



**What's new in MidPoint 4.6 "Baumgarten"**

Richard Richter  
November 2022



- What's new in 4.5 "Nightingale"
- What's new in 4.6 "Baumgarten"
- Supported platforms

# | MidPoint 4.5 “Nightingale”

- Released on 11 April 2022
  - Feature release, 1 year support
- Main features
  - Identity matching – support for InCommon ID Match integration
  - Notification improvements and localizable Message templates
  - OpenID Connect support (oidc module for Flexible authentication)
- Selection of other improvements
  - Many GUI improvements, System configuration pages improvements
  - Added expression support for Axiom query language, supported and recommended option
  - Native fail-over support in LDAP connector



# Message templates (for notifications)

**midPoint**

≡ All message templates

Nodes

Reports

CONFIGURATION

Archetypes

**Message templates**

All message templates

New message template

Object collections

<input type="checkbox"/>	Name	Description
<input type="checkbox"/>	Manual correlation needed notification	Account XY from re
<input type="checkbox"/>	User change notification	Something change
<input type="checkbox"/>	User deleted notification	User XY was delete

+

⬆

⬇

🔄

⬅

▶

User change notification

Operations

← Back

Save

Delete object

Edit raw

● Basic

● Default content

● Localized content ⓘ

Details ⓘ ⬇ ⬅

Subject expression ⓘ

1

`<script xmlns:org="http://midpoint.evolveum.com/xml/ns/public/common/org-3">`

`2`

`<language>velocity</language>`

`3`

`<code>Something changed for user ${event.requesteeName}</code>`

`4`

`</script>`

`5`

Body expression ⓘ

1

`<script xmlns:org="http://midpoint.evolveum.com/xml/ns/public/common/org-3">`

`2`

`<language>velocity</language>`

`3`

`<code>The following changes were made for user ${event.requesteeName}:`

`4`

`## Executed deltas are too noisy + diving into its objectDelta is needed:`

`5`

`#foreach ($delta in $event.focusContext.executedDeltas)`

`6`

`$delta.objectDelta.debugDump()`

`7`

`#end`

`8`

`##`

`9`

`$event.focusContext.primaryDelta.debugDump()`

`10`

`</code>`

`11`

`</script>`

`12`

Content type ⓘ

Attachment ⓘ

+

-

Attachment expression ⓘ

1

Hide empty fields

**MessageTemplate**

defaultContent

localizedContent\*

**MessageTemplateContent**

subjectExpression

bodyExpression

attachment\*

attachmentExpression

contentType

**LocalizedMessageTemplateContent**

language

Velocity templates in action!

# | Notifications with message template

```
<notificationConfiguration>
  <handler>
    <description>Sends notification for filtered event</description>
    <generalNotifier>
      <expressionFilter>
        <script><code>return event?.something?.possibly?.returningTrue // not a real code!</code></script>
      </expressionFilter>
      <recipientExpression><script> <!-- recipientExpression can now return string OR focus object -->
        <code>actor</code> <!-- focus object, this will enter the expression lower as recipient -->
      </script></recipientExpression>
      <!-- message template ref as alternative to {subject,body,contenttype,attachment}Expression-s -->
      <messageTemplateRef oid="989a480f-4781-4e43-b31d-4b2e2af6d814"/>
      <transport>mymail</transport>
    </generalNotifier>
  </handler>
</notificationConfiguration>
<messageTransportConfiguration>
  <mail>
    <name>mymail</name>
    <logToFile>mail-notifications.txt</logToFile>
    <recipientAddressExpression><script>
      <code>recipient.emailAddress ?: 'missing-email@example.com'</code> <!-- fallback email if user has none -->
    </script></recipientAddressExpression>
    <!-- the rest of email server config -->
  </mail>
</messageTransportConfiguration>
```

## Example 1:

```
<q:equal>
  <q:path>name</q:path>
  <c:expression>
    <c:path>$c:account/c:attributes/ri:uid</c:path>
  </c:expression>
</q:equal>
```

## Example 2:

```
<gt>
  <path>metadata/createTimestamp</path>
  <expression>
    <script>
      <language>groovy</language>
      <code>
        ret = basic.addDuration(now, "-P30D");
        now = basic.currentDateTime();
        return ret
      </code>
    </script>
  </expression>
</gt>
```

