EVIDEN

Identity and Access Management

DirX Audit

History Synchronization Guide

Version 7.2, Edition June 2025

Table of Contents

Preface

This manual describes the DirX Audit History Synchronization jobs. It consists of the following chapters:

- Chapter 1 provides an overview of the DirX Audit History Synchronization jobs.
- Chapter 2 describes how to upgrade DirX Audit to use the DirX Audit History Synchronization jobs.
- Chapter 3 describes how to prepare the DirX Identity Store and the DirX Audit History Database for the DirX Audit History Synchronization jobs.
- Chapter 4 describes how to perform advanced configuration and customization of DirX Audit History Synchronization jobs.

DirX Audit Documentation Set

DirX Audit provides a powerful set of documentation that helps you configure your audit server and its applications.

The DirX Audit document set consists of the following manuals:

- *DirX Audit Introduction*. Use this book to obtain a description of DirX Audit architecture and components.
- *DirX Audit Tutorial*. Use this book to get familiar quickly with your DirX Audit installation.
- *DirX Audit Administration Guide*. Use this book to understand the basic tasks of DirX Audit connectivity administration.
- *DirX Audit Manager Classic Guide*. Use this book to obtain a description of the DirX Audit Manager Classic user interface provided with DirX Audit.
- *DirX Audit Manager Guide*. Use this book to obtain a description of the DirX Audit Manager user interface provided with DirX Audit.
- *DirX Audit Command Line Interface Guide*. Use this book to obtain a description of the command line interface provided with DirX Audit.
- *DirX Audit Customization Guide*. Use this book to customize your DirX Audit environment.
- *DirX Audit History Synchronization Guide*. Use this book to obtain information about the DirX Audit History synchronization jobs.
- *DirX Audit Best Practices*. Use this book to obtain guidelines and tips for avoiding common mistakes and improving the experience of a DirX Audit installation.
- *DirX Audit Installation Guide*. Use this book to install DirX Audit.
- *DirX Audit Migration Guide*. Use this book to migrate DirX Audit from previous versions.
- *DirX Audit Release Notes*. Use this book to understand the features and limitations of the current release.
- *DirX Audit History of Changes*. Use this book to understand the features of previous releases.

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.

install_path
The exact name of the root of the directory where DirX Audit programs and files are installed. The default installation directory is *userID_home_directory*/**DirX Audit** on UNIX systems and **C:\Program Files\Atos\DirX Audit** 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*.

install_media
The exact path where the DirX Audit installation media is located.

1. Introduction

As of DirX Audit 7.2, the new DirX Audit History Synchronization jobs replace the DirX Identity synchronization workflows for the DirX Audit History Database.

The DirX Audit History Synchronization jobs run on the DirX Audit Server and operate between a DirX Identity domain and the DirX Audit History Database. The jobs regularly synchronize important domain entries from the DirX Identity domain into the DirX Audit History Database for historical auditing purposes. They do not change anything in the DirX Identity domain.

The DirX Audit History Synchronization jobs maintain snapshots of the DirX Identity domain entries in the DirX Audit History Database. The jobs associate a validity time range (valid from - valid to) with each entry and each attribute. When an attribute value is changed in the DirX Identity domain, the DirX Audit History Synchronization jobs:

- Close the validity period for the original attribute value in the DirX Audit History Database with the entry modification date and time in the DirX Identity domain (**modifyTimestamp** LDAP operational attribute) stored in **valid to**.
- Create a new database record for the current attribute value with the entry modification date and time in the DirX Identity domain (**modifyTimestamp** LDAP operational attribute) stored in **valid from** and an empty **valid to**.

When more entry modifications are performed in a DirX Identity domain between two executions of DirX Audit History Synchronization jobs, only the most recent modification timestamp is reflected.

When an entry is deleted in the DirX Identity domain, the DirX Audit History Synchronization jobs:

- Close the validity period for the deleted entry and all its attribute values in the DirX Audit History Database with the synchronization date and time stored into the **valid to**.

1.1. About the History Synchronization Architecture

The following figure illustrates the architecture of DirX Audit History Synchronization jobs. It shows how the jobs interact with DirX Audit History Database and DirX Identity Store containing an DirX Identity domain:

