error log for other, more specific error messages. 3. Restart the server.
34818	Error	ACL failed to allocate locks.	The server is unable to allocate mutex or reader / writer locks for the ACL plug-in at initialization time.	1. Check the OS configuration and increase the file descriptors limit, if possible. 2. Check the Directory Server configuration and reduce the resource usage.
34819	Error	ACL malloc fail: *error*.	The server is unable to allocate sufficient aclpb pool memory for the ACL plug-in.	Free up resources on the machine and restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
34820	Error	ACL internal error: *error*.	This is an internal error and should not occur under normal circumstances.	1. Attempt the LDAP operation again. 2. Restart the server. 3. Copy the errors log file and contact Sun ONE Technical Support.
34822	Error	Unable to initialize the plugin: *plugin_name*	The server us unable to allocate sufficient aclpb pool memory for the ACL plug-in.	Free up resources on the machine and restart the server.
34823	Error	Error: ACIs not deleted from *entry*.	The server was unable to remove the specified ACIs from the entry. See the error log for more information.	Attempt the modify operation again.
34824	Error	ACL internal init fail: *error*.	Initialization error. The server was unable to register the specified attributes with `libaccess`. See the error log for more information.	Verify the configuration and installation of the ACL plug-in.
34826	Error	ACL error adding aci: *aci*.	There is an error (possibly invalid ACI syntax) in the ACI attribute being updated.	Correct the error in the ACI and attempt the ACI update operation again.
34827	Error	ACL parsing error: *error*.	ACL parsing error for a macro ACI. See the log file for the exact cause of the error.	Correct the error in the ACI and attempt the ACI update operation again.
34828	Error	ACL parsing error: failed to make filter for string *string*.	ACL parsing error. The server was unable to construct an LDAP filter for the specified string.	Correct the error in the ACI and attempt the ACI update operation again.
34829	Error	ACL PARSE ERR(rv=*error_code*): *aci*.	ACL parsing error. See the log file for the exact cause of the error.	Correct the error in the ACI and attempt the ACI update operation again.
34830	Error	Can't add the rest of the acls for entry: *entry* after delete.	The server failed to update ACIs in the specified entry, when an ACI was deleted.	1. Attempt the update operation again. 2. Restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
34831	Error	ACL failed to allocate locks.	The server is unable to allocate mutex or reader/writer locks for the ACL plug-in at operation time.	1. Free up resources on the machine. 2. Attempt the LDAP operation again. 3. Restart the server.
34832	Error	Operation extension allocation failed.	The server is unable to get/create an operation extension structure at operation time.	1. Free up resources on the machine. 2. Attempt the LDAP operation again. 3. Restart the server.
34834	Error	acl_get_aclpb: Invalid aclpb type	An invalid ACL operation extension was found. This is an internal error and should not occur under normal circumstances	1. Attempt the LDAP operation again. 2. Restart the server. 3. Copy the errors log file and contact Sun ONE Technical Support.
34835	Error	ACLPB parameter *parameter* value *value* exceeded allowed value *value*.	This is an internal error and should not occur under normal circumstances.	1. Attempt the LDAP operation again. 2. Restart the server.
34838	Error	ACL parent[ ] exceeded the levels limit *max_limit*: *function*.	ACL parsing error: the parent keyword has been used with more than ten levels. Check the log file to see the type of ACI in which the keyword was used incorrectly.	Correct the error in the ACI and attempt the operation again.
36865	Error	collation_unlock: PR_ExitMonitor (*variable*)=*variable*; collation_monitor = *variable*	An error occurred while releasing the collation lock.	Restart the server.
36866	Error	collation_init: PR_NewMonitor failed	An error occurred while creating the collation lock.	Restart the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
36867	Error	*variable*: line *line_no*: missing directory name in directory *directory* (ignored)	No argument was provided for the NLS parameter.	Check the configuration variable.
36868	Error	*variable*: line *line_no* ignored: only variable arguments (expected collation language country variant strength decomposition oid...)	Insufficient arguments were provided for the collation parameter.	Check the configuration variable.
36869	Error	*variable*: line *line_no*: strength *value* not supported (will use 2)	An invalid value was specified for the collation strength.	Check the configuration variable.
36870	Error	*variable*: line *line_no*: decomposition *value* not supported (will use 2)	An invalid value was specified for the collation decomposition.	Check the configuration variable.
36871	Error	Too many tokens (max *max_tokens*)	Too many items have been specified on the configuration line.	Check the configuration variable.
36872	Error	Could not open config file *filename* - absolute path.	The server was unable to open the collation configuration file.	Check the path to the collation configuration file.
36873	Error	*variable*: line *line_no*: bad config line (ignored)	The server was unable to parse a line in the collation configuration file.	Check the collation configuration file.
36874	Error	Unable to retrieve slapd configuration pathname; using default.	The location of the collation configuration file was not provided to the plug-in.	Check the path to the collation configuration file.
37121	Error	Not enough pattern space.	The regular expression being constructed for the DN substring filter could not be stored in the memory allocated.	Check the DN substring filter being provided to the server.
37122	Error	re_comp *filter* failed.	The regular expression being constructed for the substring filter could not be compiled.	Check the substring filter being provided to the server.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
37123	Error	dn_assertion2keys_ava: unknown ftype.	A filter containing an unknown type was provided to the server.	Check the filter being provided to the server.
37377	Error	statechange_init: failed to register plugin.	The state change plug-in could not be registered with the server.	Restart the server.
37378	Error	statechange: failed to create lock.	The server was unable to create a mutex for the state change subsystem.	Restart the server.
37379	Error	statechange: failed to publish state change interface.	The server was unable to publish the interface to the state change plug-in API.	Restart the server.
37380	Error	statechange_post_op: failed to get dn of changed entry.	The server was unable to determine the DN of the modified entry.	Restart the server.
37633	Error	Only one pass through plugin instance can be used	An attempt was made to configure multiple instances of the passthrough authentication plug-in.	Check the pass-through authentication plug-in configuration.
37634	Error	No pass through servers found in configuration (at least one must be listed)	An attempt was made to use the passthrough authentication plug-in without specifying any remote servers.	Check the pass-through authentication plug-in configuration.
37635	Error	Server parameters should be in the form “maxconnections maxconcurrency timeout ldapversion connlifetime” (got “*error*”)	The set of parameters specified for the remote server was invalid.	Check the pass-through authentication plug-in configuration.
37636	Error	LDAP protocol version should be *version* or *version* (got *error*)	The LDAP version specified for the remote server was invalid.	Check the pass-through authentication plug-in configuration.
37637	Error	Maximum connections must be greater than zero (got *error*)	The maximum number of connections to the remote server is specified as less than or equal to zero.	Check the pass-through authentication plug-in configuration.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
37638	Error	Maximum concurrency must be greater than zero (got *error*)	The maximum concurrency is specified as less than or equal to zero.	Check the pass-through authentication plug-in configuration.
37639	Error	Unable to parse LDAP URL "*url*" (*error*)	An error occurred while parsing the LDAP URL.	Check the pass-through authentication plug-in configuration.
37640	Error	Missing suffix in LDAP URL "*url*"	The pass-through suffix was not specified in the LDAP URL.	Check the pass-through authentication plug-in configuration.
37641	Error	Unable to parse suffix string "*suffix*" within variable	An error occurred while splitting the list of suffixes for which authentication is to be passed through.	Check the pass-through authentication plug-in configuration.
37642	Error	Suffix "*suffix*" is handled by a database backend and therefore will not be subject to pass through authentication	One of the suffixes for which pass-through authentication is configured exists in the local directory.	Check the pass-through authentication plug-in configuration.
37644	Error	ldap_charray_add() failed when building suffix list	An error occurred while adding a suffix to the list of suffixes handled by backends in the server.	Restart the server.
37645	Error	No active suffixes found	No active suffixes could be located in the local server.	Check the server configuration and / or restart the server.
37646	Error	passthruauth_init failed	The pass-through authentication plug-in could not be registered.	Restart the server.
37647	Error	Unable to get arguments	The server was unable to locate the list of arguments to the pass-through authentication plug-in.	Check the pass-through authentication plug-in configuration.
37648	Error	configuration failed (variable)	The pass-through authentication plug-in could not be configured based on the arguments provided.	Check the pass-through authentication plug-in configuration.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
37649	Error	Operation not handled (unable to retrieve bind parameters)	The server was unable to determine the required information regarding the bind operation.	Check the bind request.
37650	Error	*error*	The server was unable to retrieve the set of controls associated with the bind request.	Check the bind request.
37651	Error	*error*	The server was unable to set the DN or authentication type associated with this connection.	Restart the server.
37889	Error	referint_postop_init failed	A failure occurred while registering the referential integrity plug-in.	Restart the server.
37890	Error	referint_postop_del: could not get parameters	The server was unable to retrieve the required information about a delete operation.	Check the delete request.
37891	Error	referint_postop failed to get argc	The server was unable to determine the number of parameters to the referential integrity plug-in.	Restart the server.
37892	Error	referint_postop failed to get argv	The server was unable to retrieve the parameters associated with the referential integrity plug-in.	Restart the server.
37893	Error	referint_postop_del args are NULL	No arguments were provided for the referential integrity plug-in.	Check the configuration of the referential integrity plug-in.
37894	Error	referint_postop insufficient arguments supplied	Insufficient arguments were provided for the referential integrity plug-in.	Check the configuration of the referential integrity plug-in.
37895	Error	referint_postop_modrdn : could not get parameters	The server was unable to retrieve the required information about a `modrdn` operation.	Check the delete request.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
37896	Error	referint_postop failed to get argc	The server was unable to determine the number of parameters to the referential integrity plug-in.	Restart the server.
37897	Error	referint_postop failed to get argv	The server was unable to retrieve the parameters associated with the referential integrity plug-in.	Restart the server.
37898	Error	referint_postop_modrdn args are NULL	No arguments were provided for the referential integrity plug-in.	Check the configuration of the referential integrity plug-in.
37899	Error	referint_postop_modrdn insufficient arguments supplied	Insufficient arguments were provided for the referential integrity plug-in.	Check the configuration of the referential integrity plug-in.
37900	Error	update_integrity required config file arguments missing	No arguments were provided for the referential integrity plug-in.	Check the configuration of the referential integrity plug-in.
37901	Error	referint_postop search (base=*base* filter=*filter*) returned error *error*.	An error occurred while searching for references to the deleted / renamed entry.	1. Check the error log for details of the error. 2. Restart the server.
37902	Error	referint_postop failed to get argc	The server was unable to determine the number of parameters to the referential integrity plug-in.	Restart the server.
37903	Error	referint_postop failed to get argv	The server was unable to retrieve the parameters associated with the referential integrity plug-in.	Restart the server.
37904	Error	args were null in referint_postop_start	No arguments were provided for the referential integrity plug-in.	Check the configuration of the referential integrity plug-in.
37905	Error	referint_postop_start PR_CreateThread failed.	The server was unable to create the thread to perform integrity updates.	Restart the server.
37906	Error	referint_postop_start insufficient arguments supplied	Insufficient arguments were provided to the referential integrity plug-in to determine the update delay.	Check the configuration of the referential integrity plug-in.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
37907	Error	referint_thread_func could not get args	The server was unable to retrieve the parameters associated with the referential integrity plug-in.	Restart the server.
37908	Error	referint_postop_close could not delete *filename*	The referential integrity log file could not be deleted.	Check the permissions on the specified file and restart the server.
37909	Error	referint_postop could not open integrity log *filename*	The referential integrity log file could not be opened for writing.	Check the permissions on the specified file and restart the server.
37910	Error	referint_postop could not write integrity log: line length exceeded. It will not be able to update references to the entry *entry*.	The change to be written to the integrity log file was longer than the maximum length allowed.	Check for references to the specified entry and update manually if necessary.
37911	Error	writeintegritylog: PR_Write failed : The disk may be full or the file is unwritable :: NSPR error - *error*.	The server was unable to write data to the integrity log file.	1. Check the integrity log file. 2. Check the filesystem status.
37912	Error	writeintegritylog: failed to close the file descriptor prfd; NSPR error - *error*.	An error occurred while closing the integrity log file.	1. Check the integrity log file. 2. Check the filesystem status.
38913	Error	The default SASL configuration entry could not be read or was not found in the dse.ldif file. It is mandatory.	The mandatory SASL configuration entry (cn=SASL,cn=security, cn=config) could not be retrieved from the configuration file.	Check the existence of this entry in the configuration file and add it if it is not present. (The entry contains the dsDaslConfig object class.)
38917	Error	Can’t find localhost name.	The local host name is absent from the naming service.	Add the local host name to the naming service.

