

**Centre de Calcul**  
de l'Institut National de Physique Nucléaire  
et de Physique des Particules

# Integration of iRODS in a distributed storage system through HTTP and Python API

Debaecker Gautier & Pagani Mathia

- ❖ **IN2P3 Computing center**
  - **Brief presentation**
  
- ❖ **Federated IT Service (FITS)**
  - **Context of the project**
  - **Classical iRODS commands**
  - **iRODS at CC-IN2P3**
  
- ❖ **iRODS access interfaces**
  - **Python API**
  - **HTTP API**

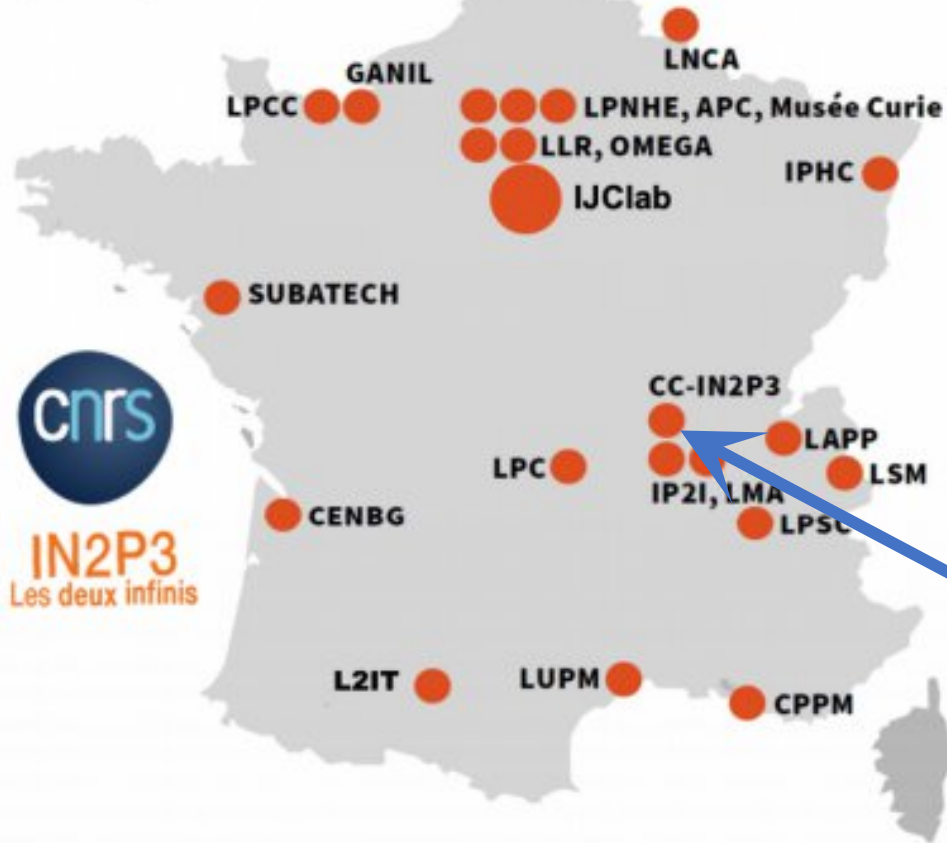


cnrs

# IN2P3 Computing center

# What is CC-IN2P3 ?

Carte de France des unités  
et plateformes nationales  
pilotees par l'IN2P3

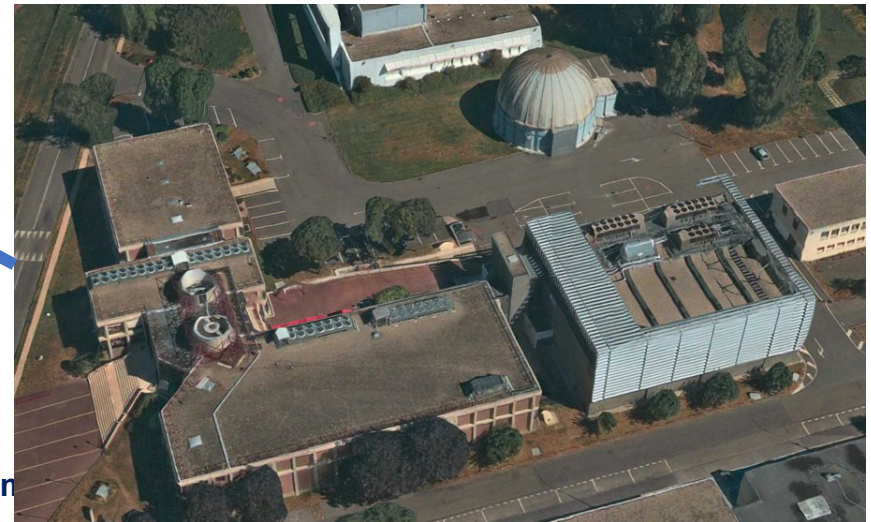


## IN2P3:

- one of the 10 CNRS institutes.
- 19 labs dedicated to research in high energy physics, nuclear physics, astroparticles.