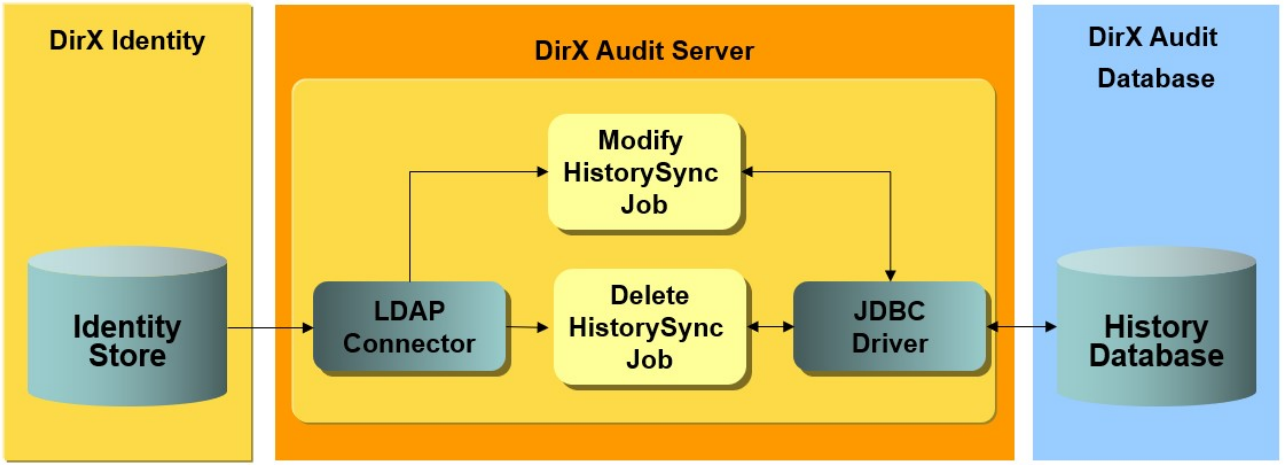

Figure 1. History Synchronization Architecture

As shown in the figure:

- DirX Audit History Synchronization Modify and Delete jobs are components of the DirX Audit Server. Both are configurable scheduled jobs and run on the DirX Audit Server. You can read more about the DirX Audit Server components in section "DirX Audit Components" in the *DirX Audit Introduction*. The jobs use the LDAP connector to read DirX Identity domain entries from the audited system. This connector is included in DirX Audit Server. The jobs use a JDBC driver that is integrated with the DirX Audit Server to read and update audit data in the DirX Audit History Database.
- The DirX Audit History Database contains snapshots of DirX Identity domain. It can be set up as a relational database with Microsoft SQL Server or Oracle Database. You can read more about the DirX Audit History Database in the *DirX Audit Administration Guide*.

1.2. About the History Synchronization Jobs Design

The DirX Audit History Synchronization Modify and Delete jobs share the same configuration and a similar design; the difference is only in the operations they perform on synchronized data. The jobs are designed to run effectively and therefore use parallel data processing.

The DirX Audit History Synchronization Modify and Delete jobs use the tenant **configuration.cfg** file to represent the necessary parameters for synchronization: the definition of the DirX Identity domain connection, the DirX Audit History Database connection and the entry type and attribute-specific configuration; for example, for attribute processing or mapping to the DirX Audit History Database.

The most basic synchronized unit is an entry type represented in DirX Audit History Database. The entry type configuration is included in the tenant **configuration.cfg** file. It contains a description of the synchronized DirX Identity entry type, a specific mapping of DirX Identity domain attributes, and the definitions of virtual attributes. The source of the virtual attributes is not the DirX Identity domain; they are created dynamically during the synchronization process and stored in the DirX Audit History Database. There is also a list of attributes excluded from synchronization. You can read more about configuration in the "Configuring and Customizing DirX Audit History Synchronization Jobs" section of this guide. You can also customize the jobs: you can set up your own custom entry types to synchronize custom DirX Identity domain entries.

The default tenant configuration already contains predefined standard entry types for all important entry type categories, one DirX Audit History Database entry type for each entry type in DirX Identity domain:

- Users and their assignments
- Roles and role parameters
- Permissions
- Groups
- Accounts
- Target systems
- Request workflow and activity definitions
- Request workflow and activity instances
- Business objects: Countries, Organizations, Locations, Organizational Units, Projects, and Cost Units
- Audit policies
- Policies: several kinds of policies like Access, Attribute, Event, Delete, Rules and Operations, Password, Risk, and SoD
- Delegations and access rights
- Tickets
- Certification campaigns, entries, assignment changes, and notifications
- Role parameters
- Domain object, configuration objects, and target system configuration objects

The DirX Audit History Synchronization **Modify job** synchronizes recently-added or modified entries and all their attributes together with validity periods from the DirX Identity domain to the DirX Audit History Database. The job processes entry types and uses the entry type configuration to read a specified set of entries from the DirX Identity domain. The job can also detect relationship changes between entries in the DirX Identity domain and reflect them in the DirX Audit History Database. The DirX Audit History Synchronization Modify job also provides LDAP schema synchronization to the DirX Audit History Database. DirX Audit Manager uses this information to correctly display non-string values.

The DirX Audit History Synchronization **Delete job** detects entries deleted from the DirX Identity domain and closes the validity of entries and their attributes in the DirX Audit History Database in the **valid to** field with the date and time of the job execution. The entry type configuration in the tenant **configuration.cfg** file is used to filter only entries related to synchronized entry type from the DirX Identity domain. This job also maintains relationships between entries in the DirX Audit History Database.

The DirX Audit History Synchronization jobs are designed as scheduled jobs on the DirX Audit Server. They are intended to be run at regular intervals according to a schedule that you set up using the DirX Audit Tenant Configuration Wizard. You can read more about jobs scheduling in the "Configuring and Customizing DirX Audit History Synchronization Jobs" section. Jobs can synchronize one or more predefined entry types or your own custom entry types. If the job is scheduled to synchronize multiple entry types, they are synchronized sequentially one by one to avoid overloading the DirX Audit History Database.

