



## 2nd Annual MidPoint Community Meetup

# MidPilot – A Look Behind the Curtain

**Evolveum**



Funded by the  
European Union  
NextGenerationEU

**RECOVERY  
AND RESILIENCE  
PLAN**

Tony Tkáčik - Backend Technical Leader

Martin Bielik – AI/ML Developer

Michal Zelenčík – AI/ML developer

# Agenda

- MidPilot overview and architecture
- Insights from development & lessons learned

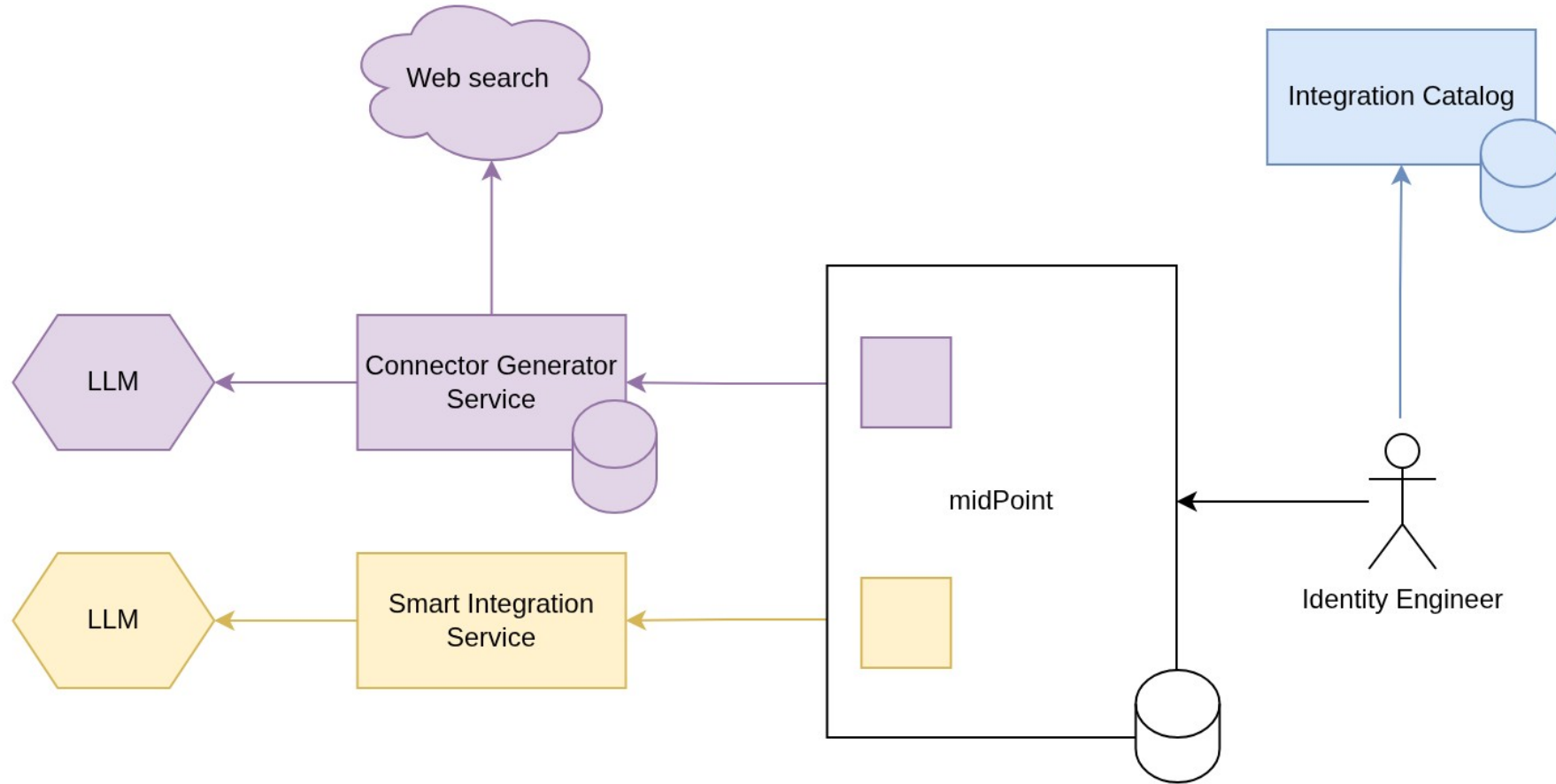


MidPilot project

# MidPilot project

- AI-powered connector development
- Assisted classification, correlation and mappings

# MidPilot project



AI-powered connector development

# AI-powered connector development



create ConnId connectors easily



REST, SCIM, DB connectors  
(including ITSM systems)