## Example 1:

```
name = $c:account/c:attributes/ri:uid
```

## Example 2 – implicit Groovy script/code:

```
metadata/createTimestamp > ```
  now = basic.currentDateTime();
  ret = basic.addDuration(now, "-P30D");
  return ret
```
```

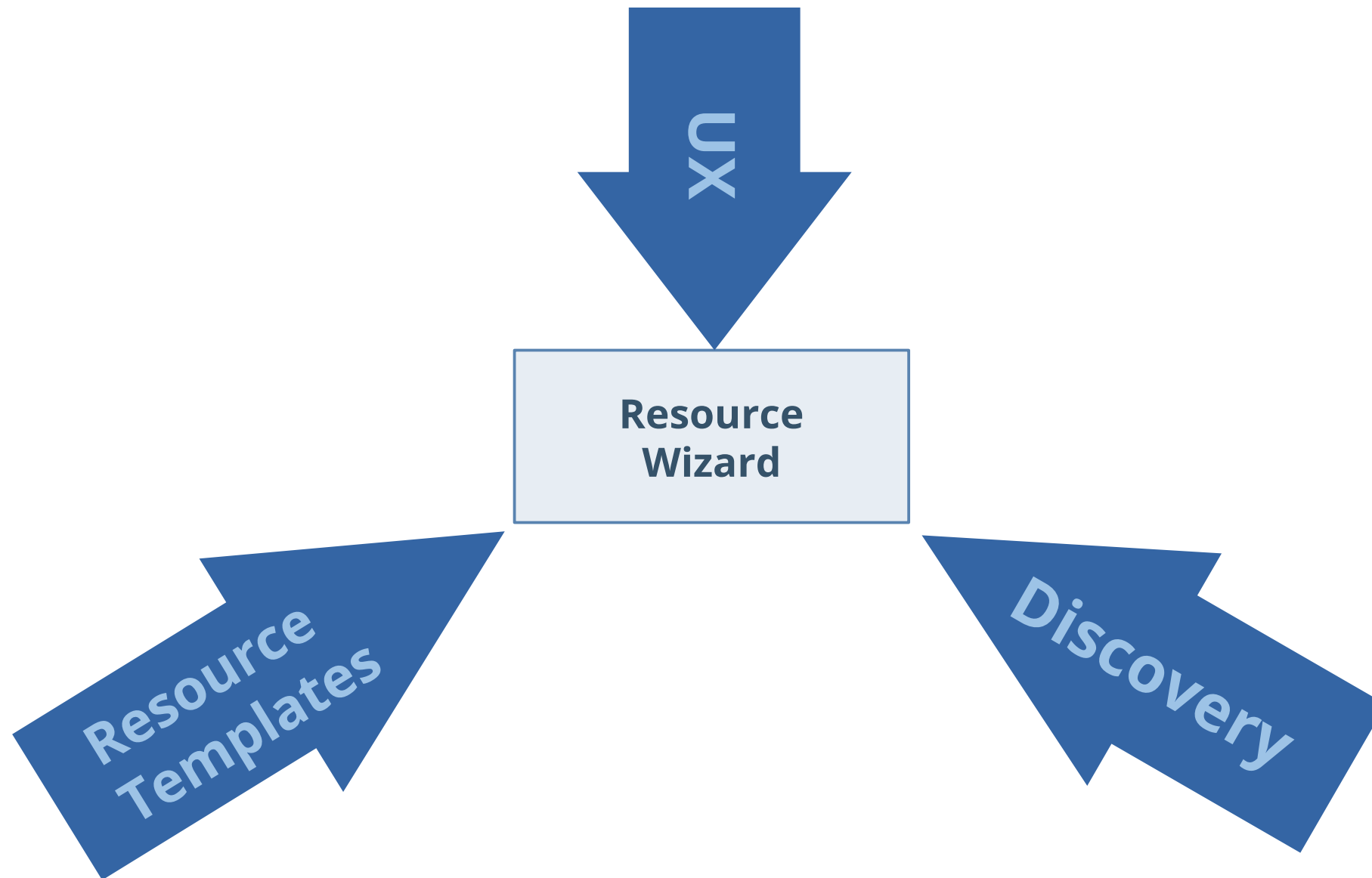
## Example 2 – universal expression:

```
metadata/createTimestamp > yaml```
  script:
    language: groovy
    code: |
      now = basic.currentDateTime();
      ret = basic.addDuration(now, "-P30D");
      return ret
```
```

# | MidPoint 4.6 “Baumgarten”

- Released on 21 October 2022
  - Feature release, 1 year support
- Main features
  - Main UX improvements: Resource wizard, Self-service
  - Smart correlation
  - Resource templates, Resource discovery
  - Many other UI/UX improvements
    - Assignment panel search can use Repository
    - Most GUI configuration changes are applied without logout now
    - Saved searches







- Basic configuration – creating new Resource
  - From scratch, selecting a Connector
  - From Resource template („abstract“ resource)
    - Resource template is a „super-resource“ and changes there affect the derived resource
- Object type configuration
  - Available as a continuation of the wizard for a new resource
  - Available from edit page via Accounts, Entitlement... and Schema handling

Docs: <https://docs.evolveum.com/midpoint/reference/admin-gui/resource-wizard/>

# Resource Wizard – New resource

**midPoint**

New resource

Resources > New resource

administrator

SELF SERVICE

Home

Profile

Credentials

Request access

Consents

ADMINISTRATION

Dashboards

Users

Org. structure

Roles

Services

Resources

All resources

New resource

Import resource definition

All connector hosts

Cases

Certification

Server tasks

Nodes

Back

Resource catalog

From scratch

Name

More...