The remainder of this document describes how to prepare the DirX Identity domain and the DirX Audit History Database for DirX Audit History Synchronization jobs, how to configure and customize the DirX Audit History Synchronization jobs, and best practices to follow when setting up the jobs.

2. Installing or Upgrading DirX Audit

This chapter covers three basic scenarios that can occur if you want to prepare the DirX Audit History Database and the DirX Identity domain for the DirX Audit History Synchronization jobs. You can:

- Install DirX Audit from scratch
- Upgrade DirX Audit using the legacy History Database synchronization workflows running on DirX Identity
- Upgrade DirX Audit using the DirX Audit History Synchronization jobs

2.1. Performing a New DirX Audit Installation

A clean installation of DirX Audit automatically prepares the DirX Audit History Database for the DirX Audit History Synchronization jobs. You can read more about DirX Audit installation in the *DirX Audit Installation Guide*. After DirX Audit is installed, perform the following steps to prepare and configure the DirX Audit History Synchronization jobs:

- **Prepare DirX Identity for DirX Audit History Synchronization.** You must configure some parameters of the DirX Identity store for auditing. For more information on DirX Identity configuration, see the section "Preparing the DirX Identity Store".
- **Configure and customize the DirX Audit History Synchronization jobs.** If you do not want to customize the synchronization process, the only required steps are to set up an LDAP connection to your DirX Identity domain and then use the DirX Audit Tenant Configuration Wizard to schedule the DirX Audit History Synchronization jobs for entry types. To schedule the jobs, see the section "Scheduled History Synchronization Jobs Configuration" in the *DirX Audit Installation Guide*. To customize the jobs, see the section "Configuring and Customizing DirX Audit History Synchronization Jobs" in this guide.

2.2. Upgrading DirX Audit Using the Legacy History Database Synchronization Workflows Running on DirX Identity

DirX Audit 7.1 and older uses DirX Audit History Database workflows running on DirX Identity to synchronize DirX Identity domain entries to the DirX Audit History Database. The DirX Audit History Synchronization jobs completely replace this functionality in DirX Audit 7.2. To upgrade to the new DirX Audit History Synchronization jobs, perform the following steps:

- **Prepare the DirX Audit History Database for DirX Audit History Synchronization jobs.** You can read more about this step in the section "Preparing the DirX Audit History Database".
- **Prepare DirX Identity for the DirX Audit History Synchronization jobs.** You can read more about this step in the section "Preparing the DirX Identity Store".

- **Back up the legacy DirX Audit History Database synchronization workflows customization.** We recommend that you check the legacy DirX Audit History Database synchronization workflows configuration objects like channels, workflows, and schedules in DirX Identity and back up all customizations; for example, custom entry types, LDAP schema extensions or custom attributes mapping to the DirX Audit History Database.
- **Remove the legacy configuration objects from DirX Identity.** You can perform this step in DirX Identity Manager's Connectivity view. To remove the legacy DirX Audit History Database synchronization workflows and connected directories:
  - Delete the corresponding objects in the scenario of the Global View.
  - Delete the workflows in the Expert View. Consider both the full and delta workflows.
  - Delete the connected directory for the DirX Audit History Database in the Expert View. Disable the **Check references to avoid broken links** option before confirming the deletion.
  - Delete the folder for the History Database channels in the Expert View beneath the connected directory representing the DirX Identity domain.
- **Apply any customizations to the DirX Audit History Synchronization jobs.** Modify the tenant **configuration.cfg** file to accomplish this task. See the section "Configuring and Customizing the DirX Audit History Synchronization Jobs" for details.
- **Configure the DirX Audit History Synchronization jobs.** Use the Tenant Configuration Wizard to accomplish this task. See the section "Scheduled History Synchronization Jobs Configuration" in the *DirX Audit Installation Guide* for details.

DirX Audit History Synchronization represents all large binary attributes as Base64 string values. On the other hand, the legacy DirX Audit History Database workflows in some special cases store binary values as strings without encoding them. In such cases, DirX Audit History Synchronization terminates the time validity of the large attribute and inserts its new Base64-encoded value.

2.3. Upgrading DirX Audit Using the DirX Audit History Synchronization Jobs

This scenario assumes that you already use DirX Audit 7.2 or newer with DirX Audit History Synchronization jobs and your DirX Audit History Database and DirX Identity domain are correctly set up. During DirX Audit tenant configuration, an automatic procedure runs and migrates your configuration and customization of DirX Audit History Synchronization jobs. You can read more about the migration process in the *DirX Audit Migration Guide*.

