A Lightweight Analysis Facility for the DARWIN Experiment

Analysis Facilities Workshop - Munich 2024

Sebastian Brommer[1] // Florian von Cube[1] // Manuel Giffels[1] // Robin Hofsaess[1] // Markus Klute[1] // Benedikt Maier[2] // Matthias Schnepf[1] // Kathrin Valerius[1]

[1] Karlsruher Institut für Technologie

[2] Imperial College London

Computing in non-LHC Collaborations

- Less data and less resource demands
- Less complex computing infrastructure required
- More flexible in adapting new concepts
- Open to new ideas

› Source: AI generated with Microsoft Copilot

Computing in non-LHC Collaborations

- Less data and less resource demands
- Less complex computing infrastructure required
- More flexible in adapting new concepts
- Open to new ideas

› Source: AI generated with Microsoft Copilot

- Less person power for computing related tasks
- Analysts often have no access to existing analysis facilities e.g. CERN, instead dependent on computing infrastructure of their university group

Computing in non-LHC Collaborations

- Less data and less resource demands
- Less complex computing infrastructure required
- More flexible in adapting new concepts
- Open to new ideas

› Source: AI generated with Microsoft Copilot

- Less person power for computing related tasks
- Analysts often have no access to existing analysis facilities e.g. CERN, instead dependent on computing infrastructure of their university group

Providing a collaboration-wide computing infrastructure can be a challenging task, especially with limited person power

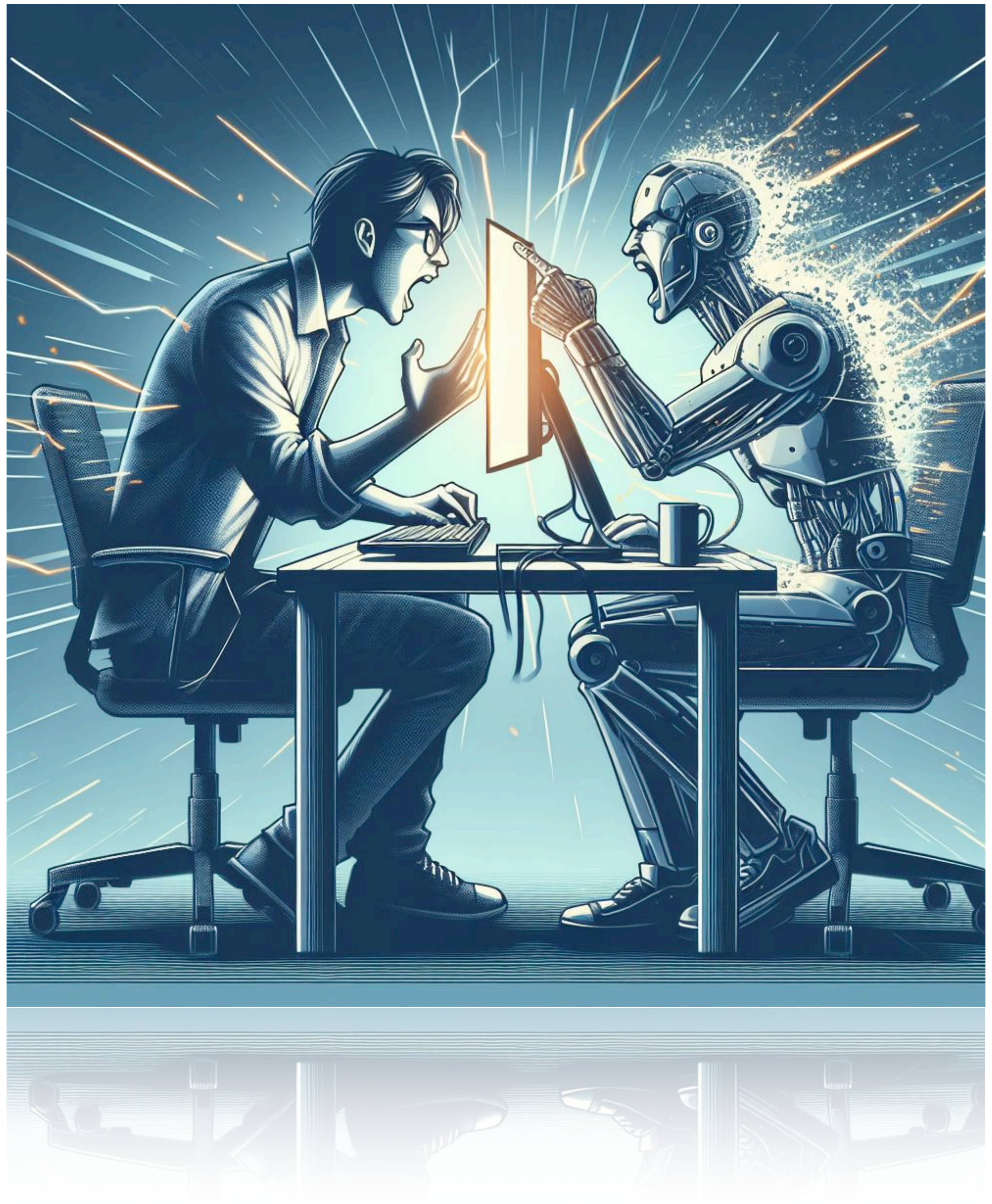
Our view on a lightweight Analysis Facility

- Develop **a future-proof concept** for an analysis facility, that can **serve both analysts and central production needs** of an experiment
- Facility should be **accessible for all collaboration members** to provide a **joint computing platform**
- Facility should be lightweight with **simple deployment**, yet **scalable** according to the computing need
- Rely on **existing and established tools** and experience gained from LHC Computing

Requirements of the Analysis Facility

User side

- Single Sign-On
- Run interactive analysis
- Traditional SSH + batch system
- Common storage entry point
- Run central productions