## CC-IN2P3:

- Computing resources provider for projects supported by IN2P3 (national and international collaborations).
- Resources opened to researchers and engineers working into these collaborations
- 2 computing rooms (2 x 850 m<sup>2</sup> or 9150 ft<sup>2</sup>):
  - 2,000 servers.
  - 800 virtual servers.
- 360 Po of storage

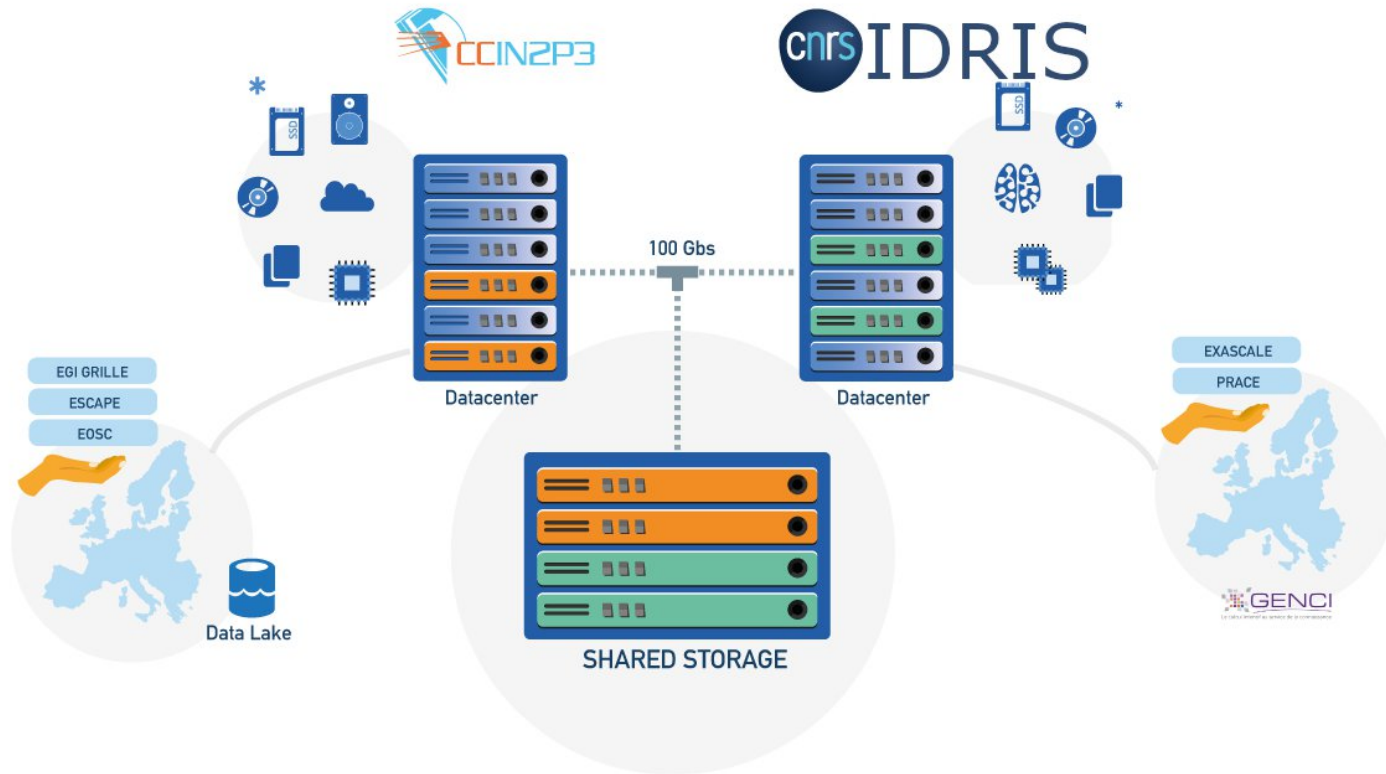


# Who is using CC-IN2P3 ?



# Context of the project

Introduction to Federated IT Service (FITS) project



❖ Pooling computing and storage resources

➤ Data access must be simple and transparent for the end user

➤ Choice of iRODS because already well implemented in CC



<https://www.fits.cnrs.fr/>

"Ce travail a bénéficié d'une aide de l'Etat gérée par l'Agence Nationale de la Recherche au titre du Programme d'Investissements d'Avenir portant la référence ANR-21-ESRE-0009

- ❖ 27 zones.
- ❖ 110 groups.
- ❖ 1236 users:
  - Maximum of 1M connections per day.
  - Maximum of 8M connections per month.
- ❖ 450 millions of files.
- ❖ 34 PB:
  - Disks: 5 PiB.
  - Tape: 29.2 PiB.
  - Up to +100 TiB daily growing rate.



## ❖ Needs

- **Durable and robust**
- **Can be implemented in pipelines/scripts**
- **Different way to connect (IAM/Token, User/Password)**
- **Accessible to all types of users and platforms (beginners to advanced)**
- **Must adapt to users' needs (with installation, web access etc)**

## ❖ Usages

- **“Classic” icommands**
- **A python API-based graphical interface**
- **A graphical/web interface utilizing HTTP API**

## ✓ **Avantages**

- ✓ Well implemented
- ✓ Robust
- ✓ Easily implemented in scripts/pipeline
- ✓ Official support

## ⚠ **Disavantages**

- ⚠ No Identity and Access Manager (IAM) authentication
- ⚠ No more official Windows support
- ⚠ Can be complicated for beginners
- ⚠ No official graphical interface

# **iRODS access interfaces**

## **Python API-based interfaces**

## ❖ Why ?

- Wrap the main iRODS i-commands
- Facilitate/automate the use of iRODS for new users
- Add the auto-completion for some iRODS i-commands
- Bring iRODS to Windows users

[https://github.com/sigau/easy\\_irods\\_commands](https://github.com/sigau/easy_irods_commands)

## First thought → wrap inside a python CLI

- Use the icommands
- Keep some information locally (tree structure, metadata use) for autocompletion

### ✓ Advantages

- ✓ Power and speed of icommands
- ✓ Easy for beginner that use CLI
- ✓ Add simple functionalities (e.g idush)
- ✓ Self installing
- ✓ Library build (can be re-uses in other scripts)