Table A-1 Directory Server Error Codes

Code	Severity	Error Text	Probable Cause	Action
38918	Error	Sasl initialization failed.	Incorrect or missing information in the SASL configuration entry in the dse.ldif file (under `cn=sasl`.)	1. Check that the entry exists in the configuration file. 2. Check that the information in the configuration entry is valid (authentication mechanism names are correct.)

Appendix B

ns-slapd and slapd.exe Command-Line Utilities

In Chapter 2, we looked at the scripts for performing routine administration tasks on the Directory Server. This appendix describes how `ns-slapd` (UNIX and Linux) and `slapd.exe` (Windows) can be used to perform some of these tasks. It contains the following sections:

- Overview of ns-slapd and slapd.exe
- Finding and Executing the ns-slapd and slapd.exe Command-Line Utilities
- Exporting Databases
- Restoring and Backing up Databases
- Creating and Regenerating Indexes

Overview of ns-slapd and slapd.exe

The `ns-slapd` and `slapd.exe` binaries perform server administration tasks. While it can be argued that they allow a greater degree of flexibility for users, we strongly recommend that you use the command-line scripts described in Chapter 2, “Command-Line Scripts.”

ns-slapd (UNIX)

`ns-slapd` is used on a UNIX operating system to start the directory server process, to build a directory database from an LDIF file, or to convert an existing database to an LDIF file. For more information on starting and stopping the Directory Server, importing from LDIF using the command line, and exporting to LDIF using the command line, see Chapter 4, "Populating Directory Contents" in the *Sun ONE Directory Server Administration Guide*.

slapd.exe (Windows)

`slapd.exe` is the Windows equivalent of `ns-slapd`.

NOTE You must stop the server before running the `ns-slapd` and `slapd.exe` command-line utilities.

Finding and Executing the ns-slapd and slapd.exe Command-Line Utilities

The `ns-slapd` utility is delivered in both 64-bit and 32-bit versions.

After a default installation, the `ns-slapd` (64-bit) utility is stored under the following paths:

Platform	Location
Solaris Packages	*ServerRoot*`/bin/slapd/server/sparcv9/ns-slapd`
Compressed Archive Installation on Solaris	*ServerRoot*`/bin/slapd/server/64/ns-slapd`
HP-UX	*ServerRoot*`/bin/slapd/server/pa20_64/ns-slapd`

After a default installation, the `ns-slapd` (32-bit) and `slapd.exe` utilities are stored under the following paths:

Platform	Location
UNIX platforms	*ServerRoot*`/bin/slapd/server/ns-slapd`
Windows platforms	*ServerRoot*`\bin\slapd\server\slapd.exe`

The *ServerRoot* is the location of the Sun ONE Directory Server product. This path contains the shared binary files of the directory server, the administration server, and LDAP commands. For more information on your default *ServerRoot* path, see Table 1 on page 25. Do not mistake `ns-slapd.exe`, the `slapd` process watchdog, for `slapd.exe` on Windows.

CAUTION In order to execute the command-line utilities, you must change to the directory in which the command-line utilities are stored. Although it is possible to set command path and library path variables to execute the utilities, this is *not* recommended procedure. You run the risk of disrupting the correct execution of other utilities and of compromising the security of the system, particularly when you have more than one server version installed.

Exporting Databases

db2ldif

Exports the contents of a database to LDIF.

Shell Script Syntax (UNIX)

```
ns-slapd db2ldif -D instancedir [-n backend_instance] [-d debug_level] [-N]
[-a output_file] [-r] [-C] [-1] [{-s include_suffix}*] [-x exclude_suffix}*]
[-u] [-U] [-m] [-M] [-Y keydb-pwd] [-y keydb-pwd-file]
```

where *instancedir* is the location of your server configuration directory. Enter the full path to the *slapd-serverID* directory.

Batch File Syntax (Windows)

```
slapd db2ldif -D instancedir [-n backend_instance] [-d debug_level] [-N]
[-a output_file] [-r] [-C] [-1] [{-s include_suffix}*] [{-x exclude_suffix}*]
[-u] [-U] [-m] [-M] [-Y keydb-pwd] [-y keydb-pwd-file]
```

NOTES

1. You must specify either the `-n` or the `-s` option.
2. `db2ldif -r` cannot be used if another `slapd` process is running, because replication writes the RUV entry into the database during export. To export the database while a `slapd` process is running, use `db2ldif.pl -r` instead.