Basic

AsyncProvisioningConnector

Built-in AsyncProvisioningConnector v1.0.0

1.0.0

AsyncUpdateConnector

Built-in AsyncUpdateConnector v1.0.0

1.0.0

ManualConnector

Built-in ManualConnector v1.0.0

1.0.0

CsvConnector

ConnId  
com.evolveum.polygon.connector.csv.CsvConnector v2.5

2.5

CsvConnector

ConnId  
com.evolveum.polygon.connector.csv.CsvConnector v2.5-SNAPSHOT

2.5-SNAPSHOT

LdapConnector

ConnId  
com.evolveum.polygon.connector.ldap.LdapConnector v3.5

3.5

AdLdapConnector

ConnId  
com.evolveum.polygon.connector.ldap.ad.AdLdapConnector v3.5

3.5

DatabaseTableConnector

ConnId  
org.identityconnectors.databasetable.DatabaseTableConnector v1.5.0.0

1.5.0.0

<<

<

1

>

>>

Powered by Evolveum® midPoint. No active subscription. Please support midPoint by purchasing a subscription.

# Resource Wizard – Basic configuration

- Designed not to overwhelm with all the details
- Connection data needed for discovery go first

1 Basic information 2 Configuration 3 Discovery 4 Schema

Basic information about the resource

Fill in basic information about your resource and for more optional settings click on "Show empty fields" button

Name ⓘ  
Csv resource

Description ⓘ

Show empty fields

← Back Next: Configuration →

1 Basic information 2 Configuration 3 Discovery 4 Schema

Establish a connection

Fill in the fields down below with correct information which midPoint needs to access your resource in order to continue in the next step

Configuration

File path ⓘ  
/example/path\_to\_source\_file.csv

Hide empty fields

← Back Next: Discovery →

1 Basic information 2 Configuration 3 Discovery 4 Schema

MidPoint discovery

Your resource has gone through the discovery process and midPoint found following parameters for you that you can configure

Configuration

Field delimiter ⓘ  
,

User password attribute name ⓘ  
password

Name attribute ⓘ  
username  
firstname  
lastname

Show empty fields

← Back Next: Schema →

1 Basic information 2 Configuration 3 Discovery 4 Schema

Schema of object types

Select all the available object classes from the list below that your future resource should work with

Schema

Selected item  
AccountObjectClass

Name	Native name	Type
AccountObjectClass	__ACCOUNT__	Structured

Rows per page 20 1 to 1 of 1 << < 1 > >>

← Back Create resource


- After connection, discovery suggests possible options/values.
- After selection of object types from schema the new Resource is saved.


# | Resource Wizard – after basic configuration


- You can continue with other sub-wizards or go to Resource edit

A new resource has been created

Your resource has been successfully created and configured, now select what you want to do next

  
Preview Resource Data

  
Configure Object Types

  
Go To Resource

[Exit wizard](#)

Csv resource

Operations: [Back](#) [Save](#) [Delete object](#) [Edit raw](#)

Resource operations: [Test connection](#) [Toggle maintenance](#) [Refresh schema](#)

- Basic
- Details
- Connector configuration
- Defined Tasks
- Accounts
- Entitlements
- Generics
- Uncategorized
- Schema handling
- Connector statistics

[Configure Synchronization](#) [Configure Mappings](#)

Intent: default (Object Class: AccountObjectClass)

### Summary

State	Count	State	Count
Deleted	0	Linked	0










☐ **Name** **Identifiers**

Resource data preview

Here you can see a list of all objects that are on the resource

Object Class: AccountObjectClass

[More...](#) [Basic](#)

<input type="checkbox"/>	Name	Identifiers	Situation	Intent	Owner	Pending operations
<input type="checkbox"/>	 user01	username: user01		unknown		 
<input type="checkbox"/>	 user02	username: user02		unknown		 
<input type="checkbox"/>	 test	username: test		unknown		 

[Add](#) [Download](#) [Refresh](#) [Play](#)

Rows per page: 20 Displaying 1 to 20, unknown number of matching results. << < 1 2 3 4 5 > >>

[Exit wizard](#)

Object type manager

Here is a table with all the objects available in the selected resource, manage existing or create a new one

[Basic](#)

Display name	Kind	Intent	Member	Description
No matching result found.				

Rows per page: 20

[Exit wizard](#) [Add new object type](#)

# | Resource Wizard – new Object Type

- Also available from Schema handling panel

The image displays three overlapping screenshots of the Resource Wizard interface, illustrating the steps for creating a new object type.

**Step 1: Basic information**

Basic information about the object type

Fill in the basic information about your object type and when you are done proceed further

● Basic information

Display name  
Default account

Description

Kind  
Account

Intent  
default

Security policy  
Choose

Default  
Undefined

Hide empty fields

Exit wizard Next: Resource data →

**Step 2: Resource data**

Specify the resource data

Fill in the necessary fields to delineate the boundary of objects that belong to that type and contains supporting instructions regarding classification of objects into types

● Resource data

Object class  
AccountObjectClass

Auxiliary object class  
Add value Clear all

Search hierarchy scope  
Undefined

Filter  
Add value Clear all

Classification condition

Hide empty fields

Back Exit wizard Next: MidPoint data →

**Step 3: MidPoint data**

Specify the midPoint data

Fill in the necessary fields to specify focus objects corresponding to given resource object type

● MidPoint data

Type  
User

Archetype  
Choose

Hide empty fields


Back Exit wizard Save settings

# | Resource Wizard – Object Type wizards


Csv resource / Object types / Default account

## Object type wizard


Choose what would you like to do with this object type




Preview Data Of Object Type




Configure Mappings




Configure Synchronization




Configure Correlation




Configure Capabilities




Configure Activation



Configure Credentials



Configure Associations



Go To Resource

[↩ Back to list of object types](#)

# | Resource Wizard – Configure Mappings



Csv resource / Object types / Default account / Mappings

## Mappings

Please create a map which defines the destination places of a resource attributes in midPoint (inbound) and the destination places of midPoint attributes back to the resource (outbound). If you need an advanced optional configuration, click on the "Edit" button.