wizard guide powered  
by AI



no-code/low-code  
config-only or simple code  
fragments



every execution is  
human-approved



integrated right into  
midPoint and  
MidPoint Studio

# AI-powered connector development



**Easier code  
development,  
review, and  
maintenance**



**Safe to use**

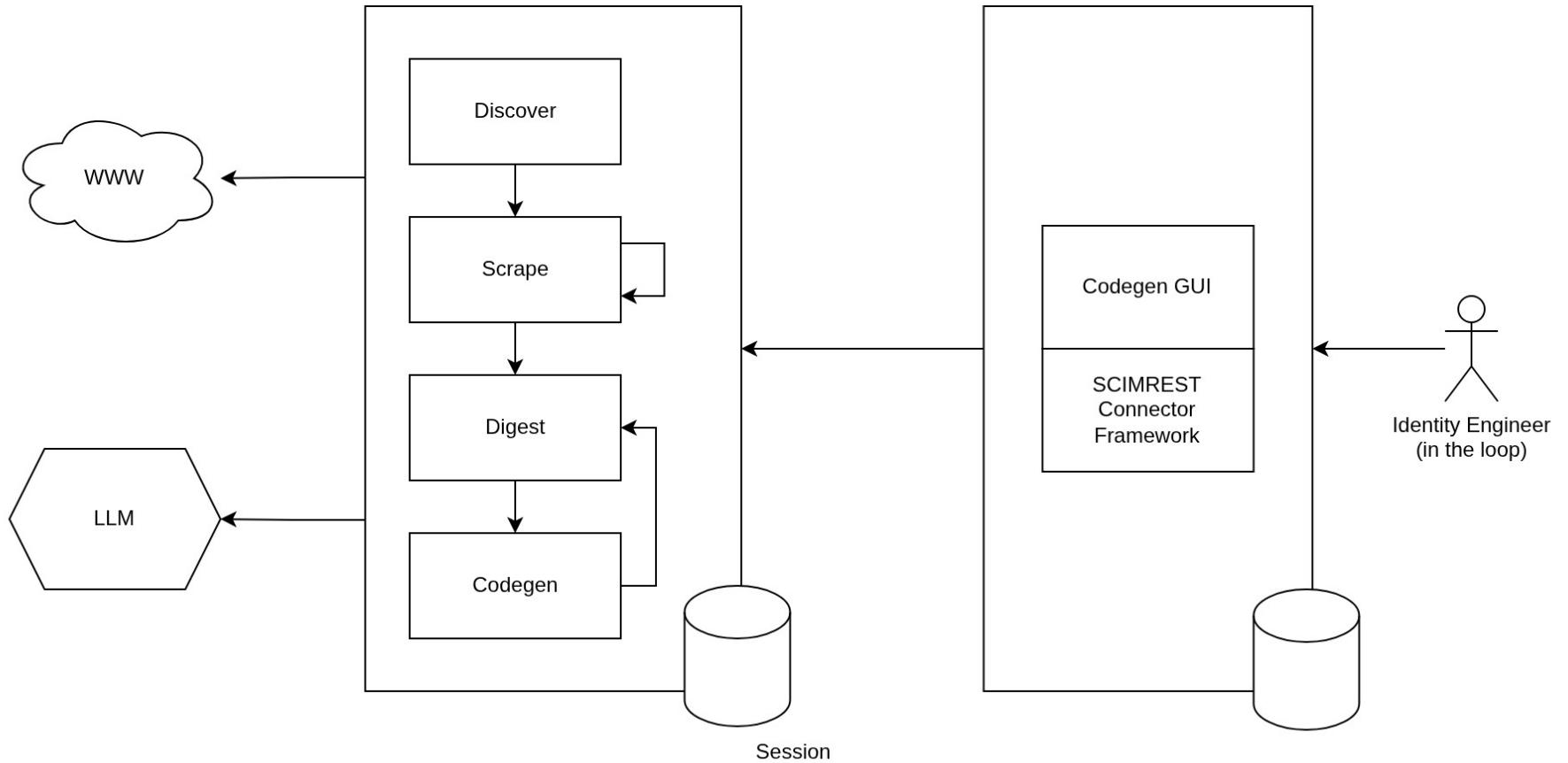


**Usable even for  
less technically  
experienced  
people**

# AI-powered connector development

Connector Generator Service

midPoint



Powered by



# MidPilot project

## Source code

- <https://github.com/Evolveum/midpoint>
- <https://github.com/Evolveum/midpilot-connector-gen>
- <https://github.com/Evolveum/connector-scimrest>

Assisted classification, correlation and mappings

# Assisted classification, correlation and mappings

1

**Classification**

how to classify  
identities

2

**Correlation**

how to match  
identities

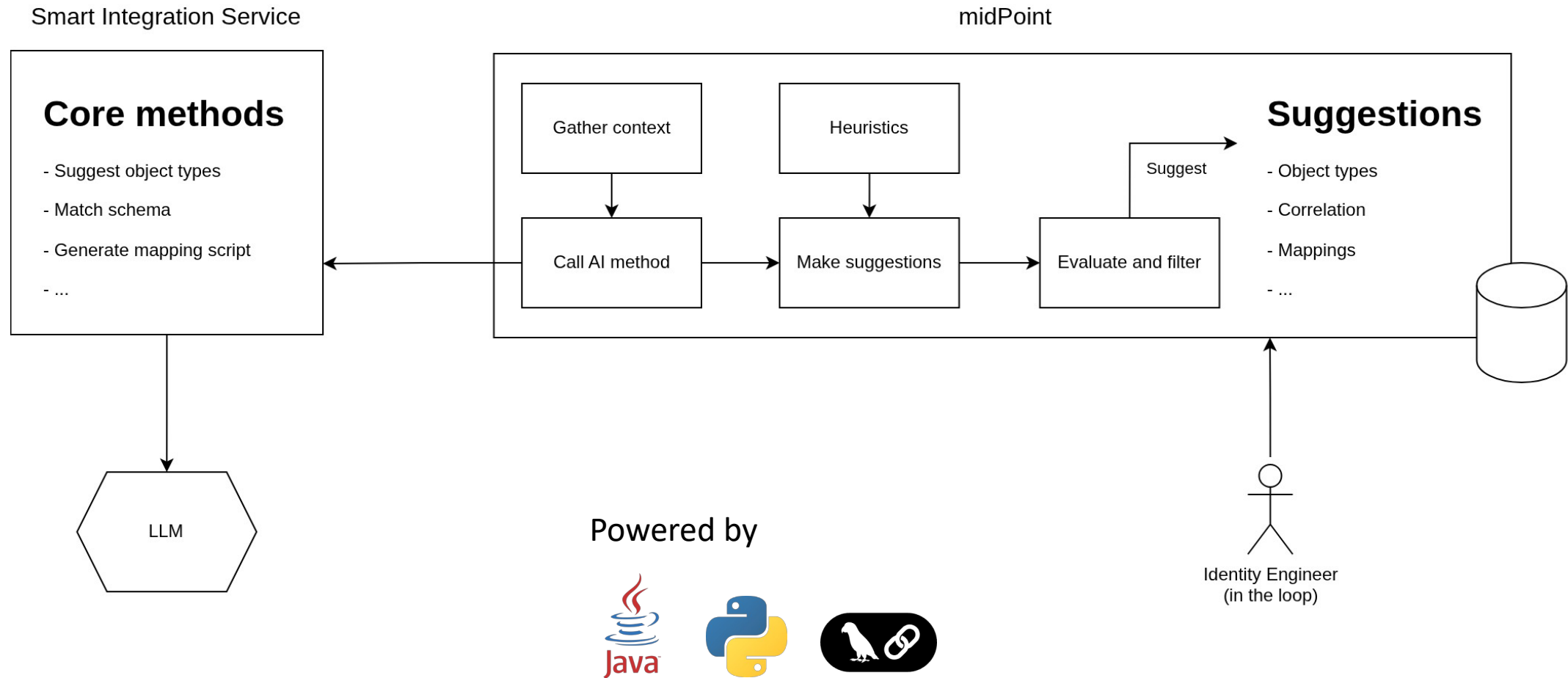
3

**Mappings**

how to translate  
identity data

**AI + Heuristics +  
UX**

# Assisted classification, correlation and mappings



# Assisted classification, correlation and mappings

Source code

- <https://github.com/Evolveum/midpoint>
- <https://github.com/Evolveum/midpilot-smart-integration>

# Large Language Models (LLMs)

# Large Language Models (LLMs)

Many possibilities these days

- GPT-5, Gemini, Claude Sonnet, Mistral, Deepseek, Qwen, ...

Different criteria

- Benchmarks
- Price
- Open vs. closed models
- On premise vs. as a service
- Security
- Legislative and geo-political considerations, ...

# Large Language Models (LLMs)

Our criteria: OSS model with good reputation and reasonable size

## **gpt-oss models family**

- Open weight models by OpenAI
  - gpt-oss-120b
  - gpt-oss-20b for smaller deployments
- Those relatively small models hit almost SOTA quality
- Evaluated and tested by Evolveum

