



**Accelerating Application Onboarding with MidPilot**

Pavol Mederly, November 2025  
Interim Chief Product Officer

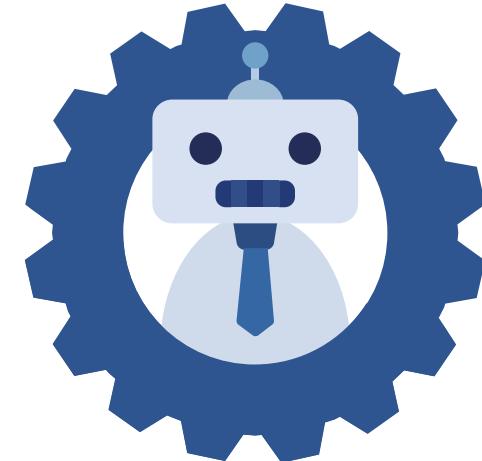
# Agenda

- MidPilot project overview
- Current state
- Your participation



# MidPilot: Connecting application faster and easier

- **Integration catalog:** connectors ready to be used
- New **low-code frameworks** for REST, SCIM and DB connectors along with new GUI and AI support
  - Improved support for **manual** (offline, ITSM) **connectors**: no more overlays
- Improved GUI for **configuring resources** with advanced algorithms and heuristics, and AI support
- **Simplifying midPoint configuration** where possible



Funded by the  
European Union  
NextGenerationEU

RECOVERY  
AND RESILIENCE  
PLAN

*This project has received funding from the European Union through the Recovery and Resilience Plan of the Slovak Republic.  
(#NextGenerationEU)*

# Integration catalog

- A service to be run by Evolveum
- Will contain **open source connectors** ready to be used
- Organized by applications, capabilities, connection technologies, versions, ownership, and so on
  - single application - multiple connectors and versions
  - detailed metadata for each connector
- Functions
  - **download** a connector
  - **upload** a connector (hosted internally or externally)
  - **request** a connector
- Accessible from midPoint or through Evolveum web page

The screenshot shows the 'Integration Catalog' interface. At the top left, there's a sidebar with 'Filter By' and 'App Status' sections, including buttons for 'All (124)', 'Available (88)', 'Requested By Community (25)', and 'Pending (12)'. Below this is a search bar and a 'Sort by' dropdown set to 'Popularity'. The main area displays a grid of 261 integration entries, each with a small icon, the connector name, a popularity badge ('Popular'), a brief description, and 'More Details' and 'Add Application/Instance' buttons. The connectors listed include SAP IAS, JIRA, Google Workspace Management for MidPoint, Mailchimp, SyncMate Long Application Name That Spans Two Lines, SecureSync App, JIRA Ops, Microsoft Teams Integration, SAP Lightweight, JIRA Conf, Google Workspace, and Mailchimp. At the bottom, there are navigation links for 'Previous', 'Next', and a feedback button.

# Declarative connector development framework

- **Declarative (low-code) approach** to writing / customizing connectors
- Currently supports the **search** operation; **create, modify, delete** operations are planned
- Technologies
  - **SCIM & REST** – <https://github.com/Evolveum/connector-scimrest/>
  - **Database** – planned
- Documentation, examples, methodology
  - <https://docs.evolveum.com/connectors/scimrest-framework/> (in progress)
- Connector development is also supported by new **GUI and AI**
  - currently only for REST



# Demo 1: Generating low-code connector for OpenProject

- Uses new declarative connector framework
- Generated using built-in wizard with AI
  - documentation gathering and processing
  - code generation
  - manual code adaptation
- Support for reading users, groups, and user-to-group associations

```
objectClass("User") {
    search {
        // Endpoint for listing users - supports filters
        endpoint("/api/v3/users") {
            // The API returns a HAL+JSON collection
            objectExtractor {
                // Grab the array that contains the users
                return response.body().get("_embedded").get("users")
            }

            // Standard pagination parameters used
            pagingSupport {
                request.queryParameter("pageSize",
                    .queryParameter("offset", pageSize)
                )

                // The API supports empty filters - no need to check
                emptyFilterSupported true
            }

            // ID filter - exact match on the user ID
            supportedFilter(attribute("id").eq(), attribute("id"))
                request.queryParameter("filters",
                    """[{"id":{"operator": "=", "value": "#id#}}]"""
                )
            }

            // ID filter - exact match on the user ID
            supportedFilter(attribute("id").eq(), attribute("id"))
                request.queryParameter("filters",
                    """[{"_id":{"operator": "=", "value": "#id#}}]"""
                )
        }
    }
}
```