→ Inbound mappings (to MidPoint)

↔ Outbound mappings (to Resource)

<input type="checkbox"/>	Name ⓘ	Ref * ⓘ		Target	Enabled ⓘ	
<input type="checkbox"/>	Name	username	→	name	Undefined	 

+ Add simple inbound

Rows per page 20

1 to 1 of 1

<< < 1 > >>

Exit wizard

+ Add inbound

Attribute overrides

Save mappings

### Main configuration

Ref ⓘ

username

Name ⓘ

Name

Target

name

Show empty fields

Back

Evolveum

### Main configuration

Ref ⓘ

lastName

Name ⓘ

Last name

Description

Documentation ⓘ

Trace ⓘ

Undefined

Authoritative ⓘ

Undefined

Exclusive ⓘ

Undefined

Strength ⓘ

Undefined

Channel ⓘ Add value Clear all

Except channel ⓘ Add value Clear all

Time from

Time to

Source ⓘ Add value Clear all

FocusType

Expression

Target

familyName

Condition

Enabled ⓘ

Undefined

Hide empty fields

© 2022 Ev

Back

# | Resource Wizard – available from edit page

- In Accounts, Entitlements... after choosing Intent

Csv resource

✓ Up

Operations

← Back Save Delete object Edit raw

Resource operations

? Test connection Toggle maintenance </> Refresh schema

● Basic

i Details

🔌 Connector configuration

☰ Defined Tasks

👤 Accounts

👤 Entitlements

○ Generics

👁️ Uncategorized

🔄 Schema handling

📊 Connector statistics

Configure Synchronization

Configure Mappings

Configure Correlation

Configure Capabilities

Configure Credentials

Configure Activation

Configure Associations

Intent default







(Object Class: AccountObjectClass)

Search In: Repository Resource

Summary

State	Count	State	Count	State	Count	State	Count	State	Count	State	Count	Total
Deleted	0	Linked	0	Unmatched	0	Disputed	0	Unlinked	0	Nothing	3	3

Name ⓘ More... Basic

<input type="checkbox"/>	Name	Identifiers	Situation	Intent	Owner	Pending operations
<input type="checkbox"/>	 user01	username: user01		default		 
<input type="checkbox"/>	 user02	username: user02		default		 



# | Resource Wizard – capabilities

- Available per Object Type and also in Resource/Details

### Capabilities

Capabilities are definitions of a specific things that a resource can do and as there are various resource types and configuration some resources can enable/disable an account and some cannot. Click on specific capability from down below to activate/deactivate it or trigger further configuration.

List of capabilities

Discover configuration	Activation	Activation status	Activation validity from	Activation validity to	Activation lockout	Credentials	Password	Live Sync	Async update
Create	Read	Update	Delete	Test connection	Script	Paged search	Count objects	Auxiliary object classes	Run as

Exit wizard



Save capabilities

# | Resource Wizard – Configure Correlation


Csv resource / Object types / Default account / Correlation

## Correlation rules

Correlation is a mechanism used for correlating identity data with existing focus objects in a repository. This functionality can provide an approximate matching only when correlation rules are specified.

<input type="checkbox"/> Rule name ⓘ	Description	Weight ⓘ	Tier ⓘ	Ignore if matched by ⓘ	Enabled ⓘ	
<input type="checkbox"/> Username	Correlation rule for item(s) name				Undefined ▾	 
<div><div>+ Add rule</div><div>Rows per page 20 ▾ 1 to 1 of 1 &lt;&lt; &lt; 1 &gt; &gt;&gt;</div></div>						


 Exit wizard

 Save correlation settings

Csv resource / Object types / Default account / Correlation / Username

## Configuration of correlation items

Add and specify correlation item(s) to make the correlation rule work correctly

<input type="checkbox"/> Item ⓘ	Search method ⓘ ⓘ	Match threshold ⓘ	Inclusive ⓘ	
<input type="checkbox"/> name ▾	Exact match ▾			
<div><div>+ Add correlator</div><div>Rows per page 20 ▾ 1 to 1 of 1 &lt;&lt; &lt; 1 &gt; &gt;&gt;</div></div>				

 Confirm settings

- In 4.6, the correlation mechanism is based on correlation rules.
- Correlation rule types:
  - **filter** – legacy filter-based correlation rule (since midPoint 1.7)
  - **expression** – custom expression, experimental (since 4.5)
  - **idmatch** – rule using external ID Match service (since 4.5)
  - **items** – smart item-based correlation, the suggested option now (since 4.6)
- Rule composition, Fuzzy searching (next slides)
- Experimental: Custom indexing (e.g. just prefix of the name), Multiple identity sources

