EVIDEN

Identity and Access Management

DirX Identity

DirX Directory Manager Guide

Version 3.1.0, Edition February 2026

Table of Contents

Preface

This *DirX Directory Manager Guide* provides information about DirX Directory Manager the graphical user interface for the DirX Directory. It consists of the following sections:

- Chapter 1 provides an overview on the DirX product suite for Identity and Access Management (IAM).
- Chapter 2 provides detailed information about DirX Directory Manager.
- Chapter 3 describes how to manage the DirX Directory DSA schema with DirX Directory Manager.
- Chapter 4 describes how to manage the DirX Directory DSA database with DirX Directory Manager.
- Chapter 5 describes how to use monitoring information provided by the DirX Directory LDAP server.
- Chapter 6 describes how to use the script manager.
- Chapter 7 describes DirX Directory Manager core component.
- Chapter 8 provides information about DirX Directory Manager trace window.

DirX Identity Documentation Set

The DirX Directory Manager document set consists of the following manuals:

- *DirX Directory Manager Guide*. Use this book to obtain information about DirX Directory Manager.
- *DirX Directory Manager Release Notes*. Use this book to install DirX Directory Manager and to understand the features and limitations of the current release.

Notation Conventions

Boldface type
In command syntax, bold words and characters represent commands or keywords that must be entered exactly as shown.

In examples, bold words and characters represent user input.

Italic type
In command syntax, italic words and characters represent placeholders for information that you must supply.

[ ]
In command syntax, square braces enclose optional items.

{ }
In command syntax, braces enclose a list from which you must choose one item.

In Tcl syntax, you must actually type in the braces, which will appear in boldface type.

|
In command syntax, the vertical bar separates items in a list of choices.

...
In command syntax, ellipses indicate that the previous item can be repeated.

userID_home_directory
The exact name of the home directory. The default home directory is the home directory of the specified UNIX user, who is logged in on UNIX systems. In this manual, the home pathname is represented by the notation *userID_home_directory*.

install_path
The exact name of the root of the directory where DirX Identity programs and files are installed. The default installation directory is *userID_home_directory*/**DirX Identity** on UNIX systems and **C:\Program Files\DirX\Identity** on Windows systems. During installation the installation directory can be specified. In this manual, the installation-specific portion of pathnames is represented by the notation *install_path*.

dirx_install_path
The exact name of the root of the directory where DirX programs and files are installed. The default installation directory is *userID_home_directory*/**DirX** on UNIX systems and **C:\Program Files\DirX** on Windows systems. During installation the installation directory can be specified. In this manual, the installation-specific portion of pathname is represented by the notation *dirx_install_path*.

dxi_java_home
The exact name of the root directory of the Java environment for DirX Identity. This location is specified while installing the product. For details see the sections "Installation" and "The Java for DirX Identity".

tmp_path

The exact name of the tmp directory. The default tmp directory is /tmp on UNIX systems. In this manual, the tmp pathname is represented by the notation *tmp_path*.

tomcat_install_path
The exact name of the root of the directory where Apache Tomcat programs and files are installed. This location is defined during product installation.

mount_point
The mount point for DVD device (for example, **/cdrom/cdrom0**).

1. What is DirX?

Eviden DirX suite of highly scalable and automated IAM solutions allow customers to choose from on-premise, managed and cloud-based delivery models and to benefit from enhanced regulatory compliance and audit capabilities, greater security, higher efficiency and reduced overall costs. The DirX portfolio ranges from identity management and access governance, identity analytics and intelligence to Web access management and single sign-on, authorization, identity federation, and directory services. Analysts recognize Eviden as a world leader in role management and SAP integration.

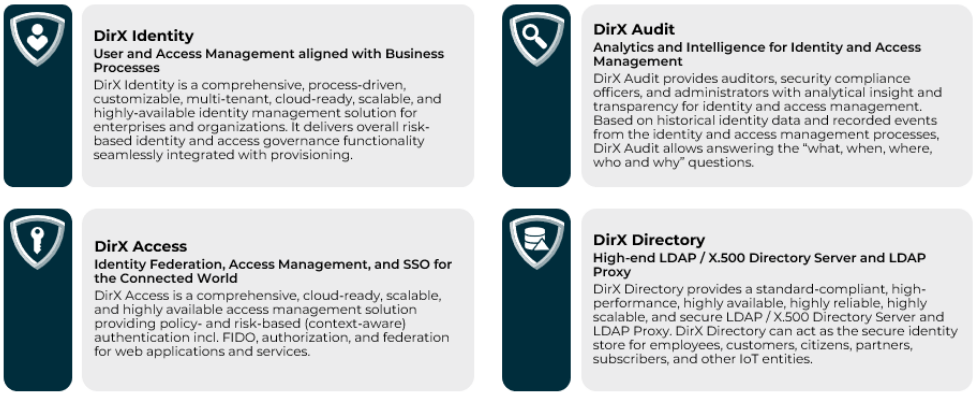


Figure 1. Figure : DirX: The integrated product suite for Identity and Access Management

With the DirX product family an integrated product suite for identity and access management solutions is provided, which consists of

- DirX Identity, a comprehensive identity management and governance solution.
- DirX Directory, the standards-compliant LDAP / X.500 directory server and LDAP Proxy.
- DirX Audit, providing analytical insight and transparency in the identity and access management processes.
- DirX Access, a policy-based Web access management, Web single sign-on, Authorization, and federation product.

The DirX products provide full coverage of the four core IAM processes

- Identity Administration Process
  The DirX Identity Business Suite delivers lifecycle management of users and organizational data, administrative and self-service management interfaces, metadirectory and provisioning capabilities.
- Entitlement Administration Process
  The DirX Identity Pro Suite includes all the features of the Business Suite plus lifecycle management of roles and entitlements, request and approval workflows, delegated administration and access certification.
- Access Process
  DirX Access protects access to resources by providing central security services for authentication, authorization, single sign on, identity federation, and Web services security.

- Intelligence Process
  DirX Audit delivers analytical insight and transparency in the identity and access management processes.

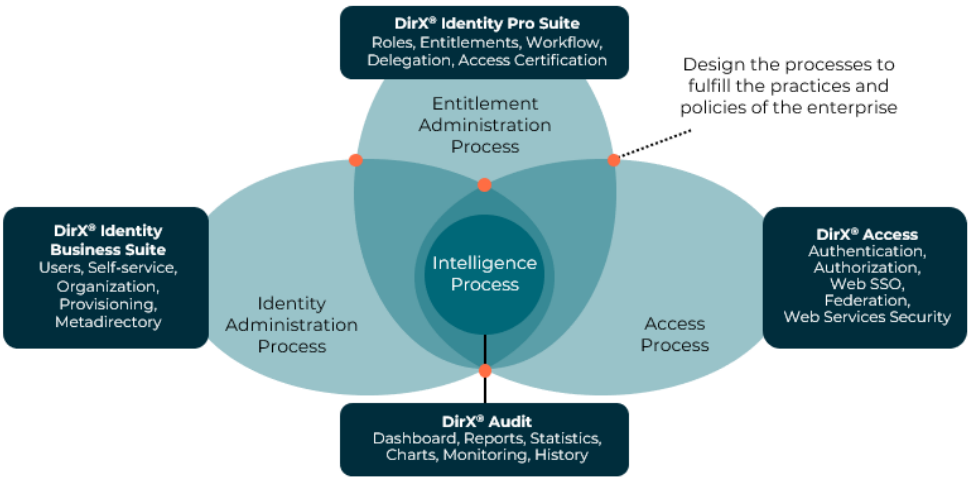


Figure 2. Figure : The DirX products provide full coverage for all 4 IAM core processes

2. What is DirX Directory Manager?

DirX Directory Manager is a Java application that provides a configurable, platform-independent administrative graphical user interface for local and remote management of DirX Directory. DirX Directory Manager is a plug-in that adds functionality to the core component. This functionality consists of:

- LDAP-based functionality
  This functional area consists of features that are designed to work (at least to some extent) with both DirX Directory and non-DirX Directory LDAP servers (note, however, that although LDAP is an Internet standard, different servers behave differently in certain details).
  The LDAP-based functionality includes:
- Managing ordinary LDAP entries (this functional area is part of the core component)
- Managing the schema
  Note that the schema and database related functionality is provided by separate plug-ins that may or may not be present in your deployment.
  See "Schema Management" for details about these functions.*
  Schema*
  Allows you to view and manage the schema.*
  Database/Indices*
  If the plug-in "Database" is installed, the schema view also allows viewing and managing indices.
- DirX Directory-specific functionality
  This functional area consists of features that are solely available with DirX Directory. It includes:
- Displaying the LDAP Root subentry
- Displaying Access Control Subentries
- Managing Collective Attribute Subentries
- Managing LDAP Configuration Subentries
- Managing LDAP SSL Configuration Subentries
- Managing LDAP Audit Subentries
- Managing Password Policy Subentries
- Managing Replication
- Monitoring administrative information provided by the LDAP Server
  Note that the monitoring functionality is provided by a separate plug-in that may or may not be present in your deployment.
  See "Monitoring Information" for details about these functions.
- A **script manager** that allows you to manage scripts and run **dirxcp** and **dirxadm** from within DirX Directory Manager, provided **dirxcp**/**dirxadm** are available locally. Moreover, you can have the script manager decode and view audit log files based on the **dirxauddecode** command (must be locally available, too).
  For details on **dirxcp**, **dirxadm** and **dirxauddecode** please refer to the documentation shipping with DirX Directory.

The script manager-related functionality may be absent because it is a separate plug-in. If present, it should be found in DirX Directory Manager's Welcome View Group.

- Finally, a "Trace Window" (which is also a separate plug-in) may allow you to enable, disable and configure trace information and display it in a window.

2.1. Getting Started

When you start DirX Directory Manager for the first time, DirX Directory Manager displays a "Welcome" view such as this one:

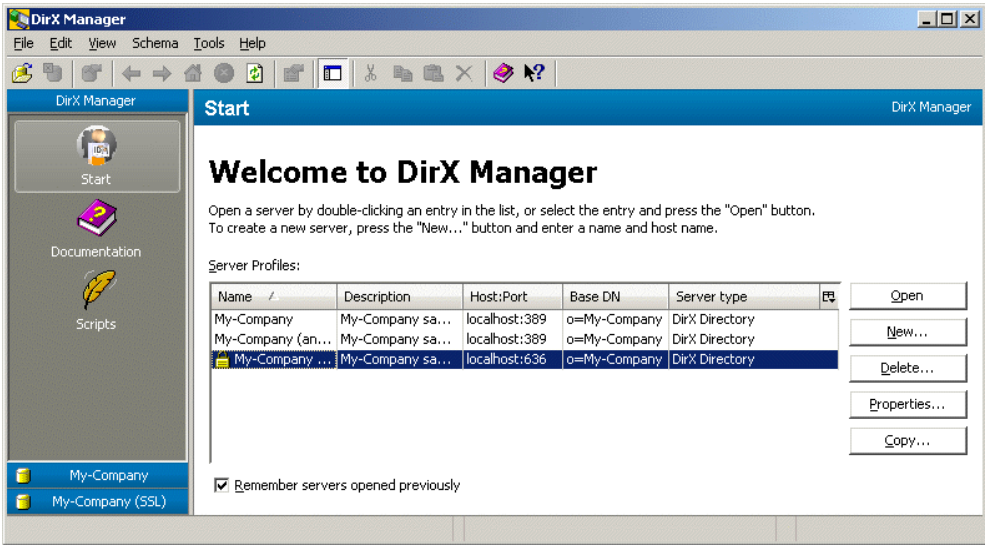


Figure 3. DirX Manager Welcome View

In the example shown here, some servers are pre-configured. Click **Properties** to modify the settings of the currently selected server according to your needs or click **New** to create a new server profile. When you opt for creating a new server profile, DirX Directory Manager opens a "Server" dialog like this one:

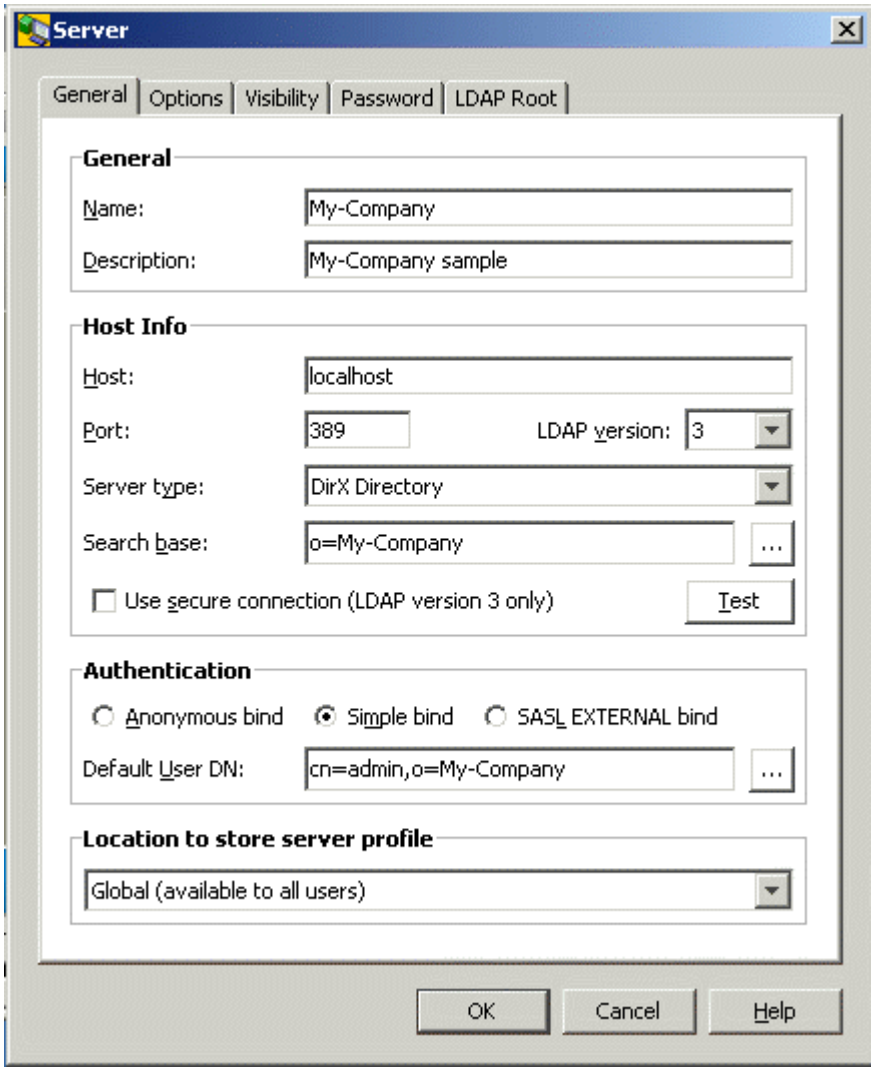


Figure 4. Server Dialog Box

The Server type field (choices are **DirX Directory** or **Other**) determines the views that are offered for the server defined by this profile. The "Basic Patterns" topics (Standard Dialogs: Server) provide further details about this dialog. Once you successfully complete this dialog, your new server profile should appear as the first entry in the Welcome dialog or as an additional entry in the list.

Now select the server profile you want to work with and click **Open** or double-click it. Unless you have decided to try anonymous access, you must now complete a "Login" dialog like this one:

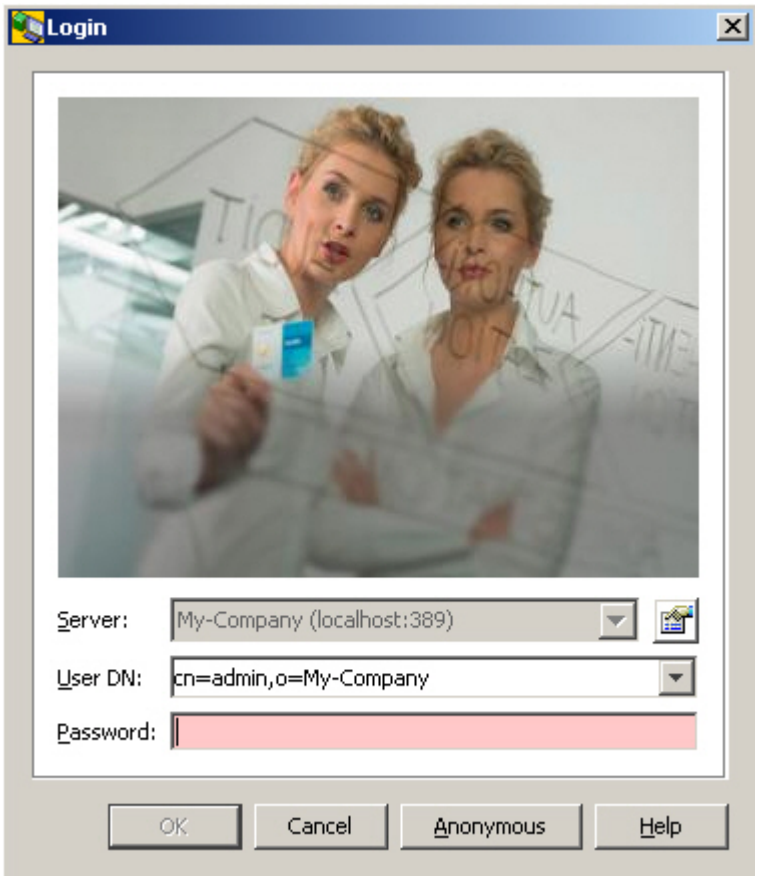


Figure 5. Login Dialog Box

The "Basic Patterns" topics (Standard Dialogs: Login) provide details on the Login dialog. Once you complete this dialog, DirX Directory Manager displays its main window in a way that shows a "View group" with a number of "views". (See the Basic Patterns: Main Window topic for more information about views and view groups.) Here is an example:

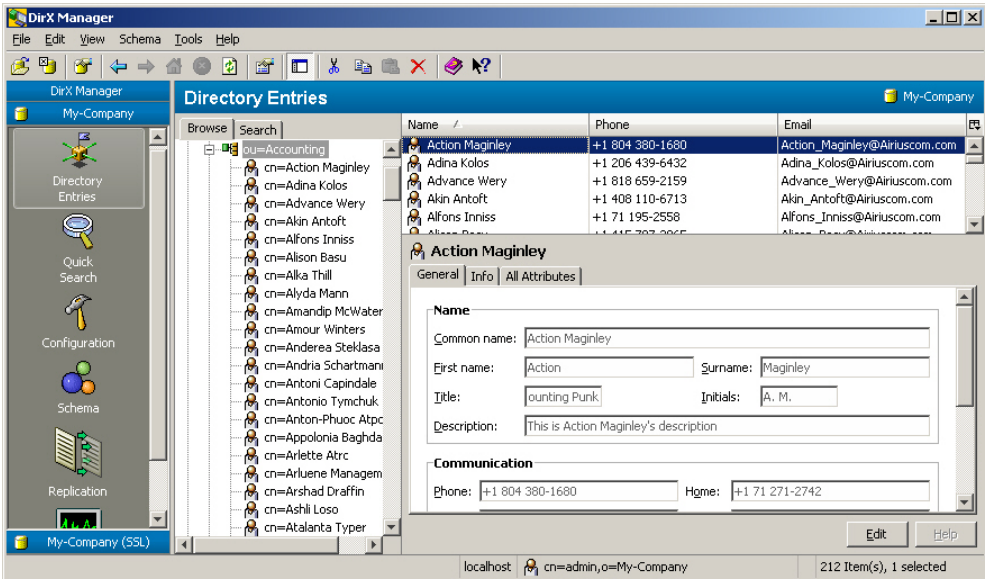


Figure 6. Screenshot of the DirX Directory Manager Main Window

In this example, you can see the following views:

- Directory Entries (the view that is currently active in the above screen shot)

- Quick Search
- Configuration (only available with server type DirX Directory)
  This view allows you to view and manage subentries such as access control, LDAP configuration and password policy subentries.
  See the "Configuration" topic for details about these functions.
- Schema (only available with server type DirX Directory)
  This view allows you to manage the schema. (See "Schema Management" for details.)
- Replication (only available with server type DirX Directory)
  This view allows you to view and manage shadowing and LDIF agreements.
- Monitoring (only available with server type DirX Directory)
  This view allows you to get monitoring and diagnostic information provided through extended operations. (See "Monitoring Information Provided by the LDAP Server" for details.)

If you want to work with several servers, you can repeat the steps described here as often as you need.

By applying the right mouse button to a view group, you can close it or move it up or down:

Figure 7. Context-sensitive Server Menu

2.2. Directory Entries View, Quick Search View

These views come with the core component.

Directory Entries
This view allows you to browse, search and manage ordinary LDAP entries. The "Core Component" topics provide details about these functions.

Quick Search
This view features a convenient, yet powerful way to find information stored in the directory. The "Core Component" topics provide details about this function.

2.3. Configuration View

The configuration view gathers various types of subentries into a virtual tree. Subentries in the narrower sense have been conceived to allow for structuring administrative information that belongs to an administrative point. In other words, they are conceptually part of the administrative entry to which they belong. This is why these subentries can only occur immediately below their administrative point. These subentries are: Access Control Subentries, Collective Attribute Subentries, Password Policy Subentries and Proxy Control Subentries.

The first level of the virtual configuration tree shows two branches:

- The context prefix(es). This branch is for the subentries in narrower sense. Below each context prefix are:
- A virtual node called Access Control Subentries
  Below this node, all access control subentries are subsumed.
- A virtual node called Collective Attribute Subentries
  Below this node, all collective attribute subentries are subsumed.
- A virtual node called Password Policy Subentries
  Below this node, all password policy subentries are subsumed. The password policy specified in this sunbentry applies to their administrative point.
- A virtual node called Proxy Control Subentries
  Below this node, all proxy control subentries are subsumed.

Note that you can edit the context prefixes here as well as in the "Directory Entries" view.

- You can create subentries by selecting the **New ... subentry...** operation from the context-sensitive menu of the virtual node or of the subentry. Note that when creating such subentries, there is no DirX Directory Manager support regarding creation of administrative points.
- You can read and edit subentries when displaying their properties in the right-hand property pane.
- The root node of the virtual tree itself provides the data that are stored in the LDAP Root.
  Just click at "Root" to display the LDAP root in the right-hand property pane.
  Alternatively, you can use the server dialog which you should find in the file menu.
- A virtual node called LDAP configuration subentries. LDAP configuration subentries can be managed.
  Below this node, you'll find the following types of subentries:
- LDAP Configuration
- LDAP Audit
- LDAP SSL Configuration
- A virtual node called Global Password Policy Subentry.
  Below this node, you'll find the global password policy subentry. The global password policy subentry specifies the password policy that applies to the DSA.

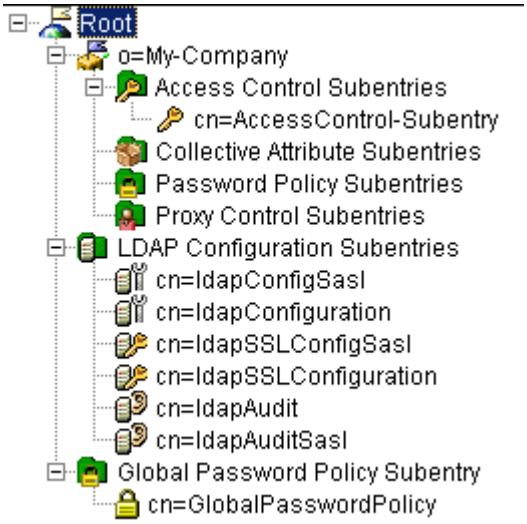


Figure 8. Virtual Configuration Tree

2.3.1. Access Control

For in-depth information on this topic, refer to the server documentation.

Access control information controls whether a user is authorized to perform particular access to the directory. The access control implementation of DirX Directory largely conforms to the respective ITU X.500 specifications (http://www.itu.int/ITU-T).

DirX Directory conforms to the general principles introduced there.

DirX Directory allows defining the scope to which an access control definition is to apply. In other words, it allows defining what collections of entries are to be affected by what access control definitions.

The items that can be protected range from attributes to collections of entries. There are also quite a number of possible permissions such as add, read, remove, etc.

Access control information (ACI) is held in ACI items, each item storing information about what users are granted/denied what types of access to what protectable items.

Initially - that is, after the installation of a server - no access control is in place. Bootstrapping the server implies establishing some primary access control definitions.

ACI Items can be contradictory. The access control decision function decides whether an access attempt is to be granted or to be denied.

2.3.1.1. General Principles

The general principles that guide access control according to the ITU X.500 specifications (http://www.itu.int/ITU-T).are:

- Information hiding
  Information in the Directory is not revealed to the user as being present unless he has permission to access it (the default result of a read operation e.g. is “no such entry” rather than “insufficient access rights”, if the access rights are actually insufficient.

However, it is possible for the user to be granted the right to be told the “truth”. In order to avoid telling a “lie” and not to have to tell the truth, the Standard would also allow configuring a result like “no information”).
The basic idea behind this principle is to keep the unauthorized user from being able to find out information on the DIT by trial and error.

- Precedence
  There may be several access control items in a list that apply to a particular user trying to access the Directory. Each access control item is given a precedence, and high precedence items overrule low precedence items. Thus if one access control item with a precedence of 50 states that everyone is denied read access to a particular attribute, and another access control item with a precedence of 51 says that Bill can read this particular attribute, the second item takes precedence over the first item and Bill will be granted read access to the attribute.
- Specificity
  Specificity means that more specific access control items overrule less specific access control items of the same precedence. Thus if one access control item says that Bill is denied read access to all attributes, but another item of the same precedence says that Bill can read the telephone number attribute, then Bill will have read access to the telephone number attribute (the protected item “telephone number” is more specific than “all attributes”). Similarly, if one access control item says that everybody is denied read access to the telephone number, but another item of the same precedence says that Bill can read the telephone number attribute, then Bill will have read access to the telephone number attribute (the user class “Bill” is more specific than the user class “anybody”).
- If in doubt, deny
  Denial beats Grant: A denied access overrules a granted access if both have the same precedence and specificity. Denial is the default: access that is not granted explicitly is denied.

2.3.1.2. Scope of Access Control Definitions

See also Administrative Authority Model.

While EntryACI items apply only to the single entry in which they are held and SubentryACI items apply only to the subentries that are immediately below the administrative point to which they belong, PrescriptiveACI items cover Access Control *Areas*. Access Control Areas include

- A complete autonomous administrative area (AAA)

or

- A fraction of it; possible fractions are:
- An “access control specific area” (ACSA);
  that is, the AAA can be partitioned into a number of non-overlapping access control specific areas, each of them comprising a subtree. The start of an ACSA is defined through the value “Access Control Specific Area” of the operational attribute “Administrative Role”. Leaf entries or new ACSAs mark the end of an ACSA.

- An “access control inner area” (ACIA);
  that is, the AAA or ACSA can contain a number of nested access control inner areas, each of them comprising a subtree – the lower nesting level may alter some rules imposed by an upper level for its own subtree (however, only for the subtree to which the inner area applies).
  The start of an ACIA is defined through the value “Access Control Inner Area” of the operational attribute “Administrative Role”. Leaf entries or new ACSAs mark the end of an ACIA.
- A “Directory Access Control Domain” (DACD, also called “entry collection”)
  which forms a subset of an ACSA or ACIA that is determined by a “subtree specification”.

The X.500 standard specifies two levels of complexity (per access control specific area):

- The quite powerful “basic access control” (BAC) scheme
- The rather simple “simplified access control” (SAC) scheme

DirX Directory treats SAC in the same way as BAC, however. According to the X.500 standard, the simplified SAC scheme would work by ignoring the existence of Entry ACI attributes and access control inner areas (ACIAs). This means that all entries in an administrative area would be protected solely by the Prescriptive ACI attributes held in the subentries below their Administrative Point, and subentries would be protected by the single Subentry ACI attribute in their administrative entry.
Doing without Entry ACIs and access control inner areas is possible with BAC, too, of course.

2.3.1.3. Protectable Items

The following items can be subject to access control:

- All entries within the scope of the ACI item (yes/no)
- All user attribute types of all those entries (yes/no)
- All user attribute values of all those entries (yes/no)
- All attribute types specified in a list of attribute types
- All values of all attributes specified in a possibly different list of attribute types

2.3.1.4. Permissions

Permissions may be granted or denied. The following permissions are possible:

- Add
  For entries: controls the ability to create an entry in the DIT (subject to the access controls on all attributes and values to be placed in the new entry). In order to add an entry, permission to add its mandatory attributes and their values must at least be granted.*
  For attributes*: controls the ability to add an attribute (subject to the ability to add all specified attribute values).*
  For attribute values*: controls the ability to add an attribute value to an existing attribute
- Browse

For entries only: controls whether directory operations (i.e. list and search operations) can access entries that they do not specifically name

- Compare (for attributes and their values only)*
  For attributes and their values*: controls whether attributes and values can be used in compare operations
- Disclose on Error
  For entries: controls whether the entry's name can be returned in an error or empty result. If not granted, the Directory returns an error as if the entry did not exist.*
  For attributes*: controls whether the attribute's presence can be disclosed in attribute or security errors. If not granted, the Directory returns an error as if the attribute did not exist.*
  For attribute values*: controls whether the attribute value's presence can be disclosed in attribute or security errors. If not granted, the Directory returns an error as if the attribute did not exist.
- Export
  For modify DN operations only: controls whether an entry can be moved (with its subordinates, if any) from its present position in the DIT to another position in which it has a different superior entry. The Export permission must be held by the old superior entry. If the last RDN of the entry being moved is changed, the entry must also possess the Rename permission.
- FilterMatch
  For attributes and their values only: controls whether the attributes and values can be matched with corresponding values in a search filter. If the cannot, the filter must be evaluated as if the a
- Import
  For modifyDN operations only: applies when an entry is being moved (with its subordinates, if any) to another position in the DIT where it has a different superior entry: The Import permission must be held by the new superior entry. The entry must itself grant Export permissions for success.
- Modify
  For entries only: controls whether the information contained in an entry can be modified using a modify operation. Note that appropriate attribute and value permissions must also be granted in order to modify any attributes and values in the entry. (Modifying the RDN is no affected by the Modify permission.)
- Read
  For entries: controls whether Directory read or compare operations that specifically name an entry can read that entry. Further permissions are required to access the attributes and their values:*
  For attributes and their values*: controls whether read or search operations can return attributes and values as entry information.*
  Note: Since LDAP does not know a read operation (in LDAP, a read is done thru a "base object search"), you need to grant browse permission.*
- Remove
  For entries: controls the ability to remove an entry from the DIT regardless of controls on attributes or attribute values within the entry.*
  For attributes*: controls the ability to remove an attribute and all of its values.*

For attribute values*: controls the ability to remove an attribute value from an existing attribute.

- Rename
  For entries only: controls whether an entry can be renamed with a new RDN subject to the consequential changes to the distinguished names of subordinate entries. (It may not be possible to rename an entry when there is a subordinate reference that is subordinate to the entry.) Note, however, that permissions within subordinate entries are not taken into account when evaluating permissions for this operation.
- ReturnDN
  For entries only: controls whether an entry's distinguished name can be returned in an operation result. If this permission is not granted, an alias name can in some cases be returned; if it is not possible to return an alias, information about the entry is withheld.

2.3.1.5. Permissions vs. LDAP or DAP Operations

Normally there is more than one permission required in order to be able to perform a single directory operation; for example, attribute value-related permissions do not imply the belonging attribute type permissions and entry-related permissions alone are not necessarily sufficient to access any information in the entry. The following table gives an overview of what permissions are required for what directory operation. The table does not contain the permission DiscloseOnError, which is not really required for any directory operation:

Table 1. Table : Permissions vs. LDAP or DAP Operations

Directory Operation (LDAP/DAP)	**Permissions that need to be granted for 'Entry' Protected Item**	**Permissions that need to be granted for Attribute Type Protected Items**
Compare/Compare	Read	Compare for attribute type
-/Read There is no LDAP read operation. See note on base object searches below	Read ReturnDN (only if an alias name is not available)	Read for each attribute type returned (note that it is possible to just request attribute types in a read operation)
One-level search/List	Browse and ReturnDN for each subordinate	None

Directory Operation (LDAP/DAP)	Permissions that need to be granted for 'Entry' Protected Item	Permissions that need to be granted for Attribute Type Protected Items
Base object search or subtree search/Search	Browse for each entry in scope of Search. **Note that *reading* entries through LDAP requires Browse permission, since LDAP reads entries through base object searches!** ReturnDN for each entry (only if an alias name is not available)	FilterMatch for each attribute type and value used to evaluate the filter Read for each attribute type returned (note that it is possible to just request attribute types in a search operation)
Add/AddEntry	Add	Add for each attribute type
Delete/RemoveEntry	Remove	None
Modify/ModifyEntry	Modify	Add for all attribute types added Add for all attribute values added Remove for all attribute types removed
Rename/ModifyDN	Rename if operation only modifies RDN else Export at old name and Import at new name	None

2.3.1.6. Access Control Information (ACI)

The information stored in an ACI item includes:

- An identification tag, which is a text string assigned by the creator of the item, and which serves to uniquely identify the item
- The precedence level, which is an integer, and is used by the Access Control Decision Function when evaluating a set of ACI items. Higher precedence items override lower precedence items. The precedence level ranges from 0 to 255, zero being the lowest one. If a type of access was denied at precedence 100, but granted at precedence 200, access would be granted. This way administrators can use the precedence level when delegating authority to an “access control inner area” ACIA (see below). They may set low precedence defaults that the inner authority may selectively override, or they may set high precedence policies that cannot be overridden (except thru higher specificity)

- The authentication level, which specifies the minimum level of authentication that the user must have undergone before this ACI item may grant him access. It takes one of three values, strong, indicating that strong authentication must have been performed on the user, simple, indicating that a name and password or protected password must have been presented by the user, or none, indicating that no authentication is needed (identification only is classified as none). The authentication level is optionally qualified by an integer, which if present in the ACI item, must be exceeded by the user when he is authenticated (how this is done, or what it means, is not specified in the Standard - and it is not supported by DirX Directory)

and either

- “Item first” permissions which lists a set of protected items, and the set of users, all potentially with varying permissions, who may or may not access these items (note that a given user has the same permissions for all the protected items)

or

- “User first” permissions which lists a set of users, and the protected items, all potentially with varying permissions, that the users may or may not access (note that a given protected item grants or denies the same permissions to all the users)

User first and Item first are equivalent; in some cases the administrator may find User first to be the more convenient way to put an access control information, in others he may prefer Item first.

Here is an example of an ACI item:

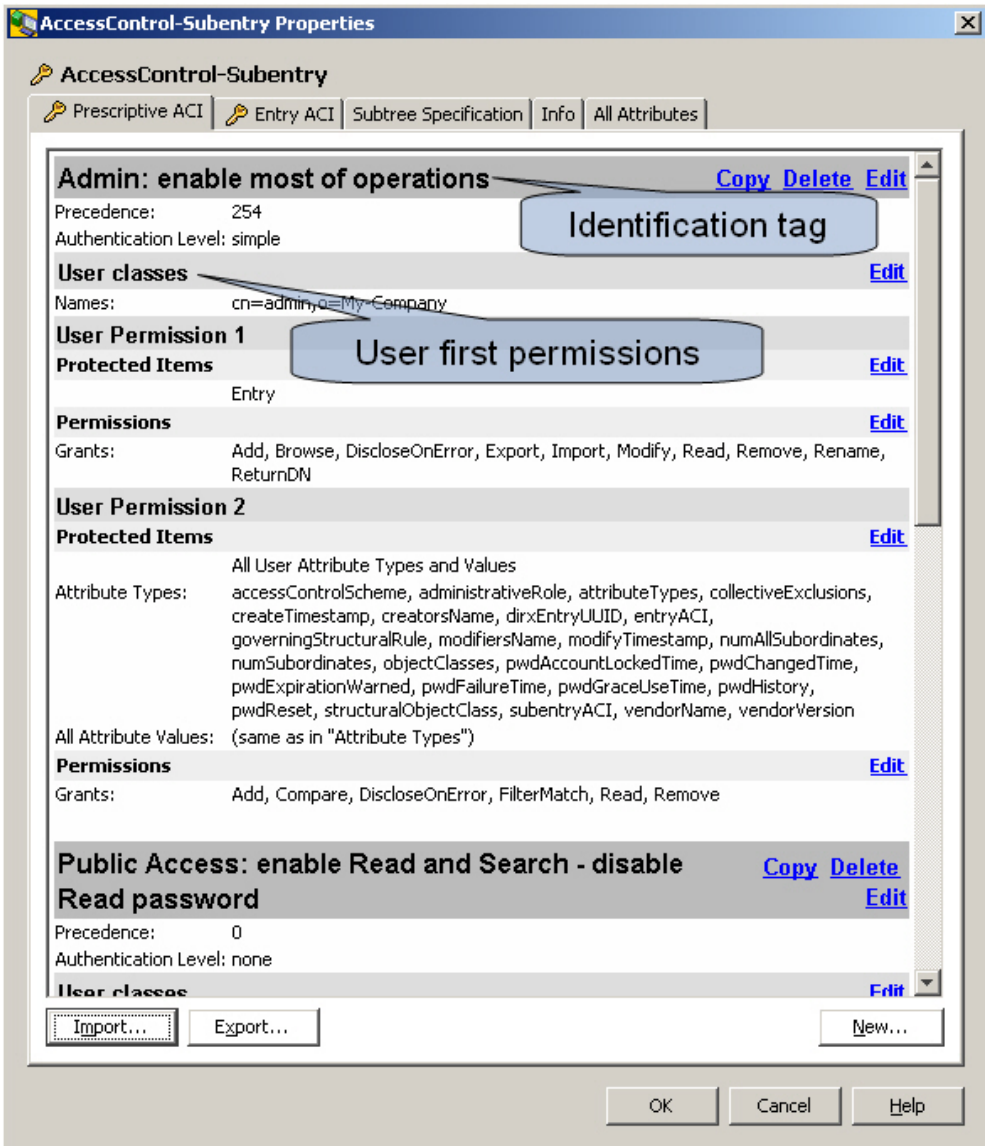


Figure 9. AccessControl-Subentry Properties Dialog Box with ACI Item

Access control information is held in so-called “ACI operational attributes” (multi-valued, the values being called ACI items). There are three types of ACI attributes:

- Access control operational attributes protecting ordinary entries
- EntryACI
  EntryACI operational attributes control access to the single entries in which they are held. It is also possible to control access to collective attributes that appear to be held in the same entry.
- PrescriptiveACI
  PrescriptiveACI attributes can only be held in subentries. They control access to the entries (and their attributes) that are within the scope of the subtree specification held in the same subentry. This scope is also referred to as ”Directory access control domain” (DACD). PrescriptiveACI attributes thus hold the access control policies that apply to the access control domain, but they do not protect the attributes held in the subentry itself. EntryACI or SubentryACI attributes have to be used for this.
- Access control attributes protecting subentries

- SubentryACI
  SubentryACI operational attributes are access control attributes that can only be held in administrative entries. They control access to the subentries immediately below the administrative entry in which they reside (they actually control access to the subentries and to the operational and user attributes held in the subentries)

PrescriptiveACI attributes can protect EntryACI attributes. SubentryACI attributes can protect PrescriptiveACI attributes.

2.3.1.6.1. [.indexref][.indexref]##Prescriptive ACI

Access control subentries display panes like these:

Variant1, read-only (property pane):

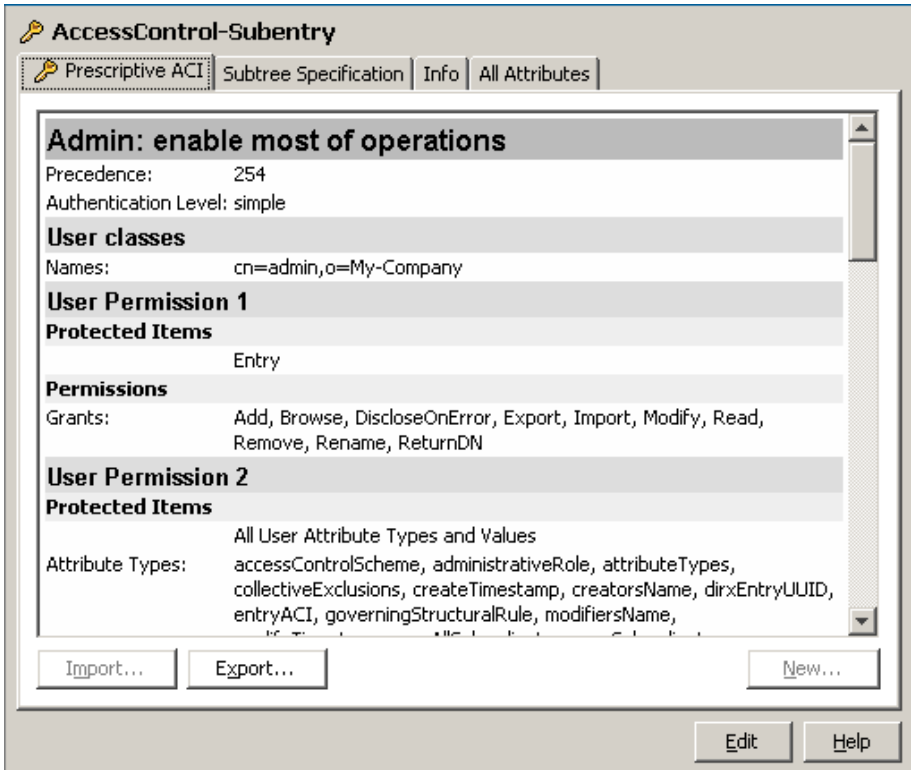


Figure 10. AccessControl-Subentry (read-only property pane)

Variant 2, editable (property pane after clicking the Edit button at the bottom of the pane (shown here) or property dialog (not shown)):

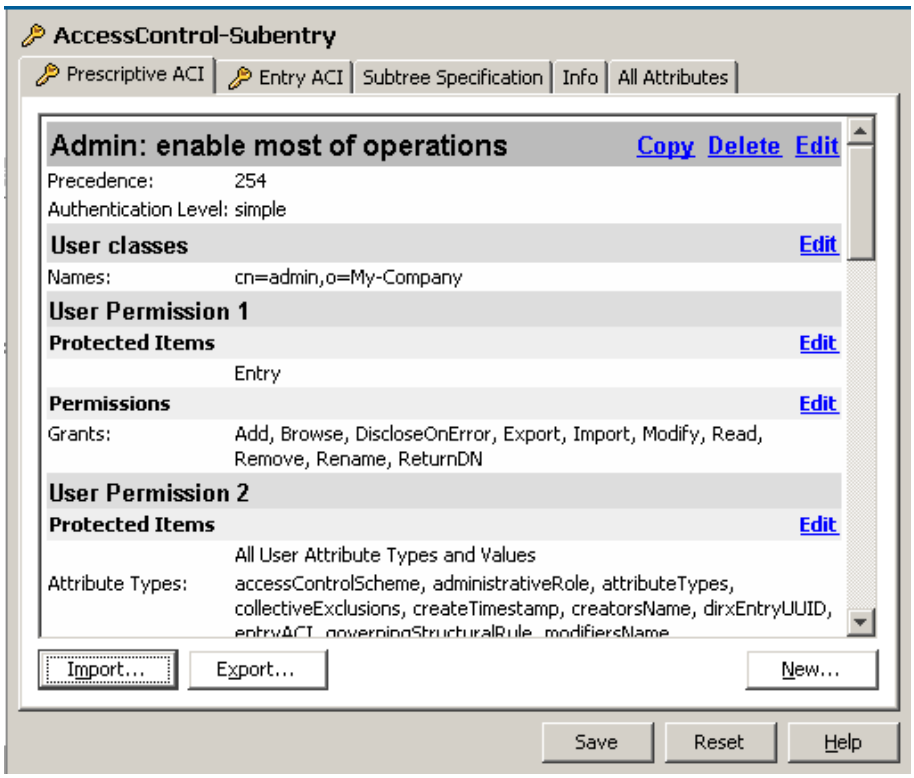


Figure 11. AccessControl-Subentry (editable property pane)

Notes:

- In order to be able to manage an access control item in a pane, you first need to click the Edit button at the bottom of the pane. This action reveals various links such as Copy, Delete and Edit. Also, if no Prescriptive ACI has been specified yet, an additional Tab "Prescriptive ACI" will appear (otherwise this tab is visible in the first place). The respective online help is incorporated directly into the various property dialogs that apply to the Copy, Edit and Delete links as depicted in above sample. You can also create new access control subentries.
- Prescriptive ACI operational attributes can only be held in subentries. They control access to the entries (and their attributes) that are within the scope of the subtree specification held in the same subentry. Subentries holding prescriptive ACI operational attributes can occur only immediately below the administrative point they belong to. This administrative point must be an entry marked as ACSA or ACIA
- You use access control subentries to define access control information that is not intended to apply to just one single entry. When you double-click an access control subentry, DirX Directory Manager will display a *dialog* like the *pane* depicted above.
- The tab Subtree Specification is described in a separate chapter.

2.3.1.6.2. [.indexref]##Entry ACI

Entry ACI operational attributes control access to the single entries in which they are held. It is also possible to control access to collective attributes that appear to be held in the same entry.

Entries that have an Entry ACI display an additional tab like this one:

Variant1, read-only (property pane):

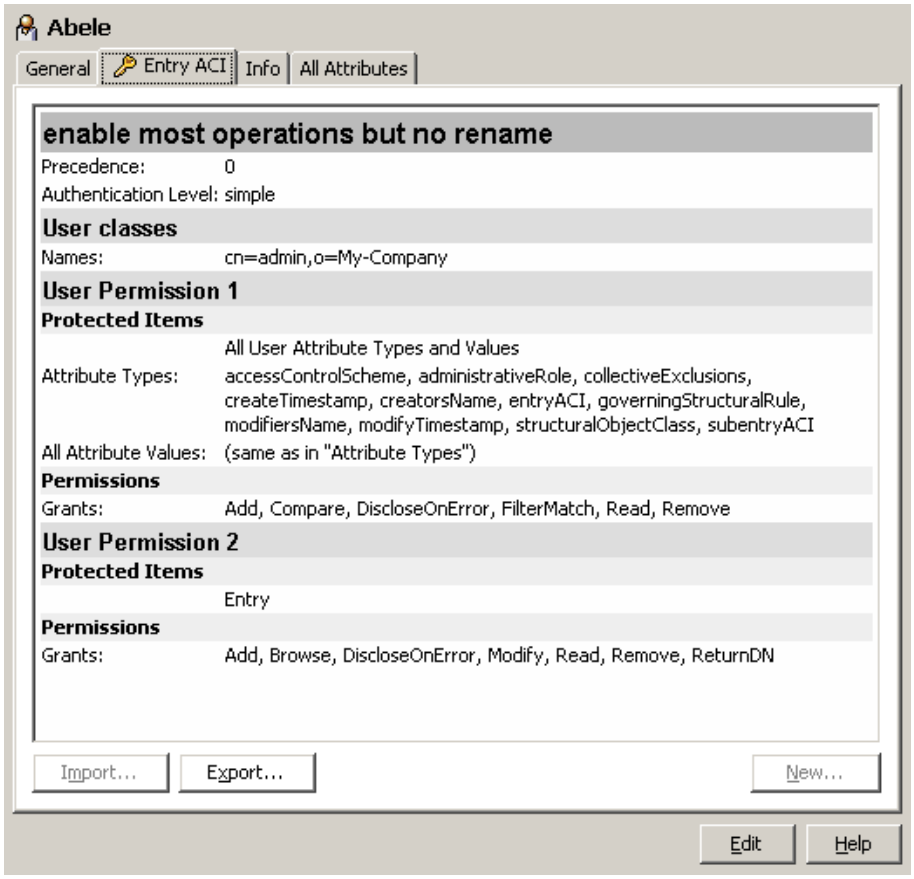


Figure 12. Entry ACI (read-only property pane)

Variant 2, editable (property pane after clicking the Edit button at the bottom of the pane (shown here) or property dialog (not shown)):

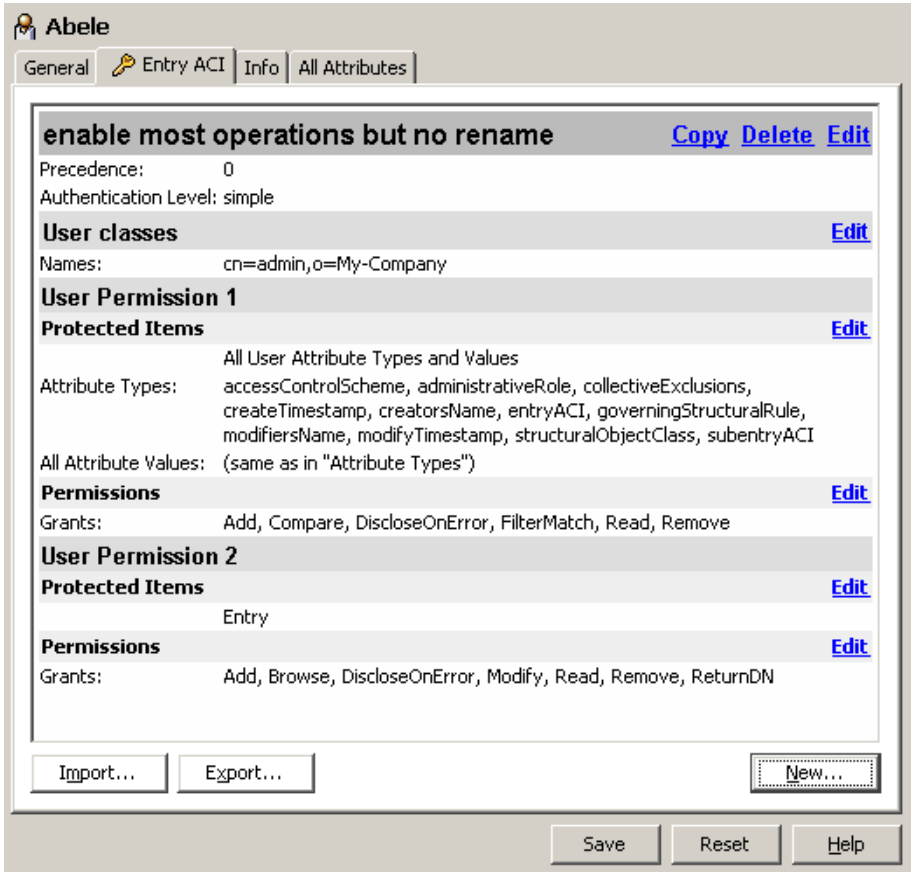


Figure 13. Entry ACI (editable property pane)

Notes:

- In order to be able to manage an access control item in a pane, you first need to click the Edit button at the bottom of the pane. This action reveals various links such as Copy, Delete and Edit. Also, if no Entry ACI has been specified yet, the Tab "Entry ACI" will appear (otherwise this tab is visible in the first place). The respective online help is incorporated directly into the various property dialogs that apply to the Copy, Edit and Delete links as depicted in above sample.
- When you double-click an entry that has an Entry ACI, DirX Directory Manager will display a *dialog* like the *pane* depicted above.

2.3.1.6.3. [.indexref]##Subentry ACI

Subentry ACI operational attributes are access control attributes that can only be held in "administrative entries". They control access to the subentries immediately below the administrative entry in which they reside (they actually control access to the subentries and to the operational and user attributes held in the subentries).

Entries having a Subentry ACI display an additional tab like this one:

Variant1, read-only (property pane):

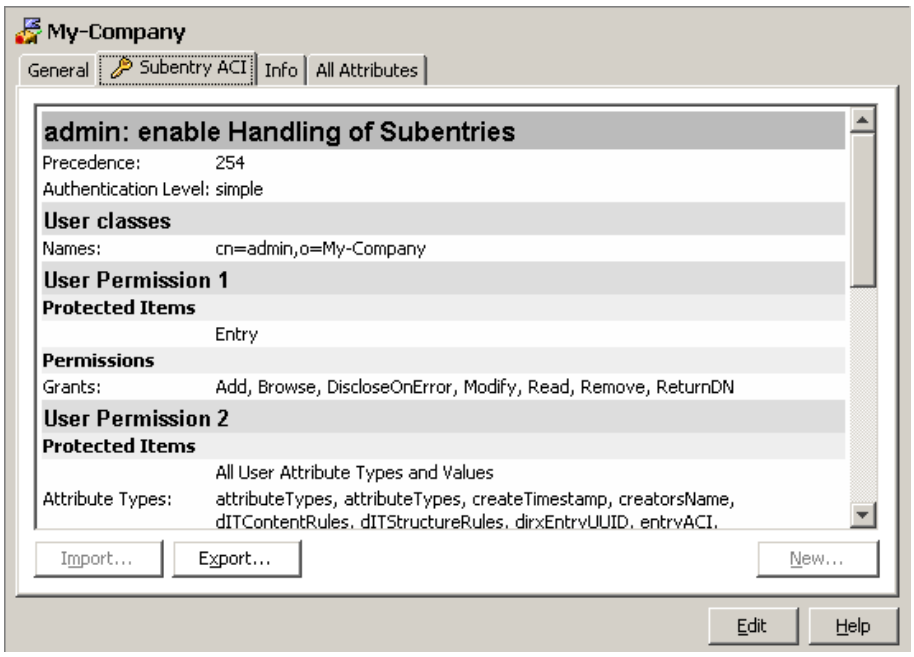


Figure 14. Subentry ACI (read-only property pane)

Variant 2, editable (property pane after clicking the Edit button at the bottom of the pane (shown here) or property dialog (not shown)):

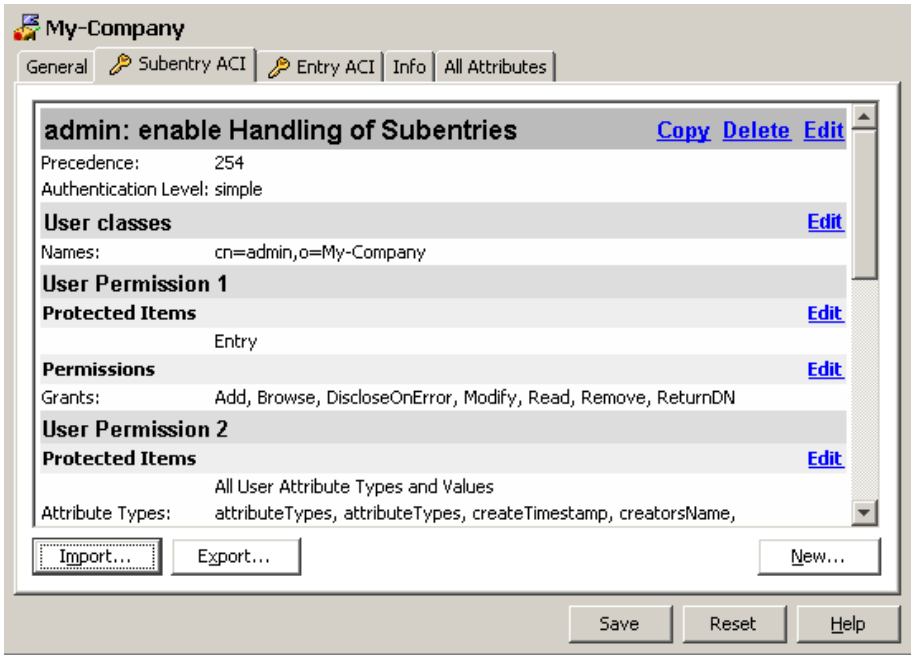


Figure 15. Subentry ACI (editable property pane)

Notes:

- In order to be able to manage an access control item in a pane, you first need to click the Edit button at the bottom of the pane. This action reveals various links such as Copy, Delete and Edit. Also, if no Subentry ACI has been specified yet, the Tab "Subentry ACI" will appear (otherwise this tab is visible in the first place). The respective online help is incorporated directly into the various property dialogs that apply to the Copy, Edit and Delete links as depicted in above sample.

- When you double-click an entry that has a Subentry ACI, DirX Directory Manager will display a *dialog* similar the *pane* depicted above.

2.3.1.7. Setting up Access Control

Initially, a DirX Directory server has no data and thus no access control information. In this stage, an anonymous user who is accessing the server is granted full control. Once the server has been bootstrapped, access control will take effect.

When installing a server and then loading the sample database that ships with DirX Directory using the **dirxload** command, you should find an administrator cn=admin, o=My-Company or similar. The subentryACI attribute of the context prefix is should say something like: "cn=admin,o=My_Company is granted pretty much everything, while anonymous users are only granted read access". If you delete this entry, which is the only one that has administrative rights, you can no longer administer the server and must reinstall it. So, provided this entry is there, logging in as cn=admin,o=My-Company entitles you to create:

- New ACSAs, ACIAs and DACDs (not supported with the current release of DirX Directory Manager)
- New administrators with varying access rights (including identical ones)

Note that - once the server has been bootstrapped - using the DirX Directory Administration Program **dirxadm** is only possible for administrators whose entry is administered in the DADM attribute of the root DSE.

2.3.1.8. The Access Control Decision Function (ACDF)

The access control items that actually apply to an entry are derived from the access control information that is stored within the entry (as EntryACI attributes), plus the PrescriptiveACI attributes from all the enclosing access control domains (DACDs). In order to locate the latter, the server software must logically traverse up the DIT from the entry towards the root, searching for inner and specific administrative points which have an 'administrative role' attribute value of 'access control inner area' or 'access control specific area' respectively, and stopping when the first specific administrative point is encountered. Each access control subentry below these administrative points is then checked to see if the entry is included within scope of the subtree specification.

The ACDF works according to the following principles:

- All non-relevant access control items are discarded. This includes items that do not include the user, either directly or indirectly, in his user class, and items that do not include the desired protected item or the requested permission. Note that items that grant access to the user, but which require a higher authentication level than that possessed by the user are discarded. Conversely, items denying access to the user, but requiring a higher authentication level than that possessed by the user are kept.
- Only keep those remaining item that have the highest precedence level.
- Keep the item(s) with the most specific user class, according to the precedence: name = thisEntry > userGroup > subtree > allUsers.

- Keep the item(s) with the most specific protected item, for example, attributeValue > allAttributeValues.

Access is granted only if all the remaining items grant access. If there are no items remaining, or at least one of them denies access, then access is denied.

2.3.2. Collective Attribute Subentry

You use collective attribute subentries to define attributes that are common to multiple entries. You can

- Create a collective attribute subentry (right-click any collective attribute subentry or the virtual node above it)
- Delete a collective attribute subentry (right-click the collective attribute subentry to be deleted)
- View and modify a collective attribute subentry (right-click or double-click the collective attribute subentry to be viewed/modified)

When you double-click a collective subentry, DirX Directory Manager displays a dialog like this (a pane that looks almost the same is likely to be configured, too):

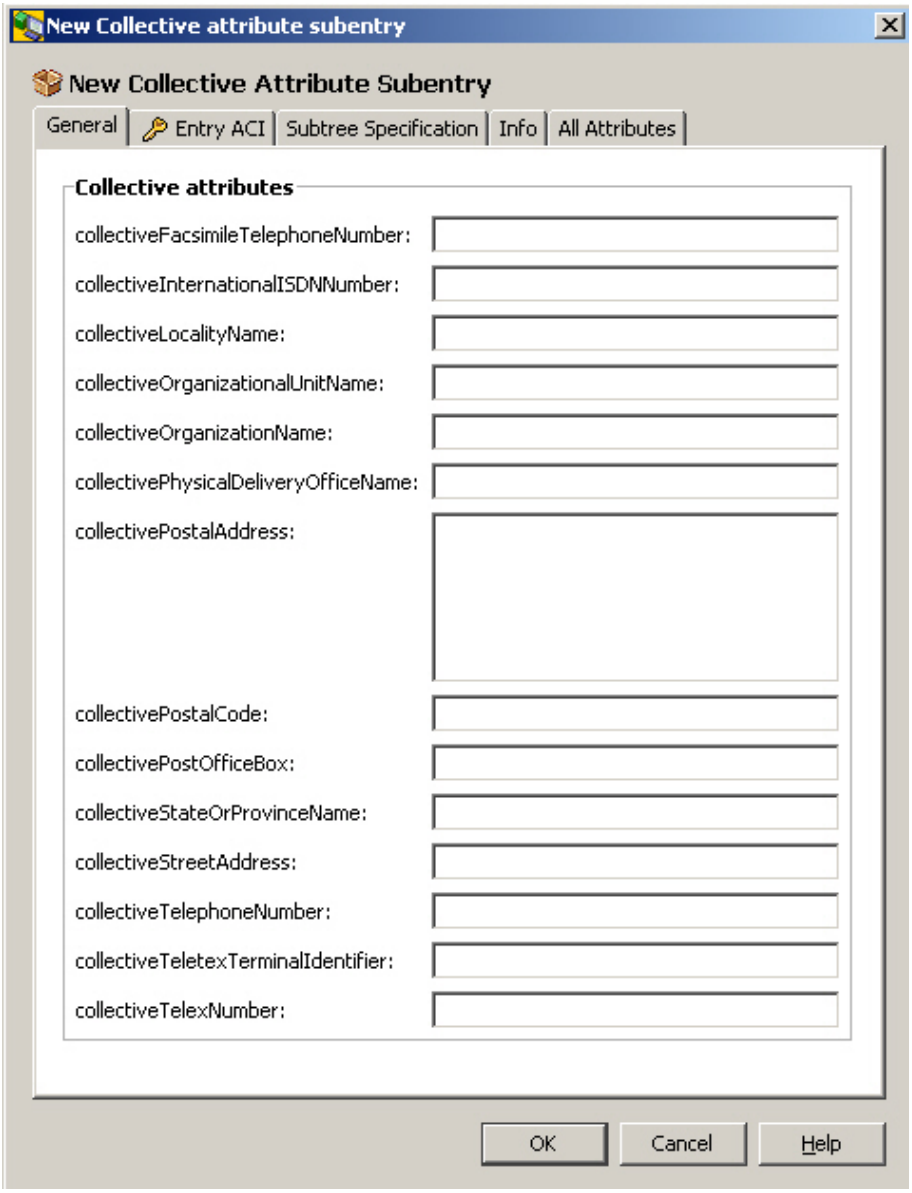


Figure 16. New Collective attribute subentry dialog box

Notes:

- The fields offered by this dialog vary depending on what attributes are administered collective in the directory schema. If the Schema plug-in is present, you can manage collective attributes using DirX Directory Manager's schema management functionality.
- The example dialog shows the context-sensitive menu that pops up when you right-click a field that is recognized as a phone number (Check Number, Cut, ...).
- The tab Subtree Specification is described in a separate chapter.
- Subentries holding collective attributes can occur only immediately below the administrative point to which they belong.

2.3.3. Proxied Authorization Control

For in-depth information on this topic, please refer to the server documentation.

DirX Directory Mananger's functionality on proxy authorization is provided through the right mouse key applied to the respective nodes: Configuration view -> Root -> Context prefix (for example, o=my-company) -> proxy control subentries (and nodes below).

You can find more information in these topics:

- General Principles
- Administering the Policy for Proxy Control

2.3.3.1. General Principles

Proxy authorization allows a client to request that an operation be processed under a provided authorization identity instead of under the current authorization identity associated with the authentication Identity, i.e. with the DN provided in the context of the bind operation. The Proxy Authorization Control provides a mechanism for specifying an authorization identity on a per-operation basis, benefiting that need to perform operations efficiently on behalf of multiple users. The model of trust in such a proxy environment is a Single-Sign-On scenario: The ldap client – typically service like applications – has performed the authentication of the end user and uses the proxy authorization control to transport the authID to the DSA

DirX Directory V8.0 and newer supports this feature according to RFC4370 and also supports the non-standard Proxied Authorization method as defined by draft-weltman-ldapv3-proxy-XX.txt which is implemented as the default Proxy Authentication by several LDAP client libraries (for example, Netscape LDAP-C-API and JDKs) The differences are automatically detected and handled by the servers and need no further configuration. The feature itself is completely controlled and configured via attribute settings in special proxy-authorization-subentries that define which users are entitled to act as a proxy and for which users he can play the role of the assumed identities.

Although it is not necessary for the server to accept and handle Proxy-Auth controls it is recommended to signal the ability of the feature to the LDAP clients by setting the following OIDs in the 'supportedControls' attribute of the LDAP root subentry:

- 2.16.840.1.113730.3.4.18 (support for RFC4370)
- 2.16.840.1.113730.3.4.12 (support for the 'old' Weltman draft)

To support the RFC4370 Proxy Authorization the following technical requirements are fulfilled:

- The DirX Directory LDAP Server understands the LDAP V3 Control defined by the OID 2.16.840.1.113730.3.4.18
  For compatibility with old implementations the OID 2.16.840.1.113730.3.4.12 is recognized as well and the corresponding controlValue is handled (see below)
- The Control is accepted in search, compare, modify, add, delete, or modifyDN or extended operation request message.
- The criticality flag in the request must be present and must be set to TRUE (if the flag is absent or set to FALSE the LDAP Server returns a 'protocolError' error.
- The ControlValue must be present. DirX Directory supports only the dnAuthID form of

the authzId as defined by RFC2829 as

authzId = dnAuthzId / uAuthzId
; distinguished-name-based authz id.
dnAuthzId = "dn:" dn
dn = utf8string ; with syntax defined in RFC 2253

- The uAuthzId is not supported. An error 123 is returned in this case.
- If the server recognizes the requested authorization identity, and the client is authorized to adopt the requested authorization identity, the request will be executed as if submitted by the proxy authorization identity; otherwise, the result code 123 is returned.

2.3.3.2. Administering the Policy for Proxy Control

When performing an operation initiated by an LDAP request carrying a Proxy Authorization Control, the DSA must decide whether the use of the control is allowed for the operation.

There are two terms used to describe this permission:

- The proxyControlOwner is the user that performed the bind operation initiating the session in which's context the operation containing the proxyControl occurs.
- The proxiedUser is the user that is named in the proxyControl value of the operation; on behalf of this user the operation shall be performed.

The control of these permissions is achieved with a Subentry - the proxyControl Subentry - The attributes of this subentry define who (the entry that performs the bind operation to the directory service. i.e.the proxyOwner) may perform directory operations on behalf of whom (the entry named in the proxy authorization Control, i.e.the proxiedUser):

A proxyControl Subentry may be created below an ADM-Point, if the Administrative Role of the ADMP carries the value ProxyAuthenticationSpecificArea (PASA).

The proxyControlUser Attribute of proxyControl Subentries defines the "who is allowed to use the proxy Control". The semantic is a full DN, i.e. the proxyControlOwner has to name an individual entry and only this entry is allowed to use the control.

ProxyControl Subentries inherit the Subtree Specification attribute from the subentry definition. This Subtree Specification defines the "for whom is proxyControlOwner allowed to use the control". The proxiedUser contained in the Control value has to match the Subtree Spec of the Subentry for the given proxyOwner.

Several users may be listed as proxyControlOwners as the attribute is defined as multi-valued, however all these proxyControlOwners are permitted to use the same set of proxiedUsers.

If no the proxyControl Subentry exists, no user is allowed to use the proxyControl.

Users that have performed an anonymous bind are never allowed to use the proxyControl.

Example: the following proxy Control Subentry cn=proxy-for-sales,o=my-company defines the user cn=proxyadmin,ou=usermanagement,o=my-company as a Proxy Control Owner;

that is, as a user who is allowed to use the proxyAuthorization Control in LDAP operations. Given that the subtree specification of this subentry is set to “entire administrative area”, there is no restriction concerning the UserDN contained in the Control values of the LDAP operations.

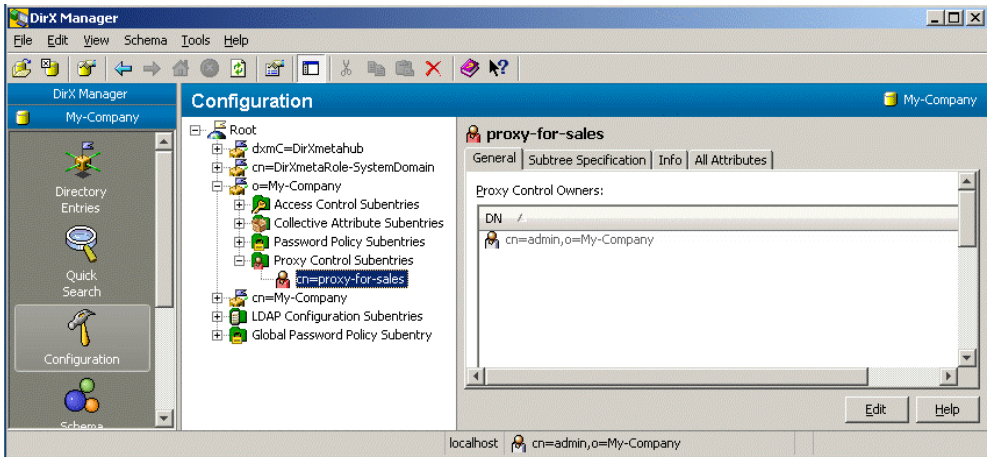


Figure 17. DirX Manager Screenshot with the Proxy Control Subentry cn=proxy-for-sales General Tab

If the permission to use the proxyAuthorization Control should be restricted, the subtree specification of the subentry cn=proxy-for-sales,o=my-company needs to be set to the respective subtree. In the example below, the subtree specification carries the value ou=sales. The effect is that the Proxy Control Owner cn=proxyadmin,ou=usermanagement,o=my-company may use the control, but the Control value in the LDAP operations must match the subtree ou=sales,o=my-company. If an LDAP operation initiated by the Proxy Control Owner carries, for example, the authorization ID “dn:cn=abele,ou=development,o=my-company”, the DSA rejects the operation.