3. Preparing the DirX Audit History Database and the DirX Identity Store for the DirX Audit History Synchronization Jobs

This chapter describes how to prepare the DirX Identity Store and the DirX Audit History Database for the DirX Audit History Synchronization jobs. Before you begin with DirX Identity Store and DirX History Database configuration for DirX Audit History Synchronization jobs, you must have:

- A functional DirX Audit History Database
- An operational DirX Identity domain

3.1. Preparing the DirX Audit History Database

If you are upgrading from an older version of DirX Audit, there may be some duplicates in the DirX Audit History Database. You must remove all these duplicates from the DirX Audit History Database before migrating to DirX Audit 7.2 version or newer. You can use the tool **dxthistdbtool** to remove all duplicates, using the command **makeunique**. For more information about the tool, see the section "Using the DirX Audit Tools" in the *Dirx Audit User Interface Guide*.

The next step is to migrate the DirX Audit History Database. For more information on migrating the History Database, see the section "Manual Migration" in the *DirX Audit Migration Guide*.

3.2. Preparing the DirX Identity Store

The DirX Audit History Synchronization jobs require an attribute in all DirX Identity domain entries that must be unique, always exists and does not change over time. The DirX Audit History Synchronization jobs use the **dirxEntryUUID** operational attribute, which is created with a unique value by the DirX Directory server and exists for all LDAP entries. This attribute is necessary for identifying the corresponding entry in the DirX Audit History Database. It is often referred to as "join an entry from the DirX Identity domain to the DirX Audit History Database". You must make this attribute readable by the technical account reading DirX Identity domain entries used by the DirX Audit History Synchronization jobs.

The DirX Audit History Synchronization Delete job sorts the result sets by the dirxEntryUUID attribute. You must ensure that there is an index for the dirxEntryUUID operational attribute. Please follow the next sections to meet these conditions.

3.2.1. Making dirxEntryUUID Unique

In some scenarios, it may accidently happen that there are DirX Identity domain entries with duplicate dirxEntryUUID values. Duplicates must be removed before starting to use the History Synchronization jobs. You can use the **dxthistdbtool** tool to remove all dirxEntryUUID duplicates from DirX Identity domain, using the command **ldapremdup**. For more information about this tool, see "Using the DirX Audit Tools" in the *DirX Audit User Interface Guide*.

The next step is to apply a unique constraint in DirX Directory for the dirxEntryUUID attribute to prevent duplicate values. You can use **dirxadm.exe** command line tool for this task:

- Navigate to the *dxd_install_path***bin** folder.
- Open command line and run **dirxadm.exe**.
- Bind first:
  `dirxadm> bind -host localhost -user /O=My-Company/CN=admin -authentication simple -password` *pwd*
- Check unique contstraint checking:
  `dirxadm> show / -attr enui -p`
  If unique constraint is enabled, the operation returns:
  ```
  1) /
  Enable-Unique-Index : TRUE
  ```
  If unique constraint is disabled, the operation returns /
- If disabled, enable the unique constraint checking:
  `dirxadm> modify / -add ENUI=TRUE`
- Show the current attribute and indexes for dirxEntryUUID (**-attr** is **DUUID**):
  `dirxadm> db show -attr DUUID`
  The operation returns:
  ```
  ATTR=DUUID,INDEX=INITIAL,OPTR=TRUE
  ```
- Add the unique constraint if not yet set:
  `dirxadm> db attrconfig DUUID -index UNIQUE`
- Check the attribute configuration:
  `dirxadm> db show -attr DUUID`
  If everything is set correctly, the operation returns:
  ```
  {ATTR=DUUID,INDEX=INITIAL;UNIQUE,OPTR=TRUE}
  ```

3.2.2. Making dirxEntryUUID Indexed

To run the History Synchronization Delete job, you must enable sorting entries by setting the index for the dirxEntryUUID attribute. Use DirX Directory Manager (not DirX Identity Manager) to perform this task as follows:

- Log in with sufficient access rights, for example, as system administrator.
- In the **Browse** view of the **Schema** view, select the **Database** entry.
- In **Edit** mode, uncheck **Hide attributes with no index assigned** to see all the attributes including **dirxEntryUUID** in the **Indices** tab. Check the **Initial Index** flag for this attribute.
- **Save** your changes.

3.2.3. Making dirxEntryUUID Readable for the Technical Account

By default, a DirX Identity technical account is not allowed to read the **dirxEntryUUID** attribute. You must create a DirX Directory access control subentry to allow it. As of DirX Identity V8.10, the DirX Identity installation creates this access policy automatically.

If you use an older DirX Identity version, you must make dirxEntryUUID readable for the technical account accessing the DirX Identity domain manually. Use DirX Directory Manager (not DirX Identity Manager) as follows:

- Log in to DirX Manager with sufficient access rights, for example, as system administrator (cn=SystemAdmin, cn=DirXmetaRole-SystemDomain).
- In the **Configuration** view of the domain, select the entry **Access Control Subentries** below the domain root, for example, `cn=My-Company`. From the context menu, select **New Access control subentry ...** and follow the wizard.
- Enter a meaningful **Common name**, for example, readDirXEntryUUID, with the domain root as the administrative point and then click **OK**.
- In the next wizard step, select the **Prescriptive ACI** tab and then click **New ...**
- In the wizard step **Type of ACI Item**, select, for example **User classes and their permissions (UserFirst)** and then click **Next**.
- In the wizard step **General**, set a meaningful text in **Identification tag**, set the authentication level (for example, **simple**) and then click **Next**.
- In the wizard step **User Classes**, select the user group **DomainAdmins** from the target system **DirXmetaRole** (or an appropriate one where the bind user of the workflow is a member) and then click **Next**.
- In **User Permissions**, in the **Protected Items** tab, select the attribute type **dirxEntryUUID** and check the flag **Same as in "Attribute types"**. As **Permissions**, select: **Compare**, **DiscloseOnError**, **FilterMatch** and **Read** into the **Granted permissions**. Click **OK**. Click **Next**.
- In the wizard step **Summary**, click **Finish** to close the wizard.
- Click **OK** to save your changes.

