EVIDEN

Identity and Access Management

DirX Identity

Real-time Synchronization within an Identity Domain

Version 8.10.12, Edition August 2025

Table of Contents

Copyright ii
Preface 1
DirX Identity Documentation Set 2
Notation Conventions 3
1. Overview 5
2. Requirements 6
3. Differences from Conventional Workflows 8
4. Creating the Workflow 10
4.1. Setup Prerequisites 10
4.2. Setup Steps 10
4.2.1. Create the LDAP Synchronization Workflow 10
4.2.2. Create the Channel Folders 10
4.2.3. Create the Channels 11
4.2.4. Configure the IdentityDomain Port 12
4.2.5. Configure the TS Port 12
4.2.6. Configure the Channels 12
4.2.7. Configure the Controller 13
4.2.8. Configure the Workflow Object 13
4.2.9. Assign the New Workflow to the Global View 14
4.2.10. How to send Provisioning Events 14
Legal Remarks 17

Preface

This document presents a use case that illustrates how to use the extension to the real-time synchronization workflow functionality for supporting real-time synchronization within a DirX Identity domain. This feature has been implemented in DirX Identity in response to a customer request.

The document describes the requirements for the extension, how it differs from the conventional real-time workflow approach, and how to configure it. It consists of the following chapters:

- Chapter 1 provides an overview of the use case.
- Chapter 2 explains its requirements.
- Chapter 3 explains the differences between this real-time workflow model and the conventional approach.
- Chapter 4 describes how to configure the new real-time workflow.

DirX Identity Documentation Set

The DirX Identity document set consists of the following manuals:

- *DirX Identity Introduction*. Use this book to obtain a description of DirX Identity architecture and components.
- *DirX Identity Release Notes*. Use this book to understand the features and limitations of the current release. This document is shipped with the DirX Identity installation as the file **release-notes.pdf**.
- *DirX Identity History of Changes*. Use this book to understand the features of previous releases. This document is shipped with the DirX Identity installation as the file **history-of-changes.pdf**.
- *DirX Identity Tutorial*. Use this book to get familiar quickly with your DirX Identity installation.
- *DirX Identity Provisioning Administration Guide*. Use this book to obtain a description of DirX Identity provisioning architecture and components and to understand the basic tasks of DirX Identity provisioning administration using DirX Identity Manager.
- *DirX Identity Connectivity Administration Guide*. Use this book to obtain a description of DirX Identity connectivity architecture and components and to understand the basic tasks of DirX Identity connectivity administration using DirX Identity Manager.
- *DirX Identity User Interfaces Guide*. Use this book to obtain a description of the user interfaces provided with DirX Identity.
- *DirX Identity Application Development Guide*. Use this book to obtain information how to extend DirX Identity and to use the default applications.
- *DirX Identity Customization Guide*. Use this book to customize your DirX Identity environment.
- *DirX Identity Integration Framework*. Use this book to understand the DirX Identity framework and to obtain a description how to extend DirX Identity.
- *DirX Identity Web Center Reference*. Use this book to obtain reference information about the DirX Identity Web Center.
- *DirX Identity Web Center Customization Guide*. Use this book to obtain information how to customize the DirX Identity Web Center.
- *DirX Identity Meta Controller Reference*. Use this book to obtain reference information about the DirX Identity meta controller and its associated command-line programs and files.
- *DirX Identity Connectivity Reference*. Use this book to obtain reference information about the DirX Identity agent programs, scripts, and files.
- *DirX Identity Troubleshooting Guide*. Use this book to track down and solve problems in your DirX Identity installation.
- *DirX Identity Installation Guide*. Use this book to install DirX Identity.
- *DirX Identity Migration Guide*. Use this book to migrate from previous versions.

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. Overview

DirX Identity allows extensive customization of its features using various methods; for example, with schema extensions and object descriptions and by setting options and parameters through wizards or object pages.

DirX Identity now provides a method in the Java-based real-time workflow feature set that allows you to synchronize data from a customer-specific contract or ticket store to the DirX Identity user store without having to set up and configure a separate target system for the store. The customer-specific contract or ticket store acts as the source for synchronization.

Using this approach avoids additional configuration and data synchronization and can be used to synchronize two different subtrees within a DirX Identity domain.

For detailed information about Java-based real-time workflows and how to configure them, see the following documents:

DirX Identity Connectivity Administration Guide

DirX Identity Customization Guide

2. Requirements

According to the requirement specified in CR 25787, there is a customer-specific contract or ticket store that should act as the source for synchronization to the DirX Identity user store. Synchronization should be started by **TOPIC_USER_CHANGE** events. For example, if a new contract is added, the workflow should be started via the **TOPIC_USER_CHANGE** event; the workflow then copies the contract object to the user tree, allowing for a complex mapping scheme to be applied.