- Source: AI generated with Microsoft Copilot

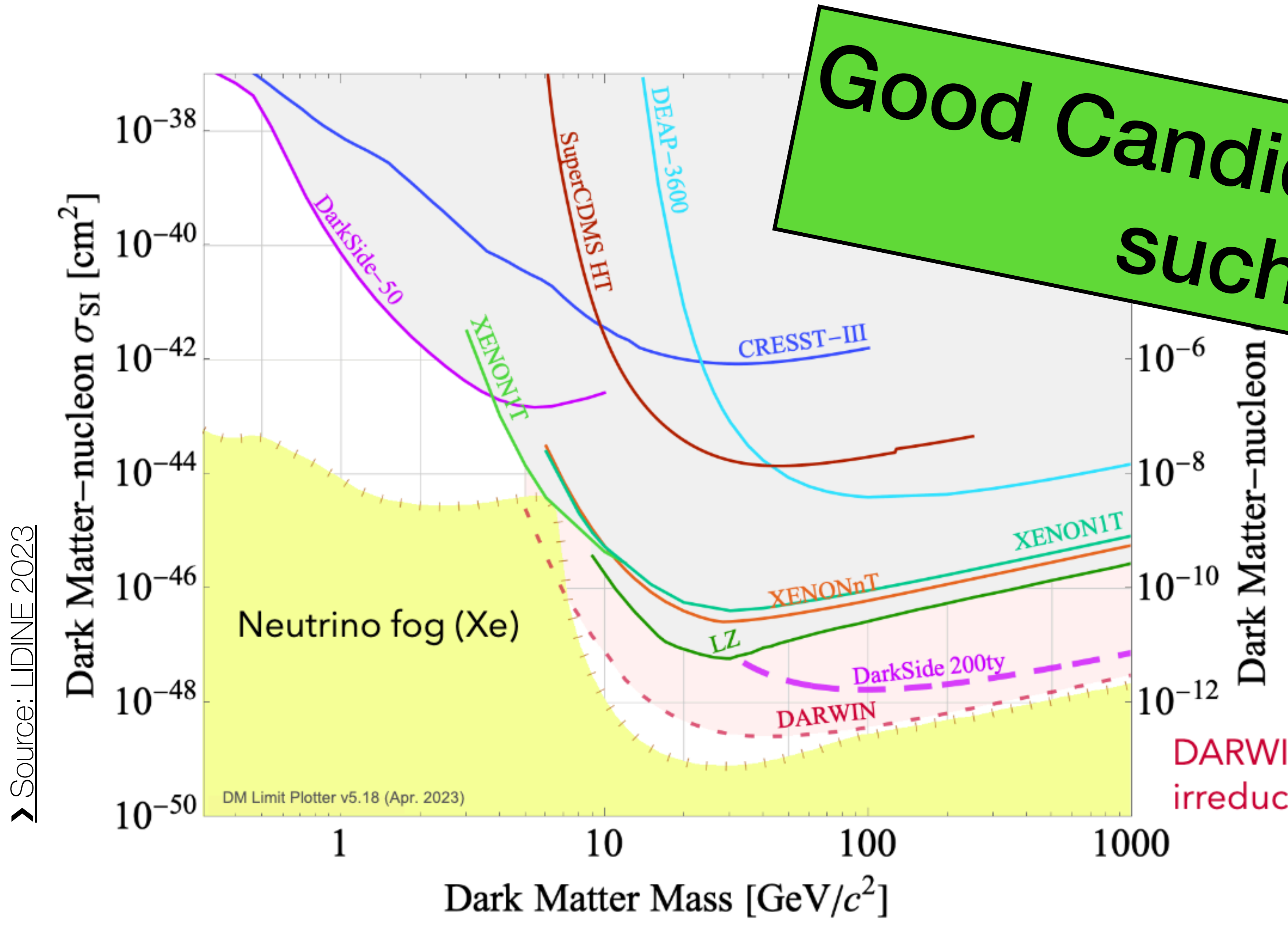
Admin side

- Little user management
- Low maintenance
- Easy deployment
- Scalability

The DARWIN Experiment

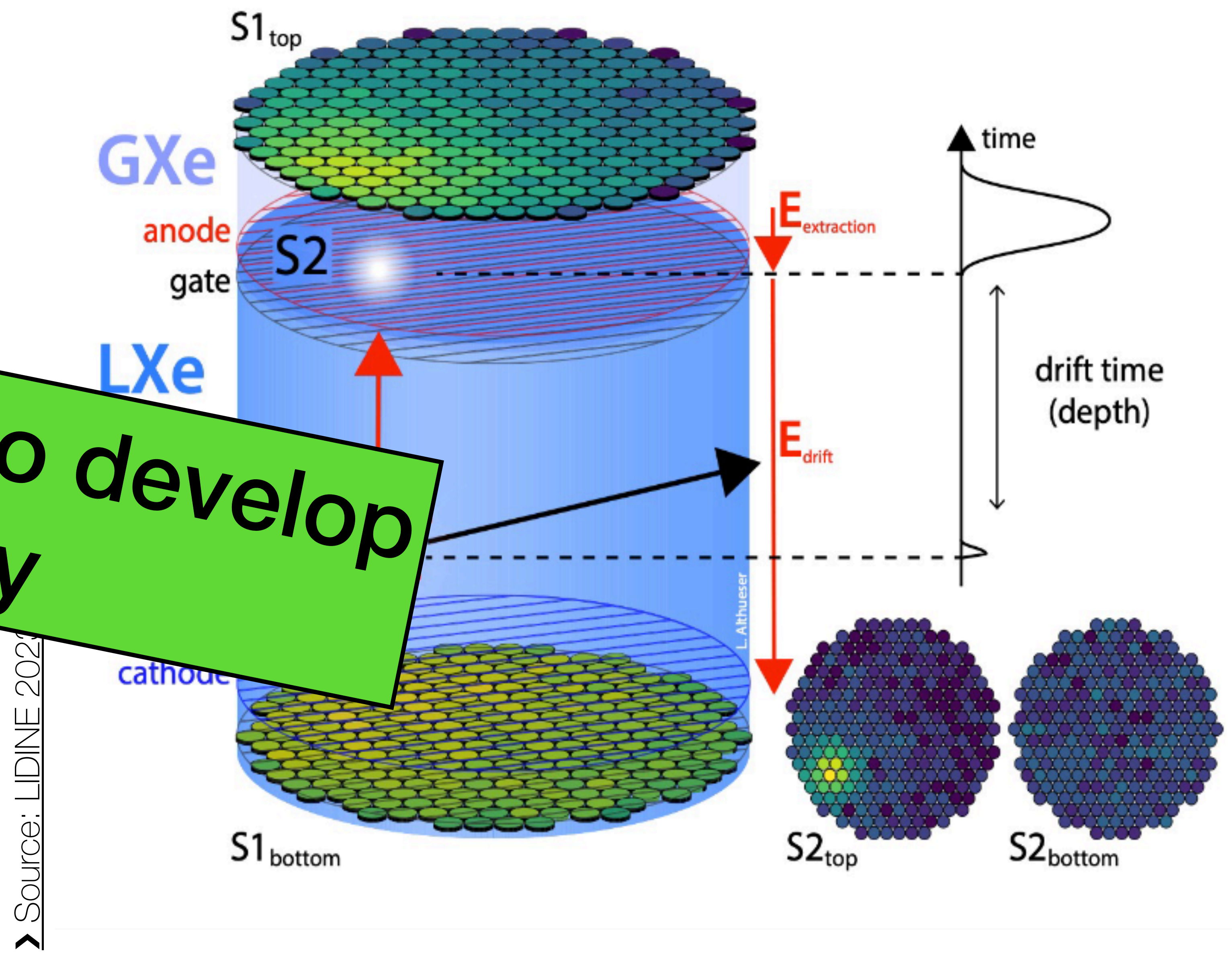
- Direct Dark Matter search experiment with a 50-tonne liquid Xenon detector
- Collaboration has ~200 Members from 35 Institutes
- Currently in R&D Phase, main computing needs are simulations and analysis software development

Good Candidate to profit from and help to develop such a lightweight analysis facility



Source: LIDINE 2023

DARWIN: reach the irreducible neutrino fog



Source: LIDINE 202

JINST (2016) DOI: 10.1088/1475-7516/2016/11/017

Authentication and Single Sign-On

The post x509 era

- User management done in IAM instance hosted at CNAF (thanks!)
- Approval of new users resides with manager from the collaboration
- More **detailed permissions** handled via **group memberships** and protected scopes (to distinguish between analysts and production users)

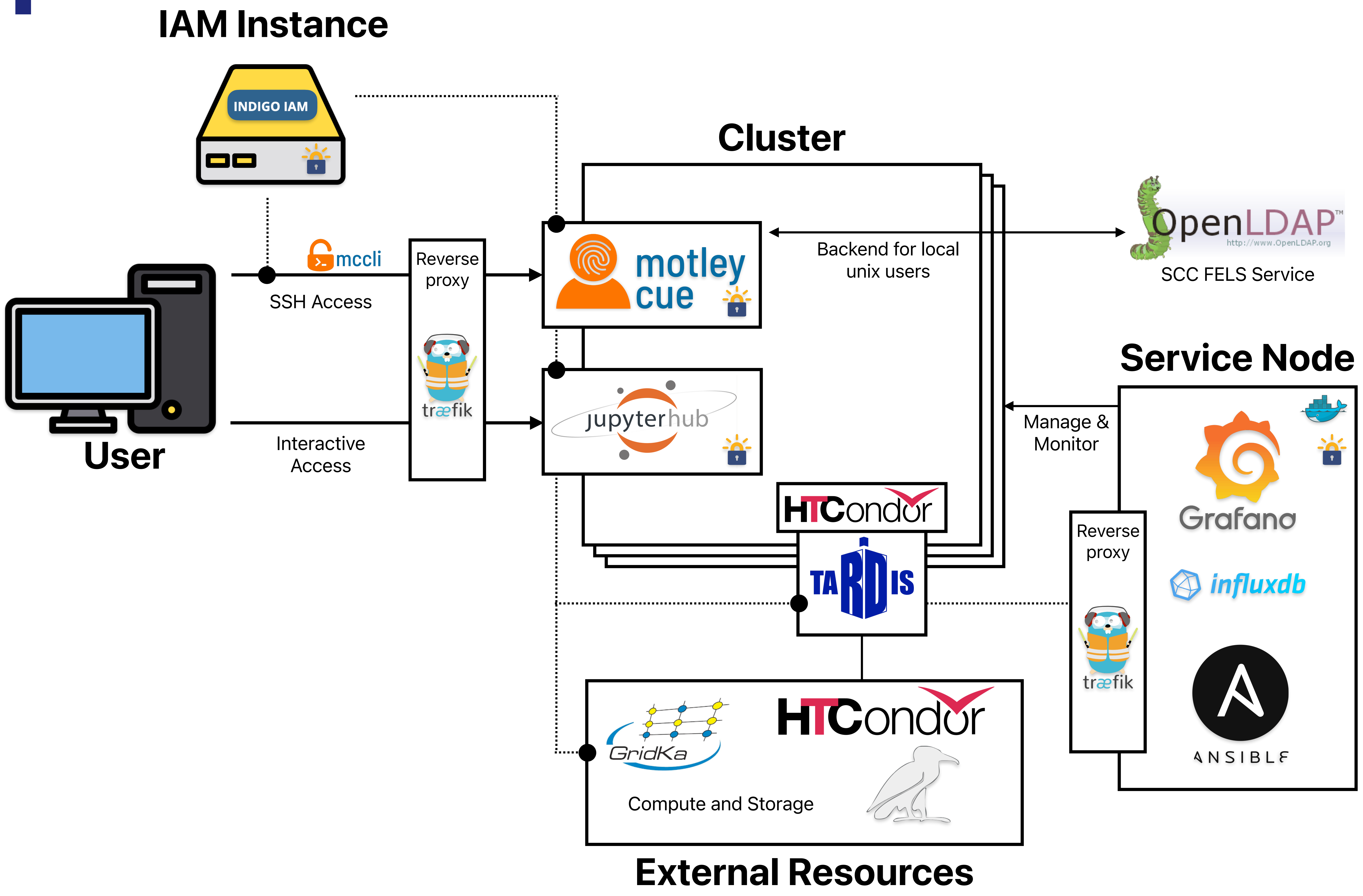
All set for a fully token-based facility

› Source: AI generated with Microsoft Copilot

The Prototype Setup

Prototype Setup consists of **(for now)**

- one cluster node as entry point (RHEL9, AMD EPYC 9654P 96-Core Processor)
- one service node for management and deployment (Ubuntu 22)
- IAM Instance at CNAF
- Computing and storage resources from GridKa