## Poll 1: Connectors

## Demo 2: Configuring a resource (schema handling)

- Scenario:
  - HR: already in midPoint
  - LDAP: being connected
- For demo purposes, LDAP is also a source system here
  - so let's take it as an additional source system for now
- We'll go through the whole object type configuration
  - except for the capabilities



## Limitations when using AI

- Artificial intelligence here means **large language models**
  - running on powerful GPUs or other AI accelerators
  - locally or in the cloud
- Cost can limit local execution
- Data protection requirements (to various extent) the remote one
- **Algorithms and heuristics** are here to help



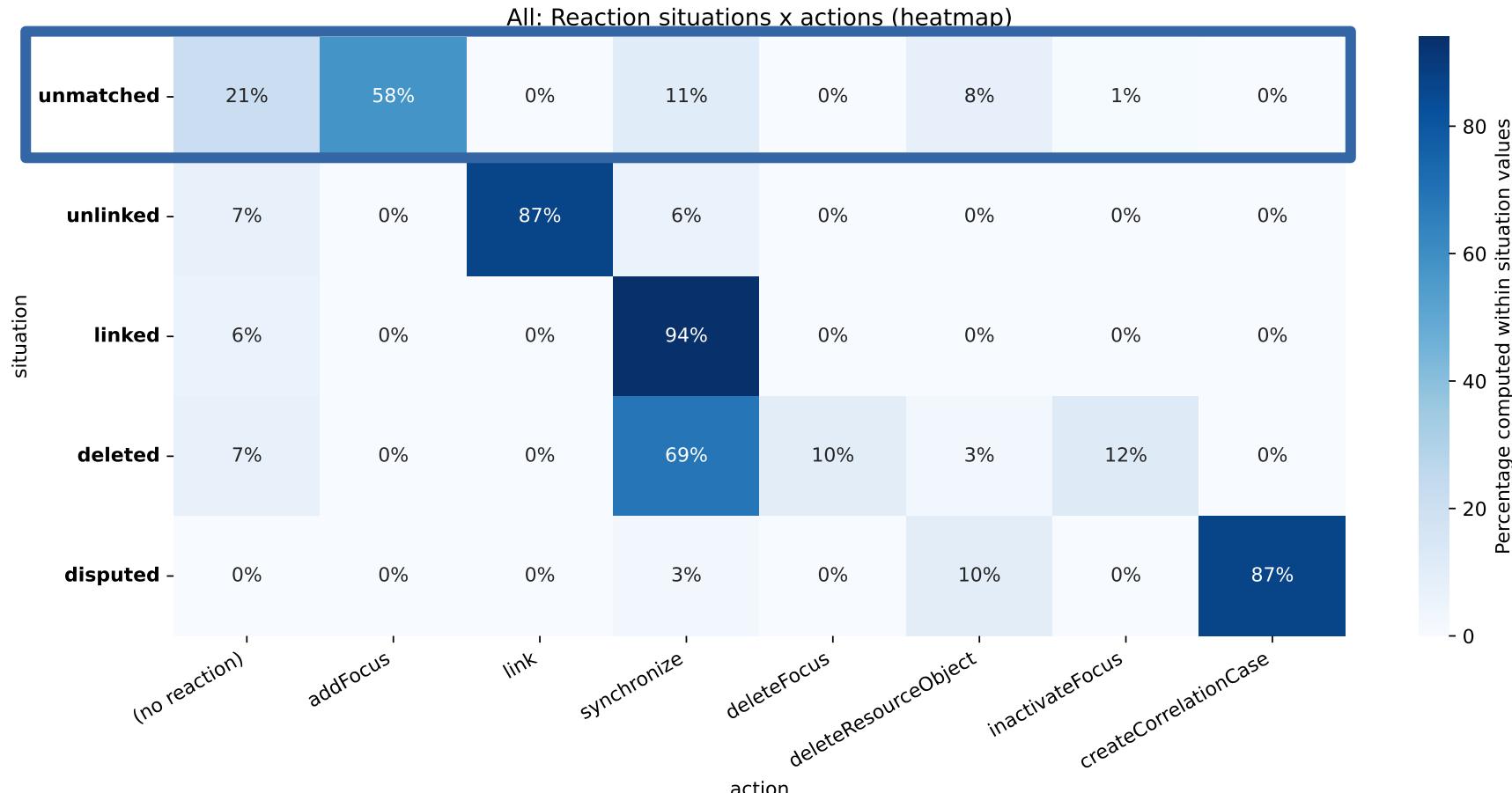
## Poll 2: AI Usage

## Algorithms and heuristics

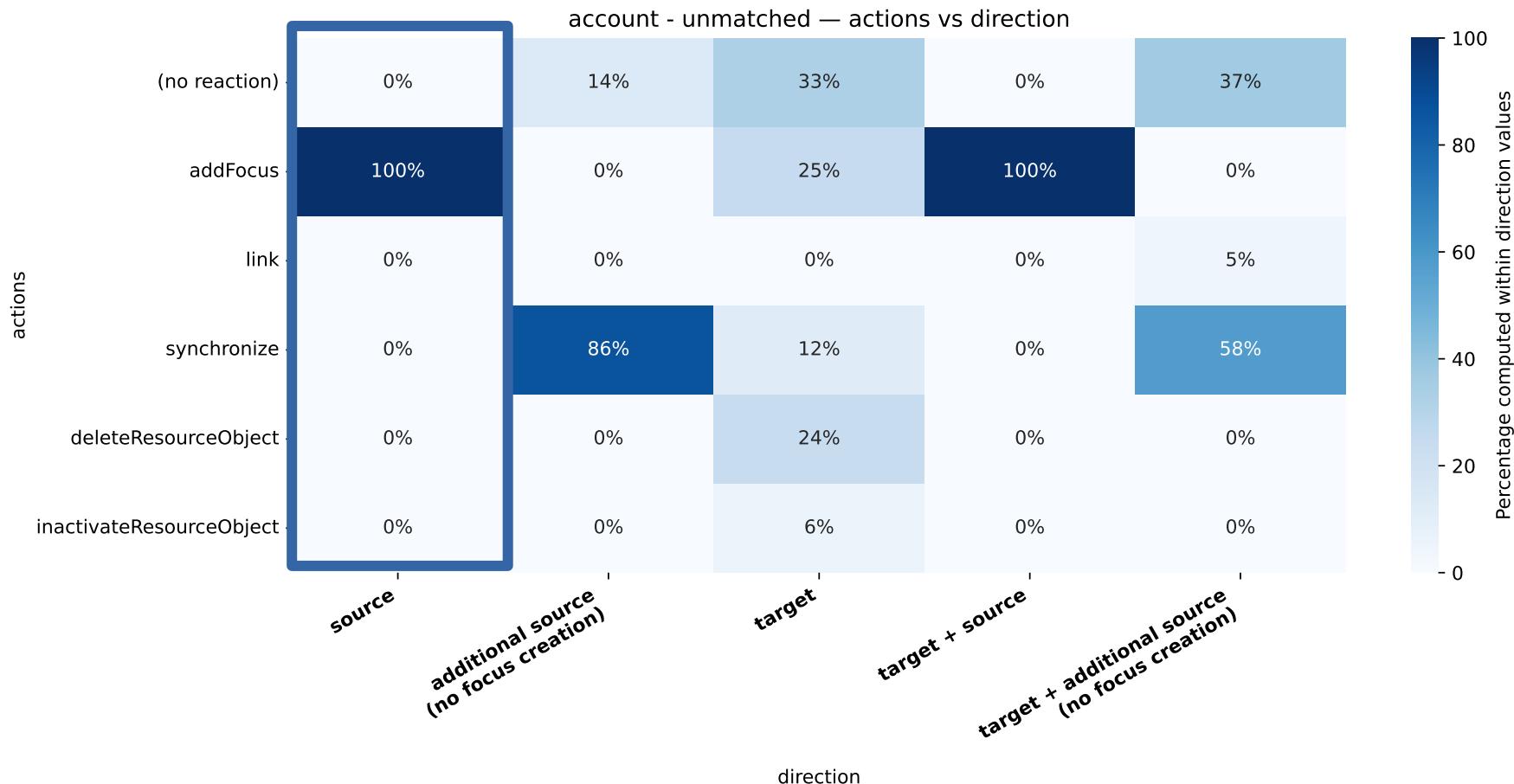
- Main inputs:
  - common sense
  - the experience of our engineering team
  - exact data
- We gathered a limited dataset we're analyzing
- ... but we need **more information.**