This approach has two requirements:

- The workflow should synchronize objects inside the Identity Store; that is, both the source and the target are the Identity Store. The source objects reside in a source subtree and the target objects reside in target subtree.
- The workflow should be started via a **dxm.event.ebr** event.

The following figures illustrate the configuration of the workflow and its join activity.

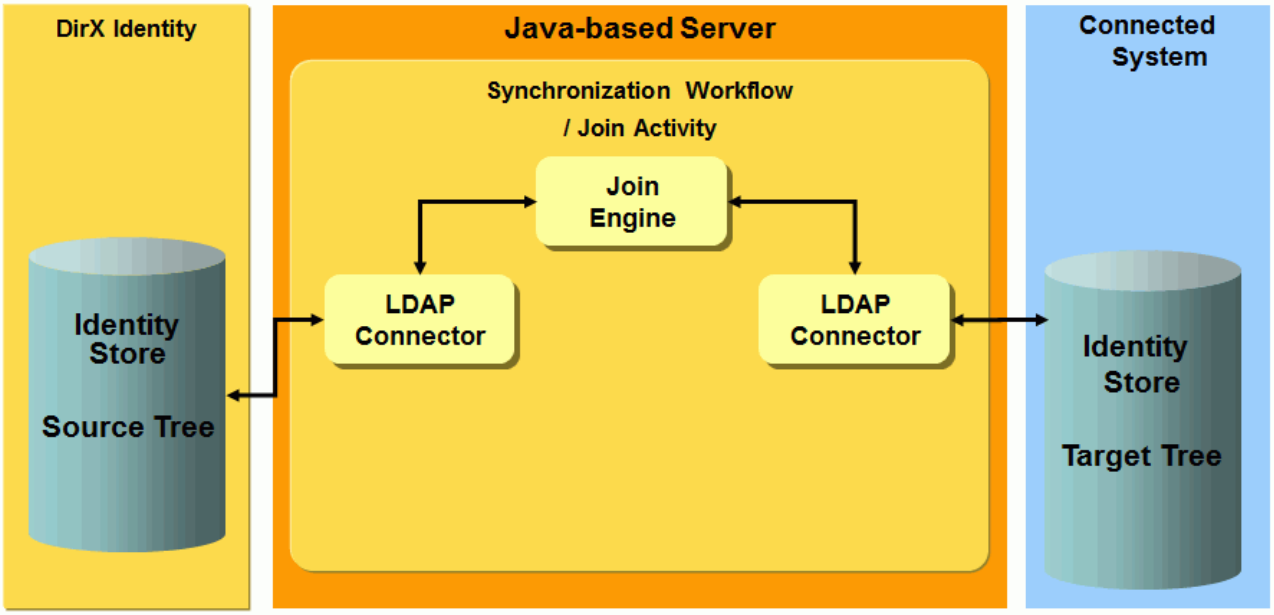

Figure 1. Real-time Synchronization Workflow Components

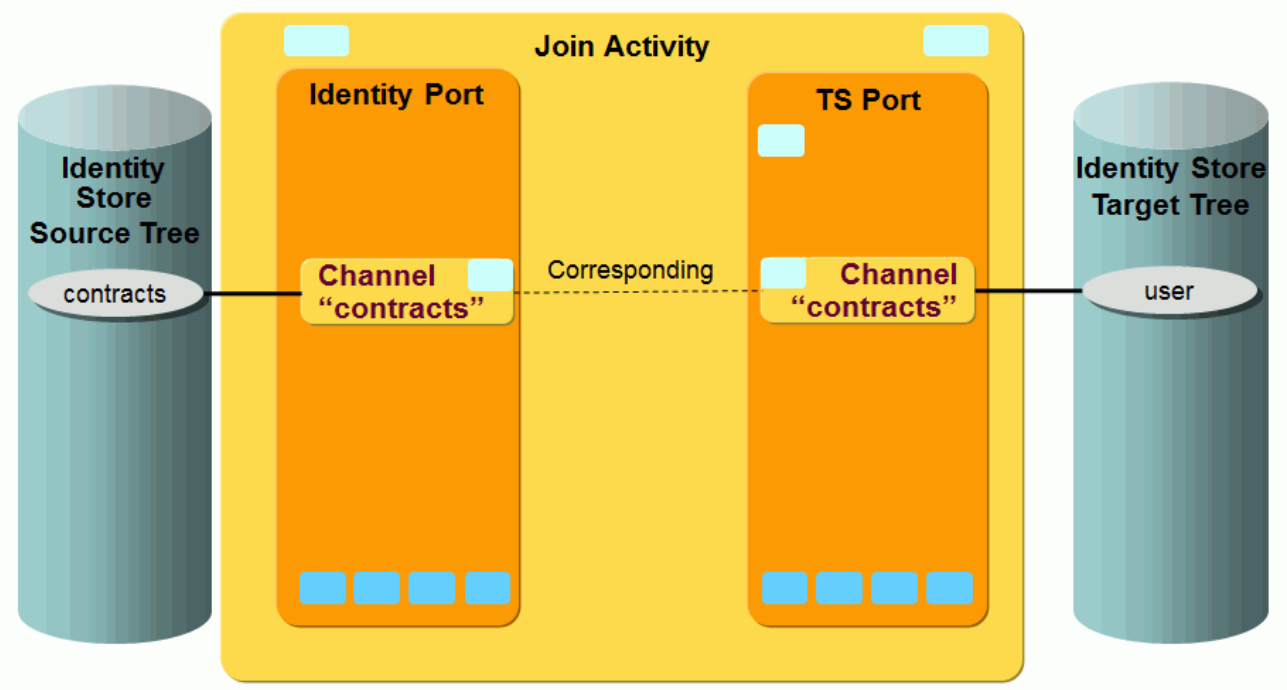

Figure 2. Real-time Synchronization Workflow Join Activity

3. Differences from Conventional Workflows

The conventional Java-based real-time workflows operate between the Identity Store and a connected directory that represents the connected system. As a result, an additional connected directory, including channels for accounts, groups, members and passwords exists. This connected directory corresponds to a target system in the Provisioning view.

The following screenshot highlights these elements, which are located under the **testLDAP** node for this use case, as displayed in the DirX Identity Manager's Connectivity Expert View.