### ⚠ Disadvantages

- ⚠ Need the icommands (no Windows)
- ⚠ No graphical interface
- ⚠ No IAM Identification

```
-----| ^ /-----| |-----| /-----| \ / | |-----| \
|---| / ^ / | (-----| | | | | | | | | | | | | | | |
|---| / ^ / \-----| | | | | | | | | | | | | | | |
|-----| / / \ \-----| | | | | | | | | | | | | | | |
|-----| / / \ \-----| | | | | | | | | | | | | | | |

Possible COMMANDS :

add_meta      : add_meta or add_meta [irods path]
                if you don't give an irods path you'll be asked an option ([f] fo

help          : print this help and leave
idush         : equivalent to du -sh for an irods folder

imkdir        : imkdir -p reinforce by autocompletion

irm           : irm [option]
                option are [-f] for a file and [-C] for a folder
                allow to irm one or multiple (if * used) folder/file in irods. Yo

pull          : pull [option] [local path]
                irsync/iget folder/file from irods to local with autocompletion
                For a file add option -f
                For a folder add option -C
                path can be full path or '.' for current folder
                if no path given, a list of all the folder from root will be pro

push          : irsync/iput folder/file (given by a path) from local to irods w

rm_meta       : rm_meta or rm_meta [irods path]
                if you don't give an irods path you'll be asked an option ([f] f

search_by_meta : search_by_meta [option] or search_by_meta
                option are [-f] for a file, [-C] for a folder and [-u] for a user

search_name    : search_name [option]
                option are [-f] for a file and [-C] for a folder
                search for a file or a folder in irods

show_meta     : show_meta [option] or show_meta
                option are [-f] for a file and [-C] for a folder

synchro       : synchro [local path to folder] [optional:irods path]
                synchronise the contain of a local folder with irods [in irods pa
                the folder will be synchronised on /zone/home/user/
                can be fully automated with the help of when-changed (https://git
```

## Second thought → add a graphical interface

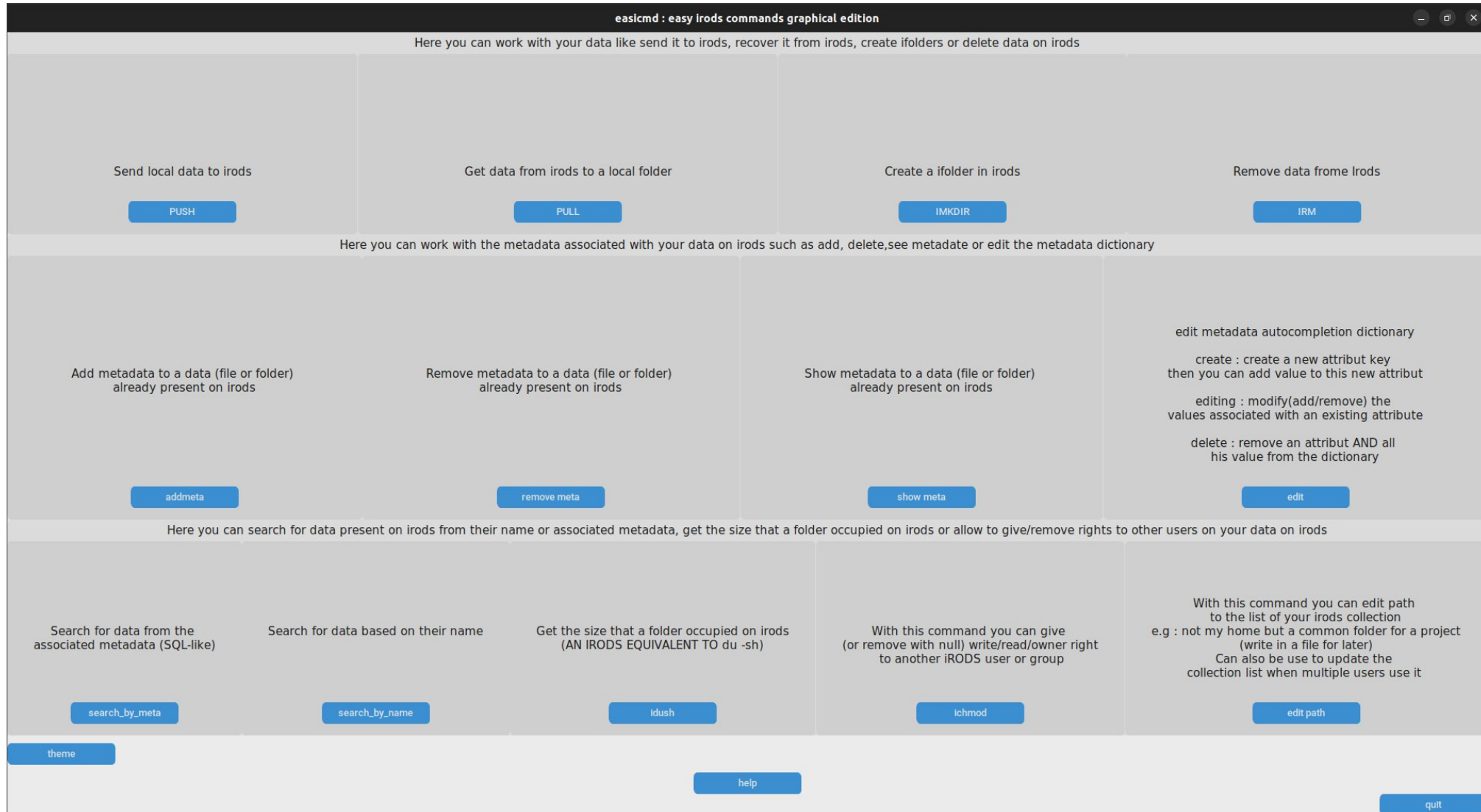
### ✓ **Avantages**

- ✓ Very beginner friendly
- ✓ Use of an explorer for data
- ✓ Use the function of the previous script
- ✓ Based on tkinter (no installation need)