Options

Option	Meaning
`-D`	The full path to the `slapd-`*serverID* directory
`-a`	File name of the output LDIF file.
`-d`	Specifies the debug level. For more information, see "nsslapd-errorlog-level (Error Log Level)" on page 104.l
`-1`	For reasons of backward compatibility, delete the first line of the LDIF file which gives the version of the LDIF standard.
`-C`	Only the main db file is used.
`-m`	Minimal base64 encoding.
`-M`	Use of several files for storing the output LDIF, with each *instance* stored in *instance_output_file* (where *output_file* is the file name specified for `-a` option).
`-n`	Instance to be exported.
`-N`	Specifies that entry IDs are not to be included in the LDIF output. The entry IDs are necessary only if the `db2ldif` output is to be used as input to `db2index`.
`-r`	Export replica.
`-s`	Suffix(es) to be included or to specify the subtree(s) to be included if `-n` has been used.
`-u`	Request that the unique id is not exported.
`-U`	Request that the output LDIF is not folded.
`-x`	Suffix(es) to be excluded.
`-Y`	The password to the certificate key database (used for certificate-based client authentication).
`-y`	The file containing the certificate key database passwords (used for certificate-based client authentication).

Restoring and Backing up Databases

ldif2db

Imports LDIF files to the database.

Shell Script Syntax (UNIX)

```
ns-slapd ldif2db -D instancedir [-d debug_level] [-n backend_instance] [-O]
[-g uniqueid_type] [--namespaceid uniqueID] [-Y keydb-pwd] [-y
keydb-pwd-file] [{-s include_suffix}*] [{-x exclude_suffix}*] {-i ldif_file}*
```

where *ldif_file* is the name of the file containing the LDIF to be imported and *instancedir* is the location of your server configuration directory.

Batch File Syntax (Windows)

```
slapd ldif2db -D instancedir [-d debug_level] [-n backend_instance] [-O] [-g
uniqueid_type] [--namespaceid uniqueID] [-Y keydb-pwd] [-y keydb-pwd-file]
[{-s include_suffix}*] [{-x exclude_suffix}*] {-i ldif_file}*
```

NOTE You must specify either the `-n` or the `-s` option.

Options

Option	Meaning
`-D`	The full path to the `slapd-`*serverID* directory
`-d`	Specifies the debug level. For more information, see “nsslapd-errorlog-level (Error Log Level)” on page 104.
`--namespaceid`	Generates a namespace ID as a name-based unique ID. This is the same as specifying the `-g deterministic` option.

`-g` *uniqueid_type*	Generation of a unique ID. Type `none` for no unique ID to be generated and `deterministic` for the generated unique ID to be name-based. By default, a time-based unique ID is generated. If you use the `deterministic` generation to have a name-based unique ID, you can also specify the namespace you want the server to use as follows: `-g deterministic namespace_id` where `namespace_id` is a string of characters in the following format 00-xxxxxxxx-xxxxxxxx-xxxxxxxx-xxxxxxxx Use this option if you want to import the same LDIF file into two different directory servers, and if you want the contents of both directories to have the same set of unique IDs. If unique IDs already exist in the LDIF file you are importing, then the existing IDs are imported to the server regardless of the options you have specified.
`-i`	File name of the input ldif file(s). When you import multiple files, they are imported in the order in which you specify them on the command line.
`-n`	Instance to be imported. Ensure that you specify an instance that corresponds to the suffix contained by the LDIF file. Otherwise the data contained by the database is deleted and the import fails.
`-O`	Request that only the core db is created without attribute indexes.
`-s`	Suffix(es) to be included or to specify the subtree(s) to be included if -n has been used.
`-x`	Suffix(es) to be included.
`-Y`	The password to the certificate key database (used for certificate-based client authentication).
`-y`	The file containing the certificate key database passwords (used for certificate-based client authentication).

CAUTION If you are importing the LDIF file into your configuration directory, make sure the `o=NetscapeRoot` suffix and its contents are included in the LDIF file *before* you import. Do not exclude the suffix `o=NetscapeRoot` using `-s`, `-x`, or combination of the two. The Sun ONE Administration Server uses this suffix to store information about installed Sun ONE servers. Failure to import `o=NetscapeRoot` into your configuration directory could force you to reinstall (or restore from backup) all of your Sun ONE servers, including the Directory Server.

archive2db

Restores database from the archives.

Shell Script Syntax (UNIX)

```
ns-slapd archive2db -D instancedir [-d debuglevel] -a archivedir [-R]
```

Batch File Syntax (NT)

```
slapd archive2db -D instancedir [-d debuglevel] -a archivedir [-R]
```

Options

Option	Meaning
`-D`	Specifies the server configuration directory that contains the configuration information for the index creation process. You *must* specify the full path to the `slapd-`*serverID* directory.
`-d`	Specifies the debug level. For more information, see “nsslapd-errorlog-level (Error Log Level)” on page 104.
`-a`	Specifies the archive directory.
`-R`	Restores the database without restoring the changelog. If this option is used, the restored database will not include the list of changes made prior to the archive. Use this option with caution.

db2archive

Backs up all databases to the archives.

Shell Script Syntax (UNIX)

```
ns-slapd db2archive -D instancedir [-d debuglevel] -a archivedir
```

Batch File Syntax (Windows)

```
slapd db2archive -D instancedir [-d debuglevel] -a archivedir
```

Options

Option	Meaning
`-D`	Specifies the server configuration directory that contains the configuration information for the index creation process. You *must* specify the full path to the `slapd-`*serverID* directory.
`-d`	Specifies the debug level. For more information, see “nsslapd-errorlog-level (Error Log Level)” on page 104.
`-a`	Specifies the archive directory.

Creating and Regenerating Indexes

db2index

Creates and regenerates indexes.

Shell Script Syntax (UNIX)

```
ns-slapd db2index -D instancedir [-d debug_level] -n backend_name
{-t attribute_type}* {-T VLVSearchName}*
```

Batch File Syntax (Windows)

```
slapd db2index -D instancedir [-d debug_level] -n backend_name
{-t attribute_type}* {-T VLVSearchName}*
```

Options

Option	Meaning
`-d`	Specifies the debug level to use during index creation. For further information see “nsslapd-errorlog-level (Error Log Level)” on page 104.
`-D`	Specifies the server configuration directory that contains the configuration information for the index creation process. You must specify the full path to the `slapd-`*serverID* directory.
`-n`	Specifies the name of the backend containing the entries to index.
`-t`	Specifies the attribute to be indexed as well as the types of indexes to create and matching rules to apply (if any). If you want to specify a matching rule, you must specify an index type. You cannot use this option with option `-T`.
`-T`	Specifies the VLV tag to use to create VLV indexes. You can use the console to specify VLV tags for each database supporting your directory tree. You can also define additional VLV tags by creating them in LDIF, and adding them to the directory server configuration. You cannot use this option with option `-t`.

Appendix C

Directory Internationalization

Directory Server allows you to store, manage, and search for entries and their associated attributes in a number of different languages. An internationalized directory can be an invaluable corporate resource, providing employees and business partners with immediate access to the information they need in the languages they can understand.

The directory supports all international characters by default because directory data is stored in UTF-8. Further, Directory Server allows you to specify matching rules and collation orders based on language preferences in search operations.

NOTE You must use ASCII characters for attribute and object class names.

This appendix contains the following sections:

- About Locales
- Identifying Supported Locales
- Supported Language Subtypes

About Locales

Directory Server provides support for multiple languages through the use of locales. A locale identifies language-specific information about how users of a specific region, culture, and / or custom expect data to be presented, including how data of a given language is interpreted and how data is to be sorted, or collated.

In addition, the locale may indicate what code page an application should select for interaction with an end user concerning this data. A code page is an internal table that the operating system uses to relate keyboard keys to character font screen displays.

More specifically, a locale specifies:

- Collation order

  The collation order provides language and cultural-specific information about how the characters of a given language are to be sorted. It identifies things like the sequence of the letters in the alphabet, how to compare letters with accents to letters without accents, and if there are any characters that can be ignored when comparing strings. The collation order also takes into account culture-specific information about a language, such as the direction in which the language is read (left to right, right to left, or up and down).

- Character type

  The character type distinguishes alphabetic characters from numeric or other characters. In addition, it defines the mapping of upper-case to lower-case letters. For example, in some languages, the pipe (|) character is considered punctuation while in others it is considered alphabetic.

- Monetary format

  The monetary format specifies the monetary symbol used by a specific region, whether the symbol goes before or after its value, and how monetary units are represented.

- Time and date formats

  The time and date formats determine the customary appearance of times and dates in the region. The time format indicates whether the locale uses a 12- or 24-hour clock. The date format includes both the short date order, for example `MM/dd/yy` (month / day / year) or `dd/MM/yy` (day / month / year), and the long date format, including the names of months and days of the week in the given language. For example, the date “January 10, 2003” is represented as “10. leden 2003” in Czech and “10 janvier 2003” in French.

Identifying Supported Locales

When performing directory operations that require you to specify a locale, such as a search operation, you can use a language tag or a collation order object identifier (OID).

A language tag is a string that begins with the two-character lowercase language code that identifies the language (as defined in ISO standard 639). If necessary to distinguish regional differences in language, the language tag may also contain a country code, which is a two-character string (as defined in ISO standard 3166). The language code and country code are separated by a hyphen. For example, the language tag used to identify the American English locale is `en-US`.