- Previously, correlation filter searched focus by resource object attributes.
- Item-based correlation compares focus-like object after the mapping.
- Element **correlation** can appear in the Resource configuration under **schemaHandling /objectType** or inside an Object template:

```
<correlation>
  <correlators>
    <items>
      <item>
        <ref>employeeNumber</ref>
        <!-- optional search/matching options here, otherwise exact match is used -->
      </item>
    </items>
  </correlators>
</correlation>
```











- Item correlation can be defined directly inside attribute definition:

```
<schemaHandling>
  <objectType>
    ...
    <attribute>
      <ref>icfs:name</ref>
      <correlator/> <!-- enables correlation-time mapping evaluation -->
      <inbound>
        <target>
          <path>name</path> <!-- focus property is correlated -->
        </target>
      </inbound>
    </attribute>
    ...
  </objectType>
</schemaHandling>
```

# | Rule composition

$$total\ confidence_{cand} = \min\left(\sum_{cor} \frac{confidence_{cand, cor} \cdot weight_{cor}}{scale}, 1\right)$$

- Rules can be grouped to tiers, next tier is used only when needed.
- “Ignore if matched by” implies order of processing within the tier.

<input type="checkbox"/>	Rule name <small>i</small>	Description	Weight <small>i</small>	Tier <small>i</small>	Ignore if matched by <small>i</small>	Enabled <small>i</small>	 
<input type="checkbox"/>	name similarity	Matches based on the name similarity	0.5	1	empNo match ×	Undefined ▾	 
<input type="checkbox"/>	empNo match	Exact match of employee number	1	1		Undefined ▾	 
<input type="checkbox"/>	personal id match	Optional ID matches	1	2		Undefined ▾	 
<input type="checkbox"/>	personal id similarity	Optional ID similarity	0.5	2	personal id match ×	Undefined ▾	 

**Tier 1**

**Tier 2**

+ Add rule

Rows per page 20 ▾ 1 to 4 of 4 << < 1 > >>

# | Fuzzy match/search

- Available functions: Levenshtein distance and Trigram similarity
- Supported in-memory and with the Native repository
- Usable in correlation rules:

Configuration of correlation items

Add and specify correlation item(s) to make the correlation rule work correctly

<input type="checkbox"/> Item <small>i</small>	Search method <small>i</small> <small>💡</small>	Match threshold <small>i</small>	Inclusive <small>i</small>	
<input type="checkbox"/> name <small>▼</small>	Levenshtein distance <small>▼</small>	3	True <small>▼</small>	
<div><input type="button" value="+ Add correlator"/></div>				

Exact match

Levenshtein distance

Trigram similarity

Rows per page 20 ▼ 1 to 1 of 1 << < 1 > >>

✓ Confirm settings

- Can be used in other queries, example of Axiom filter:

familyName similarity ('gren', 0.5, true)

# | Changes on the Self Service Home page

- All widgets shown on Home dashboard are configurable
- **homePage** element is added to **adminGuiConfiguration** for Home dashboard page configuration
- It's possible to configure the number of the widgets on the Home page, their visibility for different roles, their look, actions etc.
  - See <https://docs.evolveum.com/midpoint/reference/admin-gui/self-service/#home-page>
- Both standard (implemented in midPoint) and custom panels can be configured to be shown on the Home dashboard page
  - For more info see <https://docs.evolveum.com/midpoint/reference/admin-gui/admin-gui-config/>  
Panel configuration section
- **userDashboardLink** is deprecated



# New look of the Home dashboard page

midPoint

Home

Home

US

Power

administrator

SELF SERVICE

Home

Profile

Credentials

Request access

ADMINISTRATION

Dashboards

Users

Org. structure

Roles

Services

Resources

Cases

Enter the name to search for

Users

Profile

View/edit your profile

Credentials

View/edit your credentials

List resources

List users

My accesses

Name	Activation
Superuser	enabled

View all

Request access

My requests

Name	Description	Object	Opened	Outcome	State
Approving and executing change of user "enduser" (started Nov 2, 2022, 11:36:34 AM)		enduser	11/2/22, 11:36:34 AM	1	open

View all

# | Changes on the Self Service Profile page

- Self Service Profile page configuration was separated from the object details page configuration.
- **selfProfilePage** element is added to **adminGuiConfiguration** for Profile page configuration
- Documentation for Profile page configuration:  
<https://docs.evolveum.com/midpoint/reference/admin-gui/self-service/#profile-page>
- Example – default midPoint End user role was updated with Profile page configuration:

```
<selfProfilePage>  
  <type>UserType</type>  
  <panel>  
    <identifier>history</identifier>  
    <visibility>hidden</visibility>  
  </panel>  
</selfProfilePage>
```

# | Changes on the Self Service Profile page

Administrator without End user role versus user with End user role:

The screenshot shows the Self Service Profile page for an administrator user. The user's name is "administrator" and their role is "midPoint Administrator (administrator)". The left sidebar contains a "SELF SERVICE" section with links to Home, Profile (highlighted), Credentials, and Request access. Below this is an "ADMINISTRATION" section with links to Dashboards, Users, Org. structure, Roles, Services, Resources, Cases, Certification, and Server tasks. The main content area has a red header with the user's name and a profile icon. Below the header is a "Operations" section with buttons for Back, Save, and Preview changes. The main content area also has a "Basic" section with a list of items: Projections (0), Assignments (1), Activation, Password, History (highlighted with a red box), Cases (0), Personas (0), Delegations (0), Delegated to me (0), and Triggers (0).