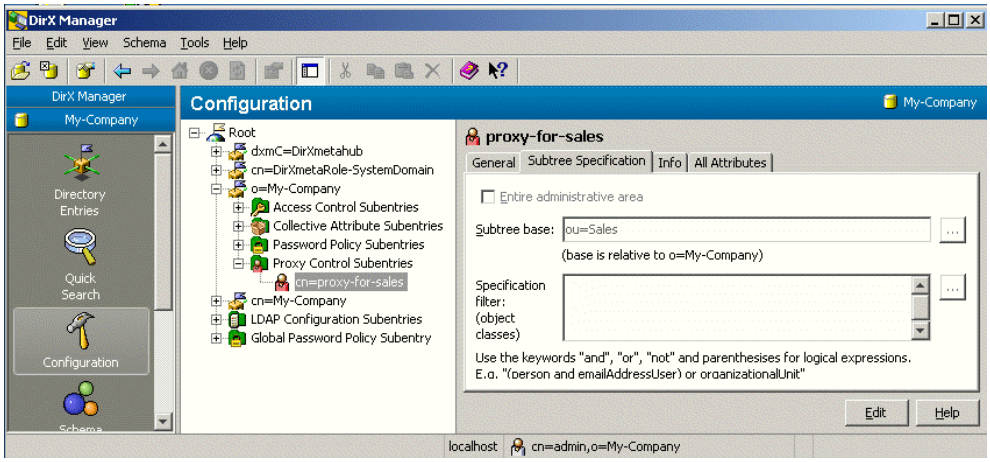


Figure 18. DirX Manager Screenshot with the Proxy Control Subentry cn=proxy-for-sales Subtree Specification Tab

2.3.4. LDAP Audit Subentry

The LDAP Audit Subentry stores audit parameters for a DirX Directory LDAP server. You can

- Add an LDAP audit subentry (right-click any LDAP audit subentry or the virtual node above)

- Delete an LDAP audit subentry (right-click the LDAP audit subentry to be deleted)
- View and modify an LDAP audit subentry (right-click or double-click the LDAP audit subentry to be viewed/modified)

DirX Directory Manager organizes the information in an LDAP audit subentry into the following tabs:

- General
- Operations
- All Attributes

2.3.4.1. Tab: General

Use the LDAP audit subentry's **General** tab to configure audit startup/shutdown, audit log file generation, and enabling or disabling session tracking.

Note that a missing property value means that the server default is effective for that property. As server defaults can vary between different builds of the server, please refer to the LDAP server documentation for the defaults applying to the server you are contacting currently.

The available fields are:

Startup Status
Specifies whether LDAP server auditing is to be enabled at LDAP server startup. The available fields are:

- **On**
  Enables LDAP server auditing.
- **Off**
  Disables LDAP server auditing.

Audit path
Specifies the initial pathname of the directory in which the LDAP server's audit log file with the name **audit.log** is located.*
Note: if uncertain, leave this field empty*

Max. records per file
Specifies the maximum number of audit records of the LDAP server's audit log file. The value of 0 allows files with an unlimited number of records. If the limit is exceeded, the Overflow strategy specifies the action to take.

Max. audit file size
Allows limiting the file size in order to make reasonably sure the decoded file does not exceed 2 Gigabyte (otherwise it could not be handled).
Notes:

- This limitation adds to the limitation of **Max. records per file**: the first limit reached (size or number) will trigger the overflow action.

- As a rule of thumb, the ASCII output is about 5-10 times as big as the binary file.: It's impossible to calculate the output size easily from the input size because the requests can vary from small to very big.
- Do not use undersize limits (wrapping or moving the audit file is rather expensive compared to normal audit record writing).

Internal buffer size
Specifies whether audit information is buffered internally before it is written to the audit log file. A value of 0 specifies that all audit information is written to the audit log file when an operation is completed. A value greater than 0 specifies the buffer size in number of bytes. The audit information is written to the audit log file when the buffer size is exceeded.

Overflow strategy
Specifies the action to take when the number specified in Max. records per file is reached. Specify one of the following values:

- **Stop service**
  Shut down the LDAP server. An exception message is written. The LDAP server is not re-started automatically.
- **Stop audit**
  Stop the auditing process. An exception message is written.
- **Wrap around**
  Begin writing at the top of the log file, overwriting the old content. (Default value)
- **Move file**
  Close and rename the audit file currently in use. A new audit log file is created, and auditing is continued using the current configuration settings. You can use the **dirxauddecode** command to evaluate the renamed audit file. See the *DirX Directory Administration Reference* for details.
- **Multi wrap, Max number of wrap files**
  If the wrap around mode is configured, this setting allows specifying more than one wrap files

Start time
Specifies the time at which the LDAP server starts recording of audit information. A value of **none** specifies no start time and does not affect the audit.

Stop time
Specifies the time at which the LDAP server stops recording of audit information. A value of **none** specifies no stop time and does not affect the audit behavior.

Specify the times in the format *DD**.***MM***.***YYYY hh***:**mm**:**ss*, where

- *DD* specifies the day of month (01 up to 31)
- *MM* specifies the month (01 up to 12)
- *YYYY* specifies the year
- *hh* specifies the hour (00 up to 23)
- *mm* specifies the minutes (00 up to 59)

- *ss* specifies the seconds (00 up to 59)

Cron Job
The intention is to allow configuring audits for several servers at the same time interval. Note that **Start/Stop time** (see above) and **Cron job** cannot be enabled at the same time.

- **Start time**
  Defines an absolute time for the cron job to start for the *first* time. When the cron job runs for the first time it applies the action defined by the overflow strategy
- **Cron Job Interval**
  Defines the seconds to wait before the next cron job is run after the previous one.

Encryption type
Specifies whether or not the audit log file is encrypted. Specify one of the following values:

- **No encryptio**n
  The audit log file is not encrypted.
- **Scramble**
  The audit log file is encrypted using the scramble mechanism.

Encryption Key File
Specifies the full pathname of the file that contains the key material to encrypt the audit information to be saved in the audit log file. This key material may be a password that is used as a cipher key, a pass phrase, or a PIN that is used by Personal Security Environment to access the current encryption key. If the key material is incorrect or the specified file does not exist or cannot be read, the LDAP server terminates.*
Note: if uncertain, leave this field empty*

Session tracking
Specifies whether session tracking is enabled. The available fields are:

- **On**
  Enables session tracking.
- Off
  Disables session tracking.
- Max. info length ... characters
  Specifies the maximum number of characters that are written for session tracking.

2.3.4.2. Tab: Operations

Use the LDAP audit subentry's **Operations** tab to configure the level of detail audited and the operations that are audited.

The available fields are:

Attribute detail level
Specifies the level of detail of audit information to be logged for attribute values of the request parameters of add, modify, compare operations and of the LDAP v3 control parameters. You can use this field to limit the amount of audit information collected.

Specify one of the following values:

- **maximum**
  The object name, all attribute types, and the attribute values are logged: If the value length exceeds the length specified in the **Limit for attribute values** field, only the length of the value is logged. If the value contains unprintable characters, **dirxauddecode** delivers a hex dump format of the value. See the *DirX Directory Administration Reference* for further details about **dirxauddecode**.
- **medium**
  The object name, all attribute types, and the length of the attribute values are logged. (default value)
- **minimal (no attributes)**
  Only the object name is logged. No attribute information is logged.
- **Limit for attribute values**
  Specifies the maximum length of an attribute value that is logged. If the length of the attribute value exceeds this limit, only the value's length is logged. The value of 0 allows the logging of values with a length of an unlimited number of bytes. The default value is 64.

Include operations
Lists the operations that can be logged. Select one or more of:

- **All**
  All operations (default).
- **None**
  No operation. Note that you still can have **Audit errors** (see below) enabled.
- **LDAP**
  All LDAP server operations except other and unknown operations.
- A**bandon**
  All abandon operations.
- **Add**
  All add operations.
- **Bind**
  All bind operations.
- **Compare**
  All compare operations.
- **Extended**
  All LDAPv3 extended operations; for example, unsolicited notification or start_TLS.
- **Modify DN**
  All moddn operations.
- **Modify**
  All modify operations.
- **Remove**
  All remove operations.

- **Search**
  All search operations.
- **Unbind**
  All unbind operations.
- **Other**
  All other unexpected client operations.
- RPC
  All RPC operations.
- Unknown
  All unexpected client operations that indicate client misbehavior; for example, closing the socket layer without initiating an LDAP unbind. operation

Audit errors
Enables auditing of erroneous operations.

2.3.4.3. Tab: All Attributes

Use the **All Attributes** tab to display and manage the LDAP Audit Subentry attributes.

2.3.5. LDAP Configuration Subentry

The LDAP configuration subentry stores configuration attributes for a DirX Directory LDAP server. You can

- Add an LDAP configuration subentry (right-click any LDAP configuration subentry or the virtual node above it)
- Delete an LDAP configuration subentry (right-click the LDAP configuration subentry to be deleted)
- View and modify an LDAP configuration subentry (right-click or double-click the LDAP configuration subentry to be viewed/modified)

DirX Directory Manager provides the information contained in an LDAP configuration subentry in the following tabs:

- General
- Cache
- User Filtering
- IP Filtering
- Flow Control
- LDAP V2 Settings
- Service Controls
- ExtOp Users
- ExtOp Groups
- User Policies

- Group Policies
- All Attributes

Notes:

- Exercise care in administering LDAP subentries. Incorrect values may cause the server to be unavailable for some or even all users (including yourself!). If you lock yourself out, use the DirX Directory command **dirxadm** to correct the problem.
- Values in gray are the defaults used by the LDAP server if no specific value is supplied.

2.3.5.1. Tab: General

Use the LDAP configuration subentry's **General** tab to define general configuration parameters for a DirX Directory LDAP server.

Note that a missing property value means that the server default is in effect for that property. As server defaults can vary between different builds of the server, please refer to the server documentation for the defaults applying to the server you are currently contacting.

The available fields are:

Port number
Specifies the port number on which the LDAP server is to listen. The default port number is **389**.

Unbind delay time
Controls the LDAP server's re-use of authenticated DAP binds. Possible values are:

- **0 or <0**
  Directs the LDAP server to close an authenticated DAP bind right after the last shared client has closed the LDAP connection to the backend (DSA).
- **n>0**
  Directs the LDAP server to close an authenticated DAP bind *n* seconds after the last shared client has closed the LDAP connection to the back end (DSA).

Secure port number
Specifies the port number on which the LDAP server is to listen for Secure Socket Layer (SSL) secured LDAP protocol requests. The default port number is **636**. Note that secure port number **0** means that the LDAP server does not accept secure connections. You can read more about secure connections in the topics about SSL/TLS ("Using LDAP").

Connection idle time
Specifies the number of seconds of inactivity after which the LDAP server is to close an LDAP connection to an LDAP client.

Max. connection number
Specifies the maximum number of simultaneous connections an LDAP server maintains with its clients. The maximum is 4000. Do not confuse this number with the maximum number of connections the LDAP server maintains with the backend (DSA).

Thread pool size
Specifies the maximum number of operation threads available to operate LDAP client requests in parallel. This pool size is static and cannot be modified without restarting the server. If the value is set too high for your system there might be the effect of permanent re-scheduling of threads for the cost of lowered LDAP performance.

Anonym DAP pool size
Specifies the number of simultaneous DAP connections that the LDAP server establishes to the DSA for anonymous binds.
Additionally the LDAP server establishes one DAP connection for internal operations. This attribute is an optional, single-valued attribute.

Increasing the value of this attribute can improve the performance of anonymous binds. However, the maximum number of connections established for authenticated binds decreases. The maximum number of simultaneous connections that can be established is determined by system resources, by the maximum number of connections specified in the LDAP **Max Connection number** attribute, and by the behavior concerning LDAP **Backend Sharing**.

SSL conf. subentry name
Specifies the common name attribute of the LDAP SSL configuration subentry. This attribute is an optional single-valued attribute. When this attribute is omitted, the LDAP server reads the LDAP SSL configuration subentry with the default common name value LdapSSLConfiguration.

Audit conf. subentry name
Specifies the common name attribute of the LDAP Audit configuration subentry. This attribute is an optional single-valued attribute. When this attribute is omitted the LDAP server reads the LDAP audit configuration subentry with the default common name value ldapAudit

Backend sharing
Directs the LDAP server to share connections to the directory server among all users using identical credentials.

Max. DAP share count
Specifies how many LDAP users can share the same DAP connection when they authenticate with the same credentials if LDAP backend sharing is enabled.

Read-only server
Specifies whether the LDAP server accepts or refuses update operations.

Anonymous access
Specifies whether anonymous access is allowed or denied. This field is only available if present in the schema. When "deny" is selected, any kind of anonymous access is denied. Note that this behavior is not LDAP-compliant since the schema is always supposed to be readable anonymously.

Simple Auth access
Specifies whether simple authenticated access is allowed or denied. When simple authenticated access is denied:

- If a user tries to bind using simple authentication, the operation is rejected and an implicit anonymous bind is not established (differs from the RFC).
- If anonymous access is also denied (**Anonymous access** set to **Deny**), the server becomes unavailable for LDAP clients (only RPC calls are possible).

Listen IP address
Specifies the IP address on which the LDAP server is to listen for client connection requests. If a server machine happens to provide more than one IP address, it might be make sense to select just one IP address to allow clients to connect to the LDAP server rather than having the LDAP server listen to all IP addresses.
Notes:

- If an invalid IP address is specified, the LDAP server will not start.
- If 127.0.0.1 is specified, the server is only accessible from collocated local clients (might be useful for administrative tasks).
- This setting has no impact on RPC or the OSI/IDM connections from the LDAP server to the DSA. These ports are still bound to all IP addresses.

Start TLS
Specifies whether or not the LDAP extended operation startTLS is allowed. This operation allows a client to enable the SSL/TLS security layer for an LDAP connection that was created as a normal (plain) LDAPv3 connection.

Max. request attributes
Specifies the maximum number of requested attributes that a client is allowed to specify in an LDAP search request. A value of **0** specifies that an unlimited number of requested attributes is allowed.

Max. filter items
Specifies the maximum number of filter Items that a client is allowed to specify in the filter of an LDAP search request. A value of **0** specifies that an unlimited number of filter items is allowed.

2.3.5.2. Tab: Cache

Use the LDAP configuration subentry's Cache tab to enable and disable LDAP server result caching and configure result caching operations. You can use the fields in the Cache tab to adjust an LDAP server's result cache to your local memory requirements and update frequency. When the LDAP server's result cache is enabled, the LDAP server searches the cache for a query before it sends the query on to the DSA for processing. Caching results can significantly improve performance in scenarios where LDAP clients send frequent identical search requests, at a cost of extra memory. On the other hand, it can be counterproductive, if identical search results happen never or only occasionally, particularly if in addition a high modification rate enforces frequent removals of search results from cache.

If the cache is full, a new search result will cause existing search results to be removed from the cache based on their present hit rate. In order to avoid too much overhead, the LDAP server tries to clear a "decent" amount of space at once. At first, all search results with the

lowest present hit rate will be removed. If this action does not provide "enough" available space, all search results with the second lowest hit rate will be removed and so on. Note that **Min. time to cache** (see below) may prohibit or at least defer any removal of search results from the cache.
Another cause for removing a search result is when the directory database is updated in a way that affects that search result (see **Cache update strategy** below).

Note that a missing property value means that the server default is in effect for that property. As server defaults can vary between different builds of the server, please refer to the server documentation for the defaults applying to the server you are currently contacting.

The available fields are:

Cache
Enables or disables the LDAP result cache. The available fields are:

- **Enabled**
  Enables the LDAP result cache.
- **Disabled**
  Disables the LDAP result cache.

Max. cached results
Specifies the maximum number of LDAP search results that can be stored in the LDAP result cache. An LDAP search result consists of one or more LDAP result messages. Each result message describes one resulting entry. The last result message contains the result code of the search. If the search results in no entry, the LDAP search result consists only of the last result message containing the result code. The value of this component shall be chosen in accordance with the Size of cache table field and the available system memory. Each cache entry requires about 1KB.

Max. memory size
Specifies the maximum memory size in MB used for the LDAP result cache.

Min. time to cache
Specifies the minimum number of seconds that an LDAP search result is stored in the LDAP cache (unless it must be removed earlier because that search result is affected by an update of the directory database, see also below "**Cache update strategy"**).

Max. time to cache
Specifies the maximum number of seconds that an LDAP search result is stored in the LDAP cache. The LDAP server will automatically remove search results from cache as soon as they exceed **Max. time to cache** - regardless of which fill factor the cache has at that point of time.

Min. cache entries
Specifies the minimum number of entries that an LDAP search result must have in order to be stored in the LDAP cache.

Max. cache entries
Specifies the maximum number of entries that an LDAP search result can have to be

stored in the LDAP cache.

Min. cache attributes
Specifies the minimum number of attributes that an LDAP search result must have in order to be stored in the LDAP cache.

Max. cache attributes
Specifies the maximum number of attributes that an LDAP search result can have to be stored in the LDAP cache.

Min. cache values
Specifies the minimum number of attribute values that an LDAP search result must have in order to be stored in the LDAP cache.

Max. cache values
Specifies the maximum number of attribute values that an LDAP search result can have to be stored in the LDAP cache.

Size of cache table
Specifies the size of the internal cache table. For best performance, the value of this field should be at least (Max. cache values/ 3). It must be at least 128. Note that the LDAP server may tacitly cut down your setting to its hard-coded internal upper limit.

Cache update strategy on modifications of the directory database
Specifies how the LDAP cache is to be updated when the directory database is modified. The available fields are:

- **Clean whole cache**
  Any add, modify, delete or rename operation in the directory database causes the **entire cache** to be emptied.
- **Clean cache applying DN+scope strategy**
  The cache is **emptied partially** based on what can be inferred from a comparison of the base DNs and scopes of the cached searches with the DN of the entry currently being modified, added or deleted. If, for example, the entry cn=Smith, ou=sales, o=my-company is to be modified, there is no need to remove a search with base DN ou=development, o=my-company from the cache.
- **Clean cache applying attribute strategy**
  The cache is **emptied partially** based on what can be inferred from a comparison of the requested attribute types in the cached searches and the attribute type of the entry currently being modified. If the attribute that is to be modified is not among the requested ones of a search, there is no need to remove that search form the cache.
- **Clean cache applying DN+scope+attribute strategy**
  This option combines the "DN" and "attribute" strategy. This option is the default, since it is expected to be the best trade-off in typical usage scenarios.

2.3.5.3. Tab: User Filtering

Use the LDAP configuration subentry's User Filtering tab to control access to an LDAP server by clients communicating over specific user DNs and specific group DNs.

Be careful not to lock yourself and your users out with inappropriate settings, or you may be prevented from using DirX Directory Manager to undo your settings. If this situation occurs, you'll need to try to undo them from a different, still "allowed" and *not* "denied" user or use the DirX Directory command line program dirxcp via DAP.

The available fields are:

Allowed users
Check one of

- All users
- No users
- Users specified below and specify the distinguished names of the users and groups to which the LDAP server grants access.

Denied users
Check one of

- All users
- No users
- Users specified below and specify the distinguished names of the users and groups for which the LDAP server denies access.

See also: IP Filtering

2.3.5.4. Tab: IP Filtering

Use the LDAP configuration subentry's IP Filtering tab to control access to an LDAP server by clients communicating over specific IP addresses.

Be careful not to lock yourself and your users out with inappropriate settings, or you may be prevented from using DirX Directory Manager to undo your settings. If this situation occurs, you'll need to try to undo them from a different, still "allowed" and *not* "denied" user or use the DirX Directory command line program dirxcp via DAP.

The available fields are:

Allow IP addresses
Lists the IP addresses to which access is to be granted (unless the address matches the deny field, too; access is granted only if the IP address matches the Allow list and does not match the Deny list).

Deny IP addresses
Lists the IP addresses for which access is to be denied.

Use the value ***.*.*.*** or the keyword **all** to specify all possible IP addresses.

Use the value **0.0.0.0** or the keyword **none** to specify no IP address.

Use the wildcard (*****) character to specify subnets.

See also: User Filtering

2.3.5.5. Tab: Flow Control

Flow control basically intends to allow making sure the server is not occupied by load that can be avoided.

The available fields are:

TCP keep alive
Directs the LDAP server to prompt the TCP system to send 'alive' packets periodically to the peer in order to check whether the peer is still connected. Otherwise, the TCP system (and so the LDAP server) would not recognize a LAN cable being non-functional. Note that the default intervals for keep-alive packets are rather large (e.g., two hours) on some OSes; in such cases, a cable drop would not be detected until two hours have elapsed. So it might be reasonable to reduce these timers).

Async. sockets
Blocking ("sync") sockets can cause a thread created by the LDAP server to be lost. Note, however, that asynchronous (non-blocking) sockets can be slightly slower than blocking ones.

Max. send timeout for async sockets, Max. receive timeout for async sockets
If asynchronous sockets are used, these settings limit the amount of time the LDAP server concedes for a send or receive operation, if an error condition occurs.

Max. incomplete ops per connection
Though unlikely to occur, a client can conduct a denial of service attack by sending requests without collecting any result. This setting limits the number of operations per connection. If the client reaches the limit, the socket will be disabled for new incoming events until the number of outstanding operations goes below the limit.

Op stack size limit
This setting allows limiting the willingness of the server to accept incoming requests in an overload condition by limiting the number of requests that are queued. Instead, the client will be informed that the DSA is currently busy.

Number of overflow threads
If **Op stack size limit** is set to a value other than **0** (no limit), there must be at least one overflow thread that will handle overflow requests. Note that this number cannot exceed the thread pool size.

2.3.5.6. Tab: LDAP V2 Settings

Use the LDAP configuration subentry's LDAP V2 settings tab to specify how an LDAP server is to handle requests from LDAP v2 clients, if the LDAP server is configured to allow requests from LDAP v2 clients.

The available fields are:

Character Conversion Request

Specifies the character set encoding to use when converting printable-string attribute values supplied in LDAP v2 requests. Choose one of the following:

- **LATIN1**
  To select ISO 8859-1 format.
- **T61**
  To select T61 format.
- **UTF8**
  To select UTF-8 format.

Character conversion result
Specifies the character set encoding to use for search results generated by LDAP v2 operations. Choose one of:

- **UTF8**
  To select UTF8 as format.
- **LATIN1**
  To select ISO 8859-1 format.
- **NONE**
  To select no conversion.

X.509 attribute presentation
Specifies the representation to use when returning X.509 certificates to LDAP v2 clients. Choose one of:

- **Binary ASN1 encoding**
  To return X.509 certificates in binary ASN1 encoding format.
- **Hexdump of ASN1 encoding**
  To return X.509 certificates as a hexdump of ASN1 encoding.

2.3.5.7. Tab: Service Controls

Use the LDAP configuration subentry's Service Controls tabs to set LDAP operation service controls for a new LDAP configuration subentry and to change LDAP operation service controls for an existing subentry.

- Use the Add tab to define service controls for LDAP create operations
- Use the Compare tab to define service controls for LDAP compare operations
- Use the Modify tab to define service controls for LDAP modify operations
- Use the ModifyDN tab to define service controls for LDAP modifyDN operations
- Use the Remove tab to define service controls for LDAP delete operations
- Use the Search tab to define service controls for LDAP search operations

The available fields in each tab are:

Don't deference alias
Controls whether or not aliases found in the path of a query are dereferenced to the entries

to which they refer. Alias dereferencing is not supported for modify operations.

Partial copy OK
Controls whether or not the operation can be completed using a partial copy of the entry.

Don’t use copy
Controls whether or not an object’s shadow entry can be used to satisfy a request.

Subentries returned
Controls whether or not a list or search operation returns subentries and normal entries become inaccessible.

Priority
Sets the priority of a request. Choose one of the following:

- **Low**
  To set a low priority
- **Medium**
  To set a medium priority
- **High**
  To set a high priority

Time limit
Sets the maximum elapsed time in seconds for completion of an operation. If a list or search operation does not complete by the specified limit, the operation returns an arbitrary selection of results accumulated before it exceeded the time limit. Type the number of seconds.

Size limit
Sets the maximum number of objects returned in list and search operations. Type the number of objects.

Attribute size limit
Sets the maximum size (in octets) of any attribute (the type and all its values) to be returned. If an attribute exceeds this limit, all of its values are omitted.

Local scope
Controls whether or not an operation is limited to the objects in the DSA to which DirXmanage is currently bound, or whether other DSAs can be contacted.

Prefer chaining
Controls whether or not the preferred behavior of the LDAP server is chaining rather than referrals.

Chaining prohibited
Controls whether or not chaining and other methods of distributing a request around the Directory are prohibited.

Scope of referral
Sets the scope for the return of DSA referrals. Choose one of:

- **Within country only**
  Only referrals to DSAs within the country scope are returned.
- **Within management domain only**
  Only referrals to DSAs within the domain management scope are returned.
- **Unlimited**

2.3.5.8. Tab: ExtOp Users

Use this tab to specify the distinguished names of the administrators that are allowed to perform LDAP extended operations. For details, see the *DirX Directory Administration Reference* -> "DirX Directory Attributes" chapter -> "Attributes for LDAP Server Configuration" section -> "Attributes Controlling LDAP Extended Operations" subsection.

2.3.5.9. Tab: ExtOp Groups

Use this tab specify the distinguished names of the groups whose members are allowed to perform LDAP extended operations. For details, see the *DirX Directory Administration Reference* -> "DirX Directory Attributes" chapter -> "Attributes for LDAP Server Configuration" section -> "Attributes Controlling LDAP Extended Operations" subsection.

2.3.5.10. Tab: User Policies

Use the LDAP configuration subentry's User Policies tab to display, modify, add or delete user policies. You can use user policies to specify policies that apply to a single user or a set of users.

The User Policies Tab displays a list of all user policies. It provides the class, the user's distinguished name or the keywords **all** or **anonymous**, and the priority for each user policy.

Select a user policy and press the **Delete**-button to delete the policy.

Press the **Add**-button to add a new user policy or select a policy and press the button to display or modify a user policy. The LDAP User Policy Properties page opens.

The available fields are:

User Class
Specifies the user(s) for which the policies apply. Check one of the following keywords:

- **User Entry**
  Specifies that the policies apply to a single user. This policy cannot be overruled by policies of any other class.
- **All**
  Specifies that the policies apply to all users. If there are multiple policies, the user policies for **all** have the lowest priority. Anonymous users belong to all if there are no policies for **anonymous**.
- **Anonymous**
  Specifies that the policies apply to all users bound anonymously to the LDAP server; that is, to all users that performed an anonymous bind, to all LDAP clients that did not

perform a bind, or to users that performed a bind with an **invalid credentials** error code.

- **User Subtree**
  Specifies that the policies apply to all users below a node in the DIT. Policies of this class are not applied to users bound with SASL EXTERNAL authentication.
- **User Wildcard**
  Specifies that the policies apply to all users matching the regular expression syntax string according to Linux/Perl standards in the field **Choose DN**. A colon (**:**) is not allowed in the distinguished name; for example, ^cn=Digger.*.

Choose DN
Specify the user's full qualified distinguished name in LDAP format or press the **Choose DN** -button to select a node from the DIT. For details about distinguished names in LDAP format see the section "Distinguished Names" in the chapter "DirX Directory String Representation for LDAP Binds" in the *DirX Directory Adminsistration Reference*.

For binds with SASL EXTERNAL authentication, specify the distingushed name in DAP format preceded by the prefix **X500DN:**; for example, X500DN:/o=my-company/cn=admin. For details about distinguished names in DAP format, see the section "Distinguished Names" in the chapter "DirX Directory String Representation for DAP Binds" in the *DirX Directory Adminsitration Reference* for details. The LDAP server does not convert user names specified in DAP format to LDAP format or vice versa.

Priority
Specifies the priority of policies in the same class. Specify a positive integer. The highest priority is **0**. The default priority is **1**. The lower the priority value, the higher is the priority. Only one value can be specified.

If there are multiple policies for the same user, the policies with the highest priority (the lowest priority value) apply.

If there are multiple policies for the same user with the same priority, the first policies found apply. It is not predictable which policies are the first ones found.

Connection Limit
Specifies the maximum number of concurrent LDAP connections a user can open. Specify a positive integer; for example, 5. Only one value can be specified.

Size Limit
Specifies the maximum number of objects returned for a list or search operation. Specify a positive integer; for example, 100. Only one value can be specified.

This value overrides the value of the LDAP Search Service Controls attribute. However, it cannot increase the size limit configured for the DSA.

Time Limit
Specifies the maximum elapsed time in seconds for completion of an operation. Specify a positive integer; for example, 10. Only one value can be specified

This value overrides the value of the LDAP Search Service Controls attribute. However, it cannot increase the size limit configured for the DSA.

Maximum Operations
Specifies the maximum number of operations that a user is allowed to perform in the specified time interval. Enter a positive integer specifying the maximum number of operations in the first field. Enter a positive integer in the second field and select seconds, minutes, hours or days from the drop down menu to specify the time interval.

The operations abandon and unbind do not increase the number of operations.

Only one value can be specified.

Maximum Result Bytes
Specifies the maximum number of bytes that a user is allowed to receive from search operations in the specified time interval. Enter a positive integer specifying the maximum number of bytes in the first field. Enter a positive integer in the second field and select seconds, minutes, hours or days from the drop down menu to specify the time interval.

Only one value can be specified.

Minimum Search Base Level
Specifies the minimum level starting at the root object at which a base object must exist for a search request. Specify a positive integer; for example, 3. Only one value can be specified.

TLS required
Indicates whether or not a bind over SSL/TLS is required. Check **TLS required** to indicate that a bind over SSL/TLS is required. The default value is that any bind is allowed.

Disclose violation
Indicates whether or not detailed information about the specified limit is provided in the error message if the limit is exceeded. Check **Disclose violation** to indicate that detailed information is provided. The default value is that detailed information is provided. This flag affects violations of **Connection Limit**, **Maximum operations** and **Maximum result bytes**.

Must contact DSA
Specifies the DSA that must be contacted in a multiple contact DSA configuration. (See "Using Multiple Contact DSAs" in the *DirX Directory Administration Guide* for details.) Specify the name of the DSA as specified in the LDAP server configuration file; for example, /CN=DSA1. (See "LDAP Server Configuration File" in the *DirX Directory Administration Reference* for details.)

If the **Must contact DSA** is not available and no **Preferred contact DSA** is specified, the operation fails.

If the **Must contact DSA** is not available and a **Preferred contact DSA** is specified, the LDAP server continues with contacting the **Preferred contact DSA**.

If the **Preferred contact DSA** is also unavailable, the LDAP server starts a round-robin selection to contact a DSA.

Only one **Must contact DS**A can be specified.

Preferred contact DSA
Specifies the DSA that is preferred to be contacted in a multiple contact DSA configuration. (See "Using Multiple Contact DSAs" in the *DirX Directory Administration Guide* for details.) Specify the name of the DSA as specified in the LDAP server configuration file; for example, /CN=DSA3. (See "LDAP Server Configuration File" in the *DirX Directory Administration Reference* for details.)

If the **Preferred contact DSA** is unavailable, the LDAP server starts a round-robin selection to contact a DSA.

Only one **Preferred contact DSA** can be specified.

Forbidden Search Bases
Specifies a base object that is not allowed in search requests. Enter the full qualified distinguished name of a leaf or a node in the DIT in LDAP format; for example, ou=finance,o=my-company. For details about distinguished names in LDAP format, see the section "Distinguished Names" in the chapter "DirX Directory String Representation for LDAP Binds" in the *DirX Directory Administration Reference*. Multiple values can be specified.

Press the Add-button to enter a new value.

Select a value and press the Delete-button to remove the value.

Forbidden Targets
Specifies an entry that is not allowed in update operations. Enter the full qualified distinguished name of a leaf or a node in the DIT in LDAP format; for example, ou=finance,o=my-company. For details about distinguished names in LDAP format, see the section "Distinguished Names" in the chapter "DirX Directory String Representation for LDAP Binds" in the *DirX Directory Administration Reference*. Multiple values can be specified.

Press the Add-button to enter a new value.

Select a value and press the Delete-button to remove the value.

Inherit
Indicates whether or not policies are inherited. Press the **Inherit**-button to indicate that policies are inherited. The default value is that policies are not inherited. This flag can only be checked for the **all** class policies.

2.3.5.11. Tab: Group Policies

Use the LDAP configuration subentry's Group Policies tab to display, modify, add or delete group policies. You can use group policies to specify policies that apply to a group of users.

The Group Policies Tab displays a list of all group policies. It provides the group's distinguished name and the priority for each group policy.

If a user is member of several groups, the policies with the highest priority apply. (See **Priority** for details.) If the priority is also the same, the LDAP server randomly selects the

policies.

A **Group Policies** can only be specified for groups with a maximum of one million members. The LDAP server discards groups with more than one million members.

Select a group policy and press the **Delete**-button to delete the policy.

Press the **Add**-button to add a new group policy or select a policy and press the button to display or modify a group policy. The LDAP Group Policy Properties page opens.

The available fields are:

Choose DN
Specify the group's (an entry with object class **groupOfNames** (**GON**)) full qualified distinguished name in LDAP format or press the **Choose DN**-button to select a node from the DIT. For details about distinguished names in LDAP format see the section "Distinguished Names" in the chapter "DirX Directory String Representation for LDAP Binds" in the *DirX Directory Adminsistration Reference*.

Priority
Specifies the priority of policies in the same class. Specify a positive integer. The highest priority is **0**. The default priority is **1**. The lower the priority value, the higher is the priority. Only one value can be specified.

If there are multiple policies for the same user, the policies with the highest priority (the lowest priority value) apply.

If there are multiple policies for the same user with the same priority, the first policies found apply. It is not predictable which policies are the first ones found.

Connection Limit
Specifies the maximum number of concurrent LDAP connections a user can open. Specify a positive integer; for example, 5. Only one value can be specified.

Size Limit
Specifies the maximum number of objects returned for a list or search operation. Specify a positive integer; for example, 100. Only one value can be specified.

This value overrides the value of the LDAP Search Service Controls attribute. However, it cannot increase the size limit configured for the DSA.

Time Limit
Specifies the maximum elapsed time in seconds for completion of an operation. Specify a positive integer; for example, 10. Only one value can be specified

This value overrides the value of the LDAP Search Service Controls attribute. However, it cannot increase the size limit configured for the DSA.

Maximum Operations
Specifies the maximum number of operations that a user is allowed to perform in the specified time interval. Enter a positive integer specifying the maximum number of

operations in the first field. Enter a positive integer in the second field and select seconds, minutes, hours or days from the drop down menu to specify the time interval.

The operations abandon and unbind do not increase the number of operations.

Only one value can be specified.

Maximum Result Bytes
Specifies the maximum number of bytes that a user is allowed to receive from search operations in the specified time interval. Enter a positive integer specifying the maximum number of bytes in the first field. Enter a positive integer in the second field and select seconds, minutes, hours or days from the drop down menu to specify the time interval.

Only one value can be specified.

Minimum Search Base Level
Specifies the minimum level starting at the root object at which a base object must exist for a search request. Specify a positive integer; for example, 3. Only one value can be specified.

TLS required
Indicates whether or not a bind over SSL/TLS is required. Check **TLS required** to indicate that a bind over SSL/TLS is required. The default value is that any bind is allowed.

Disclose violation
Indicates whether or not detailed information about the specified limit is provided in the error message if the limit is exceeded. Check **Disclose violation** to indicate that detailed information is provided. The default value is that detailed information is provided. This flag affects violations of **Connection Limit**, **Maximum operations** and **Maximum result bytes**.

Must contact DSA
Specifies the DSA that must be contacted in a multiple contact DSA configuration. (See "Using Multiple Contact DSAs" in the *DirX Directory Administration Guide* for details.) Specify the name of the DSA as specified in the LDAP server configuration file; for example, /CN=DSA1. (See "LDAP Server Configuration File" in the *DirX Directory Administration Reference* for details.)

If the **Must contact DSA** is not available and no **Preferred contact DSA** is specified, the operation fails.

If the **Must contact DSA** is not available and a **Preferred contact DSA** is specified, the LDAP server continues with contacting the **Preferred contact DSA**.

If the **Preferred contact DSA** is also unavailable, the LDAP server starts a round-robin selection to contact a DSA.

Only one **Must contact DS**A can be specified.

Preferred contact DSA
Specifies the DSA that is preferred to be contacted in a multiple contact DSA configuration. (See "Using Multiple Contact DSAs" in the *DirX Directory Administration Guide* for details.)

Specify the name of the DSA as specified in the LDAP server configuration file; for example, /CN=DSA3. (See "LDAP Server Configuration File" in the *DirX Directory Administration Reference* for details.)

If the **Preferred contact DSA** is unavailable, the LDAP server starts a round-robin selection to contact a DSA.

Only one **Preferred contact DSA** can be specified.

Forbidden Search Bases
Specifies a base object that is not allowed in search requests. Enter the full qualified distinguished name of a leaf or a node in the DIT in LDAP format; for example, ou=finance,o=my-company. For details about distinguished names in LDAP format, see the section "Distinguished Names" in the chapter "DirX Directory String Representation for LDAP Binds" in the *DirX Directory Administration Reference*. Multiple values can be specified.

Press the Add-button to enter a new value.

Select a value and press the Delete-button to remove the value.

Forbidden Targets
Specifies an entry that is not allowed in update operations. Enter the full qualified distinguished name of a leaf or a node in the DIT in LDAP format; for example, ou=finance,o=my-company. For details about distinguished names in LDAP format, see the section "Distinguished Names" in the chapter "DirX Directory String Representation for LDAP Binds" in the *DirX Directory Administration Reference*. Multiple values can be specified.

Press the Add-button to enter a new value.

Select a value and press the Delete-button to remove the value.

2.3.5.12. Tab: All Attributes

Use the **All Attributes** tab to display and manage the LDAP Configuration Subentry attributes.

2.3.6. LDAP SSL Configuration Subentry

The LDAP SSL configuration subentry stores security configuration parameters for a DirX Directory LDAP server. Modifications to an LDAP SSL configuration subentry become effective after a restart of the LDAP server. The LDAP server SSL configuration subentry is only read at startup when SSL is enabled (when the value of the LDAP Secure port number attribute is greater than zero).

You can

- Add an SSL configuration subentry (right-click any LDAP SSL configuration subentry or the virtual node above)
- Delete an SSL configuration subentry (right-click the LDAP SSL configuration subentry

to be deleted)

- View and modify an SSL configuration subentry (right-click or double-click the LDAP SSL configuration subentry to be viewed/modified)

DirX Directory Manager provides the information contained in an LDAP SSL configuration subentry in the following tabs:

- General
- Cipher Suite Details for TLS lower 1.3
- Cipher Suite Details TLS 1.3
- Client Authentication
- All Attributes

Notes:

- Exercise care in administering LDAP subentries. Incorrect values may cause the server to be unavailable for some or even all users (including yourself!). If you lock yourself out, use the DirX Directory command **dirxadm** to correct the problem.
- Values in gray are the defaults used by the LDAP server if no specific value is supplied.

2.3.6.1. Tab: General

Use the LDAP SSL configuration subentry's **General** tab to define general SSL configuration parameters for a DirX Directory LDAP server.

The available fields are:

Security Protocols
Specifies the supported security protocols:

Minimum level: select the lowest protocol version accepted by the LDAP server from the drop-down list provided.

Maximum level: select the highest protocol version accepted by the LDAP server from the drop-down list provided.

For each of the protocol version levels one of the following values is possible:

- **TLSv1.0**
- **TLSv1.1**
- **TLSv1.2**
- **TLSv1.3**

The LDAP server accepts all security protocol versions between **Minimum level** and **Maximum level**.

Security Levels
The security levels refer to the OpenSSL security levels:

Security level A: minimum security level for own key-material - see OpenSSL function 'SSL_set_security_level' for details. If the level does not match the provided own server key-material, the server does not start.

Security level B: minimum security level for client key-material - see OpenSSL function 'SSL_set_security_level' for details. If the level does not match the client key-material, the SSL-handshake fails.

Personal Security Environment (PSE)
Specifies the LDAP server's PSE. This PSE comprises the following items:

Own key material
The LDAP server's own key material, which consists of the public key certificate and the private key.

Own key material password file
The password that is used to protect the private key in the PEM file. Specifies the full pathname to the file that contains this password is stored in the LDAP PKCS12 Password File attribute.

Trace path
Specifies the full pathname of the directory where the SSL-specific log files are located. Log file names use the format: **ipl*_PID_**.log*, where *PID* is the process ID of the LDAP server process that created the log file.
Samples:
(Windows): C:\Program Files\DirX\Directory\ldap\log
(Linux): /opt/dirx/ldap/ldap/log
Note: if uncertain, leave this field empty

Trace level
Specifies the level of SSL internal logging. 0 indicates that SSL logging is turned off.

Personal Security Environment (PSE) in files
These fields are obsolete and are supported only for compatibility reasons. It is not allowed to use these fields to store key material for DirX Directory V8.2B LDAP servers and newer. The **Personal Security Environment (PSE)** must be used instead.

2.3.6.2. Tab: Cipher Suite Details for TLS lower 1.3

Use the **Cipher Suite Details for TLS lower 1.3** tab to display and manage the cipher suite details either by specifying cipher suite shortcuts and / or by specifying explicitly the accepted cipher suites.

Cipher Suite Shortcuts
Specifies the cipher suites that are accepted in the SSL handshake protocol. Cipher suites are algorithms for signing, encryption, and hashing. Valid values for cipher suite shortcuts are:

- **All**
  The LDAP server accepts all cipher suites specified.
- **High**

The LDAP server accepts only cipher suites for high encryption strength.

- **Medium**
  The LDAP server accepts only cipher suites for medium encryption strength.
- **Low**
  The LDAP server accepts only cipher suites for low encryption strength.
- **RSA**
  The LDAP server accepts only cipher suites that use RSA algorithms.

Cipher Suites
Specifies individual cipher suites. Check or un-check individual cipher suites.

2.3.6.3. Tab: Cipher Suite Details for TLS 1.3

Use the **Cipher Suite Details for TLS 1.3** tab to display and manage the cipher suites that are supported in a TLS 1.3 handshake. TLS 1.3 supports a set of five cipher suites.

2.3.6.4. Tab: Client Authentication

Use the **Client Authentication** tab to display and manage the client authentication.

The available fields are:

Client Authentication
Specifies whether the LDAP server requires SSL client authentication for SSL-protected connections. When a client performs SSL client authentication, he/she requires a user certificate and a private key.

- **Not required**
  The client does not authenticate itself to the SSL layer.
- **Required**
  The client authenticates itself to the SSL layer.

SASL Authorization ID Mapping
Specifies how a client entity authenticated by an SASL-authenticated bind over SSL-protected LDAPv3 protocol (ldap sasl bind) with the mechanism EXTERNAL is mapped onto an Authorization ID.

Trusted CA certificates
The list of Certification Authority (CA) certificates the LDAP server trusts to issue user certificates, if SSL/TLS client authentication is required.

Personal Security Environment (PSE) in files
These fields are obsolete and are supported only for compatibility reasons. It is not allowed to use these fields to store key material for DirX Directory V8.2B LDAP servers and newer. The **Personal Security Environment (PSE)** must be used instead.

Certificate Revocation List Checking
Enables or disables certificate revocation list checking in the context of an LDAP SASL EXTERNAL bind. If enabled an LDAP SASL EXTERNAL bind is also checked against a

certificate revocation list (CRL).

Once CRL checking is enabled, at least one CRL file must be specified. If no proper CRL file is configured, the server does not start.

CRL file list
The name(s) of the files containing the PEM-formatted certificate revocation lists (CRLs) that are to be used to check the certificate for revocation in the context of an LDAP SASL bind with the mechanism type EXTERNAL. DirX Directory does not support delta CRLs.

Specify the filename values either as fully-qualified pathnames or as filenames without any path. If a filename without any path is specified, the LDAP server expects the file to reside in the directory *install_path*/ldap/conf*.

See "LDAP SSL CRL Filenames" in "DirX Directory Attributes" in the *DirX Directory Administration Reference* for details.

If CRL checking is enabled, the following fields control the check procedure:

Tolerate missing CRL
Controls whether or not a client certificate presented in the context of an LDAP SASL bind with the mechanism EXTERNAL is rejected if a suitable CRL cannot be found in the given CRL list.

Allow not yet valid CRL
Controls whether or not a client certificate presented in the context of an LDAP SASL bind with the mechanism EXTERNAL is rejected if the corresponding CRL is not yet valid.

Allow expired CRL
Controls whether or not a client certificate presented in the context of an LDAP SASL bind with the mechanism EXTERNAL is rejected if the corresponding CRL has expired.

The DirX Directory LDAP server uses CRLs stored locally in Privacy Enhanced Mail (PEM)-formatted files that are read during LDAP server process startup or when explicitly required. Press the **Refresh CRLs** button if it is necessary to load the updated CRLs to the LDAP server. Note that this action puts heavy load on the server.

2.3.6.5. Tab: All Attributes

Use the **All Attributes** tab to display and manage the LDAP SSL Configuration Subentry attributes.

2.3.7. LDAP Root Subentry

The LDAP root subentry stores information that describes the capabilities and supported features of an LDAP server. The properties of the Root node in the tree pane of the configuration view show the properties of the LDAP root subentry tailored for DirX Directory (the core component of this application displays the LDAP Root, too, but in a different, generic form that is supposed to work for non DirX Directory servers, too (see File->Server->Tab: LDAP Root).

DirX Directory Manager displays the information contained in the LDAP root subentry in two tabs: the General tab and the Supported tab.

Note that the values stored in LDAP subentries are not effective for LDAP servers that haven't read them. LDAP servers read LDAP subentries on restart; for additional options through **dirxadm**, please refer to the server manual.

2.3.7.1. Tab: General

Use this tab to specify the general properties of the LDAP server.

The available fields are:

- **Vendor name**
  The name of the vendor of the server software.
- **Vendor version**
  The version of the server software.
- **Naming contexts**
  The distinguished names of the naming contexts that a contact DSA masters or shadows.
- **Subschema subentry**
  The distinguished name of the LDAP subschema subentry.
- **Alternate server**
  The URLs of other LDAP servers that can handle client requests.

2.3.7.2. Tab: Supported

Use this tab to specify the features that the LDAP server supports.

The available fields are:

- **Supported LDAP versions**
  Specifies the LDAP protocol versions that the LDAP server described by this subentry supports. This is a mandatory field. Check the protocols that apply:
- Version 2
  To indicate that the LDAP server supports LDAP v2.
- Version 3
  To indicate that the LDAP server supports LDAP v3.
- **Supported profiles**
  Lists the names of Open Group profiles that an LDAP server can support.
- **Supported controls**
  Lists the controls that an LDAP server can support, in case of DirX Directory typically the ones that are depicted above and listed subsequently:
- Simple Paged Results (Request & Response Control)
  Used implicitly by this application in functions such as export, delete and in clipboard operations

- Server Side Sorting (Request Control)
  Not used by this application
- Server Side Sorting (Response Control)
  Not used by this application
- Password Policy (Request & Response Control)
  Used implicitly by this application in functions such as login, add entry, modify entry
- Subentries (Request Control)
  Used implicitly by this application when reading subentries

Note that the user of this application is not offered by this application to specify "Supported controls" himself.

- **Supported extensions**
  Lists the LDAP protocol extensions that an LDAP server can support. Typically empty.
- **Supported features**
  Lists the LDAP features that an LDAP server can support. Typically empty.
- **Supported SASL mechanism**
  The Supported SASL Mechanism attribute specifies the names of the SASL authentication mechanisms that the LDAP server supports. The LDAP server supports the EXTERNAL SASL (attribute value EXTERNAL) mechanism, which indicates that the LDAP server supports SSL/TLS certificate-based client authentication.

2.3.8. Password Policy Subentry

A password policy is a set of rules that controls how passwords are used and administered. Password policies are designed to improve the security of directories and make it difficult for password cracking programs to break into them. You use password policy subentries to establish password policies in a DirX Directory server. The global password policy subentry is located directly under "Root" and affects the user password attribute of all entries that are within its scope (which is the entire DSA). The global password policy subentry is the only subentry that specifies the password storage scheme.

The available settings are organized into two tabs:

- General
  This tab provides password syntax, hashing algorithm related settings and an exclusion list. For the global password policy, this tab provides the fields for defining the password storage scheme.
- Aging & Account Lockout
  This tab provides password aging and account lockout related settings.

Subtree-specific password policy subentries are located under the password policies subentries node under the context prefix. These subentries do not provide the password storage scheme related fields on their general tab. Use the subtree specification tab to configure which entries this password policy subentry affects.

Notes

- It is up to the client, how it deals with the password policy related information returned by the server. Some clients may e.g. display a message such as "invalid credentials", where the adequate message would rather be something like "password has expired". Others may display a combination of both messages. Clients like DirX Directory Manager that to a large extent conform to https://tools.ietf.org/html/draft-behera-ldap-password-policy-07 are supposed to react adequately.
- If you managed to lock yourself out (e.g. your password expired or you exceeded the maximum number of failures), the DirX Directory administration program **dirxadm** is supposed to help you out.
- Since the password policy entry is a *sub*entry, it has a subtree specification; if the password policy is intended to apply globally, its subtree specification is inoperative.
- The complete functionality presumes for an object that is supposed to do binds (i.e. that has the attribute "userPassword") that it has also the respective password policy related attributes such as

  pwdAccountLockedTime, pwdChangedTime, pwdExpirationWarned, pwdExpiryTime, pwdFailureTime, pwdGraceLoginsLeft, pwdGraceUseTime, pwdHistory, pwdReset.

2.3.8.1. Tab: General

For general remarks on password policy refer to the chapter Password Policy Subentry. Additional settings are described in Aging & Lockout Settings.

Use this tab to specify the general properties of the password policy.

The available fields are:

User Password Storage Scheme

- **No hashing**
  The server does not apply a secure hashing algorithm. The hashing support level is insignificant in this case.
  With this setting, the DSA is not able to handle user passwords that are available hashed or salted hashed, that is a bind operation with correct credentials will still fail if the password is stored in a hashed or salted hashed format in the DSA.
- **Hashed**
  The server applies the secure hashing algorithm according to the hashing support level and the hashing algorithm selected.
- **Salted Hashed** (recommended)
  The server applies the salted secure hashing algorithm according to the hashing support level and the hashing algorithm selected. A salt is a random number added to and stored with the hash value. The salted hashed format is even more secure than the hashed format, since it makes it drastically more expensive to detect passwords by generating pairs of clear text and corresponding hashed passwords and search these pairs as soon as a hash value becomes known.

These properties are provided for global password policies only.

Hashing support level

- **Basic (level 1)**
  The DSA can handle user passwords available in clear text as well as user passwords available in the hashed or salted hashed format, that is a bind operation with correct credentials will complete successfully, no matter how the relating password is stored in the DSA (clear text, hashed or salted hashed, regardless of the used hash algorithm). This setting presumes that hashed or salted hashed has been selected.
- **DSA converts for read (level 2)**
  Same as level 1, but the DSA also returns user passwords that are stored in clear text in hashed or salted hashed format (depending upon the SHA selected). A user password that is already stored in hashed or salted hashed format will be returned as is.
- **DSA converts for read and write (level 3)** (recommended)
  Same as level 2, but the DSA also automatically applies hashing or salted hashing (depending on the SHA selected) to any user passwords that are available in clear text when importing them. A user password that is already provided in hashed or salted hashed format will be imported as is.

These properties are provided for global password policies only.

Hashing algorithm

Check the variant of the secure hash algorithm to be applied. The default algorithm for hashing user passwords is SHA-1; furthermore the DSA supports the algorithms SHA-224, SHA-256, SHA-384 and SHA-512.

These properties are provided for global password policies only.

Check password syntax

- **Syntax check is disabled**
  The syntax of passwords is not checked for minimum and maximum password length and minimum number of lower case characters, upper case characters, digits and special characters (the settings in minimum password length, maximum password length, minimum number of lower case characters, upper case characters, digits and special characters are ignored).
- **Check if in cleartext, accept otherwise** (recommended)
  If present in cleartext, passwords are syntax checked.

  If not present in cleartext (this is typically the case if they are hashed or salted hashed), password are ***accepted*** regardless of their syntax (syntax check is impossible for passwords in hashed or salted hashed format)).

  Note that while the bind operation always expects the password in clear text, other operations like add entry or modify entry can take passwords in hashed and salted hashed format as well.
- **Check if in cleartext, reject otherwise**
  If present in cleartext, passwords are syntax checked.

  If not present in cleartext (this is typically the case if they are hashed or salted hashed), password are ***rejected*** regardless of their syntax (syntax check syntax check is

impossible for passwords in hashed or salted hashed format)).

Note that while the bind operation always expects the password in clear text, other operations like add entry or modify entry can take passwords in hashed and salted hashed format as well.

- **Minimum password length**
  Specifies the minimum number of characters that can make up a password. A value of 0 indicates that there is no minimum.
- **Maximum password length**
  Specifies the maximum number of characters that can make up a password. A value of 0 indicates that there is no maximum.
  If the password is propagated to other systems that have a password length limit, this setting can help users avoid assigning invalid passwords.
- **Minimum number of lower case characters**
  Specifies the minimum number of lower case characters (a-z) that a password must contain. A value of 0 indicates that there is no minimum.
- M**inimum number of upper case characters**
  Specifies the minimum number of upper case characters (A-Z) that a password must contain. A value of 0 indicates that there is no minimum.
- M**inimum number of digits**
  Specifies the minimum number of digits (0-9) that a password must contain. A value of 0 indicates that there is no minimum.
- **Minimum number of special characters**
  Specifies the minimum number of special characters (any character outside the range of A-z and 0-9) that a password must contain. A value of 0 indicates that special character use in passwords is unrestricted.
- **Exclude Name**
  Specifies that a password is rejected if it contains a substring of the user's name. A value of 0 indicates no restriction. Specify one of the following values:
- **0** - No restriction.
- **1** - Passwords are rejected, if the value is a case-ignored substring of the user's distinguished name (DN) in LDAP format; for example the user **cn=abele,ou=sales,o=my-company** cannot set his password to **Bele,Ou**.
- **2** - Passwords are rejected, if the user's relative distinguished name (RDN) is a case-ignored substring of the password; for example the user **cn=abele,ou=sales,o=my-company** cannot set his password to **Abele1**.
- **3** - Passwords are rejected, if either the condition for value 1 or the condition for value 2 applies.

Exclusions

Allows specifying a number of DNs the global password policy is not supposed to apply at.

2.3.8.2. Tab: Aging & Lockout

For general remarks on password policy refer to the chapter Password Policy Subentry. Additional settings are explained in General Settings.

Use this tab to specify the properties for aging and account lockout of the password policy.

The available fields are:

Aging

- **Minimum Age**
  Specifies the minimum elapsed time between two password modifications.
  The main intention of this setting is to prevent users from trying to circumvent a password **History** mechanism by quickly performing a series of password changes having in mind to kick their favorite password out of the history list.
  Makes sense without a password history mechanism, too, as it may keep users from quickly switching back to their favorite password after it has expired.
- **Maximum Age**
  Specifies the maximum elapsed time 'till a password expires if not changed in between. Can’t be smaller than **Minimum Age**.
- **Expire Warning**
  Specifies the period of time from when authenticating users - due to imminent password expiration - are getting warning messages such as "Warning: Your password will expire at...". If this attribute has no value, or if its value is 0, no warnings will be issued. If not 0, the value must be smaller than **Maximum Age** (see above).
  Sample: Let **Maximum Age** be 30 days and **Expire Warning** be 5 days. This means that a user authenticating after 25 days at the earliest is supposed to receive an **Expire Warning**. A noteworthy effect of the Expire Warning is that the user can let elapse more than **Maximum Age**, i.e. 30 days and is still able to authenticate. Once he has received the first **Expire Warning** however, he must change his password within **Expire Warning**, i.e. 5 days - unless **Logins after expiration** allows him a number of additional "grace" logins.
- **Logins after expiration**
  Specifies the number of times an expired password can be used to authenticate (also known as "grace logins"). If this attribute has no value or if the value is 0, authentication will fail.
  Note that the number of grace logins may be "nibbled" undiscernibly, since times of inactivity may cause implicit disconnects and re-logins.
- **History**
  Specifies the maximum number of used passwords stored in the password history list of each user. A password cannot be reused, as long as it is found in the password history list. If this attribute has no value, or if the value is 0, used passwords are not stored in the password history list and thus may be reused.

Account Lockout

There are two reasons that can cause a DN to get locked out (till expiration of the **Lockout duration**, see below):

- Explicitly: The administrator locks the DN out thru the according entry related context menu item
- Implicitly: The server locks the DN out according to the settings described in this section

You can find out the currently locked out DNs by searching for "pwdAccountLockedTime is present"

- **Enable Account Lockout**
  Implicit lockout:
  Indicates, when checked, that the password may not be used to authenticate after a specified number of consecutive failed bind attempts. The maximum number of consecutive failed bind attempts is specified in **Number max. Failure** (see below). If **Enable Account Lockout** is not checked, the password may be used to authenticate even if the number of failed bind attempts has been reached.

  Explicit lockout:
  Entries that have been locked out explicitly by an administrator are still able to login, as long as **Enable Account Lockout** has not been checked off.
- **Number max. Failure**
  Specifies the number of consecutive failed bind attempts after which the password may not be used to authenticate (provided **Enable Account Lockout** is checked).
- **Failure count interval**
  Holds the period of time after which the password failures are purged from the failure counter, even though no successful authentication occurred. If this attribute has no value, or if its value is 0, the failure counter is only reset by a successful authentication.
- **Lockout duration**
  Holds the period of time that the password cannot be used to authenticate due to too many failed bind attempts. If this attribute has no value or if the value is 0 the password cannot be used to authenticate until reset by an administrator.
  Increasing the lockout duration afterwards will usually apply to existing lockouts as well. However, for those, whose lockout duration has expired and that had logged in successfully in between, the lockout duration has no effect any more.
- **Must Change after first Login**
  Specifies when checked that users must change their passwords when they first bind to the directory after a password is set or reset by the administrator. If this attribute is not checked users are not required to change their password upon binding after the administrator sets or resets the password.

2.4. Replication View

For in-depth information on replication, refer to the server documentation.

Replication of directory information among various directory servers is a way to make the directory service fail-safe, since data replicated e.g. from server 1 to server 2 may still be accessible through one server even if the other one shuts down for whatever reason. Besides, replication can be used to optimize communication traffic.

DirX Directory provides two mechanisms for replicating directory information:

- Shadowing
- LDIF file synchronization

As opposed to the LDAP based functionality, DirX Directory Manager contacts the DSA directly when administering replication. For this reason, it may - though it is successfully connected to an LDAP server - be unable to contact the DSA. In this case, a dialog pops up that gives you the opportunity to specify the appropriate connection data:

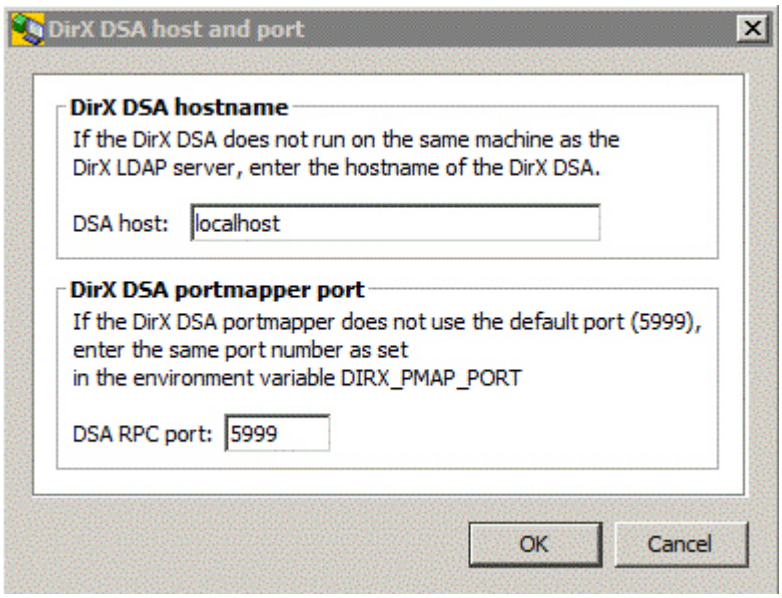


Figure 19. DirX DSA host and port dialog box

2.4.1. Shadowing

Shadowing automates the replication of directory information. What information is when to be shadowed among what servers must be configured by means of "shadowing agreements". For details about the functionality, see the topic "Shadowing Functions".

Shadowing should work between DirX Directory servers and non-DirX Directory servers provided they conform to the respective specifications put forth in the X.500 1993 Directory standard.

Pre-conditions

- The communication between two DSAs that are involved into a shadowing agreement is based on their "PSAP" addresses. Each DSA provides its PSAP address in a *system environment variable* called "DIRX_OWN_PSAP" on installation. This address contains the IP address as an essential part. Make sure that it is the one that is to be used by other DSAs that want to connect.
- The environment variable "DIRX_DSA_NAME" must contain a unique identification of the local DSA in LDAP DN syntax. Each DSA provides a presumably unique default DSA name on installation.

DSAs that are affected by a shadowing agreement

- In the narrower sense:

- A "Supplier"
  A supplier DSA masters the entire or a well-defined subset of the entire directory information; disjoint subsets can be mastered by different supplier DSAs. The supplier replicates changes of some or all of the data it masters to one ore - through several shadowing agreements - more consumers.
- A "Consumer"
  A DSA that consumes some directory information cannot at the same time master the same information for another DSA (no "secondary shadowing"). However, a consumer for one subset can be a supplier for another one.
- In the broader sense, in addition to supplier and consumer:
- The "Master of the cooperating DSAs table"
  There is one distinguished DSA that is the "master of the cooperating DSAs table". This table stores all shadowing agreements between all participating DSAs. It is created and maintained automatically by the DSA you are bound to when creating the first shadowing agreement.*
  Note that this is the DSA you must be bound to when administering shadowing agreements.*

  The concept of the master of the cooperating DSAs table disburdens you from the task of having to separately administer the suppliers and the consumers in a consistent way.

  The cooperating DSAs table can be viewed at as sort of a set of implied shadowing agreements. The context prefix is "cn=Cooperating-DSAs-Subentry" in any case. There are separate agreements between the DSA that masters the cooperating DSAs table and each consumer DSA.

  Though one single Supplier will possibly do and the Master of the cooperating DSAs table will possibly coincide with this Supplier in many deployments, you are not prevented from locating the mastership of the cooperating DSAs table at a separate DSA.
- All other participating DSAs
  The cooperating DSAs table implies individual shadowing agreements that are automatically shadowed to all participating DSAs immediately on change (unless this feature has been disabled).

Floating DSAs
Floating DSAs can take over the supplier role completely or on a per-shadowing area basis (context prefix + subtree base) either in a controlled way without loss of any data or through an "emergency switch" with the risk of losing the latest changes. A floating DSA becomes the master of the cooperating DSAs table when taking over the supplier role and

- Making an emergency switch
- Globally switching all context prefixes mastered by the DSA that also masters the cooperating DSAs table to another DSA

Functionality
The administration of shadowing agreements is (along with the administration of LDIF agreements) offered in an separate view called Replication.

The **Replication** view is organized into these panes:

- A tree pane
- A graph pane (applies to shadowing agreements only, not to LDIF agreements)
- A property pane

A number of functions are available by clicking the right mouse button in a node in the tree pane or in an element in the graph pane.

2.4.1.1. Shadowing Functions

For shadowing agreements, tree pane and graph pane offer the following functions through the right mouse button:

- **Create a new shadowing agreement**

  Presents the property dialog in an editable form.*

  Note* that once you have confirmed your data by clicking the OK button, you can no longer modify it. What you can do is to delete the agreement and then create a new one.

- *Establish an existing shadowing agreement
  Terminate an existing shadowing agreement
  Enable an existing shadowing agreement
  Disable an existing shadowing agreement*_

  Establish_ makes a shadowing agreement effective as long as it is not *disable*d. It is also called "Activated" or "Cooperative" at this point._
  Terminate_ makes it ineffective (also called "Not activated" or "Non cooperative"). A *terminate*d shadowing agreement stops collecting information that is needed to perform the shadowing. Re-*establish*ing a terminated agreement requires a total update.
  A *disable*d shadowing agreement continues collecting that information and executes it as soon as it gets re-*enable*d. Note that a DSA may also *disable* an agreement itself due to certain error conditions.

  Notes

- You cannot *terminate* a cooperating DSAs agreement.
- *Enabling* or *disabling* a shadowing agreement affects one single DSA.
  That is, if you are bound to a supplier, disabling a respective agreement will keep the supplier from trying to send update information to its consumer. If you are bound to a consumer, disabling a respective agreement will keep the consumer from accepting update information, while the supplier will still be sending updates
- *Enabling*/*Disabling* DSAs for agreements are the exceptions to the rule that you must be bound to the master of the coordinating DSAs table when administering shadowing agreements. Instead, you must be bound to the DSA you want to *enable*/*disable* for a shadowing agreement. This DSA can be the master of the coordinating DSAs table; for example, if you want to *enable*/*disable* a shadowing agreement where it is the master

of the data to be shadowed, or if you want to prevent the coordinating DSAs table from being shadowed to a certain consumer for some reason.

- **Switch supplier DSA**
  Changes a current supplier to a consumer and a current consumer to a supplier for the respective shadowing agreement(s). A consumer that can become a supplier is also called "Floating DSA". The respective box in the "General" tab of the shadowing agreement property dialog must be checked in order for a DSA to be considered able to float. When switching DSAs, you must be bound to the DSA that masters the cooperating DSAs table. What exactly will be switched depends on your current selection:
- If a supplier DSA is selected: All respective shadowing agreements.
- If a consumer DSA is selected: The respective shadowing agreements.
- If an agreement is selected: that agreement
- If the node "Cooperating DSAs subentry agreements" is selected: Switches the master of the cooperating DSAs table.

If the affected supplier is the master of the cooperating DSA table and if all respective shadowing agreements are to be switched, the floating DSA will also become the new master of the cooperating DSAs table.

- **Delete an existing shadowing agreement**.
  Affected implied shadowing agreements are implicitly removed from the cooperating DSAs table.
- **Show the properties of an existing shadowing agreement** (read-only, see also the very top of this topic).

2.4.1.2. Shadowing Tree Pane

The tree pane basically displays the information stored in the "cooperating DSAs table" in a tree-like manner where the suppliers form the first level below the node presenting the "shadowing agreements" and the consumers are arranged below their suppliers.

All the functionality is available through the right mouse button applied at those nodes.

The tree pane also gives you access to the LDIF File Sync. Functions.

2.4.1.3. Shadowing Graph Pane

Alternatively to the Shadowing Tree Pane, the Shadowing Graph pane presents the shadowing relations among the participating DSAs.

The Master of the cooperating DSAs table is colored **orange**, while the others are colored **blue**.

2.4.1.4. Shadowing Property Pane

Due to the similarity between shadowing agreement properties and LDIF agreement properties, they are treated together here for the most part.

The properties shown depend on the type of shadowing- or LDIF-related node currently selected in the Replication Tree Pane.

- Root node
- Supplier node
- Shadowing or LDIF agreement node
- General
- Update Mode
- Attribute Selection
- Shadowing Policies or LDIF Policies

2.4.1.4.1. Shadowing/LDIF Root Properties

Shows all shadowing or LDIF agreements in a multi-column list. Double-clicking an agreement leads to the property dialog for agreements.

2.4.1.4.2. Shadowing/LDIF Supplier Properties

Shows:

- Supplier DSA name.
- Supplier PSAP address.
- All belonging shadowing agreements in a configurable multi-column list. Double-clicking an agreement leads to the property dialog for agreements.

2.4.1.4.3. Shadowing Agreement Properties

The properties of shadow or LDIF agreements are shown when you:

- Click an involved consumer node in the tree pane (shadowing agreements only).
- Right-click on a consumer node in the tree pane and then select "Properties" (shadowing agreements only).
- Double-click a shadow or LDIF agreement in the list of shadowing/LDIF agreements that is displayed in the Shadowing/LDIF Supplier Properties pane.
- Double-click a shadow or LDIF agreement in the list of shadowing/LDIF agreements that is displayed in the Shadowing/LDIF Root Properties pane.