An object identifier (OID) is a decimal number used to uniquely identify an object, such as an attribute or object class. The OIDs you use when searching or indexing an internationalized directory identify specific collation orders supported by the Directory Server. For example, the OID 1.3.6.1.4.1.42.2.27.9.4.74.1 identifies the Finnish collation order.

When performing an international search in the directory, use either the language tag or the OID to identify the collation order you want to use. When setting up an international index, you must use the OIDs. For more information on indexing, Chapter 10, “Managing Indexes” in the *Sun ONE Directory Server Administration Guide*.

Table C-1 lists each locale supported by Directory Server and identifies the associated language tags and OIDs. The old OID is provided for backward compatibility.

Table C-1 Supported Locales

Locale	Tag	Collation Order OID	Backward Compatible OID
Afrikaans	af	1.3.6.1.4.1.42.2.27.9.4.1.1	
Amharic Ethiopia	am	1.3.6.1.4.1.42.2.27.9.4.2.1	
Arabic	ar	1.3.6.1.4.1.42.2.27.9.4.3.1	2.16.840.1.113730.3.3.2.1.1
Arabic United Arab Emirates	ar-AE	1.3.6.1.4.1.42.2.27.9.4.4.1	
Arabic Bahrain	ar-BH	1.3.6.1.4.1.42.2.27.9.4.5.1	
Arabic Algeria	ar-DZ	1.3.6.1.4.1.42.2.27.9.4.6.1	
Arabic Egypt	ar-EG	1.3.6.1.4.1.42.2.27.9.4.7.1	
Arabic India	ar-IN	1.3.6.1.4.1.42.2.27.9.4.8.1	

Table C-1 Supported Locales

Locale	Tag	Collation Order OID	Backward Compatible OID
Arabic Iraq	ar-IQ	1.3.6.1.4.1.42.2.27.9.4.9.1	
Arabic Jordanar	ar-JO	1.3.6.1.4.1.42.2.27.9.4.10.1	
Arabic Kuwait	ar-KW	1.3.6.1.4.1.42.2.27.9.4.11.1	
Arabic Lebanon	ar-LB	1.3.6.1.4.1.42.2.27.9.4.12.1	
Arabic Lybia	ar-LY	1.3.6.1.4.1.42.2.27.9.4.13.1	
Arabic Morocco	ar-MA	1.3.6.1.4.1.42.2.27.9.4.14.1	
Arabic Oman	ar-OM	1.3.6.1.4.1.42.2.27.9.4.15.1	
Arabic Qatar	ar-QA	1.3.6.1.4.1.42.2.27.9.4.16.1	
Arabic Saudi Arabia	ar-SA	1.3.6.1.4.1.42.2.27.9.4.17.1	
Arabic Sudan	ar-SD	1.3.6.1.4.1.42.2.27.9.4.18.1	
Arabic Syria	ar-SY	1.3.6.1.4.1.42.2.27.9.4.19.1	
Arabic Tunisia	ar-TN	1.3.6.1.4.1.42.2.27.9.4.20.1	
Arabic Yemen	ar-YE	1.3.6.1.4.1.42.2.27.9.4.21.1	
Byelorussian	be	1.3.6.1.4.1.42.2.27.9.4.22.1	2.16.840.1.113730.3.3.2.2.1
Bulgarian	bg	1.3.6.1.4.1.42.2.27.9.4.23.1	2.16.840.1.113730.3.3.2.3.1
Bengali India	bn	1.3.6.1.4.1.42.2.27.9.4.24.1	
Catalan	ca	1.3.6.1.4.1.42.2.27.9.4.25.1	2.16.840.1.113730.3.3.2.4.1
Czech	cs	1.3.6.1.4.1.42.2.27.9.4.26.1	2.16.840.1.113730.3.3.2.5.1
Danish	da	1.3.6.1.4.1.42.2.27.9.4.27.1	2.16.840.1.113730.3.3.2.6.1
German	de or de-DE	1.3.6.1.4.1.42.2.27.9.4.28.1	2.16.840.1.113730.3.3.2.7.1
German Austria	de-AT	1.3.6.1.4.1.42.2.27.9.4.29.1	2.16.840.1.113730.3.3.2.8.1
German Belgium	de-BE	1.3.6.1.4.1.42.2.27.9.4.30.1	
German Swiss	de-CH	1.3.6.1.4.1.42.2.27.9.4.31.1	2.16.840.1.113730.3.3.2.9.1
German Luxembourg	de-LU	1.3.6.1.4.1.42.2.27.9.4.32.1	
Greek	el	1.3.6.1.4.1.42.2.27.9.4.33.1	2.16.840.1.113730.3.3.2.10.1
English (US)	en-US	1.3.6.1.4.1.42.2.27.9.4.34.1	2.16.840.1.113730.3.3.2.11.1
English Australian	en-AU	1.3.6.1.4.1.42.2.27.9.4.35.1	
English Canada	en-CA	1.3.6.1.4.1.42.2.27.9.4.36.1	2.16.840.1.113730.3.3.2.12.1
English Great Britain	en-GB	1.3.6.1.4.1.42.2.27.9.4.37.1	2.16.840.1.113730.3.3.2.13.1

Table C-1 Supported Locales

Locale	Tag	Collation Order OID	Backward Compatible OID
English Honk Kong	en-HK	1.3.6.1.4.1.42.2.27.9.4.38.1	
English Ireland	en-IE	1.3.6.1.4.1.42.2.27.9.4.39.1	2.16.840.1.113730.3.3.2.14.1
English India	en-IN	1.3.6.1.4.1.42.2.27.9.4.40.1	
English Malta	en-MT	1.3.6.1.4.1.42.2.27.9.4.41.1	
English New Zealand	en-NZ	1.3.6.1.4.1.42.2.27.9.4.42.1	
English Philippines	en-PH	1.3.6.1.4.1.42.2.27.9.4.43.1	
English Singapore	en-SG	1.3.6.1.4.1.42.2.27.9.4.44.1	
English Virgin Island	en-VI	1.3.6.1.4.1.42.2.27.9.4.45.1	
English South Africa	en-ZA	1.3.6.1.4.1.42.2.27.9.4.46.1	
English Zimbabwe	en-ZW	1.3.6.1.4.1.42.2.27.9.4.47.1	
Esperanto	eo	1.3.6.1.4.1.42.2.27.9.4.48.1	
Spanish	es or es-ES	1.3.6.1.4.1.42.2.27.9.4.49.1	2.16.840.1.113730.3.3.2.15.1
Spanish Argentina	es-AR	1.3.6.1.4.1.42.2.27.9.4.50.1	
Spanish Bolivia	es-BO	1.3.6.1.4.1.42.2.27.9.4.51.1	
Spanish Chile	es-CL	1.3.6.1.4.1.42.2.27.9.4.52.1	
Spanish Colombia	es-CO	1.3.6.1.4.1.42.2.27.9.4.53.1	
Spanish Costa Rica	es-CR	1.3.6.1.4.1.42.2.27.9.4.54.1	
Spanish Dominican Rep	es-DO	1.3.6.1.4.1.42.2.27.9.4.55.1	
Spanish Ecuador	es-EC	1.3.6.1.4.1.42.2.27.9.4.56.1	
Spanish Guatemala	es-GT	1.3.6.1.4.1.42.2.27.9.4.57.1	
Spanish Honduras	es-HN	1.3.6.1.4.1.42.2.27.9.4.58.1	
Spanish Mexico	es-MX	1.3.6.1.4.1.42.2.27.9.4.59.1	
Spanish Nicaragua	es-NI	1.3.6.1.4.1.42.2.27.9.4.60.1	
Spanish Panama	es-PA	1.3.6.1.4.1.42.2.27.9.4.61.1	
Spanish Peru	es-PE	1.3.6.1.4.1.42.2.27.9.4.62.1	
Spanish Puerto	es-PR	1.3.6.1.4.1.42.2.27.9.4.63.1	
Spanish Paraguay	es-PY	1.3.6.1.4.1.42.2.27.9.4.64.1	
Spanish Salvador	es-SV	1.3.6.1.4.1.42.2.27.9.4.65.1	
Spanish US	es-US	1.3.6.1.4.1.42.2.27.9.4.66.1	