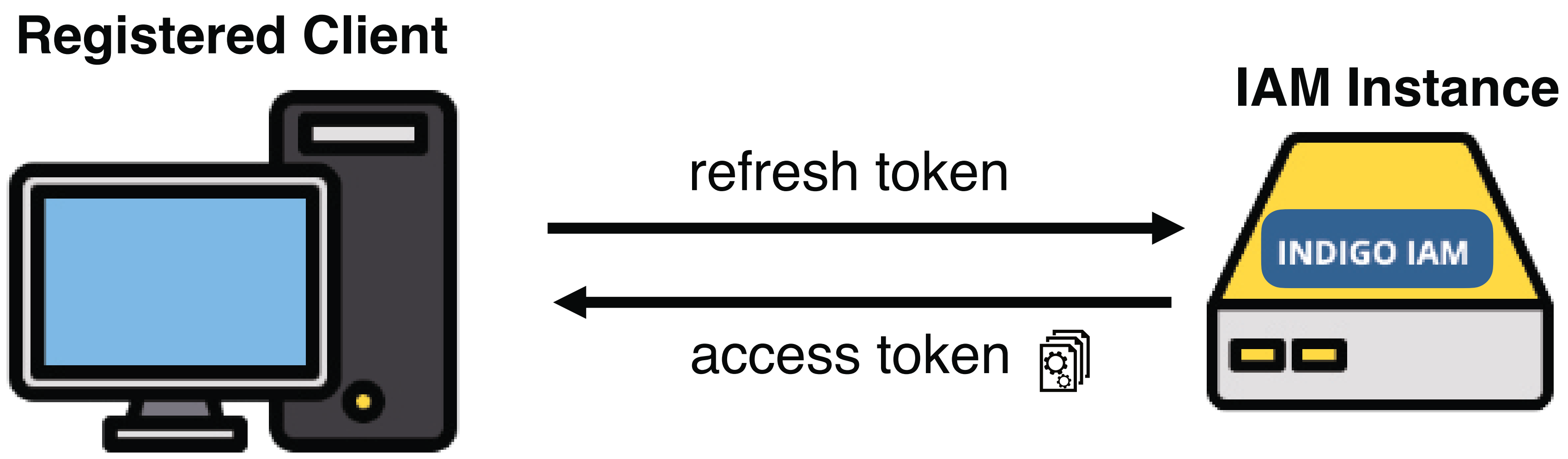


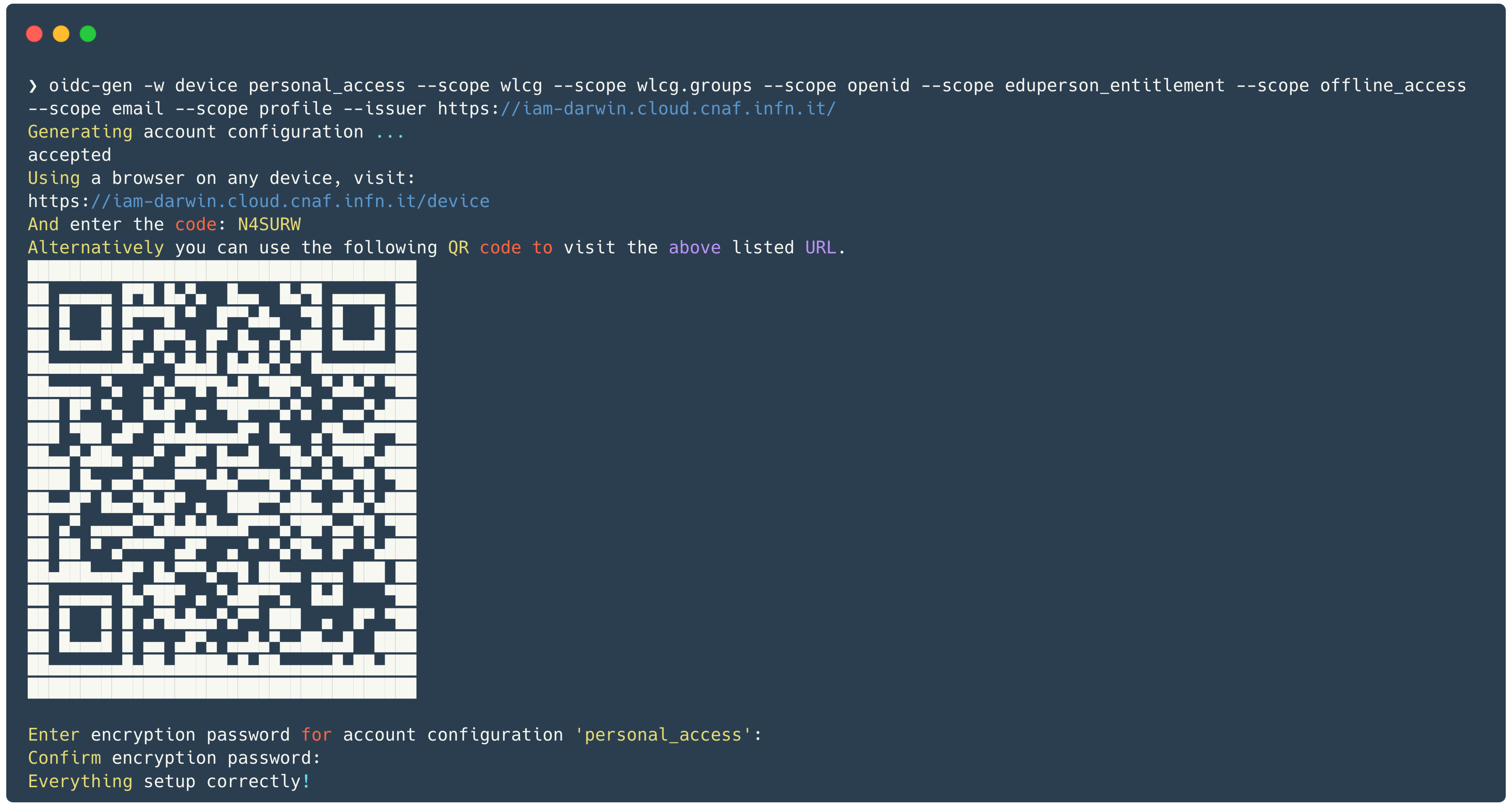
SSH with tokens

Client side

› https://github.com/indigo-dc/oidc-agent

- ***oidc-agent*** to allow users to obtain access tokens on the command line, by registering the client as a device with the IAM





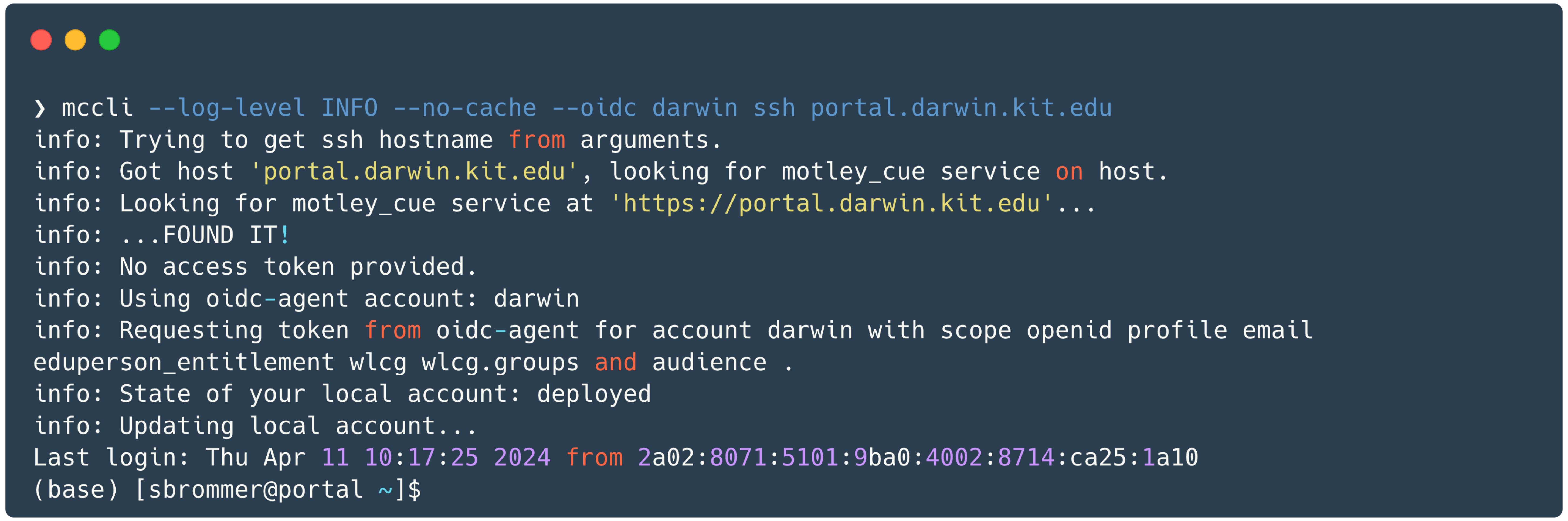
```
❯ oidc-gen -w device personal_access --scope wlcg --scope wlcg.groups --scope openid --scope eduperson_entitlement --scope offline_access
--scope email --scope profile --issuer https://iam-darwin.cloud.cnaf.infn.it/
Generating account configuration ...
accepted
Using a browser on any device, visit:
https://iam-darwin.cloud.cnaf.infn.it/device
And enter the code: N4SURW
Alternatively you can use the following QR code to visit the above listed URL.

Enter encryption password for account configuration 'personal_access':
Confirm encryption password:
Everything setup correctly!
```