The agreement properties are organized in a dialog or pane with the following tabs:

- General
- Update Mode
- Attribute Selection
- Shadowing Policies or LDIF Policies

Tab: General

This tab allows you to specify the basic settings:

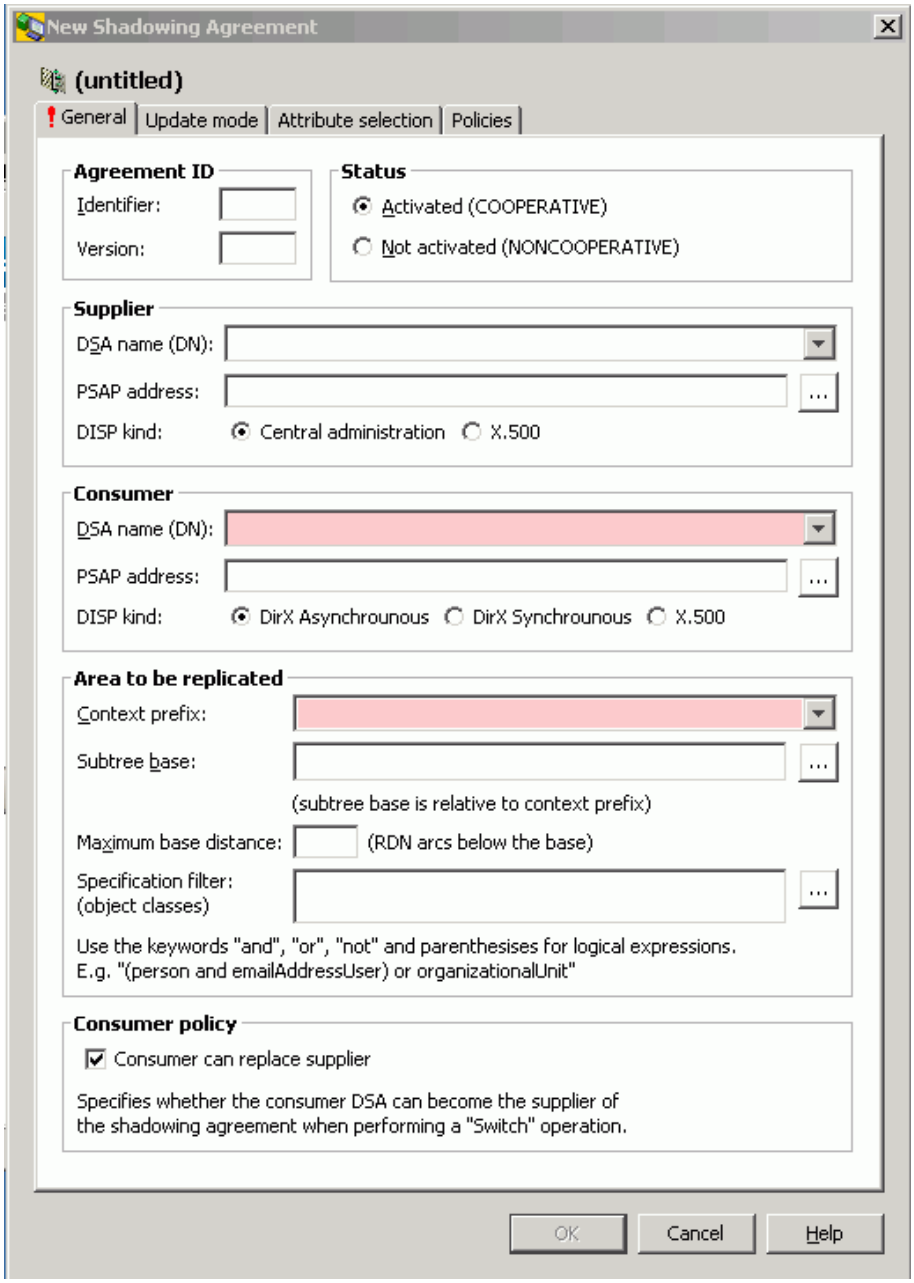


Figure 20. New Shadowing Agreement General Tab

Agreement Id

Each shadowing agreement must be uniquely identified by an agreement id. You can leave it to the DSA to allocate an identifier or you can allocate one yourself. The version is always allocated by the server. It starts with 0 and is incremented by 1 each time the agreement is re-established.

Status

The Establish function sets the status to "Activated (COOPERATIVE)", the Terminate function sets it to "Not activated (NONCOOPERATIVE). The Update status (shown in the tooltip) is affected by Enable or Disable.

Supplier or Consumer (Consumer: not shown for LDIF replication)
The supplier or the consumer is identified by its DSA name and its PSAP address. You can leave the supplier fields empty, if you do, the values are implicitly acquired from the DSA to which you are bound.

DSA name

The DSA name must coincide with the one that is found in the environment variable DIRX_DSA_NAME, as illustrated for Windows in the following screen shot (by default, the DSA name is supposed to be *cn=DIRX-DSA-**hostname*):

Figure 21. Edit System Variable Dialog Box

Note: If you leave the DSA field empty and proceed with providing the PSAP address in the subsequent field, the DSA field will take on the value *DIRX-DSA-*hostname_as_specified_PSAP_address*

PSAP address

The PSAP address must coincide with the one that is found in the environment variable DIRX_OWN_PSAP. This variable is implicitly set when installing DirX Directory. You can leave it unchanged, as long as the IP address and the port number are correct and you agree to communicate through the IDM stack. 21200 is the default port.

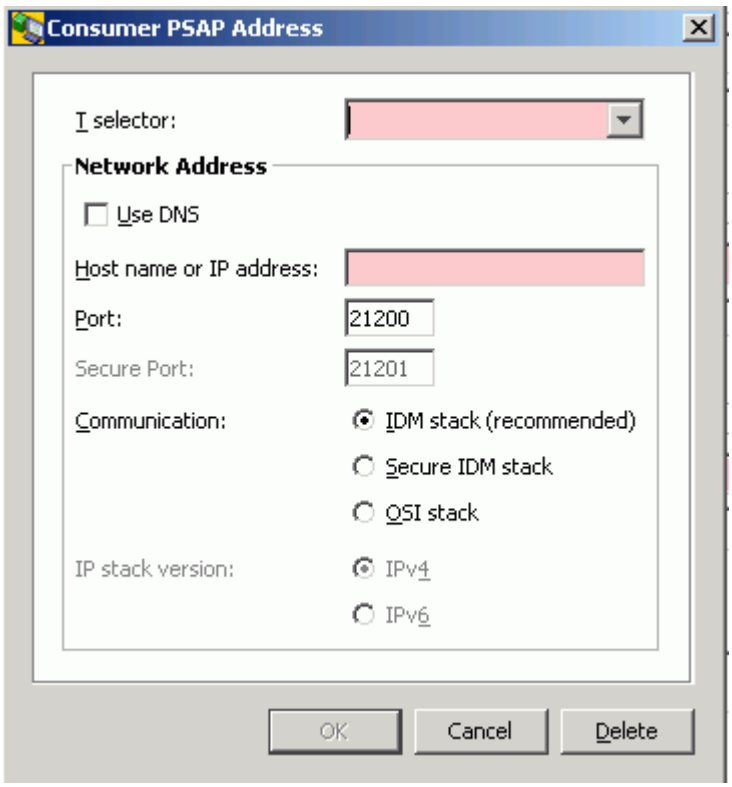


Figure 22. Consumer PSAP Address Dialog Box

T selector
By default, the T-selector specified in the environment variable DIRX_OWN_PSAP is "DSA1". You can usually leave the default as it is. All participating DSAs can use it at the same time. Be sure, however, to choose the one that is used in the corresponding environment variable DIRX_OWN_PSAP. You can take the default value from the combo box adjacent to the T selector field.

Use DNS
Check this box if you want to use the name service for resolving the IP address of the communication partner. This option is only supported for the IDM and secure IDM stack.

Host name or IP address
When typing a host name rather than an IP address, you might be asked in a subdialog to choose one of two or more IP addresses that were found to be assigned to that host. You must choose the one that is used in the corresponding environment variable DIRX_OWN_PSAP. If the preceding field (DSA name) is empty while you type in the host name here, it will be implicitly set to *DIRX-DSA-**hostname_as_specified_here*.

Port
Choose the one that is used in the corresponding environment variable DIRX_OWN_PSAP. If DIRX_OWN_PSAP says "localhost" or the like, change it in the environment variable DIRX_OWN_PSAP to the real IP address and specify the same IP address here.

Secure Port
Specify the secure port number here if you checked **Secure IDM stack**. Choose the one that is used in the corresponding environment variable DIRX_OWN_PSAP. If DIRX_OWN_PSAP says "localhost" or the like, change it in the environment variable DIRX_OWN_PSAP to the real IP address and specify the same IP address here.

Communication
The DirX Directory DSA supports the X.500 DAP, DSP, and DISP directory protocols directly over TCP/IP through the Internet Directly Mapped (IDM) protocol, as specified in ISO/IEC 9594-5: 2001 (E). The communication can be performed either directly over plain IDM or over a secure connection over TLS/SSL (secure IDM stack). For interworking scenarios the DirX Directory DSA also supports Open Systems Interconnection (OSI) communications.

IP stack version
Check the IP stack version for the IDM communication. DirX Directory supports IPv4 and IPv6.

Disp. kind (not shown for LDIF replication)
Check "Central Administration" or "DirX Asynchronous", if the participating DSAs are DirX Directory DSAs. In this case, you can benefit from the convenience of being able to centrally administer the shadowing. LDAP/DAP client update requests are returned immediately after commitment by the master DSA.
Check "DirX Synchronous", if the participating DSAs are DirX Directory DSAs. In this case, you can benefit from the convenience of being able to centrally administer the shadowing. LDAP/DAP client update requests are returned only after commitment by the master DSA and by all synchronous-configured consumer DSAs in the network.
Check "X.500" for non-DirX Directory DSAs. In this case, you must administer the shadowing agreements locally.

Area to be replicated
Specifies the area that is to be replicated in terms of context prefix and subtree base:

Context Prefix: Select the context prefix or - if there is more than one - select one of the available context prefixes.

Subtree base: Specify a subtree base relative to the context prefix.

Maximum base distance: Specify the maximum distance from the subtree base expressed in number of RDNs that are replicated.

Specification filter: Specify a filter that must match for the replicated entries.

Consumer Policy (not shown for LDIF replication)
If not checked, the consumer cannot become a supplier through the "Switch supplier DSA" function.
If checked, the field **Subtree base** must be left empty.

Tab: Update mode

This tab allows you to specify details regarding the update mode:

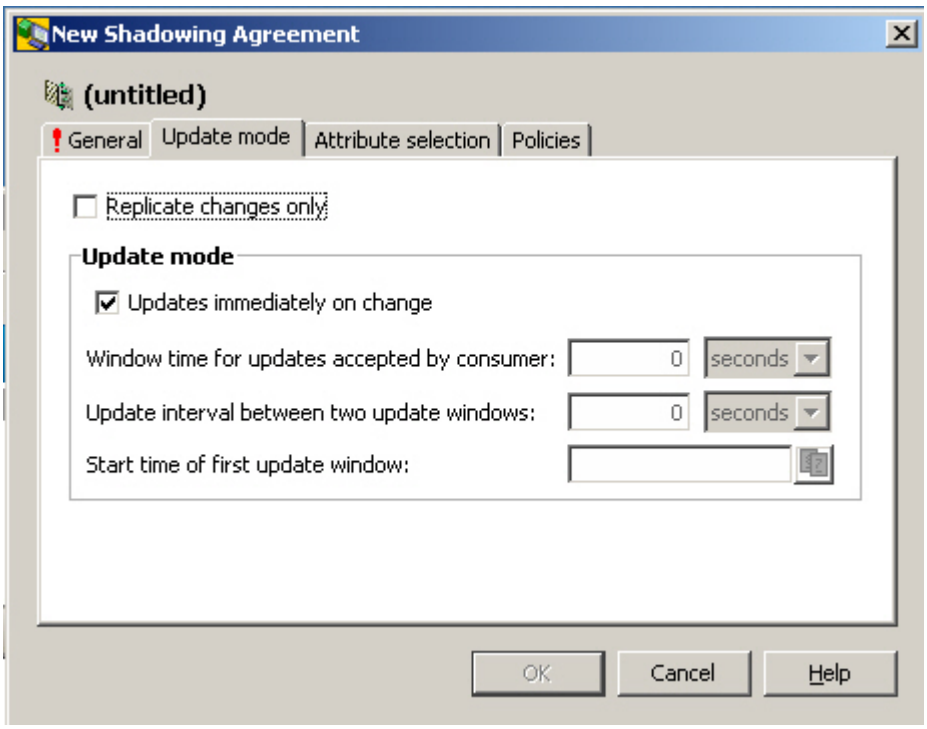


Figure 23. New Shadowing Agreement Update mode Tab

Replicate changes only (if shown, this check box is inoperative for LDIF agreements). Note that Tab "LDIF Policies" provides an appropriate check box called "Save changes only").
A boolean value that specifies whether only updates are replicated (checked) or the total content is replicated (unchecked). By default, the latter is only done initially; to allow total updates on errors, you must check the box "Automatic total update" in Tab "Shadowing policies".

Update immediately on change
"Checked" means that replicated entries are updated immediately when the master entries are modified or created. "Unchecked" means that no updates will be performed automatically. Default is "checked".

Window time for updates accepted by consumer
Only of interest if "Update immediately on change" is unchecked. Defines the time period during which the shadow consumer will accept updates from the shadow supplier. A shadow window allows updates to occur at anytime within the window time. It is useful if for some reason the updates are unable to start at the exact scheduled moment or for a retry if some update failure occurs.

Update interval between two update windows
Only of interest if "Update immediately on change" is unchecked. Specifies the interval between the opening of the update windows. For example, if "Update interval between two update windows" is set to 30 minutes and "Window time for updates accepted by consumer" is set to 5 minutes, the update cycle is 35 minutes. Updates of up to five minutes in duration can occur every 30 minutes.

Start time of first update window
Only of interest if "Update immediately on change" is unchecked. The start time of the first update window in Generalized Time format. When this start time is after the validity of the

shadowing agreement, the update cycle begins at this time. Once started, the update cycle begins until the shadow agreement is terminated. The default is the time the shadowing agreement was activated.

Tab: Attribute selection

This tab allows you to specify the attributes you do not want to be shadowed.

Here is an example of the Attribute selection tab:

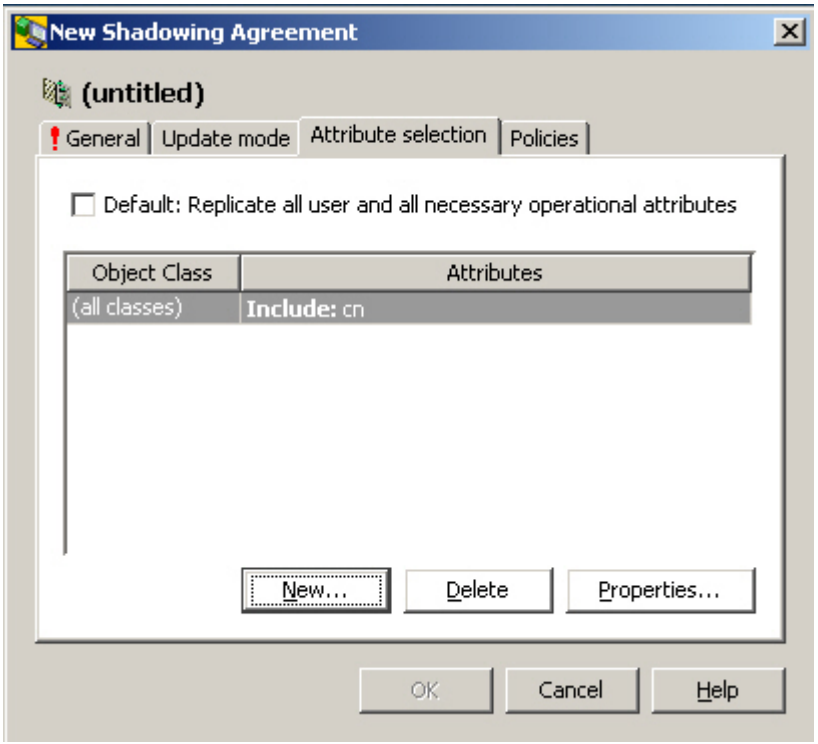


Figure 24. New Shadowing Agreement Attribute selection tab dialog box

It is possible to specify conflicting settings. In this case, the following rules apply:

- Explicit **Include** takes priority over explicit **Exclude**
- Explicit **Exclude** takes priority over **implicit** include

Tab: Policies

For policies of LDIF agreements, see LDIF policies.

This tab allows you to control the behavior in case of errors:

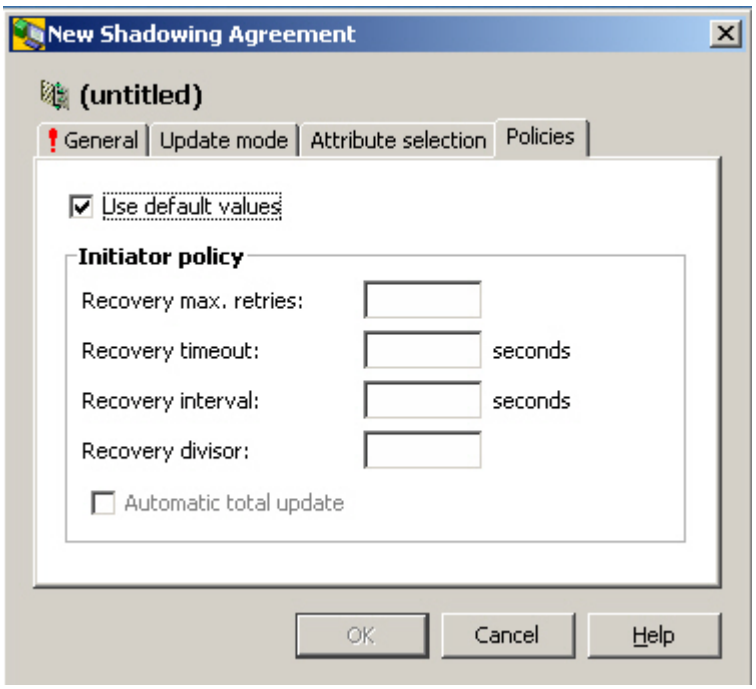


Figure 25. New Shadowing Agreement Policies Tab

Use default values
Specifies if checked that default policies should be used. See the other components for a description of the default policies.

Initiator policy

Recovery max. retries
The maximum number of retries of recovery for shadowing agreements where the local DSA is the initiator. The default is 20. If **recovery timeout** and **recovery max. retries** are equal to 0, recovery is configured as unlimited (that is, there is no limitation by timeout or by number of retries, and recovery never stops). If either **recovery timeout** or **recovery max. retries** is equal to 0 (and the other one is equal to a non-zero value), no recovery will be done. If **recovery timeout** and **recovery max. retries** are equal to a non-zero value, recovery will be attempted until both values have been exceeded.

Recovery timeout
The timeout in seconds of a recovery attempt for shadowing agreements where the local DSA is the initiator. The default is 1200 (that is, 20 minutes). If **recovery timeout** and **recovery max. retries** are equal to 0, recovery is configured as unlimited (that is there is no limitation by timeout or by number of retries, and recovery never stops). If either **recovery timeout** or **recovery max. retries** is equal to 0 (and the other one is equal to a non-zero value) no recovery will be done. If **recovery timeout** and **recovery max. retries** components are equal to a non-zero value, recovery will be attempted until both values have been exceeded.

Recovery interval
The time interval in seconds between the end of a recovery attempt and the start of the next attempt for shadowing agreements. Relevant only for shadowing agreements where the local DSA is the initiator and the update mode is checked ("on change"). The default is 240 (that is, 4 minutes).

Recovery divisor
The time interval between two recovery attempts for shadowing agreements is the window time divided by the **recovery divisor**. This component is relevant only for the shadowing agreements where the local DSA is the initiator and update mode is not "Immediately on change". The default is 16.

Automatic total update
Specifies whether the initiator DSA performs a total update automatically if it is required by the cooperating DSA (box is checked) or the administrator must trigger off the total update by performing an establish operation (box is unchecked). The default is FALSE (box is unchecked).

Tab: Status

This tab allows you to keep track of the progress the consumer is making when executing a shadowing agreement.

The following screenshot shows a typical example (for details, please refer to the server documentation):

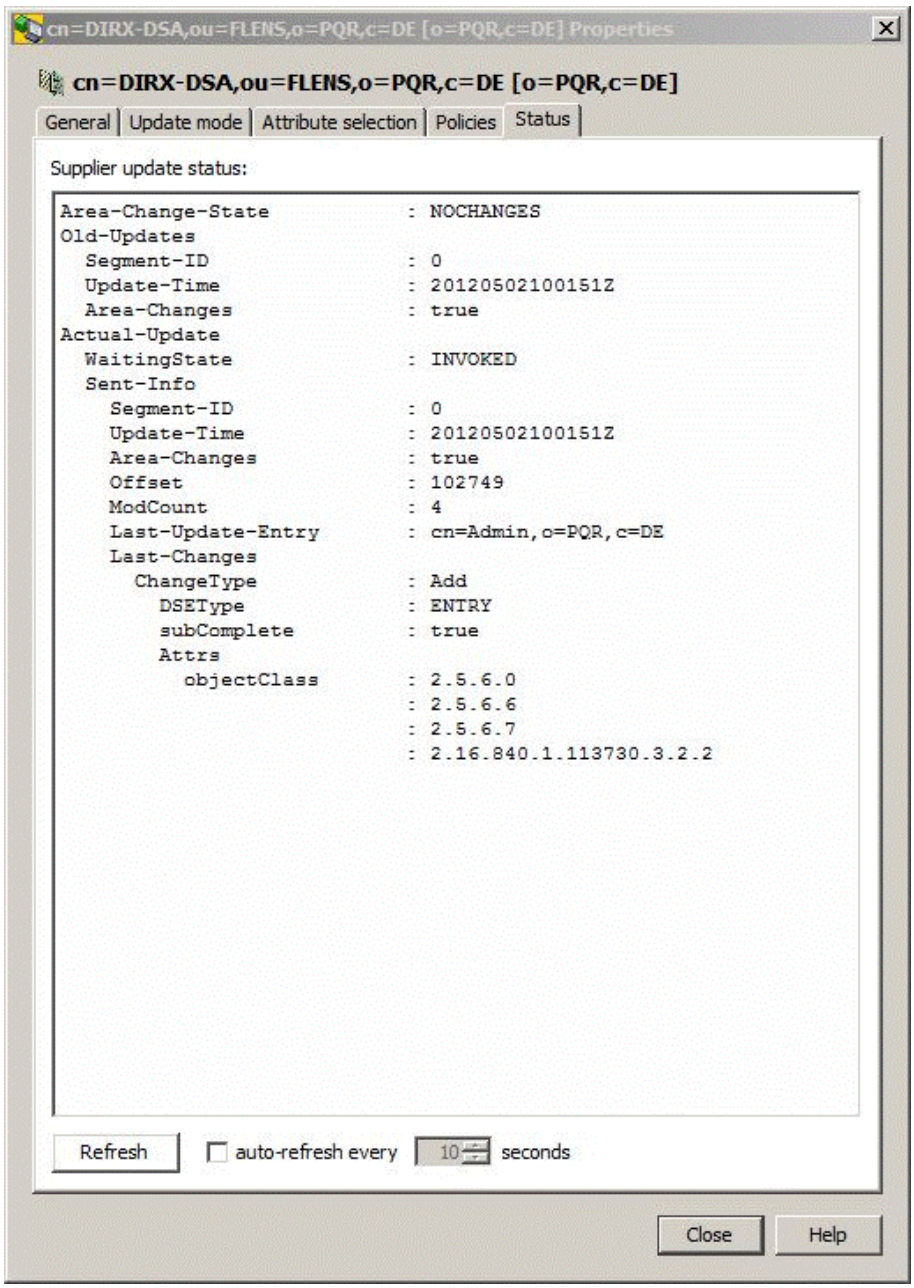


Figure 26. Status Tab of the DSA

2.4.2. LDIF File Synchronization

While shadowing covers both the export of data from one server and import of this data into another server, LDIF file synchronization only deals with the first step: it exports data into LDIF files (LDIF = LDAP Data Interchangeable Format, specified in RFC 2849) and allows associating triggers to initiate further processing.

The server should write the resulting LDIF files to *install_path*\Server\Ldif*.

LDIF files can be used to replicate changes to a DirX Directory DSA's directory database with DirX Identity to non-DirX Directories such as Windows Active Directory, NDS and Lotus Notes; this process is called directory synchronization and is described in detail in the DirX Identity documentation set. DirX Directory administrators can also use LDIF file generation as part of an overall backup strategy.

What information is when to be exported must be configured by means of so-called LDIF-agreements. For details about the functionality, see "LDIF File Sync. Functions".

Note that you must be bound to the master of the cooperating DSAs table when administering LDIF agreements.

LDIF file synchronization is should work between DirX Directory servers and with any LDAP-conformant non-DirX Directory servers.

2.4.2.1. LDIF File Synchronization Functions

For LDIF agreements, the tree pane and graph pane offer the following functions through the right mouse button:

- **Create a new LDIF agreement**

  Presents the property dialog in an editable form.*

  Note* that once you have confirmed your data by clicking the OK button, you can no longer modify it. What you can do is to delete the agreement and then create a new one.

- *Establish an existing LDIF agreement
  Terminate an existing LDIF agreement
  Enable an existing LDIF agreement
  Disable an existing LDIF agreement*_

  Establish_ makes an LDIF agreement effective as long as it is not *disable*d. It is also called "Activated" or "Cooperative" then._
  Terminate_ makes it ineffective (also called "Not activated" or "Non cooperative"). A *terminate*d LDIF agreement stops collecting information
  A *disable*d LDIF agreement continues collecting that information and executes it as soon as it gets re-*enable*d. Note that a DSA may also *disable* an agreement itself due to certain error conditions.

  Note
  Enabling/*Disabling* DSAs for agreements are the exceptions to the rule that you must be bound to the master of the coordinating DSAs table when administering agreements. Instead, you must be bound to the DSA you want to *enable*/*disable* for an LDIF agreement.

- **Delete an existing LDIF agreement**.
- **Show the properties of an existing LDIF agreement** (read-only, see also the very top of this topic).

2.4.2.2. LDIF File Synchronization Tree Pane

The tree pane basically displays the information stored in the "cooperating DSAs table" in a tree-like manner where the suppliers form the first level below the node presenting the "shadowing agreements" and the consumers are arranged below their suppliers.

All the functionality is available through the right mouse button applied at those nodes.

The tree pane gives you also access to the Shadowing Functions.

2.4.2.3. LDIF File Synchronization Property Pane

See Shadowing Property Pane.

2.4.2.3.1. LDIF File Synchronization Root Properties

See Shadowing Root Properties.

2.4.2.3.2. LDIF File Synchronization Supplier Properties

See Shadowing Supplier Properties.

2.4.2.3.3. LDIF Agreement Properties

See Shadowing Agreement Properties.

Tab: General

This tab shows the same information as Tab: General in Shadowing Agreement Properties but without Consumer and Consumer policy.

Tab: Update Mode

This tab shows the same information as Tab: Update mode in Shadowing Agreement Properties but without the checkbox "Replicate changes only" (if present, the field is inoperative for LDIF agreements). Note that Tab "LDIF Policies" provides an appropriate checkbox called "Save changes only".

Tab: Attribute selection

This tab shows the same information as Tab: Attribute selection in Shadowing Agreement Properties.

Tab: LDIF Policies

For policies of shadowing agreements, see Shadowing policies.

This tab allows you to specify some additional constraints:

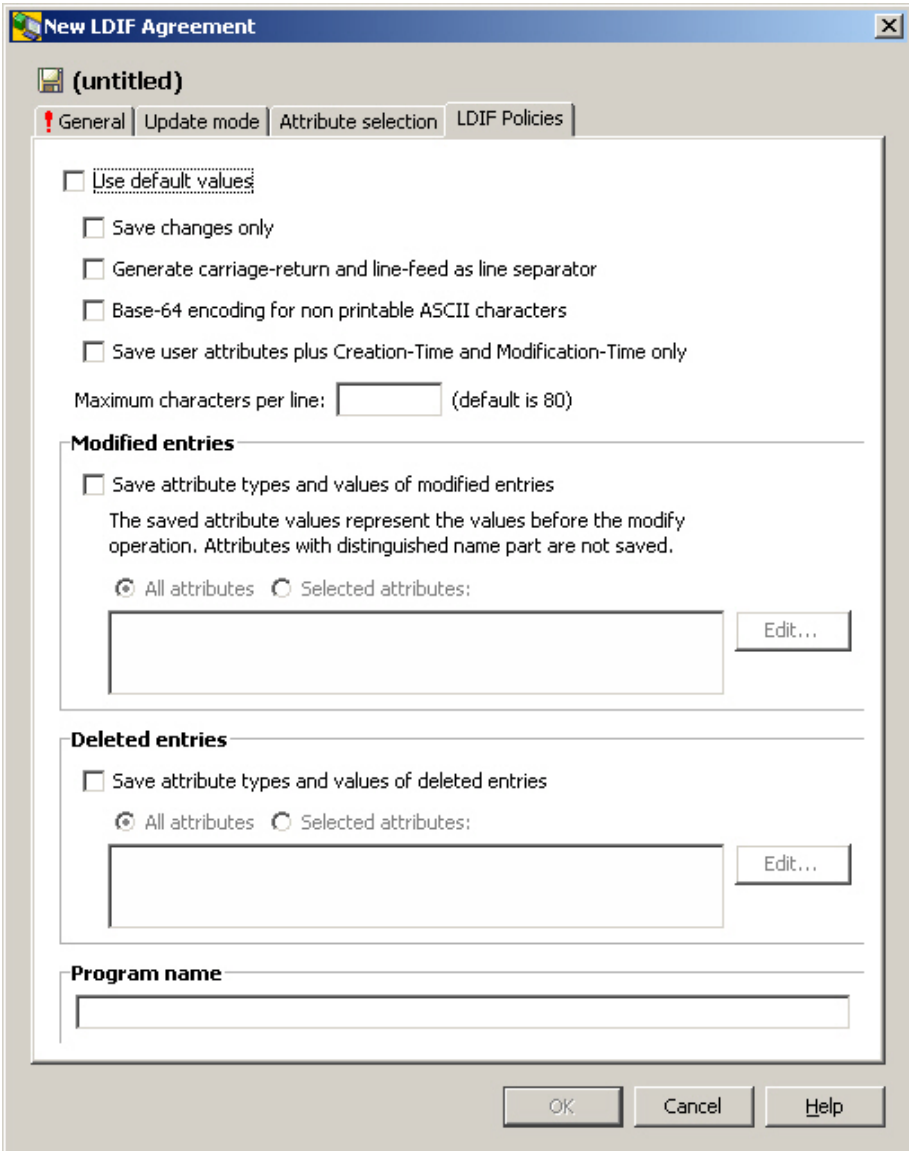


Figure 27. New LDIF Agreement LDIF Policies Tab

Save changes only

There is no initial total.

Generate carriage return and line feed as line separator

By default, only line feed is used (appropriate for Linux; Carriage return/line feed is appropriate for Windows).

Base-64 encoding for non printable ASCII characters

Base64 (see also RFC 2045) regulates how to transform binary data into printable data using nothing but the 64 characters "A-Z, a-z, 0-9, +, /". Base64 increases the original size by 33%; it is not an encryption.

Save user attributes plus Creation-Time and Modification-Time only

Causes mandatory attributes other the createTimestamp and modifyTimestamp to be omitted.

Maximum characters per line

Limits the number of characters per line in the LDIF file.

Modified entries: Save attribute types and values of modified entries
Deleted entries: Save attribute types and values of deleted entries

LDIF agreements cause LDIF change files to be created with the minimum necessary information; in particular, attribute values of modified or deleted entries will not appear in the LDIF file by default. If you want the LDIF file to be more informative, specify the details here.

Program name

Specifies a command string to be performed after the LDIF file has been created.

2.5. Subtree Specification

In subentries such as Access Control Subentry or Collective Attribute Subentry, a subtree specification may restrict the collection of entries to which the subentry applies.

Note that despite its name, a subtree specification does not necessarily define a subtree but may also define more or less scattered partitions of a subtree (therefore also referred to as “entry collections”); this is because subtree specifications allow filtering for object classes and limiting the number of RDNs).

Here is an example of the Subtree Specification tab:

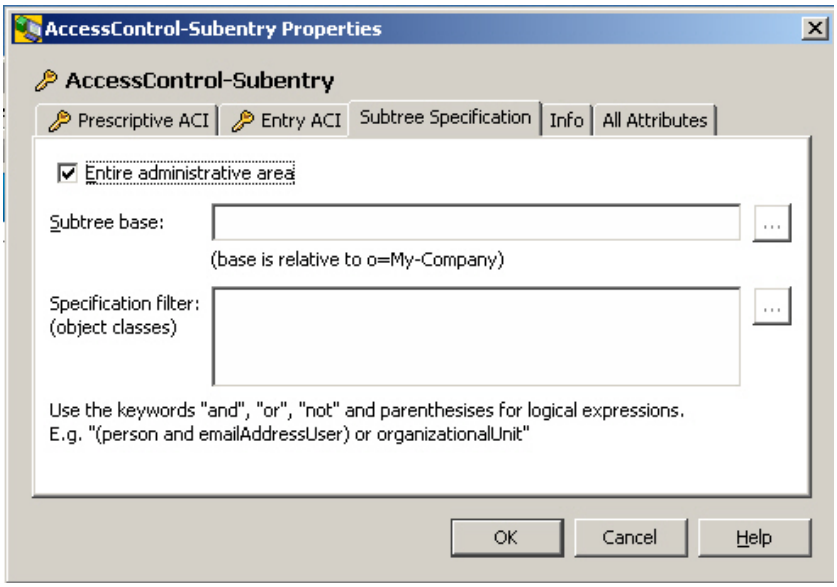


Figure 28. AccessControl-Subentry Properties Dialog Box with Subtree Specification Tab

The available fields are:

Entire administrative area
Subentry applies to the entire administrative area. The remaining fields are only relevant if this field is not checked.

Subtree Base
The real subtree specification.

Specification filter
Subentry applies to all entries below the subtree base matching this filter. Note that this filter must be composed of a logical association of object classes.

Examples:

- Subtree base:

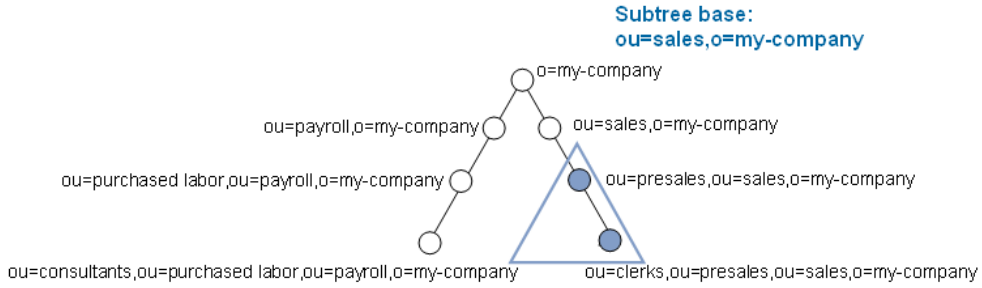


Figure 29. Subtree base example

- Specification filter:

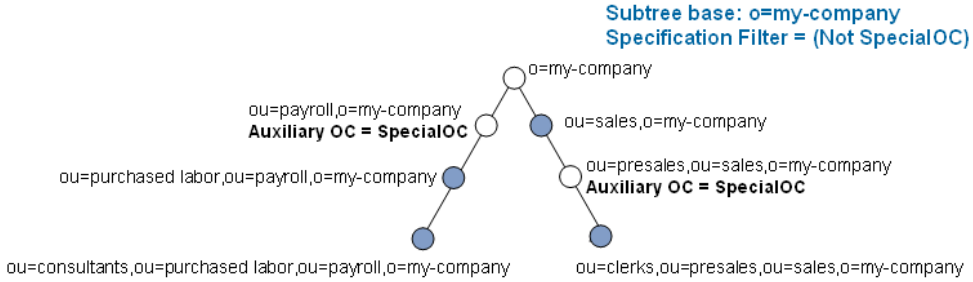


Figure 30. Specification filter example

2.6. About the Administrative Authority Model

The administrative authority model aims at enabling different parts of the DIT to be managed by different authorities. This can be achieved by structuring the DIT into administrative areas. There are two types of administrative areas, autonomous administrative areas (AAA, aka AA for "autonomous area") and inner administrative areas (IAA)

An **autonomous administrative area** starts at an AAP - an "autonomous administrative point" (see more below) - and continues downwards until either leaves or other autonomous administrative points are encountered.

An **inner administrative area** starts at an IAP - an "inner administrative point" (see more below) - that is within an autonomous administrative area. IAPs are always subordinate to AAPs. Inner administrative areas continue downwards until the end of the autonomous administrative area is reached. Entries within an inner administrative area are still under the overall control of the autonomous administrative authority, but some degree of control

is also exercised over them by the administrator of the inner administrative area. Inner administrative areas may be nested, but the bottom of all inner areas is always the bottom of the enclosing AAA.

An entry is called **administrative entry** or **administrative point** if it forms the starting point of an administrative area. Analogous to the administrative areas, there are two types of administrative points: autonomous administrative points (AAP) and inner administrative points (IAP).

Here is an example with four autonomous administrative areas AAP 1 through AAP4. AAP 1 through AAP 3 do not have an inner administrative area, while AAP 4 has three inner administrative areas, where IAP1 and IAP3 are nested:

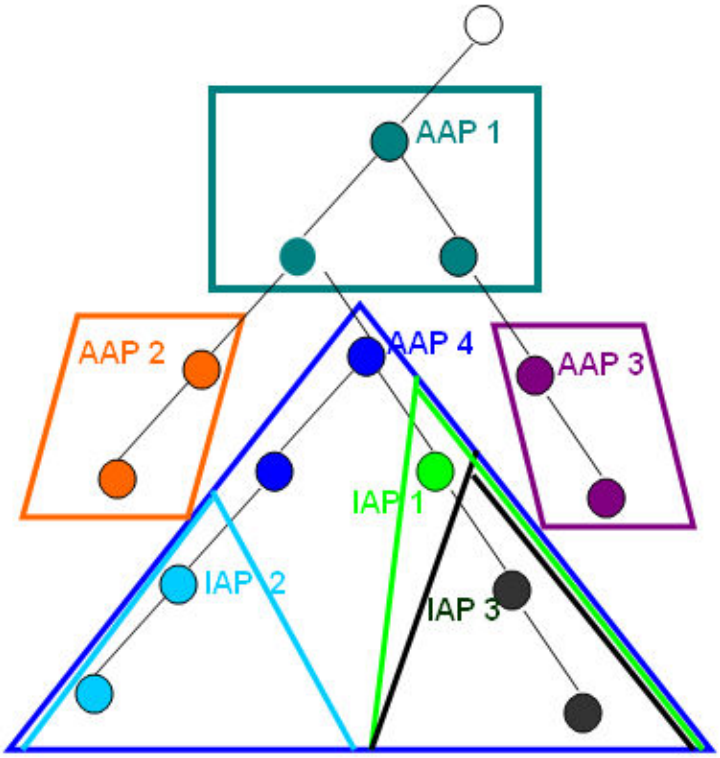


Figure 31. Autonomous Administrative Areas

According to X.500, administrative areas control three specific aspects of administration:

- Subschema administration
  is concerned with administrating the Directory subschema that is in operation in this part of the DIT
  Support discontinued by DirX Directory.
- Access control administration
  is concerned with administrating the security policy that is in force in this part of the DIT
- Collective attribute administration
  is concerned with administrating the collective attributes in this part of the DIT

Access control and collective attributes may need to be partitioned into administrative

areas according to different requirements. That's why it is possible to further partition an AAA into access control specific administrative areas (ACSAs) and collective attribute specific administrative areas (CASAs). Accordingly the IAAs can be further partitioned into access control inner areas (ACIAs) and collective attribute inner areas.(CAIAs).

Here is an example with one AAA partitioned into two ACSAs:

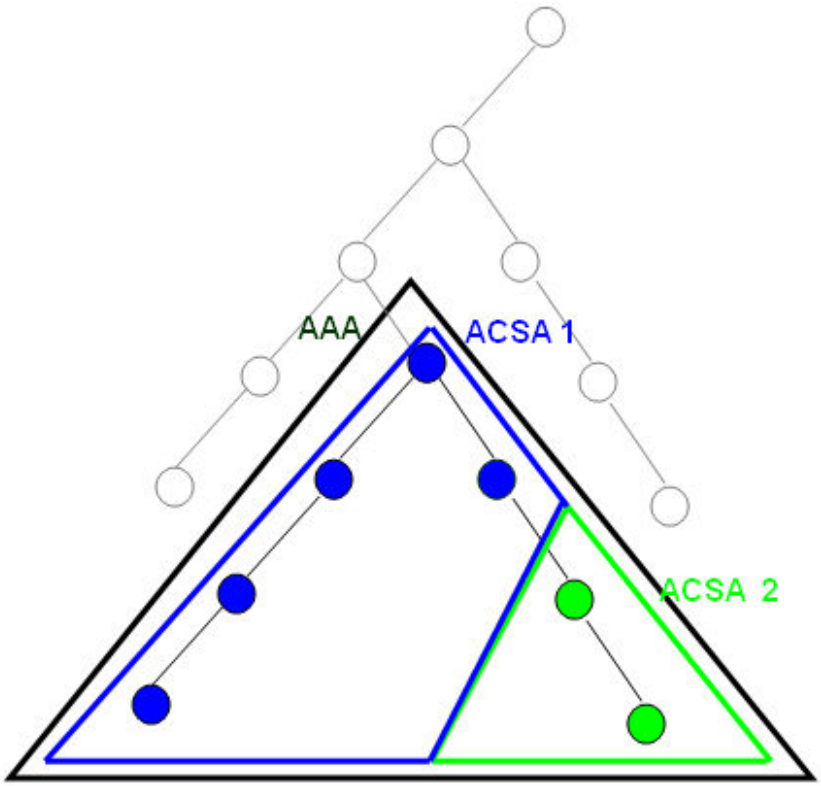


Figure 32. Autonomous Administrative Area (AAA) with Two Access Control Specific Administrative Areas (ACSAs)

In order to administer this, an operational attribute called Administrative Role is available (use the DirX Directory Administration Program **dirxadm** for this purpose).

Since Administrative Role is an operational attribute, it is subject to access control (attribute type administrativeRole). Administrative Role is recurring and can take any reasonable combination of these values:

- Autonomous administrative area (AAA)
  Dispensable. Marks the affected entry as an autonomous administrative point, i.e. as starting point of an AAA. However, in order to effect something, you need to set ACSA or CASA.
- Access control specific area (ACSA)
  Marks the affected entry as starting point of an ACSA. This implicitly makes the entry as an AAP for access control.
  Entries representing the starting point of an ACSA have also an operational attribute called "access control scheme" possible values: BAC oder SAC). This attribute has no effect, since DirX Directory always assumes "BAC".

- Collective attribute specific area (CASA)
  Marks the affected entry as starting point of a CASA. This implicitly makes the entry as an AAP for collective attributes.
- Access control inner area (ACIA)
  Marks the affected entry as starting point of an ACIA. This implicitly makes the entry as an IAP for access control.
- Collective attribute inner area (CAIA)
  Marks the affected entry as starting point of a CIA. This implicitly makes the entry as an IAP for collective attributes.
- Subschema administrative area (SASA)
  Has no effect at all, since DirX Directory has discontinued supporting subschemata in favor of solely supporting the "global schema".

There is in addition a DSE type called "administrative point", which is implicitly set and removed by the server depending on what administrative roles the entry has. An administrative point is insignificant anyway, as long as the administrative role operational attribute is absent.

Specify access control related directives in access control subentries. Specify collective attribute related directives in collective attribute subentries.

3. Schema Management

A directory schema specifies the types of objects ("object classes") the directory can contain and the types of attributes that objects must have and may have. It also specifies quite a number of properties that object classes and attribute types must have and may have. Read more about schema in the topic What is the Schema?.

The schema-related functionality that this application provides is based on LDAP. It may work to some extent with a variety of LDAP servers, but has been tailored to DirX Directory.

The functionality available for schema management includes:

- Available from a special schema view
- Managing the schema, including viewing, modifying, adding, deleting and searching for schema elements.
- Available from Menu
- Exporting the schema in place at application startup, or parts of it, into an LDIF file.
  To export the entire schema, right-click the top node in the tree pane "Export Schema..." or choose from the main menu. To export part of the schema, right-click a node other than the top level node. Note that the application automatically exports the schema in place when it starts up unless it has not changed since the previous startup.
- Importing an LDIF schema file, which merges the selected file (change file or content file) into the current schema.
  To import an LDIF schema file, right-click "Schema" in the tree pane "Import Schema..." or choose from the main menu. Note that servers usually impose restrictions regarding schema modifications.
- Comparing two schemata, which allows you to save the differences into an LDIF file.
  To compare two schema files, right-click "Schema" in the tree pane "Compare Schema... "or choose from the main menu.
- Opening LDIF content files that contain schema elements.

Note that most of this functionality is only accessible while the schema view is selected.

3.1. What is the Schema?

Databases typically have a schema. A schema is the collection of provisions that regulate the content of a database more or less rigidly. Since a directory is a kind of specialized database, it has a schema, too. A directory schema is configurable rather than fixed. However, directory servers typically come with a built-in schema that can be extended but which supports either restricted modification or no modification at all.

A directory contains entries. Entries are composed of a "Distinguished Name" (DN) and a number of attributes. An entry is uniquely identified by its DN. DNs are composed of relative DNs (RDNs). RDNs must be derived from attributes. Entries must have a special recurring attribute that stores object classes. Object classes typically define the attributes that an object must have and may have. However, some directory servers allow you to switch off the "schema check" function. In this case, the object class is not much more than

another possible search filter (read more about this in the "Using LDAP" topics).

A directory schema, then, basically consists of:

- The supply of available attributes
- The supply of available object classes

3.1.1. Attributes

Attributes have or may have

- A **name**
  A list of short names (meaningful names, if possible) of the attribute. In this application, **name** is split up into "Name" (the first one) and "Aliases" (any remaining ones).
- A **description**
  A short descriptive string.
- An **object identifier**
  A numeric object identifier ("OID").
  An object identifier is a dot-separated sequence of numbers that should uniquely identify the attribute. Numerous attributes have already been assigned a registered object identifier (for details, refer to the *DirX Directory Administration Reference*). Registration authorities own particular disjoint sets of object identifiers.
- Optional: a **superior**
  Superiors allow for a kind of grouping of attributes. For example, if the attributes "commonName" and "surname" both have the same superior "name", requesting "name" to be returned in a search implicitly includes commonName and surname to be returned as well. Searching for "name contains Smith" matches all entries with "commonName contains Smith" or "surname contains Smith" as well.
  Note that the subtypes share the syntax and matching rules of their supertype (=superior).
- A [.indexref]##**Syntax**
  Examples of possible syntaxes are: Boolean, Country String, Certificate, DN, Directory String, Generalized Time, IA5, Integer, JPEG, Numeric String, Postal Address, Telephone Number.
- Optional: **Single-value**
  By default, attributes are multi-valued (also referred to as "recurring").
- Optional: **Collective**
  Indicates that the attribute type is collective. Collective attributes provide a means by which many entries can share a single attribute that is administered in a single place. Collective attributes are not necessarily supported by the server.
- Optional: **No user modification**
  Indicates that the attribute is not user modifiable.
- Optional: **Obsolete**
  Indicates that the attribute type is not active; that is, you can no longer assign it when creating new entries or modifying existing ones.
- Optional: [.indexref]##**Matching rules**

What is returned as search result depends upon the conditions under which the software considers a value presented by a user to correctly match a value stored in the directory. This is where matching rules come into play. Matching rules must be compatible with the corresponding attribute syntax. A matching rule specifies how attribute values are to be matched for equality, for ordering, and for substring comparison. Note that attributes that do not have a matching rule assigned to them cannot be used in search filters.

- Optional: **Usage**
  Indicates the application of the attribute type:
- userApplication means it is a user attribute
- directoryOperation means it is a directory operational attribute
- distributedOperation means it is a DSA-shared usage operational attribute
- dSAOperation means it is a DSA-specific operational attribute

3.1.2. Object Classes

Object classes have or may have

- A **name**
  A short name (which is meaningful, if possible) of the object class.
- A **description**
  A short descriptive string.
- An **object identifier**
  A numeric object identifier ("OID").
  An object identifier is a dot-separated sequence of numbers that should uniquely identify the object class. Numerous object classes have already been assigned a registered object identifier (for details refer to the *DirX Directory Administration Reference*). Registration authorities own particular disjoint sets of object identifiers.
- Optional: a **superior**
  Object classes inherit the mandatory and optional attributes from their superiors.
- Kind
  There are three possible kinds of object classes:
- A**bstract**
  Abstract object classes are used only to derive other object classes. Top is an abstract object class from which every structural object class is directly or indirectly derived (auxiliary object classes are usually derived from Top, too, but this is not mandatory). In many real world schemata, it is the only abstract object class you will encounter.
- **Structural**
  Each entry must have at least one structural object class. If an object class happens to be derived from another object class, the affected entry has both object classes and inherits all attribute types from the parent object class. However, not counting "parent classes", each entry has one and only one structural object class.
- **Auxiliary**
  In addition to structural object classes, entries may have one or more auxiliary object classes. As opposed to structural object classes, auxiliary object classes can be added to

and removed from an entry at any time. Auxiliary attributes are associated with a (possibly empty) set of attribute types. Auxiliary classes provide a convenient means to add attribute types (the ones that are associated with that auxiliary object class) dynamically to entries that are already defined by a structural object class. So, auxiliary object classes allow you to dynamically extend the permissible attribute set of entries beyond the one defined by their structural object classes. If the set of associated attribute types is empty, the auxiliary object class is an object class that at the same time is sort of an ordinary attribute. It may however serve as a filter for access control definitions in case the access control provided by the server follows the X.500 specification

- Optional: **Obsolete**
  Indicates that the object class is not active; that is, you can no longer assign it when creating new entries or modifying existing ones.
- **Mandatory Attributes**
  A list of all attributes any entry with the object class must have.
- Optional: **Optional Attributes**
  A list of all attributes an entry with the object class may have.

3.2. Core Functionality

The core functionality allows to you manage schema elements. Particularly, you can:

- View, modify, add and delete attribute types
  Note that the server may refuse the operation
- Add: Right-click "Attributes" in the tree or list pane "New attribute" (or choose from the main menu).
- Delete: The delete function just causes the attribute to be marked "obsolete".
- Undelete: Causes to undo a previous delete, i.e. the attribute is no longer marked "obsolete".
- View/Modify:
- Double-click the attribute in question in the tree or list pane *or*
- Right-click it in a field within a property pane or property dialog and select "Properties"
- View, modify, add and delete object classes
  Note that the server may refuse the operation
- Add: Right-click "Object classes" in the tree or list pane "New object class" (or choose from the main menu).
- Delete: Apply the Delete function (right mouse button) to the object class in question in order to mark that attribute obsolete. Deleting means this object class can no longer be assigned but remains visible in existing entries that already have it. Object classes marked obsolete are displayed grayed out and cannot be modified unless the Undelete function (right mouse key) has been applied.
- View/Modify
- Double-click the object classes in question in the tree or list pane *or*

- Right-click it in a field within a property pane or property dialog and select "Properties"

Additionally, the application provides some convenience functionality:

- Search
  A search tab offers diverse ways to find all object classes and/or attributes matching a given search filter.*
  Note that the preferable way to find out to what object classes a particular attribute is assigned is by clicking the attribute in question in the list pane and finding the object classes listed in the corresponding property pane (or property dialog).*
- Associated object classes
  The property dialogs/panes of attributes offer a tab ("Associated object classes") that displays the object classes to which the actual attribute is currently assigned.
- Column headers
  As in many lists throughout this application, right-clicking a column header allows you to choose what properties you can see in a list of attributes and/or object classes and sort them according to your needs.

The schema management plug-in typically presents the schema in a combination of a tree pane, a list pane and a property pane.

Here is an example of the tree pane:

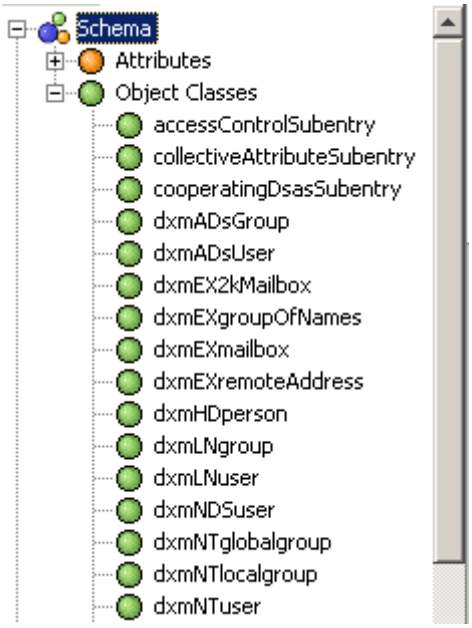


Figure 33. Schema List Pane

Notes:

- As opposed to the list pane, when you expand attributes or object classes, the tree pane displays the *hierarchy* that is defined by the "*superior*" attribute.
- Obsolete attributes/object classes are displayed *grayed out and italic*.

Here is an example of the search pane:

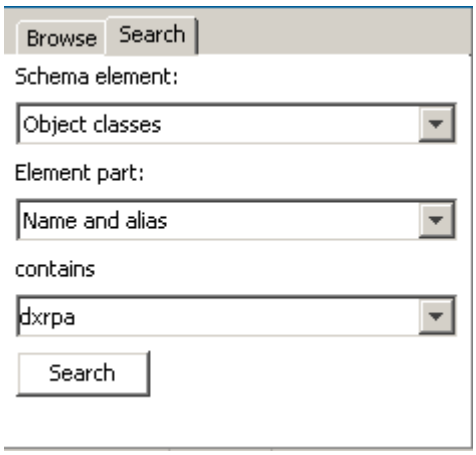


Figure 34. Schema Search Pane

Here is an example of the list pane:

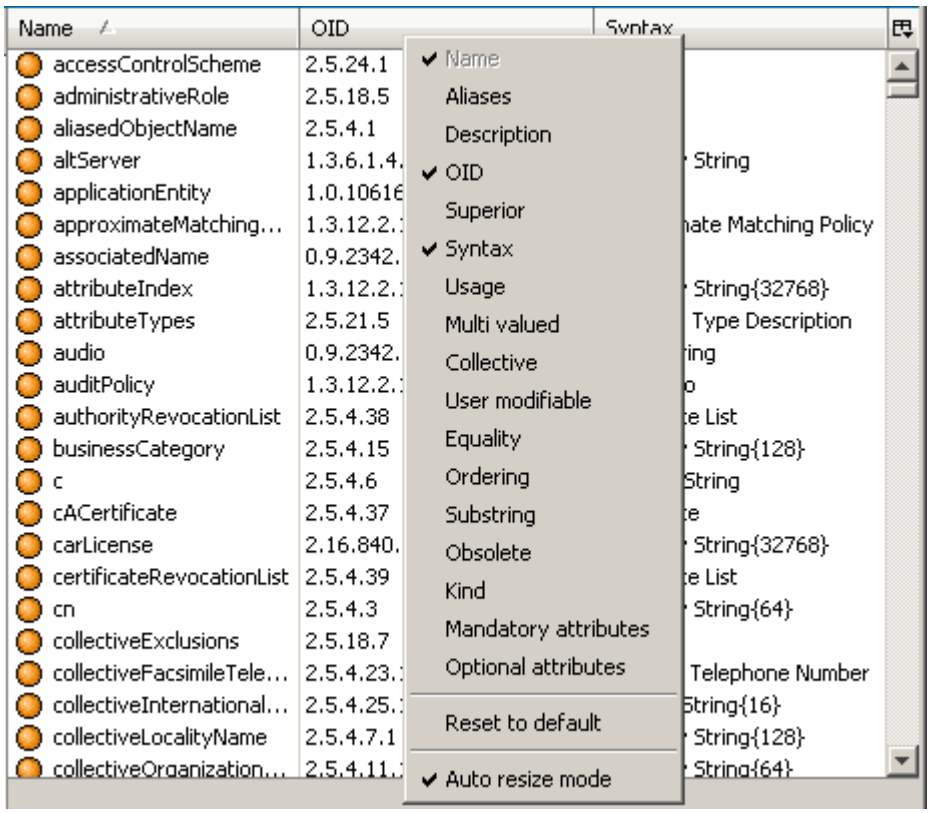


Figure 35. Schema List Pane

The list pane does not show any hierarchies. The list pane displayed when you click "Attributes" or "Object Classes" comprises *all* attributes resp. *all* object classes, while the list displayed when you click lower-level nodes in the tree only displays the respective entries.

Here is an example of the property dialog that displays the associated object classes of the attribute "cn":

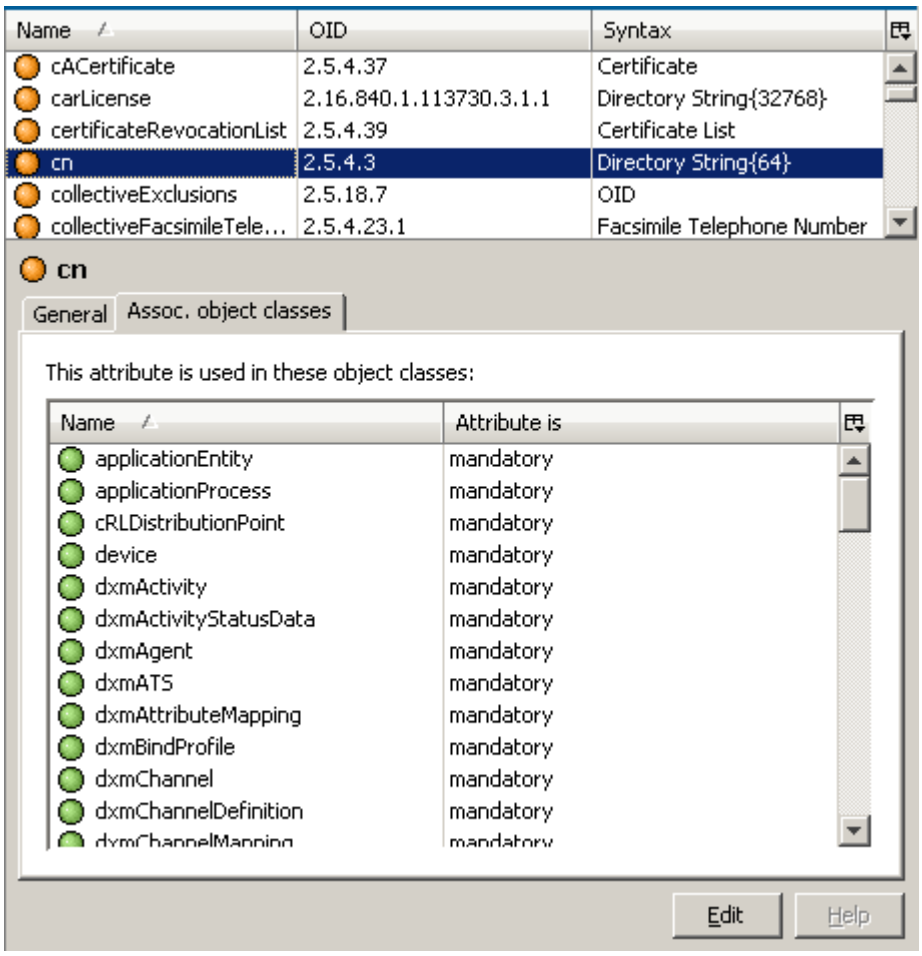


Figure 36. Schema Property Dialog

Refer to the topics Managing Attribute Types or Managing Object Classes for more information about the property pane/property dialog.

3.2.1. Managing Attribute Types Overview

This functional area allows you to manage the attribute types that are to be known by the server.

In addition to the ordinary functionality of managing the attribute types, you can directly view, in which object classes a currently selected attribute is used.

3.2.1.1. Managing Attribute Types

When managing a schema that is stored in a file system file rather than in a directory server, pay regard to the respective note on the property panels provided within the function "Open LDIF Schema File".

To create a new attribute type, right-click "Attributes" or an entry representing an attribute in a tree or list pane and select "New Attribute". A dialog like the one shown in the following figure is displayed:

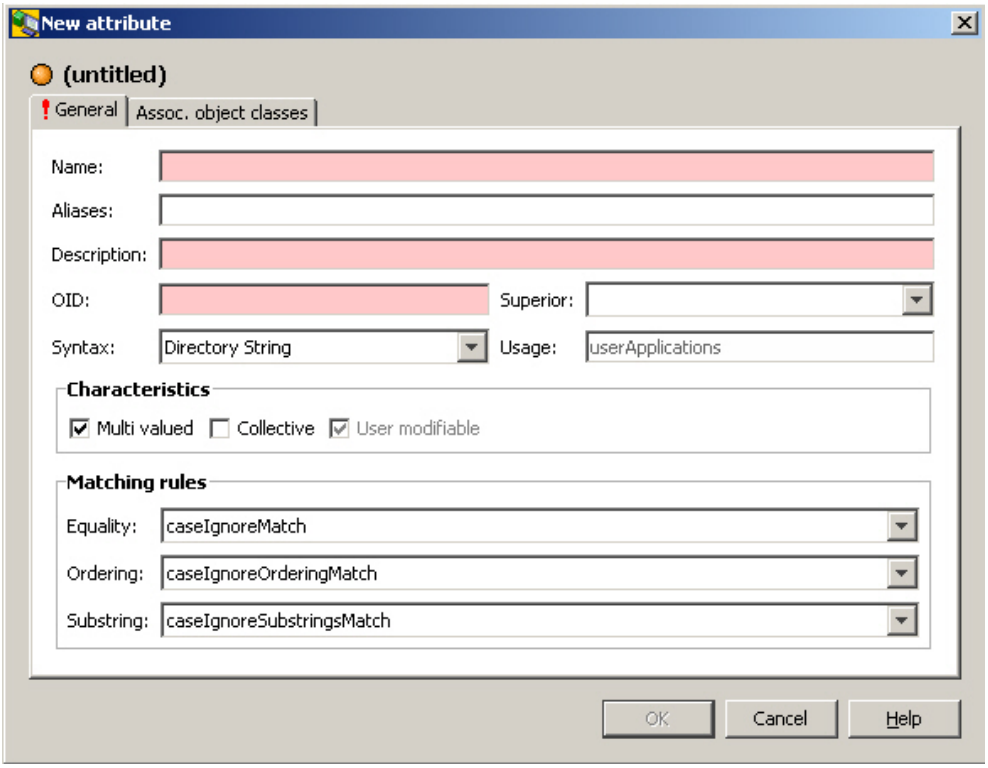


Figure 37. New Attribute Dialog Box

Notes (also see the topic What is the schema? -> Attributes):

- Attribute names must not contain blanks nor must they contain special characters like umlauts. They must begin with a letter.
- Name/Aliases
  If an LDAP server returns more than one name, the first one is displayed in the "Name" field, the other ones are displayed in the "Aliases" field (blank, comma or semicolon separated). As a result, you must separate any names you specify in the aliases field. Note that although the name is optional according to LDAP, this application treats it as mandatory.
- Description
  Note that although the description is optional according to LDAP, this application treats it as mandatory
- OID (Object Identifier; mandatory)
  The OID should comply with a registration authority. Even if it does not, you may still be required to conform with certain rules imposed by the server.
- Superior
  If you select a superior, you can no longer select a syntax, since the syntax is now inherited from the superior.
- Syntax
  Note that although the syntax is optional according to LDAP, this application treats it as mandatory. The available choice of syntaxes varies depending on the server.
- Usage
  You can only create attributes with "usage" is "userApplications". Other usages are subject to server-specific attributes only.

- Multivalued
  Specifies that the attribute can get multiple values.
- Collective
  Specifies that the attribute is a collective attribute valid for a number of entries.
- User modifiable
  An attribute marked as "user modifiable" states that users can (within the bounds of their access rights) add/delete/modify values of that attribute. Server specific attributes like createTimestamp are typically not user modifiable. Attributes created by this application are implicitly marked user modifiable. This application does not support changing this attribute characteristic.
- Obsolete
  Setting this flag usually means the attribute can no longer be assigned but remains visible in existing entries that already have the attribute assigned. Setting this flag is only possible when modifying an attribute; you cannot set it when creating one.
  To make an attribute obsolete, apply the Delete function (right mouse button) to it. Attributes marked obsolete are displayed grayed out and cannot be modified unless the Undelete function (right mouse key) has been applied.
- Matching rules
  Specifies the matching rules for the attribute type.

To view or modify an attribute type, right-click or double-click the attribute in a tree or list pane. A dialog like the one shown here is displayed (if a property pane has been configured, too, it looks almost the same):

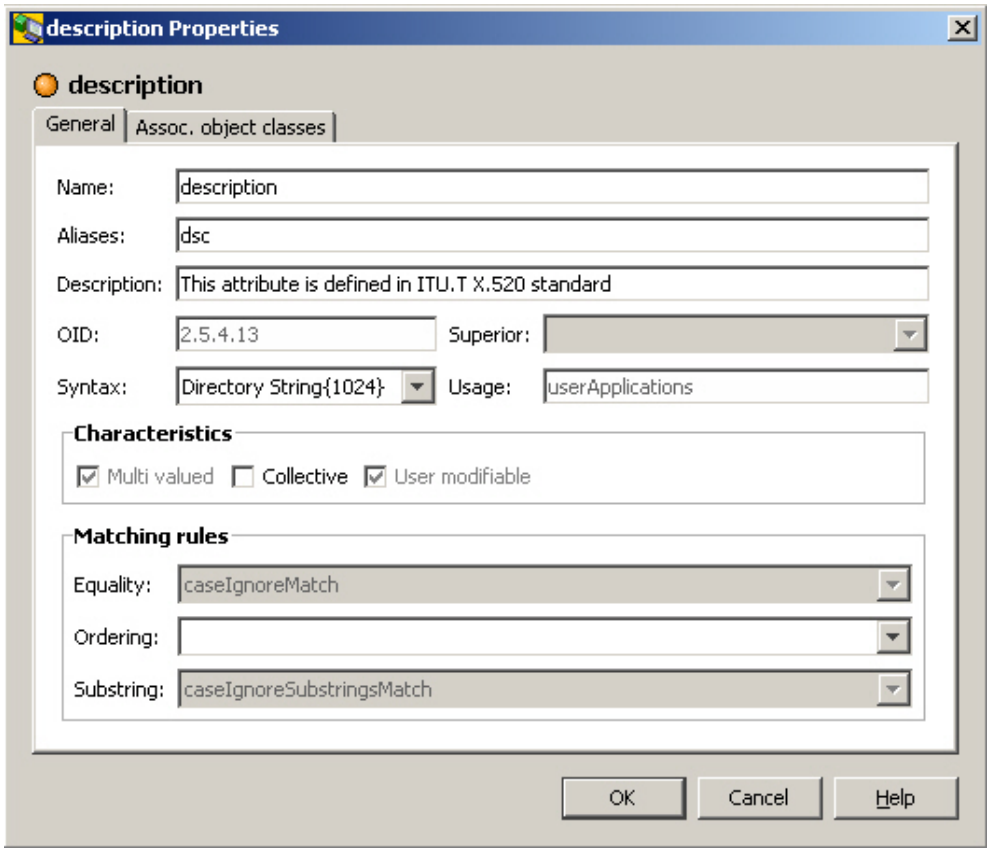


Figure 38. Description Properties Dialog Box

Attributes whose usage is dSAOperation cannot be modified at all.

As for existing attributes whose usage is other than dSAOperation, you can only modify:

- Name/Aliases
- Description
- Multivalued
  Note that you cannot clear the check button, only check it off. Once an attribute is marked multivalued, it cannot be changed to single-valued.
- Collective
- Matching rules
  Note that you can only provide a yet missing equality matching rule or - provided an equality matching rule is specified - a yet missing ordering and/or substring matching rule.

Click the tab "Assoc. object classes" to see quickly the object classes to which the attribute is assigned.

By right-clicking an attribute in a tree or list panel you can "delete" it, which means that it will be marked "obsolete".

3.2.1.2. Object Classes that Use a Particular Attribute

To determine the object classes to which a particular attribute is assigned, click the attribute in the list pane. The object classes that have the attribute assigned to them are listed in the corresponding property pane (or property dialog).

Here is an example that shows all object classes that have the attribute **cn** assigned.

3.2.2. Managing Object Classes

Object Classes with Attribute cn

When managing a schema that is stored in a file system file rather than in a directory server, pay regard to the respective note on the property panels provided within the function "Open LDIF Schema File".