### ⚠ **Disavantages**

- ⚠ Still need the icommands (no Windows)
- ⚠ Still no IAM Identification
- ⚠ Cannot be implemented in pipeline

# A Python API-based graphical interface



The screenshot shows a web-based graphical interface titled "easiCmd : easy irods commands graphical edition". The interface is organized into several sections:

- Top Section:** "Here you can work with your data like send it to irods, recover it from irods, create ifolders or delete data on irods". It contains four buttons: "PUSH" (Send local data to irods), "PULL" (Get data from irods to a local folder), "IMKDIR" (Create a ifolder in irods), and "IRM" (Remove data from irods).
- Middle Section:** "Here you can work with the metadata associated with your data on irods such as add, delete, see metadata or edit the metadata dictionary". It contains four buttons: "addmeta" (Add metadata to a data), "remove meta" (Remove metadata to a data), "show meta" (Show metadata to a data), and "edit" (edit metadata autocompletion dictionary). The "edit" button is accompanied by instructions: "create : create a new attribut key then you can add value to this new attribut", "editing : modify(add/remove) the values associated with an existing attribute", and "delete : remove an attribut AND all his value from the dictionary".
- Bottom Section:** "Here you can search for data present on irods from their name or associated metadata, get the size that a folder occupied on irods or allow to give/remove rights to other users on your data on irods". It contains five buttons: "search\_by\_meta" (Search for data from the associated metadata), "search\_by\_name" (Search for data based on their name), "ldush" (Get the size that a folder occupied on irods), "lchmod" (With this command you can give (or remove with null) write/read/owner right to another iRODS user or group), and "edit path" (With this command you can edit path to the list of your irods collection e.g : not my home but a common folder for a project (write in a file for later) Can also be use to update the collection list when multiple users use it).
- Footer:** A "theme" button on the left, a "help" button in the center, and a "quit" button on the right.