SSH with tokens

Client side

- ***oidc-agent*** to allow users to obtain access tokens on the command line, by registering the client as a device with the IAM instance
- ***mccli*** as a wrapper around the regular SSH client



```
❯ mccli --log-level INFO --no-cache --oidc darwin ssh portal.darwin.kit.edu
info: Trying to get ssh hostname from arguments.
info: Got host 'portal.darwin.kit.edu', looking for motley_cue service on host.
info: Looking for motley_cue service at 'https://portal.darwin.kit.edu'...
info: ...FOUND IT!
info: No access token provided.
info: Using oidc-agent account: darwin
info: Requesting token from oidc-agent for account darwin with scope openid profile email
eduperson_entitlement wlcg wlcg.groups and audience .
info: State of your local account: deployed
info: Updating local account...
Last login: Thu Apr 11 10:17:25 2024 from 2a02:8071:5101:9ba0:4002:8714:ca25:1a10
(base) [sbrommer@portal ~]$
```

› https://github.com/indigo-dc/oidc-agent

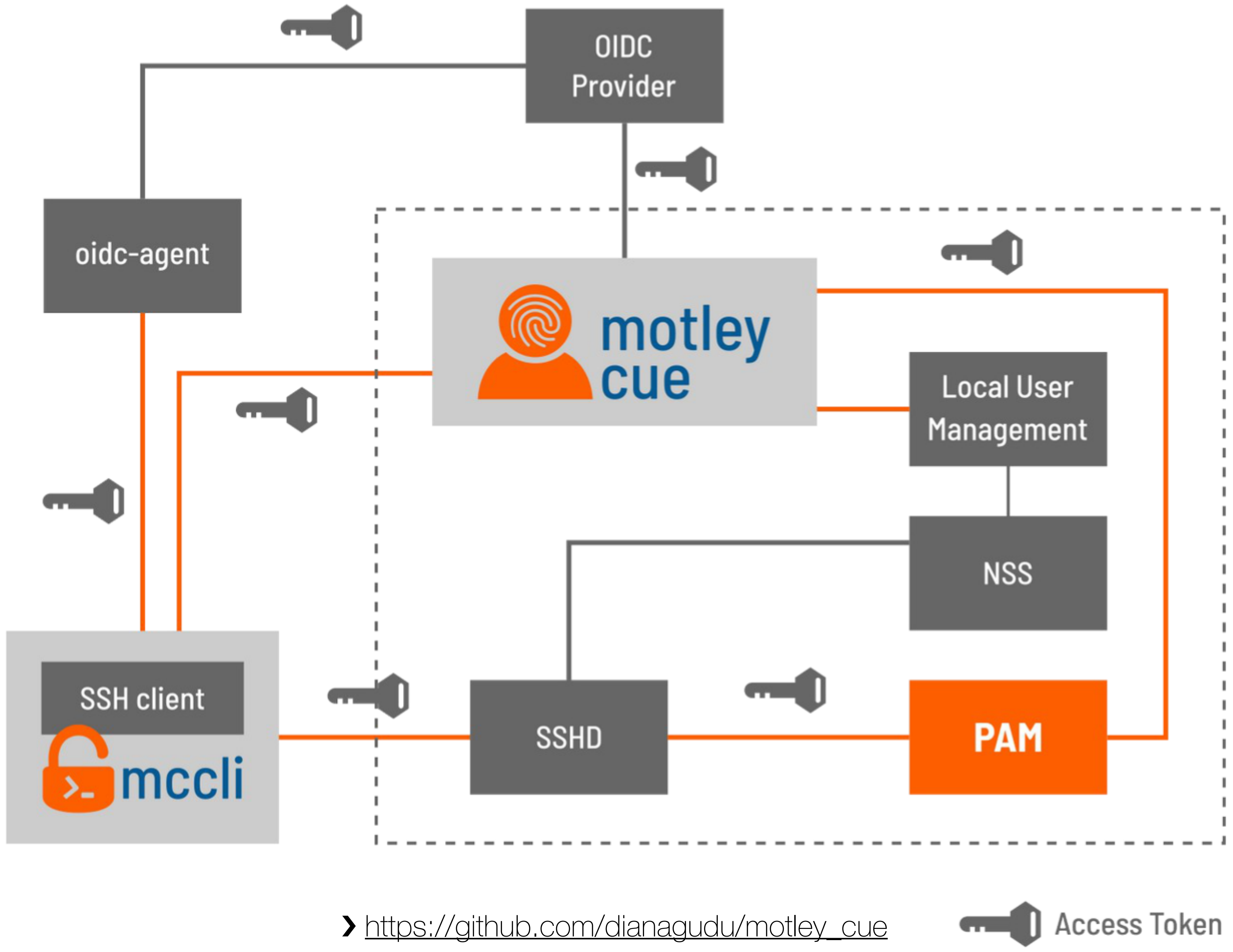
› https://github.com/dianagudu/mccli

SSH with tokens

Server side

motley cue service on the server is able to

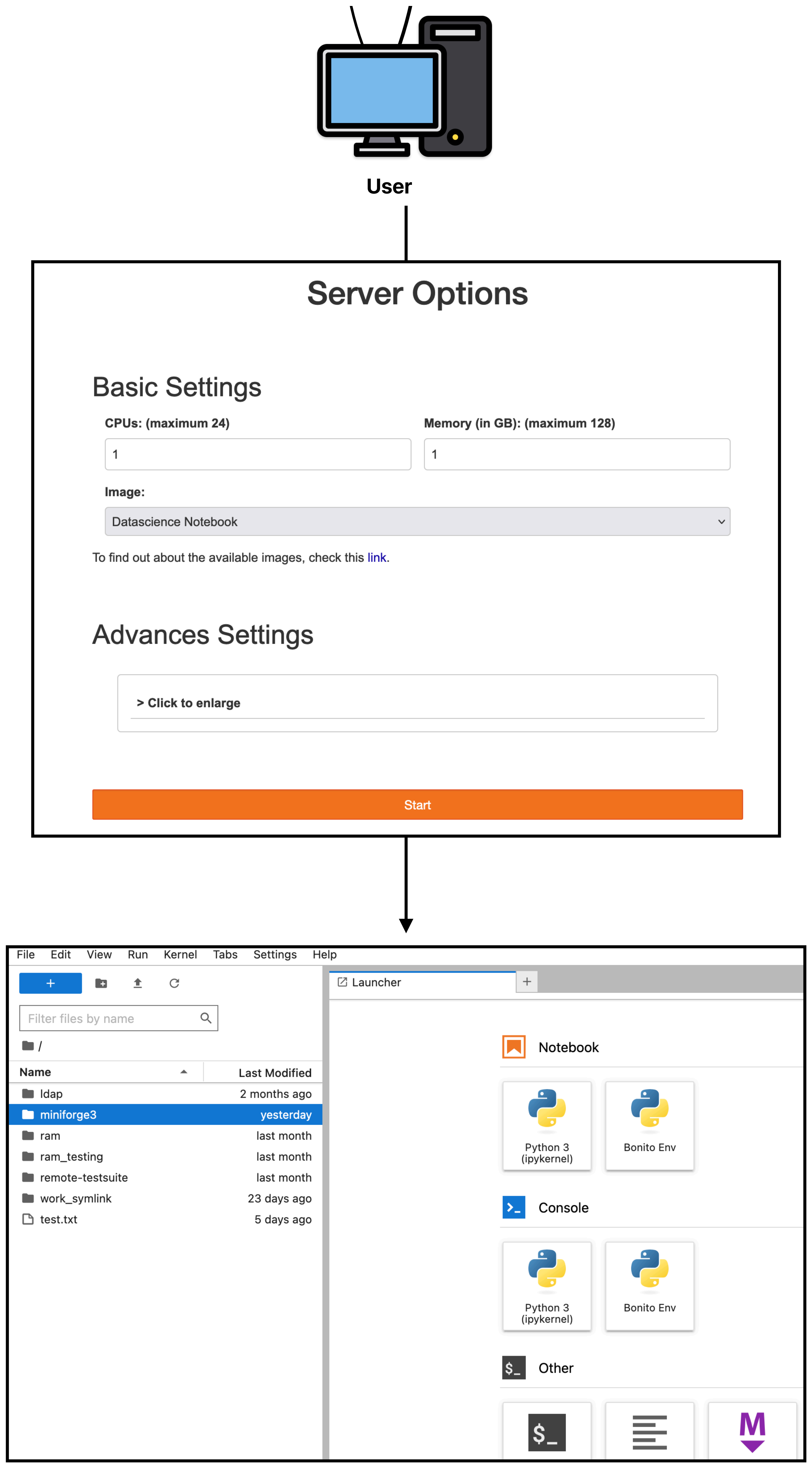
- Validate a access token
- Map the token to a local unix user
- Automatically create a new user if its the first login of the user
- Update groups with every new login, depending on IAM information
- LDAP instance as backend for local users



› https://github.com/dianagudu/motley_cue

Interactive analysis

- Provide interactive usage via a ***juptyerhub*** instance
- After OAuth login, users can spawn their own interactive notebooks
- Notebooks run as docker containers on the cluster, **mapped to their local unix account**
- Users have **full access to local directories**
- Other methods e.g. VSCode server via Tunnel also supported



Server Options

Basic Settings

CPUs: (maximum 24): 1

Memory (in GB): (maximum 128): 1

Image: Datascience Notebook

To find out about the available images, check this link.

