OSCARS
Open Science Clusters' Action for Research & Society

Funded Project

MADDEN Project: Multi-RI Rucio with POSIX-like Interfaces for Collaborative Data Access

Principal Investigator: Federica Legger, INFN Torino
Project team members: N. Avdeev (INFN Torino), K. Jansson (UCLouvain), L. Lavezzi (INFN Torino), A. Tanasijczuk (UCLouvain)

Eighth Rucio Community Workshop
Nov 3 – 7, 2025
Square Kilometre Array Observatory Global Headquarter

Funded by the European Union

The MADDEN project

- **MADDEN - Multi-RI (Research Infrastructure) Access and Discovery of Data for Experiment Networking**
  - funded by 1st call OSCARS cascading grants
  - Start date: **February 1st 2025**
  - Duration: **24 months**
  - Funding: **210 K€**
  - Institutes: **INFN Turin, UCLouvain**
  - Personnel: hired **2 postdocs** (in Torino and Louvain)
- Gitlab repo: https://gitlab.et-gw.eu/eib/div1/oscars/madden

[Slides by F. Legger]

Objectives

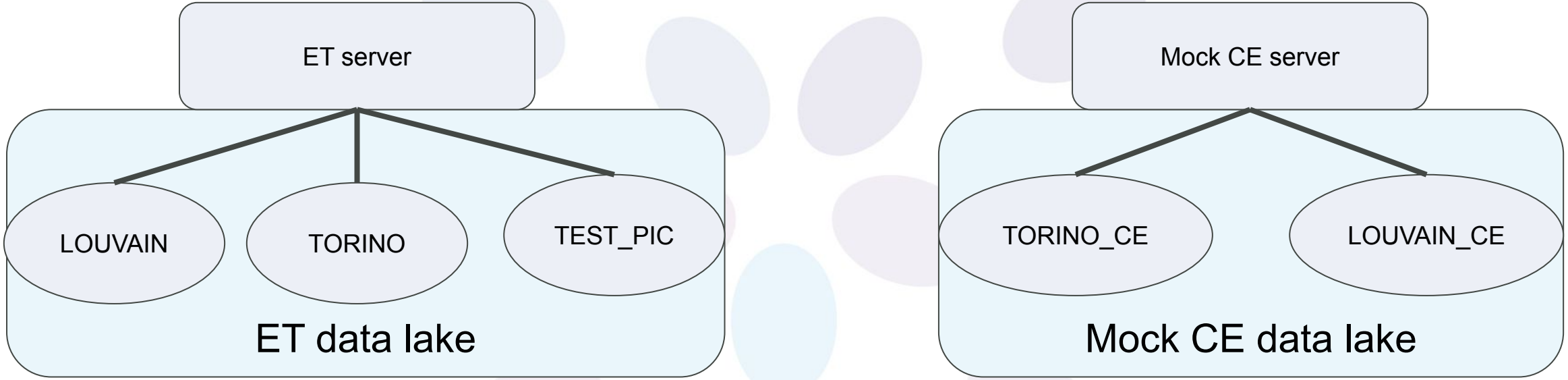
- Showcase **Rucio** for Distributed Data Management for ET(Einstein Telescope) (and CE(Cosmic Explorer)) community
  - similar to the current LVK use case - **see Lia Lavezzi's report**
- Build a multi-RI(multi Research Infrastructure) Rucio Data Lake integrated with:
  - **ET Indigo IAM** (Identity and Access Management)
  - **VRE** (Virtual Research Environment) from partner project **ETAP**
- Provide POSIX view of Rucio catalog
  - Early prototype: **RucioFS**
  - Recent idea: **Rucio-CVMFS plugin**
- Build a mini Rucio Data Lake for **Virgo** to be used as a technology demonstrator for LVK

[Slides by F. Legger]