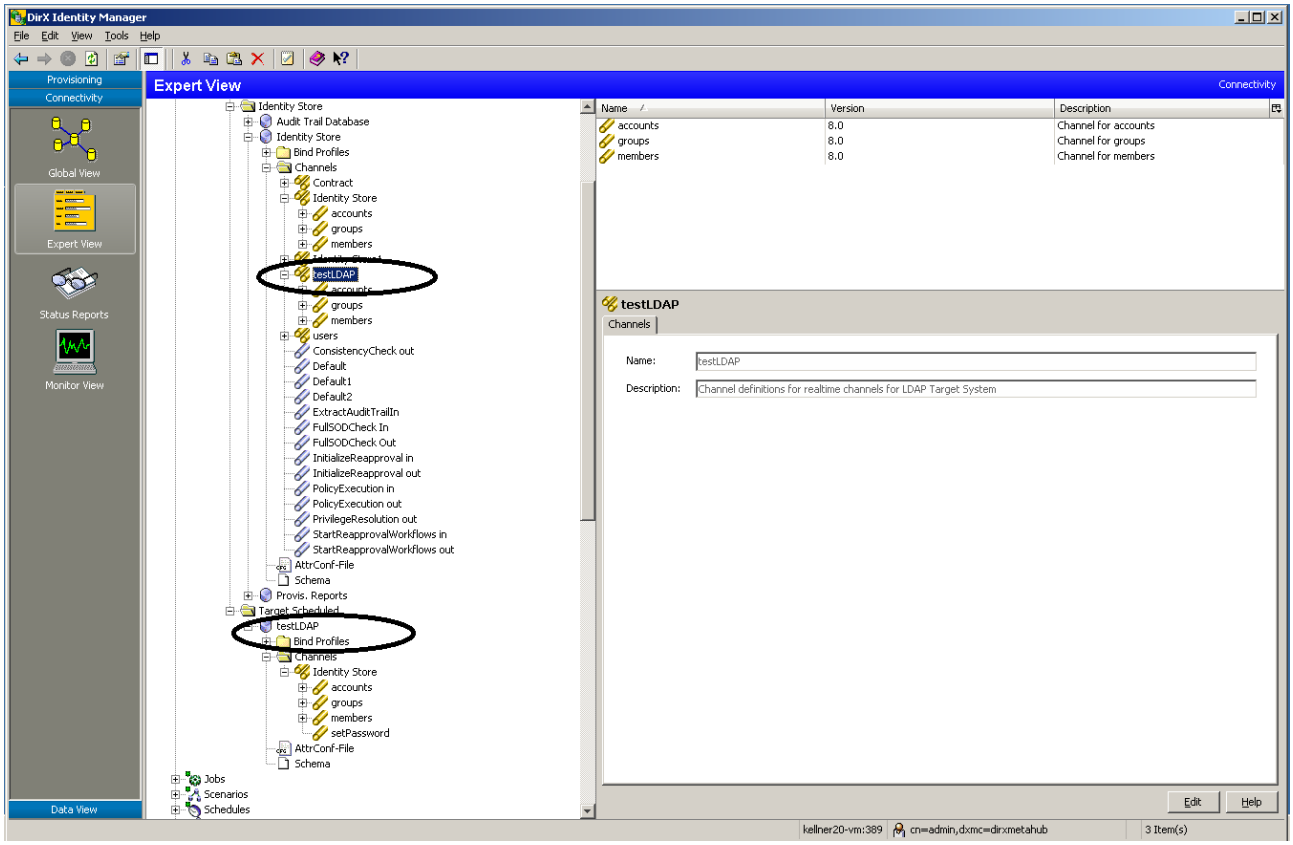

Figure 3. testLDAP Target System Real-time Workflow

For the approach described in this use case, we need the following items:

- An Identity Store with two channel folders: one for the source tree (**ContractStore** in this use case) and one for the target tree (**UserStore** in this use case). This configuration corresponds to the requirement that both the source and the target are the Identity.