## Third thought → use the Python iRODS API

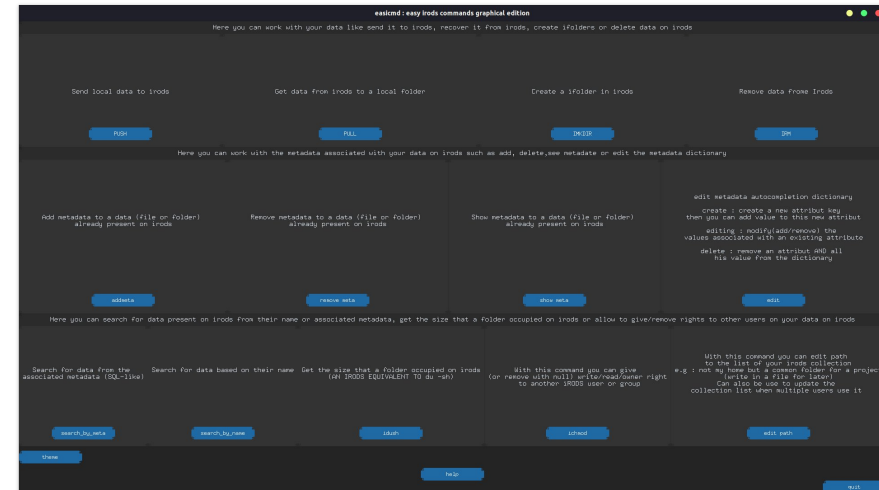


### ✓ Benefits

- ✓ No more icommands need
- ✓ Windows friendly
- ✓ Mostly as fast as icommands

### ⚠ Disadvantages

- ⚠ Still no IAM Identification



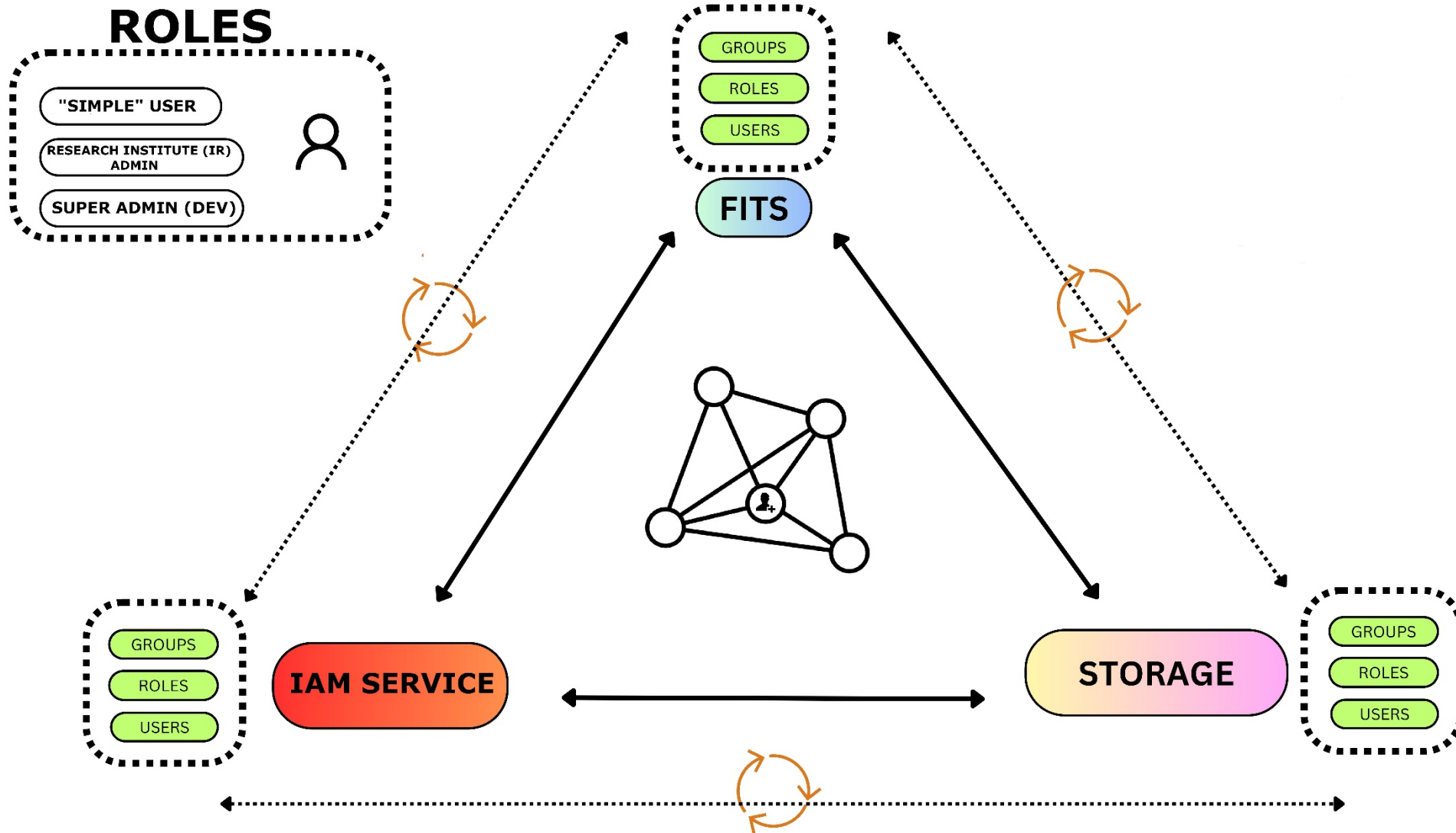
```
(base) gdebaecker@phil-deb:~/irul$ ./easicmp.py idush
ifolder (empty = /zone/home/user) : /lbbeZone/home/gdebaecker/irods_test/test_C-test/raw_test/raw_test3
/
lbbeZone/home/gdebaecker/.hidden_folder
/
lbbeZone/home/gdebaecker/irods_test
/
lbbeZone/home/gdebaecker/irods_test/raw_data
/
lbbeZone/home/gdebaecker/irods_test/raw_data/fast5
/
lbbeZone/home/gdebaecker/irods_test/test_C-test
/
lbbeZone/home/gdebaecker/irods_test/test_C-test/raw_test
/
lbbeZone/home/gdebaecker/irods_test/test_C-test/raw_test/raw_test2
/
lbbeZone/home/gdebaecker/irods_test/test_C-test/raw_test/raw_test3
/
lbbeZone/home/gdebaecker/irods_test/test_C-test/test_mkdir
```