Multi-RI data management

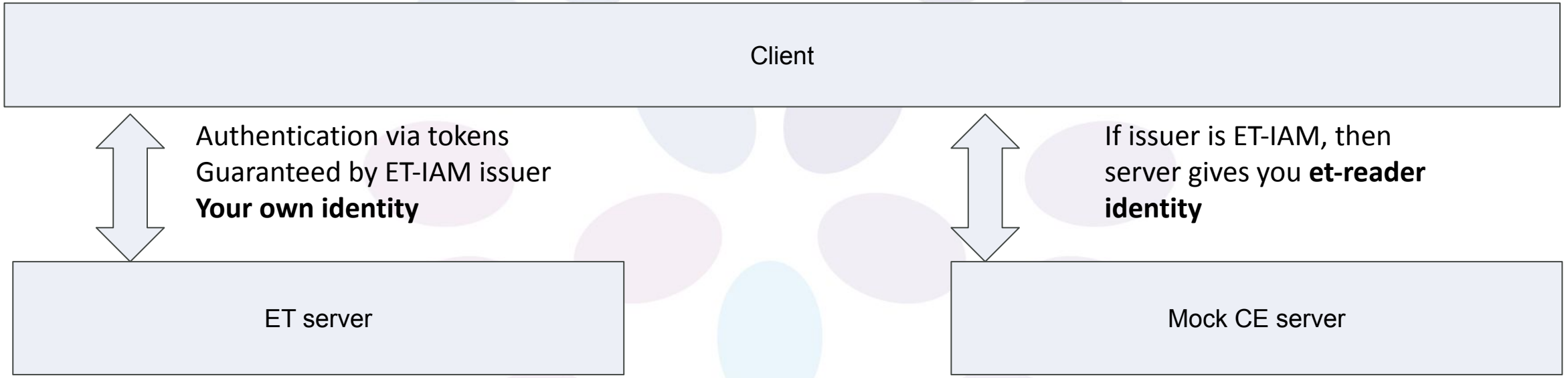
ET and CE Data Lakes

- **Rucio server** installed in INFN Torino
- **Rucio Storage Elements** (RSEs):
  - UCLouvain: OSDF(Open Science Data Federation) origin for MDC1(Mock Data Challenge) data distribution
  - INFN Torino: for tests
  - PIC: for MDC data (20 TB at PIC to hold MDC1 and MDC2 data)
- **Authentication**: connected to ET Indigo IAM



