

Siebel Connector

Status

Functionality	Experimental
Support status	Unknown
Support provided by	Support may be available from the original author.
Origin	Community (AMI Praha)
Target systems	Siebel

Description

Oracle Siebel Connector for midPoint is community developed connector that manages user accounts in Siebel user management.

Protocol	WS SOAP
Transport	HTTP or HTTPs
Framework	Polygon 1.4.2.14
Connector name	com.evolveum.polygon.connector.siebel.SiebelConnector
Licence	Apache License 2.0
Source	https://github.com/Evolveum/connector-siebel

Status

Some code changes may be needed before connector deployment!

This connector is using native Siebel WS. However, account data model exposed to IdM was configured on Siebel side to match particular deployment.
Connector was developed and tested with single customer with Siebel web services exposed at 3rd party middleware layer.

Siebel connector was introduced as a contribution to midPoint project by [AMI Praha](#) and is not officially supported by Evolveum.

Siebel connector contains support for ACCOUNT entity type only.

Overview

Supported Operations

The Siebel connector supports the following operations:

- Create
- Get
- Schema
- Paged Search
- Test
- Update
- Activation

Connector does not support delete operation, consider using [disable](#) instead.

Configuration

The following table lists all the configuration properties you can specify when setting up the Siebel connector:

Configuration Property	Required	Type	Example	Description
WS URL	X	String		
WS username	X	String		
WS password	X	String		
Maximum page size	X	int	100	Default page size used when explicit setting is not provided (e.g. during reconciliation task)
Connection timeout	X	int	60000	Timeout in [ms] to establish HTTP connection
Receive timeout	X	int	600000	Timeout in [ms] to retrieve data from single WS call. Should be set high enough for Search All operation to finish.
Basedir of SOAP log files		String		Enter path to the file system directory, if detailed SOAP logging is to be enabled. Leave empty to disable SOAP logging.

Getting started

Building the connector

[Download](#) and build the project with usual:

```
mvn clean install
```

Import connector JAR file to one of these locations:

- icf-connectors directory in midPoint home (recommended)
- midPoint WEB-INF/lib
- application server (e.g. Tomcat) lib directory

Creating resource

Sample resource XML can be found on [midPoint GitHub](#).

Changing the interface

As stated in the beginning, some code changes may be necessary, here are some tips:

- Connector is using wsdl2java component to generate interface classes at the build time.
- Connector comes with predefined WSDLs (src/main/resources/wsdl/) that may NOT fit your actual Siebel deployment.
- By changing the WSDL files, adapting connector and possibly fixing or disabling automated tests you may create your own Siebel connector.
- Of course, if you don't wish to do that, you can make your Siebel to adjust to data model described in connector WSDLs. Good luck 🍀

Advanced business logic

Siebel is complex system and managing some account attributes is not always easy. Here are some tips regarding sample connector:

- In Siebel's data model, some values are more important than the other. These are marked as "primary" and connector supports that behaviour via proper attribute naming.
- Related positions (PrimaryPosition, SecondaryPositions) are identified with IDs, not names.
- When creating Siebel roles definition, you may be interested in [Radio Button Roles](#) so you can e.g. replace jobtitle value when new role assignment is approved in midPoint.
- When approving role assignment, your Siebel admin may enter some resource attribute values manually (e.g. primary position) using midPoint's [forms](#). (also look for approvalSchema-level-formRef)