The screenshot shows the Self Service Profile page for a user with the End user role. The user's name is "jack" and their role is "Jack Sparrow (jack)". The left sidebar contains a "SELF SERVICE" section with links to Home, Profile (highlighted), Credentials, and Request access. The main content area has a red header with the user's name and a profile icon. Below the header is a "Operations" section with buttons for Back and Save. The main content area also has a "Basic" section with a list of items: Projections (0), Assignments (2), Activation, Password, Cases (0), Personas (0), Delegations (0), Delegated to me (0), and Triggers (0).

# Changes on the Self Service Credentials page

Self Service Credentials page was reworked to comply with the latest UX designs.  
The functionality remains the same.

**midPoint** Credentials

administrator

SELF SERVICE ▾

- Home
- Profile
- Credentials**
- Request access

ADMINISTRATION ▾

- Dashboards
- Users
- Org. structure
- Roles
- Services
- Resources
- Cases

**Password**

**Current password**

.....

**New password**

.....

.....

**Password requirements**

- ✓ Characters at least 5
- ✓ Unique characters at least 3

☒ Change password on individual systems

Individual systems

<input type="checkbox"/> Name	Resource	Account enabled	Password validation	Propagation result
<input checked="" type="checkbox"/> administrator	Identity Manager Repository	✓	ⓘ	

Change password


- Completely rewritten, with first two steps optional (configuration/authz)

The screenshot displays the 'midPoint' web application interface for requesting access. The top navigation bar is blue with the 'midPoint' logo, a hamburger menu, and the text 'Request access'. On the right of the bar are icons for 'Request access', a shopping cart, a US flag, a moon, and a power button. A left sidebar shows the user 'administrator' and two main sections: 'SELF SERVICE' (with links to Home, Profile, Credentials, and Request access) and 'ADMINISTRATION' (with links to Dashboards, Users, Org. structure, Roles, Services, Resources, Cases, Certification, Server tasks, and Nodes). The main content area has a progress bar with four steps: 1. Person of interest, 2. Relation, 3. Role catalog, and 4. Shopping cart. Below the progress bar, the heading 'Who are you requesting for?' is followed by the instruction 'Select for whom do you want to request access whether it's you or group of team members'. Three selection cards are shown: 'Myself' (with a person icon), 'Users With Name Containing B' (with a pencil icon), and 'Users From Collection' (with a building icon). A 'Back' button is at the bottom center. A footer message states 'Powered by Evolveum® midPoint. No active subscription. Please support midPoint by purchasing a subscription.' and a badge in the bottom right corner shows the number '0'.

# | Request Access – choosing objects

### Define group of users


Select users for whom you are going to request access


 User selection from 'Users with name containing b'


Select manually


← Back



### Choose object




Family name 



Given name 

Name 

More... 

 Basic 

<input type="checkbox"/>	^	Name	Given name	Family name	Full name	Email
<input type="checkbox"/>		barkeeper	Barkeeper	Scumm	Horridly Scarred Barkeep	
<input type="checkbox"/>		bob	Bob		Ghost Pirate Bob	
<input type="checkbox"/>		guybrush	Guybrush	Threepwood	Guybrush Threepwood	

1 to 3 of 3 << < 1 > >> 20  

Add Cancel

# | Request Access – relation

1 Person of interest


2 Relation


3 Role catalog


4 Shopping cart


What type of access are you requesting for?

Select the type of access you wish to request for (e.g. membership, ownership, etc.)

  
Default

  
Manager

  
Approver

  
Owner

← Back

Next: Role catalog →

# | Request Access – role catalog

- Menu sections: a) org structure, b) collections, c) roles of teammates

The screenshot displays the Evolveum Role catalog interface. The top navigation bar shows four steps: 1 Person of interest, 2 Relation, 3 Role catalog (active), and 4 Shopping cart. The main area is titled 'Role catalog' with a 'Requesting for 1 users' status. A sidebar on the left contains a menu with options: Executive Division, Technology Division, Human Resources, All roles, All organizations (selected), All services, roles containing mid, and Roles of teammate. The main content area shows a grid of role cards, each with a circular icon, a name, and an 'Add to cart' button. The roles shown are Executive Division, Sales Department, Technology Division, Software Department, and Java Development. A detailed view of the 'All organizations' menu section is shown on the right, listing various organizations with checkboxes and status indicators.