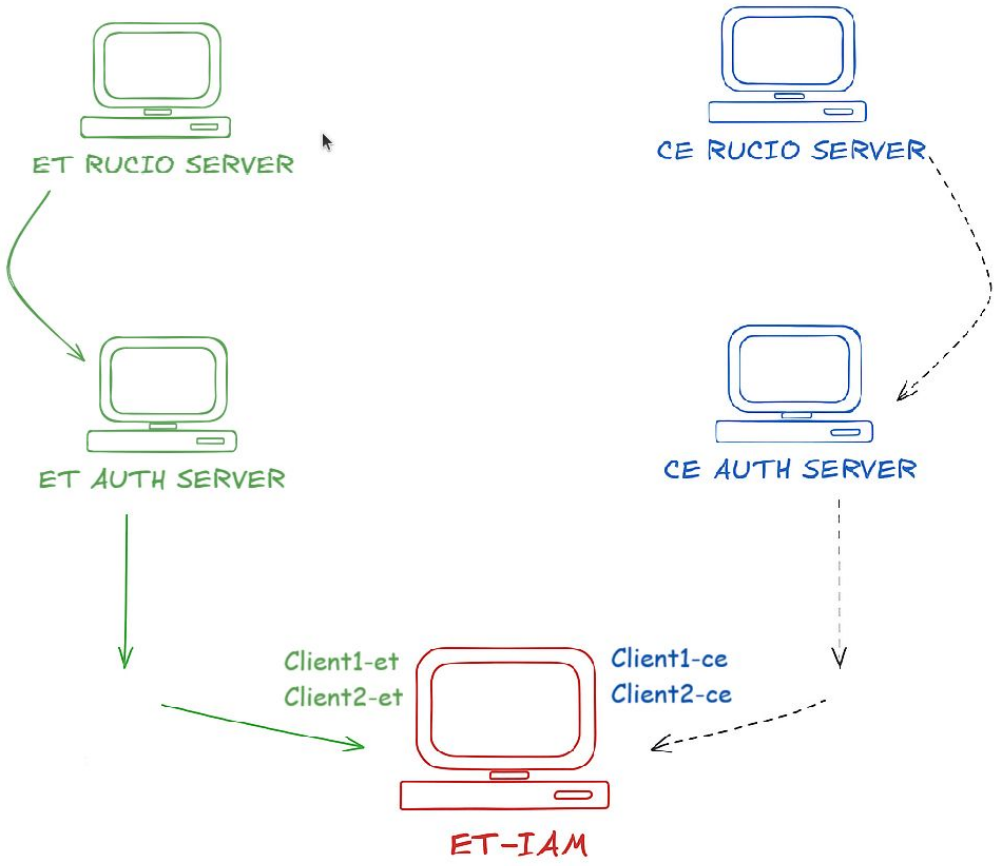
Multi-RI Rucio

- The Multi-RI extension of Rucio enables simultaneous access to multiple Rucio instances.
- Access to many servers:
  - Oidc authentication to many servers
- How does it work?



Multi-RI data management

- **OIDC authentication on ET and CE servers:**



- Currently, both ET and CE clients in ET-IAM obtain tokens with the same oidc_audience = rucio and the same oidc_scope.
- As a result, tokens issued by ET-IAM are accepted by both Rucio servers.

Multi-RI Rucio

- **Multi-RI version of Rucio has two parts:**
  - Access to more than one Rucio server
  - Running the same Rucio commands at more than one Rucio server
- **Modifications of Multi-RI version of Rucio:**
  - Client side:
    - rucio - file inside client
    - More than one configuration file
  - Server side:
    - oidc.py

Multi-RI Rucio

- How does it look like?

```
[root@rucio-client bin]# rucio whoami
Result from ET server.

Please use your internet browser, go to:

    https://et-rucio-server.to.infn.it/auth/oidc_redirect?HZKsZu8cs9rFLm3ZQdIyPaD_polling

and authenticate with your Identity Provider.
In the next 3 minutes, Rucio Client will be polling
the Rucio authentication server for a token.
------------------------------------------
suspended_at : None
account      : avdeev
status       : ACTIVE
account_type : USER
created_at : 2025-04-08T13:01:41
email      : nikita.avdeev@to.infn.it
deleted_at : None
updated_at : 2025-04-08T13:01:41
Result from CE server.
suspended_at : None
account_type : USER
email      : defaultemail@et-gw.eu
created_at : 2025-07-29T12:25:29
account    : etreader
status     : ACTIVE
deleted_at : None
updated_at : 2025-07-29T12:25:29
```

Multi-RI Rucio

- What are the changes to config?
- Two optional values under client:
  - multi_ri_commands
    - Execute if command or subcommand is present in the list
  - multi_ri_servers
    - Used as section headers and user feedback
- One section for each server
  - Only has value config_file
    - Commands get executed using each config file