3.3. Setting Up Correct Collation

The DirX Audit History Synchronization Delete job requires the same collation to be set on both the DirX Directory and the DirX Audit History Database in order to function properly. To check the matching rule for the **dirxEntryUUID** attribute that is used for DirX Directory server-side sorting by the DirX Audit History Synchronization Delete job, open the **Schema** view in DirX Directory Manager and then select **dirxEntryUUID** from the **Schema - Attributes** list. You should keep the DirX Directory settings and adjust the collation on the DirX Audit History Database side. For example, if your matching rule for **Equality** is **caseIgnoreMatch**, use case-insensitive collation on your DirX Audit History Database.

4. Configuring and Customizing DirX Audit History Synchronization Jobs

You can perform basic configuration operations like jobs scheduling and setting up DirX Audit History Database and DirX Identity domain connections in the DirX Audit Tenant Configuration Wizard. Each job can synchronize one or more entry types. See the section "Scheduled History Synchronization Jobs Configuration" in the *DirX Audit Installation Guide* for details.

This chapter describes how to perform advanced History Synchronization configuration and customization such as custom entry type configuration, entry type-specific attribute mapping, and advanced processing configuration by modifying the tenant-specific **configuration.cfg** file. See the next sections for details.

4.1. Navigating in the Tenant Properties File

There are two basic types of tenant **configuration.cfg** file:

- The **tenant default configuration.cfg file**, which contains the default tenant configuration. The default configuration may be changed by a DirX Audit Server update. You should never change this file, but you can check it to the see the default DirX Audit History Synchronization configuration:

  *install_path***/conf/defaults/tenant/configuration.cfg**
- The **tenant-specific configuration.cfg** file, which contains only modifications from the tenant default file and is never affected by a DirX Audit Server update. Each tenant has its own tenant-specific file, located at:

  *install_path***/conf/tenants/***tenant_id***/configuration.cfg**

If you have not modified any history synchronization settings either manually or via the configurator, you will not find a **[historysync.*]** section in your tenant-specific file. In this case, you should copy the relevant section from the default file and then modify it in your tenant-specific configuration - that is, if you want to customize your history synchronization settings for the specific tenant.

Both **configuration.cfg** files have the same structure. They are divided into sections. Each section starts with the section name enclosed in square brackets. Configuration under the section name belongs to the section. Some sections are divided into subsections. Section names are hierarchical with the following name conventions:

[*<section_name>.<subsection_name>***]**

For example, the section dedicated to entry types has the following name:

```
[historysync.objects]
```

The file sections contain key-value pairs. Some keys or values are multi-value and separated by **@**. Some values can be lists. All lists of values are separated by commas and without spaces. If you define a list with no items, an empty list is passed as a parameter. Some values are optional and can be omitted.

You can use **#** at the beginning of the line to write a comment. For example:

```
#your comment
```

The tenant file contains two types of DirX Audit History entry attribute names for History Synchronization:

- A DirX Identity attribute name, which is an existing DirX Identity domain attribute name (LDAP attribute). Use this attribute name only within the common attribute configuration section **`[historysync.attributes]`** for setting up the source attribute name and in lists of synchronized attributes within the entry type configuration **`[historysync.objects]`**.
- A DirX Audit History Database attribute name, which is used in the DirX Audit History Database. By default, it is the same name as the DirX Identity domain attribute name. You can customize it and use your own attribute name in the DirX Audit History Database that is unique within the tenant file. A DirX Identity attribute name is mapped to this name, and you must use this name in the **configuration.cfg** file for the rest of the History Synchronization configuration tasks. For more details, see the section "Attribute-Specific Configuration".

The next sections describe the structure of the tenant-specific **configuration.cfg** file sections related to DirX Audit History Synchronization and how to modify them. If you want to modify a section, first check the tenant-specific file. If the section is already there, you can just change it to your desired configuration. If the section is missing, add it to the tenant-specific file together with all configuration key-value pairs from the tenant default file and then change them to your desired configuration.