# AI Observability

# AI Observability

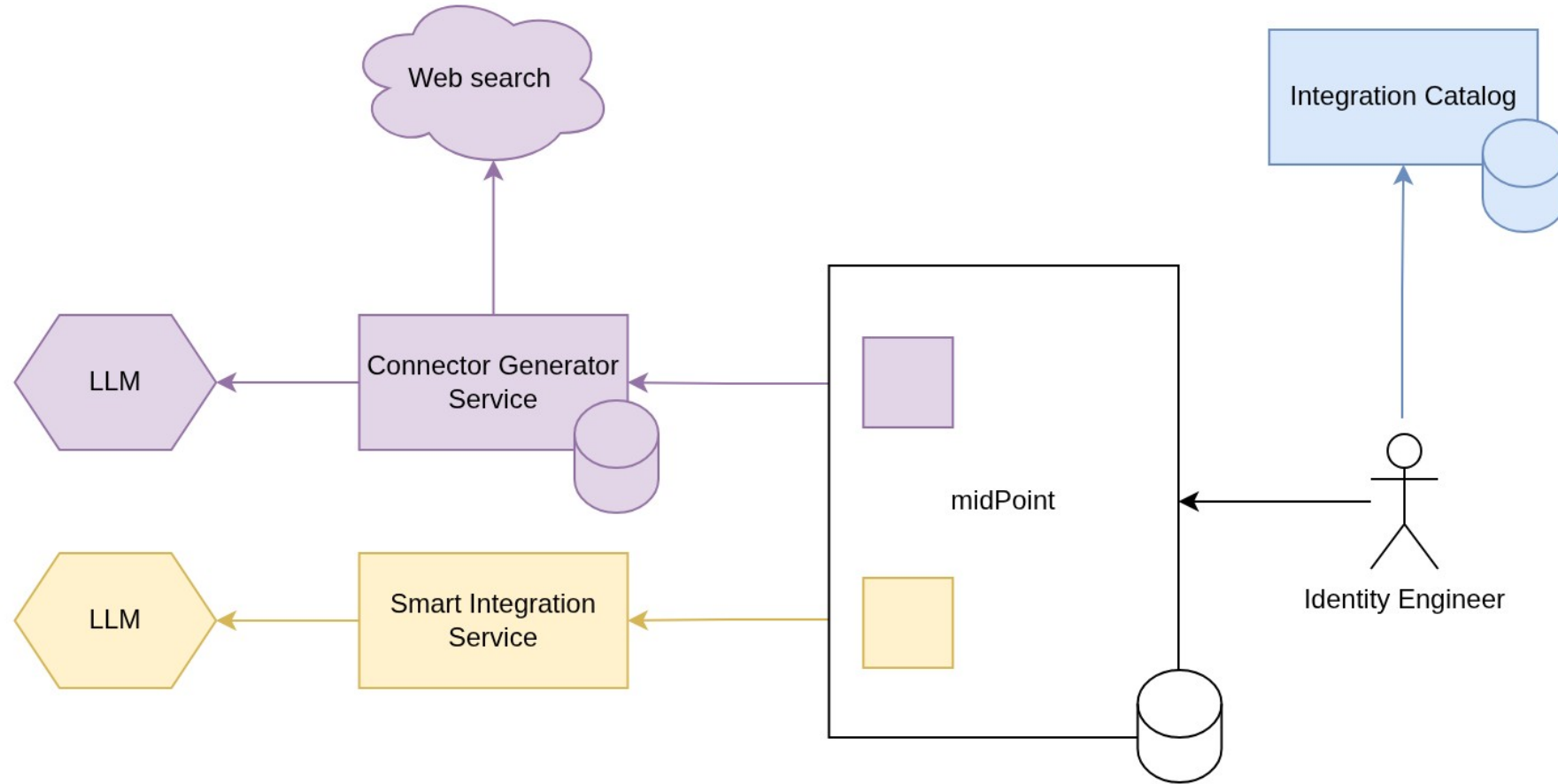
Visibility to the AI application is important

- During development for tuning and debugging
- Monitoring existing application

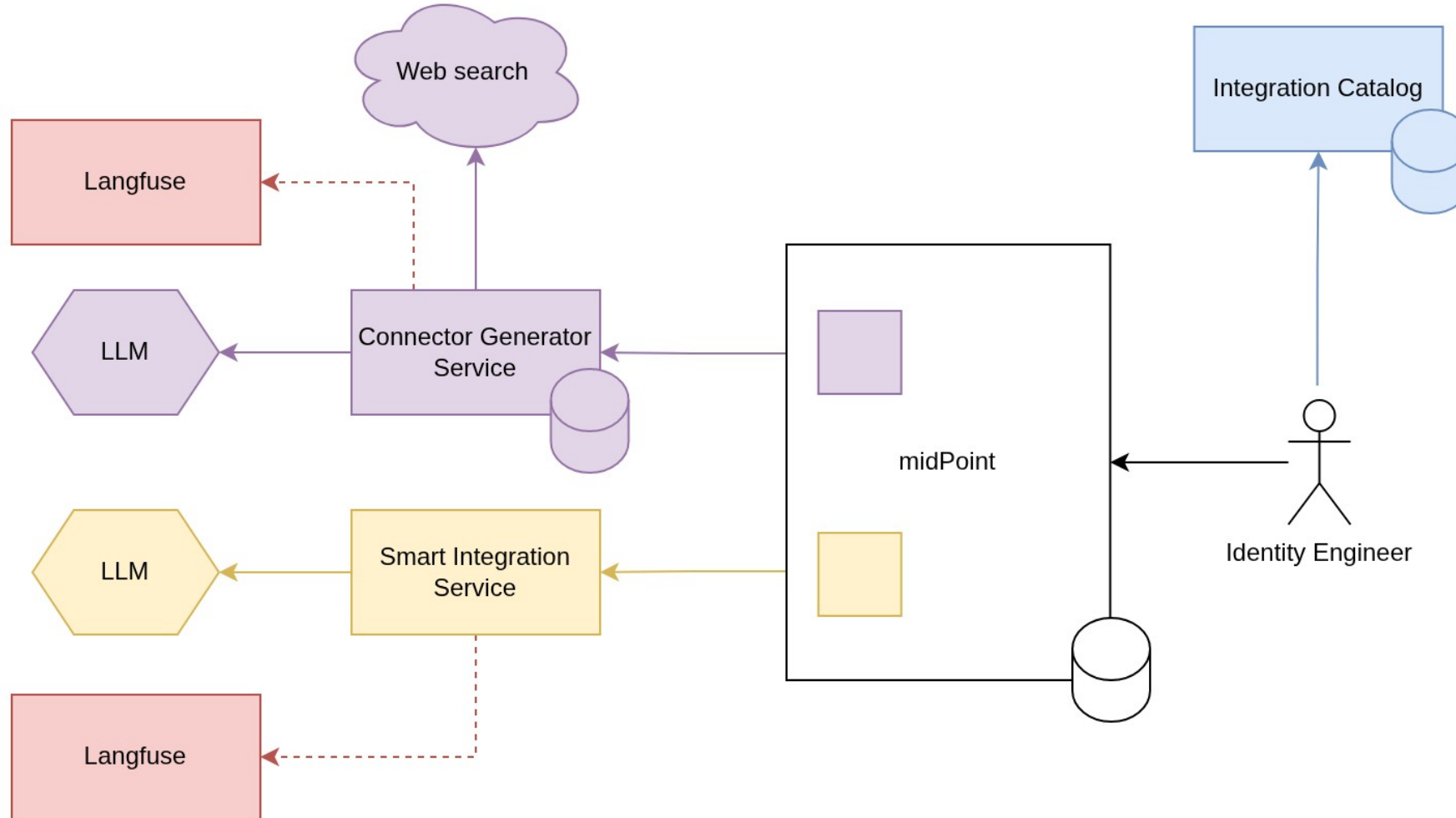
## Langfuse

- Open-source LLM engineering platform
- Features
  - **LLM Application Observability**
  - Prompt Management
  - Evaluations
  - ...