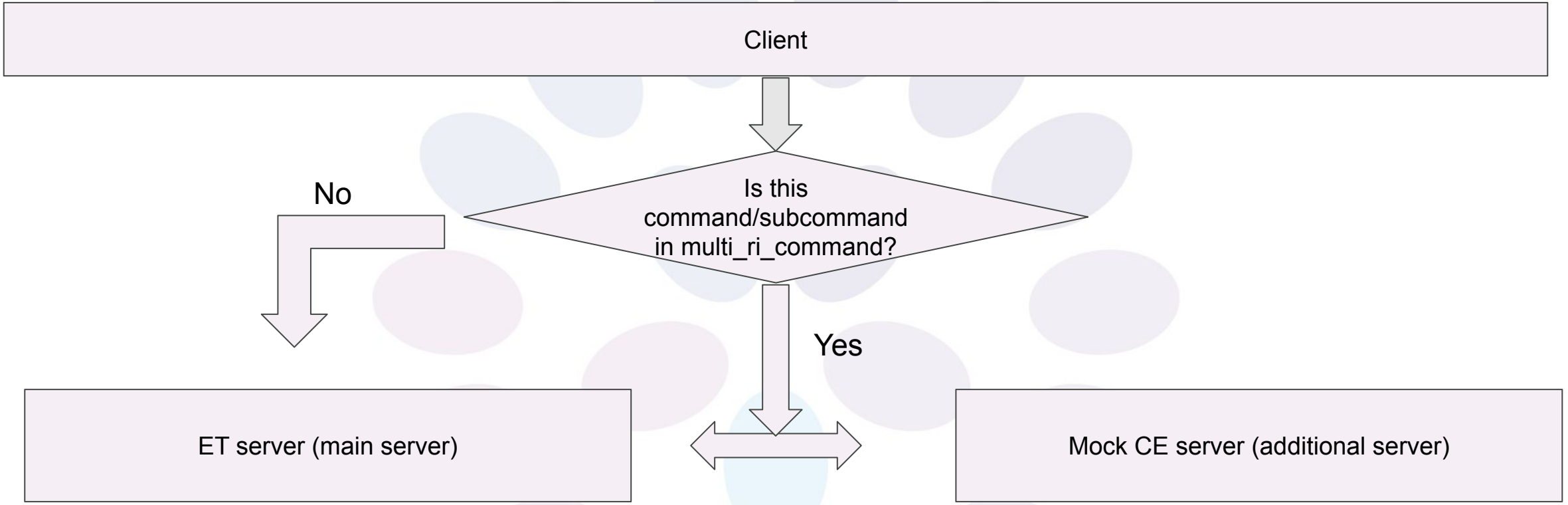
```
jansson@admin-sise-Latitude-7450:~/kod/rucio_2$ tail etc/rucio.cfg
request_retries = 3
protocol_stat_retries = 6
multi_ri_commands = whoami, ping, list
multi_ri_servers = ET server, CE server

[ET server]
config_file = /opt/rucio/etc/rucio.cfg

[CE server]
config_file = /opt/rucio/etc/rucio_CE.cfg
jansson@admin-sise-Latitude-7450:~/kod/rucio_2$
```

Multi-RI Rucio

- Running some commands at many servers
- How does it work?



Multi-RI Rucio (examples from our server using version 32/35)

- How does it look like?

```
[root@rucio-client home]# rucio list-rses
Result from ET server.
LOUVAIN
TORINO
TEST_PIC
Result from CE server.
LOUVAIN_CE
TORINO_CE
```

- How to use command only for one server?

```
[root@rucio-client etc]# rucio \
> --config /opt/rucio/etc/rucio_CE.cfg \
> list-rses
LOUVAIN_CE
TORINO_CE
```

<u>Current status:</u> We are preparing a pull request for the Rucio developers to review, and we hope it will be included in the next release at the end of November.

POSIX-like view Rucio

- **Objective:**
  - Provide a read-only POSIX-like view with minimal Rucio changes
- **Why do we need it?**
  - Many GW users expect a familiar POSIX interface for analysis
  - It is more convenient to find data files
- **How can we solve this?**
  - RucioFS
  - CVMFS plugin for Rucio

Possible solutions

RucioFS

RucioFS is a POSIX-like FUSE-based filesystem.

https://github.com/rucio/fuse-posix

- Maps the Rucio scope/container/dataset/file structure into a hierarchical directory tree.
- Provides users with a familiar POSIX-style interface.
- Caching system is very simple - local cache, unordered_map

Advantages:

- Light local solution

Shortcomings:

- There is no cache refreshing system
- Scalability?

CVMFS plugin

CVMFS provides a mechanism for distributing files globally across computing centers.