# **iRODS access interfaces**

**A graphical/web interface using HTTP API**

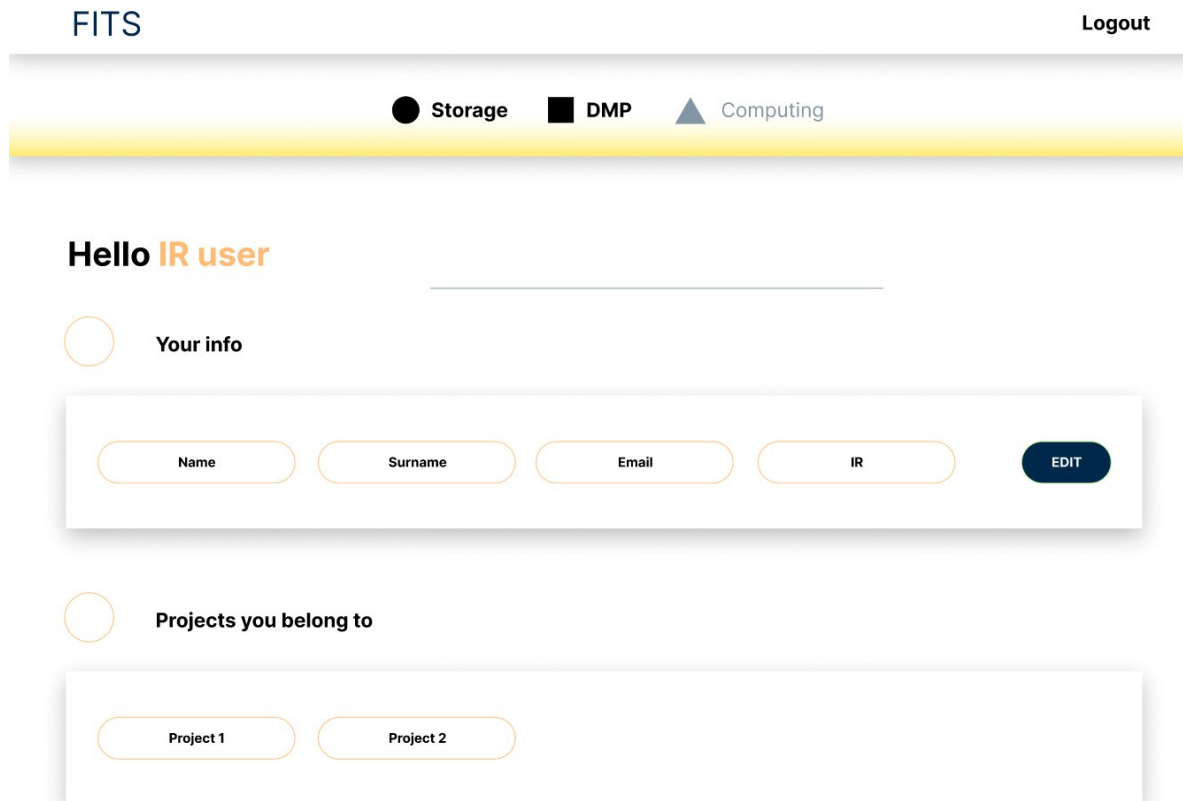
# Triad : Portal <-> IAM <-> iRODS



# A graphical/web interface using HTTP API

## ❖ User Dashboard:

- Different information
- Access/explore their project collection



FITS Logout

● Storage ■ DMP ▲ Computing

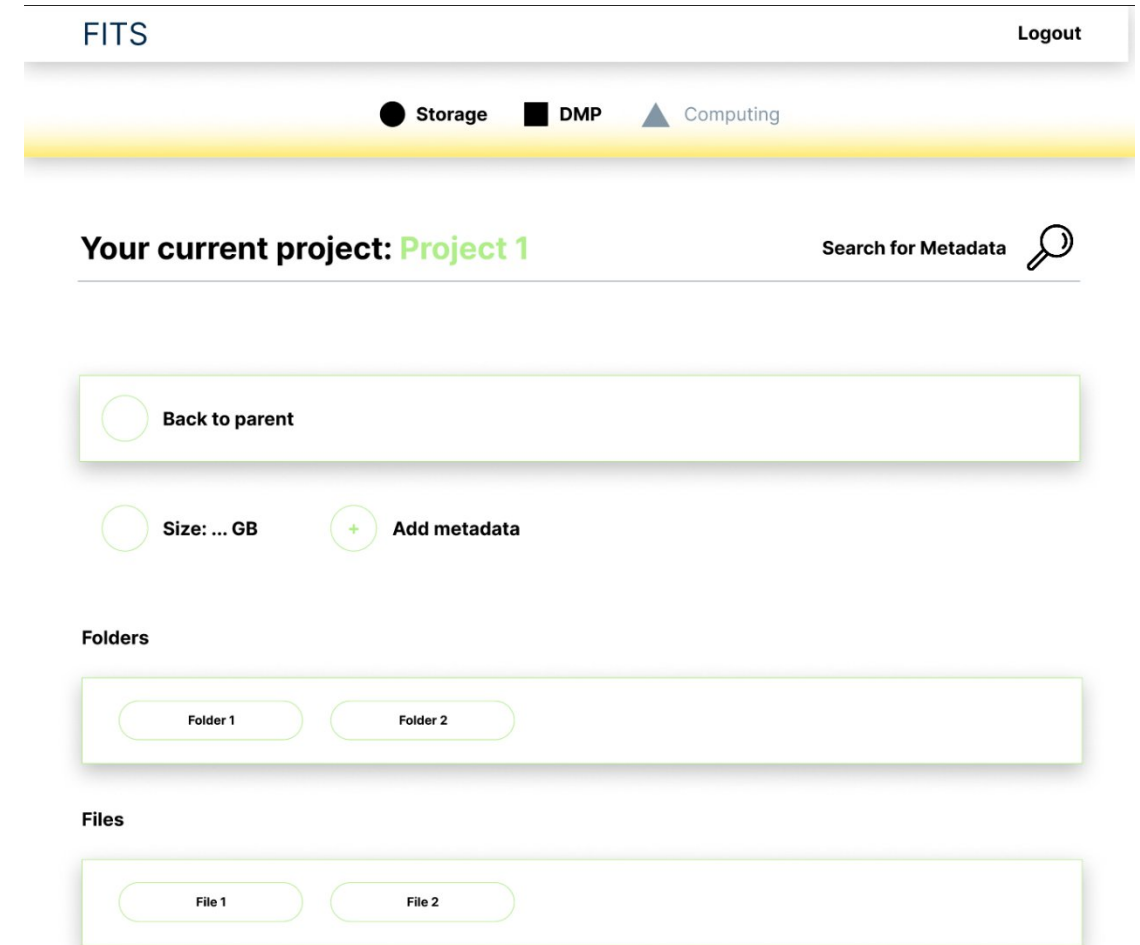
**Hello IR user**

○ Your info

Name Surname Email IR EDIT