- A synchronization workflow that is applicable for **dxm.event.ebr** instead of **dxm.request.provisionToTS**.

The following screenshot shows the **ContractStore** and **UserStore** subtrees in DirX Identity Manager's Connectivity view (Expert View).

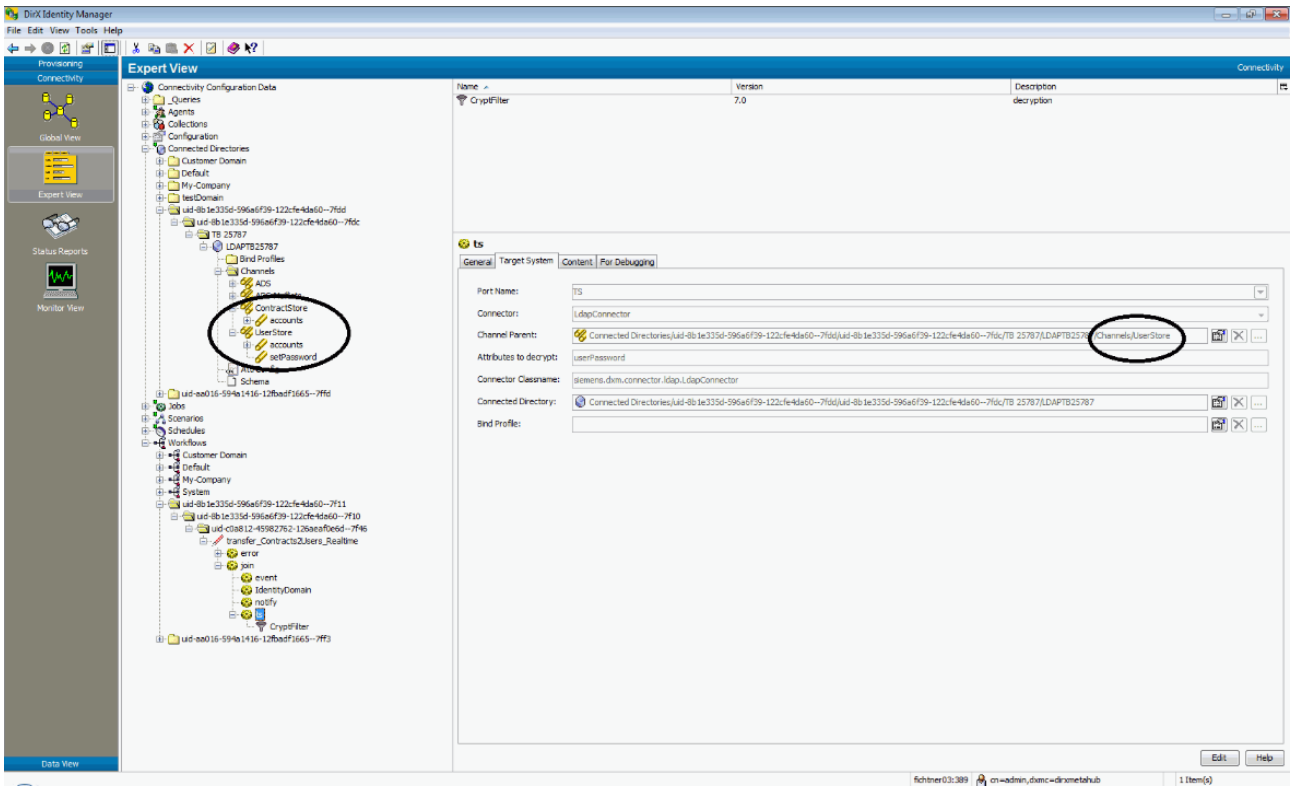

Figure 4. ContractStore and UserStore Subtrees

4. Creating the Workflow

This chapter describes how to set up the Java-based real-time synchronization workflow that illustrates the use case.