Role catalog | Requesting for 1 users

Display Name | Name | Parent Organization

Executive Division <

Technology Division <

Human Resources

All roles

All organizations

All services

roles containing mid

Roles of teammate

Executive Division

Sales Department

Technology Division

Software Department

Java Development

Add to cart

Details

Role catalog | Requesting for 1 users

Display Name

Executive Division <

Technology Division <

Human Resources

All roles

All organizations

All services

roles containing mid

Roles of teammate

Name

Executive Division ✓

Sales Department ✓

Technology Division ✓

Software Department ✓

Java Development ✓

IT Administration Department ✓

Human Resources ✓

Active Employees ✓



# | Request Access – checkout

1 Person of interest

2 Relation

3 Role catalog





4 Shopping cart

[← Back](#)

Shopping cart

 1 conflict found

 Items in cart

Access name	Requesting for	
Prisoner	<span>1 User</span>	 
<div><a href="#">Clear the shopping cart</a></div> <div>1 to 1 of 1 &lt;&lt; &lt; 1 &gt; &gt;&gt; 20  </div>		

 Checkout

Validity ⓘ

Choose one ▾

Comment ⓘ

 Open conflict solver

 Submit my request

# | Request Access – conflict solver

Back

## Conflict solver

1 conflict found

Unresolved 1

Solved

Skipped

Role "Prisoner" excludes role "Judge", for bob **Fatal conflict**

There is a conflict with **Prisoner** and **Judge**

### Reason

Violation of SoD policy: Role "Prisoner" excludes role "Judge", they cannot be assigned at the same time

### Solution

This conflict can be solved by removing one of role from your shopping cart

☐ I want to keep **Prisoner** (new)

☐ I want to keep **Judge** (already assigned)

Fix conflict

Back

## Conflict solver

All conflicts were resolved successfully

There are no conflicts remaining, continue back to shopping cart and finish your request

Back to shopping cart

Unresolved

Solved 1

Skipped

Role "Prisoner" excludes role "Judge", for bob **Fatal conflict**

There is a conflict with **Prisoner** and **Judge**

### Reason

Violation of SoD policy: Role "Prisoner" excludes role "Judge", they cannot be assigned at the same time

### Solution

This conflict can be solved by removing one of role from your shopping cart

☐ I want to keep **Prisoner** (new)

☒ I want to keep **Judge** (already assigned)

Fix conflict

# | Assignment panel with repository search

- Assignment panel in object detail uses in-memory search by default.
- Since 4.6 you can switch it to repository search. (Native repository only!)
  - Search by assignment's **targetRef** target conditions, e.g. target object name.
  - Search by target object full-text index. (Not assignment's! No full-text index on them.)
  - Search by combination of target reference criteria and target object criteria.
  - Search by assignment extensions.
- Object with assignments has to be saved for consistent results.
- Order by assignment extensions possible (must be single-value).
- Enabled by **dataProvider** element in **adminGuiConfiguration**.

# Assignment panel with repository search examples

The image displays three examples of the Evolveum Assignment panel, each showing a search query and its results. The first example shows a search for 'extension/reason contains "testing"' which returns two results: Approver and Superuser. The second example shows a search for 'decisions' which returns two results: Approver and Reviewer. The third example shows a search for 'targetRef/@roleMembershipRef/@name = "System role"' which returns four results: Approver, Delegator, Reviewer, and Superuser. A blue arrow points from the third search query to the first one.

**Example 1: Search for 'extension/reason contains "testing"'**

Name	Activation	Assignment Reason
Approver	enabled	Because of testing.
Superuser	enabled	Because of testing.

**Example 2: Search for 'decisions'**

Name	Activation	Assignment Reason
Approver	enabled	
Reviewer	enabled	

**Example 3: Search for 'targetRef/@roleMembershipRef/@name = "System role"'**

Name	Activation	Assignment Reason
Approver	enabled	Because of testing.
Delegator	enabled	
Reviewer	enabled	
Superuser	enabled	Because of testing.

Docs: <https://docs.evolveum.com/midpoint/guides/assignment-repository-search/>

# | New query filters – ref filter with target filter

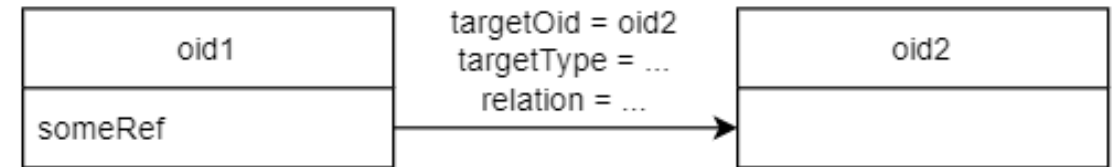
- Extended **ref** filter allows combining criteria for the reference itself (type, relation) with the criteria for the target object.
- Useful for multi-value references, allowing things not possible before.
- Example: users that are members of roles with names starting with the specified string:

```
roleMembershipRef matches (  
  targetType = RoleType  
  and  
  // @ represents ref target, target filter is inside (...)  
  @ matches (  
    name startsWith[origIgnoreCase] "super" ) )
```

# | New query filters – referencedBy

- **referencedBy** allows to filter *objects* by conditions on another object/container that is referencing the queried object.

- No relation specified means „any“.