○ Projects you belong to

Project 1 Project 2



FITS Logout

● Storage ■ DMP ▲ Computing

**Your current project: Project 1** Search for Metadata 🔍

○ Back to parent

○ Size: ... GB + Add metadata

**Folders**

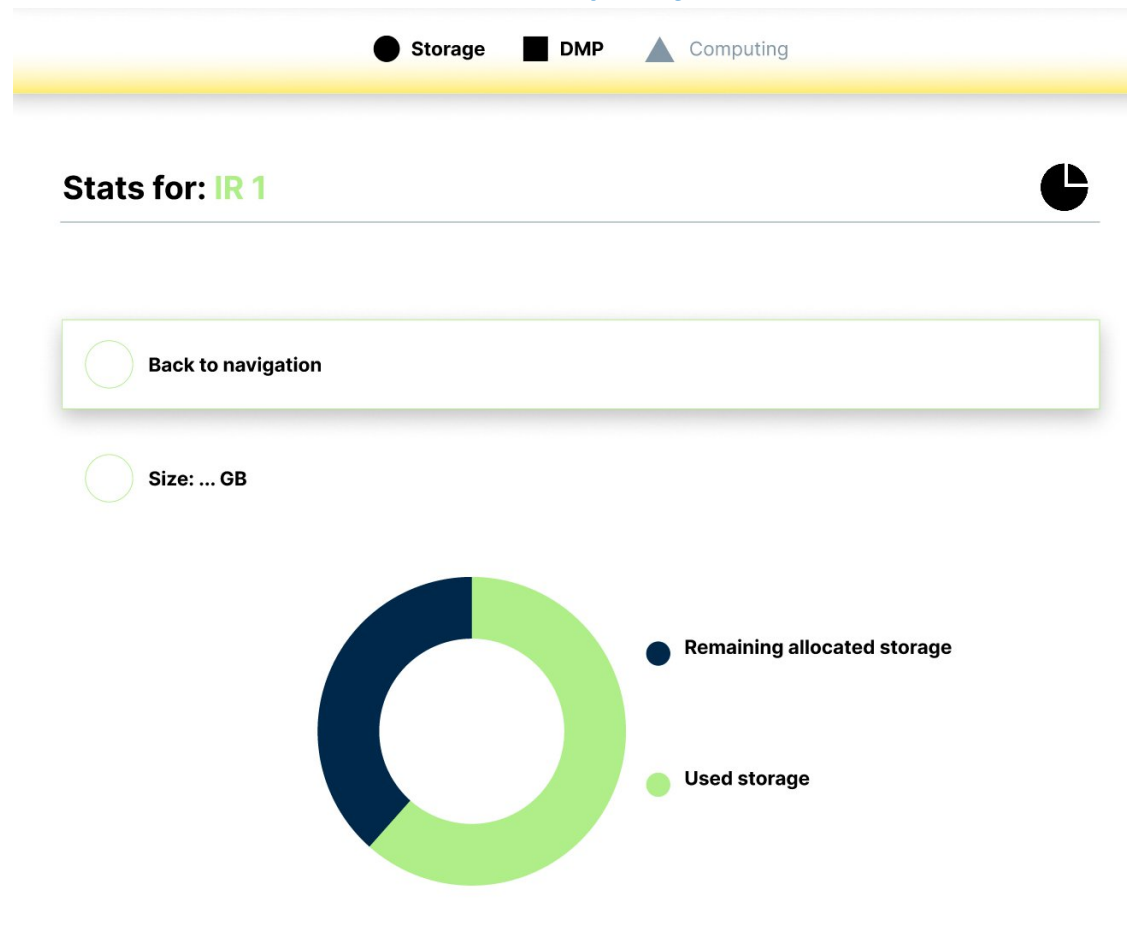
Folder 1 Folder 2

**Files**

File 1 File 2

## ❖ User Dashboard:

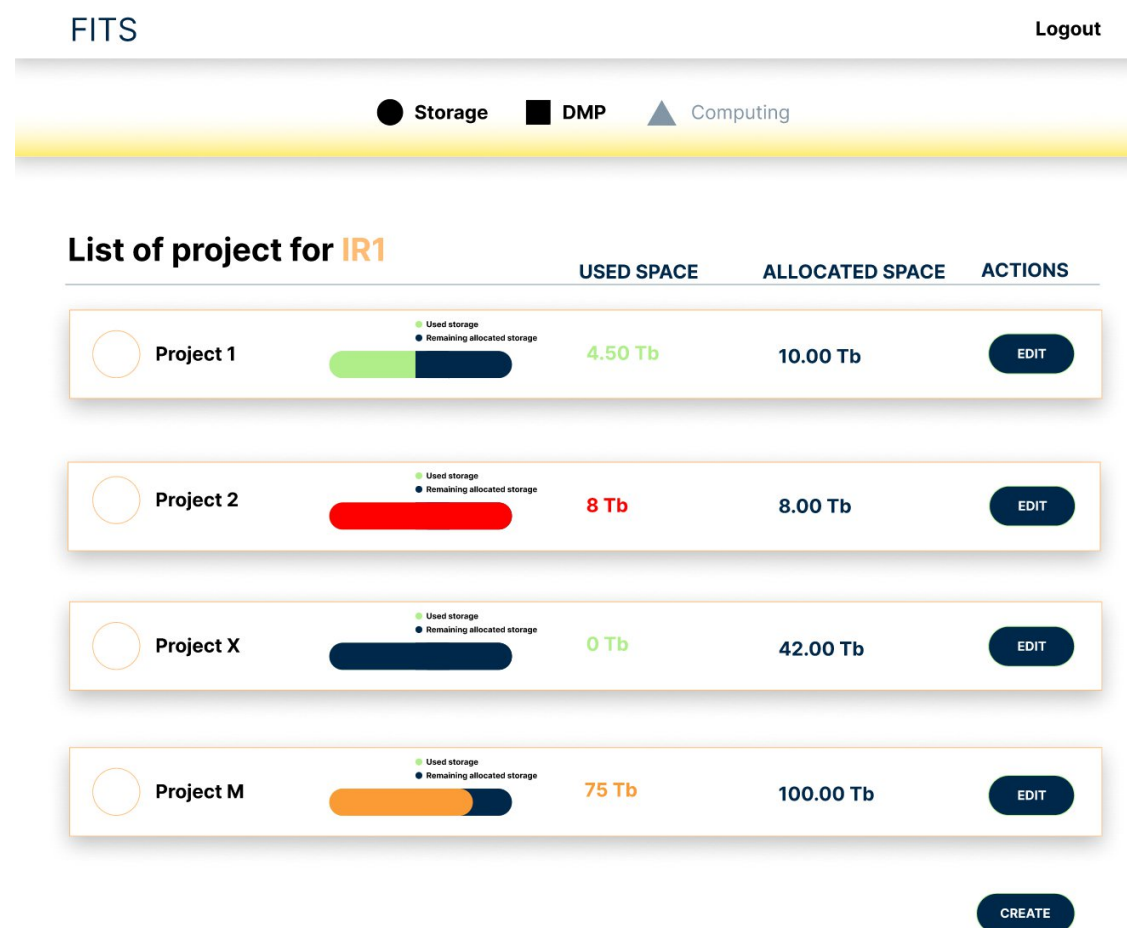
- Get statistics information about their projects