Table C-1 Supported Locales

Locale	Tag	Collation Order OID	Backward Compatible OID
Spanish Uruguay	es-UY	1.3.6.1.4.1.42.2.27.9.4.67.1	
Spanish Venezuela	es-VE	1.3.6.1.4.1.42.2.27.9.4.68.1	
Estonian	et	1.3.6.1.4.1.42.2.27.9.4.69.1	2.16.840.1.113730.3.3.2.16.1
Basque	eu	1.3.6.1.4.1.42.2.27.9.4.70.1	
Persian	fa	1.3.6.1.4.1.42.2.27.9.4.71.1	
Persian India	fa-IN	1.3.6.1.4.1.42.2.27.9.4.72.1	
Persian Iran	fa-IR	1.3.6.1.4.1.42.2.27.9.4.73.1	
Finnish	fi	1.3.6.1.4.1.42.2.27.9.4.74.1	2.16.840.1.113730.3.3.2.17.1
Faeroese	fo	1.3.6.1.4.1.42.2.27.9.4.75.1	
French	fr or fr-FR	1.3.6.1.4.1.42.2.27.9.4.76.1	2.16.840.1.113730.3.3.2.18.1
French Belgium	fr-BE	1.3.6.1.4.1.42.2.27.9.4.77.1	2.16.840.1.113730.3.3.2.19.1
French Canada	fr-CA	1.3.6.1.4.1.42.2.27.9.4.78.1	2.16.840.1.113730.3.3.2.20.1
French Swiss	fr-CH	1.3.6.1.4.1.42.2.27.9.4.79.1	2.16.840.1.113730.3.3.2.21.1
French Luxembourg	fr-LU	1.3.6.1.4.1.42.2.27.9.4.80.1	
Irish	ga	1.3.6.1.4.1.42.2.27.9.4.81.1	
Galician	gl	1.3.6.1.4.1.42.2.27.9.4.82.1	
Gujarati	gu	1.3.6.1.4.1.42.2.27.9.4.83.1	
Manx (Isle of Man)	gv	1.3.6.1.4.1.42.2.27.9.4.84.1	
Hebrew	he or iw	1.3.6.1.4.1.42.2.27.9.4.85.1	2.16.840.1.113730.3.3.2.27.1
Hindi	hi	1.3.6.1.4.1.42.2.27.9.4.86.1	
Croatian	hr	1.3.6.1.4.1.42.2.27.9.4.87.1	2.16.840.1.113730.3.3.2.22.1
Hungarian	hu	1.3.6.1.4.1.42.2.27.9.4.88.1	2.16.840.1.113730.3.3.2.23.1
Armenian	hy	1.3.6.1.4.1.42.2.27.9.4.89.1	
Indonesian	id	1.3.6.1.4.1.42.2.27.9.4.90.1	
Icelandic	is	1.3.6.1.4.1.42.2.27.9.4.91.1	2.16.840.1.113730.3.3.2.24.1
Italian	it	1.3.6.1.4.1.42.2.27.9.4.92.1	2.16.840.1.113730.3.3.2.25.1
Italian Swiss	it-CH	1.3.6.1.4.1.42.2.27.9.4.93.1	2.16.840.1.113730.3.3.2.26.1
Japanese	ja	1.3.6.1.4.1.42.2.27.9.4.94.1	2.16.840.1.113730.3.3.2.28.1
Greenlandic	kl	1.3.6.1.4.1.42.2.27.9.4.95.1	

Table C-1 Supported Locales

Locale	Tag	Collation Order OID	Backward Compatible OID
Kannada	kn	1.3.6.1.4.1.42.2.27.9.4.96.1	
Korean	ko	1.3.6.1.4.1.42.2.27.9.4.97.1	2.16.840.1.113730.3.3.2.29.1
Konkani	kok	1.3.6.1.4.1.42.2.27.9.4.98.1	
Cornish	kw	1.3.6.1.4.1.42.2.27.9.4.99.1	
Lithuanian	lt	1.3.6.1.4.1.42.2.27.9.4.100.1	2.16.840.1.113730.3.3.2.30.1
Latvian or Lettish	lv	1.3.6.1.4.1.42.2.27.9.4.101.1	2.16.840.1.113730.3.3.2.31.1
Macedonian	mk	1.3.6.1.4.1.42.2.27.9.4.102.1	2.16.840.1.113730.3.3.2.32.1
Marathi	mr	1.3.6.1.4.1.42.2.27.9.4.103.1	
Maltese	mt	1.3.6.1.4.1.42.2.27.9.4.104.1	
Dutch	nl or nl-NL	1.3.6.1.4.1.42.2.27.9.4.105.1	2.16.840.1.113730.3.3.2.33.1
Dutch Belgium	nl-BE	1.3.6.1.4.1.42.2.27.9.4.106.1	2.16.840.1.113730.3.3.2.34.1
Norwegian	no or no-NO	1.3.6.1.4.1.42.2.27.9.4.107.1	2.16.840.1.113730.3.3.2.35.1
Norwegian Nynorsk	no-NO-NY	1.3.6.1.4.1.42.2.27.9.4.108.1	2.16.840.1.113730.3.3.2.37.1
Norwegian Nynorsk	nn	1.3.6.1.4.1.42.2.27.9.4.109.1	
Norwegian Bokmul	nb or no-NO-B	1.3.6.1.4.1.42.2.27.9.4.110.1	2.16.840.1.113730.3.3.2.36.1
Oromo (Afan)	om	1.3.6.1.4.1.42.2.27.9.4.111.1	
Oromo Ethiopia	om-ET	1.3.6.1.4.1.42.2.27.9.4.112.1	
Oromo Kenya	om-KE	1.3.6.1.4.1.42.2.27.9.4.113.1	
Polish	pl	1.3.6.1.4.1.42.2.27.9.4.114.1	2.16.840.1.113730.3.3.2.38.1
Portuguese	pt or pt-PT	1.3.6.1.4.1.42.2.27.9.4.115.1	
Portuguese Brazil	pt-BR	1.3.6.1.4.1.42.2.27.9.4.116.1	
Romanian	ro	1.3.6.1.4.1.42.2.27.9.4.117.1	2.16.840.1.113730.3.3.2.39.1
Russian	ru or ru-RU	1.3.6.1.4.1.42.2.27.9.4.118.1	2.16.840.1.113730.3.3.2.40.1
Russian Ukraine	ru-UA	1.3.6.1.4.1.42.2.27.9.4.119.1	
Serbo-Croatian	sh	1.3.6.1.4.1.42.2.27.9.4.120.1	2.16.840.1.113730.3.3.2.41.1
Slovak	sk	1.3.6.1.4.1.42.2.27.9.4.121.1	2.16.840.1.113730.3.3.2.42.1
Slovenian	sl	1.3.6.1.4.1.42.2.27.9.4.122.1	2.16.840.1.113730.3.3.2.43.1

Table C-1 Supported Locales

Locale	Tag	Collation Order OID	Backward Compatible OID
Somali	so or so-SO	1.3.6.1.4.1.42.2.27.9.4.123.1	
Somali Djibouti	so-DJ	1.3.6.1.4.1.42.2.27.9.4.124.1	
Somali Ethiopia	so-ET	1.3.6.1.4.1.42.2.27.9.4.125.1	
Somali Kenya	so-KE	1.3.6.1.4.1.42.2.27.9.4.126.1	
Albanian	sq	1.3.6.1.4.1.42.2.27.9.4.127.1	2.16.840.1.113730.3.3.2.44.1
Serbian	sr	1.3.6.1.4.1.42.2.27.9.4.128.1	2.16.840.1.113730.3.3.2.45.1
Swedish	sv-SE	1.3.6.1.4.1.42.2.27.9.4.129.1	2.16.840.1.113730.3.3.2.46.1
Swedish Finland	sv-FI	1.3.6.1.4.1.42.2.27.9.4.130.1	
Swahili	sw	1.3.6.1.4.1.42.2.27.9.4.131.1	
Swahili Kenya	sw-KE	1.3.6.1.4.1.42.2.27.9.4.132.1	
Swahili Tanzania	sw-TZ	1.3.6.1.4.1.42.2.27.9.4.133.1	
Tamil	ta	1.3.6.1.4.1.42.2.27.9.4.134.1	
Tegulu	te	1.3.6.1.4.1.42.2.27.9.4.135.1	
Thai	th	1.3.6.1.4.1.42.2.27.9.4.136.1	
Tigrinya	ti	1.3.6.1.4.1.42.2.27.9.4.137.1	
Tigrinya Eritrea	ti-ER	1.3.6.1.4.1.42.2.27.9.4.138.1	
Tigrinya Ethiopia	ti-ET	1.3.6.1.4.1.42.2.27.9.4.139.1	
Turkish	tr	1.3.6.1.4.1.42.2.27.9.4.140.1	2.16.840.1.113730.3.3.2.47.1
Ukrainian	uk	1.3.6.1.4.1.42.2.27.9.4.141.1	2.16.840.1.113730.3.3.2.48.1
Vietnamese	vi	1.3.6.1.4.1.42.2.27.9.4.142.1	
Chinese	zh	1.3.6.1.4.1.42.2.27.9.4.143.1	2.16.840.1.113730.3.3.2.49.1
Chinese China	zh-CN	1.3.6.1.4.1.42.2.27.9.4.144.1	
Chinese Honk Kong	zh-HK	1.3.6.1.4.1.42.2.27.9.4.145.1	
Chinese Mongolia	zh-MO	1.3.6.1.4.1.42.2.27.9.4.146.1	
Chinese Singapore	zh-SG	1.3.6.1.4.1.42.2.27.9.4.147.1	
Chinese Taiwan	zh-TW	1.3.6.1.4.1.42.2.27.9.4.148.1	2.16.840.1.113730.3.3.2.50.1

Supported Language Subtypes

Language subtypes can be used by clients to indicate specific attributes in characters of a language other than the default language of a deployment. For example, German users may prefer to see addresses in German when possible. In this case, you can select German as a language subtype for the `streetAddress` attribute so that users can search for either the English or the German representation of the address. If you specify a language subtype for an attribute, the subtype is added to the attribute name as follows:

attribut`e;lang-`*subtype*

The example mentioned previously would be displayed in LDIF as follows:

```
streetAddress;lang-en: 10 Schlossplatz, 76113, Karlsruhe, Germany
streetAddress;lang-de: Schloßplatz 10, 76113, Karlsruhe, Deutschland
```

Table C-2 contains the list of supported language subtypes.