4.2. Entry Type Configuration

The common entry type configuration is located in the subsection **`[historysync.objects]`**:

```
[historysync.objects]
attributes = *
excluded_attributes =
 dxmVersion,dxrAssignmentDetails,dxrTBA,dxrVersion
attr_values_ignore_case =
```

Configuration keys are:

- **attributes** - The default list of synchronized DirX Identity domain attribute names. You can use the asterisk character (*****) as a wildcard for all attributes available in the DirX Identity domain. Operational attributes are not automatically obtained from the DirX Identity domain. If you want to synchronize them, you must explicitly write them to this list. Normally, you do not need to change the default value. This value is used only for entry types without synchronized attributes definitions.
- **excluded_attributes** - The default list of DirX Audit History Database attribute names excluded from the synchronization. This value is used only for entry types without excluded attribute names definitions. The excluded attributes take precedence over attributes.
- **attr_values_ignore_case** - The default list of DirX Audit History Database attribute names of the common attributes whose values are case insensitive-compared during the attribute joining process. By default, attribute values are case sensitive-compared.

Subsections that define specific entry type configurations are named as follows:

[historysync.objects.<*object_type_name***>]**

Specific History Synchronization entry type configuration sections are predefined for all standard DirX Identity domain entry types. You can add your own custom entry type if needed. All entry types share a similar configuration structure, as shown in the following example for the account entry type:

```
[historysync.objects.account]
name = Account
hdb_name = Account
ldap_search_base_prefix = cn=TargetSystems
ldap_search_filter = (objectClass=dxrTargetSystemAccount)
excluded_attributes =
attributes =
```

Configuration keys are:

- **name** - The name of the entry type that is shown in DirX Audit Tenant Configuration Wizard for the tenant.
- **hdb_name** - The name of the entry type that is stored in DirX Audit History Database. Usually, it is the same value as **name**.
- **ldap_search_base_prefix** - The LDAP search base RDN that defines the synchronized subtree of the DirX Identity domain. The RDN is automatically extended with the DirX Identity domain root entry; for example, `cn=My-Company`. The value is optional.
- **ldap_search_filter** - The LDAP filter to restrict for the requested DirX Identity entries only. Usually, the objectClass attribute value is considered here. The value is optional.

- **excluded_attributes** - The specific list of DirX Audit History Database attribute names excluded from the synchronization for the entry type. The value is optional. The excluded attributes take precedence over attributes.
- **attributes** - The list of synchronized DirX Identity domain attribute names. You can use the asterisk character (*****) as a wildcard for all attributes available in the DirX Identity domain. You can explicitly define only those attributes for the entry type that you want the History Synchronization job to process. The value is optional.

4.3. Attribute-Specific Configuration

Entry attributes are divided into three basic categories in the DirX Audit History Database:

- Small attributes - attributes with attribute value lengths up to 850 characters.
- Large attributes - attributes with attribute value lengths larger than 850 characters. Large attributes are not currently displayed in DirX Audit Manager and are therefore excluded from synchronization by default. You can change this behavior by changing the large attributes mapping. See the section "Customizing Attribute Mapping".
- Link attributes - links to other entries, represented as DN or RDN. The link length can also be up to 850 characters.

If you try to store attribute values greater than 850 characters to the small attributes database table, the value is automatically truncated to 850 characters and only the prefix of the value is stored. You can detect this situation in the DirX Audit Server log file and adjust the attribute mapping accordingly.

Common attribute-specific configuration is in the subsection **`[historysync.attributes]`**. The following example shows how a common attribute-specific configuration might look and illustrates which attribute name is an existing name of the attribute in DirX Identity and which attribute name is used in the DirX Audit History Database.

```
[historysync.attributes]
<large_attribute_database_name>@large =
<large_attribute_identity_name>
<small_attribute_database_name> = <small_attribute_identity_name>
dxtTargetSystemLink@link = @virtual=exclusive@class=
com.dirxcloud.audit.historysync.modifydelimpl.configuration.
attributeprocessors.MapToTs@categcat=DN@visibility=1
```

You can see that the configuration key consists of the DirX Audit History Database attribute name (for example, *large_attribute_database_name*) and an optional attribute mapping category (for example, **@large**). If there is no mapping category, the attribute is automatically mapped to the DirX Audit History Database small attributes table. The DirX Audit History Database attribute name is normally identical to the attribute name in the DirX Identity domain, but you can customize it according to your needs. It must be unique within the tenant **configuration.cfg** file.

The configuration value may consist of multiple parameters such as the attribute name from DirX Identity, attribute processing, and others. There are no special requirements on the order of the parameters; you must only put the attribute DirX Identity name first. If you are configuring a virtual attribute, which is created during the synchronization process and has no corresponding DirX Identity domain attribute name, your configuration value must start with the **at** sign (**@**). The structure of the configuration value with multiple parameters is:

<*attribute_identity_name***>@<***optional_parameter_1***>@<***optional_parameter_n***>**

You can use the following parameter names (a vertical bar indicates the option choices):

- **virtual = common** | **exclusive** - This parameter defines whether the virtual attribute is common or exclusive. If you are configuring a virtual attribute, you must omit the DirX Identity domain attribute name. Common virtual attributes are automatically created for all entry types. Exclusive virtual attributes are created only for selected entry types.
- **class = <***fully_qualified_class_name***>** - This parameter specifies the fully qualified class name of the class processing attribute values. It is optional for non-virtual attributes and mandatory for virtual attributes.
- **category = DN** | **STRING** | **INTEGER** | **BINARY** - This parameter defines the attribute's category. This configuration is used by DirX Audit Manager to display the attribute correctly.
- **valueci** - This parameter indicates that a case-insensitive compare of attribute values should be performed during the attribute joining process.
- **visibility = 1** | **0** - This parameter indicates to DirX Audit Manager whether the attribute value should be presented. **1** (visible) is the default value.