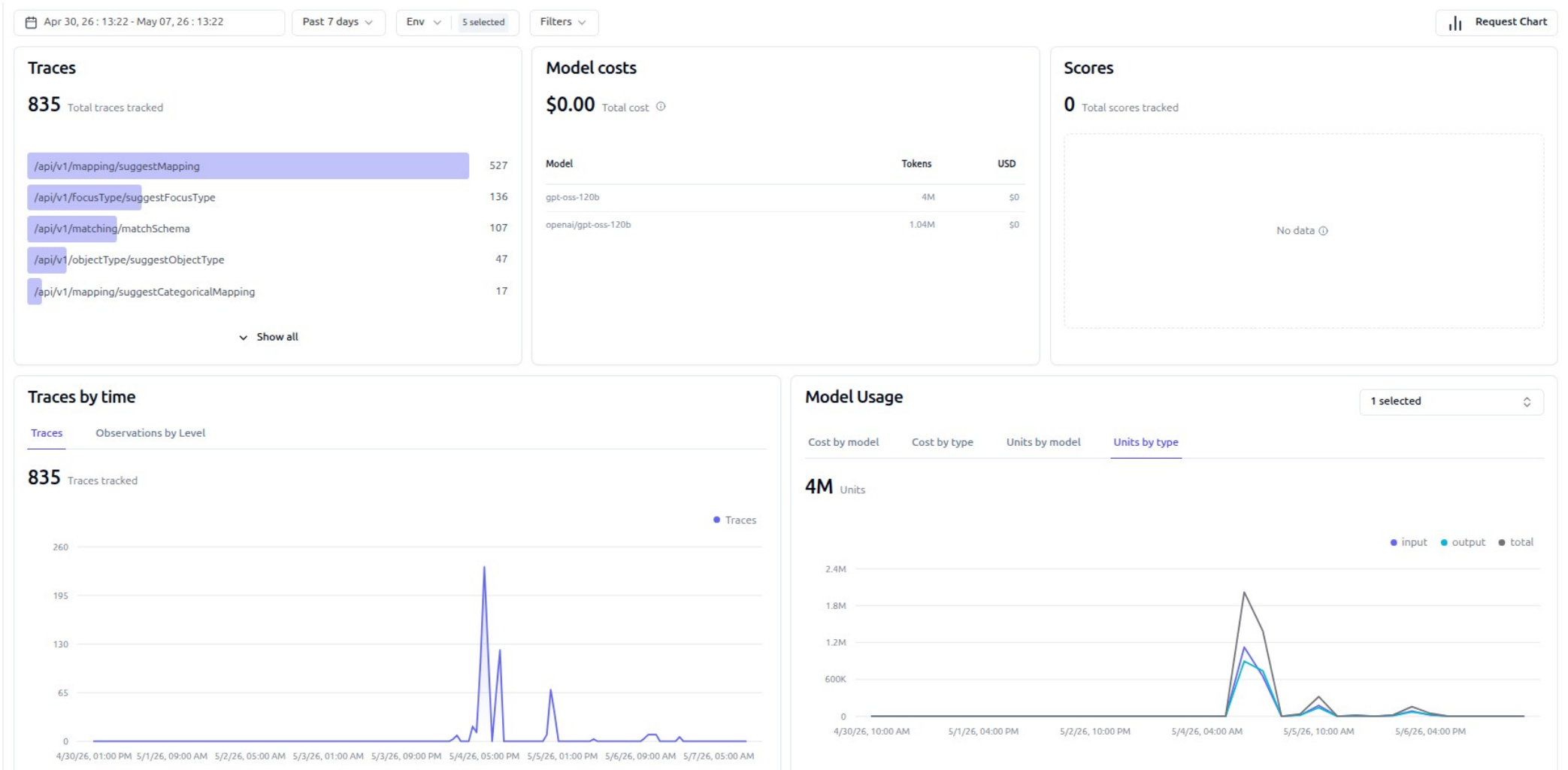
# AI Observability



# AI Observability



# AI Observability



# AI Observability

Evolveum Development traces

Traces

Search... IDs / Names Past 7 days Env

Timestamp	Name
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:45	/api/v1/mapping/suggestMapping
2026-05-05 10:08:38	/api/v1/matching/matchSchema
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:22	/api/v1/mapping/suggestMapping
2026-05-05 10:08:17	/api/v1/matching/matchSchema
2026-05-05 10:08:09	/api/v1/mapping/suggestMapping

Trace 809bad1b74f9eba3462912d5712e5653

Search (type, title, id) Timeline

/api/v1/matching/matchSchema 6.69s

- api\_request 6.69s
  - RunnableSequence 6.66s
    - RunnableParallel<completion,prompt\_val 6.66s
      - RunnableSequence 6.66s
        - PromptTemplate 0.00s
        - ChatOpenAI 6.65s 6090 → 10360 (Σ 16450)
          - PromptTemplate 0.00s
        - PromptTemplate 0.00s
      - parse\_with\_retry 0.00s

ChatOpenAI ID

2026-05-05 10:08:38.126

Latency: 6.65s Env: dev-martin 6090 prompt → 10360 completion (Σ 16450) openai/gpt-oss-120b

temperature: 1

Preview Formatted JSON

User

### IAM Schema Matching Task

Map MidPoint attributes to semantically corresponding Resource attributes. Multiple Resource matches for one MidPoint attribute are allowed.

#### Goal

- Prefer including plausible matches rather than missing true matches.
- The most important objective is to avoid missing correlator matches.
- Extra matches are acceptable; missing true correlator links is worse.
- If uncertain, include the candidate when there is plausible semantic evidence.
- Exclude only clearly unrelated matches.
- In ambiguous cases, favor semantic coverage of the concept over strict lexical matching.

#### Correlators (All Object Types)

Correlators are MidPoint attributes used to find and link an existing focus object for the same entity across systems.

Correlator sets:

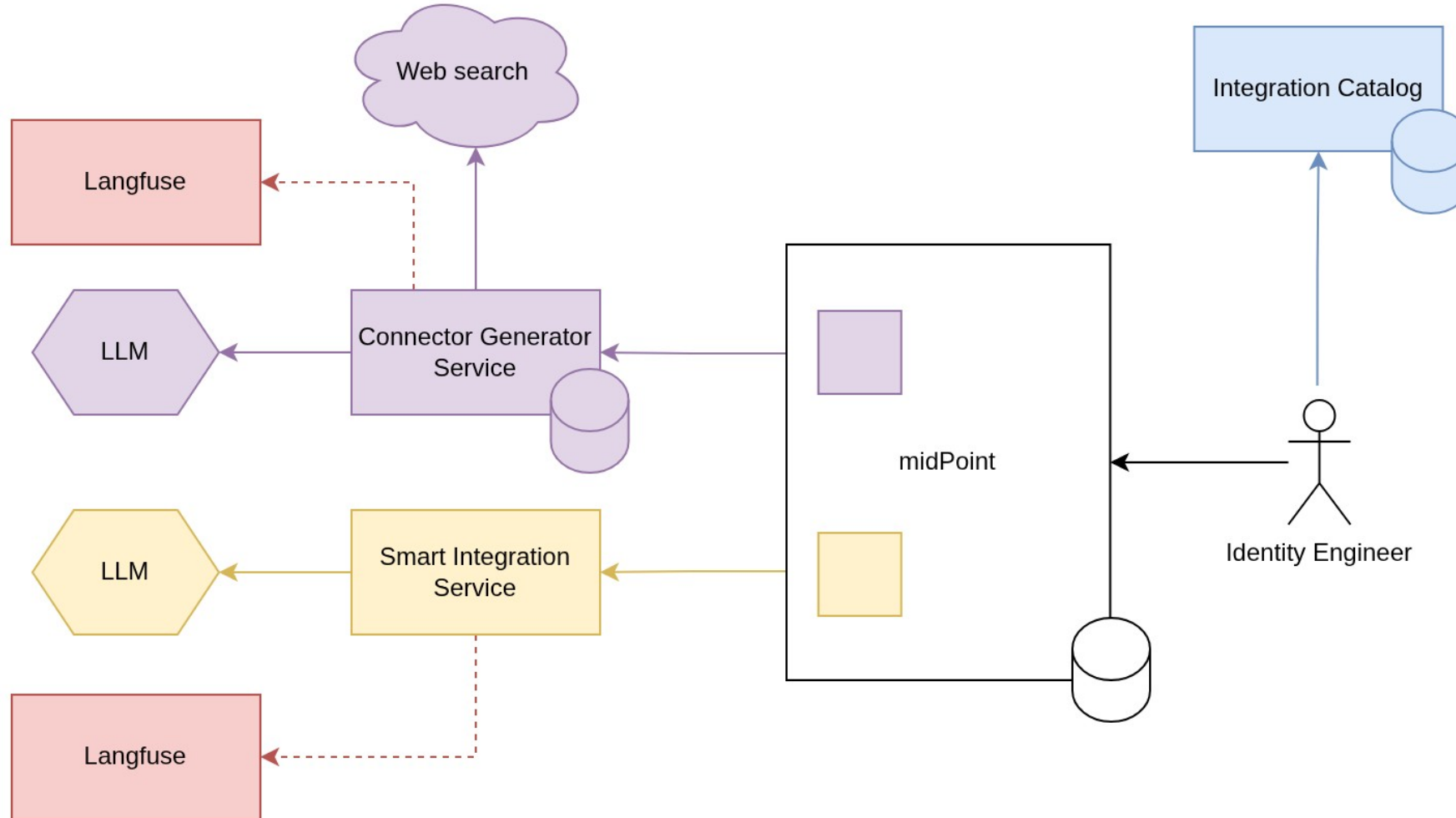
- User correlators: `c:name`, `c:emailAddress`, `c:personalNumber`
- Non-User correlators: `c:name`, `c:identifier`, `c:emailAddress`

#### Semantic meaning (generalized)

- `c:name`: treat as the primary identity handle used to find/reconcile the same entity. Match username/login handle, principal-style sign-in name, canonical object/account name, directory naming identity, alias, service/role/group handle, or provider-specific lookup handle. If a stable key is operationally used as primary lookup identity, it is a plausible `c:name` candidate even if formatted like an identifier.