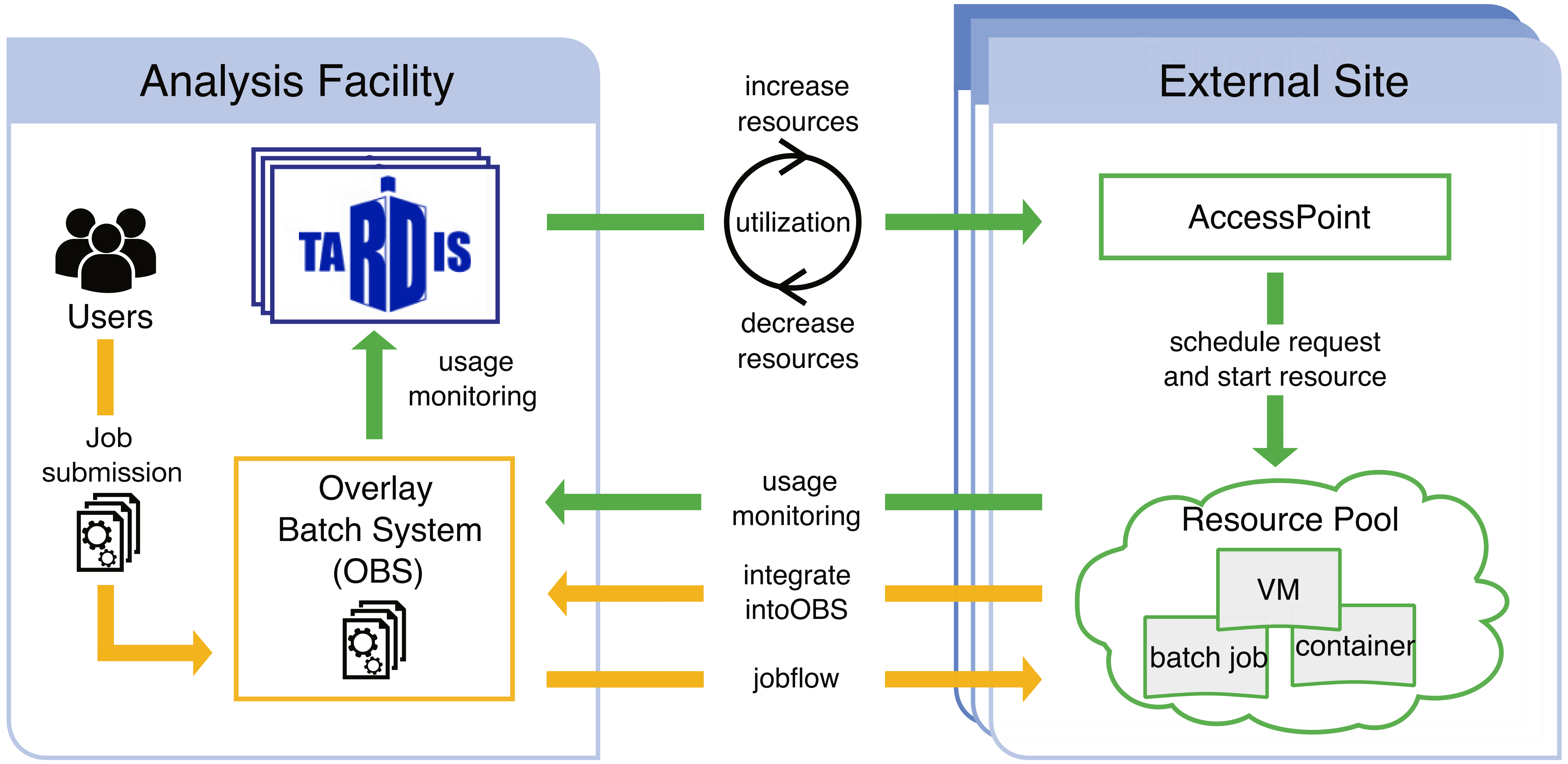
Advances Settings

> Click to enlarge

Name	Last Modified
ldap	2 months ago
miniforge3	yesterday
ram	last month
ram_testing	last month
remote-testsuite	last month
work_symlink	23 days ago
test.txt	5 days ago

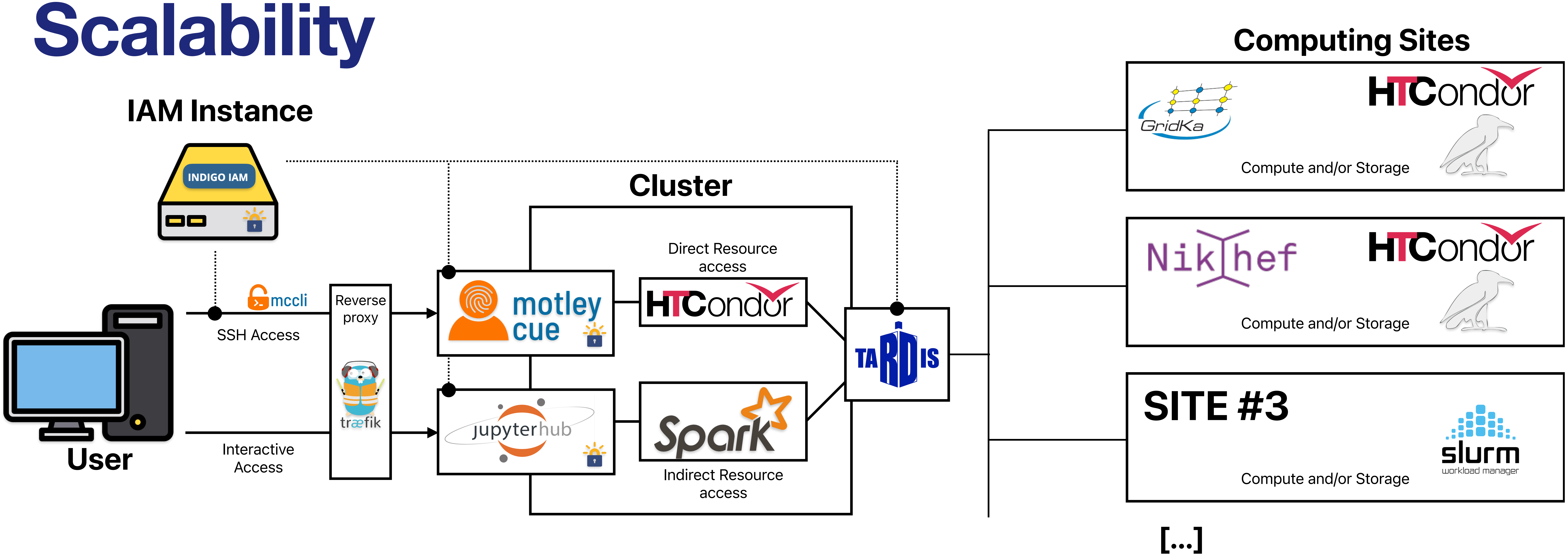
Batch System & Opportunistic Resources

- Batch system **HTCondor** for processing
- Resources from GridKa are integrated into the OBS of the facility via ***COBalD/TARDIS*** using grid standards (submission to GridKa HTCondor CE)
- Dynamic **allocation of additional resources**, if there is demand
- Easy configuration allows integration of resources from **other computing sites**



- https://github.com/MatterMiners/cobald
- https://github.com/MatterMiners/tardis