Table C-2 Supported Language Subtypes

Language	Language Tag
Afrikaans	af
Albanian	sq
Amharic Ethiopia	am
Arabic	ar
Armenian	hy
Basque	eu
Bengali India	bn
Bulgarian	bg
Byelorussian	be
Catalan	ca
Chinese	zh
Cornish	kw
Croatian	hr
Czech	cs
Danish	da
Dutch	nl

Table C-2 Supported Language Subtypes *(Continued)*

Language	Language Tag
English	en
Esperanto	eo
Estonian	et
Faeroese	fo
Finnish	fi
French	fr
Galician	gl
German	de
Greek	el
Greenlandic	kl
Gujarati	gu
Hebrew	he or iw
Hindi	hi
Hungarian	hu
Icelandic	is
Indonesian	id
Irish	ga
Italian	it
Japanese	ja
Kannada	kn
Konkani	kok
Korean	ko
Latvian or Lettish	lv
Lithuanian	lt
Macedonian	mk
Maltese	mt
Manx (Isle of Man)	gv
Marathi	mr
Norwegian	no

Table C-2 Supported Language Subtypes *(Continued)*

Language	Language Tag
Oromo	om
Persian	fa
Polish	pl
Portuguese	pt
Romanian	ro
Russian	ru
Serbian	sr
Serbo-Croatian	sh
Slovak	sk
Slovenian	sl
Somali	so
Spanish	es
Swahili	sw
Swedish	sv
Tamil	ta
Tegulu	te
Thai	th
Tigrinya	ti
Turkish	tr
Ukrainian	uk
Vietnamese	vi

Appendix D

LDAP URLs

One way to express an LDAP query is to use a URL to specify the directory server host machine and the DN or filter for the search. Sun ONE Directory Server will respond to queries sent as LDAP URLs and return an HTML page representing the results. This allows web browsers to perform searches of the directory, if anonymous searching is permitted.

You can also use LDAP URLs to specify target entries when managing Directory Server referrals or access control instructions.

This chapter contains the following sections:

- Components of an LDAP URL
- Escaping Unsafe Characters
- Examples of LDAP URLs

Components of an LDAP URL

LDAP URLs have the following syntax:

`ldap[s]://`*hostname*`:`*port*`/`*base_dn*`?`*attributes*`?`*scope*`?`*filter*

When `ldap://...` is specified, standard LDAP is used to connect to the LDAP servers. When `ldaps://...` is specified, LDAP over SSL is used to connect to the LDAP server.

Table D-1 LDAP URL Components

Component	Description
hostname	Name (or IP address in dotted format) of the LDAP server. For example: `ldap.example.com` or `192.202.185.90`
port	Port number of the LDAP server (for example, 49153). If no port is specified, the standard LDAP port (389) or LDAPS port (636) is used.
base_dn	Distinguished name (DN) of an entry in the directory. This DN identifies the entry that is the starting point of the search. If no base DN is specified, the search starts at the root of the directory tree.
attributes	The attributes to be returned. To specify more than one attribute, use commas to separate the attributes (for example, `"cn,mail,telephoneNumber"`). If no attributes are specified in the URL, all attributes are returned.
scope	The scope of the search, which can be one of these values: • `base` retrieves information about the distinguished name (*base_dn*) specified in the URL only. • `one` retrieves information about entries one level below the distinguished name (*base_dn*) specified in the URL. The base entry is not included in this scope. • `sub` retrieves information about entries at all levels below the distinguished name (*base_dn*) specified in the URL. The base entry is included in this scope. If no scope is specified, the server performs a `base` search.
filter	Search filter to apply to entries within the specified scope of the search. If no filter is specified, the server uses the filter `(objectClass=*)`.

The attributes, scope, and filter components are identified by their positions in the URL. If you do not want to specify any attributes, you must still include the question marks delimiting that field. For example, to specify a subtree search starting from `"dc=example,dc=com"` that returns all attributes for entries matching `"(sn=Jensen)"`, use the following LDAP URL:

```
ldap://ldap.example.com/dc=example,dc=com??sub?(sn=Jensen)
```

The two consecutive question marks ?? indicate that no attributes have been specified. Since no specific attributes are identified in the URL, all attributes are returned in the search.

Escaping Unsafe Characters

Any *unsafe* characters in the URL must be represented by a special sequence of characters. This is called escaping unsafe characters. For example, a space is an unsafe character that must be represented as `%20` within the URL. Thus, the distinguished name `"o=example corporation"` must be encoded as `"o=example%20corporation"`.

The following table lists the characters that are considered unsafe within URLs and provides the associated escape characters to use in place of the unsafe character:

Unsafe Character	Escape Characters
space	%20
<	%3c
>	%3e
"	%22
#	%23
%	%25
{	%7b
}	%7d
|	%7c
\	%5c
^	%5e
~	%7e
[	%5b
]	%5d
‘	%60

Examples of LDAP URLs

- The following LDAP URL specifies a base search for the entry with the distinguished name `dc=example,dc=com`.

  ```
  ldap://ldap.example.com/dc=example,dc=com
  ```

  - Because no port number is specified, the standard LDAP port number (389) is used.
  - Because no attributes are specified, the search returns all attributes.
  - Because no search scope is specified, the search is restricted to the base entry `dc=example,dc=com`.
  - Because no filter is specified, the directory uses the default filter (`objectclass=*`).

- The following LDAP URL retrieves the `postalAddress` attribute of the entry with the DN `dc=example,dc=com`:

  ```
  ldap://ldap.example.com/dc=example,dc=com?postalAddress
  ```

  - Because no search scope is specified, the search is restricted to the base entry `dc=example,dc=com`.
  - Because no filter is specified, the directory uses the default filter (`objectclass=*`).

- The following LDAP URL retrieves the `cn`, and `mail` attributes of the entry for Barbara Jensen:

```
ldap://ldap.example.com/cn=Barbara%20Jensen,dc=example,
 dc=com?cn,mail
```

  - Because no search scope is specified, the search is restricted to the base entry `cn=Barbara Jensen,dc=example,dc=com`.
  - Because no filter is specified, the directory uses the default filter (`objectclass=*`).

- The following LDAP URL specifies a search for entries that have the surname `Jensen` and are at any level under `dc=example,dc=com`:

  ```
  ldap://ldap.example.com/dc=example,dc=com??sub?(sn=Jensen)
  ```

  - Because no attributes are specified, the search returns all attributes.
  - Because the search scope is `sub`, the search encompasses the base entry `dc=example,dc=com` and entries at all levels under the base entry.

- The following LDAP URL specifies a search for the object class for all entries one level under `dc=example,dc=com`:

  ```
  ldap://ldap.example.com/dc=example,dc=com?objectClass?one
  ```

  - Because the search scope is one, the search encompasses all entries one level under the base entry `dc=example,dc=com`. The search scope does not include the base entry.
  - Because no filter is specified,the directory uses the default filter (`objectclass=*`).

NOTE The syntax for LDAP URLs does not include any means for specifying credentials or passwords. Search requests initiated through LDAP URLs are unauthenticated (anonymous), unless the LDAP client that supports LDAP URLs provides an authentication mechanism.

Appendix E

LDAP Data Interchange Format

Sun ONE Directory Server uses the LDAP Data Interchange Format (LDIF) to describe a directory and directory entries in text format. LDIF is commonly used to build the initial directory database or to add large numbers of entries to the directory simultaneously. LDIF is also used to describe changes to directory entries. For this reason, most of Directory Server's command-line utilities rely on LDIF for either input or output.

Because LDIF is a text file format, you can create LDIF files using virtually any language. All directory data is stored using the UTF-8 encoding of Unicode. Therefore, the LDIF files you create must also be UTF-8 encoded.