## MidPilot Deployment

# MidPilot Deployment



# MidPilot Deployment

```
LLM__OPENAI_API_BASE=https://api.openai.com/v1  
LLM__OPENAI_API_KEY=my_api_key  
LLM__MODEL_NAME=openai/gpt-5.5
```

```
LLM__OPENAI_API_BASE=https://bedrock-runtime.us-west-2.amazonaws.com/openai/v1  
LLM__OPENAI_API_KEY=my_api_key  
LLM__MODEL_NAME=openai.gpt-oss-20b-1:0
```

```
LLM__OPENAI_API_BASE=http://localhost:11434/v1  
LLM__OPENAI_API_KEY=ollama  
LLM__MODEL_NAME=mistral-small3.2
```



# MidPilot Deployment

- Self-hosted services
  - Self-hosted LLM models
  - External LLM providers and models
- Evolveum hosted services and LLM models

## Notes on using AI in midPoint

# Notes on using AI in midPoint

- AI features in midPoint are optional
- Always know when working with AI
- Always in control of your data

# Notes on using AI in midPoint

No mapping to show

There are no mappings yet. Create one, or ask for a suggestion.

[+ Add inbound](#)

[Generate suggestions](#)

## Allow AI to analyze your resource data?

To generate accurate suggestions, selected resource data may be sent to an external AI service for processing.

No authentication credentials or user passwords will be shared.

Without the selected data, suggestions may be limited to built-in heuristics or be unavailable.

Select which data can be used:

- Schema  
Use the resource and midPoint schema.
- Raw data  
Use a small sample of resource and midPoint data.

[Learn more](#)

Cancel

Allow and continue

# Notes on using AI in midPoint

● Resource data

**Object class \*** ⓘ  
inetOrgPerson

**Auxiliary object class** ⓘ + Add value 🗑️ Clear all

**Filter** ⓘ + Add value 🗑️ Clear all

**Object class (base context)** ⓘ  
organizationalUnit 🔍 AI

**Filter (base context)** ⓘ  
c:attributes/ri:dn = "ou=users,dc=example,dc=com" 🔍 AI

👁️ Show empty fields

🔍 Suggestion ⋮

**Name correlator**

Suggested based on matching of attributes/employeeNumber to name

+ name (Exact)

**Stats**

1.0 Weight	1 Tier	97.0 Efficiency
---------------	-----------	--------------------

**Actions**

✕ Discard ✓ Accept

[📄 View rule](#)

AI-powered connector development

Insights & Lessons learned

# MidPoint Connector Generator: Agenda

- Expectation vs. reality
- Where AI brings value
- Where AI fails
- What worked in practice
- Lessons learned and bottlenecks



## Why lessons learned matter

- Proofs of concept are easy; reliable production workflows are much harder
- In identity systems, a confident wrong answer can create security and operational risk
- The key question is whether AI output can be trusted, validated, and operated safely



# Expectations vs. reality

## Initial assumption

One generic prompt can handle any vendor

AI can fully automate identity mapping

A larger model will understand every document

Fluent output is probably correct

## Practical lesson

Small, task-specific prompts produce more reliable outputs

AI provides a fast first draft; engineers remain accountable

Context must be selected, structured, and filtered first

Validation, dry-runs, and expert review are required

# Where AI adds value

## 1. Discovery

Summarize large documentation sets and identify likely entities, operations, and endpoints.

## 2. Drafting

Prepare first-pass schema matches, mappings, connector logic, and test cases.

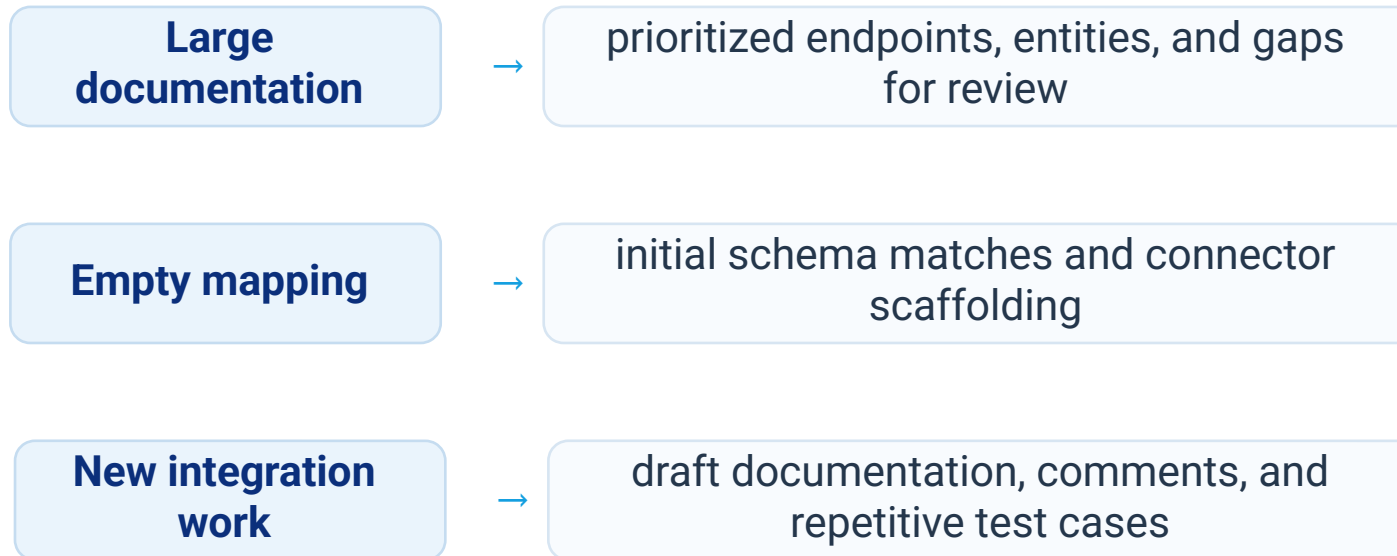
## 3. Review

Explain unfamiliar code, surface assumptions, and suggest the next engineering question.

## 4. Assistance

Support documentation, internal knowledge sharing, and repeated explanations.

# Concrete delivery value



# Where AI creates risk

## Hallucinated fields

The model may invent attributes, endpoints, or operations that do not exist in the target system.

## Incomplete logic

Generated code can cover the happy path while missing exceptions, permissions, or lifecycle details.

## Vendor-specific details

API behavior, pagination, filtering, and error handling often differ from generic examples.

## Confident wording

A polished explanation can hide uncertainty. The output still needs evidence and validation.

**PRACTICAL RULE: VALIDATE EVERY GENERATED ATTRIBUTE, OPERATION, AND MAPPING AGAINST REAL SCHEMA AND RUNTIME BEHAVIOR.**

# Operational risks

## 1. Non-determinism

Similar prompts can return materially different outputs. This complicates versioning, CI checks, and approval workflows.