4.1. Setup Prerequisites

To implement this use case, you need to have configured at least one conventional Java-based real-time workflow for an LDAP-based target system in your DirX Identity Connectivity domain. This workflow has the correct connected directory type and belongs to the correct domain, and can act as the base for creating the new synchronization workflow. In this sample, a workflow for the **testLDAP** target system is needed, as shown in the figure in the chapter "Differences from Conventional Workflows".

In many cases, you may already have LDAP-based real-time workflows configured in your domain. If not, just create a dummy LDAP target system with a Java-based real-time workflow. For details, see the *DirX Identity Application Development Guide*; in particular, the section on Java-based LDAP workflows in the chapter "Using the Target System (Provisioning) Workflows".

4.2. Setup Steps

Creating the workflow consists of the following steps:

- Creating the LDAP synchronization workflow
- Creating the channel folders in the Identity Store
- Creating the channels
- Updating the ports
- Defining the mapping
- Updating the workflow object
- Assigning the new workflow to the scenario (optional)

4.2.1. Create the LDAP Synchronization Workflow

In the Global View, at the workflow line of our prerequisite LDAP target system real-time workflow, create a new synchronization workflow to get the correct workflow, activity, and port structure. You only need to change its name. Keep in mind that channels are reused.

4.2.2. Create the Channel Folders

The next step is to create the channel folders beneath your Identity Store.

For source tree channels:

- Select the **Channels** folder of your **Identity Store**.
- In the context menu, select **new** → **Realtime Channel Folder**.

- Enter a name (**Contract**, for example).

For target tree channels:

- Select the **Channels** folder of your **Identity Store**.
- In the context menu, select **new** → **Realtime Channel Folder**.
- Enter a name (**Users**, for example).

4.2.3. Create the Channels

Now we create the channels:

- For the source tree:
- Copy the accounts channel **from Identity Store** → **channels** → *prerequisite_LDAP_TS_channel_folder_name* (in this sample, **testLDAP**) to the source channel folder you created earlier in this procedure. You can use drag and drop: right click, drag to channel folder drop and then select **Copy here**. (See the following screenshot for details).
- For the target tree:
- Copy the accounts channel **from IdentityStore** → **channels** → *prerequisite_LDAP_TS_channel_folder_name* (in this sample, **testLDAP**) to the target channel folder you created earlier in this procedure.
- Optionally rename them; in the sample, to **contracts**.
- Delete all the links in these two channels (Corresponding Channel, Member channel, Password primary channel).
- Link these two channels together by setting the Corresponding Channel.