This appendix provides information about LDIF in the following sections:

- LDIF File Format
- Specifying Directory Entries Using LDIF
- Defining Directories Using LDIF
- Storing Information in Multiple Languages

LDIF File Format

LDIF consists of one or more directory entries separated by a blank line. Each LDIF entry consists of an optional entry ID, a required distinguished name, one or more object classes, and multiple attribute definitions.

The LDIF format is defined in RFC 2849 *The LDAP Data Interchange Format (LDIF)*. Sun ONE Directory Server is compliant with this standard.

The basic form of a directory entry represented in LDIF is as follows:

```
dn: distinguished_name
objectClass: object_class
objectClass: object_class
...
attribute_type[;subtype]:attribute_value
attribute_type[;subtype]:attribute_value
...
```

You must supply the DN and at least one object class definition. In addition, you must include any attributes required by the object classes that you define for the entry. All other attributes and object classes are optional. You can specify object classes and attributes in any order. The space after the colon is also optional. For information on standard object classes and attributes, see “Object Class Reference,” on page 317 and “Attribute Reference,” on page 397.

Table E-1 describes the LDIF fields shown in the previous definition.

Table E-1 LDIF Fields

Field	Definition
[*id*]	Optional. A positive decimal number representing the entry ID. The database creation tools generate this ID for you. Never add or edit this value yourself.
`dn:` *distinguished_name*	Specifies the distinguished name for the entry. For a complete description of distinguished names, refer to the *Sun ONE Directory Server Deployment Guide*.
`objectClass:` *object_class*	Specifies an object class to use with this entry. The object class identifies the types of attributes, or schema, allowed and required for the entry. See “Object Class Reference,” on page 317 for a list of standard object classes.
attribute_type	Specifies a descriptive attribute to use with the entry. The attribute should be defined in the schema. See “Attribute Reference,” on page 397 for a list of standard attributes.

Table E-1 LDIF Fields *(Continued)*

Field	Definition
[*subtype*]	Optional. Specifies a subtype, which may be one of: • language (*attribute*`;lang-`*subtype*) • binary (*attribute*`;binary`) • pronunciation (*attribute*`;phonetic`) Use this tag to identify the language in which the corresponding attribute value is expressed, or whether the attribute value is binary or a pronunciation of an attribute value. For more information, see "Adding an Attribute Subtype" in Chapter 2 of the *Sun ONE Directory Server Administration Guide*.
attribute_value	Specifies the attribute value to be used with the attribute type.

The LDIF syntax for representing a change to an entry in the directory is different from the syntax described above. For information on using LDIF to modify directory entries, see Chapter 2, "Creating Directory Entries" in the *Sun ONE Directory Server Administration Guide*.

Continuing Lines in LDIF

When you specify LDIF, you can break and continue, or fold, a line by indenting the continued portion of the line by exactly one space. For example, the following two statements are identical:

```
dn: cn=Jake Lupinski,dc=example,dc=com
```

```
dn: cn=Jake Lup
 inski,dc=exam
 ple,dc=com
```

You are not required to break and continue LDIF lines. However, doing so may improve the readability of an LDIF file.

Representing Binary Data

You can represent binary data, such as a JPEG image, in LDIF using one of the following methods:

- The standard LDIF notation, the lesser than (<) symbol.
- The command-line utility `ldapmodify` with the `-b` parameter.
- Base 64 encoding.

Using Standard LDIF Notation

For example:

```
jpegphoto:< file:/path/to/photo
```

Note that this path is relative to the client, not to the server. If you use this standard notation, you do not need to specify the `ldapmodify -b` parameter. However, you must add the following line to the beginning of your LDIF file, or your LDIF update statements:

```
version:1
```

For example, you could use the following `ldapmodify` command:

```
prompt% ldapmodify -D userDN -w user_passwd
```

```
version: 1
dn: cn=Barney Fife,ou=People,dc=example,dc=com
changetype: modify
add: userCertificate
userCertificate;binary:< file: BarneysCert
```

Using ldapmodify -b

Whenever possible, you should use the standard notation described above. The method described in this section is supported only for reasons of backward compatibility with earlier versions of Directory Server.

Sun ONE Directory Server accepts the `ldapmodify` command with the `-b` parameter and the following LDIF notation:

```
jpegphoto: /path/to/photo
```

This notation indicates that `ldapmodify` should read the referenced file for binary values if the attribute value begins with a slash.

NOTE This behavior is not supported by the Directory Server console. In the console, values that begin with a slash are added literally to the directory.

Using Base 64 Encoding

You identify base 64 encoded data by using the `::` symbol. For example:

`jpegPhoto::` *encoded_data*

In addition to binary data, other values that must be base 64-encoded include:

- Any value that begins with a semicolon (;) or a space.
- Any value that contains non-ASCII data, including new lines.

Use the `directoryserver ldif` command-line utility (on Solaris 9 platforms) with the `-b` parameter to convert binary data to LDIF format:

```
/usr/sbin/directoryserver ldif -b attributeName
```

where *attributeName* is the name of the attribute to which you are supplying the binary data. The binary data is read from standard input and the results are written to standard output. Thus, you should use redirection operators to select input and output files.

The command takes any input and formats it with the correct line continuation and appropriate attribute information. It also assesses whether the input requires base 64 encoding. For example:

```
/usr/sbin/directoryserver ldif -b jpegPhoto < mark.jpg > out.ldif
```

This example takes a binary file containing a JPEG-formatted image and converts it into LDIF format for the attribute named `jpegPhoto`. The output is saved to `out.ldif`.

The `-b` option specifies that the utility should interpret the entire input as a single binary value. If `-b` is not present, each line is considered to be a separate input value.

NOTE On platforms other than Solaris 9, the `ldif` command syntax is:

/ServerRoot`/slapd-`*serverID*`/ldif`

You can then edit the output file to add the LDIF statements required to create or modify the directory entry that will contain the binary value. For example, you can open the file `out.ldif` in a text editor and add the following lines (shown in bold) at the top of the file:

```
dn: cn=Barney Fife,ou=People,dc=example,dc=com
changetype: modify
add: jpegPhoto
jpegPhoto:: encoded_data
```

In this example, *encoded_data* represents the contents of the `out.ldif` file produced by the command.

Specifying Directory Entries Using LDIF

You can store many types of entries in a directory. This section concentrates on three of the most common types of entries used in a directory: organization, organizational unit, and organizational person entries.

The object classes defined for an entry indicate whether the entry represents an organization, an organizational unit, an organizational person, or some other type of entry. For a general discussion of the types of entries you can create in a directory, see the *Sun ONE Directory Server Deployment Guide*. For a complete list of the default object classes and a list of the most commonly used attributes, see “Object Class Reference,” on page 317 and “Attribute Reference,” on page 397.

Specifying Organization Entries

Directories often have at least one organization entry. Typically this is the first, or topmost entry in the directory. The organization entry often corresponds to the suffix set for the directory. For example, if your directory is defined to use a suffix of `o=example.com`, you will probably have an organization entry named `o=example.com`.

The LDIF that you specify to define an organization entry should appear as follows:

```
dn: distinguished_name
objectClass: top
objectClass: organization
o: organization_name
list_of_optional_attributes
...
```

The following is a sample organization entry in LDIF format:

```
dn: o=example.com
objectclass: top
objectclass: organization
o: example.com Corporation
description: Fictional company for example purposes
telephonenumber: 555-5555
```

The organization name in the following example uses a comma:

```
dn: o="example.com Chile\\, S.A."
objectclass: top
objectclass: organization
o: "example.com Chile\\, S.A."
description: Fictional company for example purposes
telephonenumber: 555-5556
```

Each element of the LDIF-formatted organization entry is defined in Table E-2.

Table E-2 LDIF Elements in Organization Entries

LDIF Element	Description
`dn:` *distinguished_name*	Specifies the distinguished name for the entry. DNs are described in the *Sun ONE Directory Server Deployment Guide*. A DN is required.
`objectClass: top`	Required. Specifies the `top` object class.
`objectClass: organization`	Specifies the `organization` object class. This line defines the entry as an organization. See "Attribute Reference," on page 397 for a list of the attributes you can use with this object class.
`o:` *organization_name*	Specifies the organization's name. If the organization name includes a comma, you must escape the comma by a single backslash or the entire organization argument must be enclosed in quotation marks. However, if you are working with a UNIX shell, this backslash will also need escaping which means that you will have to use two backslashes. For example, to set the suffix to example.com Bolivia, S.A. you would enter `"o: example.com Bolivia\\, S.A."`.
list_of_attributes	Specifies the list of optional attributes that you want to maintain for the entry. See "Attribute Reference," on page 397 for a list of the attributes you can use with this object class.