The tenant **configuration.cfg** file also contains entry type-specific attribute configurations. You can set up attribute mapping specifically for a selected entry type in the section that contains the entry type name in the configuration section name. For example, the entry type-specific attribute configuration for accounts looks like this:

```
[historysync.attributes.account]
dxtTargetSystemLink =
dxtOrphaned =
```

The section defines mapping for two attributes. The **dxtTargetSystemLink** and **dxtOrphaned** are DirX Audit History Database virtual attribute names. These attributes are in the entry type-specific attribute configuration and therefore their values are calculated for all accounts during account entry type synchronization. The mapping is not defined here because the common mapping configuration is used. You can override it here if needed.

4.4. Configuring LDAP Schema Custom Data Types

If your DirX Identity domain uses custom attributes with a custom data type, you must configure this data type for DirX Audit. If it is not configured properly, you can find the following message in the DirX Audit Server log file: "There is no mapping for specified syntax id: ..."

This message indicates a missing LDAP data type in the DirX Audit configuration. You can fix it by modifying the file:

*install_path***/conf/historysync/ldapSyntaxTypes.properties**

Add a new line to the end of the file and insert your custom data type in the following way:

```
1.3.12.2.1107.1.3.5.8=STRING
```

where:

- **1.3.12.2.1107.1.3.5.8** is the LDAP schema object identifier (OID) as available in DirX Directory.
- **STRING** | **DATE** | **BINARY** | **BOOLEAN** is the data type name. Use **BINARY** for binary data.

4.5. Customizing DirX Audit History Synchronization

This section describes four methods for customizing DirX Audit History Synchronization:

- Modifying, adding, and removing entry types from History Synchronization
- Customizing attribute mapping
- Excluding attributes from synchronization
- Explicitly specifying attributes for synchronization

4.5.1. Modifying, Adding, and Removing an Entry Type from the Synchronization

To modify an existing entry type configuration, search the tenant **configuration.cfg** file for the corresponding section. For example, to modify the account entry type, search for the **`historysync.objects.account`** section.

```
[historysync.objects.account]
name = Account
hdb_name = Account
ldap_search_base_prefix = cn=TargetSystems
ldap_search_filter = (objectClass=dxrTargetSystemAccount)
```

You can change values both for the **ldap_search_base_prefix** key and the **ldap_search_filter key** to restrict account entries to be synchronized from the DirX Identity domain. You can also extend the list of excluded attributes by adding the **excluded_attributes** key and a list of attribute names. Please consider that there are already several attribute names listed in the common configuration section `historysync.objects`.

Your DirX Identity domain may include one or more custom entry types that you need to synchronize into the DirX Audit History Database. To add your custom entry type, add the following section under the last entry type configuration section in your tenant **configuration.cfg** file:

[historysync.objects.<*entrytypename***>]**
name = <*EntryTypeName***>**
hdb_name = <*HistoryDatabaseEntryTypeName***>**
ldap_search_base_prefix = <*ldap-search-base-prefix***>**
ldap_search_filter = <*ldap-search-filter***>**
excluded_attributes = <*a-list-of-excluded-attributes***>**
attributes = <*a-list-of-included-attributes***>**

You must define a tenant-unique *entrytypename*, a **name**, and an **hdb_name**. Specify the **ldap_search_base_prefix** as a relative distinguished name, not including the root DirX Identity domain object. Specify the **ldap_search_filter** to define a set of synchronized DirX Identity domain entries. You can also add lists of attributes to be excluded from and included in the synchronization.

To remove the existing entry type configuration, search the tenant **configuration.cfg** file and remove the corresponding configuration section.

4.5.2. Customizing Attribute Mapping

DirX Audit History Synchronization jobs synchronize DirX Identity entry attributes to three different tables in the DirX Audit History Database. Binary data and attributes with values longer than 850 characters should be stored in the large attributes table. Links to other DirX Identity entries should be stored in the link attributes table. The rest of the attributes are automatically mapped to the small attributes table.

If you want to set up attribute mapping, you can do it globally for all entry types within the tenant, or you can adjust mapping only for the specified entry type.

To change the attribute mapping globally, for all entry types, go to the tenant-specific **configuration.cfg** file and find the section:

[historysync.attributes]

Search for the attribute name within this section. If mapping for your attribute is already defined, you can just modify it. For example, to change mapping from the small attributes table to the large attributes table, change the original definition:

```
<attribute_database_name>@small = <attribute_identity_name>
```

to:

```
<attribute_database_name>@large = <attribute_identity_name>
```

If you cannot find the configured attribute within this section, add the following line to the end of the attributes section:

```
<attribute_database_name>@small = <attribute_identity_name>
```

where:

- *attribute_database_name* is the attribute name in the DirX Audit History Database. You can use the same name as in DirX Identity or define your own attribute name.
- *attribute_identity_name* is the attribute name as it appears in the DirX Identity LDAP schema.
- **small** | **link** | **large** is the attribute mapping specifying the target database table.