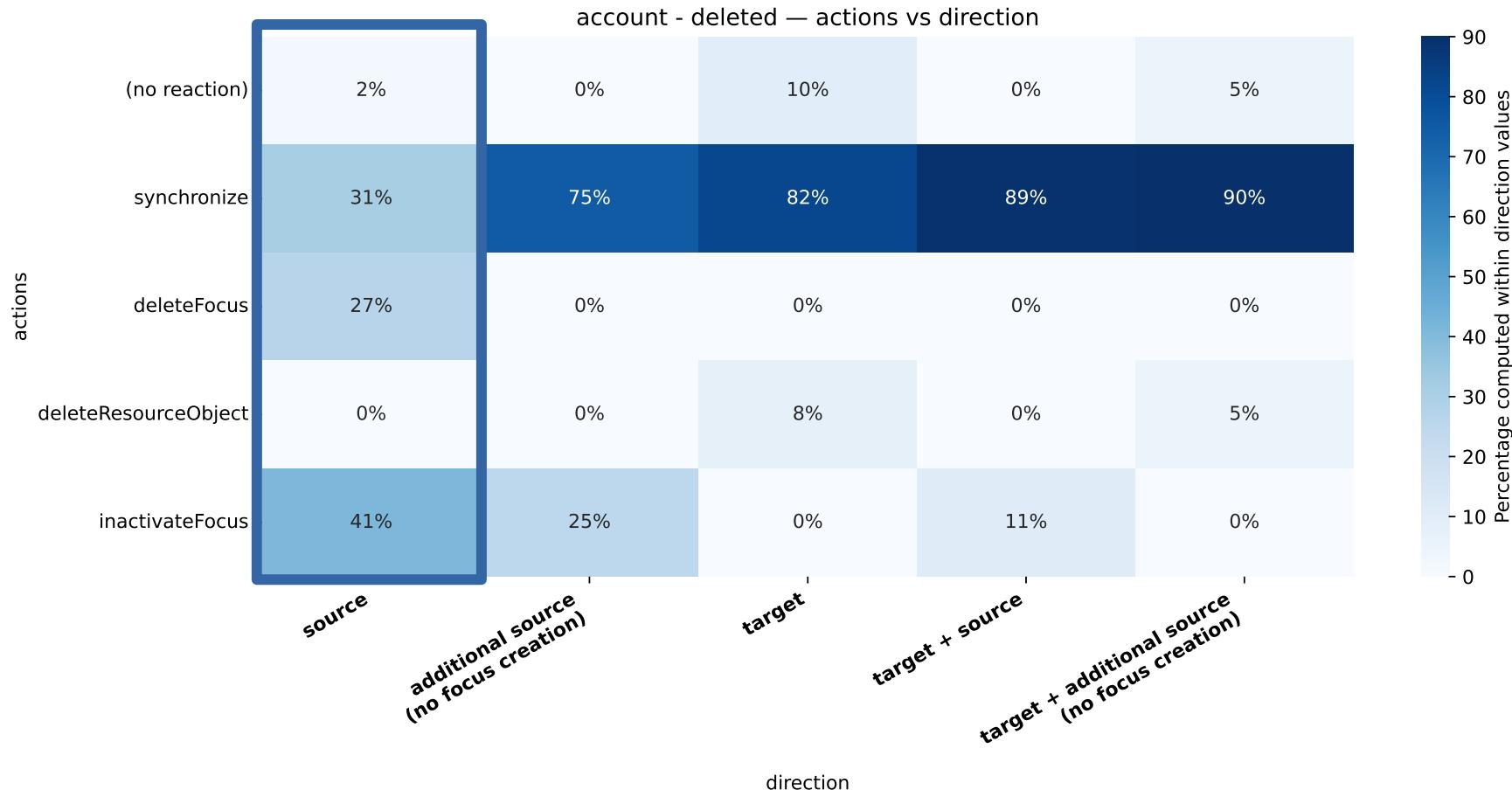
## An example: Synchronization situations x Reactions (Heatmap)