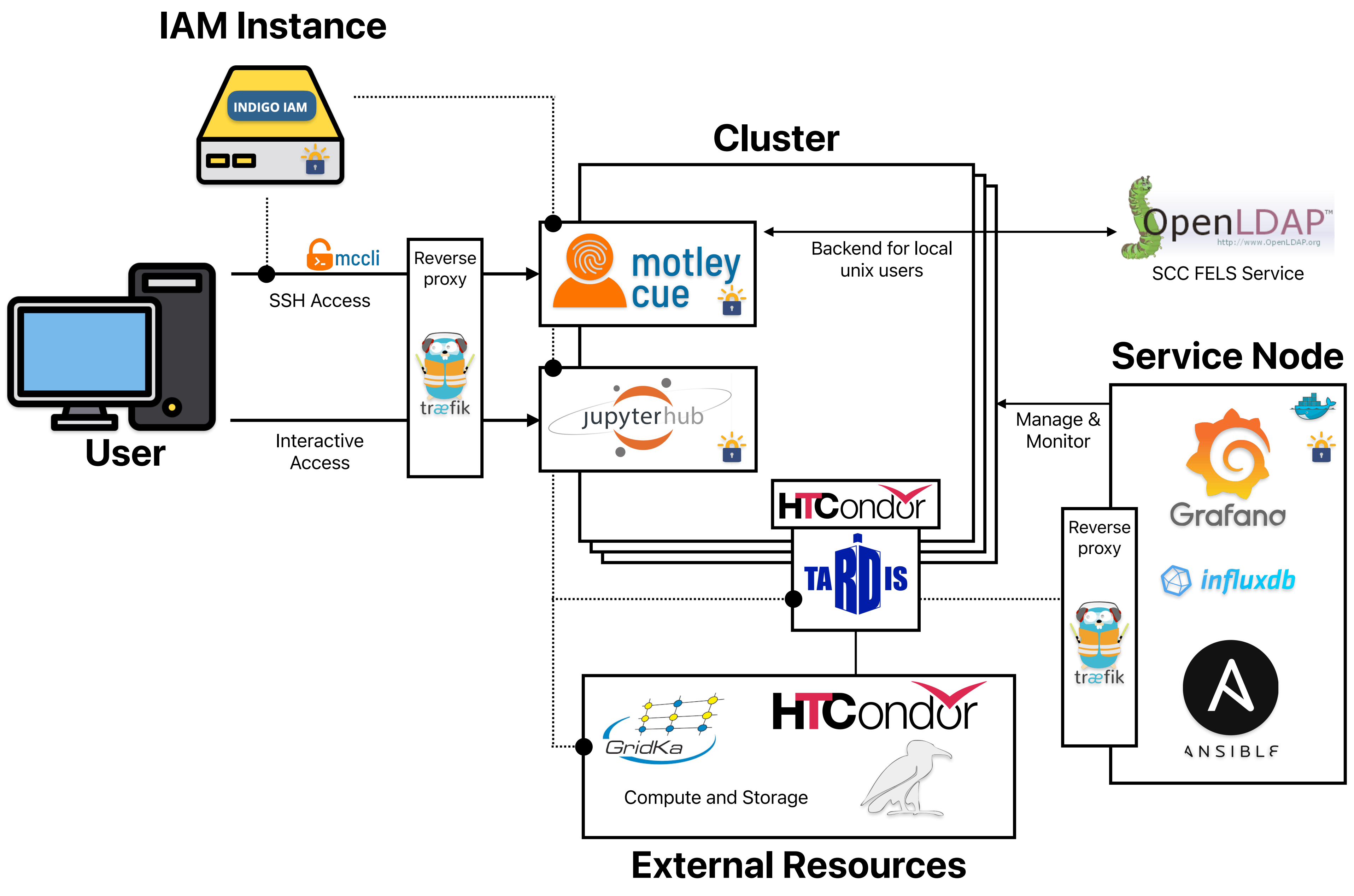
Scalability

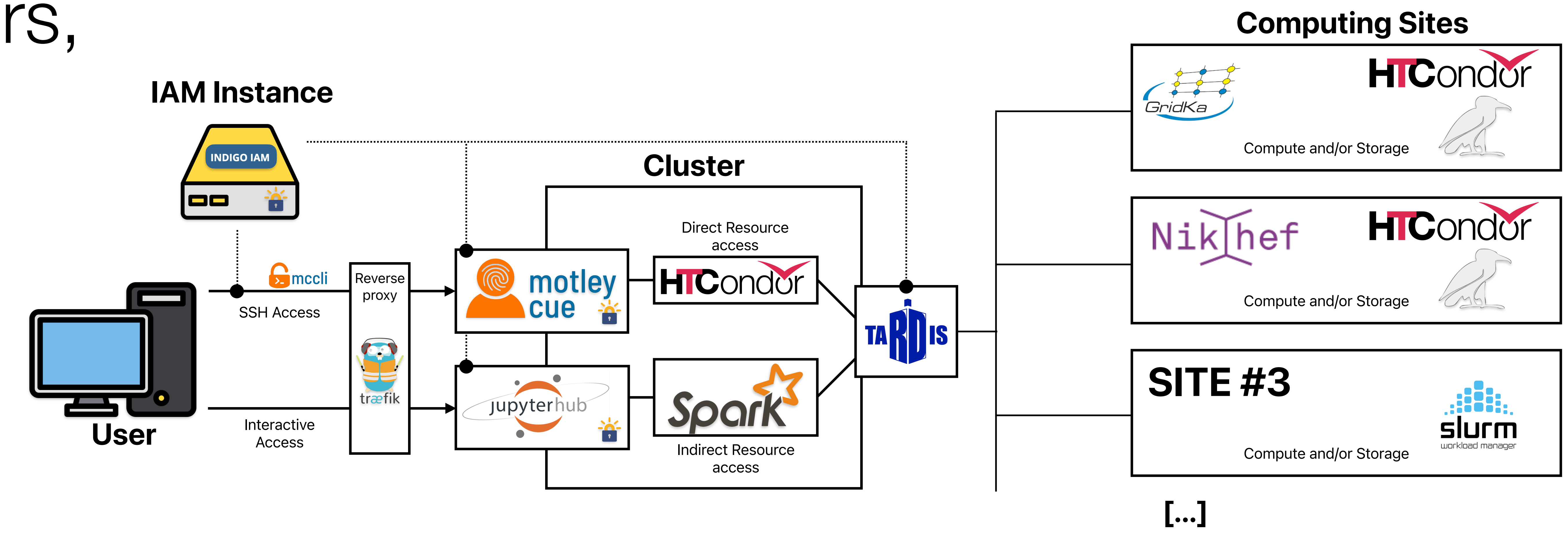


- Expand available computing resources by adding new resources to the overlay batch system using ***COBalD/ TARDIS***
- Scale Jupyterhub via dedicated yarn/spark scheduler utilising the same resources (similar to SWAN, foreseen approach presented here (link))
- Expand local cluster capabilities

Conclusion and Outlook

- Presented a lightweight Analysis Facility that can be easily setup and maintained
- Future-proof fully token based setup with SSO including grid storage (see backup)
- Running prototype instance for the DARWIN Collaboration with first users, testing the facility
- Scalability via inclusion of **external resources**





Questions ?

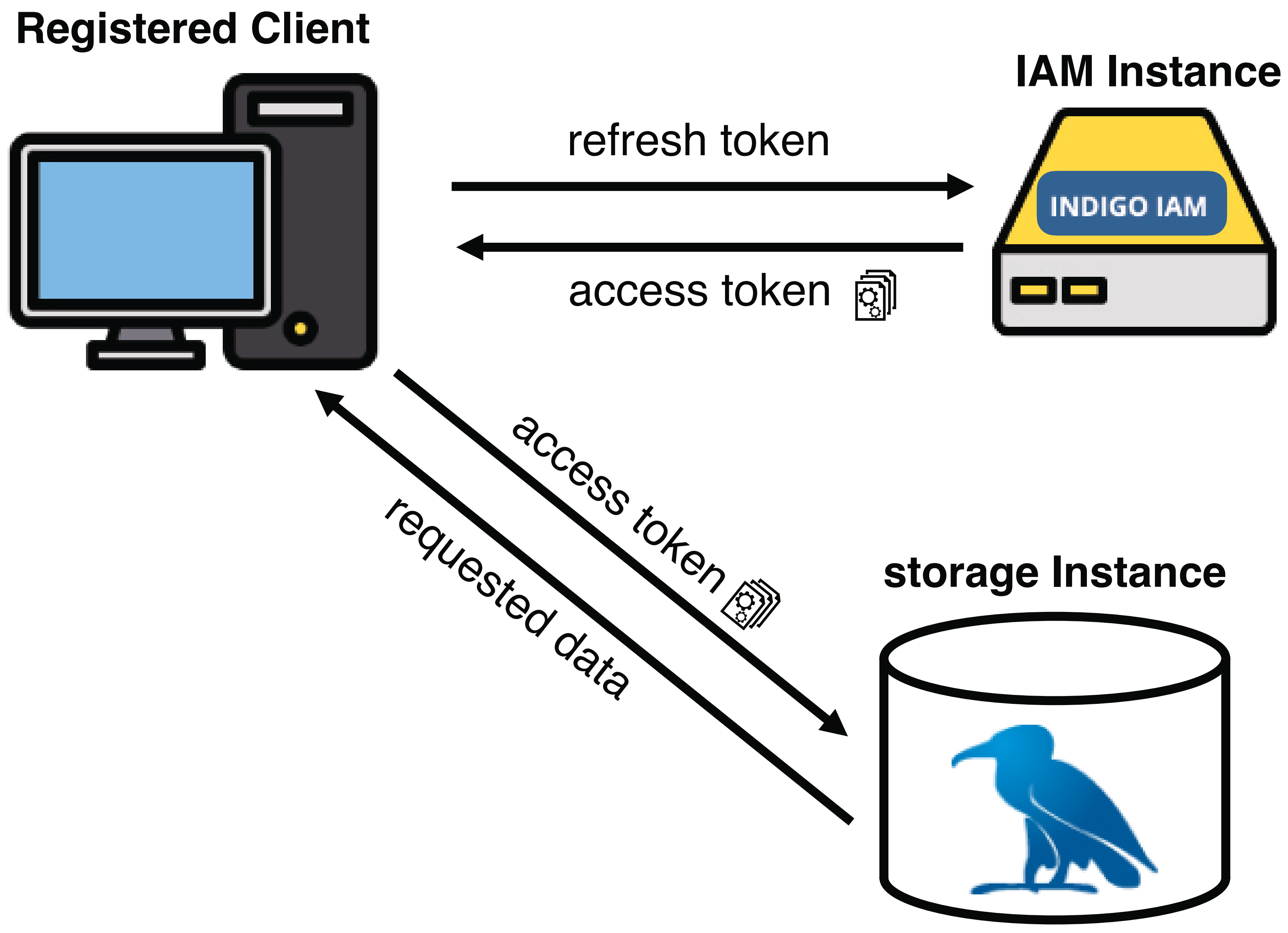
Storage Solutions

Local Storage

- Local storage for software, code development, analysis
- CVMFS for access to software stacks and analysis containers