To change the attribute mapping only for a specific entry type, search the tenant configuration file for the section:

[historysync.attributes.<*specific_entry_type***>]**

where:

- *specific_entry_type* - Name of the specific entry type.

If there is no such section, you must add it. Best practice is to include this section under the other sections for the entry type-specific attribute mapping.

The entry type-specific attribute configuration section must contain an attribute mapping configuration for your attribute. If there is no attribute mapping configuration within this section, add it. In this example, the attribute is mapped to the large attributes table:

```
<attribute_database_name> = large
```

If you are configuring virtual attributes, you must add the virtual attribute to the entry-specific attribute mapping even if you do not want to override the common attribute mapping. History Synchronization uses this information to generate virtual attributes for all entries related to the entry type. If you do not want to override the default virtual attribute mapping, you can leave the value empty. In this case, the common attribute mapping is used to map the virtual attribute:

```
<virtual_attribute_name> =
```

4.5.3. Excluding Attributes from the Synchronization

You can exclude an attribute from synchronization to the DirX Audit History Database. Excluded attributes are defined for entry types. You can also use the attribute mapping to exclude attributes.

To exclude an attribute by modifying the entry type, find the entry type section in the tenant **configuration.cfg** file. You can also define it in the default excluded attributes list, for example:

```
[historysync.objects]
excluded_attributes =
 dxmVersion,dxrAssignmentDetails,dxrTBA,dxrVersion
```

This list is overridden by the entry type-specific configuration. For example, for the following account entry type configuration:

```
[historysync.objects.account]
excluded_attributes = dxrTBA,dxrVersion
```

the default excluded attributes list is overridden because the account entry type has its own list of excluded attributes. To exclude the dxmVersion attribute, for example, from the account synchronization, modify the configuration as follows:

```
[historysync.objects.account]
excluded_attributes = dxmVersion,dxrTBA,dxrVersion
```

To exclude an attribute from the synchronization by modifying the attribute mapping, set the mapping value to excluded:

<attribute_database_name> **= excluded**

You can learn more about attribute mapping customization in the section "Customizing Attribute Mapping".

4.5.4. Explicitly Specifying Attributes for the Synchronization

By default, DirX Audit History Synchronization jobs synchronize all available DirX Identity entry attributes for most of the predefined entry types. However, some attributes are excluded by default, which means that they are not stored in the DirX Audit History Database. There are two options to control which attribute is synchronized into the DirX Audit History Database:

- Exclude the attribute from the synchronization - The attribute can be obtained from the DirX Identity domain by the DirX Audit History Synchronization job, but it is not stored in the DirX Audit History Database.
- Define the entry type-specific list of synchronized attributes - Only attributes explicitly written to this list are obtained from DirX Identity domain by the DirX Audit History Synchronization job. These attributes can still be excluded and not stored in the DirX Audit History Database if they are set up as excluded attributes.

There are some entry types like **User** for which you want to synchronize only the explicitly-defined set of DirX Identity entry attributes; for example, to restrict the amount of audited personal data to the necessary minimum. You can extend the entry type configuration to define the synchronized DirX Identity entry attributes list as shown below:

```
[historysync.objects.user]
name = User
hdb_name = User
ldap_search_base_prefix =
ldap_search_filter = (objectClass=dxrUser)
# synchronized user attributes
attributes = employeeNumber,sn,mail
excluded_attributes =
```

In this example, only the DirX Identity **employeeNumber**, **sn**, **mail** attributes are read from the DirX Identity domain by the History Synchronization jobs and synchronized into the DirX Audit History Database.

4.6. Transferring Customizations from Old History Synchronization Workflows

The DirX Audit History Synchronization jobs use a similar default configuration as the legacy DirX Audit History synchronization workflows in DirX Audit 7.1. You can check the default tenant **configuration.cfg** file to see the default DirX Audit History Synchronization configuration:

install_path/**DirX Audit/conf/defaults/tenant/configuration.cfg**

If you used the legacy DirX Audit History synchronization workflows with a customization such as customer-specific entry types or customer-specific schema extensions, you must document the difference between your customization and the DirX Audit History synchronization workflows' default configuration and then apply it to the tenant-specific **configuration.cfg** file to add it as a customization of the new DirX Audit History Synchronization jobs. For details on how to apply your customizations to the DirX Audit History Synchronization jobs, see the section "Customizing DirX Audit History Synchronization" in this guide.

During the migration from DirX Audit version 7.1 to 7.2, the Configuration Wizard attempts to migrate all configuration automatically, so ideally, no further customer actions are required. However, it is recommended to check the new DirX Audit History Synchronization setting and compare it with the previous setting to ensure the same behavior.

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

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.