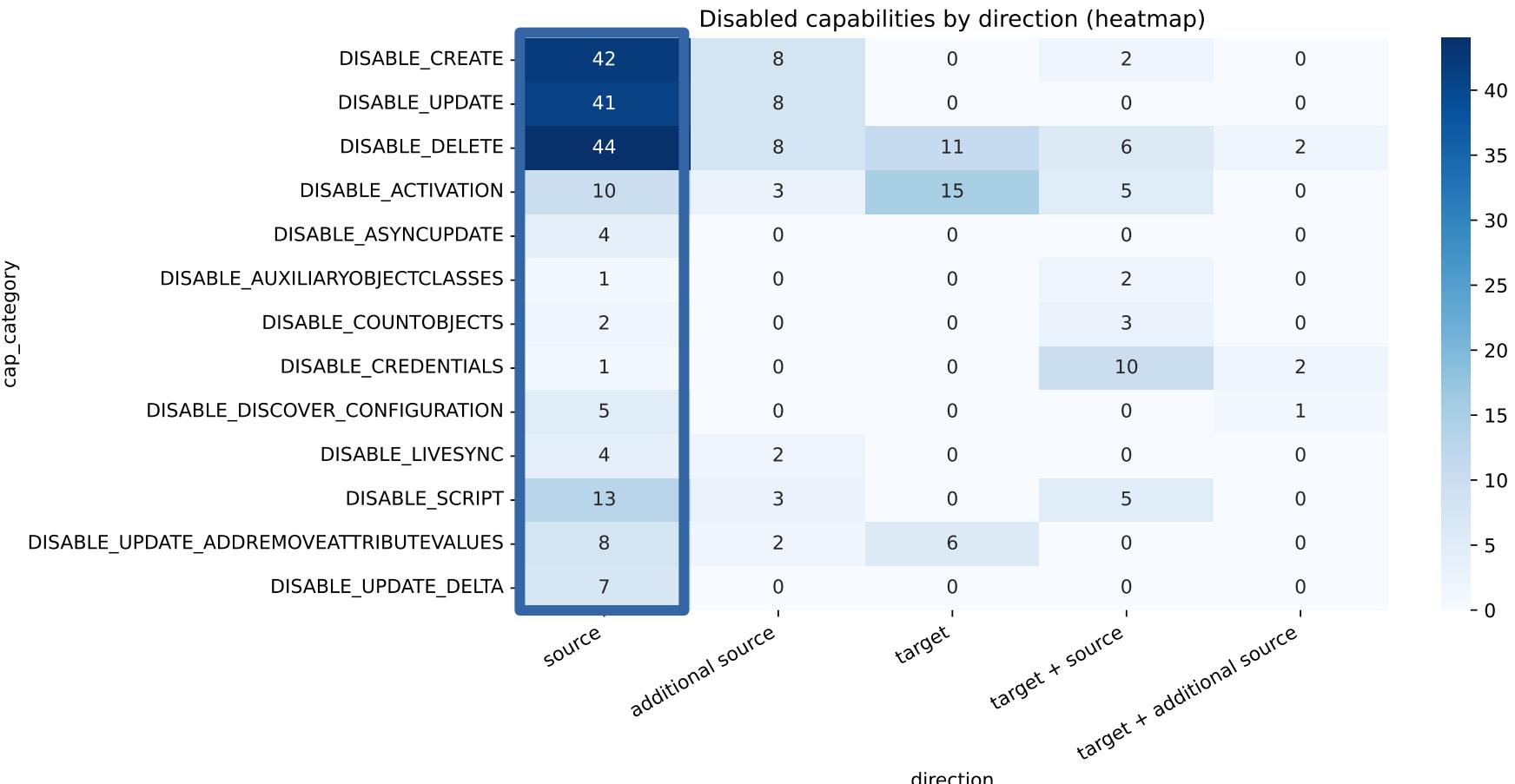
## Accounts: Unmatched situation actions x Direction (Heatmap)