To create a new object class, right-click "Object Classes" or an entry representing an object class in a tree pane and select "New Object Class". A dialog like the one shown here is displayed:

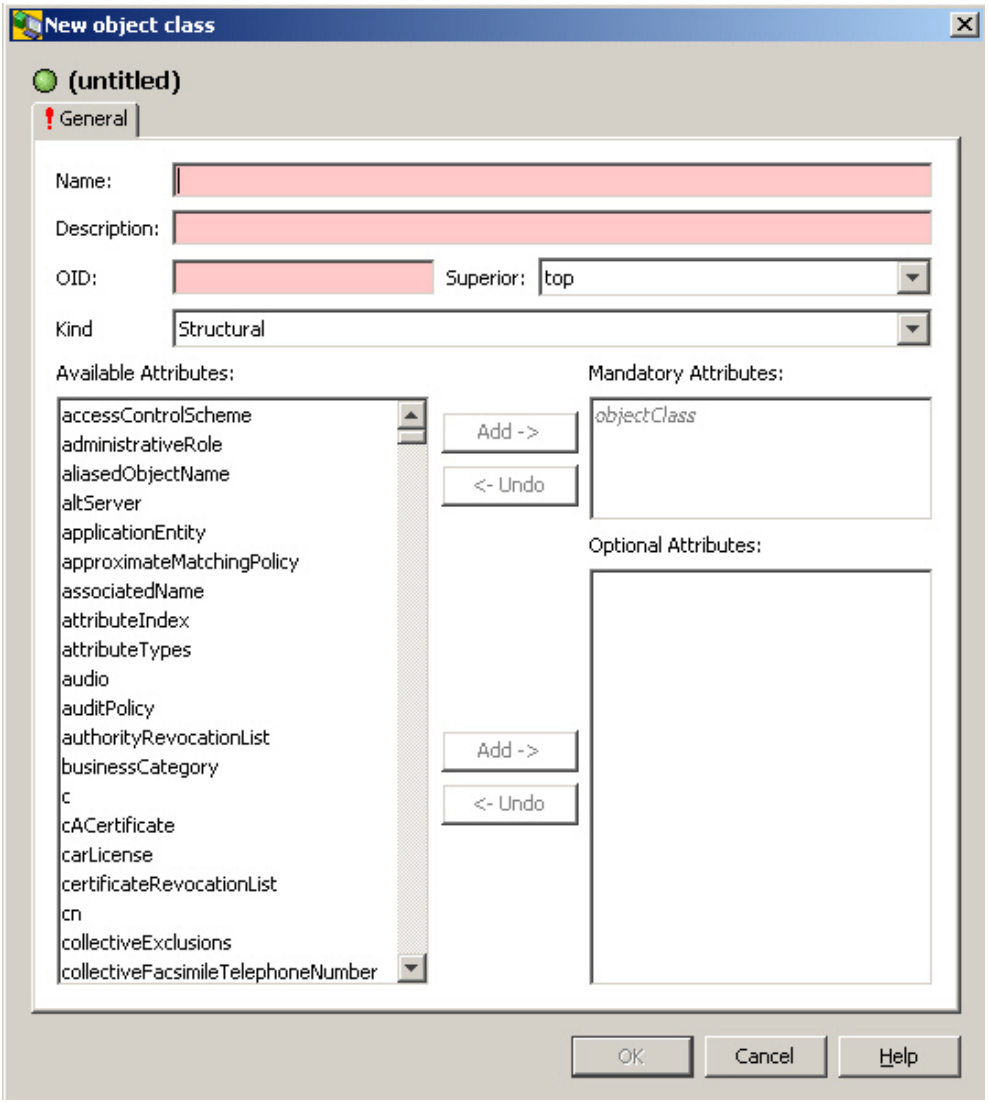


Figure 39. New Object Class Dialog Box

Notes (see also What is the Schema? → Object Classes):

- Object class names must not contain blanks nor must they contain special characters like umlauts. They must begin with a letter.
- Name
  Note that although the name is optional according to LDAP, this application treats it as mandatory.
- Description
  Note that although the description is optional according to LDAP, this application treats it as mandatory.
- Superior
  If you select a superior, that superiors's attributes are inherited. Inherited attributes are displayed *grayed out and italic*. Attributes that at the same time are inherited and directly assigned are displayed in the normal color and font.
- OID (Object Identifier; mandatory)
  The OID should comply with a registration authority. Even if it does not, you may still be required to conform with certain rules imposed by the server.

To view or modify an object class, right-click or double-click the attribute in a list or tree pane. A dialog like the one shown here is displayed:

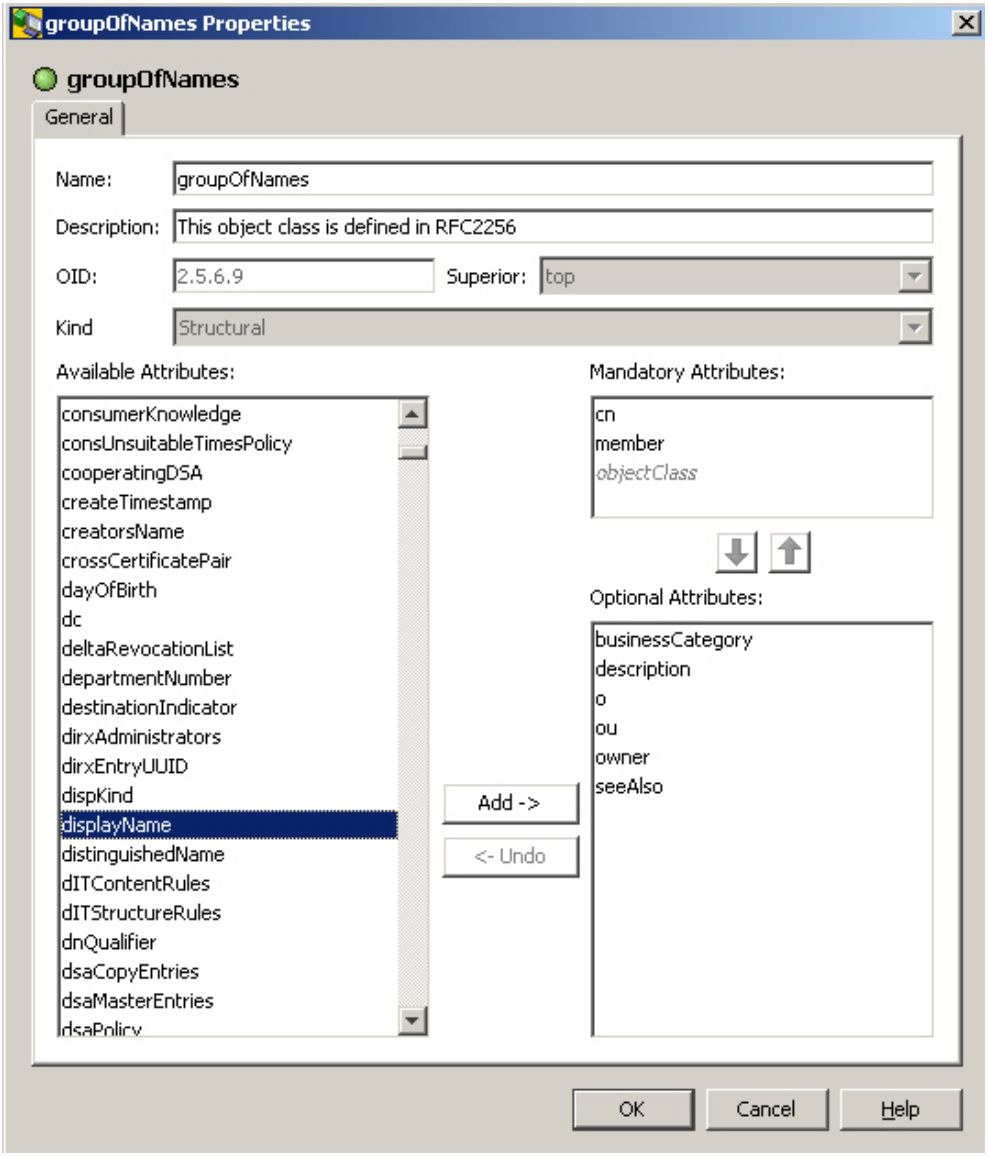


Figure 40. groupOfNames Properties Dialog Box

Once you have created an object class, you can only change

- Name
- Description
- Attributes from mandatory to optional. Not possible for derived attributes and for attribute "object class".*
  Note that moving an optional attribute to mandatory is not possible. Particularly, if you have moved an attribute from mandatory to optional, you can only undo this as long you haven't saved your modification.*
- Optional attributes (may be extended, i.e. must contain all existing attributes and may contain new values)
  You can undo an attribute assignment, so long as you haven't confirmed the assignment by clicking OK.

By right-clicking an object class in a tree or list panel you can "delete" it, which means that it will be marked "obsolete".

If a property pane is configured, panes like the ones shown in here are displayed:

Property pane (read mode):

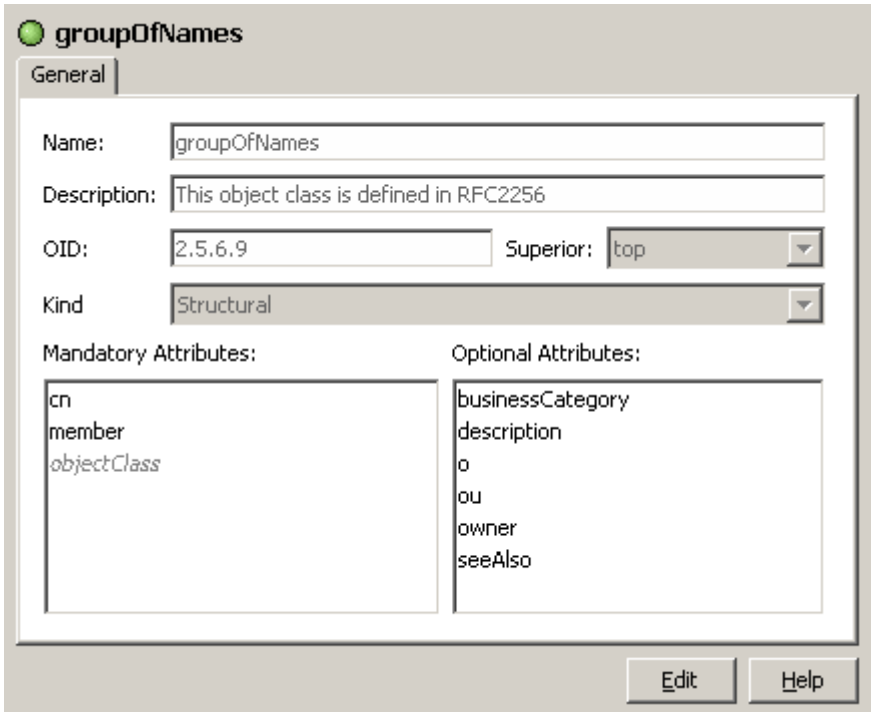


Figure 41. groupOfNames Properties Pane in Read Mode

Property pane (edit mode):

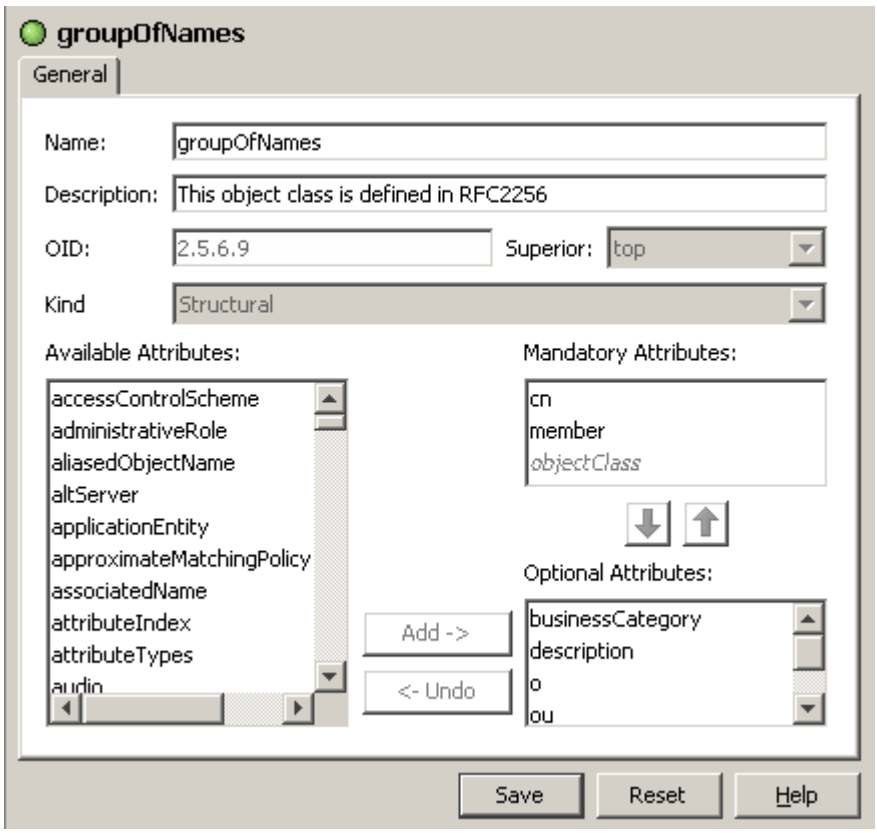


Figure 42. groupOfNames Properties Pane in Edit Mode

3.3. Complementary Functionality

Beyond managing the schema in narrower sense, you can

- Export a schema
- Import a schema
- Compare schemata
- Open a file containing a schema in LDIF format

3.3.1. Exporting a Schema

When you export the schema, a dialog like this one appears:

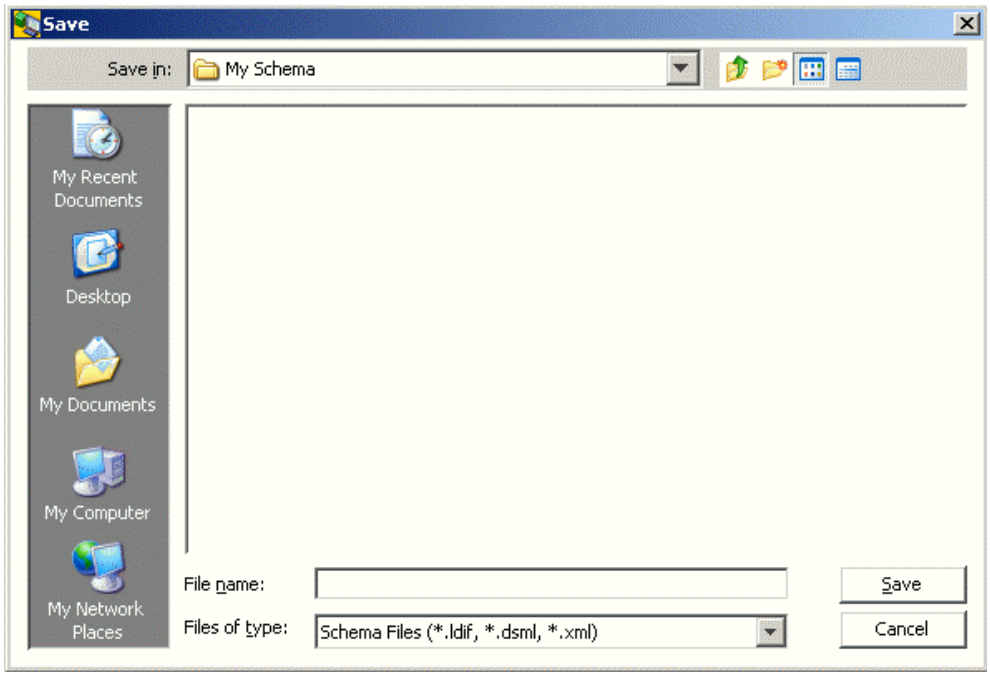


Figure 43. Exporting a Schema Save Dialog Box

Notes:

- If you click **Save** and nothing happens, check whether you forgot to specify a file name.
- You can export parts of the schema by selecting one or more attributes or object classes and right-clicking them.

3.3.2. Importing a Schema

When you import a schema contained in an LDIF schema, a regular file selection dialog appears, followed by a dialog that indicates the operations that are to be applied to the server's schema object on a per attribute/object class basis in order to import the schema from the LDIF file into the server. Note that most directory servers impose fairly rigid restrictions on schema imports, particularly on schema modifications.

The LDIF file that contains the schema to be imported can be an LDIF content file or an LDIF change file. When operating on an LDIF change file:

- Delete operations are ignored.
- Modify operations use only the "add" element from the file, while the current value in the server takes the place of the "delete" element in the file.
- Add operations are implicitly turned into modify operations for attributes that are already present in the server's schema.

When operating on an LDIF content file, "add" and/or "modify" operations are created and executed based on a comparison between the schema currently in place at the server and the schema in the LDIF file.

In the dialog that indicates the operations to be applied to the server's schema, you can:

- Uncheck attributes and/or object classes you do not want to import

- Click the details button adjacent to the "Operation" column to examine the details of the corresponding attribute or object class

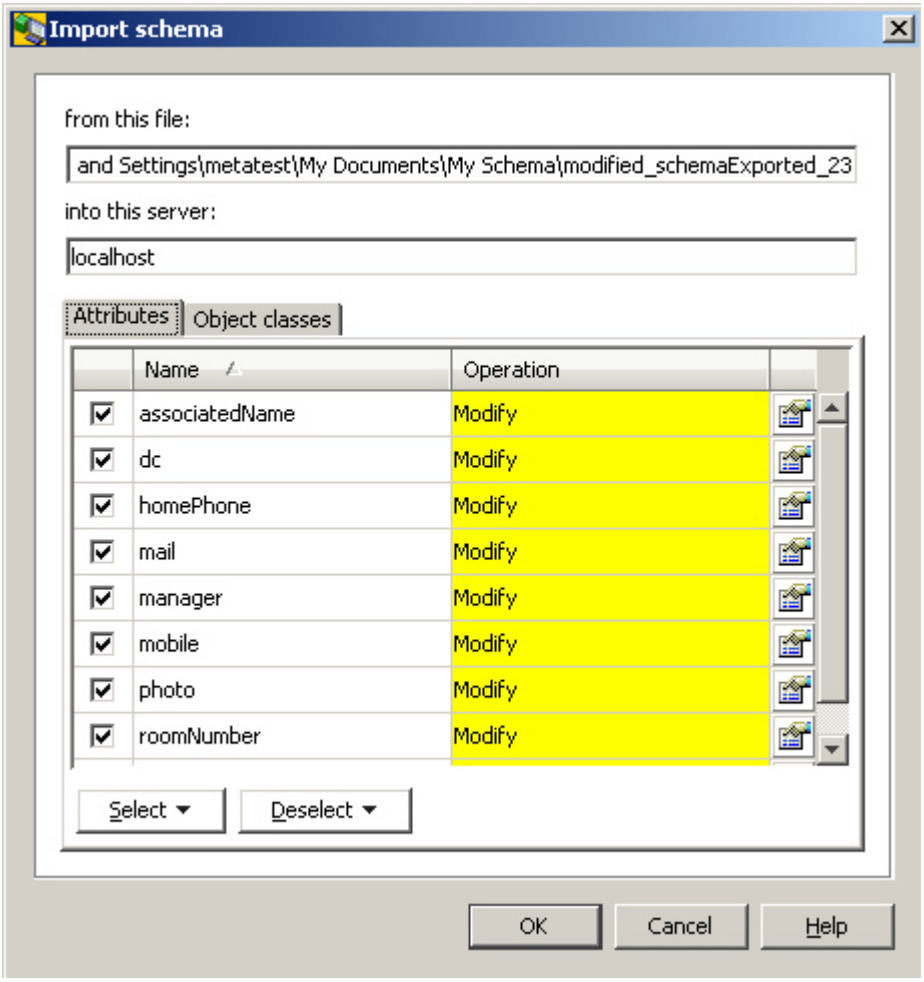


Figure 44. Import Schema Dialog Box

3.3.3. Comparing two Schemata

This functional area allows you to compare two schemata from various sources.

You also can save all or some of the differences - if you want after editing some schema information - into an LDIF file.

3.3.3.1. Comparing two Schemata

The "Compare schema" function allows you to compare two given schemata. Here is an example of the "Compare schema" dialog:

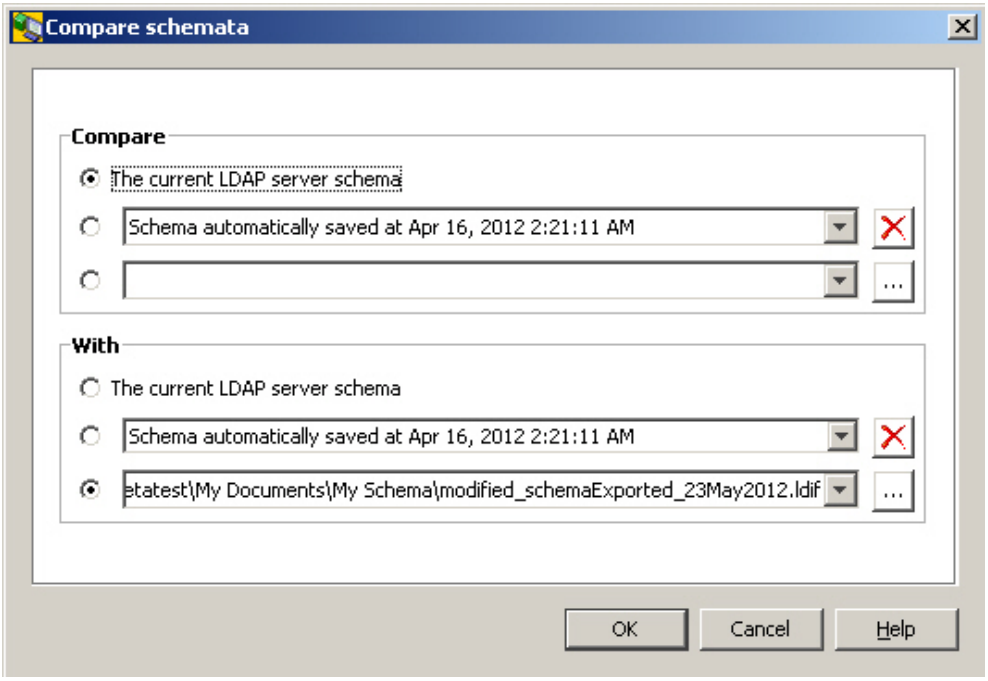


Figure 45. Compare Schema Dialog

Notes

- Schemata that are available in files can only be compared, if the format is LDIF content.
- As kind of side effect, the dialog allows you to automatically delete exported schemata by clicking the  button.

Click **OK** to run the schema comparison. The application displays a dialog that indicates the schema differences:

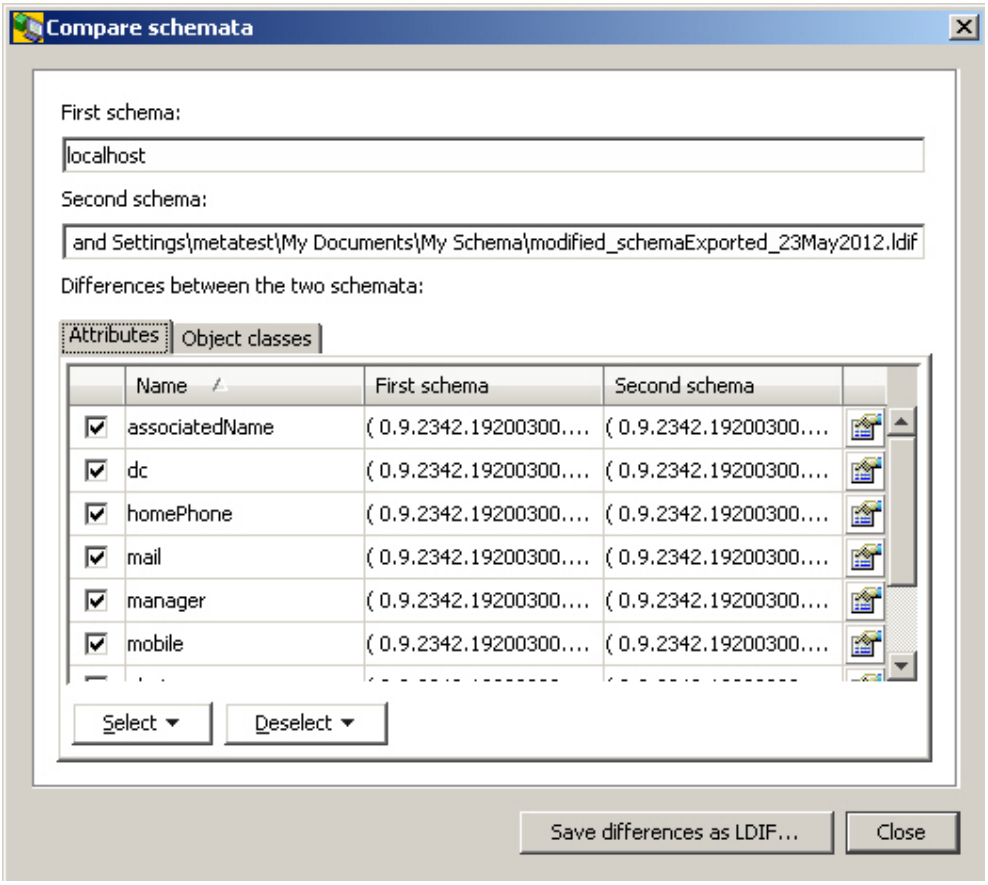


Figure 46. Compare Schema Result

You can use this dialog to save the differences into an LDIF file. Before saving the differences, you can:

- Uncheck any attributes and/or object classes that you do not want to save (or use the **Select** and **De-select** buttons)
- Edit values by clicking the **Details** button

3.3.3.2. Saving Schema Differences

Here is an example of the dialog that allows you to save the differences between two schemata:

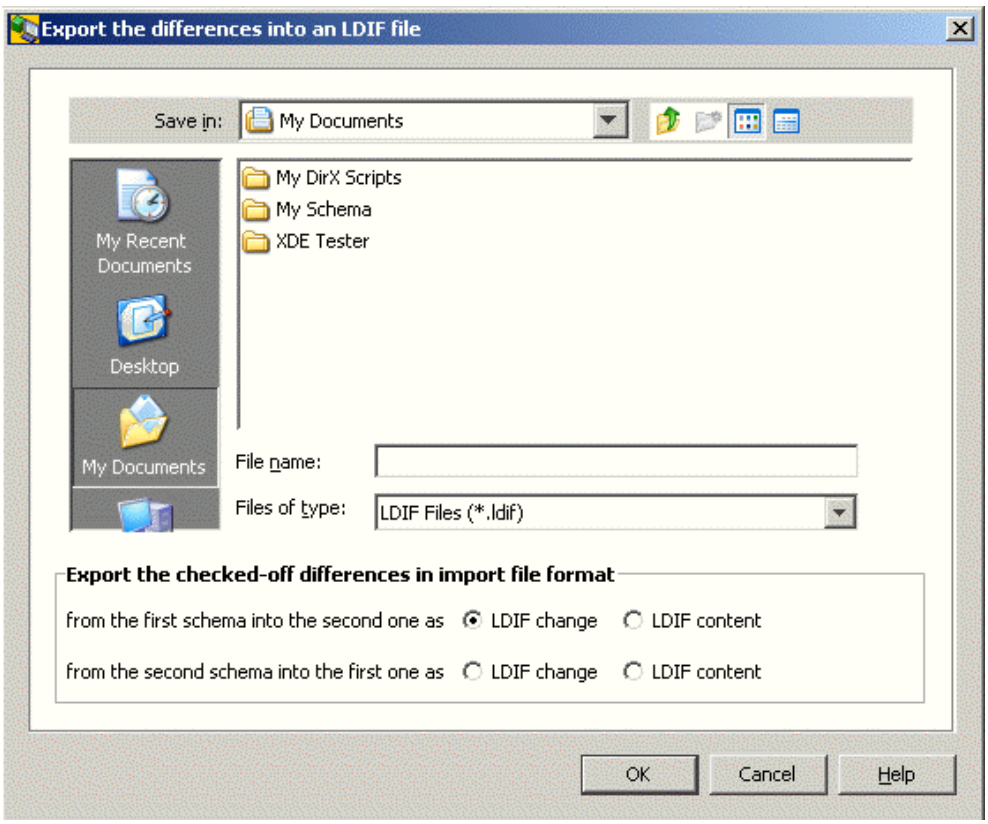


Figure 47. Saving Schema Differences Dialog

There are several ways you can save the differences:

- You can generate an import file from the *first* schema into the *second* one as an LDIF *change* file. In this case:
- If an item (attribute or object class) is present in the first schema and missing in the second one, an "add" operation is generated.
- If an item (attribute or object class) is present in the second schema and missing in the first one, the item is omitted.
- If an item (attribute or object class) is present in the first schema and (with differences) also present in the second one, a "modify" operation is generated.
- You can generate an import file from the *second* schema into the *first* one as an LDIF *change* file. In this case:
- If an item (attribute or object class) is present in the first schema and missing in the second one, the item is omitted.
- If an item (attribute or object class) is present in the second schema and missing in the first one, an "add" operation is generated.
- If an item (attribute or object class) is present in the first schema and (with differences) also present in the second one, a "modify" operation is generated.
- You can generate an import file from the *first* schema into the *second* one as an LDIF *content* file. In this case:
- If an item (attribute or object class) is present in the first schema and missing in the second one, the item is saved.

- If an item (attribute or object class) is present in the second schema and missing in the first one, the item is omitted.
- If an item (attribute or object class) is present in the first schema and (with differences) also present in the second one, the item is saved.
- You can generate an import file from the *second* schema into the *first* one as an LDIF *content* file. In this case:
- If an item (attribute or object class) is present in the first schema and missing in the second one, the item is omitted.
- If an item (attribute or object class) is present in the second schema and missing in the first one, the item is saved.
- If an item (attribute or object class) is present in the first schema and (with differences) also present in the second one, the item is saved.

3.3.4. Opening an LDIF Schema File

This function allows you to view and manage schemata stored in an LDIF **content** file rather than in a server. The interface is the nearly same as the one for managing schema elements, but the LDIF schema is displayed in an own window and can be edited almost unlimitedly, while the editing capabilities of the server schema are rather limited by server-side constraints.

Note that the "Save" button found in the property panels - while typically saving changes in the directory server - saves your changes only tentatively in main memory. To persistently store them on disk, subsequently click "Save in (different) file" (or "Close" and confirm the question that is popping up).

4. Database

The database related functionality of this application deals with

- Consistency of Subordinates
- Indices

4.1. Consistency of Subordinates

In order to check the consistency of subordinates, just apply the right mouse key to the database node in the schema view and select "Check consistency of subordinates" (but be aware that while the check is running no updates are possible):

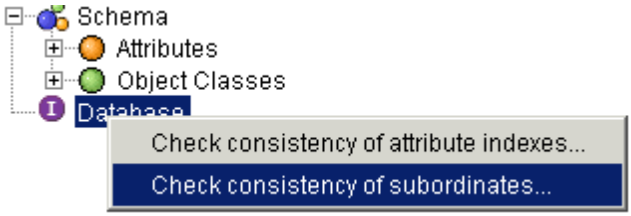


Figure 48. Schema View of Database Node

You can

- have the server check the entire tree or a specified subtree.
- have it automatically repair inconsistencies encountered during the check.

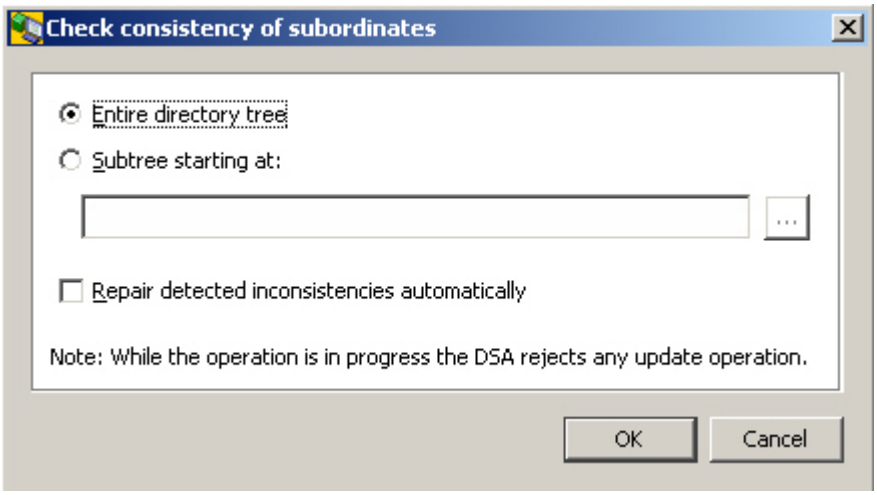


Figure 49. Check consistency of subordinates dialog box

4.2. Indices

The purpose of indices is to enable the server to respond as quickly as possible to search queries. In a nutshell, indices are the more beneficial the larger the database - and they are absolutely essential for searching large databases.

You can have the server create and maintain indices on a per attribute basis. These types of

indices are available:

- **Index**
  Optimizes the performance of search queries with a filter that contains an **equality, greater or equal, less or equal,** or **approximate** (aka "sounds like") match item or an **initial** (aka "begins with") or **final substring** (aka "ends with") in the targeted attribute.
- **Contains** (only possible in addition to "**Index**")
  Optimizes the performance of search queries with a filter that contains **substrings** that can be positioned anywhere in the targeted attribute (aka "contains", "contains any word of", "contains each word of" and the like).
- **Any** (only possible in addition to "**Index**")
  Optimizes the performance of search queries with a filter that checks for the **presence** of the targeted attribute, i.e. that checks if the targeted attribute has a whatever value (attribute is present) or no value (attribute is not present). Note that a search such as objectClass="*" is equivalent to "objectClass is present"
- **Approximate** (only possible in addition to "**Index**")
  Optimizes the performance of search queries with a filter that contains sounds-like filter items (phonetic matching).

Since indices can take up a considerable amount of disk space, it might make sense to consider for each attribute in question whether it is worth or not to have the server maintain an index for it. While in a typical directory that stores users, attributes such as commonName and surname are typical candidates for indices, it appears questionable if it pays to have the server maintain indices for attributes such as givenName or initials. The server is smart enough to adequately handle queries containing both, indexed attributes and attributes that are not indexed.

Indices can created with **UNIQUE** constraint. The **UNIQUE** constraint proves uniqueness of attribute values. (See the **dirxadm db** operation for details.)

The index management property panes differ depending on if they are in read mode or in edit mode.

4.2.1. Indices: Read Mode

The example below shows a screenshot of the respective pane (which is associated with the database node in the schema view) in read mode. For your convenience, it displays the number of indexed attributes on top. Note that the number of attributes that can have an index is limited (refer to the dbamboot command of DirX Directory).

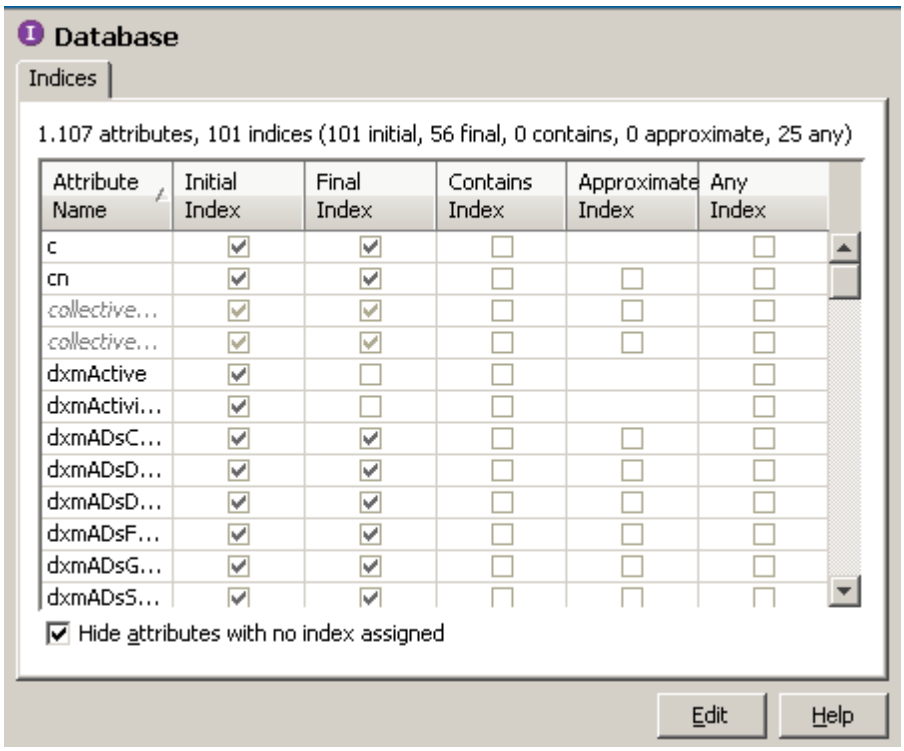


Figure 50. Database Indices in Read Mode

Notes

- Inherited attributes are shown *italic/gray*. A tooltip shows the name of the supertype.
- If you want to hand over the current settings to somebody else, you may select the entire table, copy it into the clipboard and paste it into another program, for example a spreadsheet program.

4.2.2. Indices: Edit Mode

The example below shows a screenshot of the respective pane (which is associated with the database node in the schema view) in edit mode. For your convenience, it displays the number of indexed attributes on top. Note that the number of attributes that can have an index is limited (refer to the dbamboot command of DirX Directory).

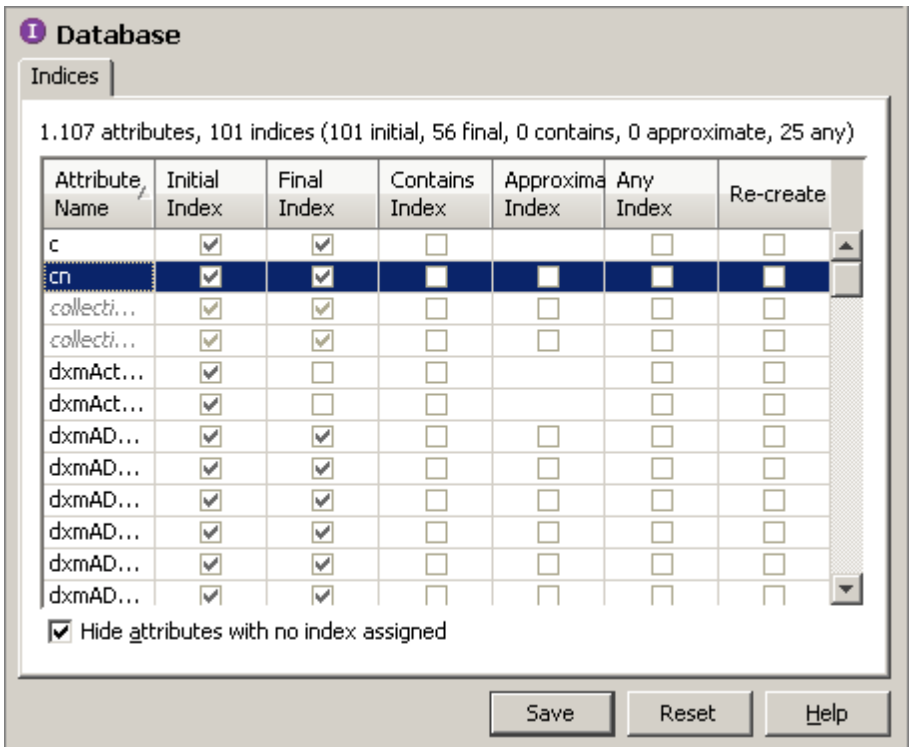


Figure 51. Database Indices in Edit Mode

Notes

- Inherited attributes are shown *italic/gray*. A tooltip shows the name of the supertype.
- Your current, yet unconfirmed settings are highlighted gray (check the affected checkbox(es) in the column titled "Recreate" and click the Save button).
- You can have the server **re**create one or more indices you feel to be corrupted.
- If you have the server create a certain type of index for an attribute that acts as supertype (aka superior) of other attributes, all those attributes will inherit that index, too. Inherited attributes are shown *italic/gray*.
- You cannot remove an index from an attribute that has a supertype/superior.
- When removing an already existing index from a supertype attribute, the indices of the inheriting attributes will not be removed implicitly.

4.2.3. Consistency of Indices

In order to check the consistency of indices, just apply the right mouse key to the database node in the schema view and select "Check consistency of attribute indices" (but be aware that while the check is running no updates are possible):

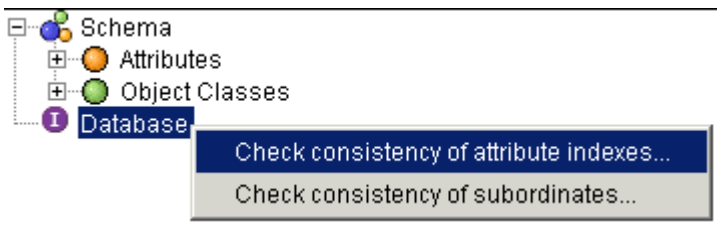


Figure 52. Schema View of Database Node

You can

- have the server check the entire tree or a specified subtree.
- have it automatically repair inconsistencies encountered during the check.

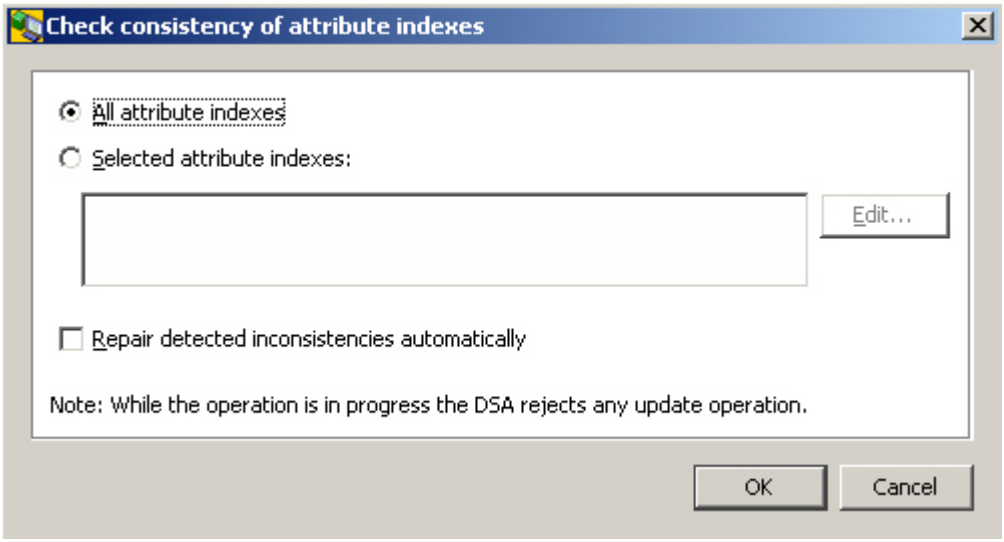


Figure 53. Check consistency of attributes indexes dialog box

5. Monitoring Information Provided by the LDAP Server

The **Monitoring** view exposes administrative information provided by the LDAP server through LDAP extended operations. (See the **dirxextop** reference page in the *DirX Directory Administration Reference* for details about LDAP extended operations.)

The source of the administrative information is either the LDAP server itself, for example LDAP process info, or the DSA the LDAP server is connected to, for example DSA exceptions. Though some values are supposed to be fairly self-explanatory, this sort of information mainly addresses maintenance staff.

The functionality provided is about

- LDAP monitoring
- DSA monitoring

It is organized into several property panes. Artificial nodes found in a tree pane just left to it allow switching between those panes.

You can have the display refreshed through:

- the F5 key
- the Refresh button in DirX Directory Manager's toolbar
- switching between the monitoring panes
- activating the auto-refresh setting that is available at the end of most monitoring panes

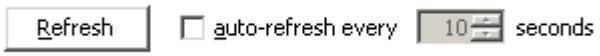


Figure 54. Auto-Refresh-Setting

Values that have changed since the previous refresh are highlighted in green color:

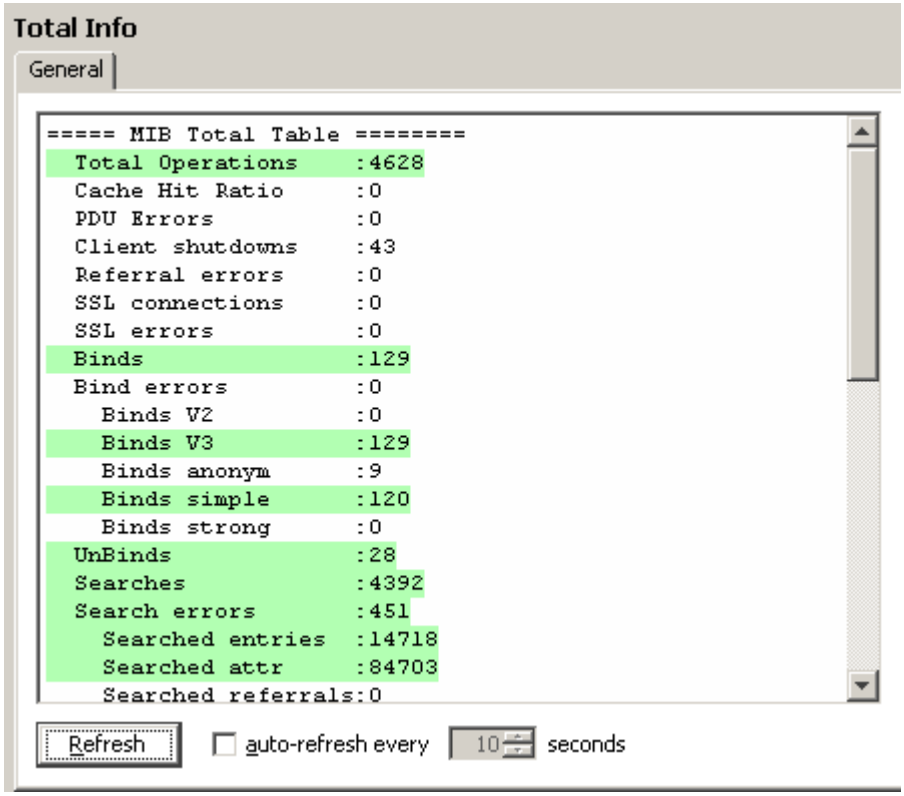


Figure 55. Monitoring View

Specify a text string in the search panel at the bottom to search the displayed text:

Figure 56. Search Panel

Press  to go to the next hit and  to go to the previous hit. All hits are highlighted in yellow. The current hit is highlighted in orange:

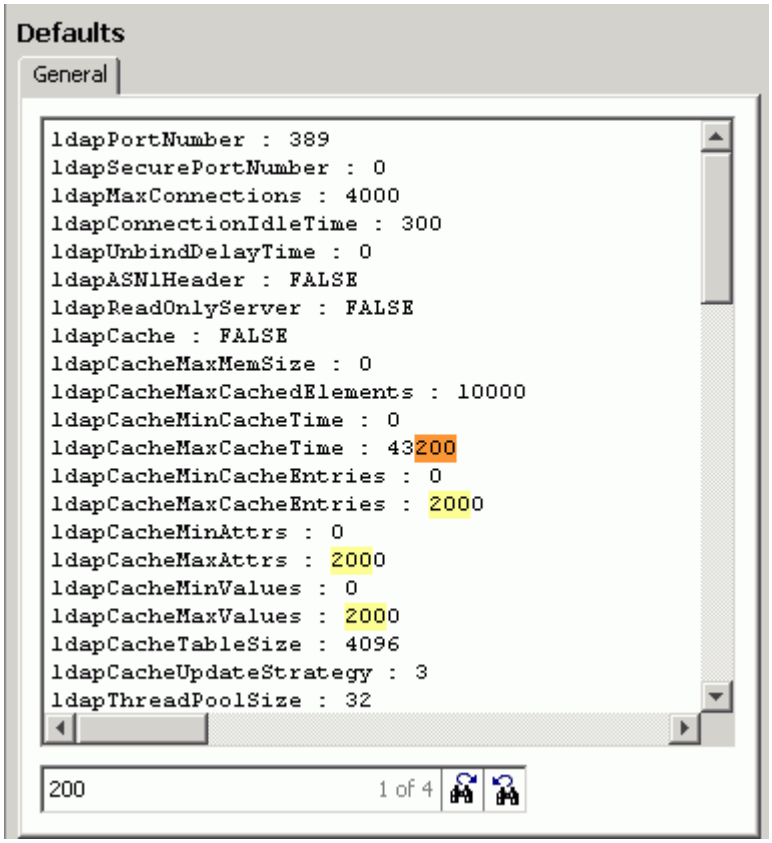


Figure 57. Monitor View with Highlighted Hits

5.1. LDAP Monitoring

This section provides information about LDAP monitoring:

- LDAP Defaults
- LDAP Extended Operations
- LDAP Configuration
- LDAP User Policies
- LDAP Proxy Server
- LDAP MIB
- LDAP Cache
- LDAP CTX Info
- SSL Cipher Names
- LDAP Audit
- LDAP Process Info
- LDAP Exceptions

5.1.1. LDAP Defaults

Lists the defaults that apply when a value is missing in any LDAP Configuration Subentries.

See also DirX Directory Manager's Configuration view.

5.1.2. LDAP Extended Operations

Displays information about LDAP extended operations. You can display the following information:

- Extop-Info:
  Lists the object identifiers (OIDs) and the LDAP names of the supported LDAP extended operations.
- Show Privileged User
  The LDAP configuration subentry provides a set of attributes for managing the accessibility to LDAP extended operations. This operation displays the values of the attributes specifying the user's privileges. These attributes are:
- ExtOp ADMIN users: users that can perform all LDAP extended operations.
- ExtOp READ users: users that can perform all LDAP extended read operations. The LDAP extended read operations are specified in the LDAP Extended Read Operations attribute.
- ExtOp EXEC users: users that can perform all LDAP extended execute operations. The LDAP extended execute operations are specified in the LDAP Extended Execute Operations attribute.
- ExtOp MONITORING users: users that can perform all LDAP extended monitoring operations. The LDAP extended monitoring operations are specified in the LDAP Extended Monitoring Operations attribute.
- Show Required Privileges
  This operation displays which privilege is required for which LDAP extended operation; that is in which attribute listed above the user's distinguished name must be contained. The information is displayed in tabular format. The columns provide the following information:
- The LDAP name of the LDAP extended operation.
- The object identifier of the LDAP extended operation.
- The privilege required for this operation: READ, EXEC, MONITORING or NO privilege. To perform the LDAP extended operation the user's distinguished name must be contained in the associated attribute. (See "Show Privileged User" above.)
- Whether the required privilege for performing the operation is either the default one or is derived from the attributes LDAP Extended Read Operations, LDAP Extended Execute Operations, LDAP Extended Monitoring Operations.

For details about LDAP extended operations, see the **dirxextop** reference page in the *DirX Directory Administration Reference*. For details about access management for LDAP extended operations, see "DirX Directory Attributes -> X.500 User Application Attributes -> Attributes for LDAP Server Configuration -> Attributes Controlling LDAP Extended Operations" in the *DirX Directory Administration Reference*.

5.1.3. LDAP Configuration

Displays configuration and statistical information regarding LDAP configuration. Furthermore, it is possible to update configuration attributes. You can:

- Display the current values for the attributes of the LDAP server's configuration subentry. A plus sign (+) next to an attribute indicates that it is available for dynamic update.
- Display the current values for the attributes of the LDAP server's SSL configuration subentry.
- Display the current values for the attributes of an LDAP server's audit configuration subentry.
- Display the list of attributes available for dynamic update.
- Activate changes to specific attributes dynamically. Using dynamic update allows changes to the LDAP server configuration to be applied without the effects of an LDAP server re-start (permanent loss of client connections to the server and temporary loss of the service itself).
- Get information about the changes made over time to the LDAP server's configuration subentry. The most recent changes appear at the top of the list.

5.1.4. LDAP User Policies

Displays configuration and statistical information regarding LDAP user policies. You can

- Display the currently active user and group policies.
- Display the status of all registered users.
- Display all rules that apply to the specified user. Specify the user's distinguished name in LDAP format or one of the following keywords:
- **all**-returns the rules that apply to all users.
- **anonymous**-returns the rules that apply to anonymous users.

5.1.5. LDAP Proxy Server

Displays and updates the configuration of a DirX Directory LDAP Proxy server. You can

- Display the current status and configuration of the LDAP Proxy server.
- Update the proxy settings from a configuration file.

5.1.6. LDAP MIB

The LDAP server's **M**anagement **I**nformation **B**ase is a subset of the specifications of the recommendation entitled Directory Server Monitoring MIB (RFC 2605) that correspond to the LDAP server and some additional information like the LDAP server's configuration.

The information exposed through the LDAP MIB is organized into:

- Static MIB

The LDAP MIB static table stores information that is usually set during initialization performed at start time of the LDAP server. It remains unchanged during lifetime of the LDAP server process.

- Total MIB
  The LDAP MIB total table stores information that is accumulated during lifetime of the LDAP server. Usually this information increases and delivers a temporary snapshot of the running LDAP server.
- Current MIB
  The LDAP MIB current table stores information that is accumulated during lifetime of the LDAP server. Usually this information increases and decreases, reflects some status information (Status), for example LDAP cache enabled (ON) and LDAP cache information is valid (valid), or provides the maximum value (MaxCounter) during LDAP server's lifetime.
- Associations MIB
  The LDAP MIB association table stores information that is dynamic concerning the content of the entire table and the values of each MIB attribute stored. It provides information concerning the number of LDAP client connections established, general information about each LDAP connection, and all operations running for each LDAP client connection.
- Environment MIB
  This table contains the current environment strings as they are known by the server.

See the appendix "LDAP MIB Tables" in the *DirX Directory Administration Reference* for details.

Additionally, you can

- Dump all information stored in the LDAP MIBs to a file. This operation writes the information in all MIB tables except the MIB association table to the file **mib**pid***.txt* where *pid* is the process ID of the LDAP server. This file is written to the same directory as the usual log files, by default to the directory *install_path**/ldap/log*.
- Display the LDAP server operation statistics of the last recent 24 hours.
- Display the LDAP server paged searched result cookie table.

5.1.7. LDAP Cache

Displays configuration and statistical information regarding the LDAP cache. Furthermore, it is possible to manage the LDAP cache. You can:

- Display configuration information and statistical information about the LDAP cache.
- Start caching of LDAP search results. All subsequent LDAP search operations query the LDAP cache first. The request is only directed to the DSA if the search result cannot be found in the LDAP cache.
- Stop caching of LDAP search results. All subsequent LDAP search operations are directed to the DSA.
- Dump configuration information, statistical information, and all saved results of the

LDAP cache.

- Clear the content of the LDAP cache. All LDAP search results are removed from the cache.

5.1.8. LDAP CTX Info

Displays a summary of the internal CTX memory consumption of the LDAP server. (CTX is the internal memory management system of DirX Directory.)

It also shows the maximum limit of usable memory of CTX together with the current size and historical high-water-mark (HWM).

5.1.9. LDAP SSL

Displays information about LDAP SSL connections. Furthermore, it is possible to manage the LDAP SSL logging and the LDAP SSL context. The following operations are available:

- Cipher Names
  Displays all cipher names that can be specified in the LDAP supported encryption strength attribute (supportedEncrytionStrength, LDAP Supported Encryption Strength). The default value of this attribute is **RSA**: all cipher suites that use the RSA algorithm are accepted. (See "LDAP Supported Encryption Strength" in the *DirX Directory Administration Reference* for details.)
- Create New Context (CRL Refresh)
  Triggers the renewal of the CRLs that the LDAP server uses to check user certificates in the context of LDAP SASL binds. The CRLs are updated with the content of the files configured in the attribute LDAP SSL CRL Filenames.
- Context-List Info
  Displays the list of created SSL contexts in use.
- SSL Logging ON
  Enables LDAP SSL logging.
- SSL Logging Status
  Displays the LDAP SSL logging status.
- SSL Logging OFF
  Disables LDAP SSL logging.
- SASL VerifyErr History
  Displays the last recent SASL certificate verification errors.

See "Attributes for LDAP Server SSL Configuration" in the *DirX Directory Administration Reference* for details.

5.1.10. LDAP Audit

Displays configuration and statistical information regarding LDAP auditing. Furthermore, it is possible to manage LDAP auditing, and evaluate LDAP audit logfiles. You can:

- Display configuration and statistical information about LDAP auditing.

- Start the recording of LDAP server audit information using the most recently read values of the LDAP audit configuration subentry. This operation has no effect on the value of the LDAP Audit On attribute of the LDAP audit configuration subentry that is evaluated at LDAP server's start-up time and when you perform a **dirxadm ldap audit -config** operation.

  The operation displays the full path name of the LDAP audit log file.
- Stop the recording of LDAP server audit information. This operation has no effect on the value of the LDAP Audit On attribute of the LDAP audit configuration subentry that is evaluated at LDAP server's start-up time and when you perform a **dirxadm ldap audit -config** operation.
- Evaluate and display the content of the current LDAP audit log file. This operation may last some time depending on the file size.
- Evaluate and display erroneous operations logged in the current LDAP audit log file. This operation may last some time depending on the file size.

5.1.11. LDAP Process Info

Provides a set of tools to analyze process internal information. Some tools for example top, pfiles or netstat are executed remotely on the server and the resulting output is returned to the caller. Other tools for example BT-Dump or IDM-Hdl-Dump provide internal information of the server and require special knowledge to interpret the output. (Please note that not all tools are available on all platforms.)

The following tools are provided:

- PStack - Displays the current thread stacks of the LDAP server process. This tool is not available on Windows systems.
- BT-Dump - Displays the DAP bind table entries.
- IDM-Hdl-Dump - Displays the IDM-Handle-Information of the LDAP server process.
- RUsage - Displays LDAP server process-specific system resource information. This tool is not available on Windows systems.
- Pfiles - Displays LDAP server process-specific file descriptor usage. This tool is not available on Windows systems.
- Top - Displays LDAP server top-process information. This tool is not available on Windows systems.
- Status - Displays LDAP server process status information. This tool is not available on Windows systems.
- Pmap - Displays process memory mapping table of the LDAP server process. This tool is not available on Windows systems.
- Netsat - Displays the TCP/IP information of all active connections.

5.1.12. LDAP Exceptions

Displays the current exception log file of the LDAP server. This operation is not available on

Windows systems.

5.1.13. Show Mapped LDAP Bind Name

In the event of a SASL-authenticated bind over SSL-protected LDAPv3 protocol with encrypted data transfer and certificate-based client authentication the LDAP server maps the certificate to a distinguished name. Use this operation to display the distinguished name.

5.2. DSA Monitoring

This section provides information about DSA monitoring:

- DSA MIBs
- DSA CTX Info
- DSA Audit
- DSA Process Info
- DSA DBAM
- DSA Exceptions
- DSA dirxadm

5.2.1. DSA MIBs

The **M**anagement **I**nformation **B**ase of the DSA and the applications is a subset of the specifications. See RFC 1565 for a definition of the Application MIB (also called the Network Services Monitoring MIB) and RFC 1567 for a definition of the DSA MIB (also called the X.500 Directory Monitoring MIB).

The information exposed through the DSA MIBs is organized into:

- NMI Show
  Displays the information in all MIB tables.
- NMI DAP Show
  Displays the information of the DSA DAP MIB table.
- NMI DSP Show
  Displays the information of the DSA DSP MIB table (information about operations for chaining.
- NMI DISP Show
  Displays the information if the DSA DISP MIB table (information about shadowing operations).
- DISP Flow Counters
  Displays information about DSA DISP update flow counters.
- 24h History DAP
  Displays the DSA operation statistics for DAP operations of the last recent 24 hours.

- 24h History DSP
  Displays the DSA operation statistics for DSP operations of the last recent 24 hours.
- 24h History DISP
  Displays the DSA operation statistics for DISP operations of the last recent 24 hours.
- DISP Monitor
  Displays detailed information about DISP status, for example information about switch or number of established SOBs.
- Paging Info
  Displays detailed bind table information about paged search results.
- DBAM Index Info
  Displays information about the database index configuration. See the **dirxadm db attrconfig** operation reference page in the *DirX Directory Administration Guide* for details about indexes.

5.2.2. DSA CTX Info

Displays a summary of the internal CTX memory consumption of the DSA. (CTX is the internal memory management system of DirX Directory.)

It also shows the maximum limit of usable memory of CTX together with the current size and historical high-water-mark (HWM).

5.2.3. DSA Audit

Displays configuration and statistical information regarding DSA auditing. Furthermore, it is possible to manage DSA auditing, and evaluate LDAP audit logfiles. You can:

- Display configuration and statistical information about DSA auditing.
- Evaluate and display the content of the current DSA audit log file. This operation may last some time depending on the file size.
- Enable DSA audit logging. Audit transactions that affect attributes and attribute values are logged. This setting is not preserved after restarting the DSA.
- Disable DSA audit logging. This setting is not preserved after restarting the DSA.

5.2.4. DSA Process Info

Provides a set of tools to analyze process internal information. Some tools for example top, pfiles or netstat are executed remotely on the server and the resulting output is returned to the caller. Other tools for example BT-Dump or IDM-Hdl-Dump provide internal information of the server and require special knowledge to interpret the output. (Please note that not all tools are available on all platforms.)

The following tools are provided:

- PStack - Displays the current thread stacks of the DSA process. This tool is not available on Windows systems.
- BT-Dump - Displays the DAP bind table entries.

- IDM-Hdl-Dump - Displays the IDM-Handle-Information of the DSA server process.
- RUsage - Displays DSA process-specific system resource information. This tool is not available on Windows systems.
- Pfiles - Displays DSA process-specific file descriptor usage. This tool is not available on Windows systems.
- Top - Displays DSA top-process information. This tool is not available on Windows systems.
- Status - Displays DSA process status information. This tool is not available on Windows systems.
- Pmap - Displays process memory mapping table of the DSA process. This tool is not available on Windows systems.
- IOstat - Displays information about input and output traffic of the DSA process. This tool is not available on Windows systems.
- Netsat - Displays the TCP/IP information of all active connections.

5.2.5. DSA DBAM

The DBAM MIB information provides statistical information for several DBAM subsystems. DirX Directory Manager provides the following operations to manage the DBAM MIB and to display the information contained:

- DBAM Mib start
  Enables the DSA DBAM MIB or displays the date when the DSA DBAM has been enabled.
- DBAM Mib show
  Displays the content of the DBAM MIB tables. See the appendix "DBAM MIB Tables" in the *DirX Directory Administration Reference* for details on the content of the DBAM MIB tables.
- DBAM Mib stop
  Disables the DSA DBAM MIB.
- DBAM DevInfo
  Displays information about the capacity of the logical and the attribute index specific devices of the database. See the **dbamdevinfo** reference page in the *DirX Directory Administration Reference* for details.
- DBAM Config
  Displays a detailed list of database profile properties. See the **dbamconfig** reference page in the *DirX Directory Administration Reference* for details.
- DBAM Preload Status
  Displays the status of the DBAM buffer cache preloader.
- DBAM Preload On
  Starts the DBAM buffer cache preloader.
- DBAM Preload Off
  Stops the DBAM buffer cache preloader.

5.2.6. DSA Exceptions

Displays the current exception log file of the DSA. This operation is not available on Windows systems.

5.2.7. DSA dirxadm (DirX Directory Server V8.10 or higher only)

These operations are supported only by DirX Directory servers installed on Linux systems.

Performs a **dirxadm** operation in a DSA that can only be accessed via the LDAP protocol. The **dirxadm** operation is sent to this DSA via the LDAP extended operation **dsa_dirxadm_cmd**. The transfer of the **dirxadm** command is performed via the LDAP extended operation, while the execution of the command is performed via the RPC protocol between **dirxadm** and the DSA.

The **dsa_dirxadm_cmd** operation requires the **Execute** permission; that is, the user's distinguished name must be contained in the **LDAP Extended Operations Execute Users** or **LDAP Extended Operations Execute Groups** attribute of the LDAP server (or in the list/group of ExtOp administrators).

The **dsa_dirxadm_cmd** operation uses the bind operation specified in the environment variable **DIRX_DSA_EXTOP_ADM_BIND** to perform the **dirxadm bind** operation to the DSA.

The result of the **dirxadm** operation is displayed in the result window.

See the **dsa_dirxadm_cmd** reference page in the *DirX Directory LDAP Extended Operations* for details.

The following **dirxadm** operations are provided as predefined shortcuts. If the cmd node is selected, an input window allows you to enter any legal **dirxadm** command. Any command you can type into **dirxadm** can be entered in this cmd line.

- show DirXDBVersion
  Displays the attribute values of the DirXDBVersion subentry **CN=DirXDBVersionSubentry**. (See "DirX Directory In Sync", "DirX Directory Recent DN", "DirX Directory Recent MSN", "DirX Directory Recent MSN Time Stamp" and "DirX Directory Recent Operation" in "DirX Directory Attributes" -> "X.500 Directory Operational Attributes" in the *DirX Directory Syntaxes and Attributes*, and "Creating a Synchronous Shadow DSA" -> "Monitoring Data Synchronicity Status" -> "Using the DirXDBVersion Subentry" in the *DirX Directory Administration Guide* for details.)
- show GlobalPasswordPolicy
  Displays the attribute values of the global password policy **CN=GlobalPasswordPolicy**. (See "DirX Directory Attributes" -> "X.500 User Application Attributes" -> "Attributes of the Password Policy Subentry" in the *DirX Directory Syntaxes and Attributes* and "Creating a Shadow DSA" -> "Password Policies in a Shadow Configuration" in the *DirX Directory Administration Guide* for details.)
- show LdapROOT
  Displays the attribute values of the LDAP root subentry **CN=ldapRoot**. (See "DirX Directory Attributes" -> "X.500 User Application Attributes" -> "Attributes of the LDAP Root Subentry" in the *DirX Directory Syntaxes and Attributes* and "Setting up the DirX

Directory Service" -> "Setting up the LDAP Server" -> "Creating the LDAP Server Subentries" in the *DirX Directory Administration Guide* for details.)

- show DB-Config
  Displays the attribute values of the schema subentry **CN=Schema**. (See "DirX Directory Attributes" -> "X.500 User Application Attributes" -> "Attributes of the Schema Subentry" in the *DirX Directory Syntaxes and Attributes* and "Setting up the DirX Directory Service" -> "Setting up the LDAP Server" -> "Creating the LDAP Server Subentries" in the *DirX Directory Administration Guide* for details.)
- nmi show
  Displays the information in all MIB Tables. (See "DSA MIBs" for details.)
- show Audit Status
  Displays configuration and statistical information about DSA auditing.
- show RootDSE
  Displays the attribute values of the root DSE **/**. The root DSE is created automatically during installation.
- show CPs
  Searches and displays DNs whose DSEType indicates a context prefix (CP).
- show SUBRs
  Searches and displays DNs whose DSEType indicates a subordinate reference.
- show all SOBs
  Displays all shadowing agreements.
- show all LOBs
  Displays all LDIF agreements.
- show DIT info
  Displays details about the root DSE and context prefixes.

All these commands read data from the DirX Directory database. They do not change any data.

Additionally, the following specific operation is provided:

- cmd
  Provides an input line at the top where you can specify any legal **dirxadm** command. The **dirxadm** command must be given as one line of input (do not use wrap-around characters like ****). Double quotes should not be needed. We do not recommend using this option to execute comprehensive commands like **sob/lob create**.

6. Script Manager

The script manager makes it easy to set up scripts and run **dirxcp** (DirX Directory Command Line Program) and **dirxadm** (DirX Directory Administration Program) commands. Moreover, as sort of side effect and not really related to scripts, you can have the script manager decode and view audit log files based on the **dirxauddecode** command.

Notes:

- The commands quoted above can only be executed if they are available *locally* on your machine.
- (Linux) In order to be able to run **dirxcp**/**dirxadm** scripts, DirX Directory Manager must be installed in the same account as DirX Directory.

The script manager is available in a special view that is organized into three panes:

- Script Explorer
  The script explorer allows you to locate and manage the folders that contain the scripts you want to view/edit/create.
- Script Editor
  The script editor ***displays*** the scripts you have double-clicked in the script explorer. The script editor can manage more than one script at a time. Click the tabs at the top of the dialog to switch between scripts.
  The script editor allows you to ***edit*** scripts and provides syntax highlighting to help you keep track of your scripts.
  You also can ***run*** scripts provided the suffix is **.cp** (causes **dirxcp** to be executed) or **.adm** (causes **dirxadm** to be executed). You can cancel a running script at any time.
  Also, the script editor ***displays*** **audit log** files you have right-clicked in the script explorer (function "Decode DirX Audit Log...").
- Structure
  The structure pane displays - where applicable - the "source" statements and the procedures contained in the script that is currently displayed in the script editor. Click a source entry or a procedure entry in the structure pane to move the cursor to the corresponding entry in the script editor.

Here is an example of the script manager view (the current script contains both, source statements and procedures):

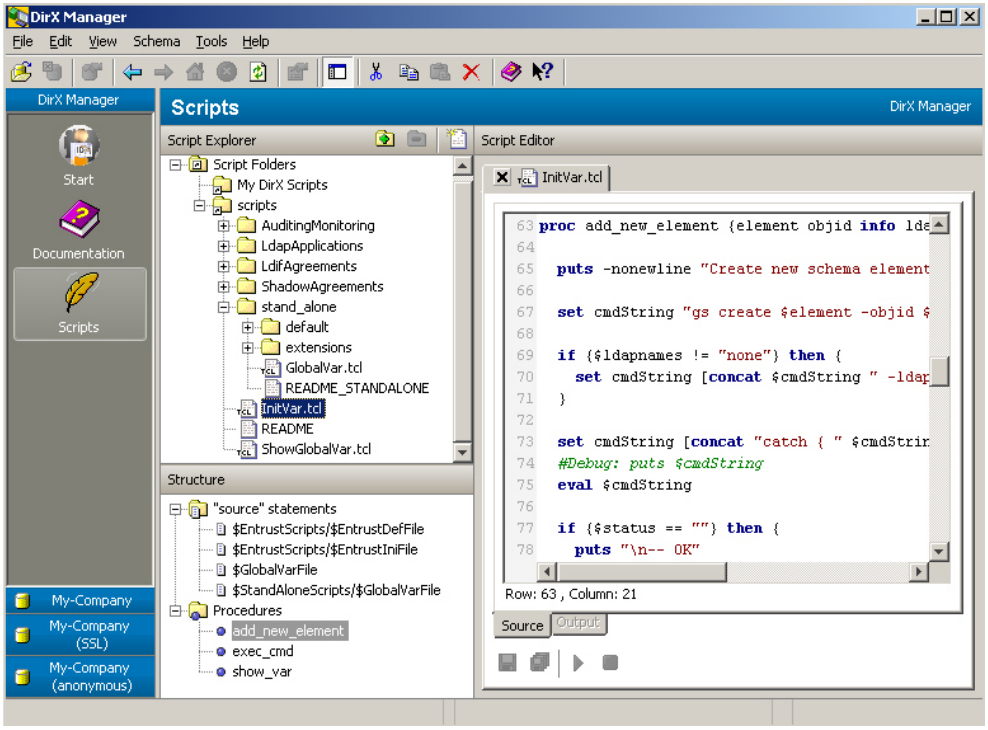


Figure 58. Script Manager View

6.1. Script Explorer

The script explorer displays a root entry called "Script Folders". The first level underneath Script Folders consists of links to "real" file system folders that contain your scripts.