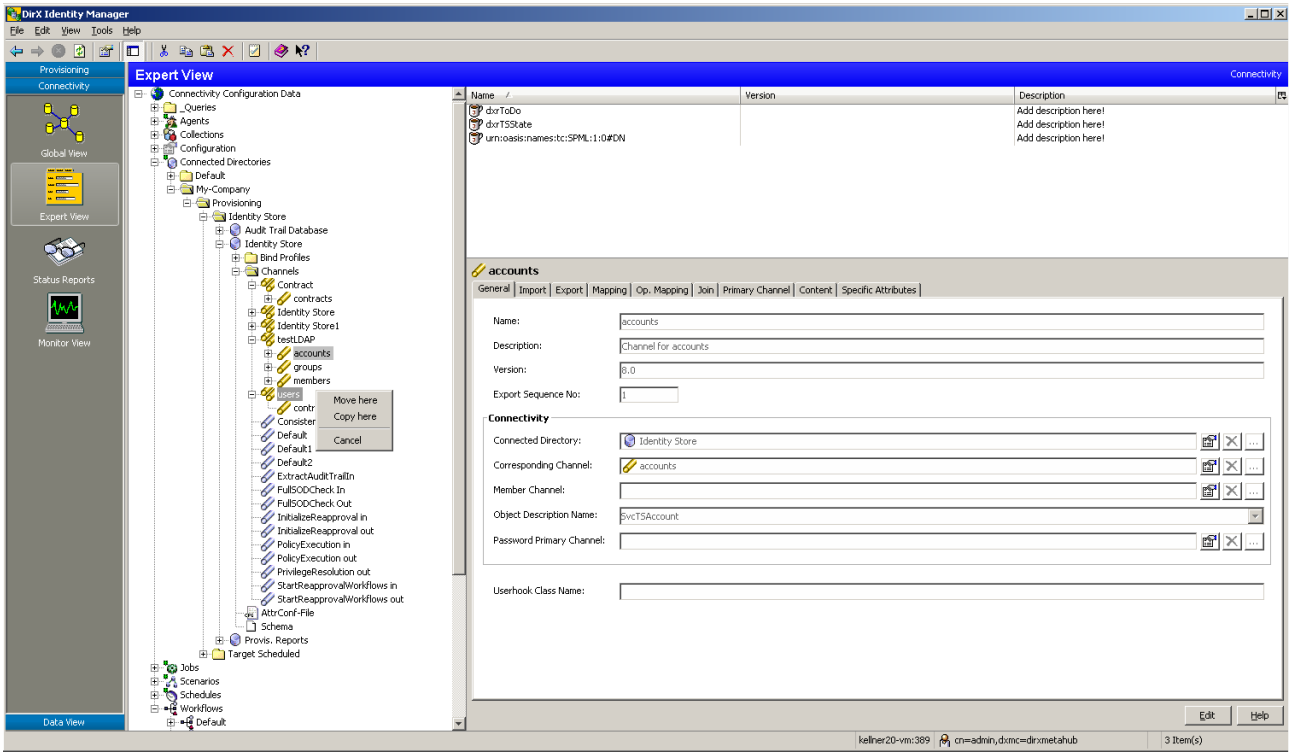

Figure 5. Copying the Accounts Channels

4.2.4. Configure the IdentityDomain Port

To configure the IdentityDomain port, change the **Channel Parent** link to the source real-time channel folder you created earlier in this procedure (named **Contract** in the previous screenshot).

4.2.5. Configure the TS Port

To configure the TS port:

- Change the **Channel Parent** link to the target real-time channel folder you created earlier in this procedure (named **Users** in the previous screenshot).
- Change the **Connected Directory** link to your Identity store.
- Change the **Bind Profile** link to a bind profile of your Identity Store (in the screenshot, the **DomainAdmin** bind profile).

4.2.6. Configure the Channels

You can adapt the channels to your requirements, defining a complex mapping here. Keep in mind that all environment variables come from the Identity Store, as it is both source and target. You can turn on design mode and then activate properties in the Target system section of the Provisioning tab of your Identity Store object, as shown in the following screenshot.