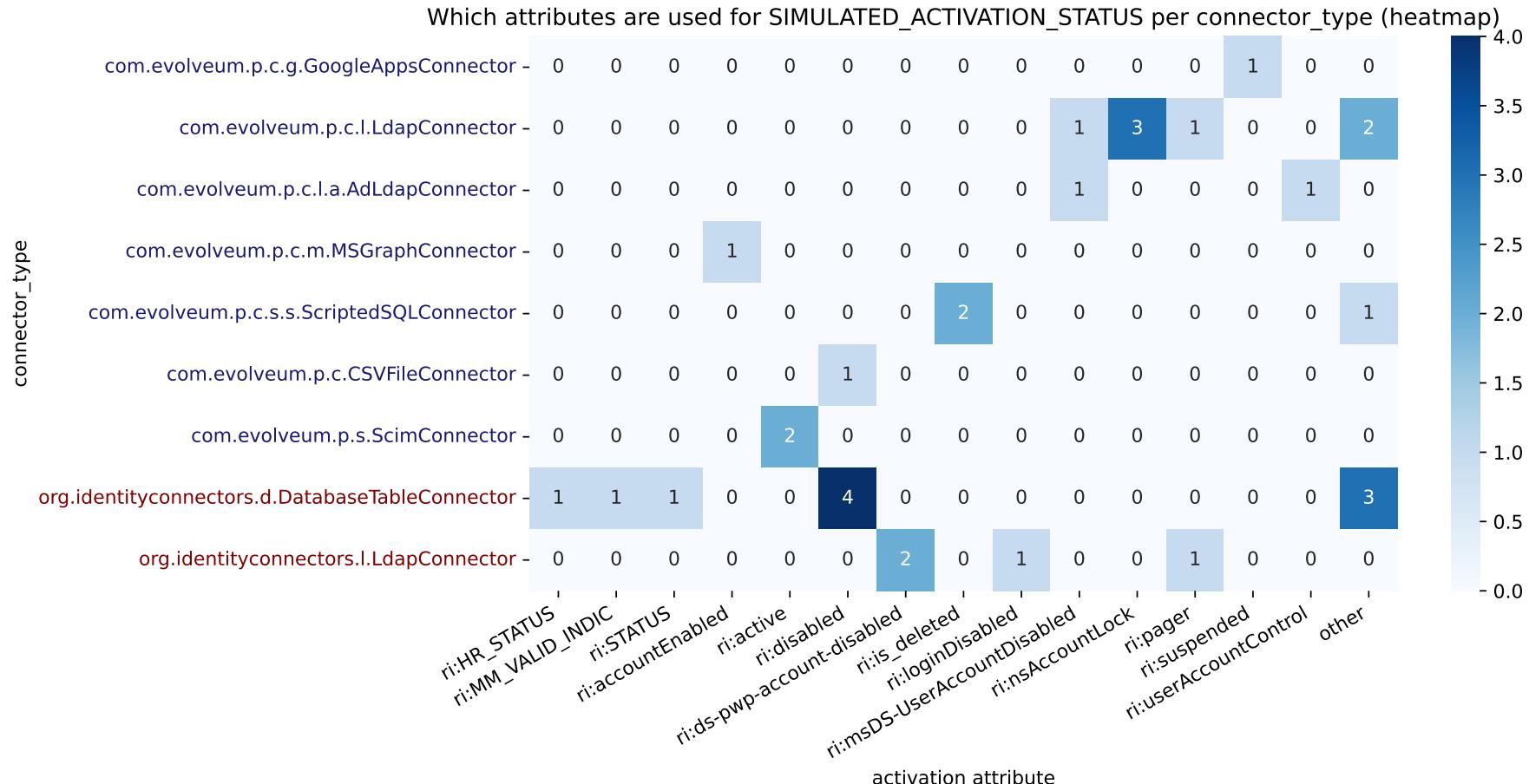
## Accounts: Deleted situation actions x Direction (Heatmap)