Here is an example of the script explorer display:

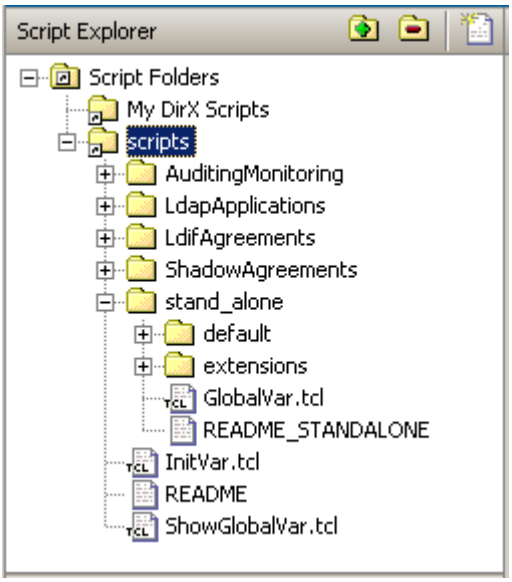


Figure 59. Script Explorer

"Link" Folders

- Link folders are like links or shortcuts (look at a link folder's tool tip to see where it points to). This is why this kind of folder's icon looks like this: (as opposed to a real file system

folder's icon, which is:).

- Initially, there is one link folder present (called "My DirX Scripts").
- To add a link folder whose real counterpart exists in the file system, right-click the topmost node (Scripts Folders), click "Add Folder" and select a folder in the file selection box that pops up.
- To create a new link folder with no real counterpart, right-click the topmost node (Scripts Folders) and click "Add Folder". In the file selection box that pops up, click the button whose tool tip is "Create New Folder" (or the like).
- To remove a link folder, right-click the folder you want to remove and click "Remove". This action deletes only the link that the corresponding folder represents.
- To rename a link folder, right-click it and click "Rename".
- You cannot delete the "real" counterpart of a "linked" folder with this application.
- You can cut/copy/paste and drag/drop script files.

Managing Scripts

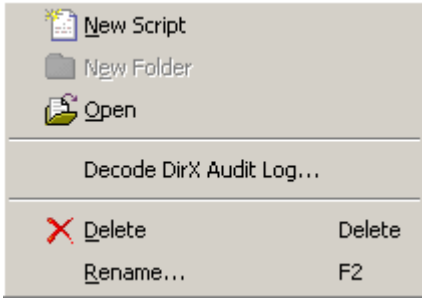


Figure 60. Managing Scripts Menu

- To create a new script, right-click a link folder, any other folder or a script, then click "New Script".
- To organize your scripts, you can create intermediate folders by right-clicking any folder but "Script folders" and then clicking "New Folder".
- To view a script, double-click the entry or right-click it, then click "Open". The script is displayed in the Script Editor pane.
- To view a DirX Directory audit log, right-click "Decode DirX Audit Log..." and fill out the form that appears. Primarily, this form allows you to reduce the size of text you are going to face by specifying filters (note that the size of the text displayable in the editor is limited, whereas the size of the original audit log file usually doesn't matter).

  When creating csv output, only a limited number of parameters such as UniqueID, StartTime, Duration and OpType will be reported (equals the "-Z" command line switch in dirxauddecode)

  "Other options" allows you to specify options in a command line like style. This field is primarily intended for use by vendor staff.

  Here is an example:

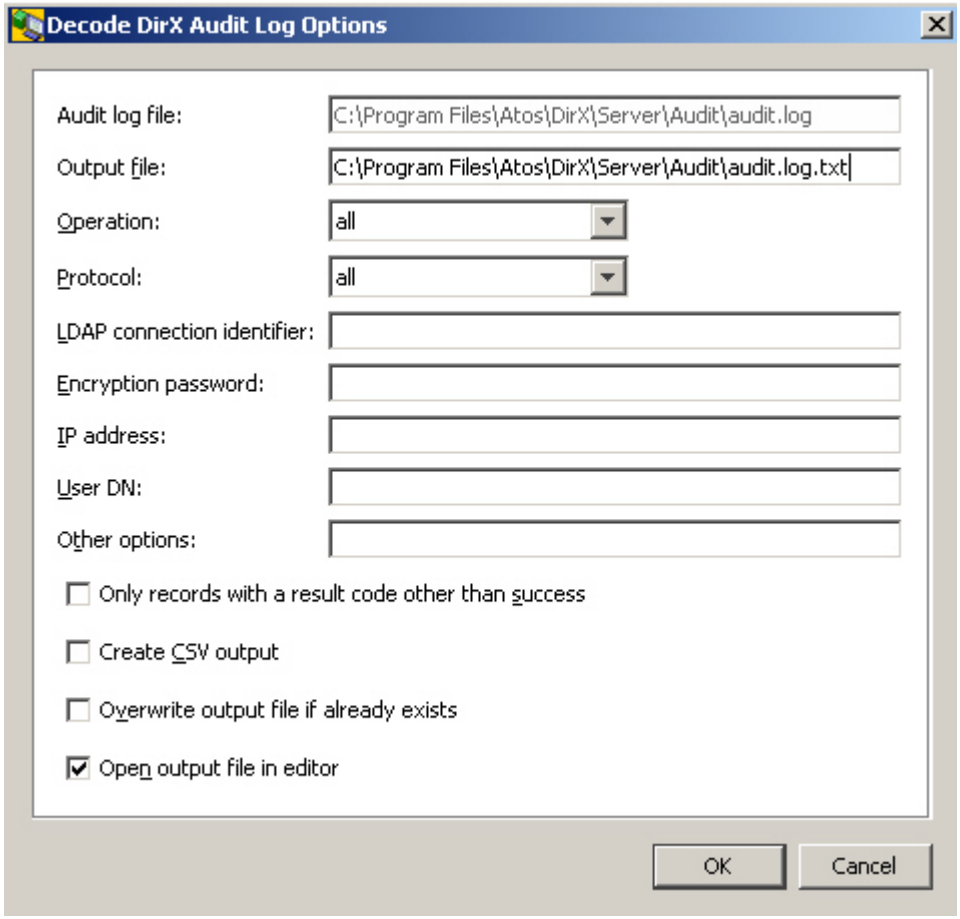


Figure 61. Decode DirX Audit Log Options Dialog Box

- To delete any file or folder except a link folder, right-click the file or folder and click "Delete". Deleting a folder deletes all of its contents.
- To rename a folder or a file, right-click it and click "Rename".

6.2. Script Editor

You can use the script editor to display, edit, and run scripts and display audit log files. Here is an example dialog:

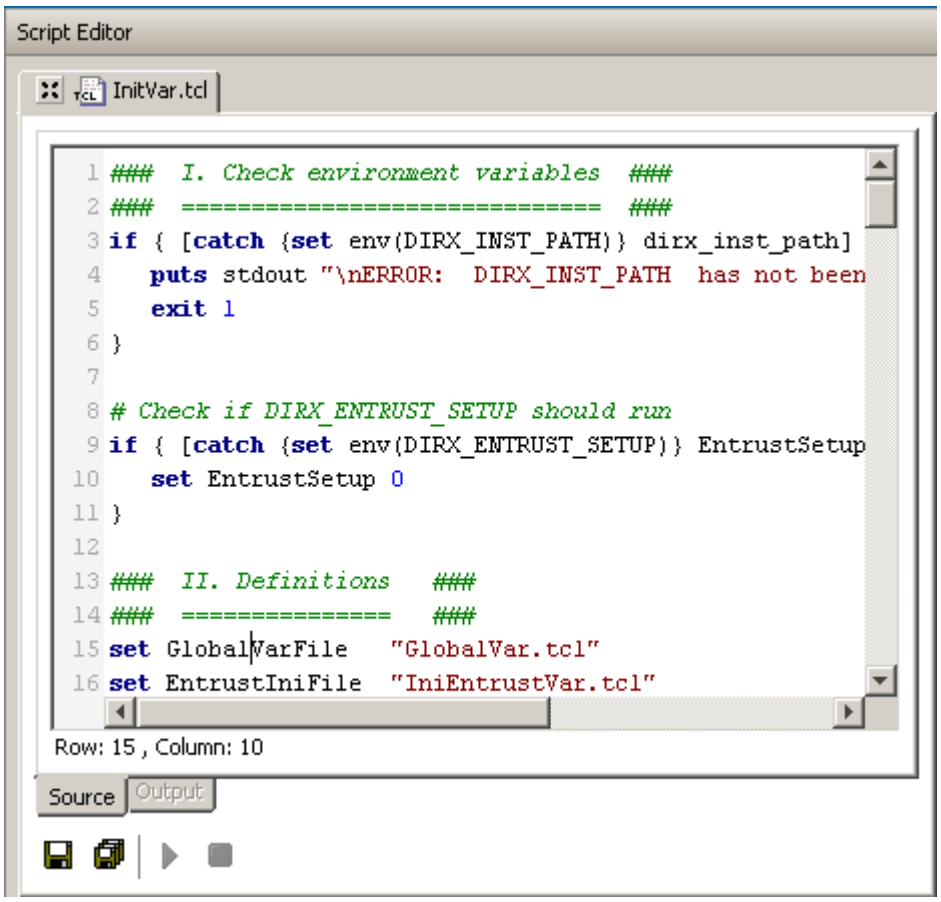


Figure 62. Script Editor

You can use the script editor on any text file. The script editor supports syntax highlighting for the file types .**tcl**,.**cp**,.**adm** (TCL Syntax: http://www.tcl.tk); .**xml** ;.**js** (JavaScript); .**java**; and .**ini:**

- Lines that begin with "#" are highlighted in green.
- Recognized keywords are highlighted in blue.
- Text enclosed in quotation marks is highlighted in red.
- Clicking a bracket, brace or parenthesis highlights it and its counterpart in pink.
- Click the right mouse button for numerous convenience functions. Here is an example:

Figure 63. Context-sensitive Menu of the Script Editor

Most functions are fairly straightforward.

- Go to insertion point
  Positions to the current insertion point of the mouse cursor. Note that you might lose track of this point when scrolling the script editor window. You will not notice any effect unless the current insertion point is outside the area that is currently visible.
- Find other block end
  When you click a bracket, brace or parenthesis whose counterpart is not in the currently visible area, this function positions the cursor on the counterpart.
- The Insert TCL command functions shown in the example menu dialog are TCL-related.

To save the script that is currently visible, click 

To save all scripts that are currently open, click 

To run a script, click  or press F12 (the script suffix must be ***.cp** or ***.adm**, otherwise this button is disabled)
Running a script implies saving it.

To stop a script that is currently running, click  .

To close a script file currently opened in the script editor click  in the respective tab or right-click the respective tab and select "Close"

To close all script files currently opened in the script editor at once right-click the respective tab and select "Close All"

Click the Output tab to see the output the actual script is creating or - if already finished - has created.

Press F11 to toggle between the *Source* and *Output* tabs.

6.3. Script Structure

The structure pane shows the "source" statements and procedures in the *.cp or *.adm script that is currently displayed in the Script Editor pane Click an entry in the structure pane to position the script editor to that entry.

The following sample structure pane shows four source statements and three procedures:

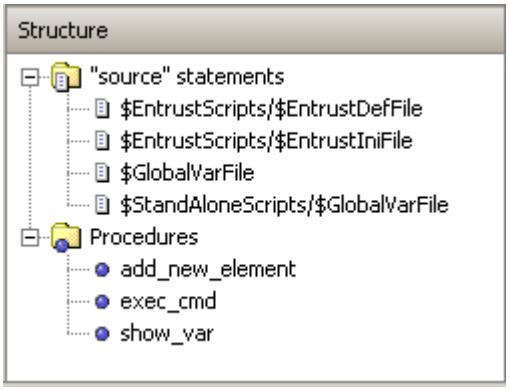


Figure 64. Structure Pane

7. Core Component

This application's functionality is primarily furnished by one or more plug-ins that are based on a core component.

The core component is composed of two subcomponents:

- A component called **LDAP** that provides a Java GUI for plain LDAP functionality.
- A component called **Framework** that provides a framework for a customizable and *extensible* Java GUI. Its main focus is on managing hierarchically-structured nodes, particularly LDAP nodes.

  "*Extensible*" refers to the plug-ins. Although plug-ins are in no way restricted to this feature, they typically deal with advanced LDAP functionality like schema management or with functionality for rather sophisticated data models that are based on LDAP (like privilege-based access management or synchronization workflows).

  Refer to the About dialog in the Help menu to see what plug-ins you have installed.

This chapter provides information about the core component, including:

- Information about LDAP in general.
  Note that this section is not intended to supersede the comprehensive literature that is available on this topic.
- The basic patterns the core component makes available to the plug-ins and uses itself for the LDAP functionality it provides.
- Some hints on possible pitfalls.

7.1. Using LDAP

(a brief introduction)

LDAP specifies a "directory access protocol". A directory accessible through LDAP stores information about users (for example, their names, their email addresses, their phone numbers, the department they belong to, and so on) and groups of users in many cases, but can also be used to store any other kind of information you can think of. There are two main reasons that explain why you frequently encounter LDAP directories in the Internet and extranet environments:

- LDAP means that the information in the directory can be accessed and managed through a standardized method specified in a series of internet standards, e.g. http://www.ietf.org/rfc/rfc2251.txt.
- LDAP directories are optimized for fast read access by numerous concurrent applications in large distributed environments.

LDAP stands for **L**ightweight **D**irectory **A**ccess **P**rotocol. As the name suggests, the original intention was to provide a simplified **D**irectory **A**ccess **P**rotocol (**DAP**). **DAP** is the subject of another series of international standards (known as X.500 (http://www.itu.int/ITU-T), which

is also - with rather insignificant deviations - available as an ISO standard (http://www.iso.org; ISO/IEC 9594). Over the course of time, however, LDAP has evolved from a simple protocol to one that favors more X.500-like capabilities.

An LDAP directory contains **entries**. An **entry** is uniquely identified by its "**Distinguished Name**" (**DN**). DNs are structured hierarchically. Here are some examples of DNs:

- c=BY (c normally stands for country, BY is a country name complying with ISO 3166 (http://www.iso.org))
- o=composers, c=BY (o normally stands for organization)
- cn=Igor Stravinsky, o=composers, c=BY (cn normally stands for common name)
- cn=Marc Chagall, ou=modern painters, o=painters, c=BY (ou normally stands for organizational unit))
- cn=Distribution List Sales, l=Thessaloniki (l normally stands for locality)
- cn=Fedor Dostoevsky+fathersname=Mikhailovich, o=writers, c=BY
- cn=Alexander Green, o=writers, c=BY

Here is a sample showing how as this application might present those entries in its so-called tree pane:

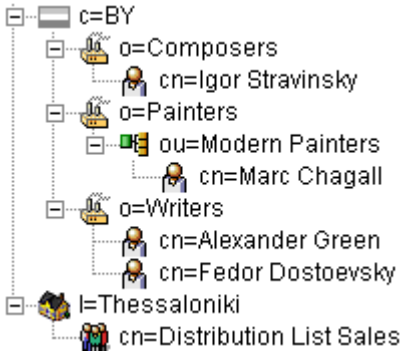


Figure 65. Tree Pane

Entries are composed of a DN and a number of attributes. For example, an entry with DN *cn=Alexander Green, o=writers, c=BY* might have the attributes:

- objectclass:
- top
- person
- organizationalPerson
- inetOrgPerson
- commonName:
- Alexander Green
- Aliaksander Gryneusky
- sn: Green
- givenName: Alexander

- telephoneNumber: +375 99 99999-99
- mail: alexander.green@writers.by

DNs are composed of **relative DN**s (**RDN**s). For example, *cn=Alexander Green, o=writers, c=BY* is composed of three RDNs. RDNs must be derived from attributes. It is possible to compose an RDN of multiple attributes joined by “+” as in *cn=Fedor Dostoevsky+fathersname=Mikhailovich,o=writers,c=BY*. RDNs like this are called **multiple naming attributes** (or **multiple attribute value assertion**).

Since RDNs must be derived from attributes, the *last RDN* (*Alexander Green* in the previous example) is always available as an attribute of a given entry, too. As opposed to the last RDN however, the underlying attribute may be recurring. In the example used here, it has two values: "Alexander Green" and "Aliaksandr Gryneusky". One of them must be identical to the last RDN. "Alias" entries represent an exception to this rule: An Alias has only the two attributes objectClass and aliasedObjectName, but can use for its DN almost any sort of attribute but attributes with DN syntax.

The hierarchy starts from "Root" (commonly marked "/"). If you consult the previous examples, one level below "/", you will find "*BY*" and "*Thessaloniki*", one level below "*BY*" there is "*composers*", "*painters*" and "*writers*" and so on.

LDAP servers have at least one **root node** (also called a **suffix** or a **context prefix**), often unequal to "/" (*o=painters, c=BY* for example). LDAP servers only store their **naming context**; that is, their own root node and the entries that are subordinate to their root node. LDAP servers may be able to be configured to contact other servers automatically (**chaining**) or to return **referrals** that point to other servers when they are handling requests outside the scope of their suffix. This application can be configured to follow referrals (to some extent) or ignore them.

Attributes may be **recurring**, also called **multi-valued** (like *object class* and *cn* in the previous example) or **single-valued**. Attributes have an **attribute syntax**. Here are some examples: Boolean, Country String, Certificate, DN, Directory String, Generalized Time, IA5, Integer, Jpeg, Numeric String, Postal Address, Telephone Number - and there are many more.

Attributes can be classified into

- **User Attributes**
  User attributes are all attributes except for operational attributes.
- **Operational Attributes**
  Operational attributes are predefined attributes that are used for administrative purposes. For example, *creatorsName* is an operational attribute that stores the DN of the user who added the entry.
- **Collective Attributes**
  Collective attributes are attributes that are common to multiple entries. These attributes are stored only once.

As specified by LDAP, entries must have a special recurring attribute that stores **object classes** (see the previous example). Object classes typically define the attributes that an object must have and may have. However, some LDAP servers allow you to switch off the

"schema check" function. In this case, the object class is not much more than another possible search filter. Some LDAP servers do not use the object class recurring attribute at all.

There are several types of object classes:

- **Abstract Object Classes**
  Abstract object classes are used only to derive other object classes. *Top* is an abstract object class from which every structural object class is directly or indirectly derived (auxiliary object classes are usually derived from *Top*, too, but this is not mandatory). In many real world schemata, it is the only abstract object class you will encounter.
- **Structural Object Classes**
  Each entry must have at least one structural object class. If an object class happens to be derived from of another object class, the affected entry has both object classes and inherits all attribute types from the parent object class. However, not counting "parent classes", each entry has one and only one structural object class.
- **Auxiliary Object Classes**
  In addition to structural object classes, entries may have one or more auxiliary object classes. As opposed to structural object classes, auxiliary object classes can be added to and removed from an entry at any time. Auxiliary attributes are associated with a (possibly empty) set of attribute types. Auxiliary classes provide a convenient means to add attribute types (the ones that are associated with that auxiliary object class) dynamically to entries that are already defined by a structural object class. So, auxiliary object classes allow you to dynamically extend the permissible attribute set of entries beyond the one defined by their structural object classes. If the set of associated attribute types is empty, the auxiliary object class is an object class that at the same time is sort of an ordinary attribute. It may however serve as a filter for access control definitions in case the access control provided by the server follows the X.500 specification.

In the previous example, you can find the object classes *top*, *person*, *organizationalPerson* (usually derived from *person*), *inetOrgPerson* (usually either derived from *organizationalPerson* or an auxiliary object class).

LDAP provides several ways of searching:

- **Subtree search** (an "all level" search)
  Searches within the whole subtree rooted at the current entry.
  If you refer to the previous example, a subtree search below *c=BY* with the search filter *cn contains chag* returns the entry *cn=Marc Chagall, ou=modern painters, o=painters, c=BY* as the search result.
- **One level search**
  Searches among the "children" of the current entry (disregarding "grand children" and their "descendants"). If you refer to the previous example, a one-level search below *c=BY* with the search filter *cn contains chag* returns "successful" with the search result being empty.
- **Base object search** (a "zero level" search)
  A base object search is not really a search. It can be used to read the current entry or to check for its existence.

Searches, particularly subtree searches, allow you to pass a more or less complex **search filter** in order to restrict the number of entries returned by the server. You may also pass a **size limit** and/or a **time limit** to save the server's resources and to avoid slow reaction times and too many entries in the search result. Note, however, that LDAP servers usually use their own limits; LDAP clients cannot exceed these limits.

What is returned as search result depends upon the conditions under which the software considers a value presented by a user to correctly match a value stored in the directory. This is where **matching rules** come into play. Matching rules are assigned to attribute types and must be compatible with the corresponding attribute syntax. A matching rule specifies how attribute values are to be matched for equality, ordering, or substring comparison. You cannot use attributes with no matching rule assigned (like *fax* in the examples to follow) in search filters.

Here are some examples of matching rules:

Table 2. Table : Matching Rules

Attribute		**Matches for**		
Type	Syntax	Equality	Substring	Ordering
department	DirectoryString	Case-Ignore-Match	Case-Ignore-Substring-Match	—
createTimeStamp	GeneralizedTime	Generalized-Time-Match	—	Generalized-Time-Ordering-Match
jpegPhoto	OctetString	Octet-String-Match	—	—
headCount	Integer	Integer-Match	—	Integer-Ordering-Match
userCertificate	Certificate	Certificate-Exact-Match	—	—
facsimileTelephoneNumber	facsimileTelephoneNumber	—	—	—

There are many more possible matching rules.

What is returned in a search result also depends on the attributes that are requested to be returned:

- Requesting **All attributes** means all user attributes of all entries matching the search filter are requested to be returned
- Requesting **All operational attributes** means all operational attributes are requested to

be returned. Note that this option is not necessarily recognized by all LDAP servers

- It is also possible to request exactly those attributes that are listed in a list of **requested attributes** passed with the search request. This is normally the most resource-saving way, since it avoids transferring unneeded data

Moreover, missing access rights (see below) may cause a search result to be incomplete.

At last, what is returned in a search result depends on the data that can be accessed: As mentioned above, a server does not necessarily store all the data itself. If not, in order to still be able to return complete search results, it relies on configuration provisions such as "Chaining" (i.e. the server "chains" to other servers, to get additional information) or "Referrals" (i.e. the server does not provide the complete search result itself, but leaves it to the client to complete it by providing referrals to it; the client may or may not "follow referrals").

Because this application is written in Java, the characters of string attributes assumed by this application are based on Unicode (http://www.unicode.org; Java specifies the character encoding form "UTF-16", which is one of a number of possible Unicode encoding forms); Unicode includes Latin, Cyrillic, Middle East, Far East characters and much more. Note however that some - not necessarily significant - issues still remain open:

- Your computer may be missing the installation of respective fonts and therefore be unable to render the corresponding characters. E.g. to have characters like these 香港劇魚涌 displayed on an operating system installed for English usually requires some additional provisions.
- Unicode is still evolving and - since new characters are continuously being invented - may never stop evolving. The Unicode support of the Java version you are using may be more or less behind the latest version of Unicode, e.g.
  Java 1,4 supports Unicode 3.0
  Java 1.5 supports Unicode 4.0
- In LDAP, Unicode support has been "officially" introduced with LDAP protocol version 3, i.e. running LDAP protocol version 2 might cause problems with certain characters. Again, the Unicode version the LDAP server supports is not necessarily the latest one. As opposed to Java, LDAP specifies the character encoding form "UTF-8", which is automatically converted to UTF-16 and back without loss.

Other - occasionally used - search options include:

- "Types only"
  You can have the search return only attribute types rather than types and values (which is the default).
- Dereference Aliases
  Indicates how alias objects are to be handled in searching. The semantics of the possible values of this field are:
- neverDerefAliases (default)
  Do not dereference aliases.
- derefInSearching
  Dereference aliases in the search result.

- derefFindingBaseObj
  The search base is de-referenced if it happens to be an alias.
- derefAlways
  Dereference aliases both in the search result and in the search base.

Other requests specified in LDAP include:

- Add entry
- Modify entry
- Rename entry
- Delete entry

LDAP controls provide a mechanism for extending an LDAP operation. A control is a way to specify additional information as part of an LDAP request and an LDAP response.

A control specifies the following information:

- A unique object identifier (OID), as defined by the creator of this control
  The OID identifies the control. If you plan to use a control, you need to make sure that the server supports the control. Servers usually list all or some of the controls they support in the supportedControl attribute in the root DSE. You can acquire the root DSE by doing a search where the scope is "Base", the base DN is "" (empty string) and the search filter is (objectclass=*). Alternatively, you can have tools like DirX Directory Manager display the root DSE including its supportedControl attribute.
- An indication of whether or not the control is critical to the operation
- Optional data related to the control (for example, for the server-side sorting control, the attribute to be used for sorting search results)

If an LDAP request contains a control, the server may respond in one of the following ways:

- If the server supports this control and if the control is appropriate to the operation, the server should make use of the control when performing the operation
- If the server does not support the control type or if the control is not appropriate, the server is supposed to do one of the following:
- If the control is marked as critical to the operation, the server should not perform the operation and should instead return the result code "Unavailable critical extension".
- If the control is marked as not critical to the operation, the server should ignore the control and should perform the operation.

Servers can react to some sorts of controls by sending controls back to clients. Example: "Server Side Sorting" is provoked thru the "Server Side Sorting *Request* Control", which specifies details such as the attribute type to use for sorting. The server is to send back a "Server Side Sorting *Response* Control", which tells whether the result is sorted or not - and if not why not.

This application inspects the LDAP Root subentry to find out what controls the server supports and implicitly makes use of all or some of the available controls. Details are

available in the section "LDAP Root Subentry" (provided the DirX Directory Manager plug-in is installed).

The LDAP server may refuse to handle a request because of missing **access rights**. Access rights depend on the user's authentication. A user authenticates himself through a **bind** (called login in this application). The default bind "**Anonymous**" is implied. The LDAP server usually configures an anonymous bind to permit read access for anyone to a well-defined subset of the data stored in the server's database. More comprehensive access rights typically require a DN and a password to be passed (**basic authentication**). Particularly for access from the Internet or an extranet, it might be a good idea to use **secure authentication** by combining basic authentication with **SSL** (secure socket layer) or its designated successor **TLS** (transport layer security). Note that the LDAP server does not necessarily reveal the fact that it did not handle a particular request due to missing access rights. Instead, it might return an error code like "not found", for instance.

The hierarchical structure of the DN suggests that you consider the directory to be a tree. Also, the tree-like collection of all DNs that comprise a particular, possibly distributed directory is called the **Directory Information Tree** (**DIT**). This is justified to some extent; however, **alias entries** allow the tree structure to be circumvented.

7.1.1. Available LDAP Functions

Unless disabled, the core component typically makes its LDAP functionality available in one or more view groups which you can access through:

- The menu
- The toolbar
- The right mouse button

These functions include:

- Browsing the LDAP directory tree
- Creating new LDAP directory entries
- Deleting LDAP directory entries
- Exporting/Importing LDAP directory entries
- Displaying administrative information on the server (the so-called LDAP root DSE, where "DSE" stands for directory-specific entry)
- Logging in/out to/from an LDAP directory
- Moving LDAP directory entries to a different parent
- Copying LDAP directory entries
- Showing and modifying properties of LDAP directory entries
  Note that some rather uncommon properties (like MHS OR address) are not or not fully supported.
- Changing passwords (your own password as well as somebody else's password)
- Renaming LDAP directory entries

- Searching the LDAP directory by specifying simple or complex search conditions (see the Search pane and search dialog topics)
- Abandoning an in-process, but uncompleted search operation.

In order to be able to access an LDAP directory, you must provide some information like the address and port number (default: 389) of the LDAP directory server. This information may already be pre-configured, but you may also be able to edit this information by right-clicking on the top node of a tree and selecting "Server" (alternatively, you may find this functionality in the tool bar/menu bar).

Additional administrative functionality (e.g. schema management) may be available in addition to the functionality the core component provides, depending on what plug-ins are installed.

The core component's LDAP access functionality can be **generic** or **customized**.

- Generic LDAP access means that the core component has no built-in knowledge about the LDAP directories to be accessed. If the core component is unable to retrieve the schema from an LDAP server, the functionality that is based on the knowledge of schema information - like objects classes and attributes - is unavailable unless the missing information is supplied in the application's configuration files. In particular, searching may become unusable.
- Customized LDAP access means that the core component has pre-configured information about the objects and attributes of one or more LDAP directories to be accessed.

LDAP directories are organized in a tree-like manner. This organization may or may not be visible in a particular view.

7.1.2. SSL/TLS

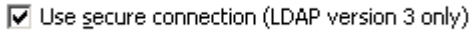
For in-depth information on this topic please refer to https://docs.oracle.com/javase/8/docs/technotes/guides/security/ (search for appropriate key words, e.g. "keytool").

SSL stands for **S**ecure **S**ocket **L**ayer. It is a common technique to have TCP/IP network communication encrypted based on "trusted" certificates. **TLS** stands for **T**ransport **L**ayer **S**ecurity; it is a not overly significant, evolutionary advancement over SSL and has de facto replaced SSL. What is used in a particular connection (TLS or SSL) is the result of a negotiation between client and server. Possibly more important, the key length is negotiated, too.

There are two security stages:

- Stage 1 ("certificate-based server authentication"): if you want to make sure the server you are talking to is actually the one you think it is. This application supports Stage 1. Stage 1 implies encryption
- Stage 2 ("certificate-based client authentication"): if the server asks you to authenticate yourself through a certificate.

In order to get SSL/TLS working, you must

- Ensure the server supports it and is configured appropriately
- Ensure that a suitable certificate is available in your local "keystore". By "suitable", we mean that the certificate transmitted by the LDAP server must be trusted by at least one of the certificates stored in the keystore that is assigned in the function "Options" (menu "Tool"). Note that
- This application ships with a keystore file called cacerts that contains a *demonstration* certificate that is aligned with the demonstration certificate DirX Directory brings along. With this demonstration certificate you are able to run SSL/TLS out-of-the-box in conjunction with DirX Directory by way of trial.
- The Java Runtime Environment, too, ships with a keystore (also a file called "cacerts") containing a number of root Certificate Authority (CA) certificates in its keystore.
- You can assign only one keystore for all secure connections within this application, i.e. it is not possible to assign one keystore for server A and another one for server B. Choosing a different keystore requires this application to be exited and started again to get the new keystore effective.
- The function "Options" allows for managing keystores. Alternatively, you can use the **keytool** command that ships with the Java Runtime Environment. If your certificate is stored in a file called, for example, **myCA.der** and you want to know it from other certificates stored in the same keystore by the name **myName** and if your keystore resides in, for example, **C:\jre\lib\security\cacerts**, you use it like this:*
  keytool -import -file myCA.der -keystore C:\jre\lib\security\cacerts -alias myName*
  You will be asked for a password. The default password for the keystore shipping with this application is "**dirx**". The default password for the keystore shipping with Oracle's JRE is "**changeit**".
  You may find the command **keytool -help** useful, too.
  For more detailed information on the **keytool** command, please refer to the URL given at the beginning of this topic.
- Activate SSL/TLS in this application. Note that the default port for secure connections is 636. To activate SSL/TLS (more exactly, certificate-based server authentication), check 

  in the Server dialog as shown in the following example of the server dialog. Note that this checkbox is not available if the Java Runtime Environment you are using does not support SSL/TLS.

Here is an example:

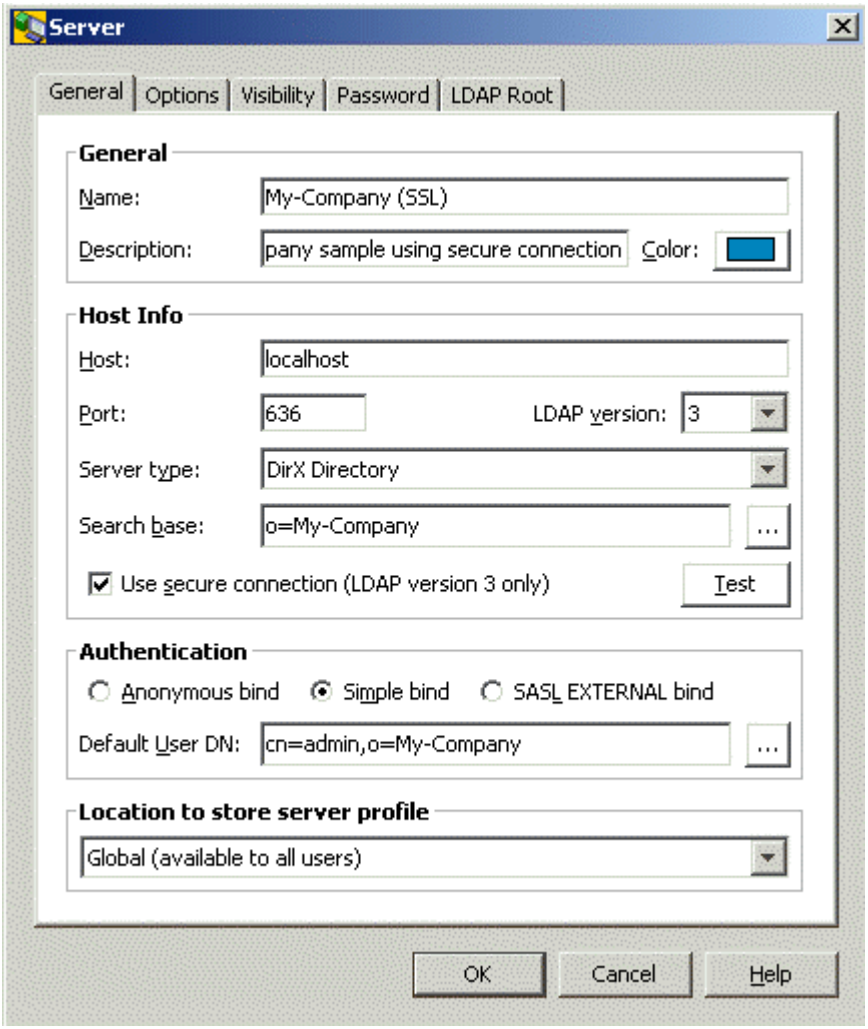


Figure 66. Server Properties General Tab for SSL/TLS

7.1.3. Smart Card Login

From the LDAPv3 protocol view Smart Card Login (SCL) maps onto an LDAP SASL bind with the mechanism type EXTERNAL. This means the security services of the underlying TLS/SSL layer is used to perform the client authentication that is based on strong cryptography.

After a successful bind the DSA determines the Authorization Identity of the user. This is the basis for the Access Control decisions made in all subsequent operations. By default, the Authorization Identity is mapped to the DN specified in the subjectDN from the user's certificate.

The following sections describe how to set-up Smart Card login with DirX Directory Manager.

7.1.3.1. Software Requirements

SCL requires that the software product Eviden CardOS API (64-bit) is installed. After a successful default installation of CardOS API the PKCS#11 library named "cardos11_64.dll" is in the folder

- C:\Windows\System32

7.1.3.2. Configuring the PKCS#11 Library for DirX Directory Manager

In order to successfully perform SCL logins the path to the PKCS#11 library **cardos11_64.dll** must be specified in DirX Directory Manager. In menu **Tools** -> **Options** specify the path in the **Smart Card** frame:

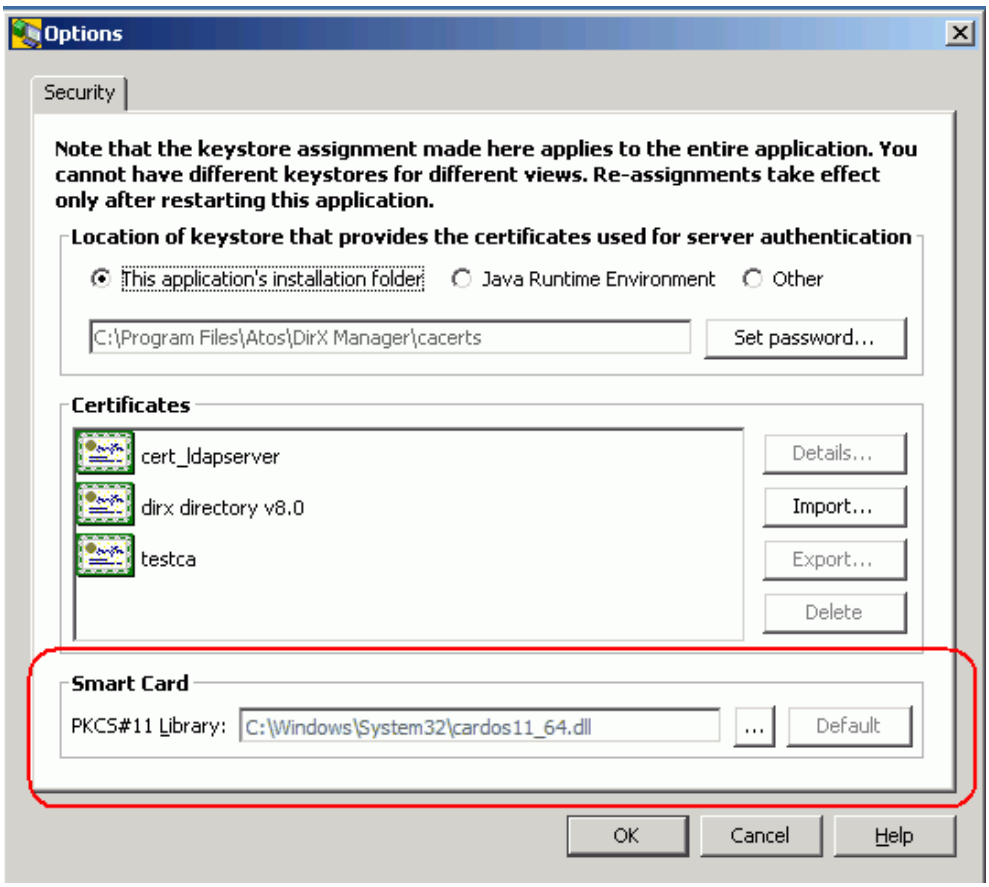


Figure 67. Configuring the PKCS#11 Library for DirX Directory Manager

7.1.3.3. Setting up the LDAP Server and the DSA for Smart Card Login

In order to successfully perform a sasl EXTERNAL bind the LDAP server must meet the following preconditions additionally to those that are required for SSL binds without client authentication:

- The LDAP server must request from the client to perform client authentication when SSL/TLS is used. This behavior is specified in the LDAP SSL Client Authentication Required attribute.
- The value of the LDAP SASL Authz Id Mapping (LSAIM, ldapSaslAuthzIdMapping) must specify how the DSA determines the requestor used for all operations following the bind. The default is **Use subjectDN from Certificate as the bind initiator**.
- The LDAP server must trust the CA that issued the clients userCertificates; that is the CA certificate must be added to the LDAP Trusted CA Certs (LSTCC, ldapTrustedCACerts) attribute.
- Depending on the value of the LDAP SASL Authz Id Mapping attribute an INITIAL index

must be created for the attribute that the mapping is based on.

- The SSL-External-DAP bind between the LDAP server and the DSA that the ldap sasl bind is mapped upon must be a local one, that is the LDAP server and DSA must run on the same host. Otherwise the environment variable DIRX_SSL_HOSTS can be used to specify the IP address that the DSA accepts as initiator hosts for LDAP SASL bind with the mechanism type EXTERNAL.

Perform the following steps to prepare the LDAP server and the DSA for Smart Card login:

7.1.3.3.1. Configuring the LDAP Server:

- In the **Configuration View**, edit the LDAP SSL configuration subentry to specify the values for the attributes:
- LDAP SSL Client Authentication Required
- LDAP SASL Authz Id Mapping
- LDAP Trusted CA Certs
- In the **Client Authentication** tab check **Client Authentication Required** to set the LDAP SSL Client Authentication Required (LSCAR, requireSSLClientAuth) attribute to **TRUE**. The LDAP server now requests from the client to perform client authentication when SSL/TLS is used.
- The value of the LDAP SASL Authz Id Mapping attribute is managed together with the **Client Authentication**. In our example the DSA uses the subject name of the userCertificate that the client provides in the context of the sasl bind operation. So leave the value **Use subjectDN from Certificate as the bind initiator** for **SASL Authorization ID mapping** unchanged.
- Finally add the CA certificate to the LDAP Trusted CA Certs (LSTCC, ldapTrustedCACerts) attribute that the LDAP server trusts the CA that issued the clients userCertificates. In the **Client Authentication** tab -> **Trusted CA certificates**, press  and import the CA certificates.
- Press the Save button to save the changes to the LDAP SSL configuration subentry.

The following figure illustrates the relevant settings of the LDAP SSL Configuration subentry:

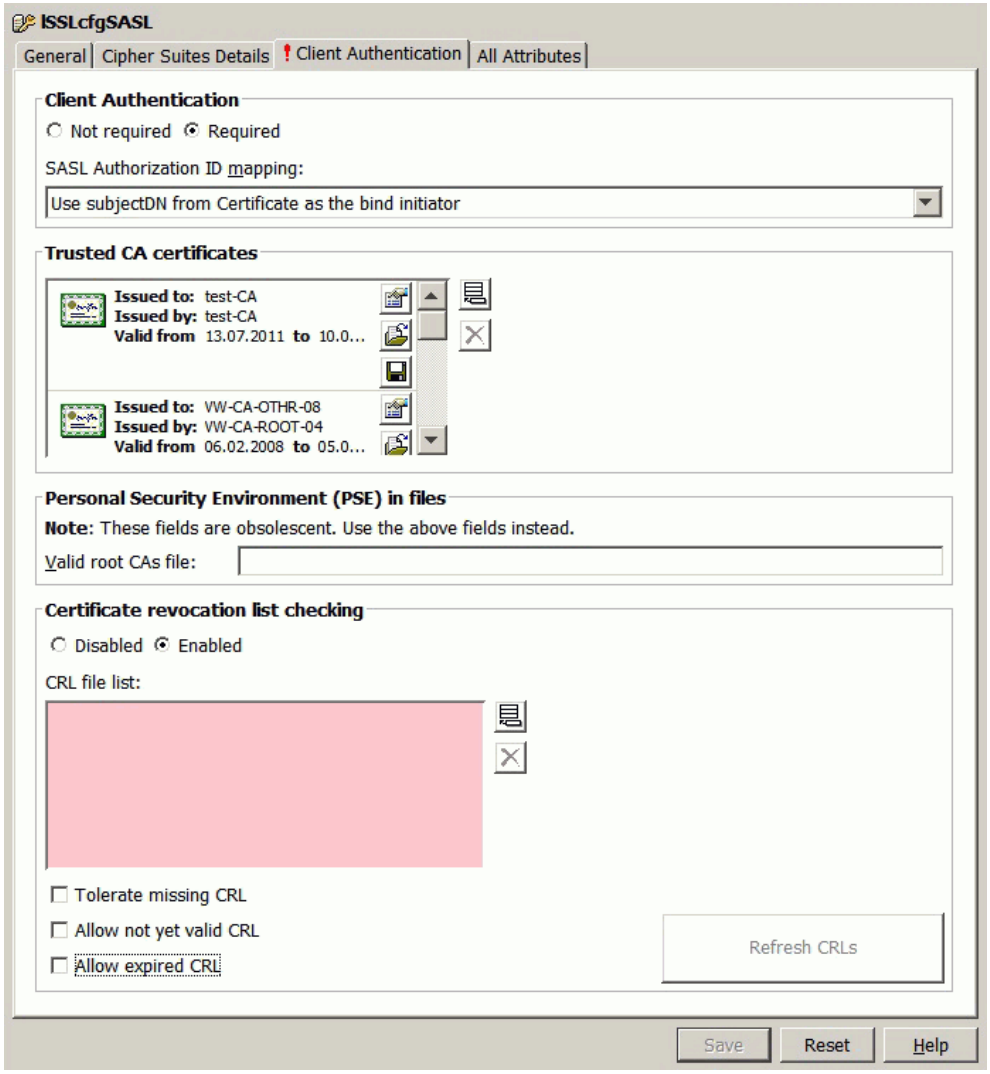


Figure 68. LDAP SSL Configuration Subentry Client Authentication Tab for Smart Card Login

7.1.3.3.2. Configuring the DSA:

- Recall from the settings of the LDAP SSL configuration subentry that the LDAP SASL Authz Id Mapping attribute specifies how the DSA determines the DN of the bind requestor from the sasl bind credentials:
- For the default **Use SubjectDN from Certificate of the bind requestor** no additional index is necessary.
- For **Use the directory entry that owns the Certificate as bind initiator** an initial index for the **userCertificate** Attribute is required.
- For **Use the directory entry that owns the email attribute of the Certificate Extension altName as bind initiator** an initial index for the **email** Attribute is required.

In our example it is not necessary to create an additional index because the default value is used.

- If the LDAP server and the DSA are not colocated add the IP address to the environment variable DIRX_SSL_HOSTS. (See "Environment Variables" in the *DirX Directory Administration Reference* for details.)

Now the DSA and the LDAP server are prepared to accept LDAP SASL binds with the mechanism type EXTERNAL.

7.1.3.4. Setting up the Client

From the LDAP client perspective SCL is one flavor of SASL EXTERNAL binds. In the **General** tab of the **Server** properties -> **Authentication** check **SASL EXTERNAL bind** and select **Smart Card (PKCS#11)** for **Client Keystore**:

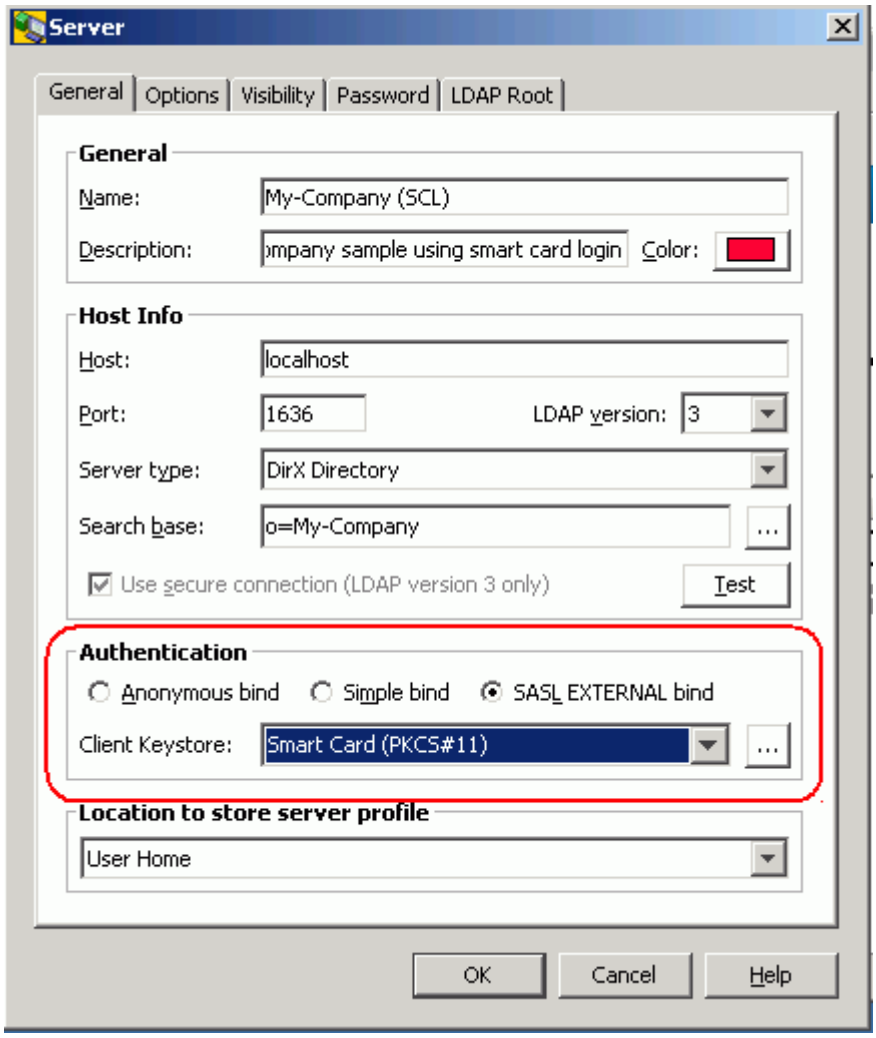


Figure 69. Server Properties General Tab for Smart Card Login

Now DirX Directory Manager and the DirX Directory service are prepared to accept smart card logins.

See the *Release Notes* for information about supported smart card types.

7.2. Basic Patterns/LDAP Functionality

This chapter provides information on the basic patterns supplied by the core component of this application. It also covers the "core" LDAP functionality that forms an integral part of the core component. Note that some advanced LDAP features like schema management or subentry management are not supplied by the core component; they are only available, if the corresponding plug-ins are installed.

It is divided into the sub chapters

- Main Window
- Special Mouse Operations
- Positioning
- Abandoning an in Process Search
- View Panes
- Property Editors
- Standard Dialogs

7.2.1. Main Window

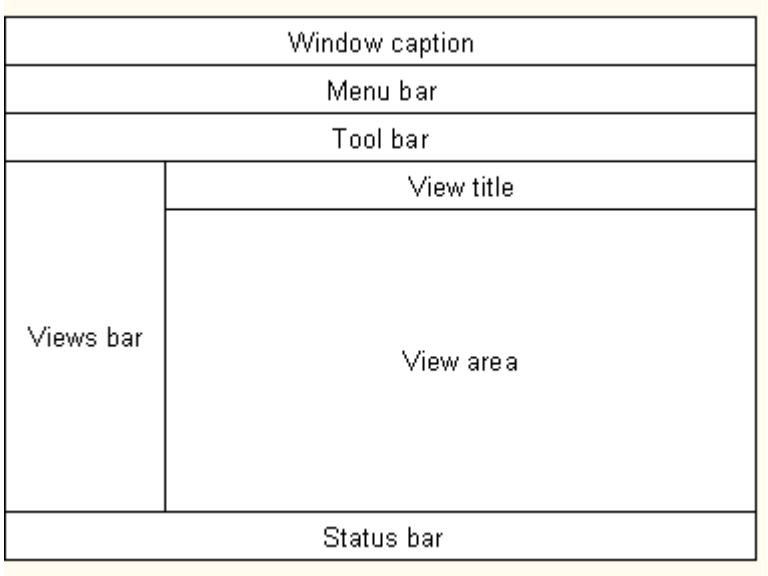


Figure 70. Main Window

The main window typically contains:

- The Window caption. The window caption is configurable. By default, it displays the name of the profile and the name this application has been given.
- The menu bar
- The tool bar
- The view title. The view title repeats the name of the view that is currently selected in the views bar
- The views bar
- The view area. The view area consists of view panes.
- The status bar

7.2.1.1. Menu Bar

The menu is configured in a file called **menubar.xml**. Plug-ins can modify and extend the menu.

Example:

Figure 71. Menu Bar

The core menu typically includes:

- File
- Edit
- View
- Schema (requires an additional plug-in, as it is shipped by DirX Directory Manager)
- Tools
- Help

Note that:

- Some menu items are available through
- The right mouse button, too
- The right mouse button only
- Drag and Drop is a way to move/copy entries you may find more convenient than the clipboard functions cut/copy/paste.
- Plug-ins may refine the behavior of certain menu items to their requirements.
- Menu items may be gray if they are available but do not apply to or are not implemented for the actual context.
- Certain menus or menu items, although implemented, may be absent because they have been disabled.

7.2.1.1.1. File Menu

Here is an example of the file selection:

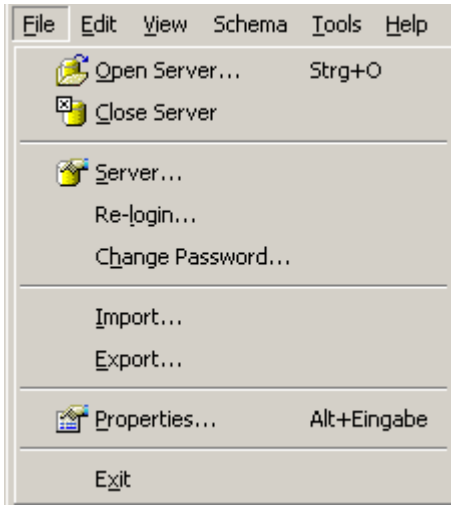


Figure 72. File Menu

- Open Server/Manage Server Profiles
- Server
- Login/Logout/Re-login
- Change Password
- Import/Export
- Properties
- Exit

Open Server/Manage Server Profiles

These two functions offer nearly the same functionality. For details refer to the description of the "Standard Dialog" Open Server.

Allows you to add, delete, and change server profiles.

There can be several server profiles per server. Server profiles can - depending on configuration data - apply to

- An entire view group
- A particular view
- A particular node in a tree pane
- A particular search pane

Note that the server profiles you need may come pre-configured with your installation.

For details refer to the description of the "Standard Dialog" Server.

Server

Allows you to enter the data required to connect to an LDAP server, especially the host name and port number of the server to be addressed.

For details refer to the description of the "Standard Dialog" Server.

Login/Logout/Re-login

Allows you to bind/unbind from the LDAP server currently addressed. You are likely to come across one of these alternatives:

Re-login

Disconnects from the LDAP server and does another bind then.

Login/Logout

Another Login assumes a preceding Logout and vice versa.

Logout disconnects from the LDAP server.

For details refer to the description of the "Standard Dialog" Login.

Change Password

Allows you to change the password that belongs to the distinguished name that was used for the currently active login. The topic "Changing Your Own Password" provides details.

Changing or deleting somebody else's password is possible, too, provided you have the required access rights. See the topic User Password in Basic GUI Functions for details.

See also: Property Tab "All Attributes", Server, Reset Password, User Password

Import/Export

Allows you to export data from the LDAP server that is currently addressed into a file or import data from a file into the LDAP server currently addressed.

Export

Export files can be:

- LDIF content files (http://www.ietf.org/rfc/rfc2849.txt)
- DSML v1 files (http://www.oasis-open.org/)

Values only containing ordinary ASCII letters (with minor exceptions) are exported as is, while others get base64 encoded. Exception: In case of DSML v1, DNs are always exported in "native" UTF-8. Base64 (see also RFC 2045) regulates how to transform binary data into printable data using nothing but the 64 characters "A-Z, a-z, 0-9, +, /". Base64 increases the original size by 33%; it is not an encryption.

Notes

- The export of collective attributes is to be taken with a grain of salt, since it is impossible to find out beyond all doubt whether an attribute is collective or not. So, as a substitute, all attributes whose type starts with "collective", are taken as collective attributes.
- Access Control may keep you from getting certain entries or certain attribute values.
- Administrative restrictions configured at the LDAP server may cause the export function to terminate uncompleted.

The advance button offers additional options.

Import

Import files can be:

- LDIF change files (http://www.ietf.org/rfc/rfc2849.txt)
- LDIF content files (http://www.ietf.org/rfc/rfc2849.txt)
- DSML v1 files (http://www.oasis-open.org/)
- DSML v2 files (http://www.oasis-open.org/)

As for the character encoding, the same regulations as with export apply; additionally, "native" UTF-8 is recognized for attributes, too.

The advance button offers additional options.

See also:

- Exporting/importing certificates
- Exporting/importing JPEG images

Properties

Allows you to display the attributes of the entry currently selected in a separate window.

May also allow you to edit them. In edit mode, the layout of this window may or may not change considerably. Push buttons explained through tool tips simplify the handling and make it more convenient.

For details refer to Property Dialog, Property Editors, Property Pane and All Attributes/Right Mouse Button.

Exit

Allows you to exit the application.

Note that this application will try to present itself at the next start-up just the way it appeared at the time you exited it. For this reason, it tries to store related information on disk or in the directory before finally exiting.

7.2.1.1.2. Edit Menu

Here is an example of the edit selection:

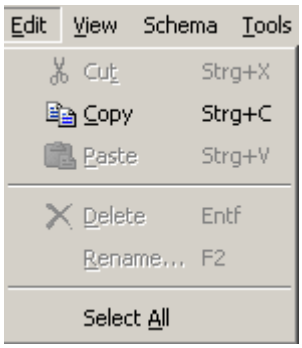


Figure 73. Edit Menu

- Cut/Copy/Paste
- Delete
- Rename
- Select All

Cut/Copy/Paste

Allows you to use typical clipboard functions on currently selected entries. Works in list panes and tree panes (crosswise, too). Works also between different instances of this application.

See also: Drag&Drop.

Delete Entry

Allows you to perform standard "delete" functions:

- Deletes the entry/entries currently selected
- Checking off the "with all children" check button will cause all children to be deleted, too (recursively)
- Note that a delete operation may fail due to the existence of administrative children that are invisible to the client

Rename

Allows you to rename the object that is currently selected.

Notice the special LDAP feature "Retain old name". It allows you to let the old name survive in the **attribute** that stores the last RDN. For the **DN** itself, however, only the new name counts. Repeated execution of the rename function with "Retain old name" checked leads to a number of values in that attribute.

For details refer to the description of the "Standard Dialog" Renaming

Select All

Allows you to select all entries in the list that is currently active.

7.2.1.1.3. View Menu

Here is an example of the view selection:

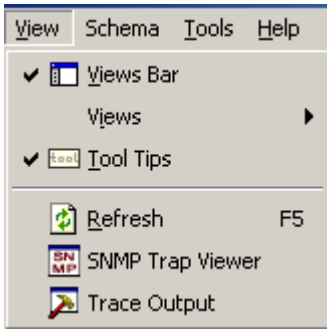


Figure 74. View Menu

- Views Bar 
- Views

- Tool Tips 
- Refresh 
- SNMP Trap Viewer 
- Trace Output 

Views Bar

Allows you to display a window that offers access to views that are grouped into view groups.

The views bar shows the available views and allows you to switch between views. In simple configurations (for example, when there is only one view group that contains one view), the views bar is not very useful. Selecting this menu item alternately checks the views bar (the views bar becomes visible) or unchecks it (the views bar is absent).

Views

Provides a cascading menu that allows you to select a view group and then a view. You can also select a view in the views bar (if it is not hidden).

Tool Tips

Allows you to turn on or off tool tips.

Refresh

Allows you to refresh the data displayed in the current view by re-reading it from the server.

This function is usually also available in the tool bar.

SNMP Trap Viewer

Allows you to open the SNMP Trap Viewer.

Trace Output

Allows you to enable, disable and configure trace information and display it in a window. This selection is only visible if the Trace Window plug-in is installed.

7.2.1.1.4. Tools: Options

This dialog allows for

- Locating the keystore that is to use for authenticating a server, if the client-server communication is to be run based on a secure connection. By specifying a keystore that doesn't yet exist, you can cause an empty keystore to be created.
- Managing keystores.
  You can view details of certificates, delete certificates from, export certificates from, import certificates into the keystore currently selected. You can change the password that is required for managing the keystore.

- Managing the smart card library.

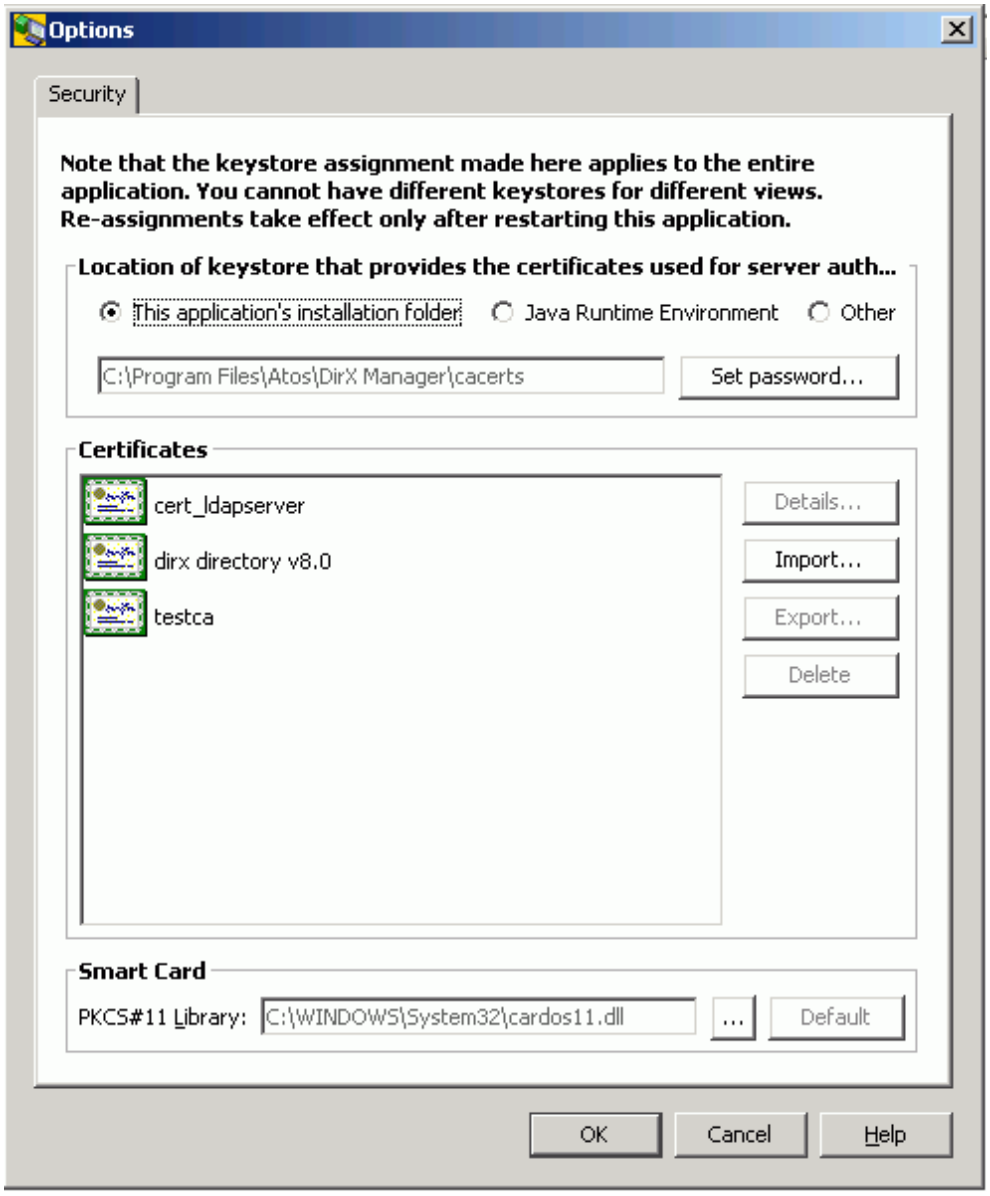


Figure 75. Tools: Options Dialog Box

See also: SSL/TLS.

7.2.1.1.5. Help Menu

Here is an example of the help selection:

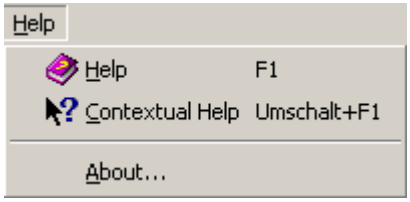


Figure 76. Help Menu

The help menu gives you access to:

- Help system
- Contextual Help
- About box

Help

Gives you access to the Help System. It may take a little while to display help information the first time you use it because it needs to perform some initialization tasks.

Contextual Help

Allows you to display contextual help information. When you use this function, the cursor changes to  . Now you can right-click anywhere within this application to display pop-up help information that is specific to the context you selected by right-clicking on it. For example, you can right-click an empty spot next to the toolbar to display pop-up help information on the tool bar. If there is no context-sensitive help available for the selected context, the cursor returns to its original shape.

About

Allows you to obtain information about:

- The version of the core component ("Framework")
- The versions of the installed plug-ins
- The version of the Java Runtime Environment (JRE) that started this application
- Additional data that is primarily used for maintenance purposes

Plug-ins may extend the about dialog.

7.2.1.2. Tool Bar

The tool bar is configured in a file called **toolbar.xml**. Plug-ins can change and extend the toolbar and add additional toolbars.

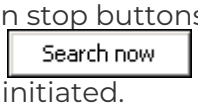
Example:

Figure 77. Tool Bar

Tool tips describe the menu items that a tool represents. Tools may be gray if they are available, but do not apply to or are not implemented for the actual context. The tools provided by the core component typically include:

- History buttons 
- Back 
  Steps back one step in the history recorded for the currently selected view. The number of steps you can go back is limited to avoid excessive memory consumption.

- Forward 
  Steps forward one step in the history recorded for the currently selected view.
- Stop button 
  Cancels the operation currently in progress. Not every potentially time-consuming operation may offer to be cancelled. A plug-in may or may not support this feature, or it may support it to some extent. Dialogs may block the main window. In this case, the stop button is not accessible in the main window while the dialog is being displayed. However, dialogs may offer their own stop buttons, typically a button like this one 

  . Note that the button 

  may automatically change to each time a search is initiated.
- Additional buttons, which are typically available both as tools and as menu bar selections. Just take a look at the button's tool tip.

7.2.1.3. Views Bar

Here is an example of the views bar:

Figure 78. Views Bar

The views bar is a window that offers access to views. You can also switch to a different view with the view menu (unless this menu has been disabled).

In the sample shown here (which, by the way, comes from a plug-in), you can see the **view groups** called "DirX Manager", "My-Company" and "My-Company (SSL)", and the views of view group "My-Company", called "Directory Entries", "Quick Search", "Configuration", "Schema", "Replication", and "Monitoring". Possible differences between different views include:

- Access to different LDAP servers
- Different restrictions, like different search bases, visibility of objects/attributes
- Different presentation (tree browsing, searching, ...)

A view is organized in

- a view title and in a
- view area made up of view panes.

7.2.1.4. View Area

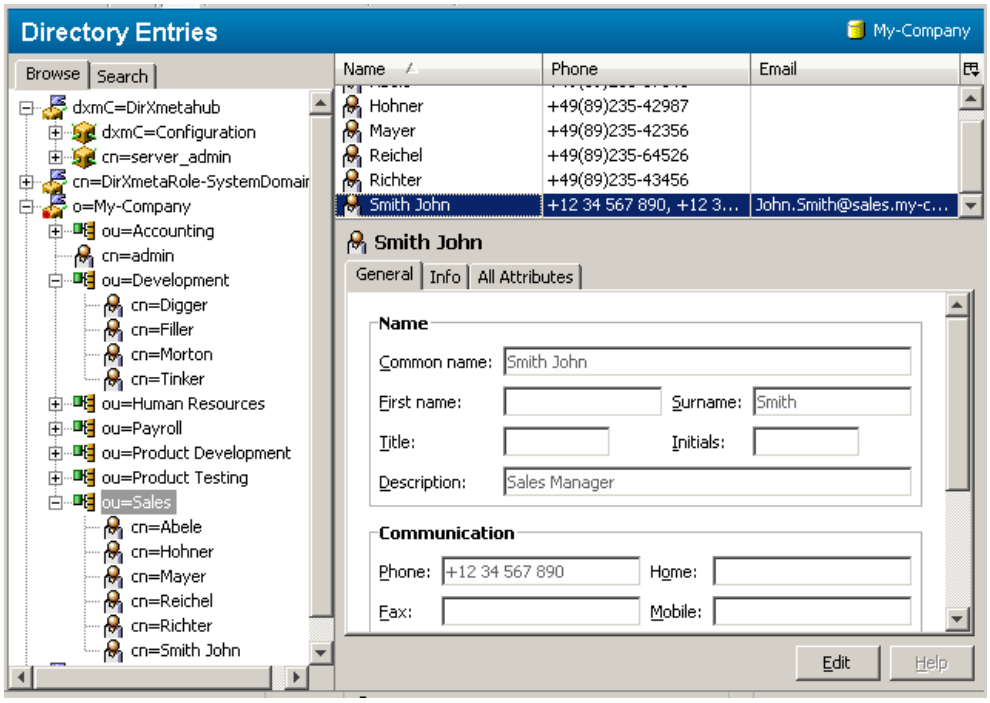


Figure 79. View Area Directory Entries

The view area presents the view that is currently selected in the views bar. The view area consists of a "view title" and of "view panes".

A particular view typically offers a search pane or a tree pane, or a combination of both (as shown in the previous screen shot), with an associated search result list pane and/or property pane (also shown in the previous screen shot).

7.2.1.5. Status Bar

Here is an example of the status bar selection:

Figure 80. Status Bar

The status information provided by the core component includes:

- Context-dependent transient information such as the number of entries copied, upon copying entries
- Connection information, such as the server currently accessed and the DN used for login.
  Use the tool tip mechanism to display server name and port number:

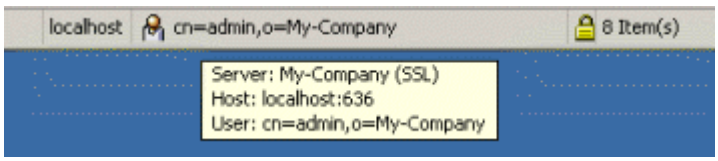


Figure 81. Status Bar with Tool Tip

- An indication of a secure connection: 
- The number of entries displayed in the currently selected pane
- The number of selected entries displayed in the currently selected pane

Plug-ins may provide additional information in the status bar.