Remote Storage

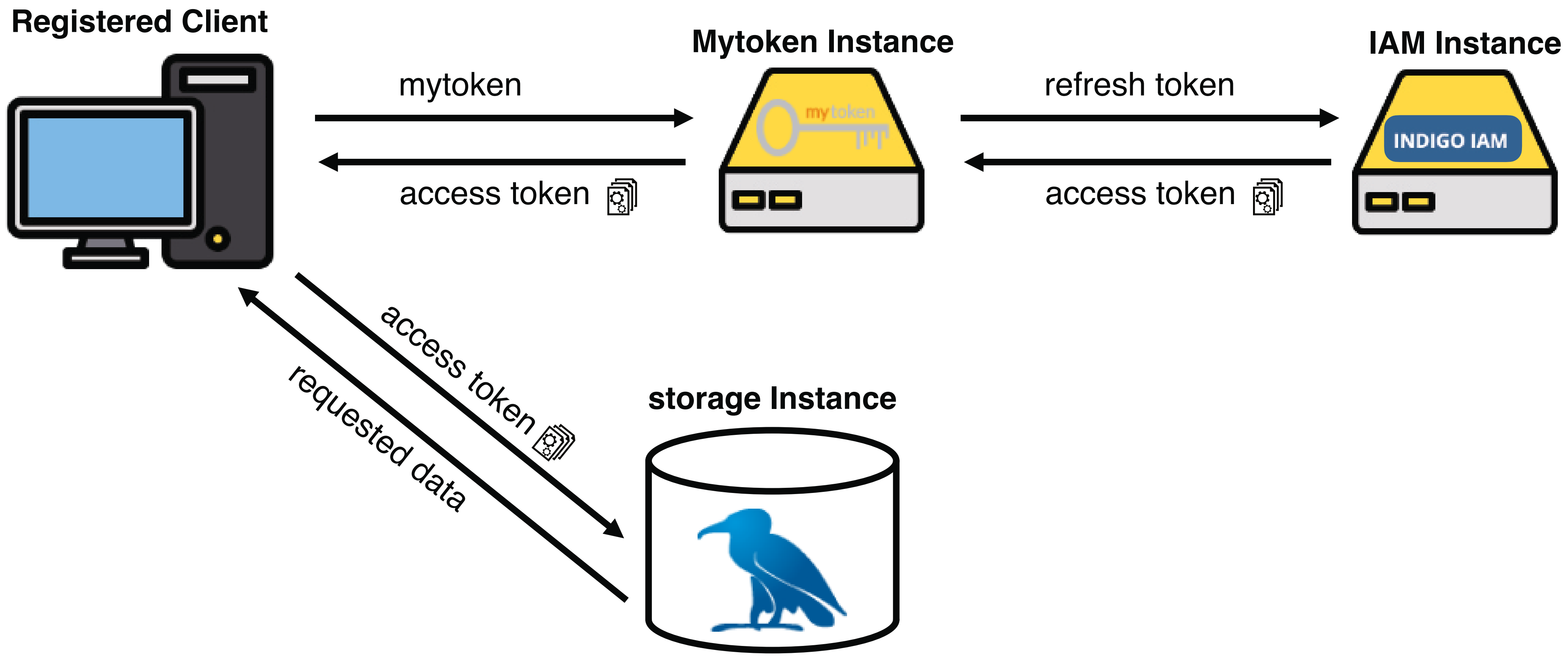
- dCache Instance at GridKa for grid storage of users and central production
- Access to dCache only via access tokens

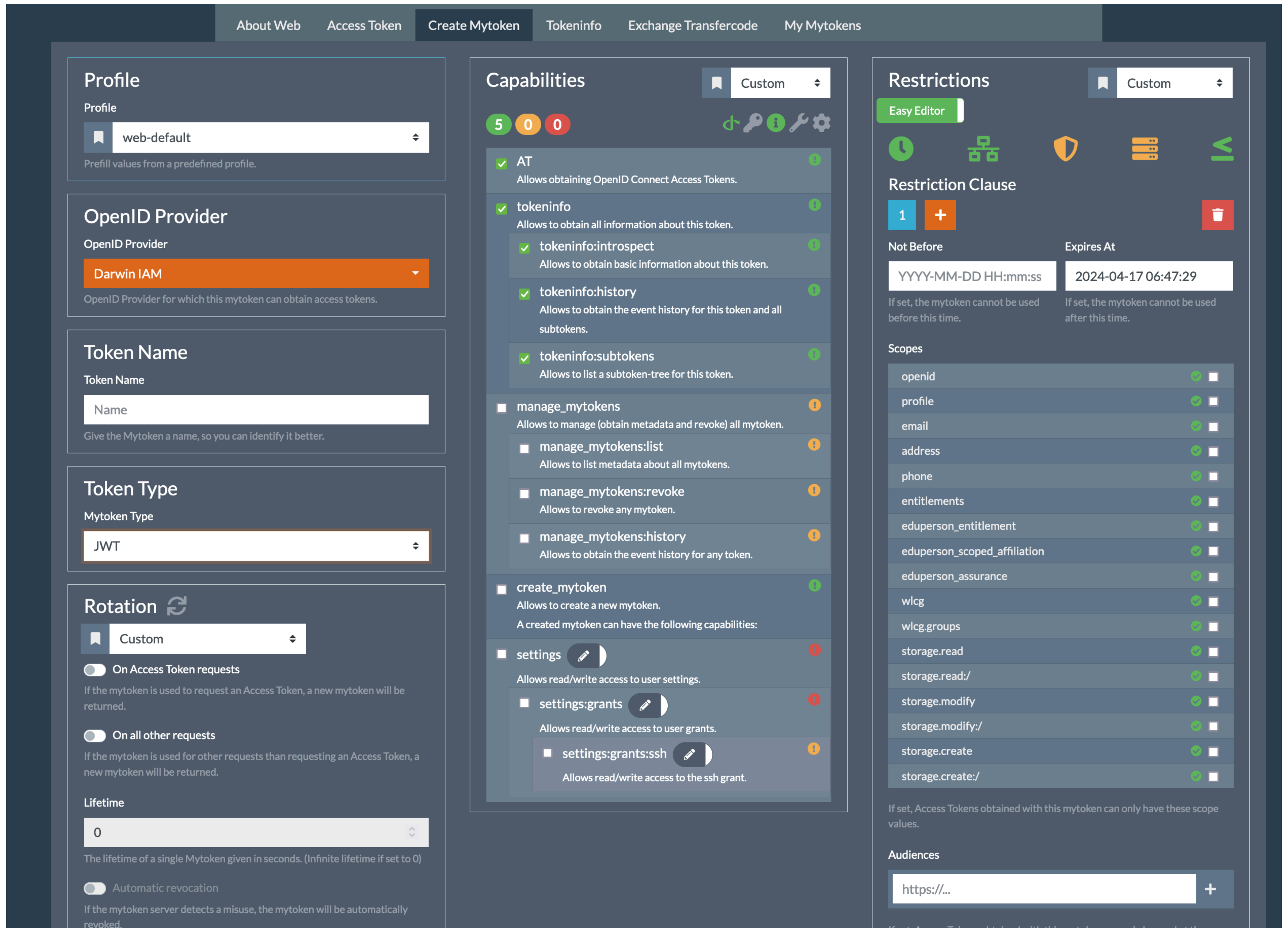


Mytoken token

› https://github.com/oidc-mytoken

- › “Proxy service” between IAM and client
- › ***mytokens*** have additional features compared to OIDC refresh tokens like additional time restrictions, geolocation restrictions…
- › Integrated with ***odic-agent***





Profile

Profile: web-default

Prefill values from a predefined profile.

OpenID Provider

OpenID Provider: Darwin IAM

OpenID Provider for which this mytoken can obtain access tokens.

Token Name

Token Name: Name

Give the Mytoken a name, so you can identify it better.

Token Type

Mytoken Type: JWT

Rotation

Custom

○ On Access Token requests

If the mytoken is used to request an Access Token, a new mytoken will be returned.

○ On all other requests

If the mytoken is used for other requests than requesting an Access Token, a new mytoken will be returned.

Lifetime: 0

The lifetime of a single Mytoken given in seconds. (Infinite lifetime if set to 0)

○ Automatic revocation

If the mytoken server detects a misuse, the mytoken will be automatically revoked.

Capabilities Custom

5 0 0

- ☑ AT — Allows obtaining OpenID Connect Access Tokens.
- ☑ tokeninfo — Allows to obtain all information about this token.
  - ☑ tokeninfo:introspect — Allows to obtain basic information about this token.
  - ☑ tokeninfo:history — Allows to obtain the event history for this token and all subtokens.
  - ☑ tokeninfo:subtokens — Allows to list a subtoken-tree for this token.
- □ manage_mytokens — Allows to manage (obtain metadata and revoke) all mytoken.
  - □ manage_mytokens:list — Allows to list metadata about all mytokens.
  - □ manage_mytokens:revoke — Allows to revoke any mytoken.
  - □ manage_mytokens:history — Allows to obtain the event history for any token.
- □ create_mytoken — Allows to create a new mytoken. A created mytoken can have the following capabilities:
- □ settings — Allows read/write access to user settings.
  - □ settings:grants — Allows read/write access to user grants.
    - □ settings:grants:ssh — Allows read/write access to the ssh grant.