## Disabled capabilities X Direction (Heatmap)



# Simulated activation status attribute X Connector type (Heatmap)



# Simplifying the configuration

- Eliminate writing scripts as much as possible
  - complex attributes
  - new expression evaluators ("map")
  - new function library methods
- If scripts are unavoidable, simplify them

```
assignment = new AssignmentType()
oReference = new ObjectReferenceType()

oReference.setOid(oid)
oReference.setType(type)
oReference.setRelation(relation)
assignment.setTargetRef(oReference)

return assignment

// midpoint.createAssignmentTo(oid, type, relation)

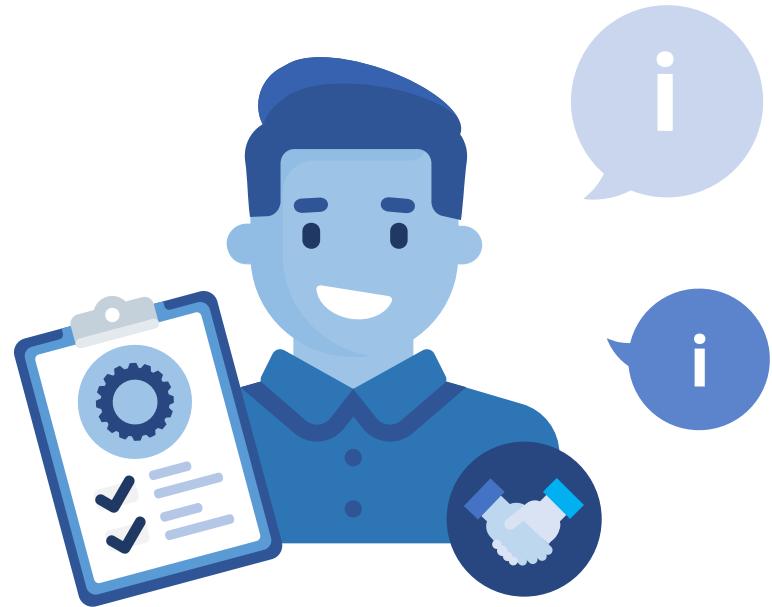
archetypes = midpoint.getArchetypes(focus)
archetypeNames = archetypeNames.size() > 0 ?
    archetypeNames.collect{ basic.stringify(it?.name) } : []
return archetypeNames.contains('Employee')

// midpoint.hasArchetypeName(focus, 'Employee')
```

## Poll 3: Pain points

## Cooperation options

- In the current state of the project we need:
  - information to analyze
  - your ideas on making application onboarding more rapid
- Later on:
  - testing development versions of the solution



# Gathering the information: configuration fragments

- What
  - ResourceType (excluding connector configuration) + names and availability of ConnectorType objects
  - ObjectTemplateType, PolicyType, ArchetypeType, FunctionLibraryType
  - parts of SystemConfigurationType
    - defaultObjectPolicyConfiguration, modelHooks, correlation
  - extension schemas
- Why
  - to find missing functionality by analyzing long and/or repeated scripts
  - to generate **synthetic** data to test our algorithms, heuristics, and LLM prompts
- How
  - use ninja tool (plus send extension schemas manually) OR send files from git + add ResourceType export (because of schemas)
  - please **anonymize** the data as needed – no company secrets, no personal information please