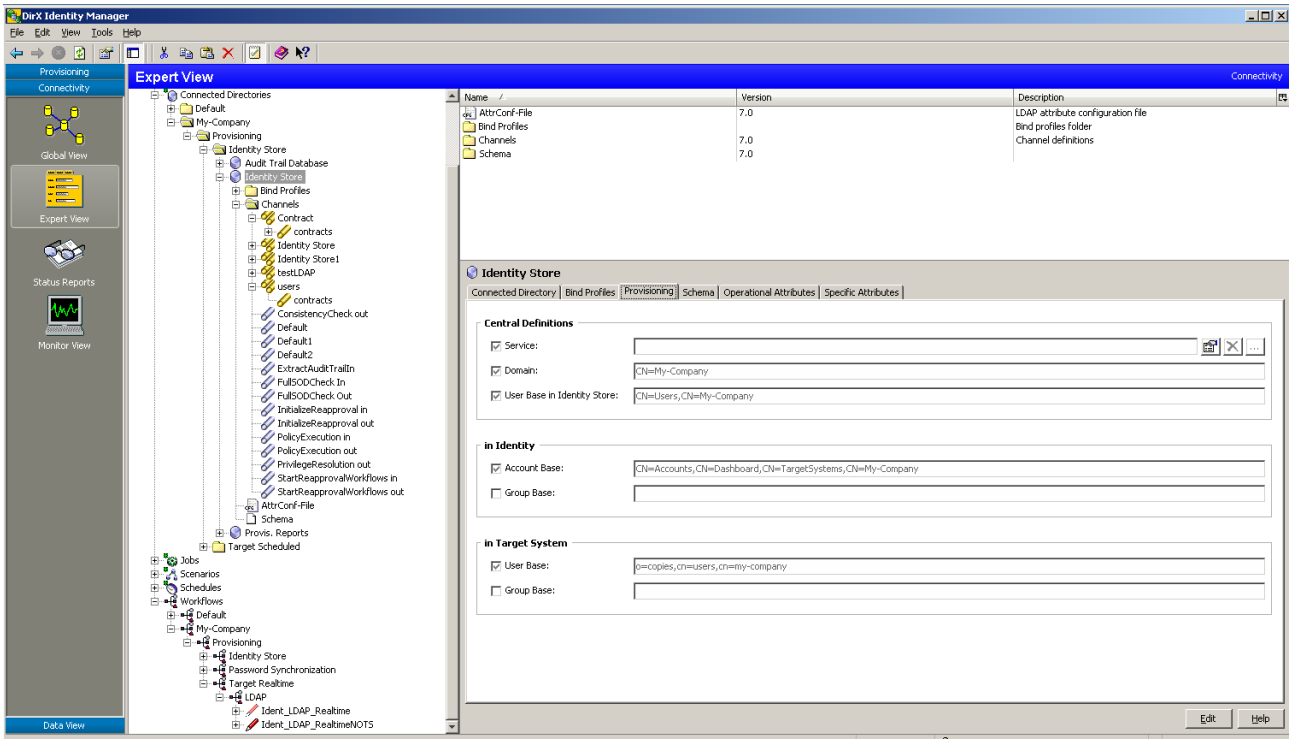

Figure 6. Identity Store Provisioning Tab

4.2.7. Configure the Controller

You can optionally change the Join Engine type in the Controller tab of the Join activity.

4.2.8. Configure the Workflow Object

You need to change the **Is applicable for** value of the new Java-based workflow object to **TOPIC_USER_CHANGE**:

- Edit the Workflow object (in this sample, **Ident_LDAP_RealTimeNOTS**).
- In the Content tab, replace the **TOPIC_PROVISION_TO_TS** value with **TOPIC_USER_CHANGE** in the following line:

<resolutionVariable name="topicSet" objectclass="dxmTopic" identAttr="dxmTopicName" value="TOPIC_PROVISION_TO_TS"/>

The following screen shot shows this line in the Content tab:

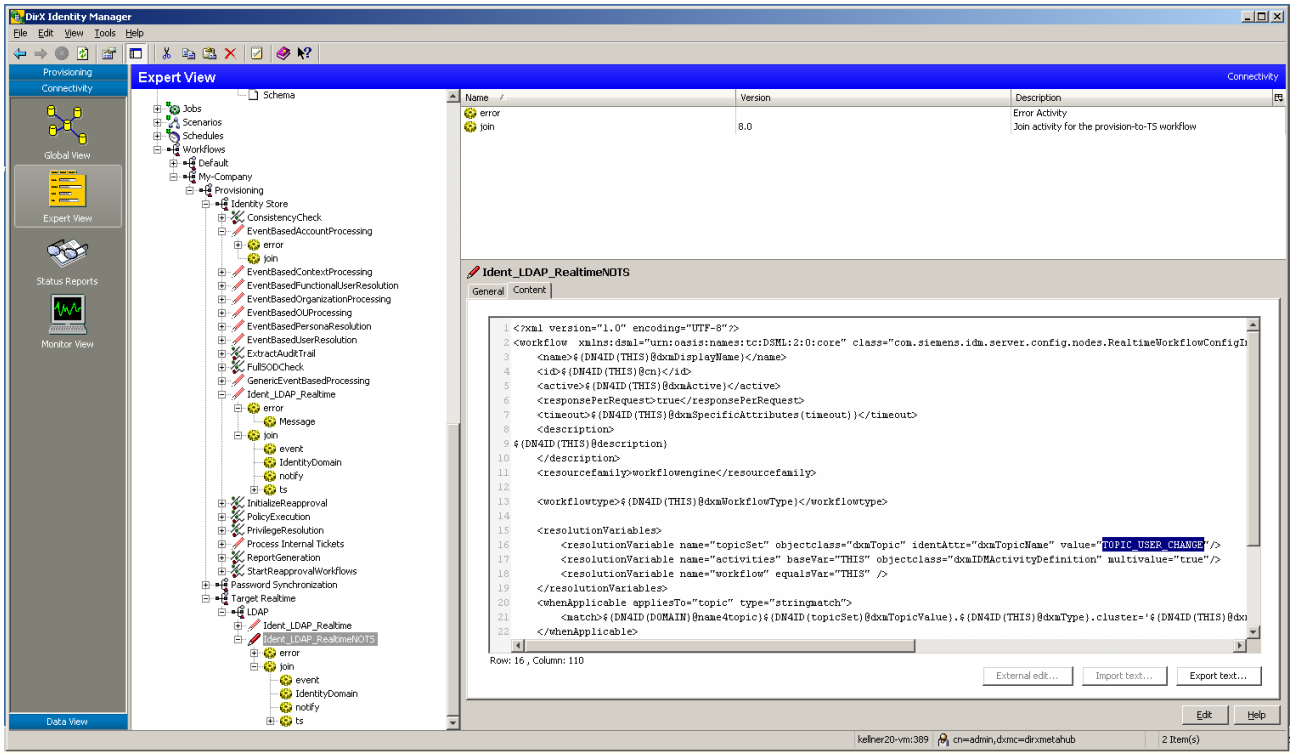

Figure 7. Editing the Workflow Content Tab

You can find valid topic prefixes in the folder **Configuration** → **Topics**. **Topic Value** is the real topic value, and **Topic Alias** is the value to be inserted in the workflow Content tab.