# A graphical/web interface using HTTP API

## ❖ Research Infrastructure Admin Dashboard:

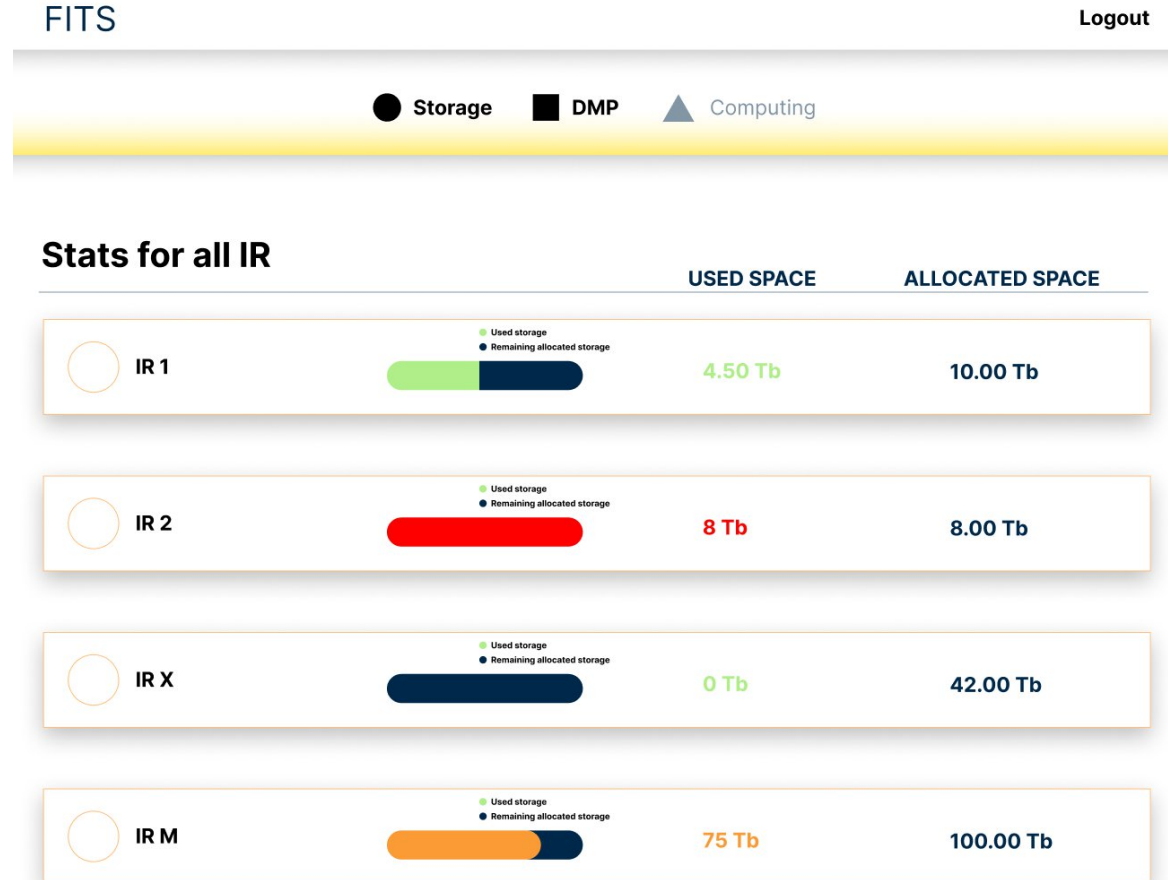
- Get statistics information about all projects of their team



# A graphical/web interface using HTTP API

## ❖ Super Admin Dashboard :

- Get statistics information about all projects



- ❖ **Authentication through keycloak or Indigo IAM**
- ❖ **Triad portal <-> IAM <-> iRODS**
- ❖ **Integration in a Symphony web portal allows :**
  - Easy collection's tree consultation
  - Get stats at a glance
  - Add/consult metadata
  - Accessible from anywhere + 0 installation
- ❖ **Problems encountered**
  - Problems until 0.3 and "OAuth Protected Resource" (user+password before)
  - What name to mapped with Indigo IAM (personalized with keycloak)
  - Not optimal for sending and downloading mass of data

- ❖ **Federate IT Service project**
  - ❖ **iRODS well implemented at CC**
  
- ❖ **Different usage :**
  - ❖ **Classical iRODS icommands**
  - ❖ **Python API :**
    - ❖ **Implementable**
    - ❖ **Graphical interface**
    - ❖ **Windows friendly**
  - ❖ **HTTP API :**
    - ❖ **Authentication through IAM**



- ❖ **Propose token authentication**
  - For icommands
  - For python API
- ❖ **Offer turnkey icommands pipelines for new user**
- ❖ **Enriched python script functionalities**
- ❖ **Add an adapted way to send data from the web**
- ❖ **Add functionalities asked/needed by the user**

# Thank you for your time !