# Gathering the information: statistical characteristics

- What

- shadow statistics: number of objects per resource + object type + synchronization situation (linked, unlinked, ...)
- focus statistics: number of objects per type + statistics related to individual properties
  - % of objects where the property value is missing
  - % of objects where the property has multiple values
  - distribution of values (without disclosing specific values – only relative number of their occurrences)
  - ratio of unique values to the number of objects

- Why

- to generate [synthetic](#) data to test our algorithms, heuristics, and LLM prompts

- How

- use ninja tool
- no need of anonymization, as the data carry (except for property names) almost no information that can be misused
- even the object counts are blurred - we distinguish only the scale: 0, 1-99, 100-999, 1000-9999, etc.

```
<shadowStatistics>
  <resourceRef oid="4809f037-d8a7-4daa-9f96-bcead9b534ef" />
  <objectClass>ri:inetOrgPerson</objectClass>
  <kind>ACCOUNT</kind>
  <intent>external</intent>
  <synchronizationSituation>LINKED</synchronizationSituation>
  <count>9999</count>
</shadowStatistics>
<shadowStatistics>
  <resourceRef oid="9c5f9902-910f-437f-9898-ed578d1c2cf3" />
  <objectClass>ri:User</objectClass>
  <kind>ACCOUNT</kind>
  <intent>default</intent>
  <synchronizationSituation>LINKED</synchronizationSituation>
  <count>999</count>
</shadowStatistics>
<focus>
  <type>org</type>
  <count>99</count>
  <property>
    <path>identifier</path>
    <multiValuedRatio>0.0</multiValuedRatio>
    <missingRatio>1.0</missingRatio>
    <cardinality>0.0</cardinality>
  </property>
  <property>
    <path>extension/building</path>
    <multiValuedRatio>0.0</multiValuedRatio>
    <missingRatio>1.0</missingRatio>
    <cardinality>0.0</cardinality>
  </property>
  <property>
    <path>displayOrder</path>
    <multiValuedRatio>0.0</multiValuedRatio>
    <missingRatio>0.0</missingRatio>
    <distribution>
      <value>0.33333334</value>
      <value>0.33333334</value>
      <value>0.33333334</value>
    </distribution>
  </property>
</focus>
```

## Our guarantees

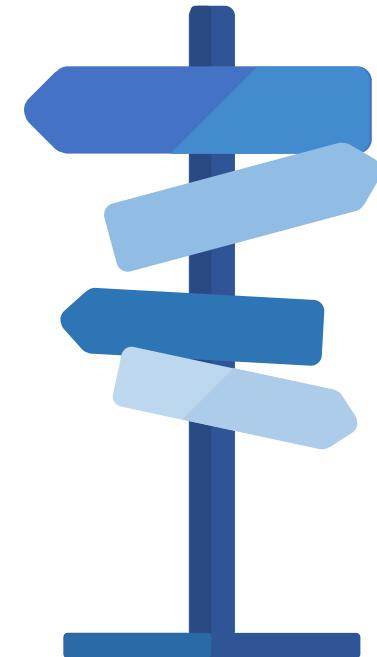
- Secure handling of the provided information
- Access to provided information strictly limited to few individuals ("need to know" basis)
- No 3<sup>rd</sup> party sharing
- No AI training based on provided information
- Limited information use
  - identification of missing midPoint features + validation of planned features
  - statistical analysis of configuration elements used in real life – to tune suggestions e.g. for sync reactions
  - internal validation of our solution during development and testing phases
  - drawing inspiration for creating synthetic configurations similar to real-life ones
  - creation of synthetic data sets with sizes and characteristics similar to the real life
- Retention period of original information is 12 months at most
  - removed automatically after that time OR earlier upon your request



## Poll 4: Cooperation options

## Summary

- Significant acceleration of the application onboarding
- Allowing you to concentrate on things bringing the real value to your customer
- We need your help to provide you with the best results



# Thank you for your attention

Do you have any **questions**? Feel free to contact us at [midpilot@evolveum.com](mailto:midpilot@evolveum.com)

Follow us on social media or **join us** at GitHub or Gitter!



/Evolveum



@Evolveum



/Evolveum



/Evolveum



/Evolveum