- In the workflow's General tab, change the **Type**, **Cluster** and **Domain** fields in the **Is applicable for** section to your requirements.

4.2.9. Assign the New Workflow to the Global View

You may want to assign the new workflow to your scenario. In the Global View:

- Select the line between two Identity Store icons.
- Right-click and then choose **Assign..** to assign the workflow.

4.2.10. How to send Provisioning Events

Assume your workflow synchronizes provisioning events with an existing target system. As the objects in the target system are not updated with the dxrServices layer no ProvisionToTS events are generated to trigger a workflow to synchronize the target system. With the following steps you can produce such events so the newly created or updated objects are synchronized immediately:

- Configure an additional event port for example eventProvToTS. Change the topicValue from TOPIC_USER_CHANG to TOPIC_PROVISION_TO_TS. You can use the edit content from the popup menu at the port if you use DXI Manager. The changed line should look like:

  <resolutionVariable identAttr="dxmTopicName" name="notifytopic" objectclass="dxmTopic" value="**TOPIC_PROVISION_TO_TS**"/>

The following screenshot shows this line in the Content tab:

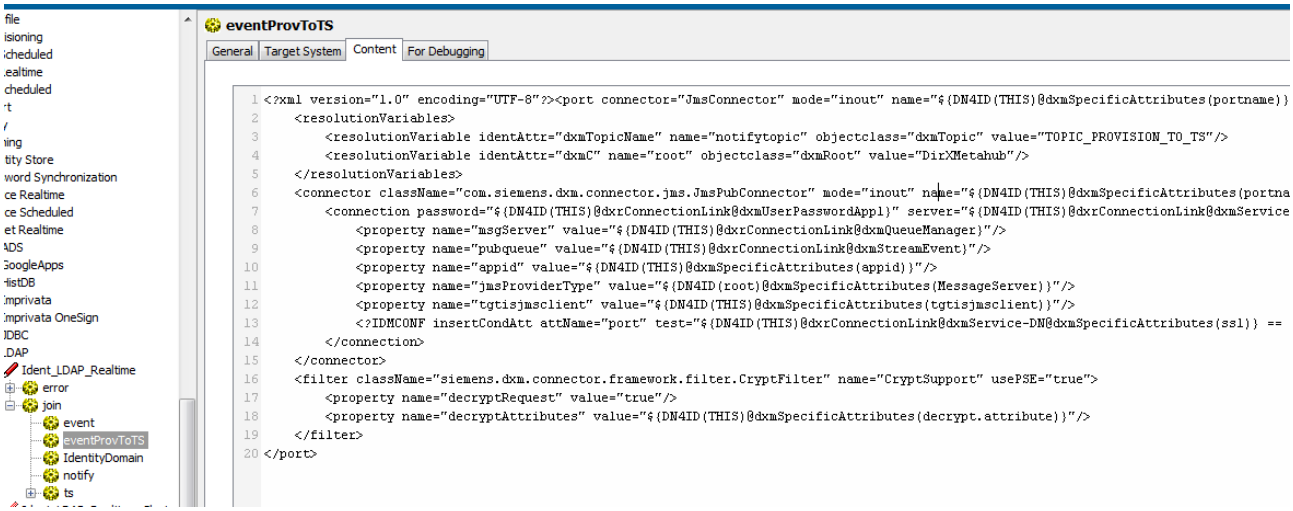

Figure 8. Editing the Workflow Content Tab

- Also set the dxmSpecificAttributes for the portname to:

  dxmSpecificAttributes: portname eventProvToTS

  The following screenshot shows this value in the All Attributes tab:

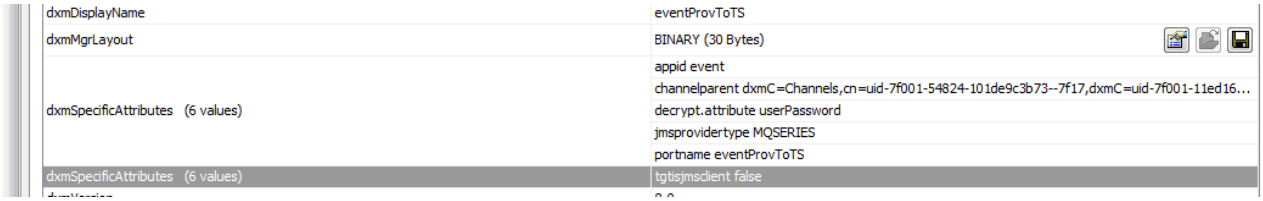

Figure 9. Editing the All Attributes Tab

In the postUpdate Userhook you can use this port to generate Provisioning events. A sample implementation is provided in MyUserhook.java.

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.