- Originally developed and optimized for software distribution.
- Also adopted for data files in the GW community.
- CVMFS has been deprecated by LVK for Data in August 2025

Advantages:

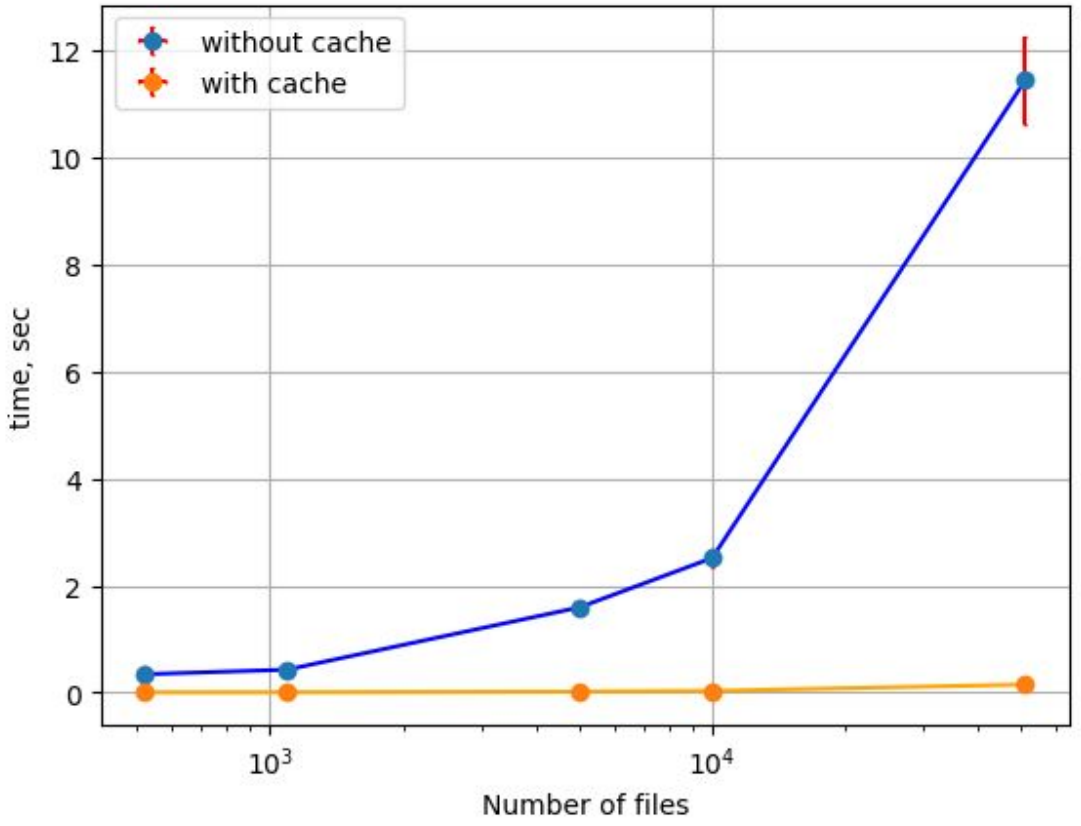
- There is good optimized caching system

Shortcomings:

- It is heavy solution with additional local and server parts

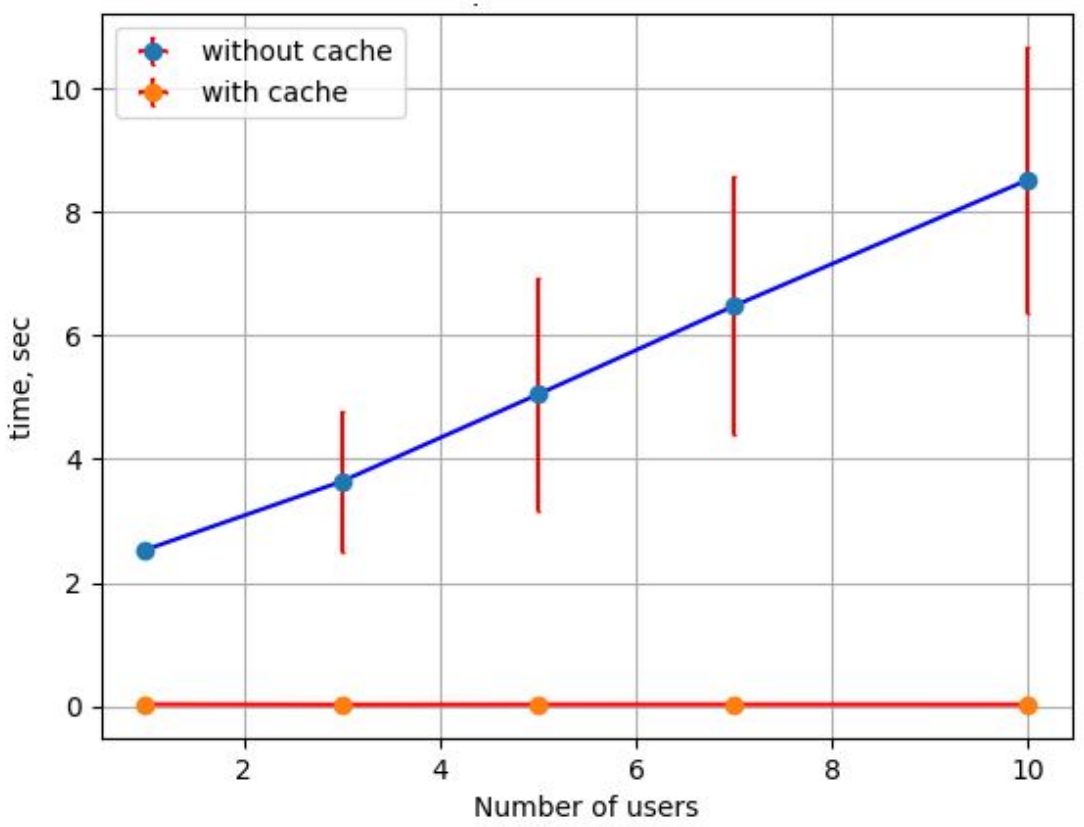
Performance tests for RucioFS (*ls* command time):

- The goal is to estimate the execution time of the **ls** command with and without caching.
- We tested it on directories containing **500**, **1,000**, **5,000**, **10,000**, and **50,000** files (each **4 KB**). Files were generated by a Bash script, and execution time was measured using the **time** command.
- For each case, **100** independent runs were performed.



Performance tests for RucioFS (*ls* command time for multiple users):

- The goal is to estimate the execution time of the **ls** command run simultaneously by multiple users, with and without caching.
- Each user listed **10,000** files, and execution time was measured using the time command.
- Here, "**users**" refer to separate clients on the same machine, so in real conditions with remote users, execution time could be longer.

