

Sun Idm Grouper adapter

= Java System Sun Identity Manager (Sun IdM) =

There is deployed Sun Java System Identity Manager (IdM) of at University of West Bohemia. Groups and memberships should be managed by Grouper.

== Data synchronization ==

There are several synchronization methods in Sun IdM. The two most important are Reconciliation and Active Sync.

Reconciliation: it's used for basic comparing of the data in identity management and on target resource. Synchronization goes through all identities, so most usual way is using it just for examination missing identities and performing basic responses (create, delete, unlink, ...). Checking attributes changes are possible, but quite exhaustive.

Active Sync: it's important for on-line updates. There is expected to update only identities changed on the source resource, so this synchronization type is usually used for transfer changed attributes too. Using this method requires support on the source resource - possibility to select all identities changed from the last processed synchronization.

== Support for groups ==

IdM has some basic support for managing groups on several types of the target resources. For example directory based (LDAP, Active Directory) and unix-like resources have manageable group members via multivalue attribute of the user identity. Group memberships on resource are managed by updating this attribute.

Groups itself and their metadata are represented as special type of resource objects on target resources in Sun IdM. Creating, updating and deleting groups itself is not supported in Sun IdM as is, but it's possible by own custom workflows. More simplistic approach could treat groups the same way, as user identities, using two separated resources - one for users and other for groups. We chose the more difficult and more interesting way - using custom workflows.

= Grouper deployment with Sun IdM =

At University of West Bohemia we keep dynamic groups up to date via our import tools, synchronizing data in Grouper directly from source agendas.

Our goal on the Sun IdM side was the direction from Grouper to Sun IdM, where Sun IdM will take care of all various target resources at university.

== Synchronize groups between Grouper and Sun IdM ==

When we will take into account removing groups too, managing users AND groups may lead to race conditions. For Sun IdM we needed to avoid to assign users to non-existing groups on target resources. Solution we chose was the adjusted order of the updates:

1. process all changed (or created) existing groups
2. process all users with changed memberships
3. process all groups for delete

As interface for Sun IdM on the Grouper side we used databases views: one for groups and one for group memberships. On Sun IdM side we used one single source resource adapter to get both - users and groups. Views are periodically polled by Active Sync, using timestamps in the views to get last changed identities.

== Sun IdM configuration files ==

- [grouper.xml](#) - main resource adapter configuration based on JDBC script adapter for IDM
- [adapter-actions.zip](#) - adapter actions. "Crazy" mix of Java and XML. The most important is grouper-as.xml (reactions to changes)
- [zcu-sample-workflow.zip](#) - our IDM workflows