## 2. Retry cost

Long context windows and repeated attempts can turn simple experiments into real production cost.

## 3. Latency

Unpredictable response times make synchronous designs fragile. Many workflows need asynchronous processing.

# What worked in practice

## 1. Rules for deterministic work

Use deterministic rules for normalization, fixed mappings, naming patterns, and simple transforms.

## 2. Validation before trust

Validate generated outputs with schema checks, linting, tests, preview mode, and dry-runs.

## 3. Human approval

Keep expert approval for security-sensitive mappings, business logic, exceptions, and user-facing changes.

Production-ready AI is hybrid, observable, and intentionally narrow.

# Defensive engineering patterns

- **Use heuristics first**
  - Use normal code for deterministic cleanup whenever it is safer and more predictable
- **Validate strictly**
  - Treat model output as untrusted until validation, linting, and dry-runs pass
- **Provide coded fallbacks**
  - Keep non-LLM fallbacks for timeouts, common defaults, and stable operations
- **Preview before commit**
  - Use preview mode so AI output can be inspected before it changes groovy code

Guardrails are part of the product; the model is only one component.

# Prompting and review patterns

## Prefer narrow prompts

Split large requests into small, operation-specific prompts with explicit output formats.

## Allow refusal

Prefer "I do not know" over guessing. Reject unknown fields and uncertain mappings.

## Keep human review

Require engineering approval for sensitive mappings, exceptions, and final commits.

## Make outputs observable

The team must be able to log, reproduce, inspect, and test each output.

Reliable AI systems are inspectable, reproducible, and reviewable.



# ClearML: Making AI Experiments Measurable

- To use ClearML for repeatable experiment evaluation
- To track selected operations, prompt/model variants, processing time, and evaluation metrics
- To compare outputs against prepared ground-truth datasets
- To measure precision, recall, F1, false positives, false negatives, and result uncertainty
- To replace subjective judgments like “this prompt looks better” with evidence-based decisions.



TASKS LIST SORTED BY

- Updated 2 months ago • Created by Alexander Brecko
- Oracle Netsuite** ✗ Failed  
Updated 2 months ago • Created by Alexander Brecko
- Oracle Netsuite** ✓ Aborted  
Updated 2 months ago • Created by Alexander Brecko
- Openproject-discovery-sc...** ✓ Completed  
Updated 2 months ago • Created by Alexander Brecko
- Openproject-discovery-sc...** ✓ Completed  
Updated 2 months ago • Created by Alexander Brecko
- Atlassian-user-endpoints-...** ✗ Failed  
Updated 2 months ago • Created by Alexander Brecko
- Atlassian-user-endpoints-...** ✗ Failed  
Updated 2 months ago • Created by Alexander Brecko

**Openproject-discovery-scrape** ID: d296ebd7...

[+ ADD TAG](#)

EXECUTION CONFIGURATION ARTIFACTS INFO **CONSOLE** SCALARS PLOTS

Hostname: Macbook-Pro-M3-Max.local [Download full log](#) Filter By Regex

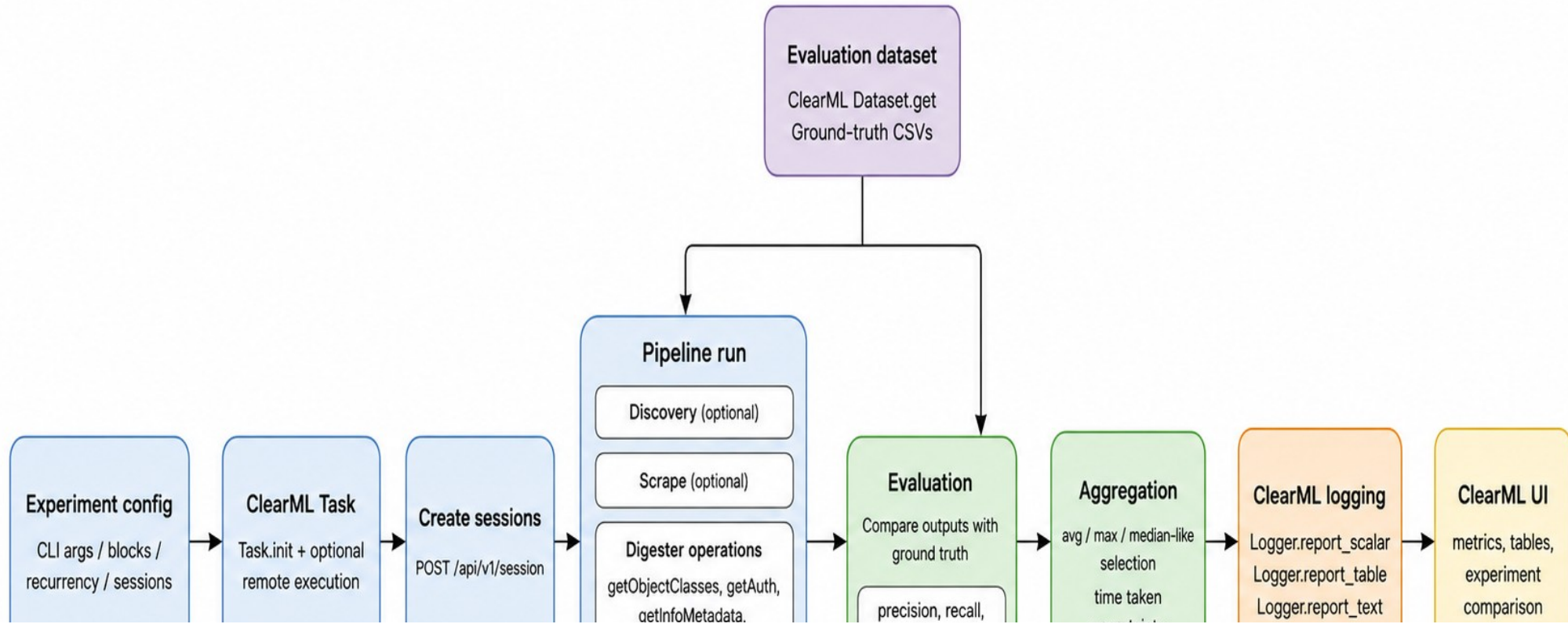
```

Job completed successfully
Result: {"jobId": "848a6525-70d1-41c2-a255-f1b0b959612a", "status": "finished", "createdAt": "2026-02-"}
Endpoints for openproject - user
Completed operation: getEndpoints
Running operation: evaluate_getEndpoints
Base Api Url not found, calling session endpoint
result: [{"endpoints": [{"path": "/my_preferences", "method": "GET", "description": "Get current user"}]}]
Evaluating object class: user
PATH EXTRACTED: /api/v3/my_preferences
PATH EXTRACTED: /api/v3/users
PATH EXTRACTED: /api/v3/users
PATH EXTRACTED: /api/v3/users/{id}
PATH EXTRACTED: /api/v3/users/{id}
PATH EXTRACTED: /api/v3/users/{id}
PATH EXTRACTED: /api/v3/users/{id}/lock
PATH EXTRACTED: /api/v3/users/{id}/lock
=== Evaluation for user ===
TP: 7  FP: 1  FN: 2
Precision: 87.50%
Recall: 77.78%
F1-score: 82.35%
Correct (True Positives):
['DELETE', '/api/v3/users/{id}/lock', '', 'application/hal+json']
['DELETE', '/api/v3/users/{id}', '', 'application/hal+json']

```

[Jump to end](#)

# ClearML Pipeline



# MidPilot-connector-gen pipeline



# Precision over recall

## Typical LLM behavior

High recall: the model suggests many plausible options, including some that may be fabricated.



## Identity requirement

High precision: only verified attributes and operations are acceptable. Guessing is unacceptable.



## Engineering response

Design prompts and validation to prefer "I do not know" over confident false fields.