Specifying Organizational Unit Entries

Organizational unit entries are often used to represent major branch points, or subdirectories, in the directory tree. They correspond to major, reasonably static entities within an enterprise, such as a subtree that contains people, or a subtree that contains groups. However, the organizational unit attribute that is contained in the entry may also represent a major organization within the enterprise, such as marketing or engineering.

There is usually more than one organizational unit, or branch point, within a directory tree. For information on how to design your directory tree, see the *Sun ONE Directory Server Deployment Guide*.

The LDIF that you specify to define an organizational unit entry must appear as follows:

```
dn: distinguished_name
objectClass: top
objectClass: organizationalUnit
ou: organizational_unit_name
list_of_optional_attributes
...
```

The following is a sample organizational unit entry in LDIF format:

```
dn: ou=people, o=example.com
objectclass: top
objectclass: organizationalUnit
ou: people
description: Fictional organizational unit for example purposes
```

Table E-3 defines each element of the LDIF-formatted organizational unit entry.

Table E-3 LDIF Elements in Organizational Unit Entries

LDIF Element	Description

`dn:` *distinguished_name*	Specifies the distinguished name for the entry. A DN is required. If there is a comma in the DN, the comma must be escaped with a backslash (\). For example:
	`dn: ou=people,o=example.com`
	`Bolivia\,S.A.`

`objectClass: top`	Required. Specifies the `top` object class.
`objectClass: organizationalUnit`	Specifies the `organizationalUnit` object class. This line defines the entry as an organizationalUnit. See the "Attribute Reference," on page 397 for a list of the attributes you can use with this object class.

Table E-3 LDIF Elements in Organizational Unit Entries *(Continued)*

LDIF Element	Description
`ou:` *organizational_unit_name*	Attribute that specifies the organizational unit's name.
list_of_attributes	Specifies the list of optional attributes that you want to maintain for the entry. See "Attribute Reference," on page 397 for a list of the attributes you can use with this object class.

Specifying Organizational Person Entries

The majority of the entries in your directory represent organizational people.

In LDIF, the definition of an organizational person is as follows:

```
dn: distinguished_name
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: common_name
sn: surname
list_of_optional_attributes
```

The following is an example organizational person entry in LDIF format:

```
dn: uid=bjensen,ou=people,o=example.com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: Babs Jensen
sn: Jensen
givenname: Babs
uid: bjensen
ou: Marketing
ou: people
description: Fictional person for example purposes
telephonenumber: 555-5557
userpassword: {sha}dkfljlk34r2kljdsfk9
```

Table E-4 defines each aspect of the LDIF person entry.

Table E-4 LDIF Elements in Person Entries

LDIF Element	Description
`dn:` *distinguished_name*	Specifies the distinguished name for the entry. A DN is required. If there is a comma in the DN, the comma must be escaped with a backslash (\). For example, `dn:uid=bjensen,ou=people,o=example.com` `Bolivia\,S.A.`
`objectClass: top`	Required. Specifies the `top` object class.
`objectClass: person`	Specifies the `person` object class. This object class specification should be included because many LDAP clients require it during search operations for a person or an organizational person.
`objectClass:` `organizationalPerson`	Specifies the `organizationalPerson` object class. This object class specification should be included because some LDAP clients require it during search operations for an organizational person.
`objectClass:` `inetOrgPerson`	Specifies the `inetOrgPerson` object class. The `inetOrgPerson` object class is recommended for the creation of an organizational person entry because this object class includes the widest range of attributes. The `uid` attribute is required by this object class, and entries that contain this object class are named based on the value of the uid attribute. See "Attribute Reference," on page 397 for a list of the attributes you can use with this object class.
`cn:` *common_name*	Specifies the person's common name which is the full name commonly used by the person. For example, `cn: Bill Anderson`. At least one common name is required.
`sn:` *surname*	Specifies the person's surname, or last name. For example, `sn: Anderson`. A surname is required.
list_of_attributes	Specifies the list of optional attributes that you maintain for the entry. See "Attribute Reference," on page 397 for a list of the attributes you can use with this object class.

Defining Directories Using LDIF

You can define the contents of an entire directory using LDIF. Using LDIF is an efficient method of directory creation when you have many entries to add to the directory.

To create a directory using LDIF, follow these steps:

Create an ASCII file containing the entries you want to add in LDIF format.

Make sure each entry is separated from the next by an empty line. You should use just one line. The first line of the file must not be blank (otherwise the `ldapmodify` utility will exit). For more information, see “Specifying Directory Entries Using LDIF,” on page 594.

1. Begin each file with the topmost, or root, entry in the database.

   The root entry must represent the suffix or sub-suffix contained by the database. For example, if your database has the suffix `dc=example,dc=com`, the first entry in the directory must be

   ```
   dn: dc=example,dc=com
   ```

   For information on suffixes, see “Suffix Configuration Attributes Under cn="suffixName",” on page 140.

2. Make sure that an entry representing a branch point in the LDIF file is placed before the entries that you want to create under that branch.

   For example, if you want to place an entry in a people and a group subtree, create the branch point for those subtrees before creating entries within those subtrees.

3. Create the directory from the LDIF file using one of the following methods:

   - Directory Server console

     Use this method if you have a small database to import (less than 1000 entries). See “Importing LDIF From the Console” in Chapter 4 of the *Sun ONE Directory Server Administration Guide*.

   - `ldif2db` command-line utility

     Use this method if you have a large database to import (more than 1,000 entries). See “Importing Using the ldif2db Command” in Chapter 4 of the *Sun ONE Directory Server Administration Guide*.

     `ldapmodify` command-line utility with the `-a` parameter

Use this method if you currently have a directory database, but you are adding a new subtree to the database. Unlike the other methods for creating the directory from an LDIF file, Directory Server must be running before you can add a subtree using `ldapmodify`. See "Adding and Modifying Entries Using ldapmodify" in Chapter 2 of the *Sun ONE Directory Server Administration Guide*.

LDIF File Example

The following example shows an LDIF file that contains one organization, two organizational units, and three organizational person entries:

```
dn: o=example.com Corp
objectclass: top
objectclass: organization
o: example.com Corp
description: Fictional organization for example purposes

dn: ou=People,o=example.com Corp
objectclass: top
objectclass: organizationalUnit
ou: People
description: Fictional organizational unit for example purposes
tel: 555-5559

dn: cn=June Rossi,ou=People,o=example.com Corp
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: June Rossi
sn: Rossi
givenName: June
mail: rossi@example.com
userPassword: {sha}KDIE3AL9DK
ou: Accounting
ou: people
telephoneNumber: 2616
roomNumber: 220

dn: cn=Marc Chambers,ou=People,o=example.com Corp
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Marc Chambers
sn: Chambers
givenName: Marc
```

```
mail: chambers@example.com
userPassword: {sha}jdl2alem87dlacz1
telephoneNumber: 2652
ou: Manufacturing
ou: People
roomNumber: 167

dn: cn=Robert Wong,ou=People,o=example.com Corp
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Robert Wong
cn: Bob Wong
sn: Wong
givenName: Robert
givenName: Bob
mail: bwong@example.com
userPassword: {sha}nn2msx761
telephoneNumber: 2881
roomNumber: 211
ou: Manufacturing
ou: people

dn: ou=Groups,o=example.com Corp
objectclass: top
objectclass: organizationalUnit
ou: groups
description: Fictional organizational unit for example purposes
```

Storing Information in Multiple Languages

If your directory contains a single language, you do not need to do anything special to add a new entry to the directory. However, if your organization is multinational, you may find it necessary to store information in multiple languages so that users in different locales can view directory information in their own language.

When information in your directory is represented in multiple languages, the server associates language tags with attribute values. When you add a new entry, you must provide attribute values used in the RDN (Relative Distinguished Name) without any language codes.

You can even store multiple languages within a single attribute. When you do, the attribute types are the same, but each value has a different language code.

For a list of the languages supported by Directory Server and their associated language tags, see “Identifying Supported Locales,” on page 573.

NOTE The language tag has no effect on how the string is stored within the directory. All object class and attribute strings are stored using UTF-8.

For example, suppose example.com Corporation has offices in the United States and France and wants employees to be able to view directory information in their native language. When adding directory entries, the directory administrator chooses to provide attribute values in both English and French. When adding a directory entry for a new employee, Babs Jensen, the administrator creates the following LDIF entry:

```
dn: uid=bjensen,ou=people, o=example.com Corp
objectclass: top
objectclass: person
objectclass: organizationalPerson
name: Babs Jensen
cn: Babs Jensen
sn: Jensen
uid: bjensen
personalTitle: Miss
personalTitle;lang-en: Miss
personalTitle;lang-fr: Mlle
preferredLanguage: fr
```

Users accessing this directory entry with an LDAP client with the preferred language set to English will see the personal title `Miss`. Users accessing the directory with an LDAP client with the preferred language set to French will see the title `Mlle`.

Index

NUMERICS

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X