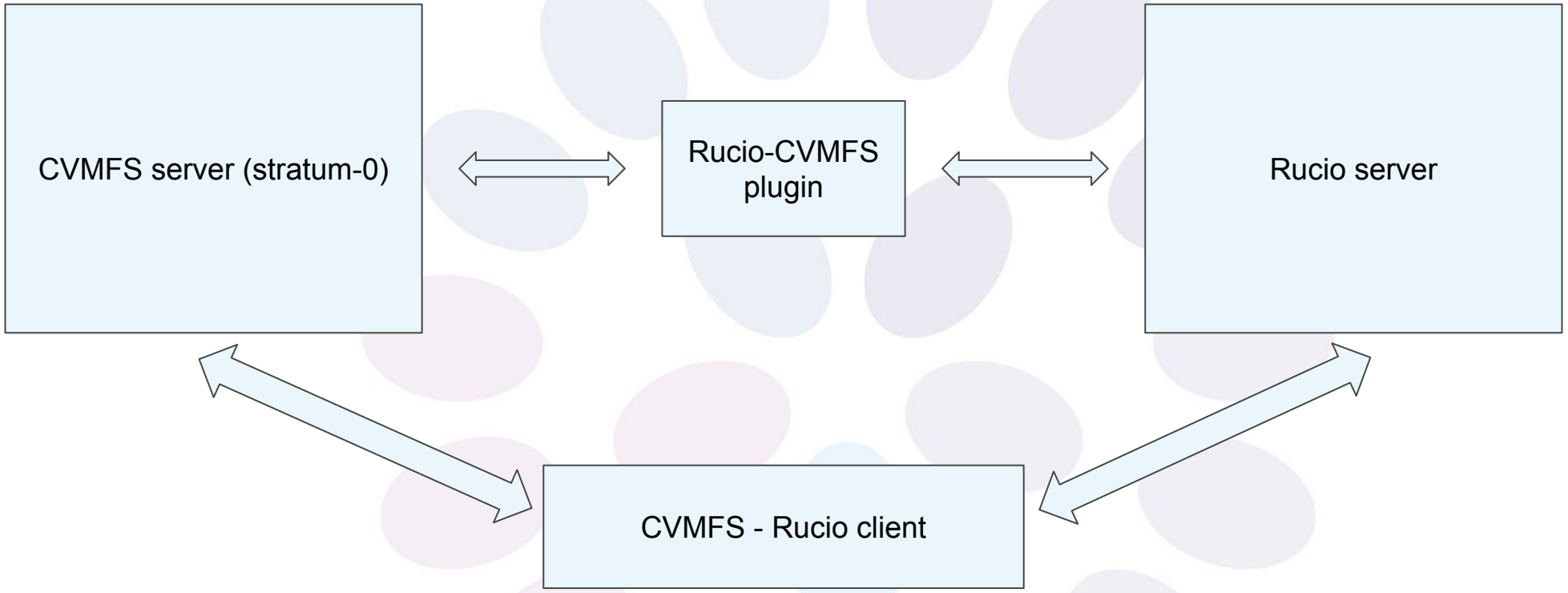


What about *cat* and *cp* commands?

- Since RucioFS is only a preliminary version of the program with Posix-like view, the ***cat*** and ***cp*** commands are not fully implemented in it. But you can successfully run it for small files
- Do we want to implement it in our solution?
- That’s a good question!

What do we think about CVMFS plugin?

- We started discussion with Valentin Volkl from CVMFS team
- First thoughts about how it would be:



Metadata queries

Why do we need metadata?

- In the GW community, metadata — especially GPS time — is essential for data analysis.
- Currently, this information is mostly encoded in file names.
- We expect metadata usage to expand in the future.

What are our goals?

- Define the minimal and recommended metadata fields required by the GW community.
- Enable efficient metadata-based queries across multiple Rucio instances.
- Integrate metadata search capabilities into the POSIX-like view extension for Rucio.

Metadata queries

Standart Rucio:

```
[root@rucio-client etc]# rucio set-metadata --did avdeev:new_test21_ce_8335.txt --key answer --value 42
[root@rucio-client etc]# rucio list-dids avdeev:* --filter "answer=42"
+-------------------------------+--------------+
| SCOPE:NAME                    | [DID TYPE]   |
|-------------------------------+--------------|
| avdeev:new_test21_ce_8335.txt |              |
+-------------------------------+--------------+
```

Multi-RI Rucio:

```
[root@rucio-client bin]# rucio list-dids avdeev:* --filter "answer=42"
Result from ET server.
+------------------------+--------------+
| SCOPE:NAME             | [DID TYPE]   |
|------------------------+--------------|
| avdeev:testteststest.txt |            |
+------------------------+--------------+
Result from CE server.
+-------------------------------+--------------+
| SCOPE:NAME                    | [DID TYPE]   |
|-------------------------------+--------------|
| avdeev:new_test21_ce_8335.txt |              |
+-------------------------------+--------------+
```

Conclusion

To summarize:

- We developed and tested a lightweight Multi-RI extension for Rucio
- Benchmarked RucioFS
- Began design toward a Rucio-CVMFS interface
- Started shaping metadata standards and metadata-aware queries

What do you think?
Any thoughts and suggestions are welcomed!


OSCARS
Thank you