# Key lessons 1–5

- 1 Use AI for drafts, not decisions** AI can prepare the first draft, but engineers remain accountable for the final result.
- 2 Fluency is not accuracy** Trust validation, linting, and tests rather than polished wording.
- 3 Avoid the magic prompt** Chained, narrow prompts are more reliable than one broad prompt.
- 4 Use heuristics where possible** Predictable tasks should remain in deterministic code whenever possible.
- 5 Limit the context** Filter and digest context before it reaches the model.

## Key lessons 6–10

6

**Build pipelines, not isolated calls**

The workflow around the model matters more than any single API call.

7

**Prioritize precision over recall**

Require the model to reject guesses when the domain requires certainty.

8

**Monitor telemetry**

Latency, retries, and API spend can become architecture issues quickly.

9

**Design for non-determinism**

CI and review workflows must be designed for output variance.

10

**Keep human review**

Human review is often the final security layer and should be designed explicitly.

## Practical outcome

- Faster discovery, drafting, and repetitive engineering work
- Senior engineers spend more time on review, exception and decisions
- Identity-critical delivery still requires expert validation and approval
- The bottleneck remains context quality, validation, and accountable handoff

**AI accelerates engineering work, but it does not replace engineering accountability.**



Assisted classification, correlation and mappings

Insights & Lessons learned

# MidPoint Smart Integration: Agenda

- Object type suggestion
- Schema matching
- Correlation suggestion
- Schema mapping



# Object Type Suggestions

## Delineations

# Delineations: Defining Object Type Boundaries

## Select object type to review

Review the suggested object types generated based on your resource data. Select the one that best matches the kind of objects you want to manage. You can discard and refresh to possibly find new suggestions to review.

### User Accounts

Human user accounts stored under the DN subtree ou=users,dc=example,dc=com. The DN suffix is a stable attribute that uniquely identifies regular user objects. Hide filter ^ ...

Kind: account Intent: user Object class: inetOrgPerson Focus type: UserType

---

Base context filter:  
c:attributes/ri:dn = "ou=users,dc=example,dc=com" AI

Base context object class:  
organizationalUnit

### Service Accounts

Technical service accounts located in the DN subtree ou=service-accounts,dc=example,dc=com. The DN suffix provides a stable partition from regular users. Hide filter ^ ...

Kind: account Intent: service Object class: inetOrgPerson Focus type: UserType

---

Base context filter:  
c:attributes/ri:dn = "ou=service-accounts,dc=example,dc=com" AI

Base context object class:  
organizationalUnit

### Administrator Accounts

Administrative accounts kept under the DN subtree ou=administrators,dc=example,dc=com. The DN suffix is a reliable, immutable marker for admin accounts. Hide filter ^ ...

Kind: account Intent: admin Object class: inetOrgPerson Focus type: UserType

---

Base context filter:  
c:attributes/ri:dn = "ou=administrators,dc=example,dc=com" AI

Base context object class:  
organizationalUnit

Delineations are used to split objects from a resource into **logical groups**, so MidPoint knows how each object type should be classified, synchronized, and processed.

# How We Use AI for Delineations

We mirrored domain experts' reasoning into measurable statistics and provided those signals to the LLM with explicit decision logic in the prompt.

Statistical data for "inetOrgPerson" object class

Sample size: 44 of 44 | Attribute count: 51

Total values: 44 | Unique values: 44 | Empty values: 0

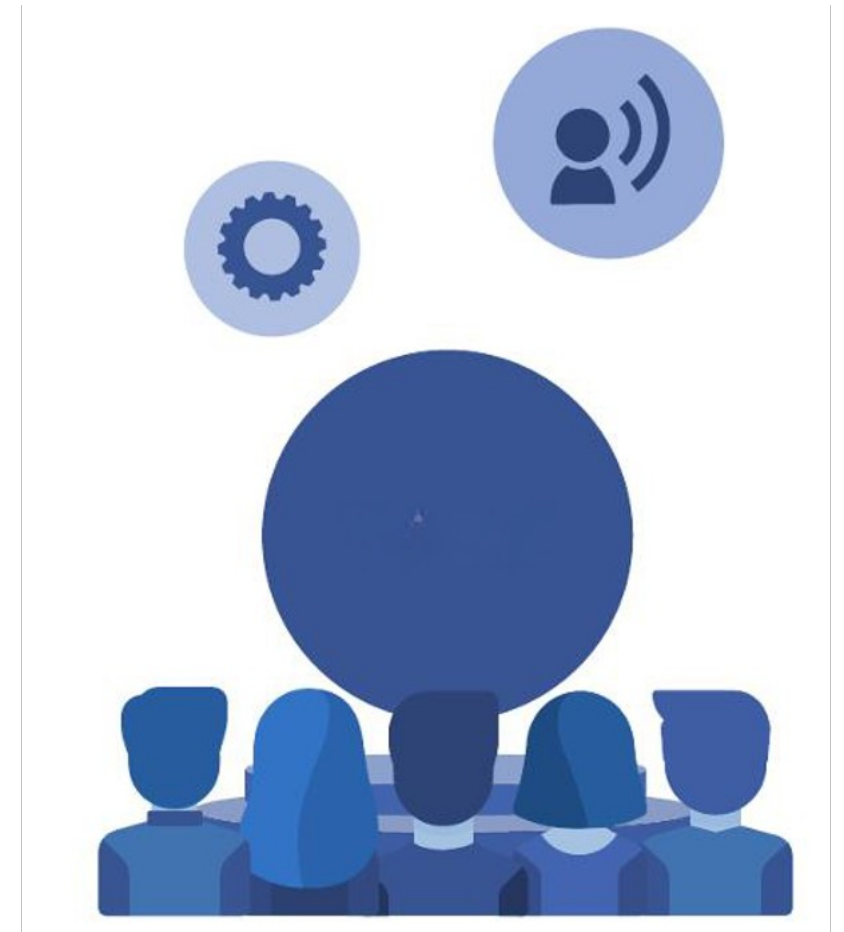
Value Counts | Value Patterns

Type	Value	Count
DNsuffix	ou=users,dc=example,dc=com	39
DNsuffix	ou=service-accounts,dc=example,dc=com	3
DNsuffix	ou=administrators,dc=example,dc=com	2