7.2.2. Special Mouse Operations

Some functionality is available through special mouse operations:

- Drag & Drop
- Tooltips
- Right Mouse button
- In tree and list panes
- In text fields
- In the All Attributes Tab of property dialogs/panes
- In the column headers of list panes

7.2.2.1. Drag&Drop

Drag and drop allows you to copy or move currently selected entries to different nodes. Drag and drop works in list panes and tree panes (crosswise, too); and it works also between different instances of this application.

See also: Clipboard functions in the Edit menu and in the property pane/dialog.

7.2.2.2. Tool Tips

This application makes extensive use of tool tips. Tool tips are contextual annotations that

are briefly displayed when the mouse cursor remains on a context-sensitive spot, for example, on a property value.

7.2.2.3. Right Mouse Button

The places where the right mouse button is functional include

- Tree and list panes
- Text fields
- The All Attributes Tab of property dialogs/panes
- The column headers of list panes

7.2.2.3.1. Tree & List Panes

Some menu items are *additionally* available through the right mouse button.

Some functions are *solely* available through the right mouse button:

- Copy to
- Filter (not implemented)
- Lock Account/Unlock Account
- Move to
- New
- Reset Password
- Search

Plug-ins can add menu items to the right mouse button and modify right mouse button menu items.

Here is an example of the right mouse button selection (right-clicking the mouse on an entry):

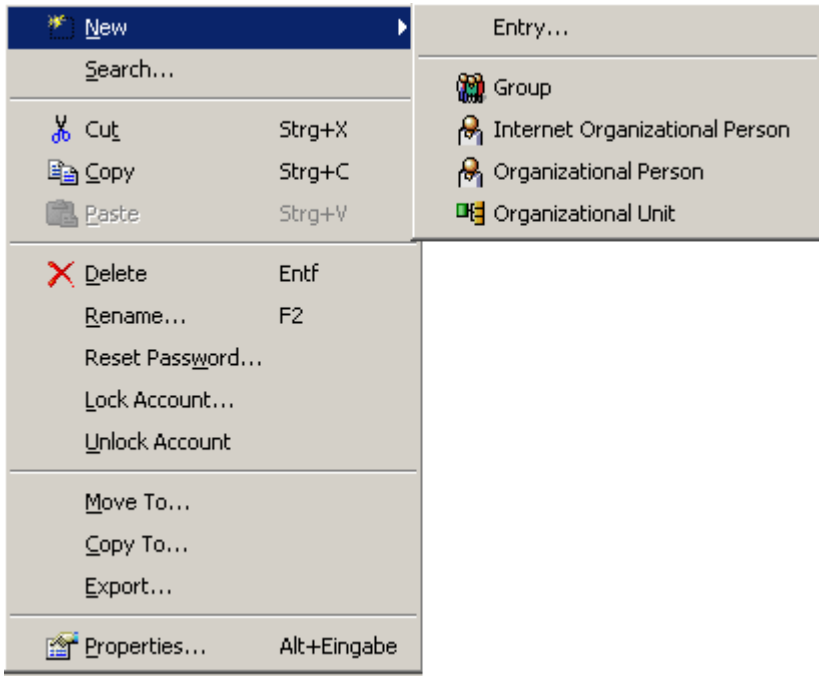


Figure 82. Context-sensitive Menu for a Selected Entry

Copy To

Copies a node including all its children to a different parent. Note that if the LDAP server does not support simple paging, restrictions like time/size limit may keep this operation from completing.

Filter

Filter (not implemented)
Displays a dialog that permits you to specify search restrictions for a one level search. Some fields already contain contextual data taken from the currently selected entry. The search result will appear in the tree as children of the actual node. That node is marked “(...)” to indicate that the list of children is likely to be incomplete.

Move To

Moves a node including all its children to a different parent. Note that not all LDAP servers support this feature.

New

Creates a new entry in the server currently addressed below the node to which the right mouse button has been applied. Note that this operation may fail for several reasons, for example:

- You do not have the required access rights.
- The LDAP server is configured so that it does not permit you to create a DN like the one you have chosen.

Usually, the function offers a limited selection of possible types of objects:

- Non-specific entry
- Customized object classes:
- Group
- Internet Organizational Person
- Organizational Person
- Organizational Unit

Depending on the number of possible types of objects, the core component provides four variants (which plug-ins may or may not use).

Entry

If the type of object is not pre-specified, you must first complete a dialog like this:

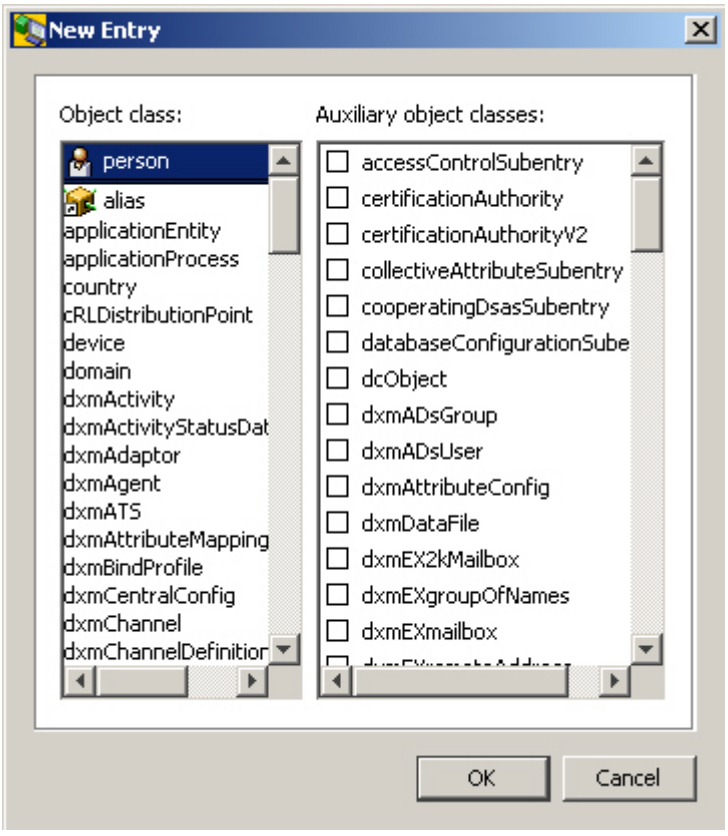


Figure 83. Selecting Object Class of New Entry

It allows you to specify (based on information provided by the server):

- One structural object class and (left half of dialog)
- A number of auxiliary object classes (right half of dialog)

For your convenience, the last recently used structural object classes are repeated in the upper part of the left half of the dialog.

After you complete this dialog, you can complete the creation of the entry in a dialog like this:

Tab: Info:

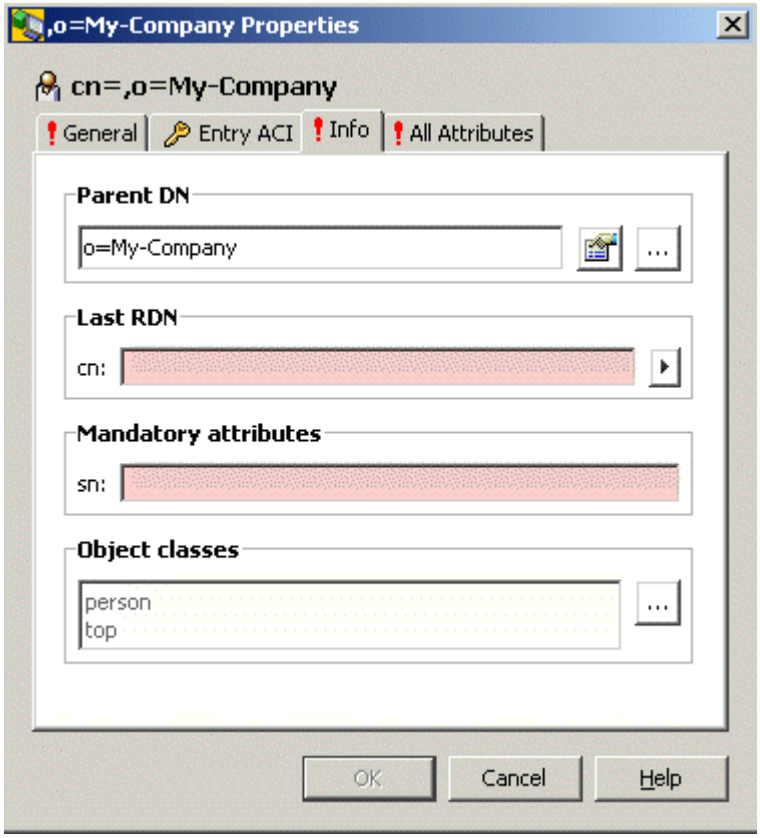


Figure 84. o=My-Company Properties Dialog with Info Tab of the New Entry

This tab displays

- The Parent DN (read-only)
- The last RDN (red shadowed)
- All additional mandatory attributes (red-shadowed; there are no additional mandatory attributes in the screenshot above)
- All Object Classes (read-only)

Tab All Attributes:

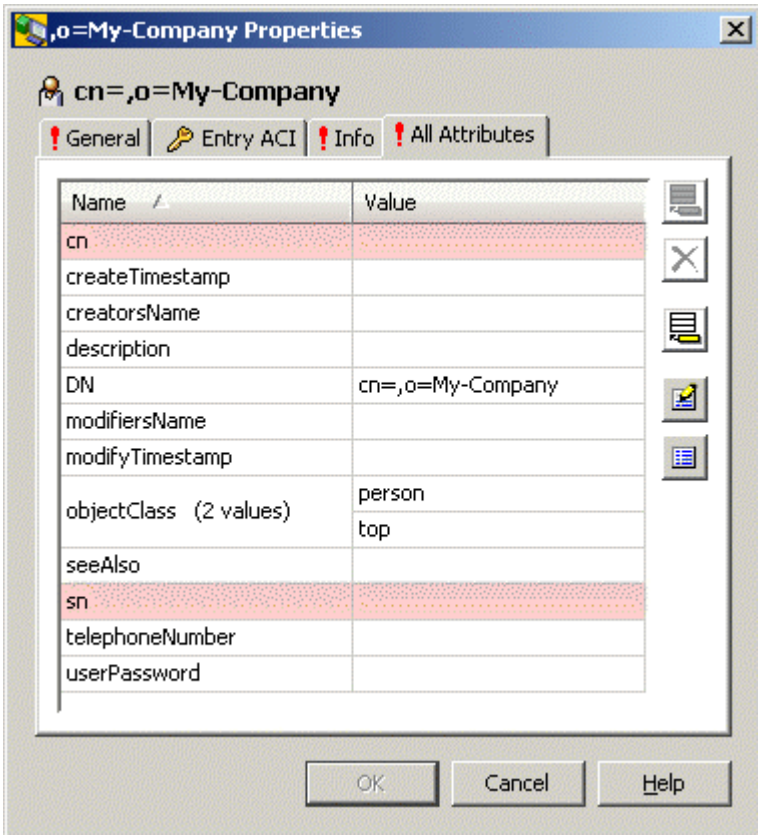


Figure 85. o=My-Company Properties Dialog with All Attributes Tab of the New Entry

This tab displays all attributes. The mandatory ones are highlighted in red.

You can assign multiple naming attributes, too. For details refer to "Renaming".

Group

If the object class "Group" is pre-configured, the dialog should look like this:

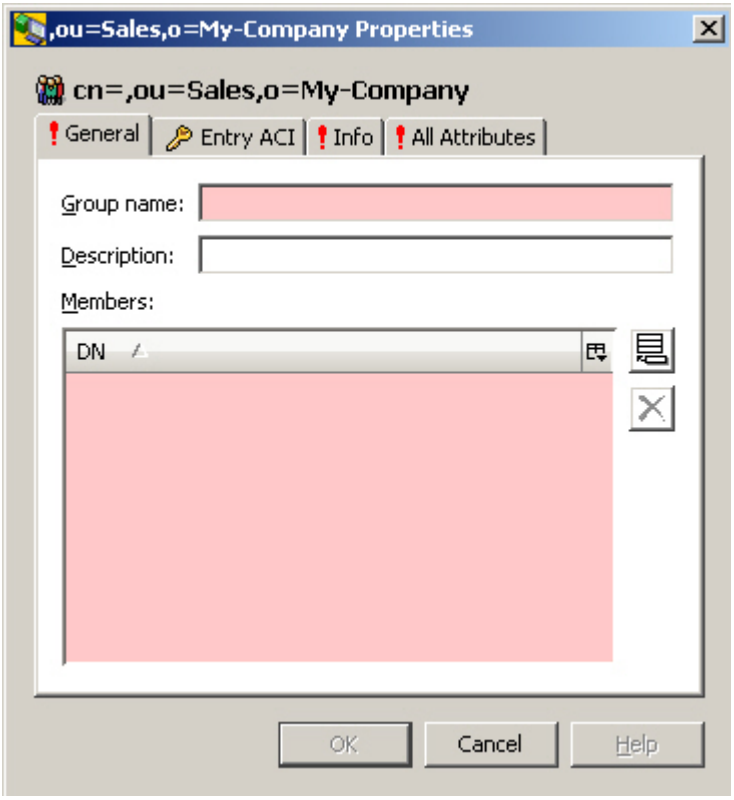


Figure 86. Group Properties Dialog Box for ou=,o=My-Company with General Tab

The "Info" and "All Attributes" tabs have the same meanings as the Info and All Attributes tabs of the "Entry" dialog.

Internet Organizational Person

If the object class "Internet Organizational Person" is pre-configured, the dialog should look like this:

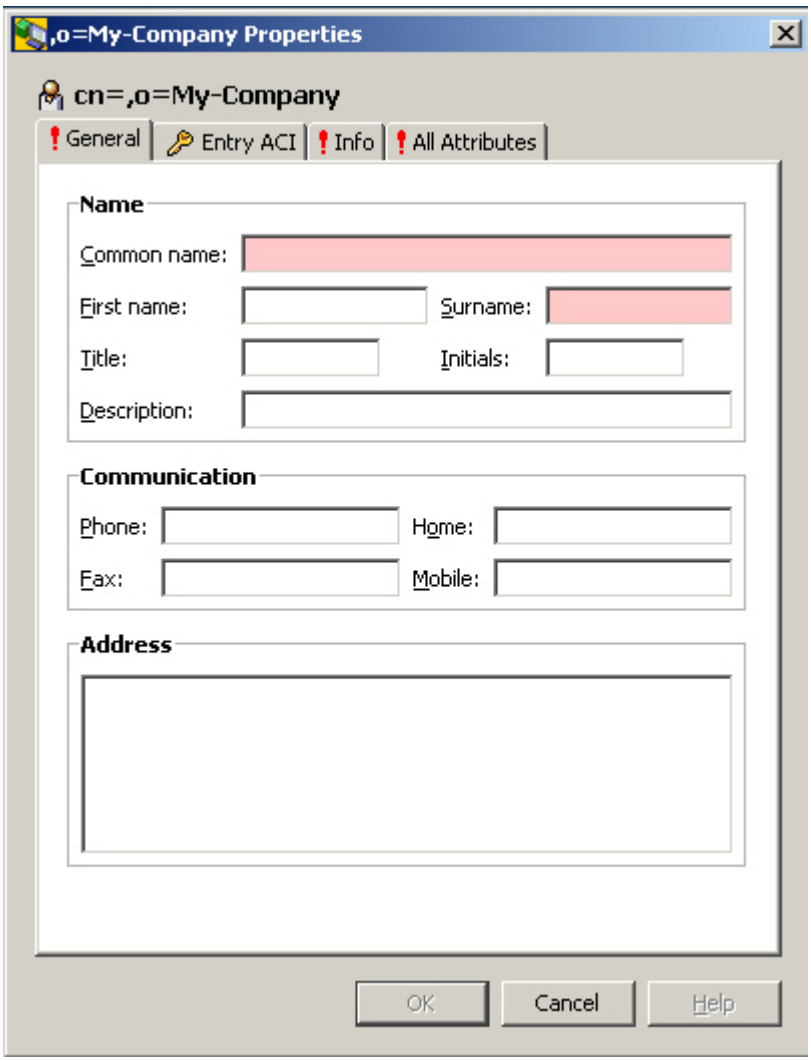


Figure 87. New Internet Organizational Person Dialog Box with General Tab

The "Info" and "All Attributes" tabs have the same meanings as the Info and All Attributes tabs of the "Entry" dialog.

Organizational Person

If the object class "Organizational Person" is pre-configured, the dialog should look like this:

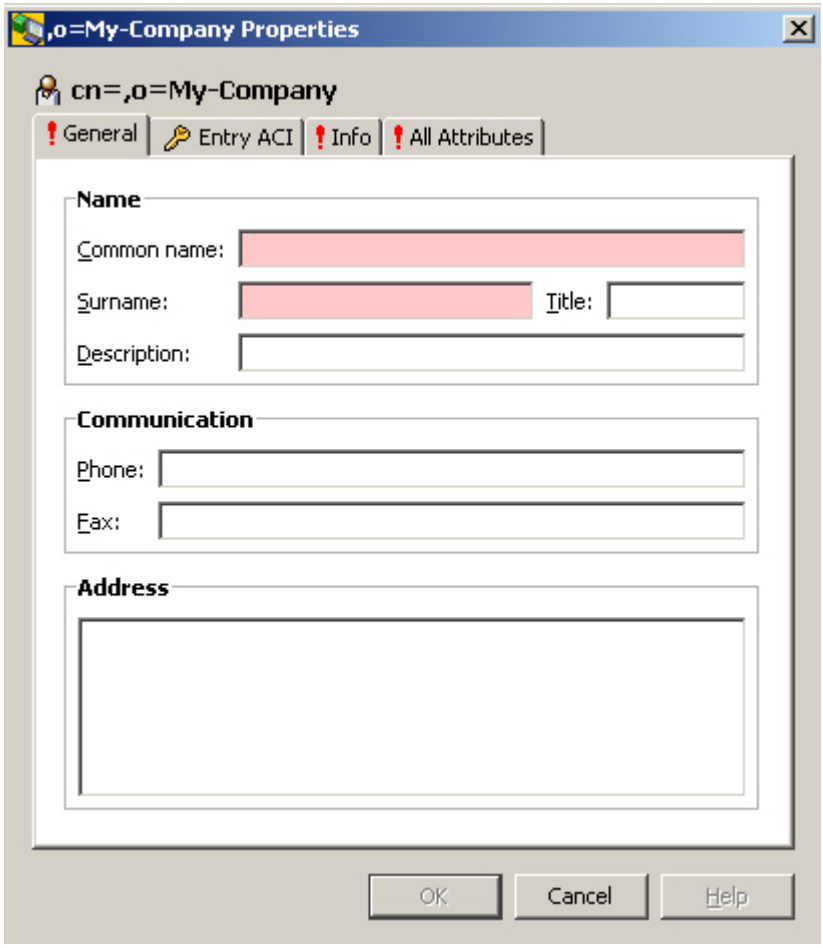


Figure 88. New Organizational Person Dialog Box with General Tab

The "Info" and "All Attributes" tabs have the same meanings as the Info and All Attributes tabs of the "Entry" dialog.

Organizational Unit

If the object class "Organizational Unit" is pre-configured, the dialog should look like this:

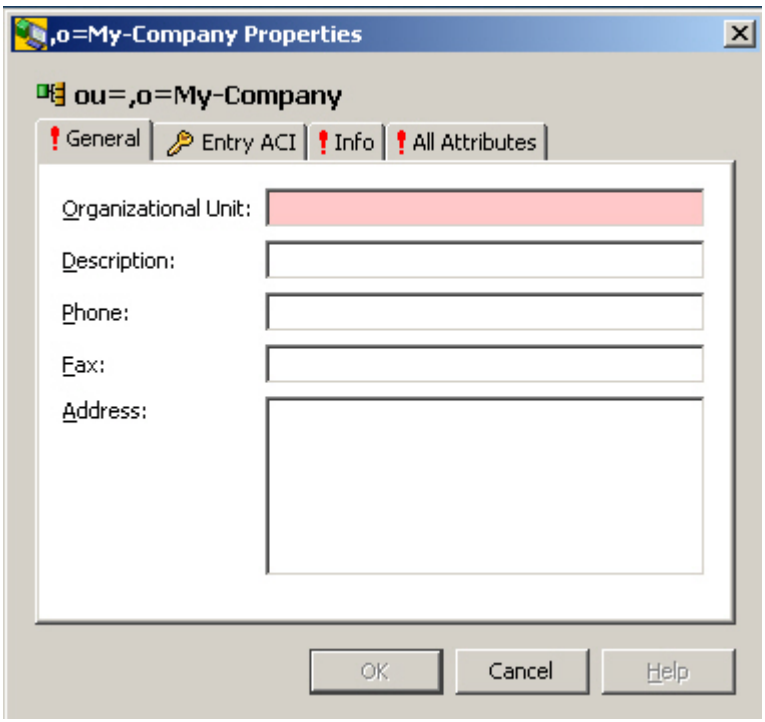


Figure 89. New Organization Unit Dialog Box with General Tab

The "Info" and "All Attributes" tabs have the same meanings as the Info and All Attributes tabs of the "Entry" dialog.

Variants

Depending on the number of possible types of objects, the core component provides four variants (which plug-ins may or may not use):

1. Objects of only one type can be created below the current node.
   Here is an example (only persons can be created):

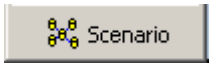


Figure 90. Only One New Object Type

2. A limited selection of object types can be created.

   Here is an example (only four types of objects can be created):

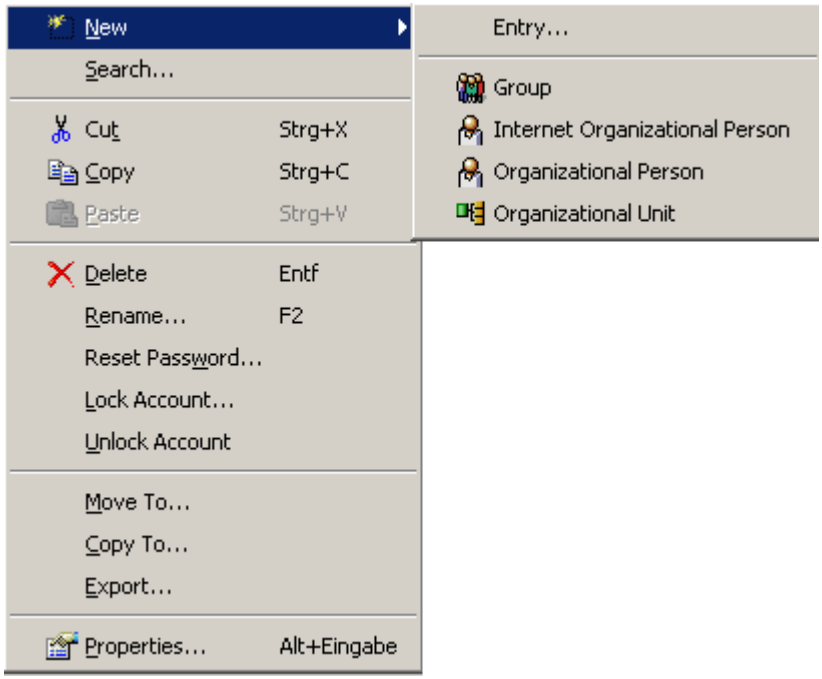


Figure 91. Four New Object Types

3. A large selection of object types can be created.

   Here is an example

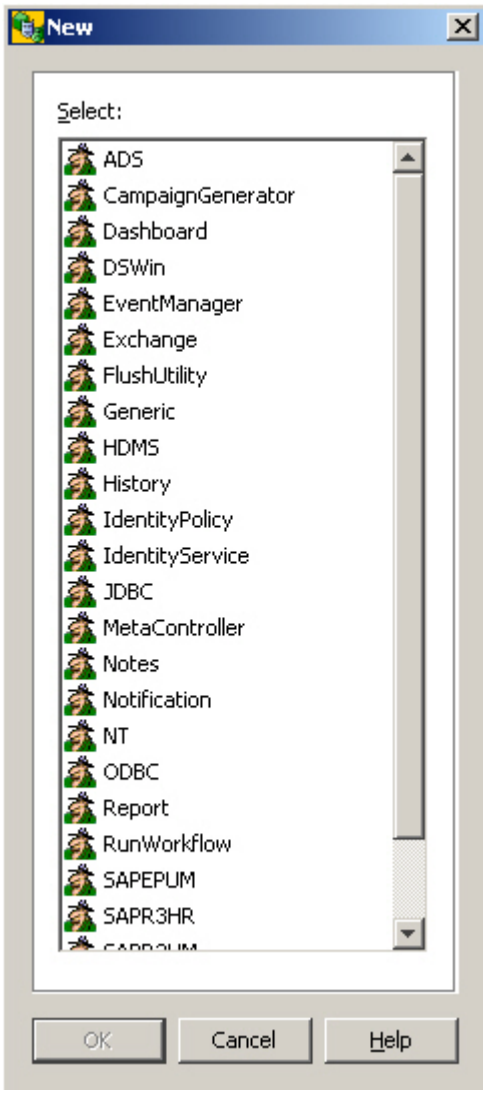


Figure 92. Large Selection of New Object Types

4. The type of the new object can be entered via the keyboard.

Lock/Unlock Account

Allows to prevent logins with the distinguished name (DN) of the currently selected entry or - if the entry is locked out already - to re-admit logins with that DN. Note that the server may lockout the DN as well implicitly, e.g. after a configured number of incorrect logins.

Here is an example of the context menu:

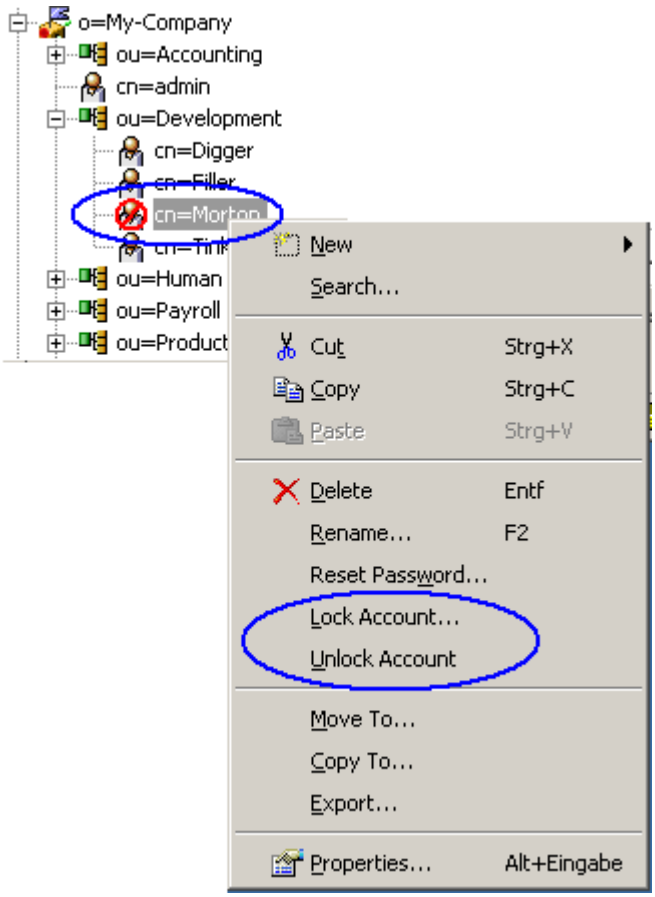


Figure 93. Lock/Unlock Account Context-sensitive Menu

In this example, the entry cn=Morton is *marked* (see note below!) locked out (visualized through the prohibitory sign). Note that in a network being comprised of several DSAs (supplier and consumers) the entry might be locked or marked locked at some DSA(s) other than the one, this application is currently bound to. An entry being locked out at some different DSA(s) only is not visualized through the prohibitory sign!

The context menu provides two items of interest here: **Lock Account...** and **Unlock Account...**.

- Both items are grayed out and not selectable, if the server does not appear to support this functionality (more precisely: if the operational attribute *pwdAccountLockedTime* is missing in the schema).
- **Unlock Account...** is offered even if the entry appears to be unlocked anyway, since it cannot be excluded that the entry is locked out at some other DSA(s) (see above). The chaining and shadowing mechanisms of DirX Directory are supposed to propagate the unlock operation to all participating DSAs, particularly to the ones that are affected (if any).
- **Lock Account...**
  You can lock an account indefinitely or you can lock it till the "lockout duration" expires. The lockout duration is an attribute of the "Password Policy" subentry that is supported by DirX Directory. You also can revise your decision and switch between these choices anytime.

  Here is an example of your choices, when the entry is currently locked out indefinitely:

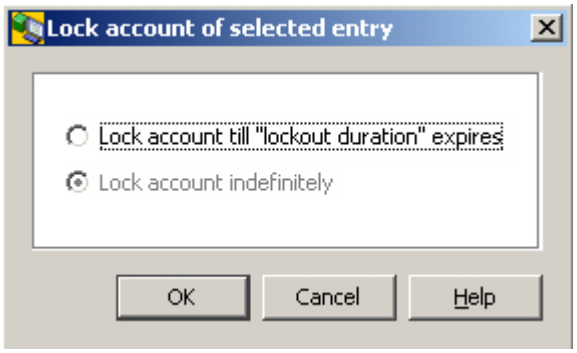


Figure 94. Lock account of selected entry dialog box for a entry locked out indefinitely

Same situation, but account lockout is not enabled or it is uncertain, whether account lockout is enabled or not (because e.g. the password policy subentry could not be read):

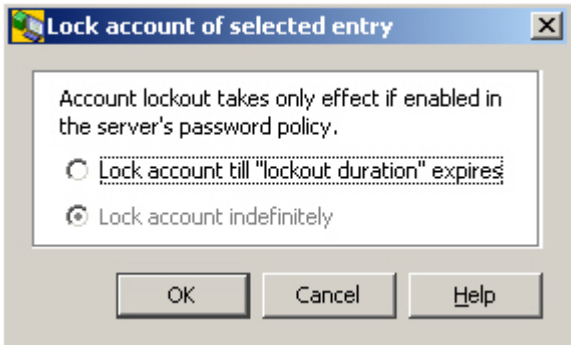


Figure 95. Lock account of selected entry dialog box for lockout not enabled or uncertain

- Click **Lock account till "lockout duration" expires**, if you want to lock out the entry temporarily for the time specified as Lockout duration; Lockout duration is one of the settings the password policy subentry provides (find more about this in the chapter "Password Policy").
- Click **Lock account indefinitely**, if you want to lock out the currently selected entry till you possibly decide to re-admit logins by this entry or to lock it till the lockout duration expires.

Notes:

- The functionality described here presupposes that (as DirX Directory do) the server supports a password policy subentry that provides a number of settings to control how passwords are used and administered, particularly a setting named "**Lockout duration**" (more exactly: an operational attribute named *pwdLockoutDuration*). If the DirX Directory Manager plug-in is installed, you are supposed to find more information on the password policy subentry in the according chapter within this help. Moreover, you may also have a look at the server documentation.
- Since **Lock Account...** and **Unlock Account...** only *mark* the DN to be" locked" or "unlocked", in order to get your setting effective, make sure, that another password policy subentry setting, namely the "**Enable account lockout**" (*pwdLockout*) flag is checked off.
- You are supposed to find the operational attribute *pwdAccountLockedTime* in the "All attributes" tab of the Properties dialog of the entries of interest. *pwdAccountLockedTime* being

- absent means "not locked"
- set to the 1st of January 1970 00:00:00 means "locked indefinitely"
- set to any other time/date means "locked, till this time/date + lockout duration has expired". So, changing the lockout duration in the password policy subentry affects all temporarily locked-out entries. A successful login after lockout expiry causes *pwdAccountLockedTime* to be removed by the server, which means the entry is not affected by modifications of Lockout durations any more

Reset Password

Allows an administrator to reset a user's password. Causes a dialog like this one to pop up:

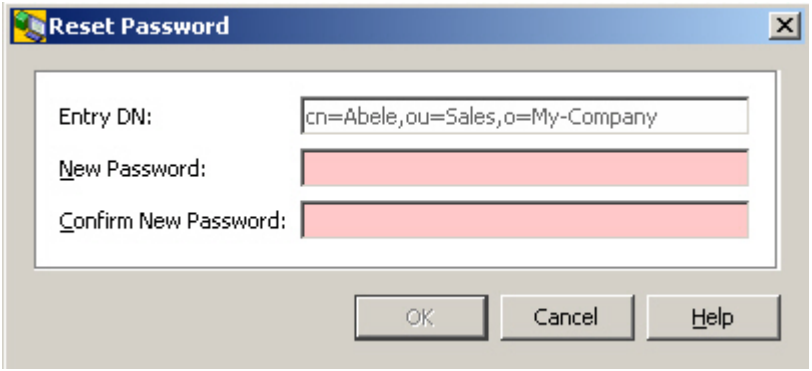


Figure 96. Reset Password Dialog Box

Note that there is a subtle difference between this function and the modify password functionality that is available as an ordinary attribute modification in all properties dialog/pane:

When changing a user's password with the reset password functionality, the server will be informed of this modification being a "reset". The server will in turn - depending on the password policy currently in place - force the user to update his password before granting him any "productive" operation.

Note that this function is only available if the server indicates its support in the LDAP Root DSE ("Password Policy Control"; Object Identifier, displayed directly or through tooltip: 1.3.6.1.4.1.42.2.27.8.5.1) and if an operational attribute named *pwdReset* can be set.

Furthermore, if the server keeping the DirX Identity users is set up, the mirrored administrator user must be present in the Connectivity under the RDN dxmC=Users,dxmC=DirXmetahub.

See also: Property Tab "All Attributes", Change Password, Server, User Password

Search

Opens a standard dialog as described in "Searching".

7.2.2.3.2. Text Fields

The right mouse button applied to a text field offers:

- Select all
  Selects the entire content of the text field
- Clipboard functions like Cut, Copy, Paste
- Occasionally, property-specific additional functions like "Check Number" in the example to follow

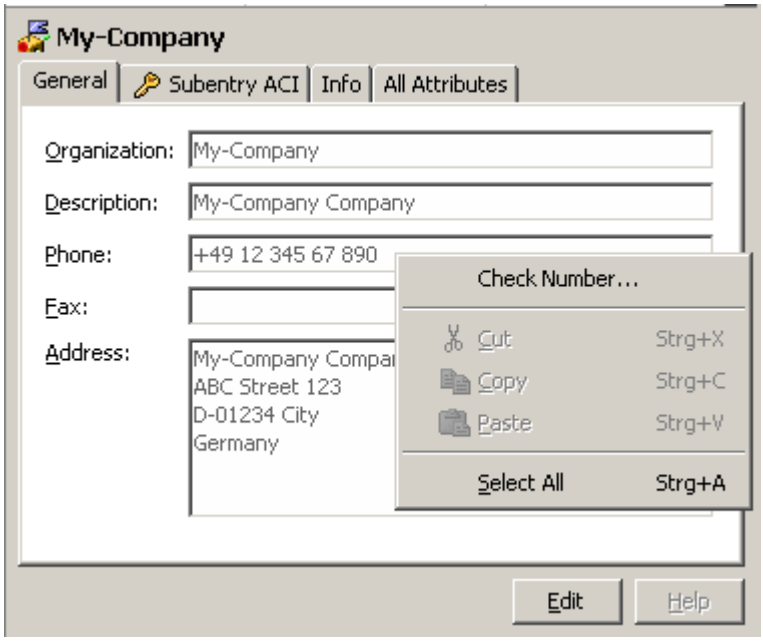


Figure 97. Property Specific Operation for Text Fields

7.2.2.3.3. All Attributes Tab

The all attributes tab that is available at some property panes and property dialogs has a number of functions on the right mouse button:

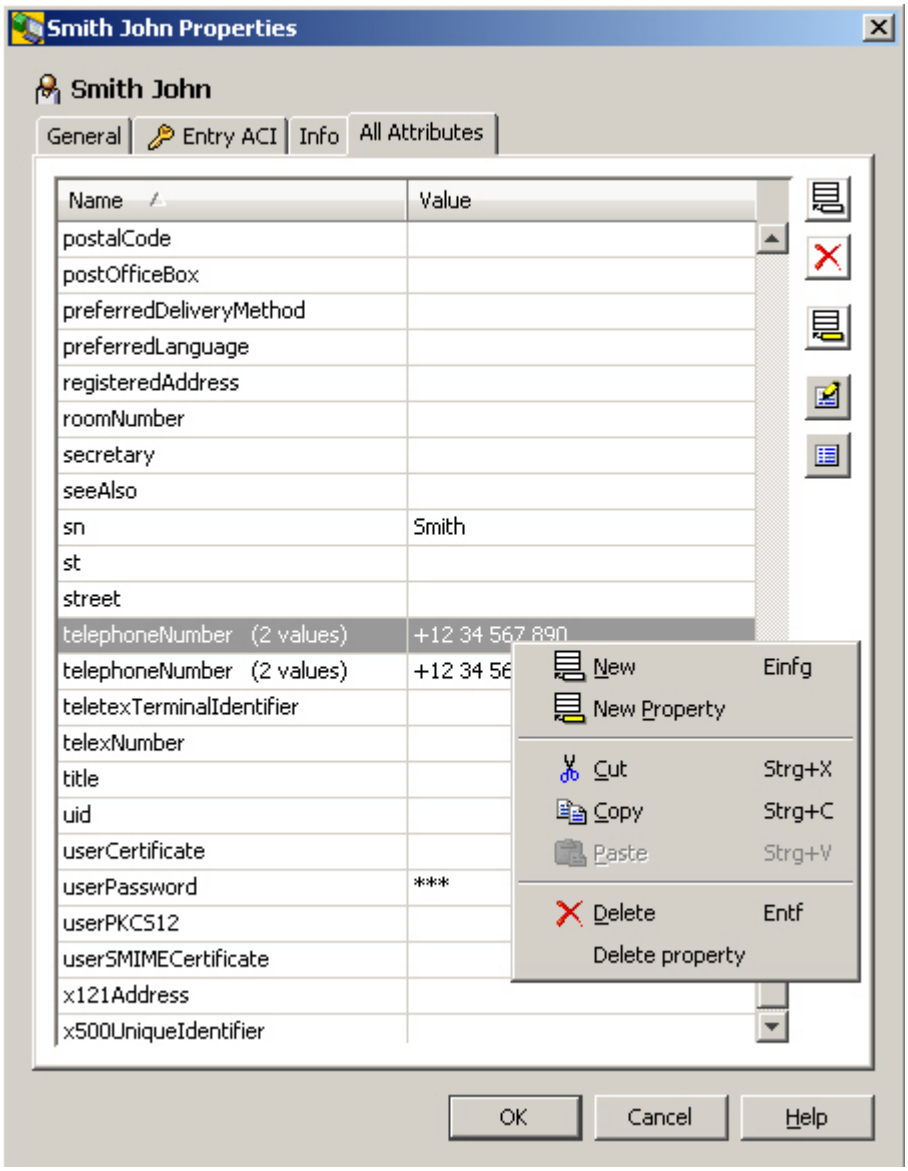


Figure 98. Properties dialog box with All Attributes Tab for Smith John

- Click "New" to insert a new value into a property.
- Click "New Property" to create a new property.
- Click "Cut"/"Copy"/"Paste" for clipboard operations that affect the selected property value.
- Click "Delete" to delete the selected property value.
- Click "Delete Property" to delete the entire property (same as "Delete", if property is single-valued)

Note the menu changes to the text field menu if you click in the value field of a property that is a text field.

7.2.2.3.4. Column Headers of Lists

Applying the right mouse button to a column header in (most) lists allows you to configure the columns that are to be presented. Here is an example:

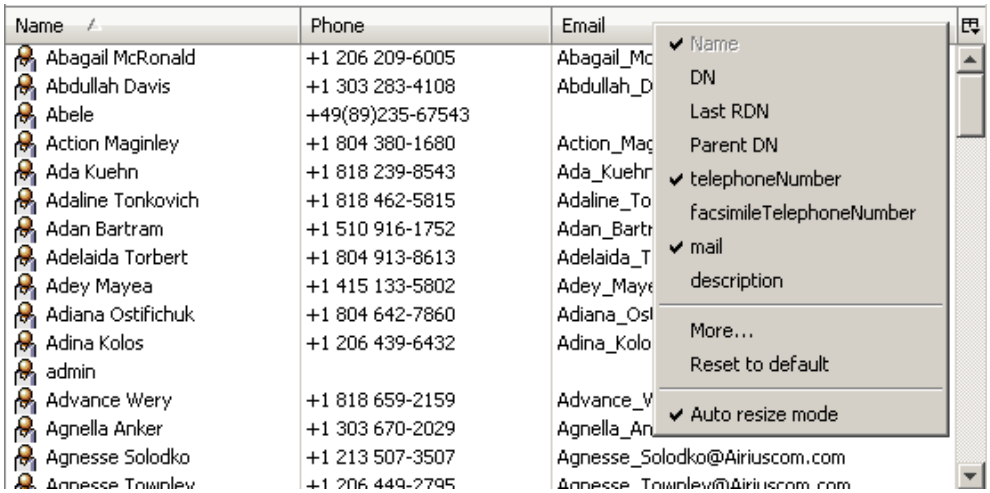


Figure 99. Context-sensitive Menu for Column Headers of Lists

If the number of possible columns exceeds a certain limit, the option "**More...**" is offered. This option leads to a dialog that offers the complete choice available for the current list.

Recurring (=multi-valued) attributes display no more than the 10 values.

Note that the attribute value(s) can also be viewed thru the related tool tip.

The "**Auto resize mode**" automatically resizes all other columns, if you resize one column. If switched off, only the column to the left of your mouse cursor resizes.

7.3. Positioning

A significant number of GUI elements present large enumerations of items, like entries in tree or list panes or elements of combo boxes. The larger the list, the more tedious it can be to scroll down or up to a particular position. In these cases, you can use this application's positioning abilities to make it easier to handle these enumerations: in tree panes, in list panes and in many non-editable combo boxes, you can position just by typing in one or more characters. A tool tip briefly indicates your typing. When using positioning:

- Make sure the GUI element in question has the focus.
- Type, for example,. "j" to position to the first item beginning with "j". Repeatedly typing "j" will position to the subsequent items starting with "jjj.." .
- Type, for example, "joe" to position to the first item beginning with "joe". Don't pause when typing "joe". Otherwise you will position to "j" then to "o" and finally to "e".

7.3.1. Abandoning an In-process, yet Uncompleted Operation

Searching

There are several ways to initiate an abandon operation, depending on the context:

- By clicking in the Toolbar:

- By clicking Stop search
  Note that the Search now button may automatically change to Stop search each time a search is initiated.

Browsing

Browsing the tree can be abandoned by clicking in the Toolbar:

Properties

Reading the properties of an entry is usually not an operation that lasts for too long. However, there are exceptions:

- Network problems or server overload may occasionally prevent the server from reacting as usual.
- Bulky entries (for example, entries having recurring attributes with numerous values) may take a while to get transferred.

For those reasons, you can abandon read property operations by clicking "Cancel" in a dialog-box displayed. Note that this dialog-box is not displayed if the server returns the result rather quickly; and it automatically disappears as soon as the result arrives.

7.3.2. View Panes

View panes are the major building blocks for defining the overall appearance of this application. Their arrangement is defined in a number of configuration files.

Plug-ins may include their own view panes.

This application’s core component provides a number of view panes (note that views normally use only varying subsets of the panes provided), including:

- The tree pane
- The search pane (closely related to the search dialog)
- The list pane
- The simple list pane
- The property pane (closely related to the property dialog)
- Container panes, which organize the layout of other panes and do not display any data. The core component provides the following container panes:
- The border pane
- The split pane
- The tabbed pane
- The titled pane

7.3.2.1. Tree Pane

The tree pane allows you to browse hierarchical structures contained in the LDAP directory in a tree-like manner. The tree view is typically accompanied by a list pane or a property pane or a combination of both.

Here is an example of a tree pane:

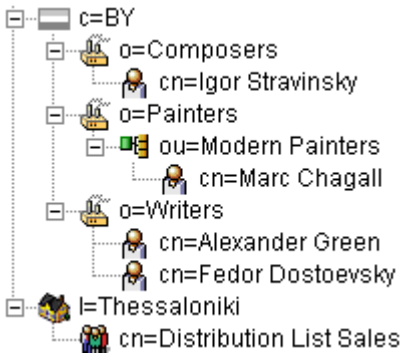


Figure 100. Tree Pane

If the server has implemented operational attributes such as "numSubordinates" and "numAllSubordiates" and if access control grants read access to those attributes, the tree pane may look like this:

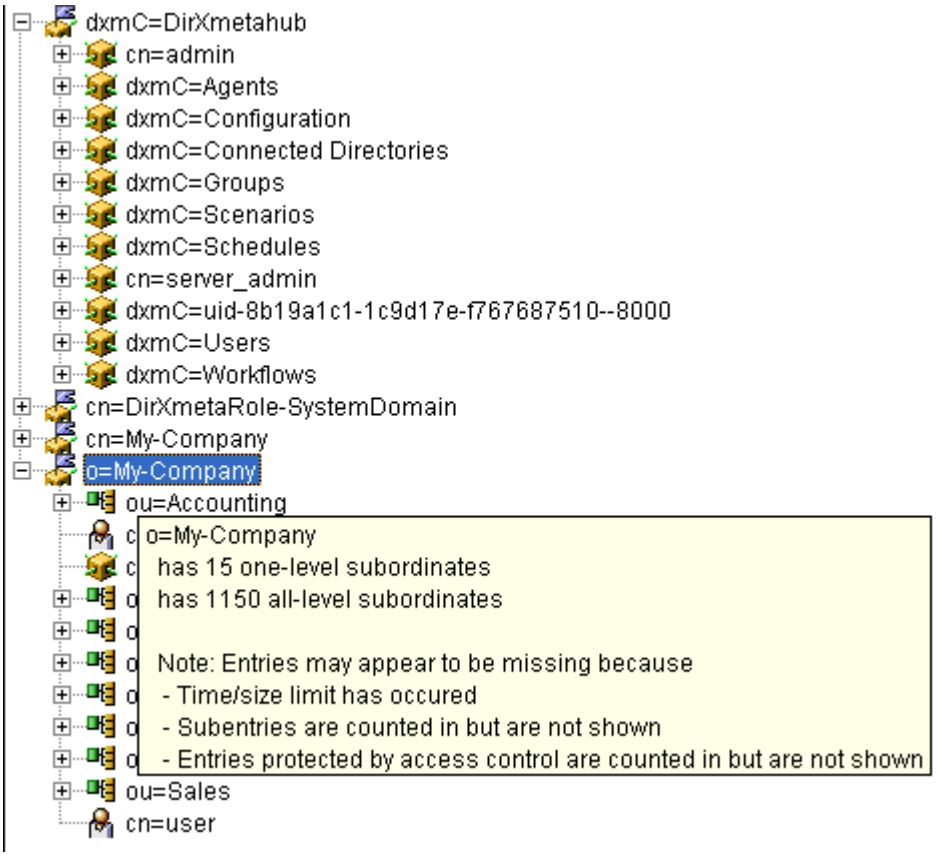


Figure 101. Tree Pane Displaying Number of Subordinates and Number of All Subordinates

- Some entries (like cn=admin in the example above) do not show ⊞ in the first place, since the server tells that there are no subordinates (do a refresh, if you expect that one or more entries have been added below)

- Tool tips show how many direct subordinates and how many overall subordinates an entry has
- A sort of a pseudo entry may indicate in red that not all entries are available (which typically happens due to a time or size limit or due to access control restrictions)

In combination with searching, you can use the tree for determining a search base - depending on how it is configured - either automatically or by clicking a button located next to the search base field.

To make a **multiple selection**:

- Click the first entry, press the *shift* key and click the last entry while keeping the *shift* key pressed.
- Click any - not necessarily adjacent - entries while keeping the *ctrl* key pressed

Click a selected entry while keeping the *ctrl* key pressed to **de-select** it.

A problem of “flat” trees is that size or time limit typically prevents all "children" from being displayed. On the other hand, with unlimited size and time, the number of entries returned by the LDAP server may just be too big to be handled reasonably. This problem can be approached in several ways:

- There is a search function available on the right mouse button
- There is a filter function available on the right mouse button

7.3.2.2. Search Pane

The search pane allows you to specify a search filter and initiate a search causing the server to return *all* entries (or a *subset* if there are too many) that match your search filter.

Here is an example of a search pane and a Tree (=Browse) pane combined through a tabbed pane:

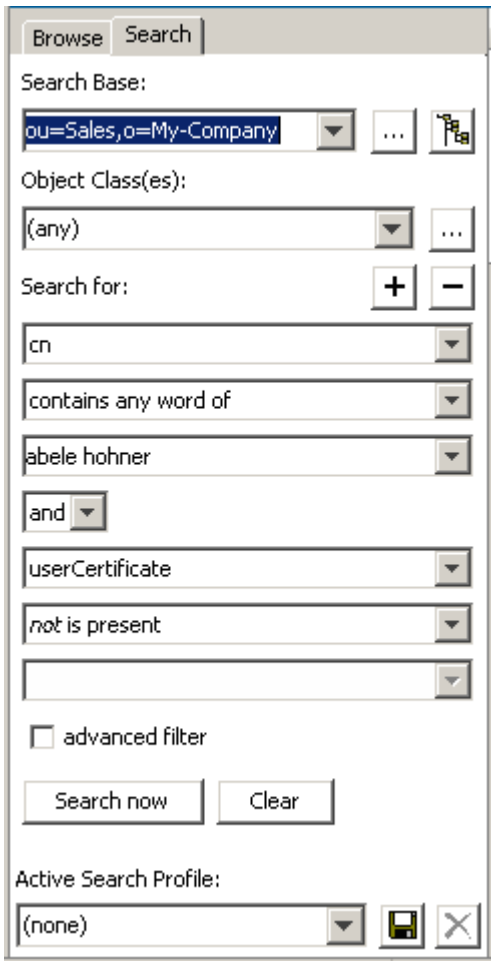


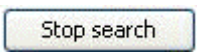
Figure 102. Combined Search Pane Tab

This example shows three panes:

- A search pane
  If the search pane is combined with a tree pane (like in this example), a special feature is available: you can take over the search base from the currently selected entry in the tree pane by clicking the button  ("Get search base form tree"). Alternatively, the search base may be configured to take the current selection automatically from the related tree pane. In any case, you can overwrite this by clicking the button  (the one to the left of the  button); this button opens a window offering to select a value from a tree. The optional server selection field is not configured.
  Read about the other search options in section Standard Dialogs: "Searching".
- A tabbed pane, which allows you to switch between browsing and searching
- A tree pane (indicated)

Here is an example of a "slimmed down" search pane:

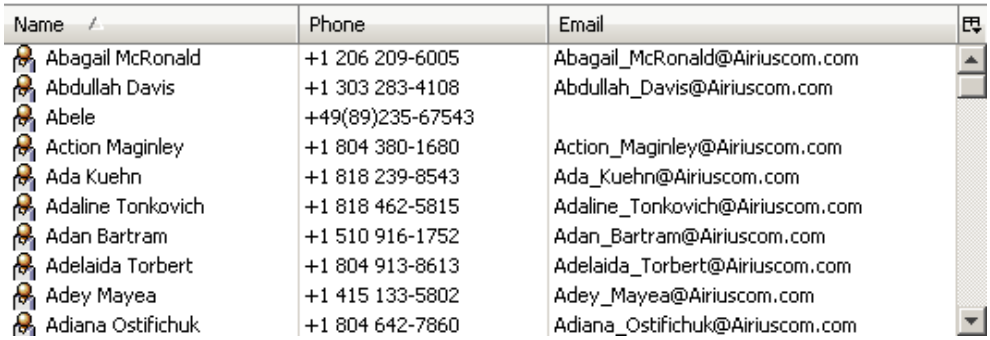
Figure 103. Slimmed Down Search Pane

Note that the button  may automatically change to  each time a search is initiated.

7.3.2.3. List Pane

The list pane allows having entries (typically search results), more detailed: your choice of their attributes, listed in a table and sorted by the attribute(s) of your choice.

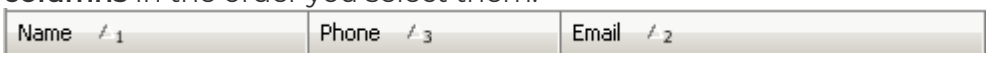
Here is an example of a list pane:



Name	Phone	Email
Abagail McRonald	+1 206 209-6005	Abagail_McRonald@Airiuscom.com
Abdullah Davis	+1 303 283-4108	Abdullah_Davis@Airiuscom.com
Abele	+49(89)235-67543	
Action Maginley	+1 804 380-1680	Action_Maginley@Airiuscom.com
Ada Kuehn	+1 818 239-8543	Ada_Kuehn@Airiuscom.com
Adaline Tonkovich	+1 818 462-5815	Adaline_Tonkovich@Airiuscom.com
Adan Bartram	+1 510 916-1752	Adan_Bartram@Airiuscom.com
Adelaida Torbert	+1 804 913-8613	Adelaida_Torbert@Airiuscom.com
Adey Mayea	+1 415 133-5802	Adey_Mayea@Airiuscom.com
Adiana Ostifichuk	+1 804 642-7860	Adiana_Ostifichuk@Airiuscom.com

Figure 104. List Pane

- Double-click an entry to **display** a **details** dialog for the entry.
- Click an entry in the list with the left mouse button and (without releasing it) draw it downwards or upwards in order to make a **multiple selection**. This way of making a multiple selection is not available in the tree pane. Other ways (that are available in the tree pane, too) of making a multiple selection include:
- Click the first entry, press the *shift* key and click the last entry while keeping the *shift* key pressed.
- Click any - not necessarily adjacent - entries while keeping the *ctrl* key pressed.
- Click a selected entry while keeping the *ctrl* key pressed to **de-select** it.
- Right-click a column header to redefine what columns are to be displayed.
- Click a column header to **sort the list** according to that column.
- Click several column headers while pressing the *shift* or *ctrl* key to **sort by multiple columns** in the order you select them:



Name ₁	Phone ₃	Email ₂

In the example shown here, the list is sorted by three columns in this priority:

1. Priority: Name

5. Priority: Email
6. Priority: Phone

7.3.2.4. Simple List Pane

A simple list pane is like a list pane, but has only one column.

Here is an example of a simple list pane:

Figure 105. Simple List Pane

7.3.2.5. Property Pane

The property pane allows you to read and edit the properties of an entry.
Double-click an entry to display a property ***dialog*** (as opposed to a property ***pane*** that is embedded into the main window: A property dialog is a separate window that you can resize up to the full screen).

The property pane usually has two modes (read and edit). In read mode, you cannot edit any data.

The clipboard functions cut/copy/paste (cut and paste only in edit mode) are available on a per attribute value basis. These functions also work between different instances of this application. Note that (in tree and list panes) clipboard (and drag and drop) functions are also available for currently selected entries.

Here is an example of a property pane in read mode:

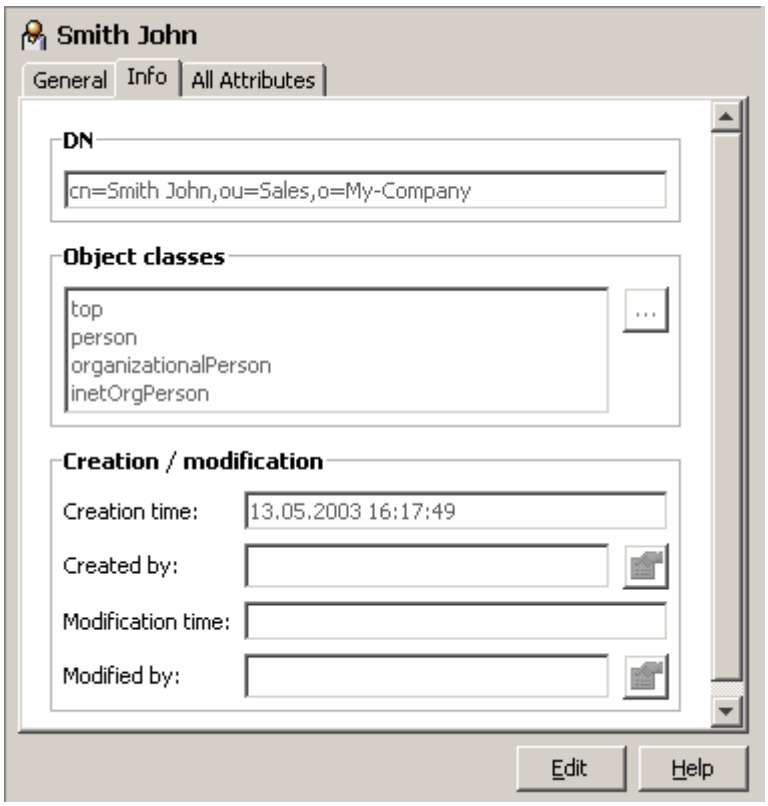


Figure 106. Property Pane in Read Mode

For information about the buttons shown on the right-hand side in the figure, see the topic Property Editors: Summary. This topic also describes how to view property values that do not fit into the value field.

Click **Edit** to change to edit mode. Click **Save** or **Reset** (these buttons appear when you click **Edit**) to change back to read mode. The property pane may change its appearance significantly between edit mode and read mode.

Read-only mode: 

(you cannot change to edit mode)

Read mode: 

Edit mode: 

Note that in edit mode the rest of the GUI is blocked.

Note that plug-ins may add additional buttons here.

Here is an example of a property pane in edit mode:

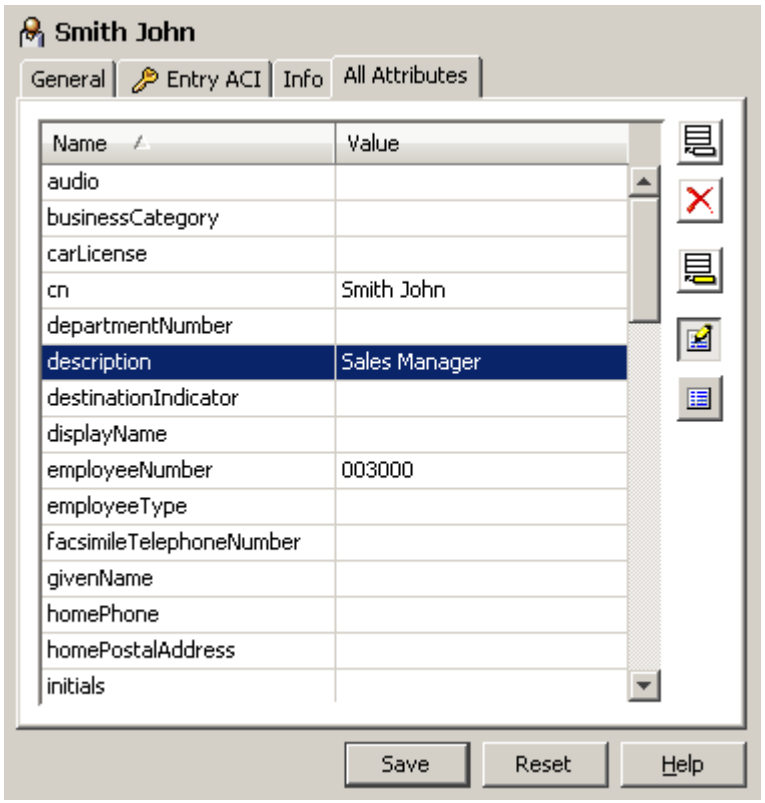


Figure 107. Property Pane in Edit Mode

For Password modifications, see also "Reset Password".

7.3.2.6. Container Panes

Container panes, which organize the layout of other panes and do not display any data. The core component provides the following container panes:

- Border
- Split
- Tabbed
- Titled

7.3.2.6.1. Border Pane

Border panes are used when the size of a pane is to dynamically extend/shorten, as soon as one or more other panes need less/more room due to some action performed on them.

Here is an example of the use of a border pane:

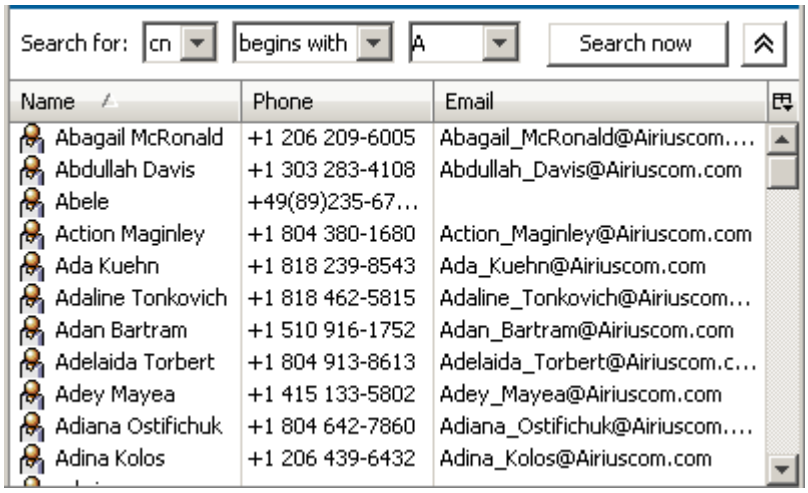


Figure 108. Border Pane with Invisible Part

In this example, the border pane (which is invisible) separates a search pane ("Search for ...") from a list pane so that the space that the search pane is to occupy is minimized. The next screen shot shows what happens when you fade-in the search base by clicking the button  : A new line appears in the search pane causing it to expand automatically just as much as needed, while the list pane automatically shrinks accordingly.

Clicking the button again (in this situation it looks like ) makes the search base disappear, the size of the search pane automatically shrinks while the size of the list pane expands; that is, the layout returns to the one previously shown.

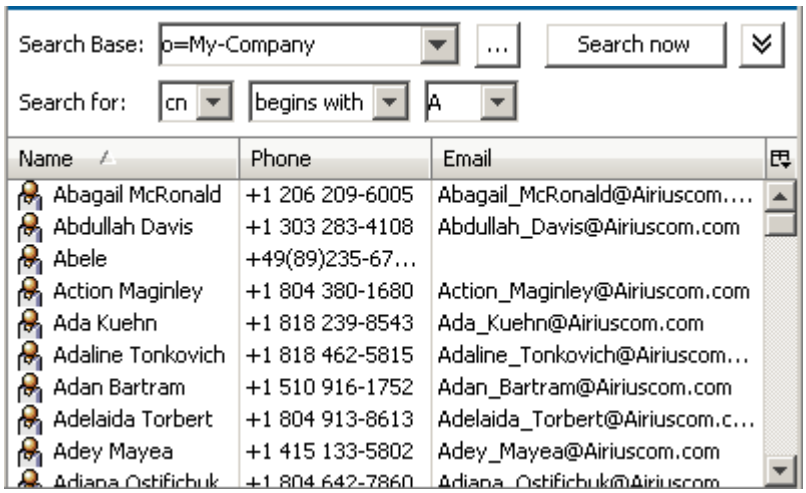


Figure 109. Expanded Border Pane

7.3.2.6.2. Split Pane

A split pane separates two other panes and allows you to increase the size of one pane at the expense of the size of the other one.

Here is an example of a split pane:

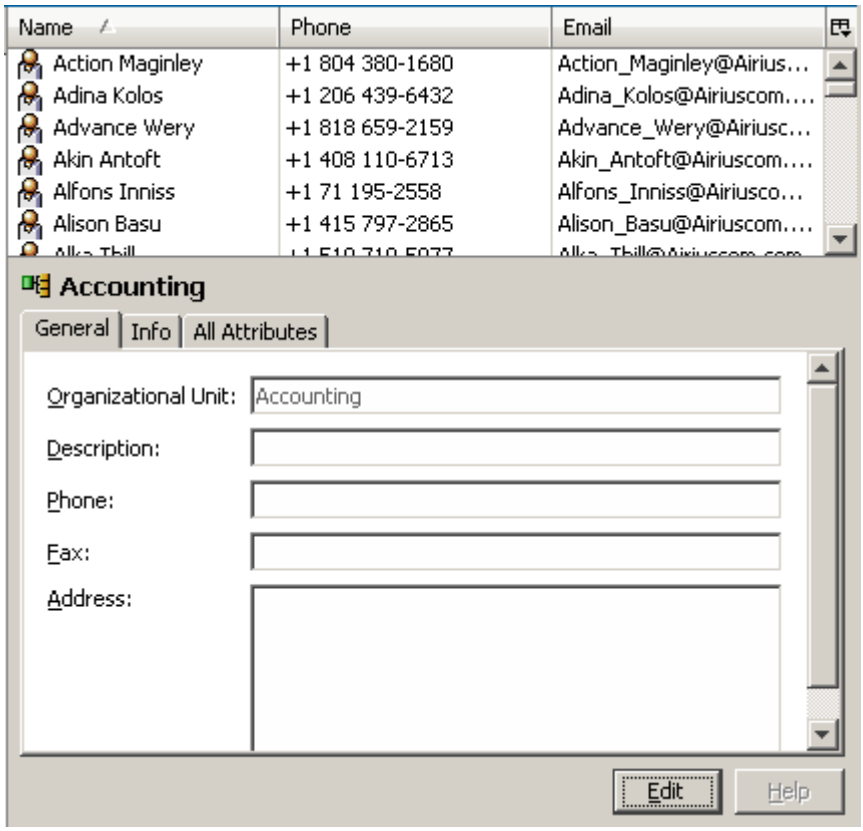


Figure 110. Split Pane with List Pane and Property Pane

This example shows three panes:

- Above: a list pane
- In the middle: the split pane (the horizontal bar)

If the mouse button hovers above a split pane, it changes its appearance like this ↕ or like this ↔. If you left-click the split pane with this appearance of the cursor, you can resize the related panes (enlarging one at the cost of the other one).

- Below: a property pane

7.3.2.6.3. Tabbed Pane

A tabbed pane allows you to switch between two or more other panes thru clicking tabs.

Here is an example of a tabbed pane:

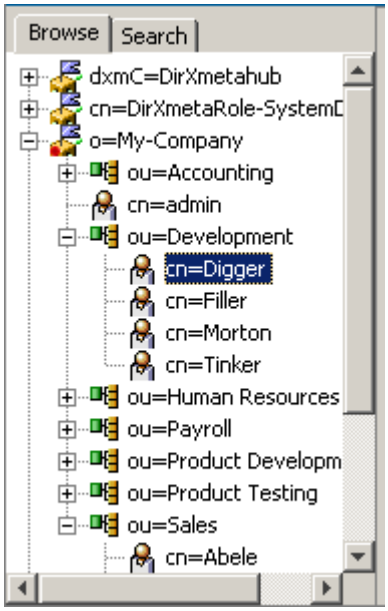


Figure 111. Tabbed Pane

This example shows three panes:

- On top: a tabbed pane
  The tabbed pane allows you to switch between two or more different panes by clicking the tab for the pane you want to display.
- Below: a tree pane
- Another pane (indicated)
  Clicking the search tab causes the search pane to be displayed while the tree pane currently on display is hidden.

7.3.2.6.4. Titled Pane

A titled pane adds a title to another pane, usually on top of the other pane.

Here is an example of a titled pane:

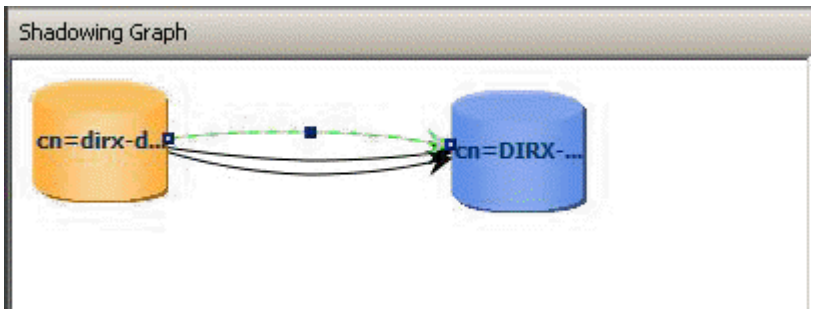


Figure 112. Titled Pane

This example shows two panes:

- On top: a titled pane: "Shadowing Graph"
  Titled panes serve just for descriptive purposes.
- Below: the shadowing graph of a shadowing agreement.