- Example:  
Search for organizations assigned to a particular user, whatever the relation.

```
. referencedBy (  
  @type = UserType  
  and @path = parentOrgRef  
  and . inOid '2b1fd02e-db31-4896-95e9-82192df00c42'  
)
```

- **ownedBy** allows to filter *containers* by conditions on another object/container that is a parent object/container of the filtered one.
- Example: Filtering assignments for a user with specified name

```
. ownedBy (  
    @type = UserType  
    and @path = assignment  
    and name = "user-3"  
)
```

# | New query filters – ownedBy

- **ownedBy** can be nested or combined with **referencedBy**
- Example: Search for **AccessCertificationWorkItemType** containers that are part of an object with specified OID:

```
<filter>
  <ownedBy>
    <type>AccessCertificationCaseType</type> <!-- parent container -->
    <filter>
      <ownedBy>
        <type>AccessCertificationCampaignType</type> <!-- owning object -->
        <filter>
          <inOid>
            <value>a4397437-db99-413d-ae60-a437624dc8c8</value>
          </inOid>
        </filter>
      </ownedBy>
    </filter>
  </ownedBy>
</filter>
```



# | Saved filters

- User can now save often used searches, stored per user

The screenshot illustrates the process of saving a filter in the Evolveum interface. It is divided into two parts: the top part shows the 'Save filter' dialog, and the bottom part shows the main interface with a saved filter applied.

**Top Part: Save filter dialog**

- 1**: The filter query is entered in the search bar: `name startsWith[origIgnoreCase] "r"`.
- 2**: The 'Save filter' button is clicked.
- 3**: The 'Filter name' field is populated with 'Name starts with r/R'.
- 4**: The 'Save' button is clicked.

**Bottom Part: Main interface**

- The search bar shows the filter 'Name i j'.
- The 'Basic' search type is selected.
- The 'Saved filters' dropdown menu is open, showing the saved filter 'Name contains "j"'. A hand icon indicates the filter is being applied.
- The table below shows the results of the search:

	Name	Given name	Family name	Full name	Email	Acco
<input type="checkbox"/>	jack	Jack	Sparrow	Jack Sparrow	jack@caribbean.com	

At the bottom, the pagination shows 'Rows per page 20' and '1 to 1 of 1'.

- Most GUI configuration and authorization changes do not require logout
  - Changes in assignments, activation and GUI configuration for user...
  - ...or for roles, orgs, archetypes... directly or indirectly assigned to the user
- Changes in admin GUI configuration are applied without logout
- User deactivation results in logout/403 of the other session
  - validFrom/validTo applied via Validity Scanner every 15 minutes by default

Docs: <https://docs.evolveum.com/midpoint/reference/admin-gui/change-application/>

## | MidPoint 4.6 “Baumgarten”

- New Resource Wizard
- New Self-service pages, including Request Access
- Smart correlation – item-based correlation, fuzzy-match
- Saved searches
- Assignment panel repository search
- New query filters (referencedBy, ownedBy)
- GUI and assignments changes applied without logout



# | Supported platforms

- Production deployment: Linux (x86\_64)
- Java 11, recommended Java 17 (both are LTS versions)
- Stand-alone deployment
  - JAR packaging only, WAR deployment is not possible anymore
- Native repository with PostgreSQL (v13, 14, 15)
  - The old Generic repository is deprecated (PG, Oracle, SQL Server)
  - Many new features are **not** supported with the old repository!
- Any recent version of Chrome, Firefox, Safari, Edge or Opera

- Distribution package now contains **lib/midpoint.jar** (not **\*.war**)
  - WAR deployment on Tomcat is not supported since midPoint 4.5 and not possible with midPoint 4.6 (and has been deprecated since midPoint 4.1)
- No change needed if **bin/\*.sh** scripts are used
- If custom **java ... -jar /.../midpoint.war** is used, including OS service definitions, the name must be fixed to **\*.jar**
- *MidPoint Overlay* is not WAR overlay anymore, but simple Spring Boot repackaging.
  - See the example project: <https://github.com/Evolveum/midpoint-overlay-example>

- Smart correlation: <https://docs.evolveum.com/midpoint/reference/correlation/>
- Resource wizard: <https://docs.evolveum.com/midpoint/reference/admin-gui/resource-wizard/>
- Self-service: <https://docs.evolveum.com/midpoint/reference/admin-gui/self-service/>
- Request access: <https://docs.evolveum.com/midpoint/reference/admin-gui/request-access/>
- GUI Assignment Repository search:  
<https://docs.evolveum.com/midpoint/guides/assignment-repository-search/>
- Saved filters (GUI User Profile): <https://docs.evolveum.com/midpoint/guides/gui-user-profile/>
- Notifications: <https://docs.evolveum.com/midpoint/reference/misc/notifications/>
- Axiom query language: <https://docs.evolveum.com/midpoint/reference/concepts/query/axiom-query-language/>
- Query API: <https://docs.evolveum.com/midpoint/reference/concepts/query/query-api/>
- Release notes:  
<https://docs.evolveum.com/midpoint/release/4.6/>  
<https://docs.evolveum.com/midpoint/release/4.5/>

# Thank you for your time!

See other talks at <https://docs.evolveum.com>

Also **follow us** on our social media for further information!



/Evolveum



/Evolveum



/Evolveum



@Evolveum



/Evolveum

**Evolveum**

© 2022 Evolveum s.r.o. All rights reserved.