Close | Data collected at 2026-05-05T16:28:31.279+02:00 | Regenerate statistics

# Lessons Learned

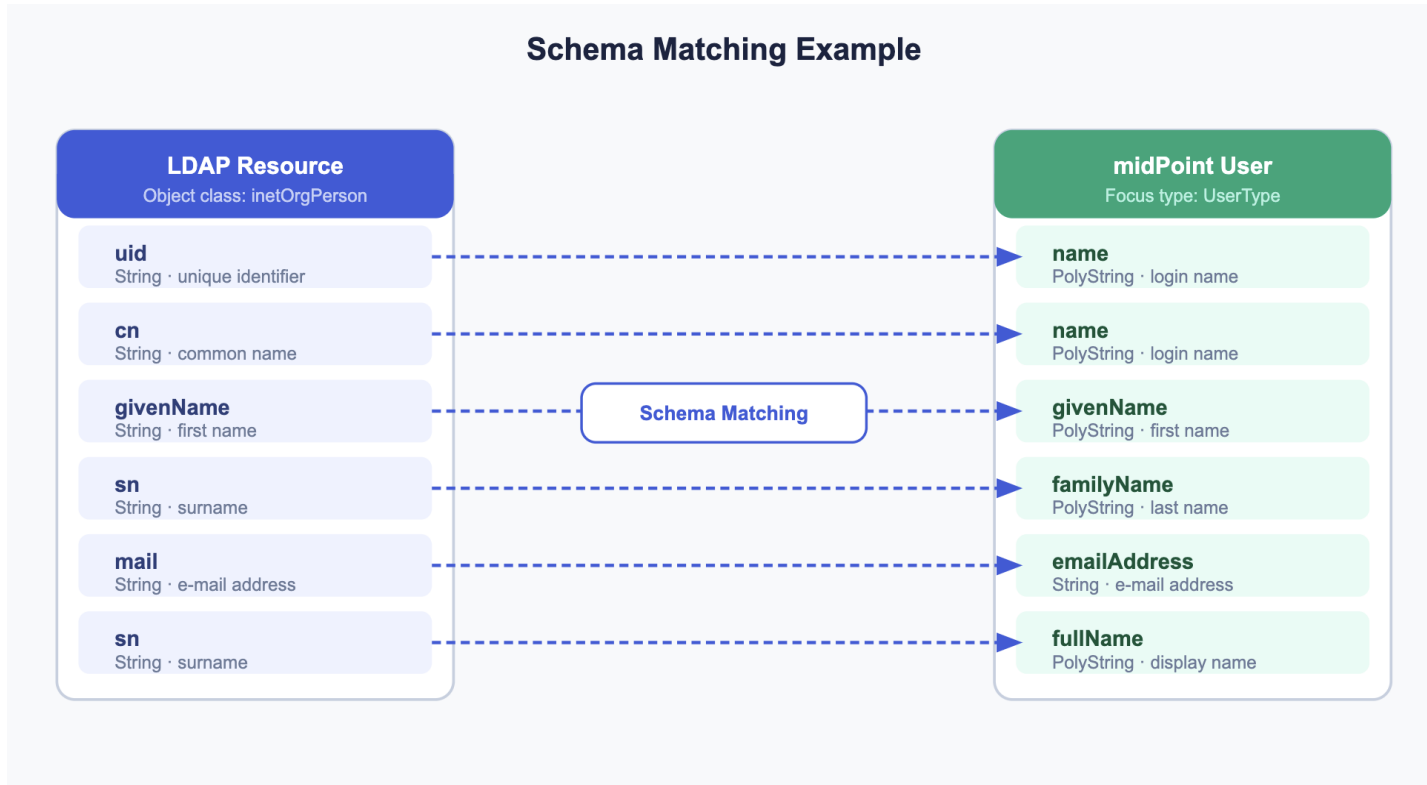
- **Domain experts** became an inseparable part of the system design
- **Human-in-the-loop** is needed also during evaluation
- The LLM worked well as a **junior domain assistant**
- **More context did not** mean better answers
- The biggest gains came from **prepared inputs**



# Schema Matching

## The Hidden Helper

# Schema Matching: The Hidden Helper



**Schema matching** helps us understand **which attributes** from an external resource **correspond to attributes** in MidPoint.

user\_first\_nm, givenname, fname, firstName, and so on.

# Schema Matching: From Heuristics to AI Suggestions

- Heuristics run first
- LLM receives only structured schemas
- LLM is asked to find semantic matches
- MidPoint merges both results
- Hallucinated matches are removed



# Schema Matching: Results

MODEL	PRECISION	RECALL	F1	HALLUCINATIONS
<b>openai/gpt-oss-20b</b>	0.510	0.822	0.600	6
<b>openai/gpt-oss-120b</b>	0.441	0.868	0.563	3
<b>google/gemini-3-flash-preview</b>	0.410	0.920	0.543	7
<b>nvidia/nemotron-3-nano-30b-a3b</b>	0.516	0.598	0.528	66
<b>openai/gpt-5.4</b>	0.310	0.962	0.456	0

# Suggest Correlation

## From Schema Match to Identity Match

# Correlation: From Schema Match to Identity Match

Once **schema matching** tells us which attributes correspond to MidPoint properties, **we score predefined candidates and select the best one.**

## Predefined correlator candidates:

**User:** name, personalNumber, emailAddress

**Non-user:** name, identifier, emailAddress

The screenshot displays the MidPoint web interface for editing a resource. The left sidebar contains a navigation menu with categories like Services, Policies, Cases, Certification, Server tasks, Nodes, Reports, Simulations, Analytics, Audit Log Viewer, and INTEGRATION. The 'Resources' section is expanded, showing options like All resources, All resource templates, New resource, Edit resource, and Import resource definit... The main content area shows a configuration for a correlation rule. At the top, it says 'one or more correlation rules using ambiguous ca'. A purple notification box indicates 'Suggestions generated' and '1 suggestions have been generated. Review and apply them.' Below this, there's a toggle for 'Suggestions Enabled' which is currently 'ON'. A modal window displays a 'Name correlator' suggestion. The suggestion is based on matching attributes/empnum to name. It shows a '+ name (Exact)' suggestion. The 'Stats' section displays '1.0 Weight', '1 Tier', and '100.0 Efficiency'. The 'Actions' section has 'Discard' and 'Accept' buttons. A 'View rule' link is at the bottom of the modal.

# Schema Mapping

## How Data Moves Between Systems

# From Schema Matching to Schema Mapping



A **mapping** defines **how** a value is **copied**, **transformed**, or **generated** between a resource object and a midPoint object.

## Simple idea:

Schema **matching** tells us what belongs together. **Mapping** tells us how the value should move or change.

# Mappings: How Data Moves Between Systems

The screenshot displays the 'Inbound mappings (to MidPoint)' section of the MidPoint configuration tool. It features a table of mappings with columns for Name, Resource attribute, Expression, MidPoint property, and Lifecycle state. The 'Suggestions Enabled' toggle is turned ON. A legend indicates that purple icons represent AI-powered suggestions and blue icons represent system suggestions. The table lists several mappings, including 'sn-into-familyName', 'cn-into-fullName', 'givenName-into-givenName', 'l-into-locality', 'employeeNumber-to-name', 'employeeNumber-into-personalNumber', and 'title-into-title'. Each mapping row includes a checkbox, a status icon, the mapping name, source and target attributes, an expression (e.g., 'As is' or 'Script All set'), a 'Show script' button, the target MidPoint property, and a lifecycle state dropdown set to 'Active (production)'. A search bar and a '+ Add inbound' button are also visible.

Name	Resource attribute	Expression	MidPoint property	Lifecycle state
sn-into-familyName	sn	As is	familyName	Active (production)
cn-into-fullName	cn	Script All set	fullName	Active (production)
givenName-into-givenName	givenName	As is	givenName	Active (production)
l-into-locality	l	Script All set	locality	Active (production)
employeeNumber-to-name	employeeNumber	As is	name	Active (production)
employeeNumber-into-personalNumber	employeeNumber	As is	personalNumber	Active (production)
title-into-title	title	As is	title	Active (production)

- The resource stores names in **uppercase** (JOHN DOE), MidPoint expects **title case**
- A date comes in as **20250101**, midPoint expects **ISO format 2025-01-01**
- An HR system uses **internal codes** (EMP, CONT, SVC) while MidPoint's **lifecycleState** has its own vocabulary (active, archived, ...)

# Mapping Suggestions: Layered Decision Pipeline

- System mappings first
- As-is evaluation
- Heuristic rules
- LLM only when needed
- Categorical mappings
- Parallel processing



# Lessons Learned & Wrap-Up

# Lessons Learned & Wrap-Up



- **Prepared inputs** matter more than model choice
- **Prompt engineering** is a first-class engineering discipline
- **Thoughtful UX** is what makes AI integration actually usable
- **Validate** before trusting
- **Human-in-the-loop** remains essential
- **Score** your outputs – and show those scores to the user
- **Model choice is always a time/quality/stability trade-off** – and the parameters matter...



# Evolveum

## Thank you for your attention

Feel free to ask your questions now!



Funded by the  
European Union  
NextGenerationEU

[RECOVERY  
AND RESILIENCE]  
PLAN



2nd Annual  
MidPoint Community Meetup