7.3.3. Property Editors

When you click a property value in the edit mode of a property dialog or a property pane, this application will show one of the following responses:

- No response at all, because the property cannot be changed this way.
  Here are some examples:
- DN (Distinguished Name)
  Use the Rename and/or MoveTo function to change a distinguished name
- cn (Common Name)
  Use the Rename function to change a common name
- CreateTimeStamp; CollectiveTelephoneNumber
  Changing operational or collective attributes is usually not possible
- The affected property becomes editable. The core component provides property editors for the following attribute types:
- Attribute with DN Syntax
- Boolean
- Country String
- **The default**: Directory String
- Generalized Time
- Fax Number
- IA5
- Integer
- Jpeg Photo
- Numeric String
- Object Class
- Postal Address

- Printable String
- Telephone Number
- User Certificate
- User Password

There is a number useful buttons located on the right-hand side of the property dialogs, as shown in the following figure (also indicated are the operations available at the right mouse key):

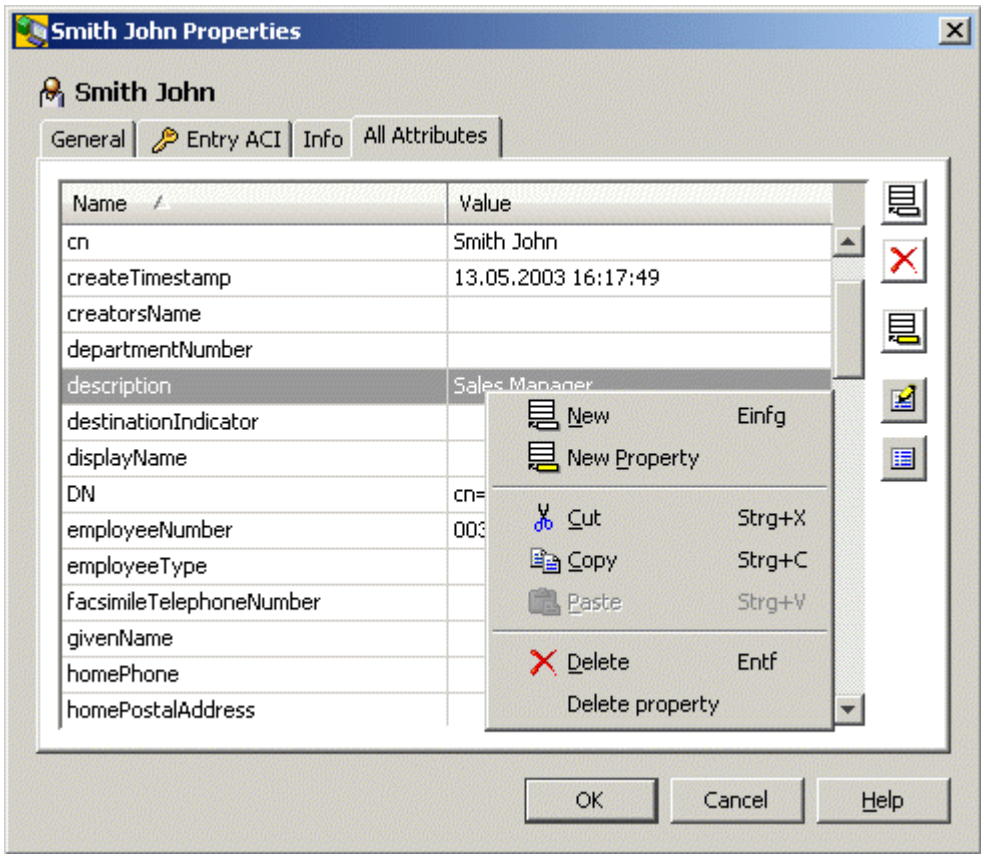


Figure 113. Properties dialog box with All Attributes Tab for Smith John

Use these buttons to:

- Insert a value  (also available through the right mouse button). This function may not be present, depending on what type of node is currently selected.
- Delete the selected property value  (also available through the right mouse button)
- Create a new property  (also available through the right mouse button)
- Display only editable properties 
- Exclude empty properties from being displayed 

If a property value is too long to be displayed in the visible part of the display field, you can resize the column or the entire window, or just position the cursor in the value field and wait for the tool tip to appear:

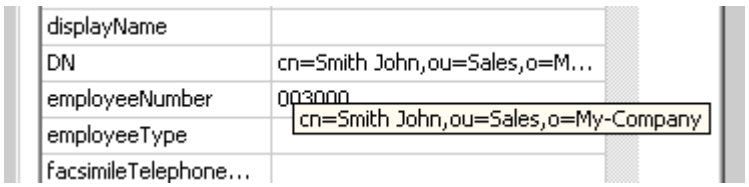


Figure 114. Tool Tip for Long Value

Note that the tooltip will be on display only for a while. Alternatively, properties with directory syntax typically offer to view/edit lengthy values in a re-sizeable dialog.

7.3.3.1. Attribute with DN Syntax

The attribute with DN syntax editor allows you to assign a DN to an attribute with DN syntax:

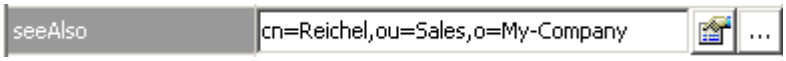


Figure 115. DN Syntax Editor for the seeAlso Attribute

Click "  " to open a dialog that offers searching and browsing facilities for the DN of interest. Alternatively, you can just type the DN into the value field.

Click "  " to display the referenced entry.

7.3.3.2. Boolean

The boolean attribute editor allows you to edit boolean attributes. A boolean attribute shows one of the following appearances:

Figure 116. Boolean Editor with Value FALSE

Figure 117. Boolean Editor with Value TRUE

7.3.3.3. Country String

The country string editor is a combo box like:

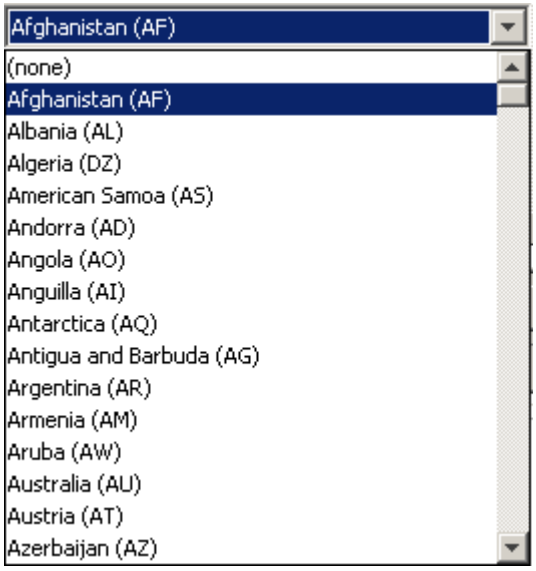


Figure 118. Country String Editor

7.3.3.4. Directory String

The directory string property editor is the default editor. It applies to attributes with syntax Directory String and to all attributes with a syntax that does not have a corresponding dedicated property editor.

This editor provides a simple editable text field. In order to make reading and editing lengthy directory strings more convenient, the context menu offered through the right mouse button typically includes an "Edit" function that pens a re-sizeable dialog:

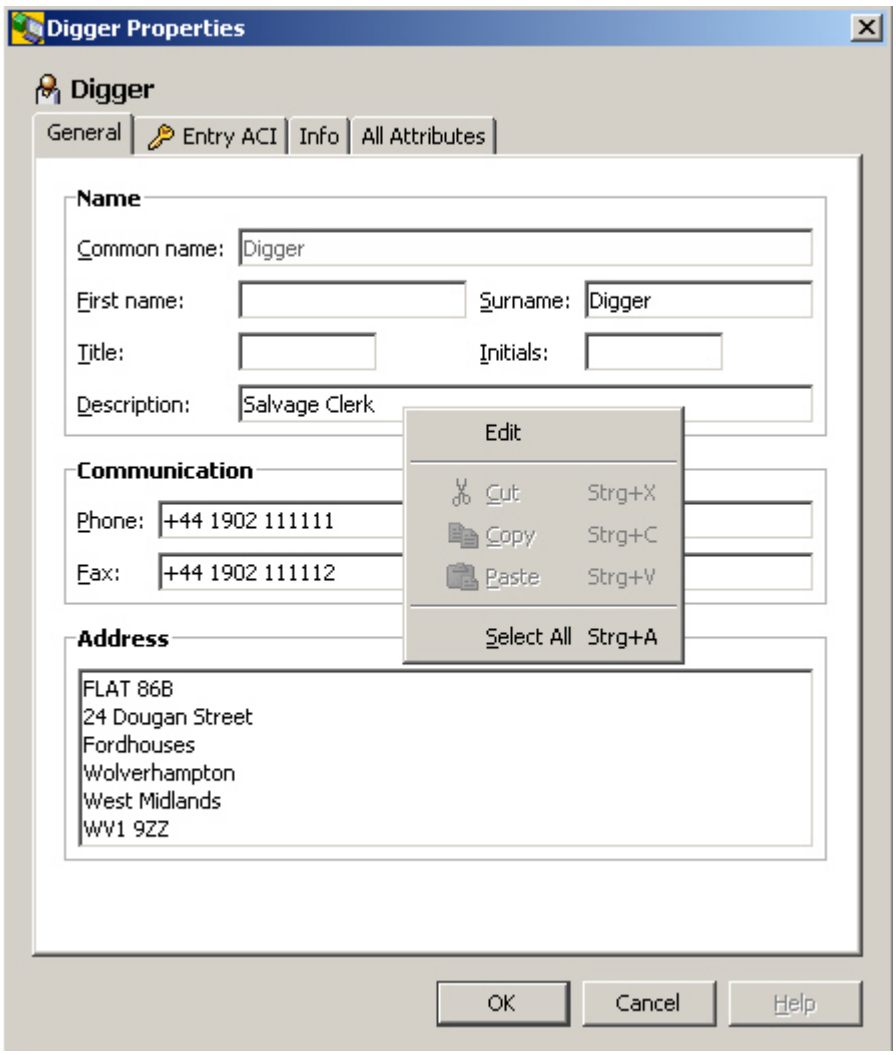


Figure 119. Digger Properties Dialog Box General Tab

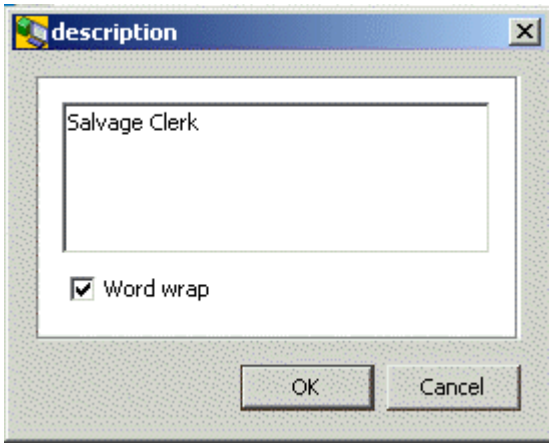


Figure 120. Directory String Editor Edit Operation

7.3.3.5. Generalized Time

The generalized time editor allows you to edit the date and (local) time.

Figure 121. Generalized Time Editor

Click  to use your mouse to edit the date/time. Alternatively, you can just type the date/time in the value field.

The line that allows you to edit the time is not available under all conditions.

7.3.3.6. IA5

The IA5 editor provides a simple editable text field that is restricted to the character set defined by IA5 (http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-T.50).

Hex 20 to hex 7E, including “@”.

7.3.3.7. Integer

The integer editor provides a simple editable text field that does not take any characters other than digits that comply with a number in the range 0 to 2^{32} -1 (=4294967295).

7.3.3.8. Jpeg Photo

Use the Jpeg editor to perform the following actions on JPEG images:

- View 
- Export 
- Import 
- Delete 

Here is an example of a Jpeg editor:

Figure : Jpeg Editor

7.3.3.9. Numeric String

The numeric string editor provides a simple editable text field that does not take any characters other than digits (0 to 9) and space.

7.3.3.10. Object Class

The object class editor allows you to edit object classes:

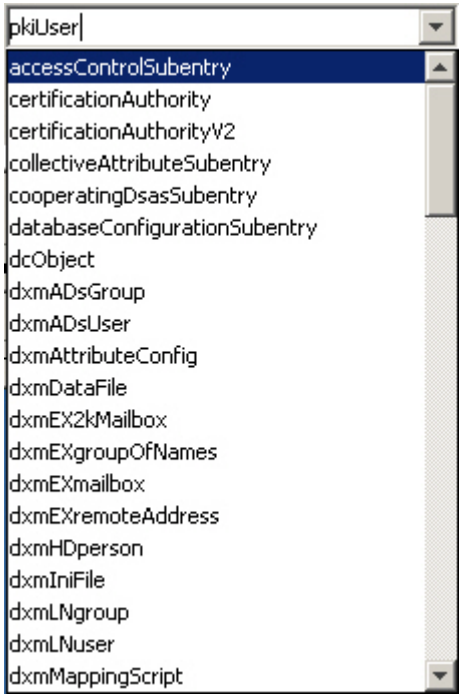


Figure 122. Object Class Editor

Note that you cannot add, delete or change *structural* object classes; only *auxiliary* object classes can be added, deleted or modified.

7.3.3.11. Phone/Fax

The phone/fax editor allows you to edit a phone/fax attribute. Click the value field to change it to *inline* edit mode. Right-click the value field and select "check number..." to have a dialog displayed like the one shown in the following figure:

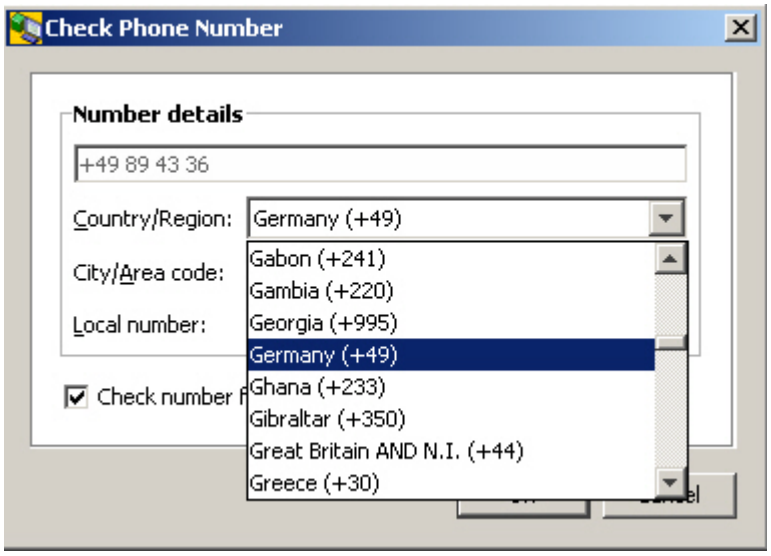


Figure 123. Phone/Fax Editor Check Phone Number Dialog Box

7.3.3.12. Postal Address

The postal address editor allows you to edit a Postal Address (6 lines à 30 characters):

Figure 124. Postal Address Editor

7.3.3.13. Printable String

The printable string editor provides a simple editable text field that is restricted to the character set known as printable string (http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-X.208):

A to Z, a to z, 9-0, <space>, ')(+,-./:=?

This editor cannot be applied to email addresses, because the "at" character (@) is not available.

7.3.3.14. User Certificate

Use the user certificate editor to perform the following actions on user certificates:

- View 
- Import 
- Export 
- Delete 

Here is an example of a user certificate editor:

Figure 125. User Certificate Editor

When clicking the view button  the Certificate Details are displayed:

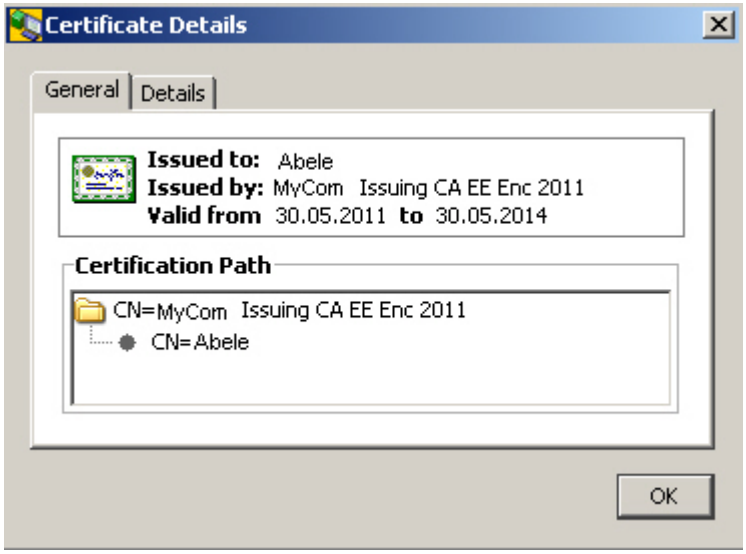


Figure 126. Certificate Details Dialog Box

7.3.3.15. User Password

The user password editor allows you to change your own or someone else's password (depending on the selected entry). Click the value field to display the editor, which is shown in the following figure :

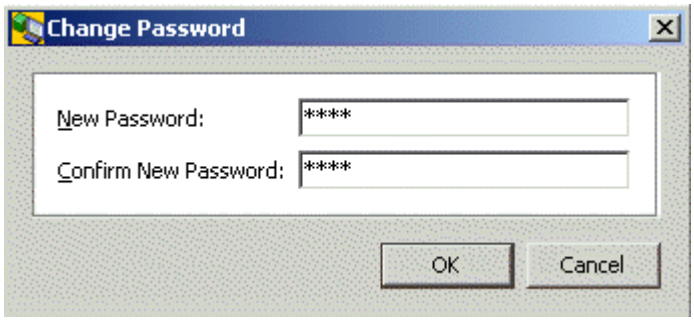


Figure 127. User Password Editor

Deleting a password works the same way as deleting any other property.

Note that multivalued user passwords are not supported by this application.

See also: Reset Password.

7.3.4. Standard Dialogs

The core component provides the following standard dialogs:

- Binary Attributes
- Changing Your Own Password
- Choosing a Distinguished Name
- Exporting
- Importing
- Login
- Naming
- Properties
- Renaming
- Searching
- Server

7.3.4.1. Binary Attributes

Binary attributes other than certificates and jpeg photos are indicated in the all attributes tab like this ("audio"):

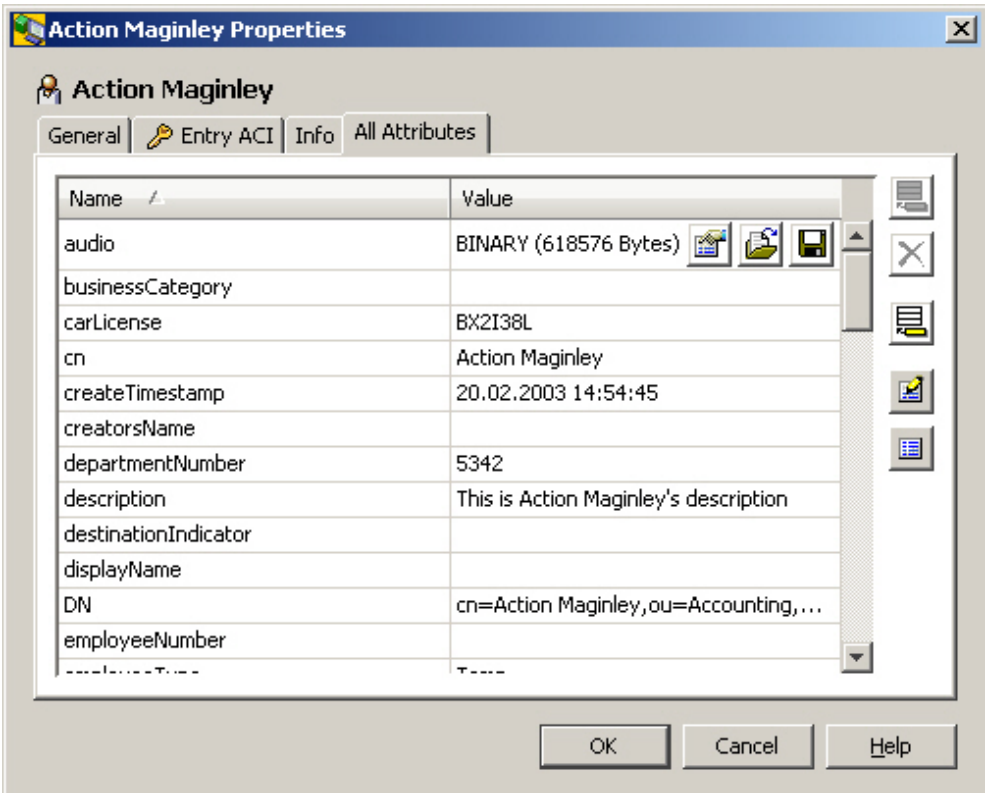


Figure 128. Properties dialog box with All Attributes Tab and binary audio value for Action Maginley

Clicking the  button allows to save the currently selected value in a file.

Clicking the  button allows to import a binary value from a file.

Clicking the  button displays a window that shows the value as hexdump:

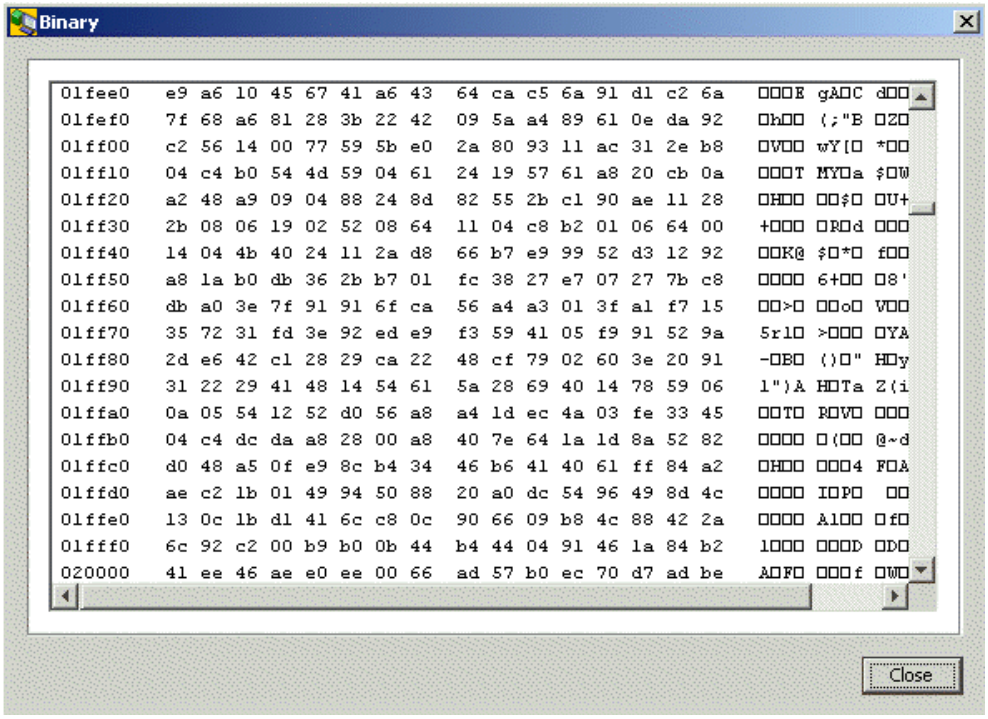


Figure 129. Binary Value Window

7.3.4.2. Changing Your Own Password

(To change somebody else's password, see the topic User Password).

This dialog allows you to change your own password; that is, the password that was used in the previous Login.

Here is an example:

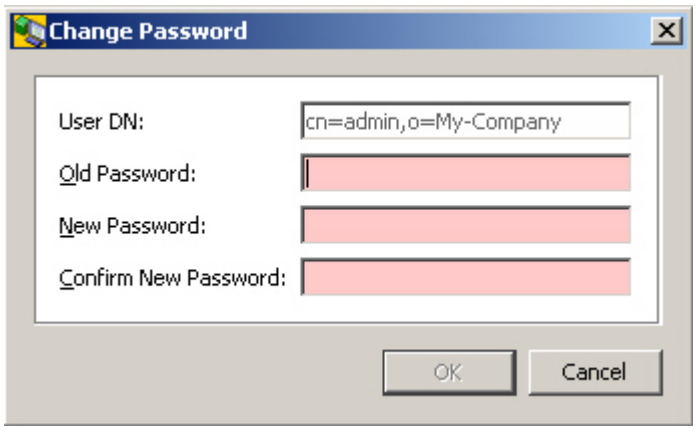


Figure 130. Change Password Dialog Box

Note that multivalued user passwords are not supported by this application.

7.3.4.3. Choosing a Distinguished Name

This dialog allows you to choose a distinguished name that contains all valid values, either by browsing a tree or by searching. Here is an example:

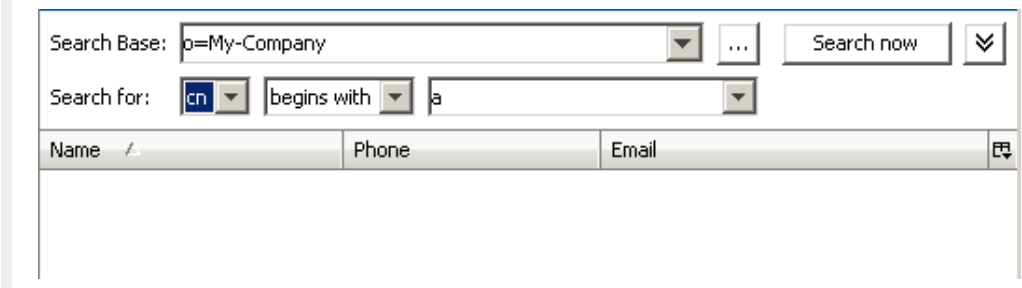


Figure : Choosing a Distinguished Name Dialog

7.3.4.3.1. Exporting

The Export dialog allows you to specify details regarding the current export, including the file format for the export, the name of the export file, and the entries to be exported. Here is an example of the Export dialog:

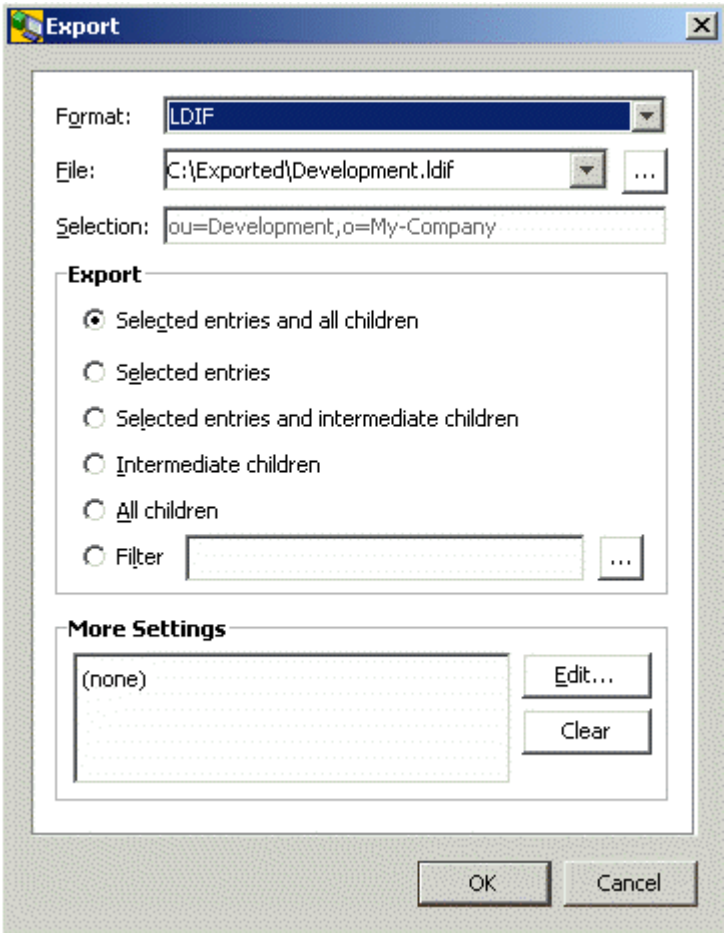


Figure 131. Export Dialog Box

More settings are available through the "Advanced Export Options" dialog. This dialog is composed of two tabs: the General tab and the Attribute Restriction tab. Here is an example of the General tab.

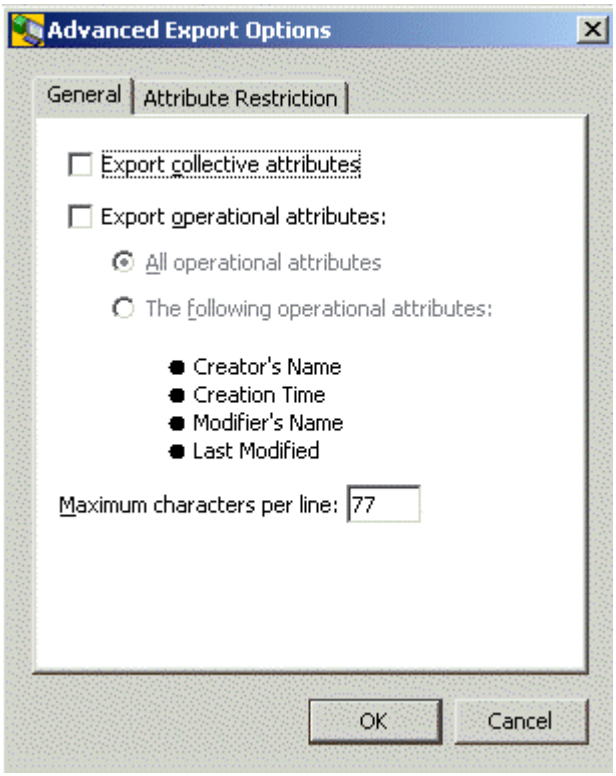


Figure 132. Advanced Export Options Dialog with General Tab

The available selections are:

- Export collective attributes
- Export all operational attributes
  Note that, apart from access control restrictions, the server may decide not to return certain operational attributes, e.g. operational attributes that are expensive to locate. Some servers do not support this option at all.
- Export a named set of operational attributes
- Limit the number of characters per line (default is 77)

Here is an example of the Attribute Restrictions dialog:

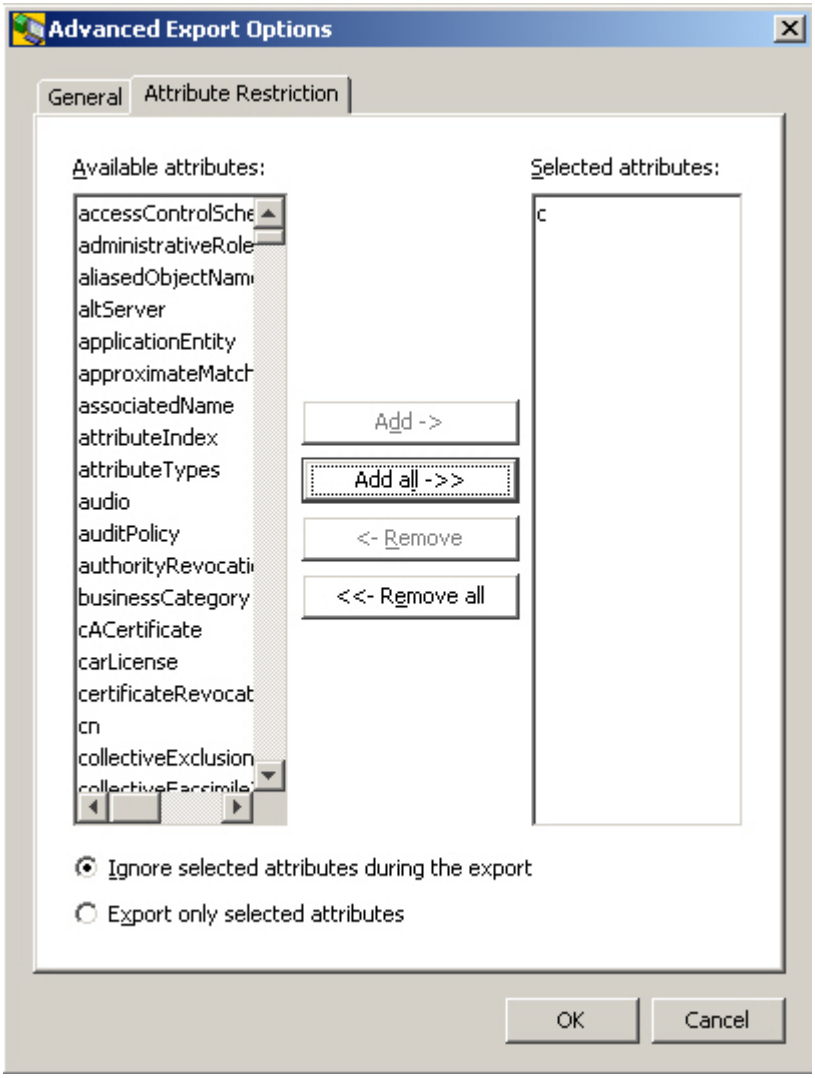


Figure 133. Advanced Export Options Dialog with Attribute Restriction Tab

7.3.4.4. Importing

This dialog allows you to specify details regarding the current import:

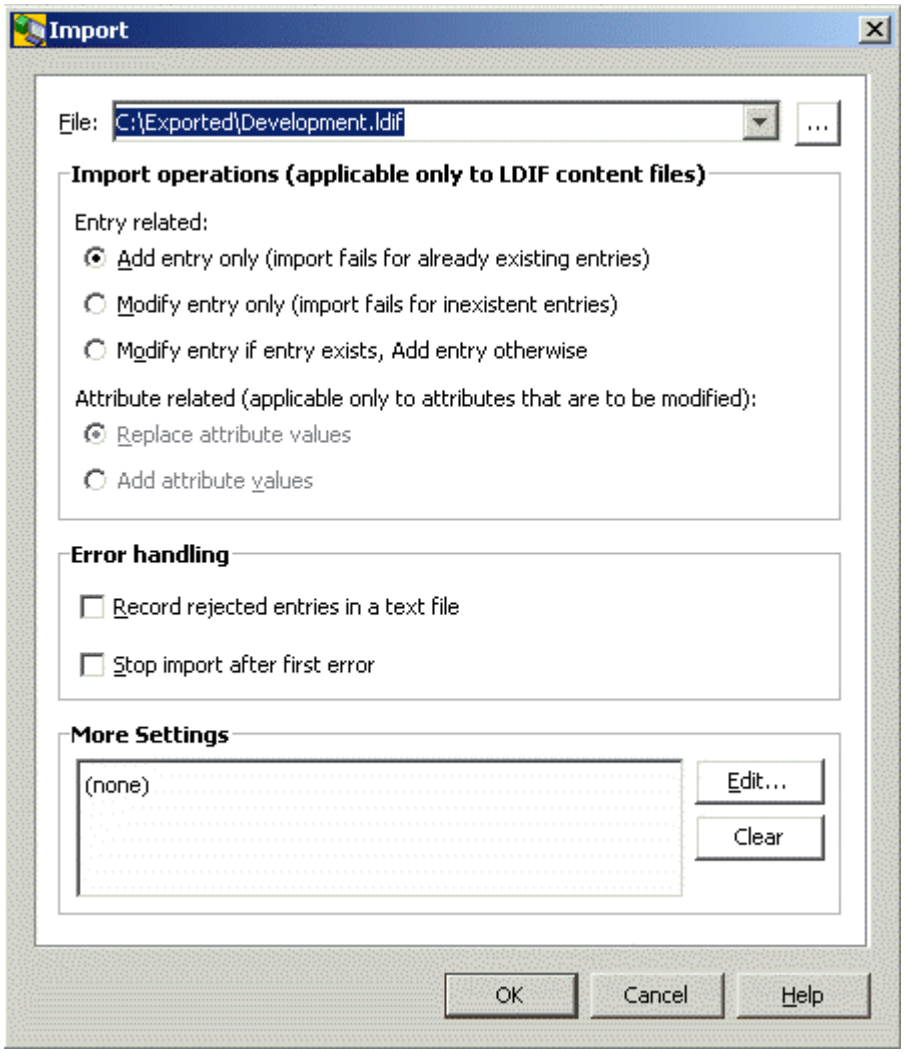


Figure 134. Import Dialog Box

- The radio buttons "**Add entry only**", "**Modify entry only**", "**Modify entry if entry exists**, **Add entry otherwise**" apply to *LDIF content* files only and are ignored otherwise (if the file is recognized as a DMSL v1 file, which is sort of a DMSL content file, only **Add entry** is possible). In case of ***modify entry***, the affected attribute values, as they exist in the servers' database, will be replaced by the ones found in the file. Attributes that are missing in the import file do not affect the corresponding attributes in the server's database at all, particularly they do not cause those attributes to be deleted.
- The radio buttons **Replace attribute values** and **Add attribute values** apply only to entries that are to be modified. **Add attribute values** fails with error
- For attributes that are single-valued and have already a value in the server's database
- For attributes that are multi-valued and the set of values possibly stored already in the server's database and the set of values found in the import file are *not* disjoint.
- Check **Record rejected entries in a text file** if you want to save rejected entries together with the associated error message in an LDIF file. The error messages pre-pend the respective entries in the form of comment lines. Just before it completes, the import function pops up a file selection dialog if there is at least one erroneous entry. If you do not check off this check button, erroneous entries are only displayed in the progress dialog.

- Check **Stop import after first error**, if you want to have the import operation discontinue as soon as the first error occurs.
- More settings are available through the push button More settings...
- A progress dialog like the one depicted below displays the DNs of the rejected entries and the belonging error messages and allows you to cancel the import any time.

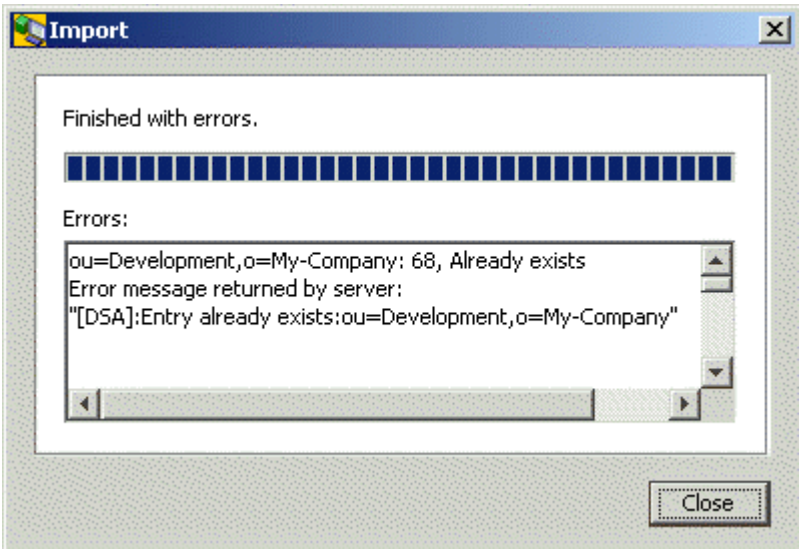


Figure 135. Import Progress Dialog Box

Note: If the file that is to be imported does not appear to be LDIF or DSML compliant, the import will be refused with an error message. There is one exception: multi-line attributes in LDIF files must be base64 encoded; however, they are also accepted here, if they appear as text with the individual lines being separated by hex "01".

7.3.4.4.1. More Import Settings

This dialog allows you to restrict the import operation to a specified set of attributes (the ones that appear or - depending on the radio button at the bottom of the dialog - do *not* appear in the **Selected attributes** field)

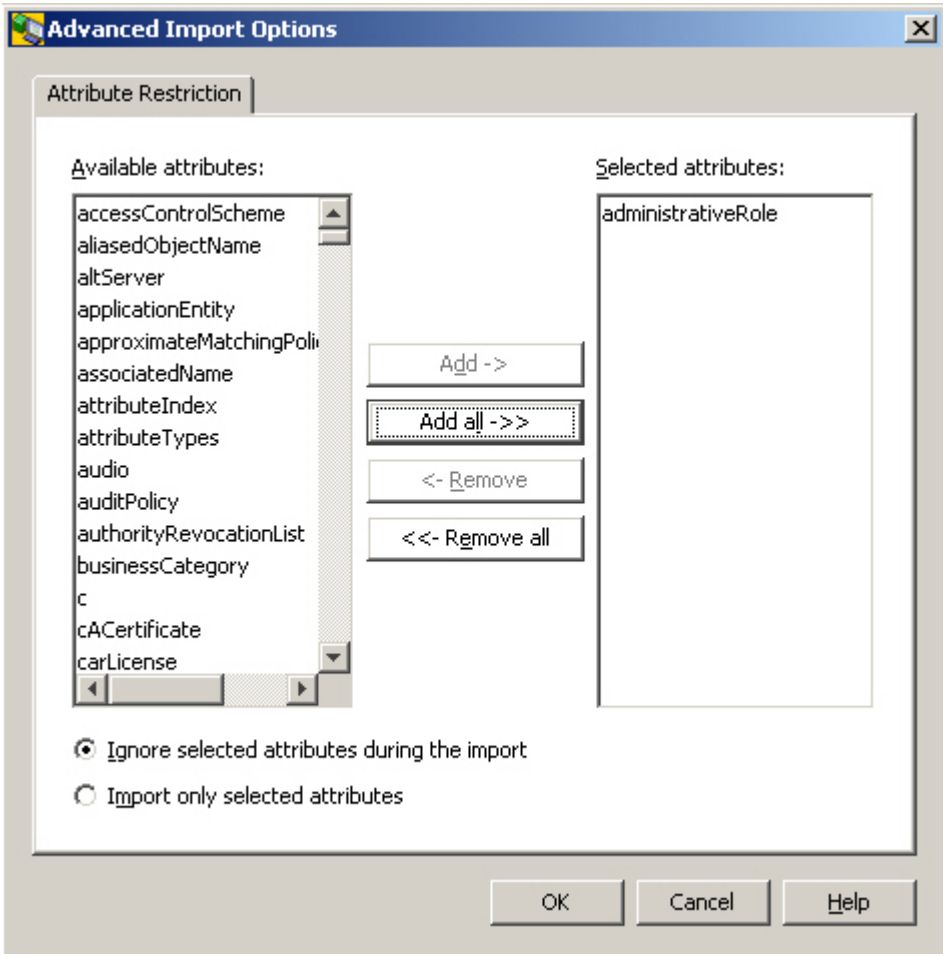


Figure 136. Advanced Import Options Dialog with Attribute Restriction Tab

7.3.4.5. Login

The login dialog allows you to exchange authentication information between you and the LDAP server.

Here are examples for the Login dialog:

- Performing a simple authenticated or anonymous bind:

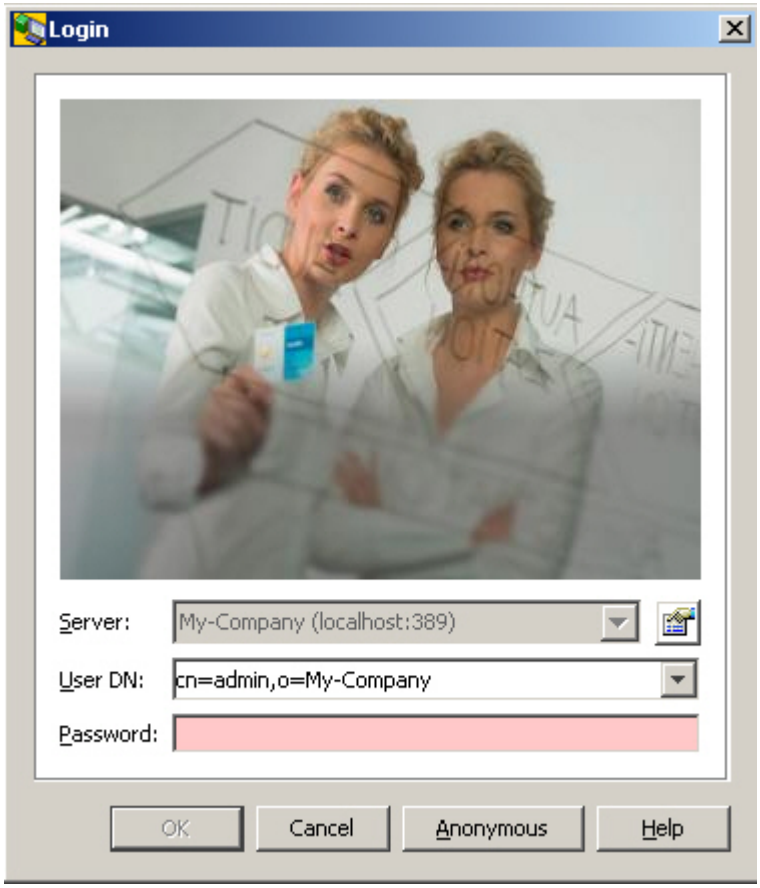


Figure 137. Login Dialog Box for Simple Authenticated or Anonymous Bind

- Performing a smart card login:

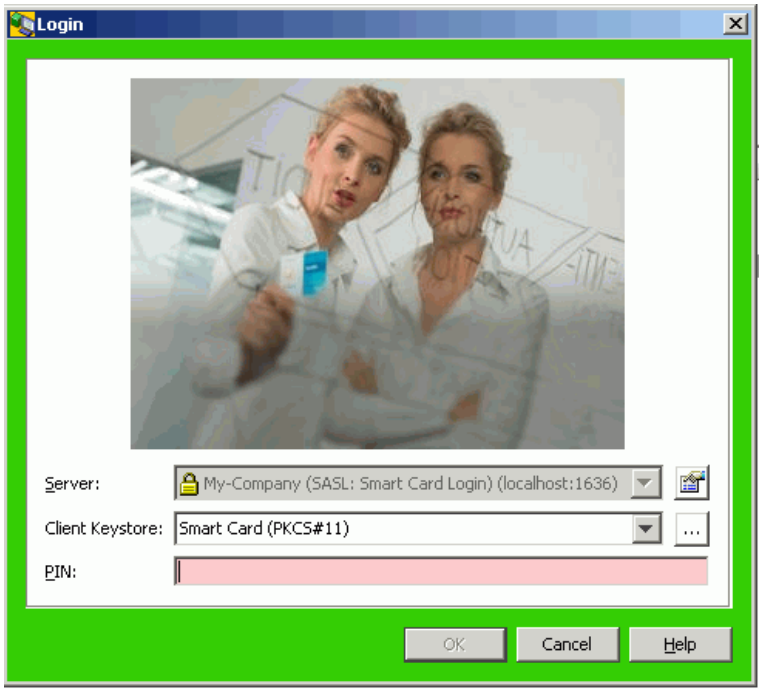


Figure 138. Login Dialog Box for Smart Card Login

- Performing a file based SASL EXTERNAL bind:

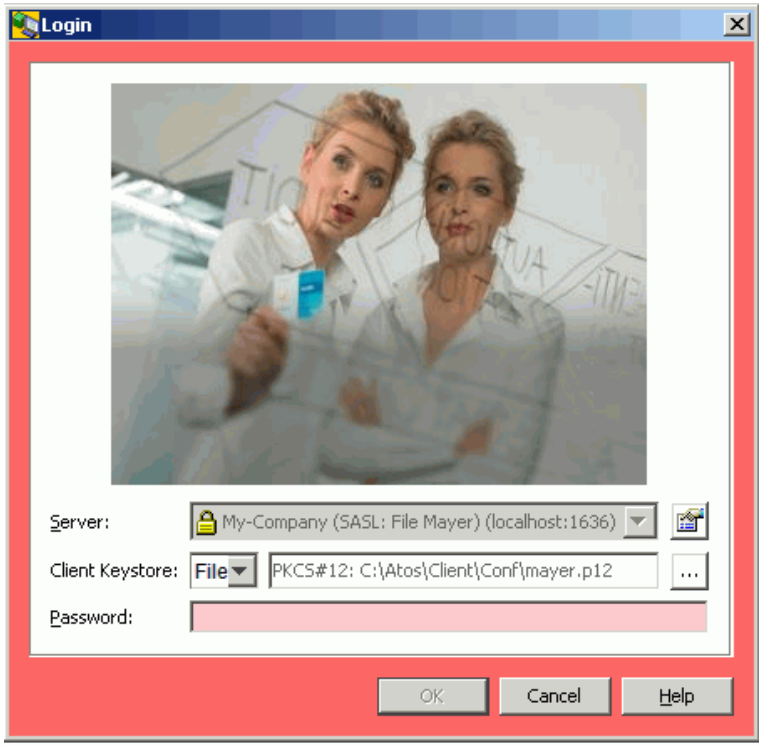


Figure 139. Login Dialog Box for File Based SASL EXTERNAL Bind

In order for this application to be able to connect to a server, you must:

- Specify a server
  Click the New... button in the Manage Server Profiles dialog to create a server profile, if no suitable profile is available, or click the button to edit the currently selected profile.
- Specify login information
- Specify a DN (Distinguished Name) and a password for a simple authenticated bind or alternatively log in "anonymously"
  If "anonymous" is checked in the selected server profile, the DN and password fields should be grayed out and are not editable.
- For smart card logins, insert the smart card into the reader and specify the PIN.
- For file based SASL EXTERNAL binds, press the ... button and browse to the file location of the JavaKeyStore (.jks) or PKCS#12 formatted container (.p12) with your private key and public key certificate(s) and specify the local passphrase (the password) that protects the .jks or .p12 file.

Note that you may be offered *read access* only, or *read and write access*- depending on the access policy settings in the server dialog.

7.3.4.6. Naming

Use the naming dialog to assign a name to an object that you are creating.

Here is an example:

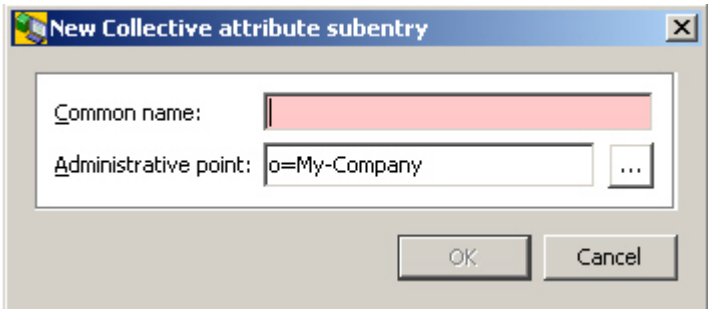


Figure 140. New Collective attribute subentry dialog box

7.3.4.7. Properties

7.3.4.7.1. Summary Properties

Properties are typically displayed and edited in

- A property pane (embedded into this applications main window)

  and/or
- A properties dialog (popping up in an own window you can resize up to the full screen, when right-clicking an entry and selecting "Properties" or when double clicking an entry).

For the functionality provided for properties, it does not really matter, if you work with property panes or prefer property dialogs. The use, however, may be different in details: while the property pane typically distinguishes between a read mode and an edit mode, the property dialog usually knows only one mode.

By default, a generic property dialog/pane has these tabs:

- An Info tab
- An "All Attributes" tab

The properties dialog/pane may be "generic" (i.e. it has the default tabs mentioned above) or customized. It displays details of the currently selected entry and may allow you to edit some of them (editable fields are ones that are not grayed out). A Close button in place of the OK and Cancel buttons indicates that the entry itself is not editable. Alternatively, the property dialog may make a distinction between view mode and edit mode. In this case you are to face buttons like the ones described in property pane.

The clipboard functions cut/copy/paste (cut and paste only in edit mode) are available on a per attribute value basis. Note that (in tree and list panes) clipboard (and drag and drop) functionality is also available for currently selected entries.

The properties dialog is usually available by:

- Right-clicking the entry of interest
- Double clicking the entry of interest
- Using the menu

See the topic Property Editors:Summary to learn about the buttons shown in the right-hand side of the properties dialog, as shown in the figure here.

Also see the topic Property Editors:Summary to learn how to view property values that do not fit into the value field.

Customized tabs

This application provides a customized tab for some types of objects (Group, Internet Organizational Person, Organizational Person, Organizational Unit) that adds to the Info and all attributes tab. Plug-ins may provide property panes/dialogs that are even more customized.

See also: Property editors.

7.3.4.7.2. Property Tab "General"

For a number of common object classes, additionally a "General" tab is shown accommodating the attributes considered most interesting. If more than one object class applies, the most specific one wins.
Examples:

- Person
- organizationalPerson
- inetOrgPerson
- organization
- organizationalUnit
- groupOfNames
- groupOfUniqueNames

Person

The pre-configured "General" tab for entries whose object class is "Person" looks like this (unless the entry has additional object classes like organizationalPerson):

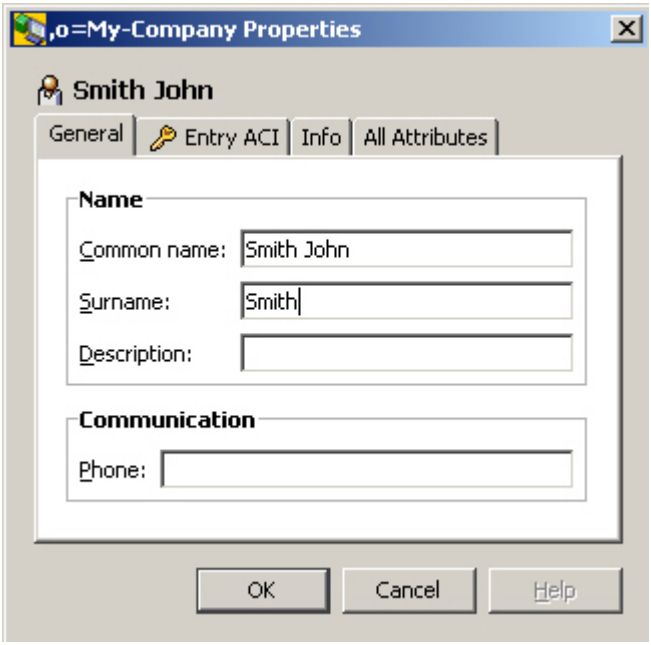


Figure 141. Person Dialog Box with General Tab

organizationalPerson

The pre-configured "General" tab for entries whose object class is "organizationalPerson" looks like this:

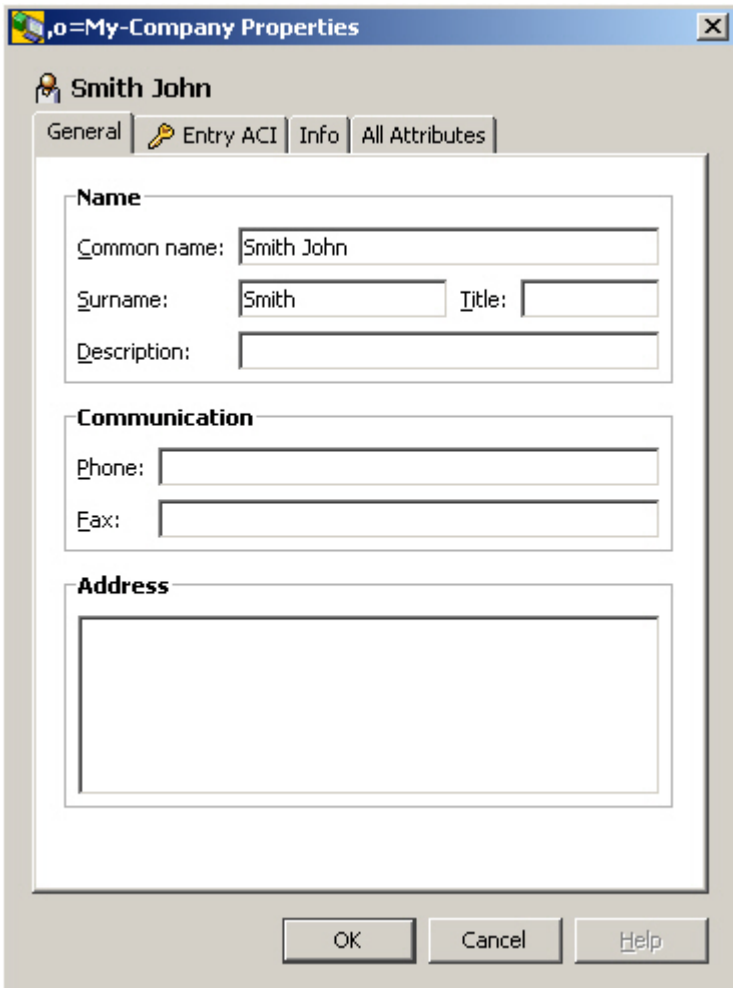


Figure 142. Organizational Person Dialog Box with General Tab

inetOrgPerson

The pre-configured "General" tab for entries whose object class is "inetOrgPerson" looks like this:

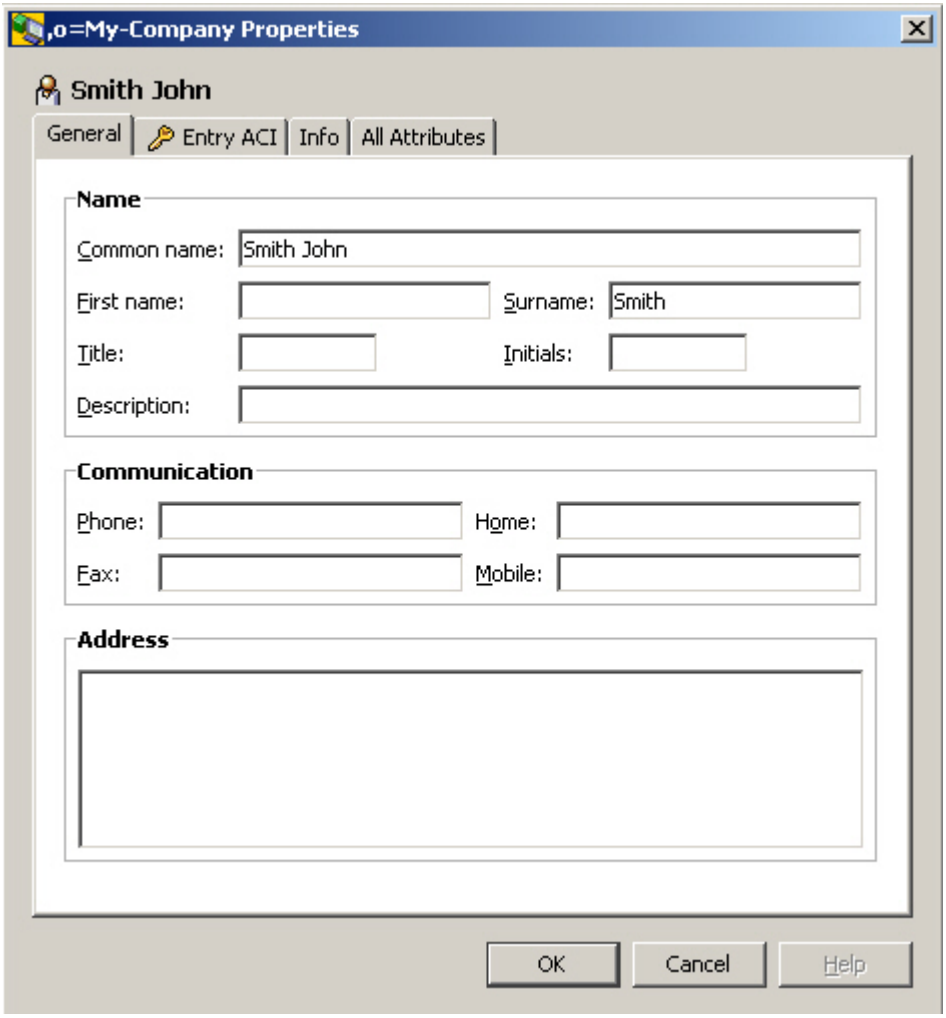


Figure 143. inetOrgPerson Dialog Box with General Tab

organization

The pre-configured "General" tab for entries whose object class is "organization" looks like this:

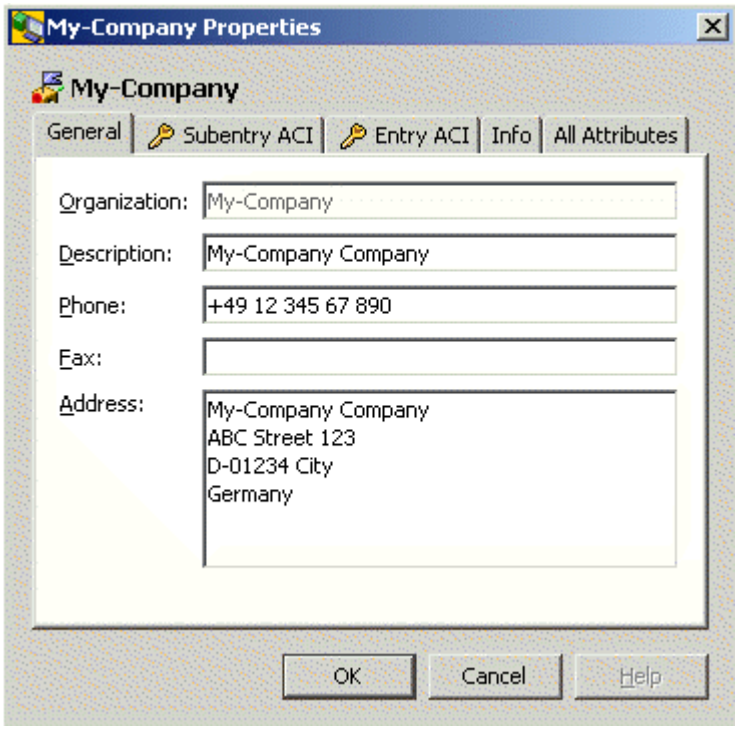


Figure 144. Organization Dialog Box with General Tab

organizationalUnit

The pre-configured "General" tab for entries whose object class is "organizationalUnit" looks like this:

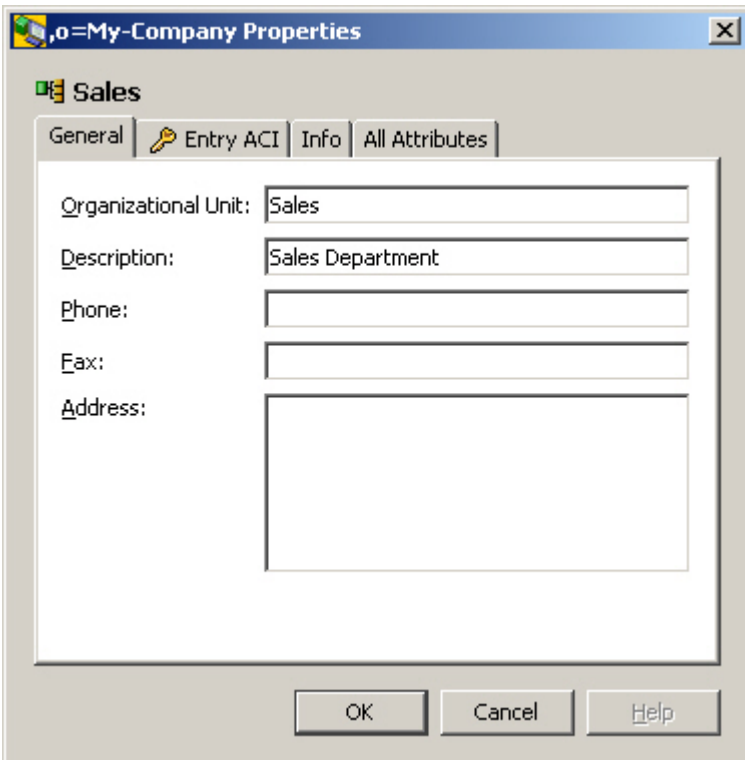


Figure 145. Organization Unit Dialog Box with General Tab

groupOfNames

The pre-configured "General" tab for entries whose object class is "groupOfNames" looks like this:

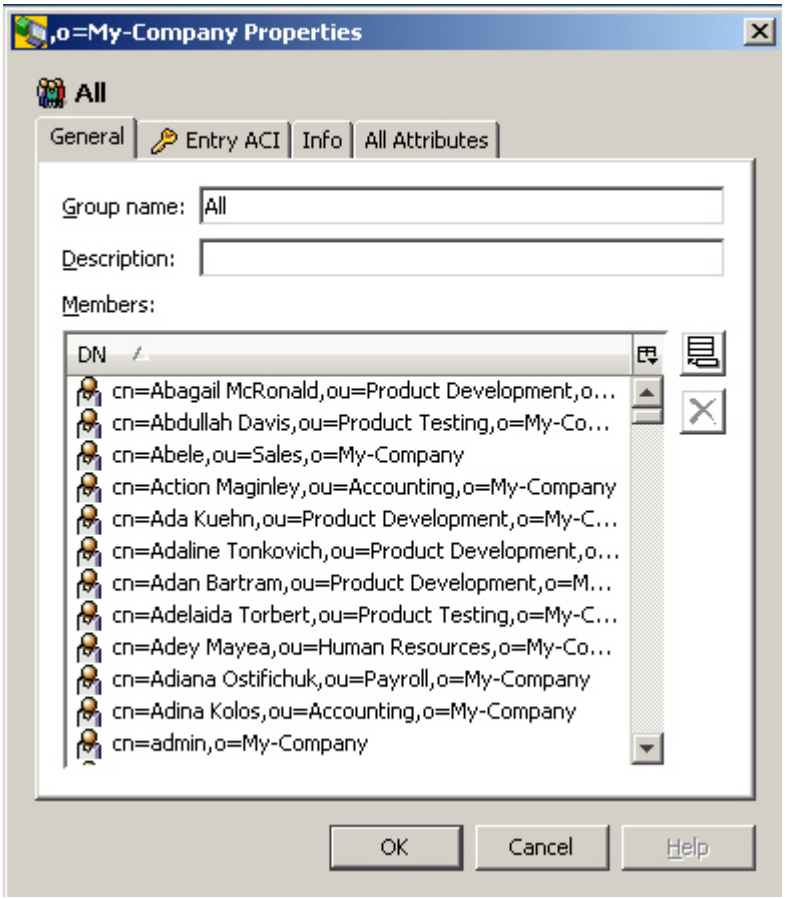


Figure 146. Group of Names Dialog Box with General Tab

groupOfUniqueNames

The pre-configured "General" tab for entries whose object class is "groupOfUniqueNames" looks like this:

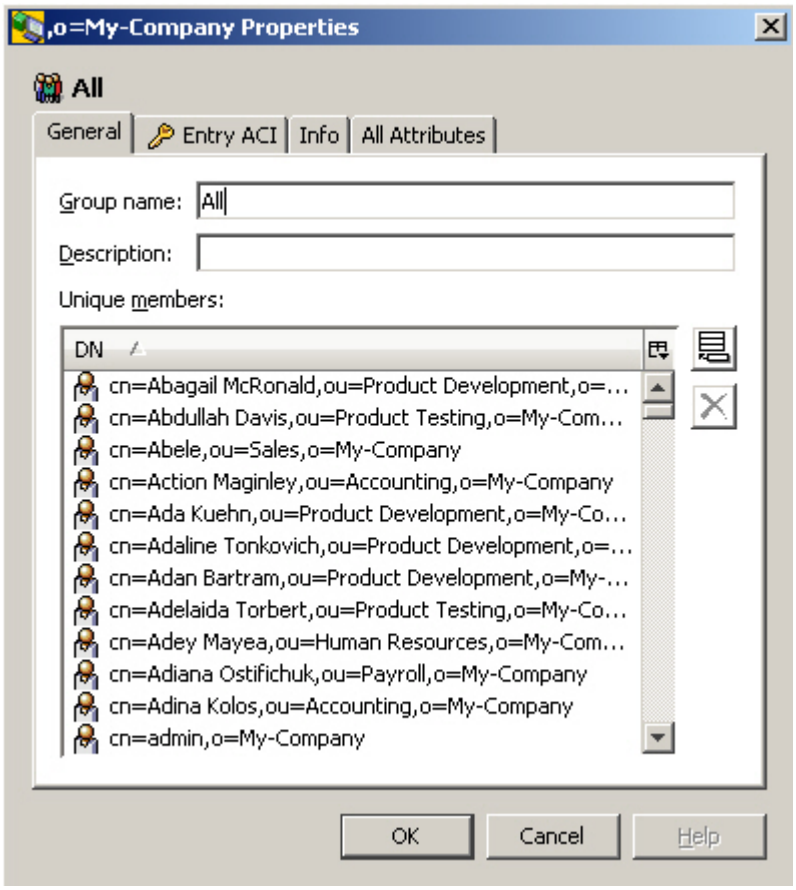


Figure 147. Group of Unique Names Dialog Box with General Tab

7.3.4.7.3. Property Tab "All Attributes"

The "all attributes" tab displays all attributes including the ones that are shown on the other tab(s).
Example:

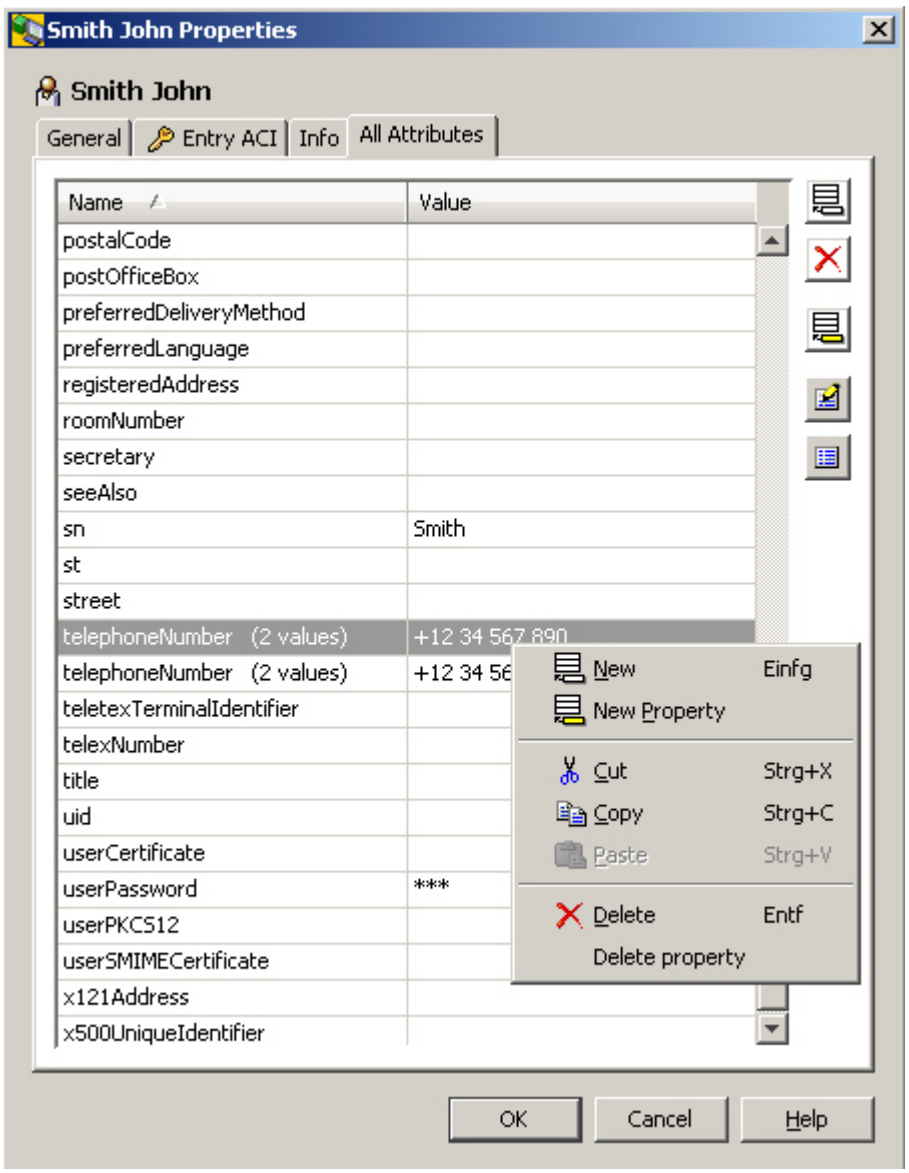


Figure 148. Smith John Property Tab All Attributes

See also Property Editors (for a description of the right-hand convenience buttons), Property Pane, Property Dialog and All Attributes/Right Mouse Button.
For Password modifications, see also: Change Password, Reset Password, User Password, Server.