Restrictions Custom

Easy Editor

Restriction Clause

1 +

Not Before: YYYY-MM-DD HH:mm:ss

If set, the mytoken cannot be used before this time.

Expires At: 2024-04-17 06:47:29

If set, the mytoken cannot be used after this time.

Scopes

- □ openid
- □ profile
- □ email
- □ address
- □ phone
- □ entitlements
- □ eduperson_entitlement
- □ eduperson_scoped_affiliation
- □ eduperson_assurance
- □ wlcg
- □ wlcg.groups
- □ storage.read
- □ storage.read:/
- □ storage.modify
- □ storage.modify:/
- □ storage.create
- □ storage.create:/

If set, Access Tokens obtained with this mytoken can only have these scope values.

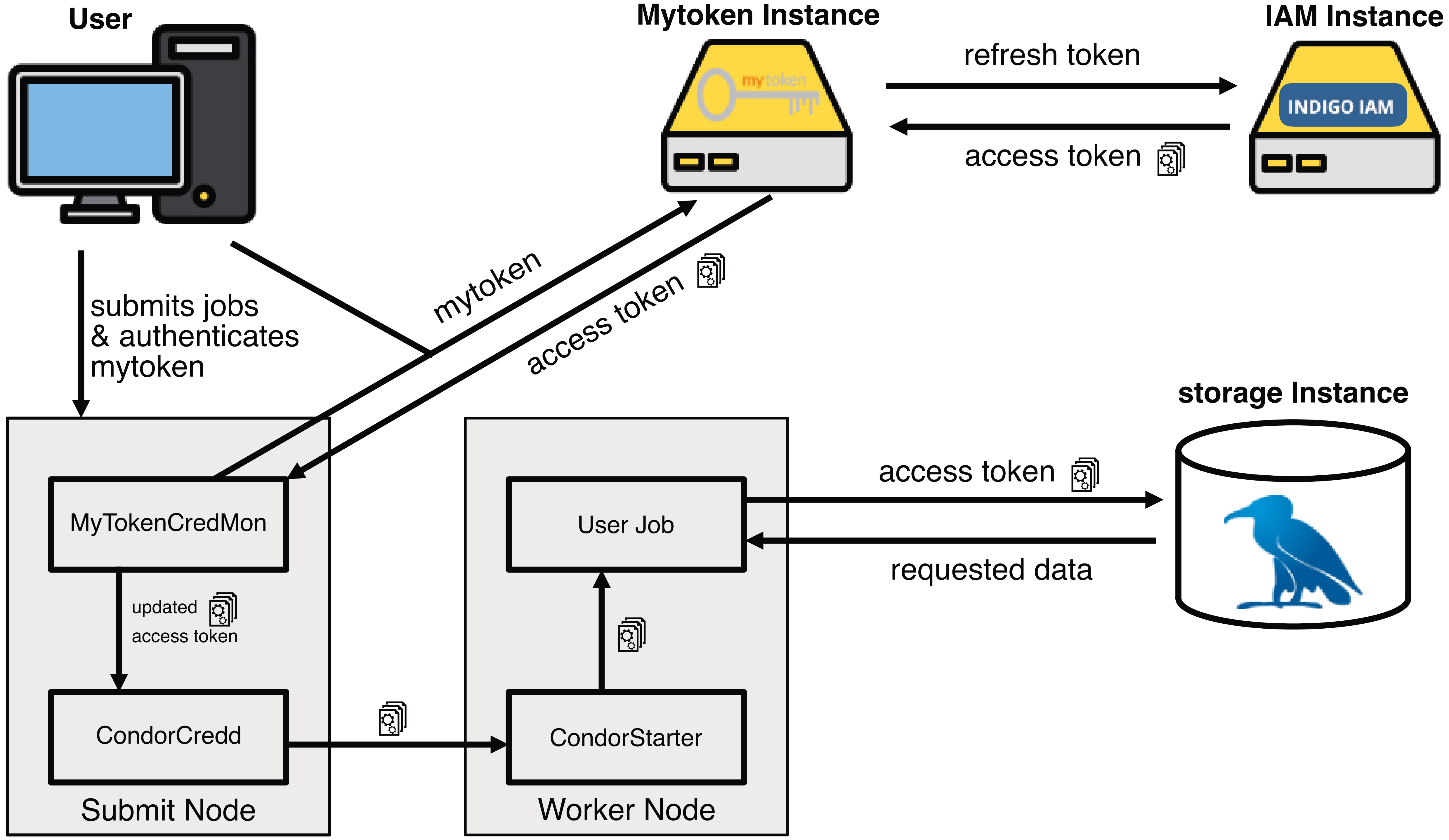
Audiences: https://... +

```
oidc-gen --mytoken-url="https://mytoken.data.kit.edu" darwin
```

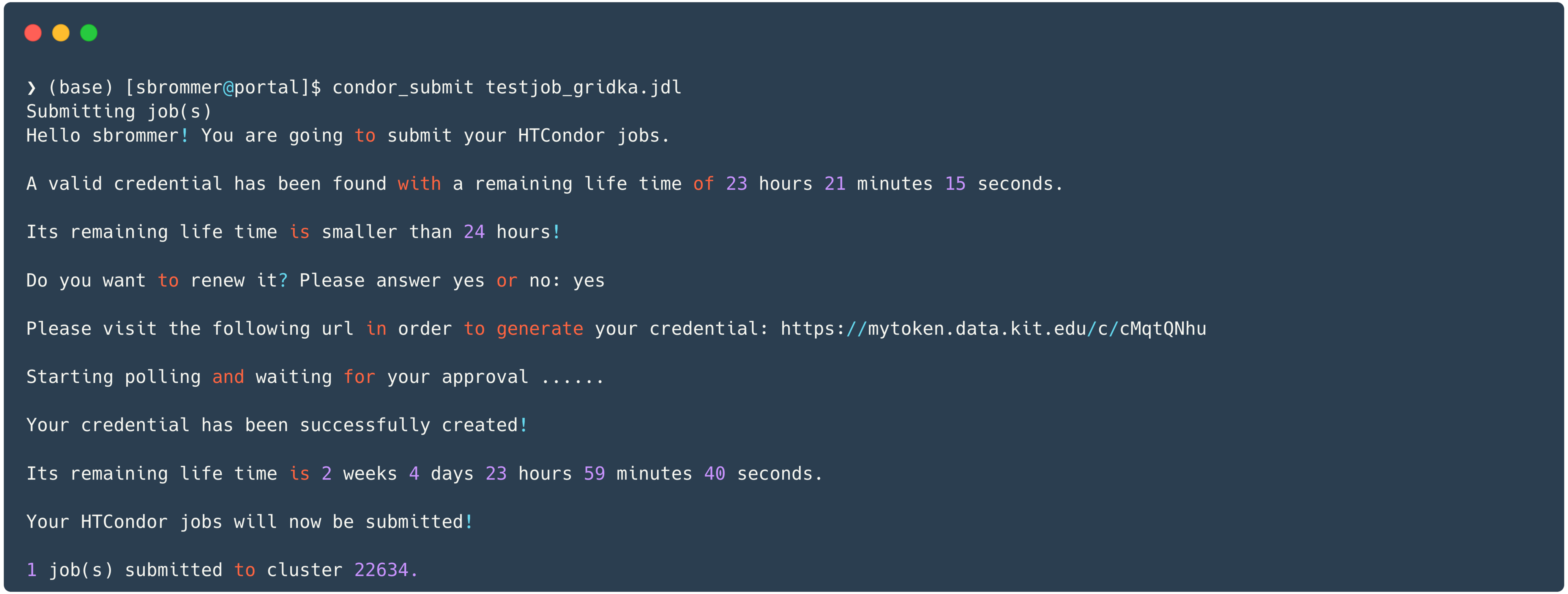
Remote Storage, Jobs and tokens

› https://github.com/oidc-mytoken

- Access tokens are short-lived, **refresh tokens are not meant to leave a registered device**
- Solution for HTCondor jobs - ***mytoken*** integration in HTCondor
- Automatic **renewal via HTCondor mechanisms**, mytoken does not leave the submit Node
- mytoken integration is open PR to HTCondor



Remote Storage with tokens



```
❯ (base) [sbrommer@portal]$ condor_submit testjob_gridka.jdl
Submitting job(s)
Hello sbrommer! You are going to submit your HTCondor jobs.

A valid credential has been found with a remaining life time of 23 hours 21 minutes 15 seconds.

Its remaining life time is smaller than 24 hours!

Do you want to renew it? Please answer yes or no: yes

Please visit the following url in order to generate your credential: https://mytoken.data.kit.edu/c/cMqtQNhu

Starting polling and waiting for your approval ......

Your credential has been successfully created!

Its remaining life time is 2 weeks 4 days 23 hours 59 minutes 40 seconds.

Your HTCondor jobs will now be submitted!

1 job(s) submitted to cluster 22634.
```