If the name column is selected, it is possible to position to a particular attribute by typing the initial letter(s).

For attributes with more than one value, the number is displayed in the name column - usually the first column - to the right of the attribute name (administrativeRole (4 values)):

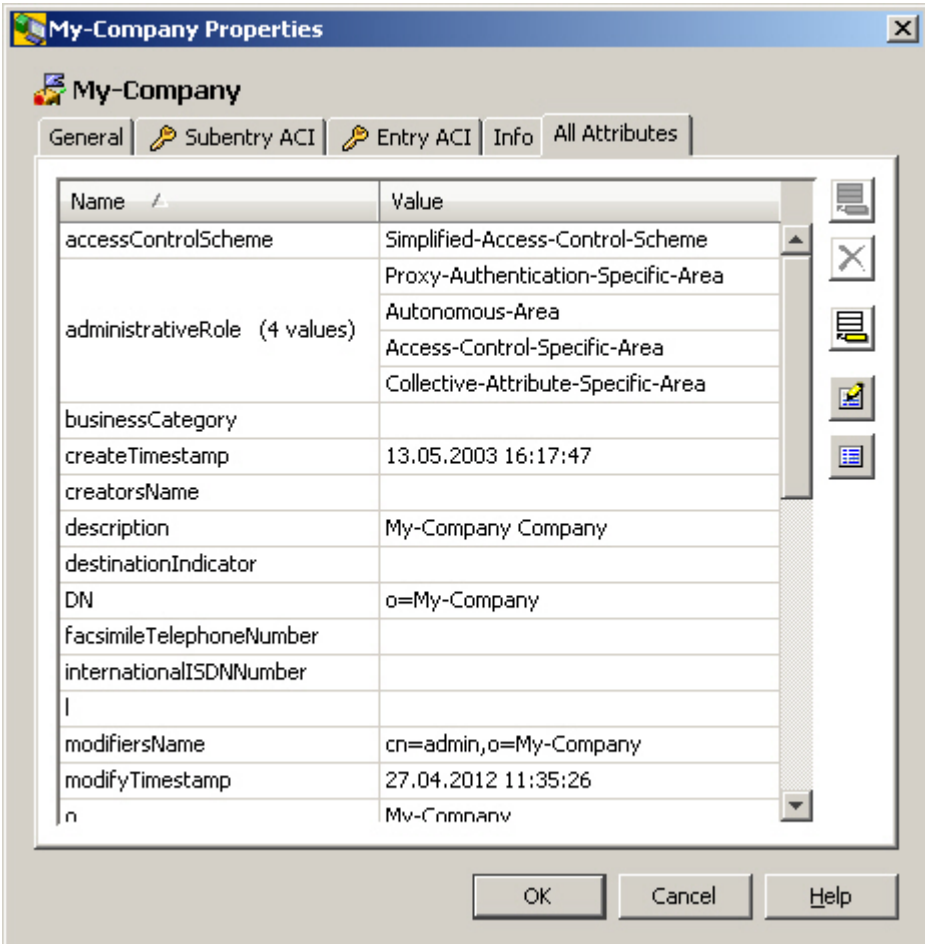


Figure 149. My-Company Property Tab All Attributes

If the attribute name is outside of the currently visible part of the pane/dialog the number of values is displayed as tool tip:

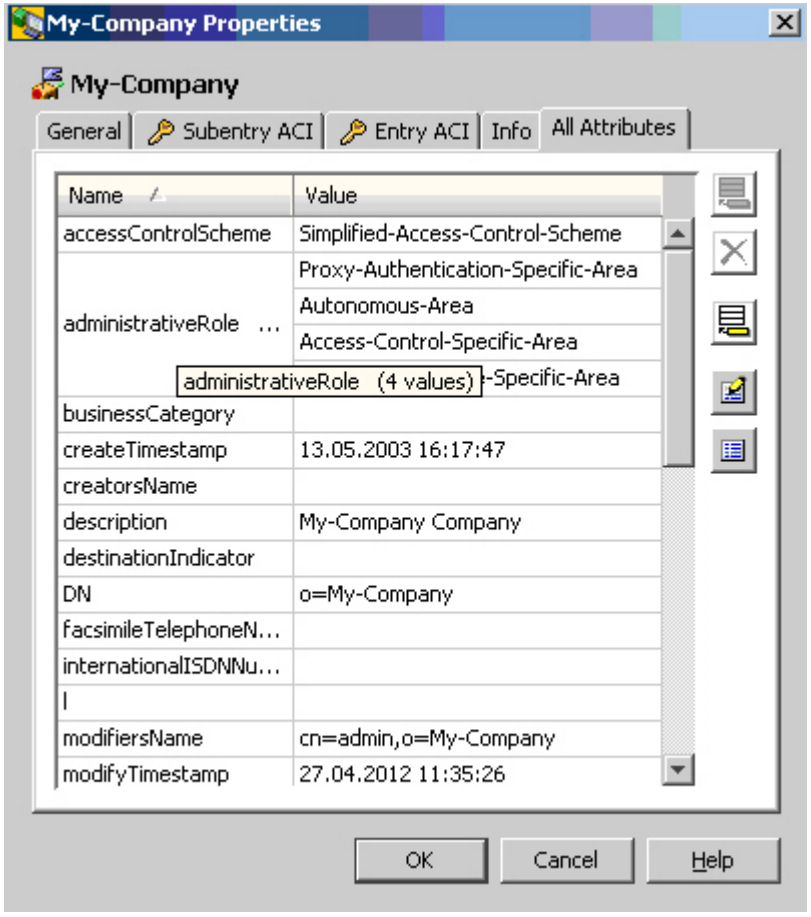


Figure 150. My-Company Property Tab All Attributes Displaying Number of Values as Tool Tip

7.3.4.7.4. Property Tab "Info"

In the following example the "Info" tab of a new entry with object class person is displayed:

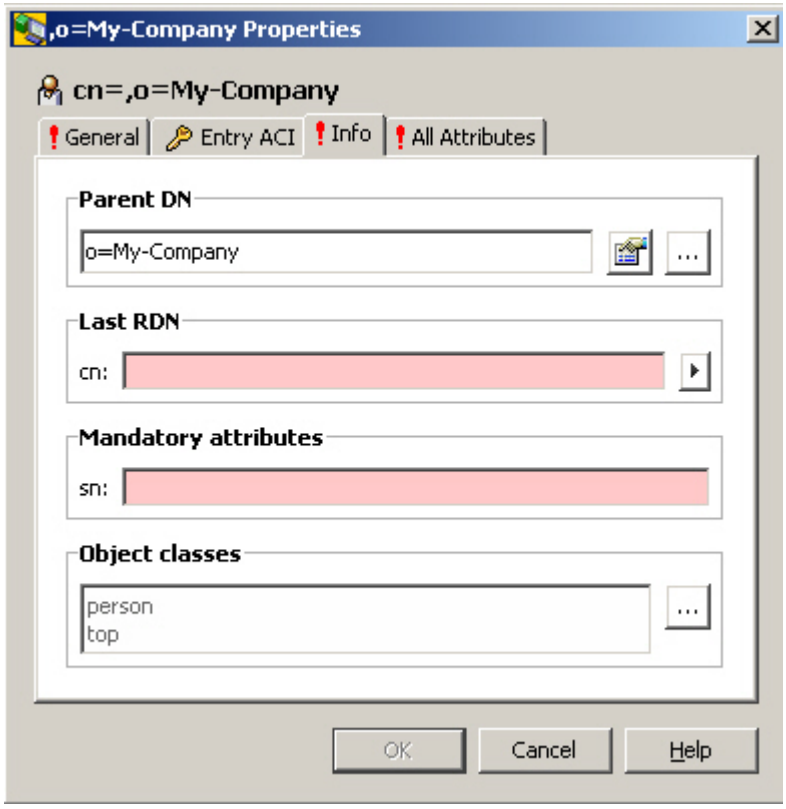


Figure 151. Property Tab Info of a New Entry

This tab displays

- The Parent DN
- The last RDN (pink, if missing because this is a mandatory field)
- All additional mandatory attributes (pink, if missing; there is just the mandatory attribute surname (sn) in the screen shot above). Mandatory attributes get additionally inserted automatically, as soon as another auxiliary object class with mandatory attributes is added.
- All Object Classes

See also Property Dialog, Property Editors and Property Pane.

7.3.4.8. Renaming

The renaming dialog allows you to rename the currently selected object. Note that the checkbox "Retain old name" does not appear unless you are renaming an LDAP entry. The same holds true for the button ▸, which allows you to assign additional naming attributes.

Here is an example (which illustrates renaming an LDAP entry):

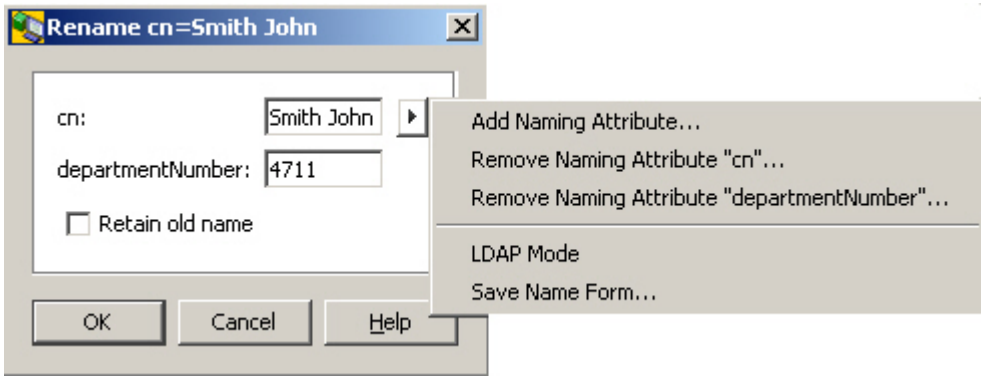


Figure 152. Renaming Dialog

The menu item contains the following selections (the menu appears when you right-click the ▸ button)

- Add Naming Attribute
  Adds an additional line that is to pick up an additional naming attribute (in the example dialog, the line "l: Atlantis" resulted from a previous use of this function)
- Remove Naming Attribute "..."
  Only present if there is more than one naming attribute on display
- LDAP Mode
  Changes the appearance of this dialog according to the screen shot shown next.
- Save Name Form...
  Stores the current name form locally for reuse when another object of the same type is to be created.

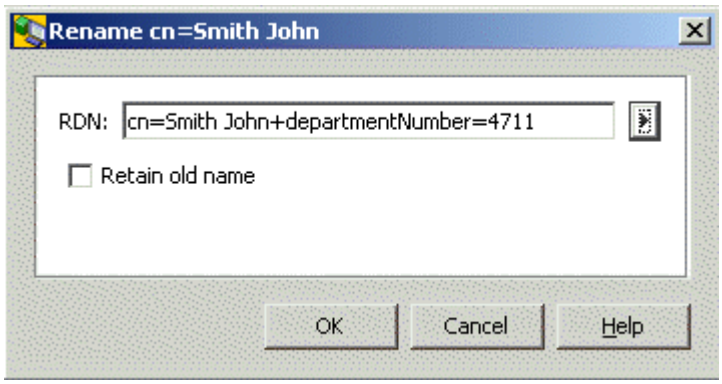


Figure 153. Renaming Dialog in LDAP Notation Mode

This is the same menu item as the previous example, but in LDAP notation. In this mode, there is only one function available through the right mouse button. This function changes the appearance of the dialog back to the look shown in the previous example.

7.3.4.9. Searching

There are two sorts of search dialogs available (may occur combined, too):

- The rather simple search that is supposed to be sufficient for most cases
  This dialog has a sub dialog that allows you to specify "compound object classes"
- The full featured advanced search that allows you to express quite complex searches

7.3.4.9.1. Simple Search

This dialog is displayed for example by the search function that is usually available on the right mouse button. The elements used in this dialog may likewise show up in a search pane (except for the search result, which is shown separately in an associated list pane).

Here is an example:

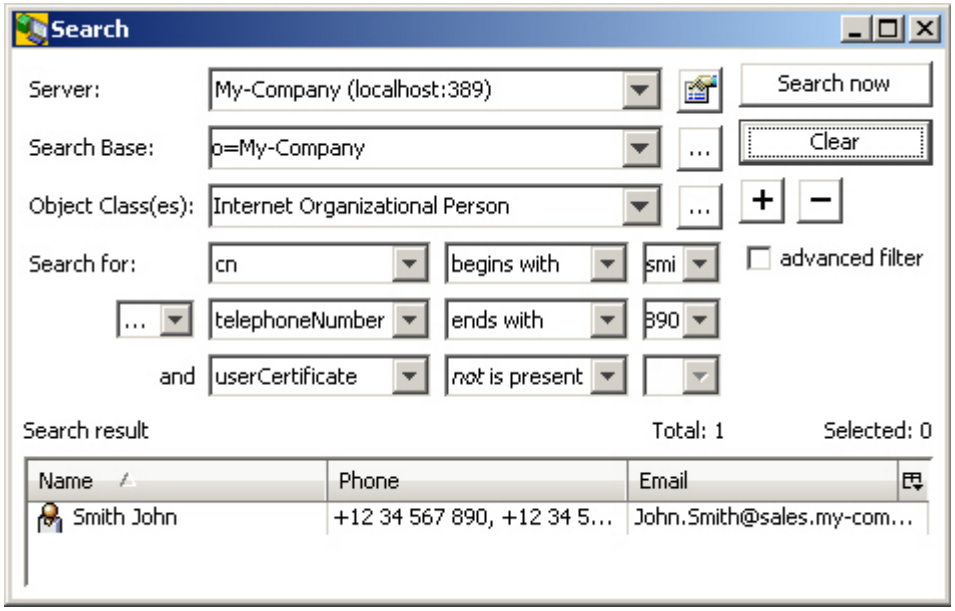


Figure 154. Simple Search

This dialog contains all or a subset of the following elements:

- A field to select the server to be addressed (suggested: the current server)
- A field to define the search base (suggested: the current entry)
  Click [...] (the one located to the right of the search base field) to display a dialog that allows you to choose a value from a tree.
  As opposed to the dialog, the search pane has an additional feature: If combined with a tree browsing pane, the search base may be taken from the entry currently selected in an associated tree pane - either automatically after switching back to the search tab or by explicitly clicking a button like 
- A field to choose the object class(es) the returned entries must have
  The choice of **filter attributes** offered in the "Search for" combo boxes that follow beneath depend on the object class(es) chosen here. If only one object class is selected, all **filter attributes** are offered that are (directly or indirectly) assigned to this object class (disregarding that attributes may generally be disabled). If more than one object class is selected (like in the example above), all **filter attributes** are offered that are (directly or indirectly) assigned to at least one of those object classes. Regarding to the sample, all filter attributes are offered that are assigned to object class "InetOrgPeron" *or* to object class "pkiUser" (or both).

  If the object classes are associated with "AND", a corresponding search result contains only entries having each of those object classes. If they are associated with "OR", a corresponding search result contains only entries having at least one of those object classes.

Not really an object class is "(any)": Choosing "(any)" causes *all* available **filter attributes** to be offered. The object class used behind (any) is "*" (LDAP jargon). Note that this means that only entries having an object class can be found. LDAP entries must have at least one object class anyway, violations of this rule are rather uncommon.

By default, the object class(es) combo box allows you to choose one single object class (or the pseudo object class "(any)"). However, you can define kind of "compound object classes" after pressing the [...] button that is located to the right of the object class field. The object class compositions you define this way are sorted ahead of the ordinary object classes in the object class combo box. You can assign a name to your definition. In this case, you can check how you have defined the current compound object class by looking at the tooltip of the object class(es) combo box. Regarding the sample above, the name is "People", the tooltip shows "inetOrgPerson OR pkiUser"

Auxiliary object classes are identified by a suffix such as "(aux)", whereas structural object classes lack any characteristic suffix.

- Fields to specify a **filter attribute**, **filter type** and **filter value**, e.g. **Common name begins with Smith Jo**
  This element may appear several times (controlled by two buttons that enable you to expand or collapse the number of these elements: [+] [−]). You can associate these elements by either logical "and" or logical "or". Mixing "and" and "or" is not possible.
  Filter types usually also offer negations (indicated thru a prefixed "*not*") as shown in the picture below:

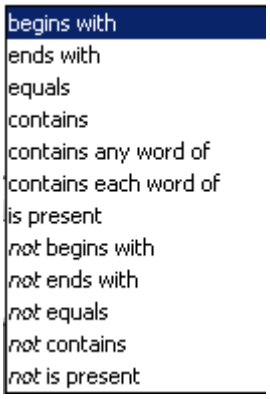


Figure 155. Filter Types

- Buttons to initiate a search or to cancel a search that has not yet completed: [Search now] or [Stop search]. Note that the button [Search now] may automatically change to [Stop search] each time a search is initiated.
- A button that causes the attribute filter elements to collapse to one element and delete its filter value: [Clear]

The dialog depicted above is technically a "frame", not a "dialog" (for example, it has no OK, Cancel or Close button), a differentiation that is generally ignored in this help. Alternatively, it can also occur as a real dialog; in that case, the search result field would be missing.

7.3.4.9.2. Compound Object Classes

A dialog like the one depicted below is popping up after pressing the [...] button that is located to the right of the object class(es) field in the simple search dialog or in the search pane. It allows you to define a logical association (AND or OR) of several object classes to be used as a filter element in a search.

The main idea is that you compose one structural object class with one or more related auxiliary object classes. For example, you could specify "inetOrgPerson AND pkiUser". In this case, you would search for entries having *both* object classes and you would only be offered the **filter attributes** that are (directly or indirectly) assigned to inetOrgPerson or to pkiUser, e.g. you would be offered "telephoneNumber" (which is commonly assigned directly to organizationalPerson and - by way of inheritance - indirectly to inetOrgPerson) as well as "userCertificate" (which is commonly assigned to pkiUser). This would otherwise only be possible with object class "(any)".

You can also compose two or more auxiliary object classes or two or more structural object classes, respectively. The latter makes sense at most if you associate them with "OR" (otherwise all object classes but one would be superfluous (if they are all in one inheritance line) or you would never get a search result (if not)).

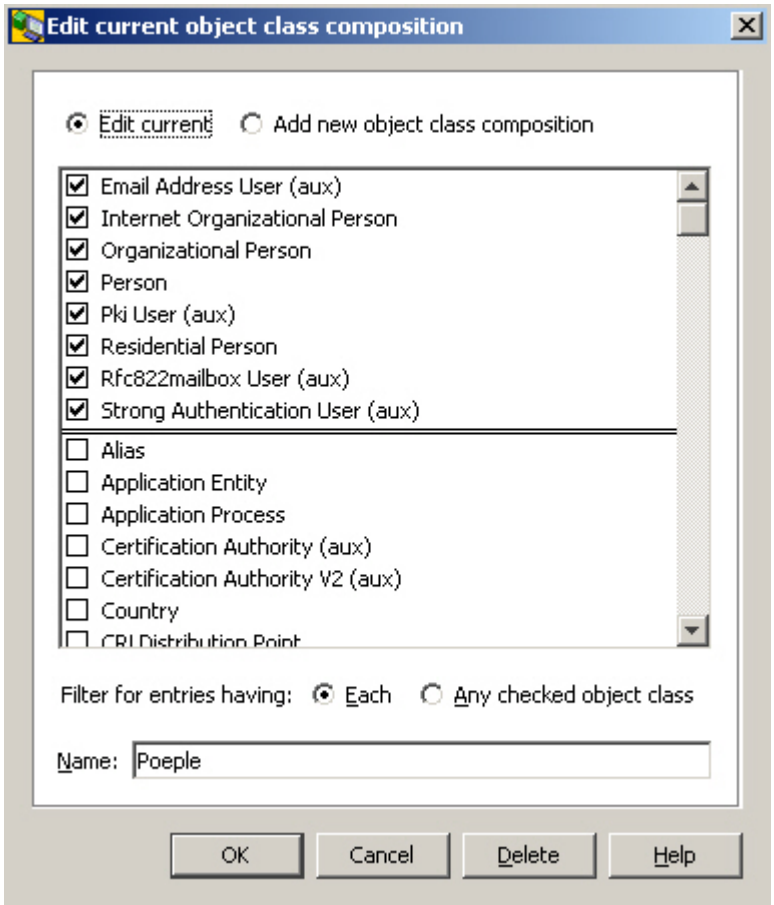


Figure 156. Edit current object class composition dialog box

Auxiliary object classes are identified by a suffix such as "(aux)", whereas structural object classes lack any characteristic suffix.

Check the **Edit current** check box if you want to edit a previously defined compound object

class. Check the **Add new object class composition** check box if you want to create a new one.

Check the **Each** check box if you want to associate the checked object classes with "AND". Check the **Any checked object class** if you want to associate them with "OR"

Click the **Delete**-button if you want to delete the currently selected object class definition (note that you cannot delete an "ordinary" object class that was not defined by yourself but was obtained from the schema).

You may or may not assign a name to your object class definition that is to be shown in the object class(es) combo box of the search dialog or the search pane instead of a list of the checked object classes. The tool tip will still provide you with information concerning the involved object classes and their logical association.

7.3.4.9.3. Advanced Search

At appropriate places, a search dialog like the one depicted below may be present; it is intended to allow for both, composing filters complex enough to meet even the most sophisticated needs as well as for comfortably composing rather ordinary filters. Unlike this filter type, the simple filter is somewhat restricted in order to be able to satisfy the majority of use cases particularly conveniently.

Example of an advanced search (without supplementary simple search tab):

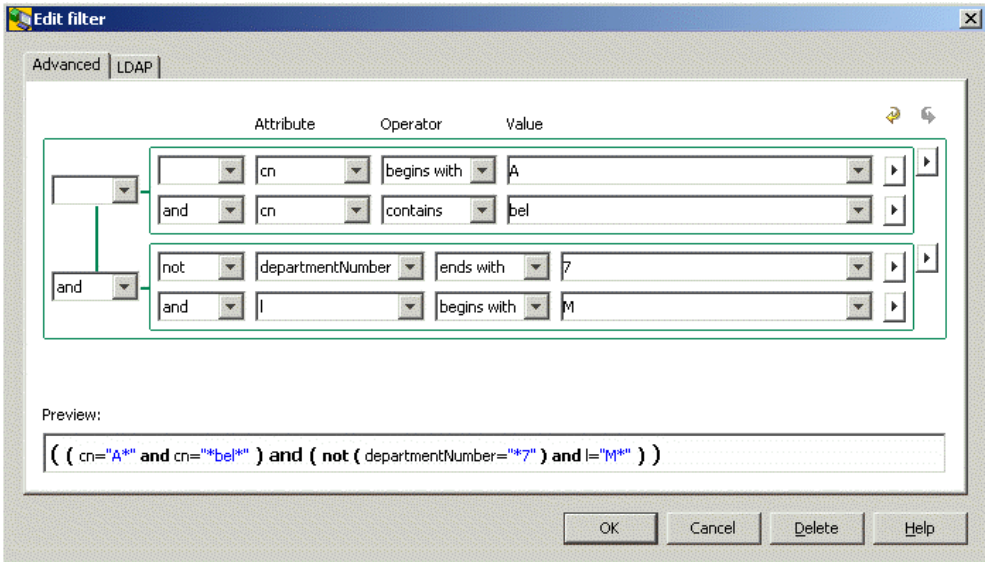


Figure 157. Edit filter dialog box with Advanced Tab

The advanced search allows to:

- Associate attributes with values thru operators like "begins with", "ends with", "equals", "contains", "is present"
- Combine such associations with "not", "and", "or", "and not", "or not" (use the "inner" buttons)
- Group such associations and combine those groups analogously (use the "outer" buttons)

- Recursively nest groups (use the "outer"  buttons)
- Delete the current row or group row (use the  buttons)

Use the button  to undo and  to redo your previous action.

The tab completes with a preview field that shows the resulting logical expression is intended to help you keep track.

The text entered in the Value field is treated as the search value itself. If the value contains the special characters backslash "\", left parenthesis "(" or right parenthesis ")" they are internally changed to the corresponding escape sequences "\5c", "\28" or "\29". However, if a backslash is followed by two hexadecimal digits, it is treated as a valid escape sequence and the backslash is kept as it is.

For example:

- "\abc" will be kept as "\abc" because "\ab" is a valid escape sequence. The filter will then search for a two-character text beginning with a character with hexadecimal code AB followed by the character "c".
- "\a\b\c" will be replaced with "\5ca\5cb\5cc" since none of "\a\", "\b\" or "\c" is a valid escape sequence. The "\" character will then be replaced with "\5c".

The Attribute field can also contain virtual attributes; for example, “dxrOptions(location)”. When converting these attributes to the LDAP search expression, the name part in the parenthesis is converted to the attribute value.

When parsing the virtual attributes from the LDAP search expression to the Attribute combo, it is first verified whether the search expression matches some of the virtual attributes in the Attribute combo list. If the LDAP search expression matches a virtual attribute, it is treated as a virtual attribute. If the LDAP search expression doesn’t match any virtual attribute, it is treated as standard attribute.

For example:

- If the search expression is "dxrOptions(location) equals Munich", the resulting LDAP search expression is "(dxrOptions=location Munich)".
- Parsing the LDAP search expression "(dxrOptions=location Munich)" results in "dxrOptions(location) equals Munich" since the "dxrOptions(location)" is present in the combo box selection list and thus recognized as a virtual attribute.
- Parsing the LDAP search expression "(dxrOptions=locate Munich)" results in "dxrOptions equals locate Munich" since the "dxrOptions(locate)" is not present in the combo box selection list. Thus, the attribute is recognized as pure "dxrOptions" without any further specification.

The dialog also offers a tab addressing those who are accustomed to genuine LDAP syntax. This tab allows you to enter the filter in the LDAP filter expression format. The search expression is taken "as is" and you are fully responsible for its correctness.

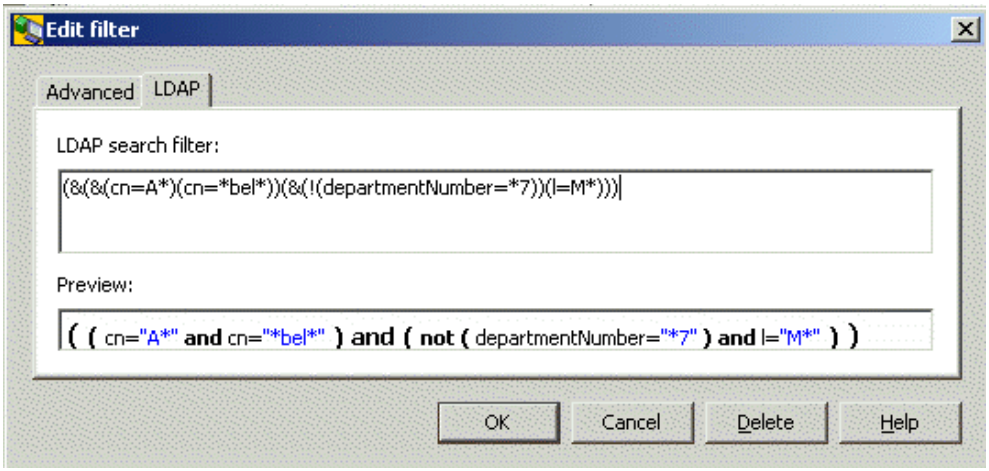


Figure 158. Edit filter dialog box with LDAP Tab

When switching from the LDAP tab to the Advanced tab, the search structure in the Advanced tab is reconstructed from the LDAP filter expression in the LDAP tab.

When switching from the Advanced tab to the LDAP tab, the LDAP search expression is created from the search structure only if some modifications were made - the "undo" button in the Advanced tab is enabled. If the "undo" button is disabled, no changes were made and the LDAP search expression in the LDAP tab remains unchanged. This feature allows you to enter any search expression without overwriting from the Advanced tab.

7.3.4.10. Server

The server dialog allows you to specify server-specific information in a "Server Profile", particularly information needed to connect to an LDAP server. Server profiles often correspond to view groups or views. However, it is also possible to address different servers (or the same server with different server profiles settings) within a single tree pane.

Plug-ins may categorize server profiles; that is, they can ensure that you can only see server profiles of a certain category within the view groups, views, nodes in tree panes under control of a plug-in. Category names are not displayed. New server profiles you create in a context where a category is in effect already, implicitly inheriting that category.

The settings you can make here are organized into these tabs:

General
Deals with LDAP connection related settings

Options
Primarily deals with controls that are to be passed to the LDAP server along with respective LDAP operations

Visibility
Deals with the visibility of attributes and object classes in search panes and search dialogs

Password
Allows you to specify, if

- the additional password of DirX Identity users should be managed

- affected users should be notified by mail on password reset

LDAP Root
Displays what the server returns if asked for "LDAP Root".

General

Here is an example of the dialog's "General" tab:

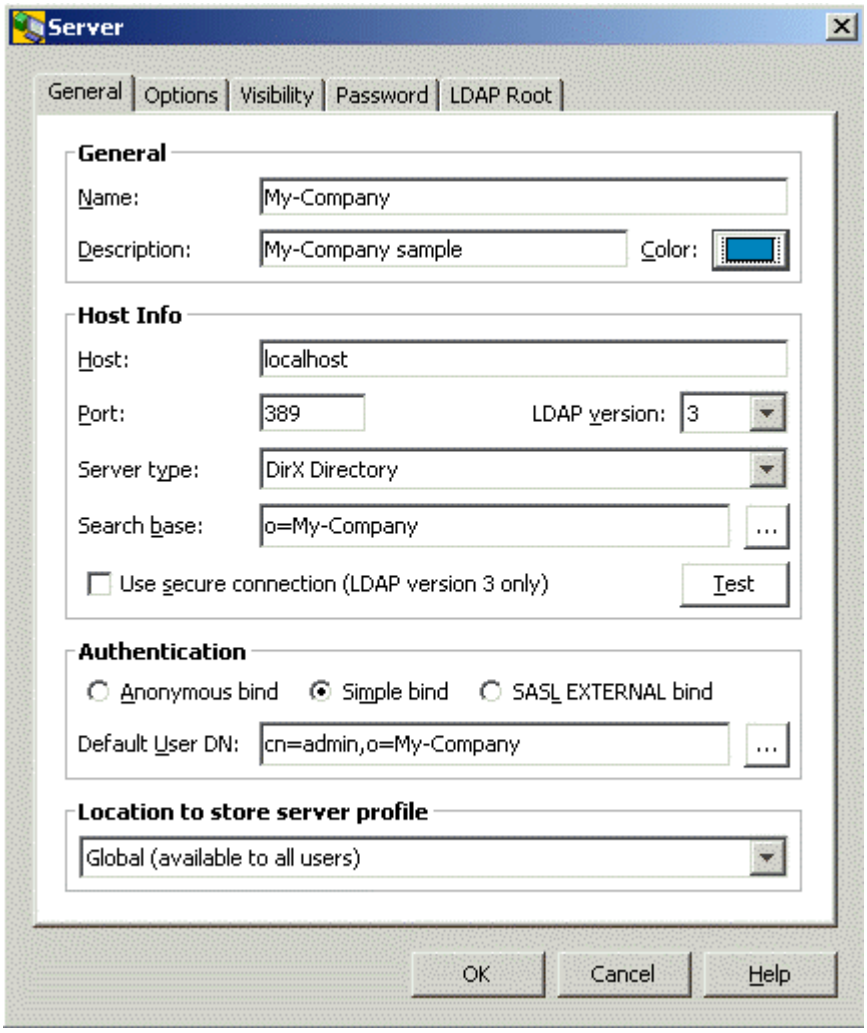


Figure 159. Server Dialog LDAP Root General Tab

This tab deals with LDAP connection related settings and contains the following fields:

Name: name of this server profile

Description: any additional information you might consider helpful

Color: the color for this server profile

Host: host name or IP address
Also possible: a blank-separated list of hosts with or without complemental port number, e.g. "localhost:389 111.222.333.444 666.777.888.999:636"
In this case, this application will connect to the first address it can successfully connect to, starting with the first one. If the port number is missing, the value specified in the following

field (Port) will be used.*
Note: when changing Host or Port here, the settings related settings in the password tab may need to be changed as well.*

Port: port number
By default, the server listens to port 389 or - if SSL is enabled - to port 636.
Note that

- The designated port may be occupied by a different program. In this case, the LDAP server cannot start up with that port and must be assigned a different one.
- The server side configuration may have been changed to correct a conflicting port problem or for another reason. In this case, the LDAP server should be able to start up, but you need to specify the corresponding port number here.

LDAP version: LDAP protocol version (normally: 3)

Server Type
This field is only present, if the "DirX Manager" plug-in is in place. There are two possible values: "DirX Directory" and "Other". The functionality DirX Directory Manager offers depends on this selection:

- The subsequent field is called "Base DN", if server type is "Other" and "Search Base" otherwise.
- Several functional areas work only with suitable versions of DirX Directory. These functional areas include:
- Database (found in the DirX Directory Manager "Schema" view)
- LDAP cache configuration (found in the Dirx Directory Manager "Configuration" view)
- Password Policy (found in the DirX Directory Manager "Configuration" view)
- Replication (DirX Directory Manager "Replication" view)
- Schema is editable (DirX Directory Manager "Schema" view)

What functional areas ("features") are mapped to what vendor versions is specified in the file config\DirX.policy. Note that this release of DirX Directory Manager is not necessarily able to interwork with future releases of DirX Directory particularly regarding the functional areas mentioned above.

- Also affected by the server type is the following field (Search Base/Base DN).

Search Base (displayed if server type is "DirX Directory") resp. **Base DN** (displayed, if server type is "Other" or if the DirX Directory Manager plug-in is absent)

- Browsing
  Ineffective, if Server type is "DirX Directory": Regardless of what is specified here, the tree panes always display the context prefixes as top nodes.
  Server type is "other": The DN specified here is used as (only) top node in tree panes.
- Searching
  The DN specified here is used as the default search base.

Use secure connection
In order for a secure connection to work, you must ensure that a suitable certificate is available in a local key store. By default, a test certificate is used that matches the default the DirX Directory LDAP server uses. Note that this certificate is only meant for demonstration. Read more about this in the chapter SSL/TLS.

Anonymous bind: to have this application try to connect without credentials (anonymously)

Simple bind: to have this application try to connect over a simple authenticated bind with credentials

SASL_EXTERNAL bind: to have this application try to connect over a SASL-authenticated external bind. If you select this option you must specify the client keystore. The client keystore is either stored in a **File** or on a **Smart Card (PKCS#11)**. Using a file is intended for demonstration or testing purposes. For details on how to perform a smart card login with DirX Directory Manager, see "Smart Card Login" above.

Default User DN: The DN you typically use to log in

Location to store server profile: the choice of locations may vary depending on the plug-in that is behind this dialog. Generally, the available choices should be all or a subset of:

- User Home: This profile is to be used only by you
  (the profile will be stored in your home directory)
- Global: This profile is to be offered all users using the same machine

Options

Here is an example of the dialog's "Options" tab:

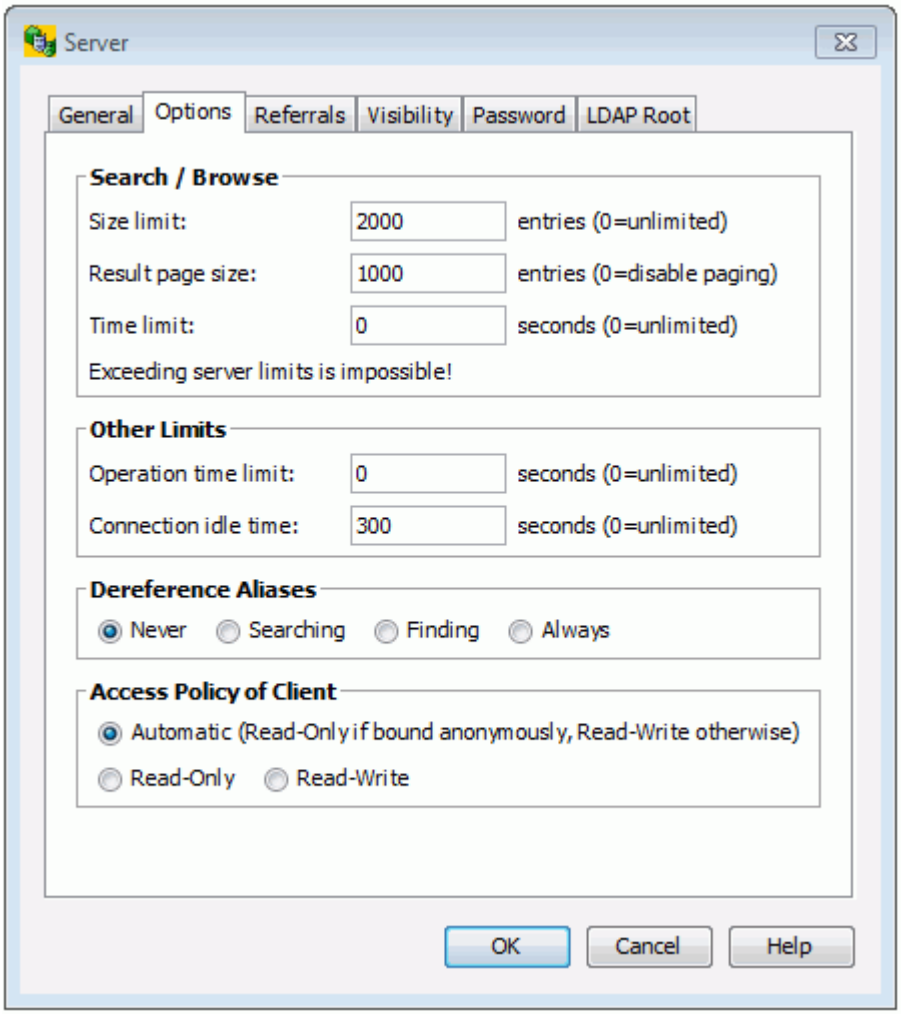


Figure 160. Server Dialog LDAP Root Options Tab

This tab primarily deals with controls that are to be passed to the LDAP server along with respective LDAP operations:

Limits

- **Size limit**: the maximum number of entries the LDAP server is to return as the result of a search or browse operation. The value **0** is unlimited number of entries. Note that the server may return fewer entries, since it may have a more restricted server-side size limit. As for DirX Directory, specify a size limit for the LDAP server in the LDAP configuration subentry, for the X.500 server in the user policy attribute(s).
- **Result page size**: the maximum number of entries of one page of the search result. The value **0** disables paging.
- **Time limit**: the maximum period of time granted to the LDAP server to complete a search or browse operation. The value **0** is unlimited number of seconds. Note that the LDAP server may have a more restricted server-side time limit. Also, the server may not observe this limit strictly.
- Note that if a size or time limit takes effect, the remaining (unread) entries can come from anywhere within the range of entries on display; that is, in the result that the server returns, some existing entries may be omitted, while others that are alphabetically behind may be present.

- **Operation time limit**: under rare conditions, a server may fail to return any result within a reasonable amount of time, even if other limits have been set. The operation time limit limits this application's "patience" with the duration of an LDAP operation. If the patience is exhausted, this application, in order to avoid getting blocked, displays a message and discards the result. Once the operation time limit has been reached, a corresponding result that arrives late will be disregarded. The value **0** is unlimited number of seconds.
- **Connection idle time** limit: Limits the amount of time this application is to maintain a connection with the server when there is no traffic on the connection. The value **0** is unlimited number of seconds. Note that the server may have a more restrictive connection idle time limit and cancel the connection on its own.
  Note that if a connection has been cancelled and the user initiates another operation, this application will normally try to implicitly re-establish that connection.

De-referencing aliases. Possible values are:

- **Never**: specifies that aliases are never de-referenced
- **Searching**: specifies that possible aliases are de-referenced in the search result
  Note that when intending to manage (particularly delete) alias entries, you should not have checked **Searching** or **Always**, since then the server does not return any alias entries. Otherwise you run the risk of mistakenly deleting the original entry rather than the alias with the alias pointing to an entry that no longer exists.
- **Finding**: specifies that the search base is to be de-referenced if it happens to be an alias
- **Always**: specifies that aliases are always de-referenced
  Note that when intending to manage (particularly delete) alias entries, you should not have checked **Searching** or **Always**, since then the server does not return any alias entries. Otherwise you run the risk of mistakenly deleting the original entry rather than the alias with the alias pointing to an entry that no longer exists.

Access Policy
An LDAP server primarily grants and denies access rights based on access control information stored at the server side. This application normally cannot determine a particular user's access rights. By default, the GUI does not offer any add/modify operations to users logging in anonymously and does offer those operations to users logging in with credentials. This behavior is not always appropriate and can therefore be changed. Possible values are:

- **Automatic**: Apply the default policy.
- **Read-only**: Do not offer add/modify operations no matter how the user has been logging himself in.
- **Read-Write**: Offer add/modify operations even if the user has been logging himself in anonymously.

Referrals

Here is an example of the dialog's "Referrals" tab:

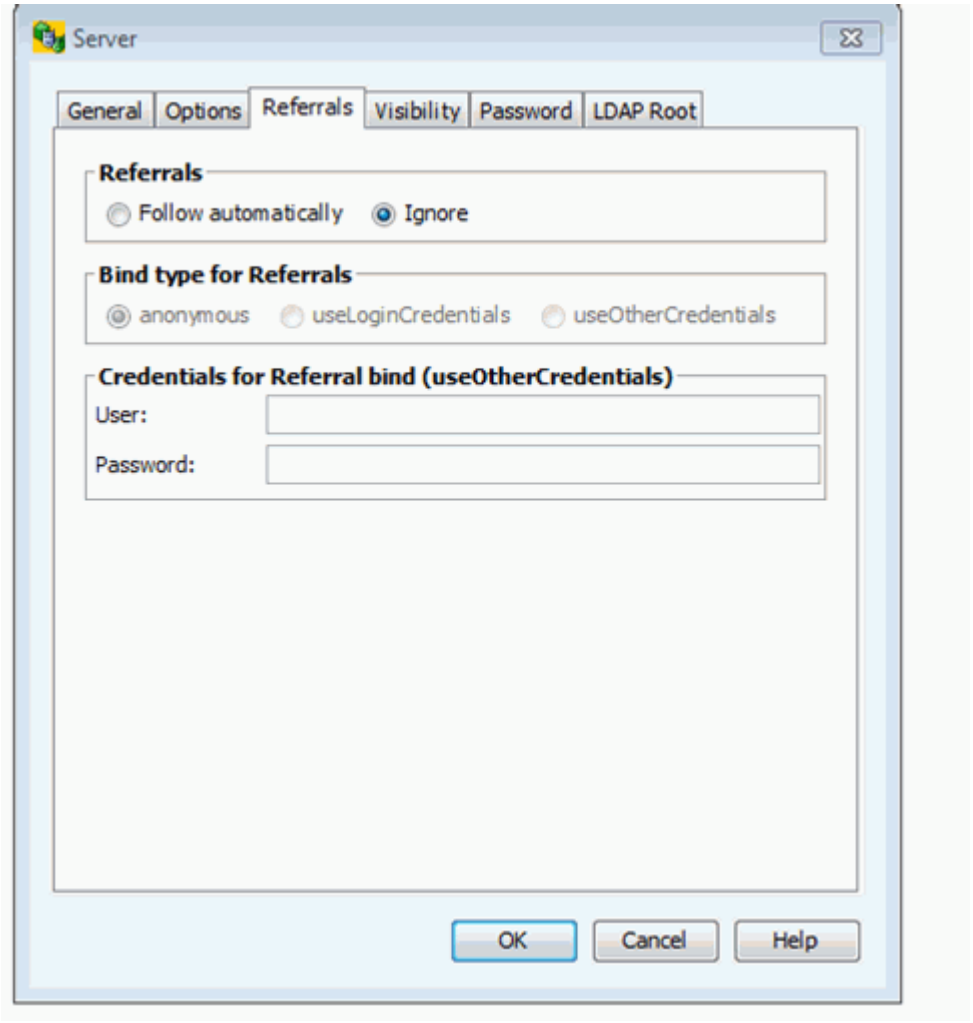


Figure 161. Server Dialog LDAP Root Referrals Tab

This tab deals with the handling of referrals.

Referrals
A server may return referrals pointing to other servers as part of a search result, thus indicating that one or more other servers should be consulted in order to complete a search operation.

- **Follow automatically**: the application automatically follows any referrals returned by the server that points to a different server until the "referral hop" limit is exceeded. The referral hop limit defaults to 10. This application does not support modifying the referral hop. Note that following referrals automatically does not work unless the server to which the request is referred accepts anonymous binds.
- **Ignore**: Referrals returned by the server are discarded.

Bind type for Referrals
Here you specify which bind you want to use for following the referrals.

- **anonymous**: an anonymous bind is used for following referrals. This is the default.
- **useLoginCredentials**: The same credentials you used for login to DirX Directory Manager are used.

- **useOtherCredentials**: the credentials defined in “Credentials for Referral bind” are used.

Credentials for Referral bind (useOtherCredentials)
Here you specify the credentials used for bind to follow the referrals. Only active if **useOtherCredentials** is selected in “Bind type for Referrals”

- **User**: the user for bind to follow referrals.
- **Password**: the password for bind to follow referrals.

Visibility

Here is an example of the dialog’s "Visibility" tab:

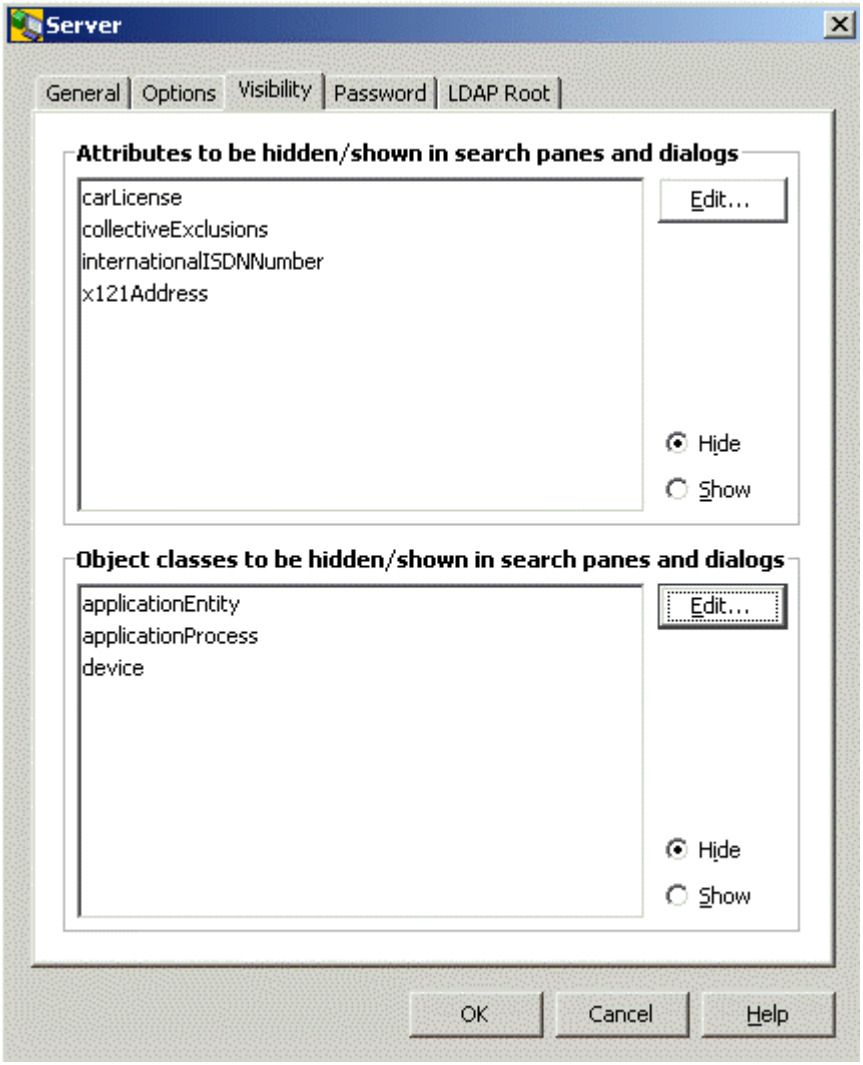


Figure 162. Server Dialog LDAP Root Visibility Tab

This tab deals with the visibility of attributes and object classes in search panes and search dialogs.

The schema usually provides quite a number of object classes and attributes you never intend to use in searches - and whose appearance in the respective combo boxes of search panes and search dialogs you may find bothersome. This dialog along with the subdialogs that pop up when you click one of the "Edit..." buttons allows you to hide unwanted object

classes and attributes in those combo boxes. You can either select all elements you want to have hidden (which means that all remaining ones are shown) or you can select all elements you want to have shown (that means all remaining ones are hidden).

Notes:

- Certain object classes, particularly object classes of subentries, are a priori hidden in search panes and search dialogs and cannot be made visible.
- Hiding all attributes of an object class (exactly: all attributes but the attribute "object class" that every object class has, at least if derived from object class "top") implicitly causes that object class to be hidden, too.

Password

Here is an example of the dialog's "Password" tab:

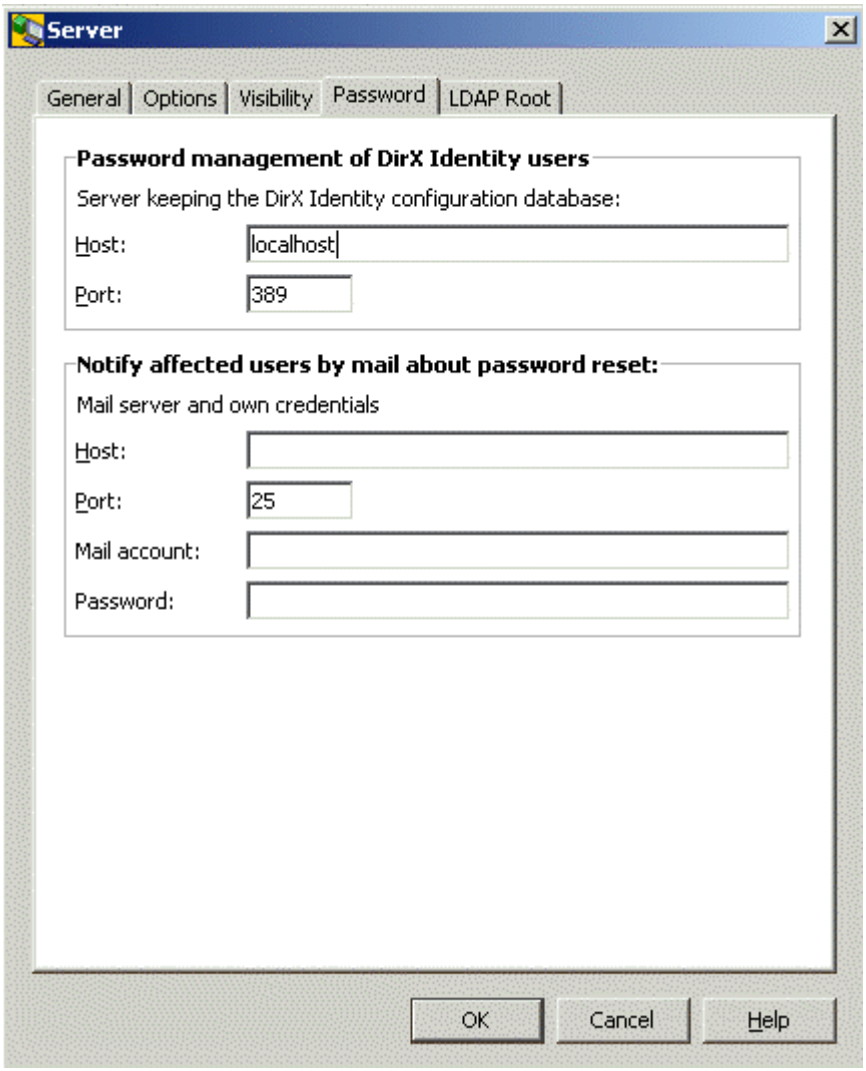


Figure 163. Server Dialog LDAP Root Password Tab

Password management of DirX Identity users
Only of interest if the server addresses a DirX Identity database. Among many other things, DirX Identity deals with DirX Identity users that are characterized by an additional password stored in the attribute *dxmPassword*. This password must - depending on the DirX Identity

configuration - either be scrambled or two-way encrypted. If you modify the ordinary attribute *userPassword* through DirX Identity functionality, *dxmPassword* will be implicitly modified, too.

The core component of this application has generic password management functionality that is originally not DirX Identity aware. However, by completing the settings you can do here, you can have the core component behave like DirX Identity when modifying *userPassword*. The Host/Port fields relating to the password management of DirX Identity users are editable, if the schema contains the attribute *dxmPassword (*note that this can only be found out, if the schema can be read). In this case, the core component will consider this server a DirX Identity database. To be able to do so, the core component must enquire the DirX Identity configuration to find out, whether *dxmPassword* is to be scrambled or to be encrypted; in the latter case, also the key must be picked up. Note that the DirX Identity configuration may reside in a different server. By default, the server storing the DirX Identity database is assumed to also store the DirX Identity configuration.

If you leave the respective Host/Port fields empty or if *dxmPassword* cannot be detected in the schema, the core component will not implicitly deal with the attribute *dxmPassword*.

Furthermore, if the server keeping the DirX Identity users is set up, the mirrored administrator user must be present in the Connectivity under the RDN dxmC=Users,dxmC=DirXmetahub.

Mail notification on password reset
If the affected users are to be notified by mail on password resets, you must specify host and port of the mail server as well as your credentials, i.e. your mail account and the belonging password. Note that the format of the mail account varies depending on the mail server in use. Common formats include: "<your domain/your name>", <your email address>, <a number>.
Note that the reset password functionality is only available, if the server indicates its support in the LDAP Root DSE ("Password Policy Control"; Object Identifier, displayed directly or thru tooltip: 1.3.6.1.4.1.42.2.27.8.5.1) and if the operational attribute *pwdReset* can be set.

See also: Property Tab "All Attributes", Change Password, Reset Password, User Password.

LDAP Root

Here is an example of the dialog’s "LDAP Root" tab:

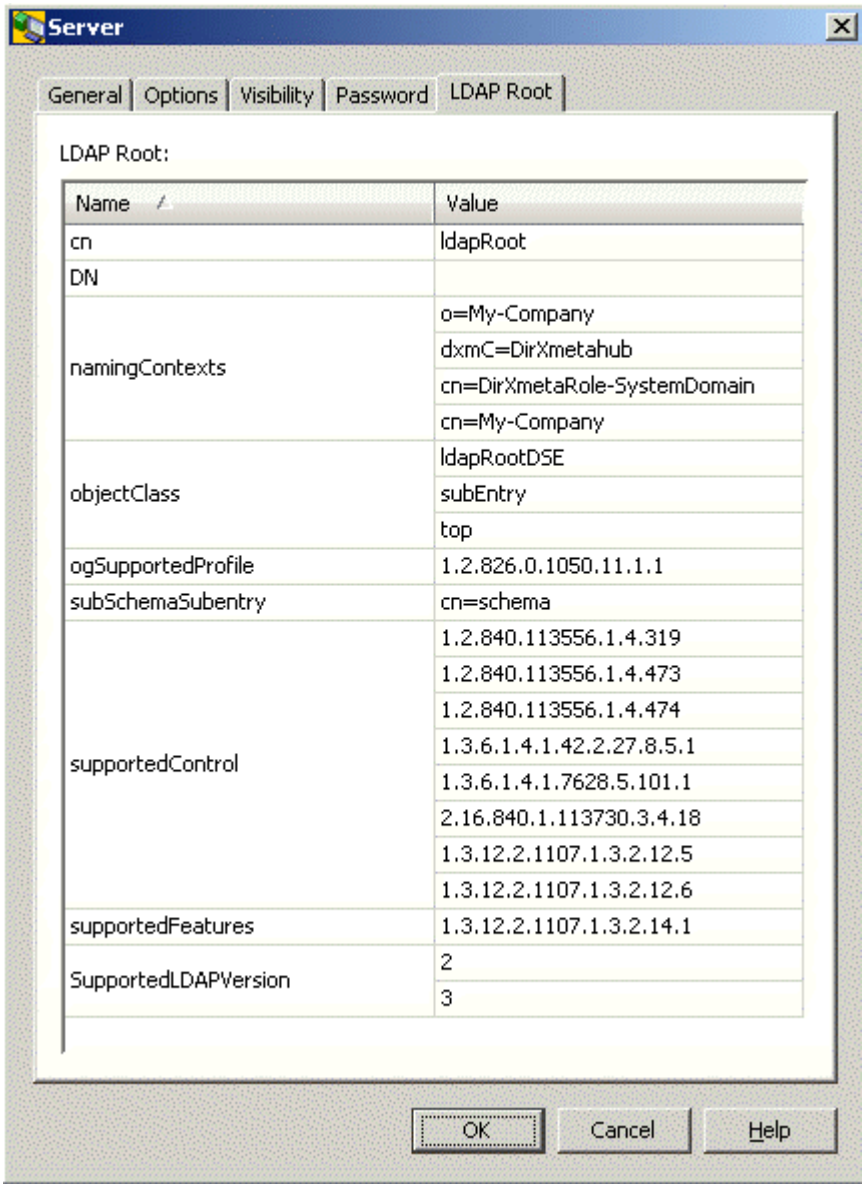


Figure 164. Server Dialog LDAP Root Tab

This tab displays what the server returns if asked for "LDAP Root" (which is done by reading an entry with its DN being empty, i.e. DN=""). Details may vary from server to server, mainly depending on the vendor. Refer to the server's original documentation for properties that are not self-explanatory. Note that modifying the LDAP Root is not supported. This application uses LDAP Root itself for a number of purposes; for example, to:

- Locate the server's schema
- Find out what controls the server supports

Open Server/Manage Server Profiles

The "**Open Server**" dialog may be configured in the menu (typically in the file menu). It allows you to open a pre-configured server profile. It also allows you to add new and delete/edit existing server groups profiles. New profiles can also be created based on existing ones ("Copy").

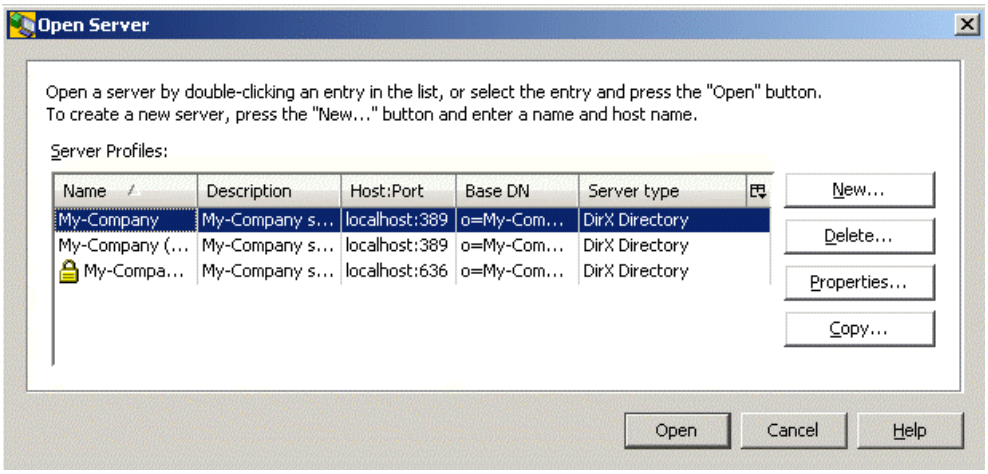


Figure 165. Open Server Dialog Box

The slightly different "**Manage Server Profiles**" dialog may be available in the Login dialog. It has a "Close" button rather than an "Open" and a "Cancel" button, since an open button would not make sense in that context.

7.4. Pitfalls

If the reaction on an operation you have initiated is not as you expected, check if one of the following instances applies:

Login

- IP filtering (to be configured in the LDAP configuration subentry) offers a powerful way to keep clients from connecting to the server from unwanted IP addresses. However, improper settings may lockout authorized users including yourself, too.
  If the DirX Directory Manager plug-in is installed, you may have access to the LDAP configuration subentry from within this application (probably from a different IP address); otherwise use a tool provided by the server like dirxadm
- SSL is a powerful way to secure the client/server communication. It presumes however that a keystore with a suitable certificate is configured.

Retrieval operations

- Access control may hide the existence of any subset of the directory information. So, not finding certain entries or attributes in a read/search/export operation does not necessarily mean that those entries or attributes are not there.
- Access control may cause the entire search functionality to be unavailable. Typical reason: Schema cannot be read.
- If tree browsing does not work, check, if the naming context is administered in the LAP Root DSE.
- If an administrative limit like time or size limit restricts the search result, the entries returned are not necessarily the alphabetically leading ones. So, in this case it is absolutely normal if an entry you are expecting because it matches your search criteria does not appear in the search result list.

Administrative operations like Add/Modify/Rename/Delete

If you run your directory with shadowing in order to balance communications traffic and increase reliability, be aware of the following possible effects: when adding/modifying/renaming/deleting an entry while connected to a shadow DSA, the operation may appear to have failed, since the entry might be missing/out-dated/still there when searching for it later on. This is because administrative operations always affect the master, while the search operation may address a shadow that is not yet up to date

Moreover, if the server does not support simple paging, a delete subtree may have trouble completing at all within a reasonable period of time, since it has to search the shadow for entries to be deleted and might receive the same entries again and again, as long as the shadow is not updated.

These effects are mitigated if you:

- Connect directly to the master (this may however not always be viable).
- Have the master update the shadow "on change". In this case, the shadow should become synchronized rather quickly (this may not always be viable either).
- Set up dedicated LDAP servers to be contacted by administrators and make sure they directly address the master DSA or the service control "Don’t use copy" (corresponding LDAP configuration subentry) is set to "true" for all operations.

8. Trace Window

The trace window allows you to enable, disable and configure trace information and display it in a window. The trace window provides a convenient way to track down and solve problems between this application and the server, either with some assistance from your vendor's support organization, or on your own.

The View → Trace Output operation displays the Trace Output window.

Click the Trace Enabled  button of the Trace Output toolbar to enable or disable displaying of trace information.

Here is an example of a Trace Output window:

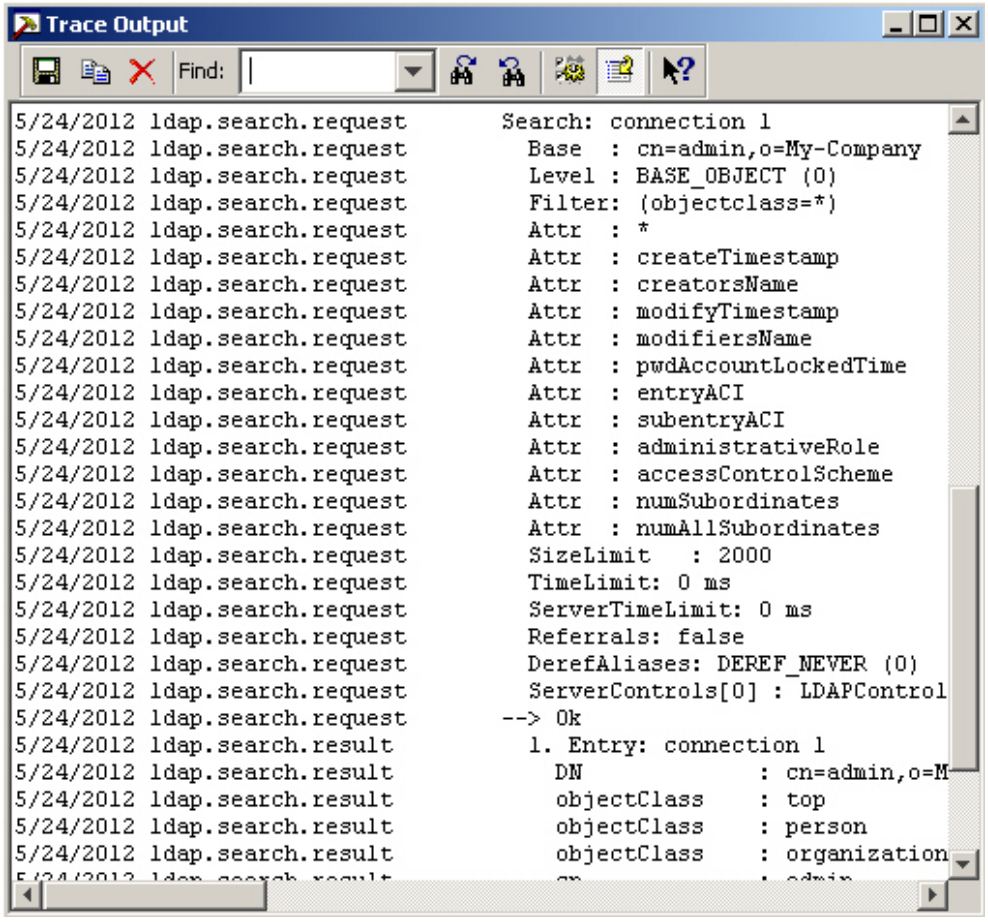


Figure 166. Figure : Trace Output Window

Use the following buttons to:

-  - Specify which trace information is displayed.
-  - Save the trace output in a file.
-  - Copy the trace information to the clipboard.
-

 - Clear the trace information currently displayed in the trace window.

- Search specific information currently displayed in the trace window:

Use the Find Next  button to go to the next occurrence of the specified string.

Use the Find Previous  button to go to the previous occurrence.

DirX Product Suite

The DirX product suite provides the basis for fully integrated identity and access management; it includes the following products, which can be ordered separately.

DirX Identity

DirX Identity provides a comprehensive, process-driven, customizable, cloud-enabled, scalable, and highly available identity management solution for businesses and organizations. It provides overarching, risk-based identity and access governance functionality seamlessly integrated with automated provisioning. Functionality includes lifecycle management for users and roles, cross-platform and rule-based real-time provisioning, web-based self-service functions for users, delegated administration, request workflows, access certification, password management, metadirectory as well as auditing and reporting functionality.

DirX Directory

DirX Directory provides a standards-compliant, high-performance, highly available, highly reliable, highly scalable, and secure LDAP and X.500 Directory Server and LDAP Proxy with very high linear scalability. DirX Directory can serve as an identity store for employees, customers, partners, subscribers, and other IoT entities. It can also serve as a provisioning, access management and metadirectory repository, to provide a single point of access to the information within disparate and heterogeneous directories available in an enterprise network or cloud environment for user management and provisioning.

DirX Access

DirX Access is a comprehensive, cloud-ready, scalable, and highly available access management solution providing policy- and risk-based authentication, authorization based on XACML and federation for Web applications and services. DirX Access delivers single sign-on, versatile authentication including FIDO, identity federation based on SAML, OAuth and OpenID Connect, just-in-time provisioning, entitlement management and policy enforcement for applications and services in the cloud or on-premises.

DirX Audit

DirX Audit provides auditors, security compliance officers and audit administrators with analytical insight and transparency for identity and access. Based on historical identity data and recorded events from the identity and access management processes, DirX Audit allows answering the “what, when, where, who and why” questions of user access and entitlements. DirX Audit features historical views and reports on identity data, a graphical dashboard with drill-down into individual events, an analysis view for filtering, evaluating, correlating, and reviewing of identity-related events and job management for report generation.

For more information: support.dirx.solutions/about

EVIDEN

Legal remarks

On the account of certain regional limitations of sales rights and service availability, we cannot guarantee that all products included in this document are available through the Eviden sales organization worldwide. Availability and packaging may vary by country and is subject to change without prior notice. Some/All of the features and products described herein may not be available locally. The information in this document contains general technical descriptions of specifications and options as well as standard and optional features which do not always have to be present in individual cases. Eviden reserves the right to modify the design, packaging, specifications and options described herein without prior notice. Please contact your local Eviden sales representative for the most current information. Note: Any technical data contained in this document may vary within defined tolerances. Original images always lose a certain amount of detail when reproduced.