The PaNOSC Data Transfer pilot

12 January 2021 - PaN ESCAPE Data Management Workshop

Authors: Marco De Simone (CERIC-ERIC) on behalf of WP6 Team

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

The Pilot

“A user wants to analyse datasets produced during one of the experiments in the EGI infrastructure using Jupyter notebooks service”

We choose to:

- Transfer data via EGI DataHub based on Onedata from an NFS storage located at Ceric-Eric/Elettra RI to EGI Infrastructure Data Analysis service
- Archive the results back to the originated RI.
- Authenticate users using the EOSC ready PaN community AAI (UmbrellaId)

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 823852

Services, technical solutions & partners

For this pilot the following services & solutions have been used:

- EGI Notebooks service for supporting big data analytics.
- EGI DataHub/Onedata for federating data sets from the PaNOSC RIs.
- UmbrellaID as community AAI proxy for the PaNOSC community.

Partners involved for the pilot:

- **EGI, CERIC-ERIC, ILL and CESNET-MCC.**

Technical support (unfunded):

- **Cyfronet** (**EGI DataHub/Onedata**).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

EGI DataHub/Onedata: an overview

EGI DataHub is built on top of the Onedata SW stack. Onedata is composed by the following main components:

- **Spaces** - distributed virtual volumes, where users can organize their data.
- **Providers** - entities who support user spaces with actual storage resources exposed via *Oneprovider* services.
- **Zones** - federations of providers, which enable creation of closed or interconnected communities, managed by *Onezone* services.
- **LUMA** - A mechanism to translate the community userid that is used to perform the analyses with the userid local to the RIs for read and write operations on files.

Available interfaces:

- User web interface, User command-line interface, python libraries, REST APIs and CDMI.

Fundings:

- **EGI DataHub will be officially supported and funded by EGI (30%), Cyfronet (30%) and by the EGI-ACE project (40%).**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Onedata: PROs

- Open Source (Apache License 2.0).
- Unified data access across globally distributed environments.
- Macaroon bearer tokens for sharing data easily.
- Different RIs storage infrastructures supported including: NFS, CEPH, POSIX, Swift, S3.
- Different interfaces: GUI, CLI, REST APIs and CDMI.
- OnedataFS python library allowing Jupyter Notebook direct access onto Onedata spaces.
- Oneclient for mounting via fuse remote spaces in your local system tree.
- Support for UmbrellaID AAI, OpenID Connect, SAML.
- LUMA in line with current authorization model.
- Supported and funded by EGI (30%), Cyfronet (30%) and by the EGI-ACE project (40%).

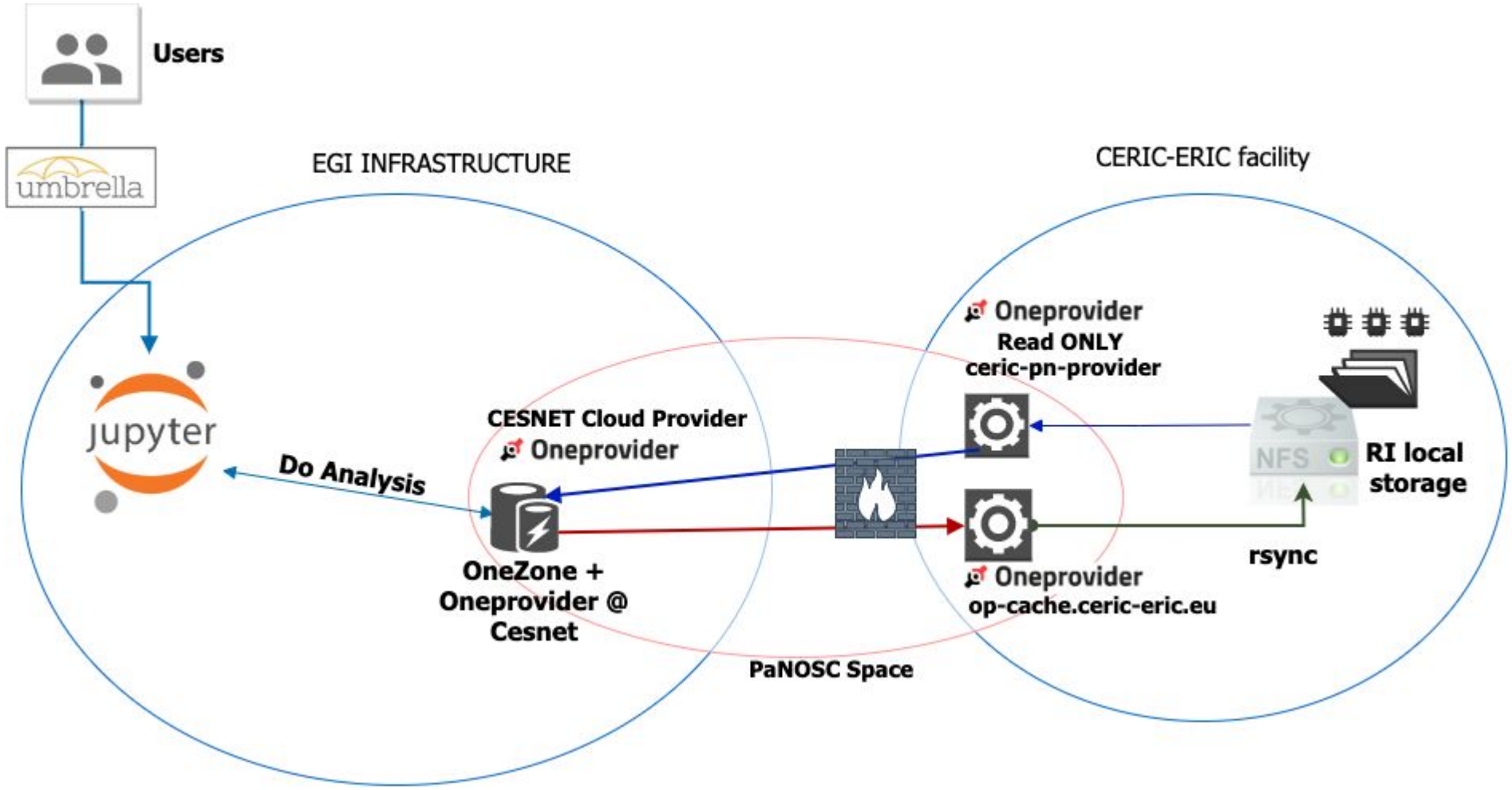
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Onedata: CONs

- Root install on storage might be an issue in some RIs
- Minor issues with not updated documentation
- Own transfer protocol (RTransfer), not interoperable with other solutions
- LUMA mapping so far has to be done manually one by one for users & for each provider
- One dedicated space for each RI exporting storage via oneprovider
- Support beyond EGI-ACE (40% of current funding)

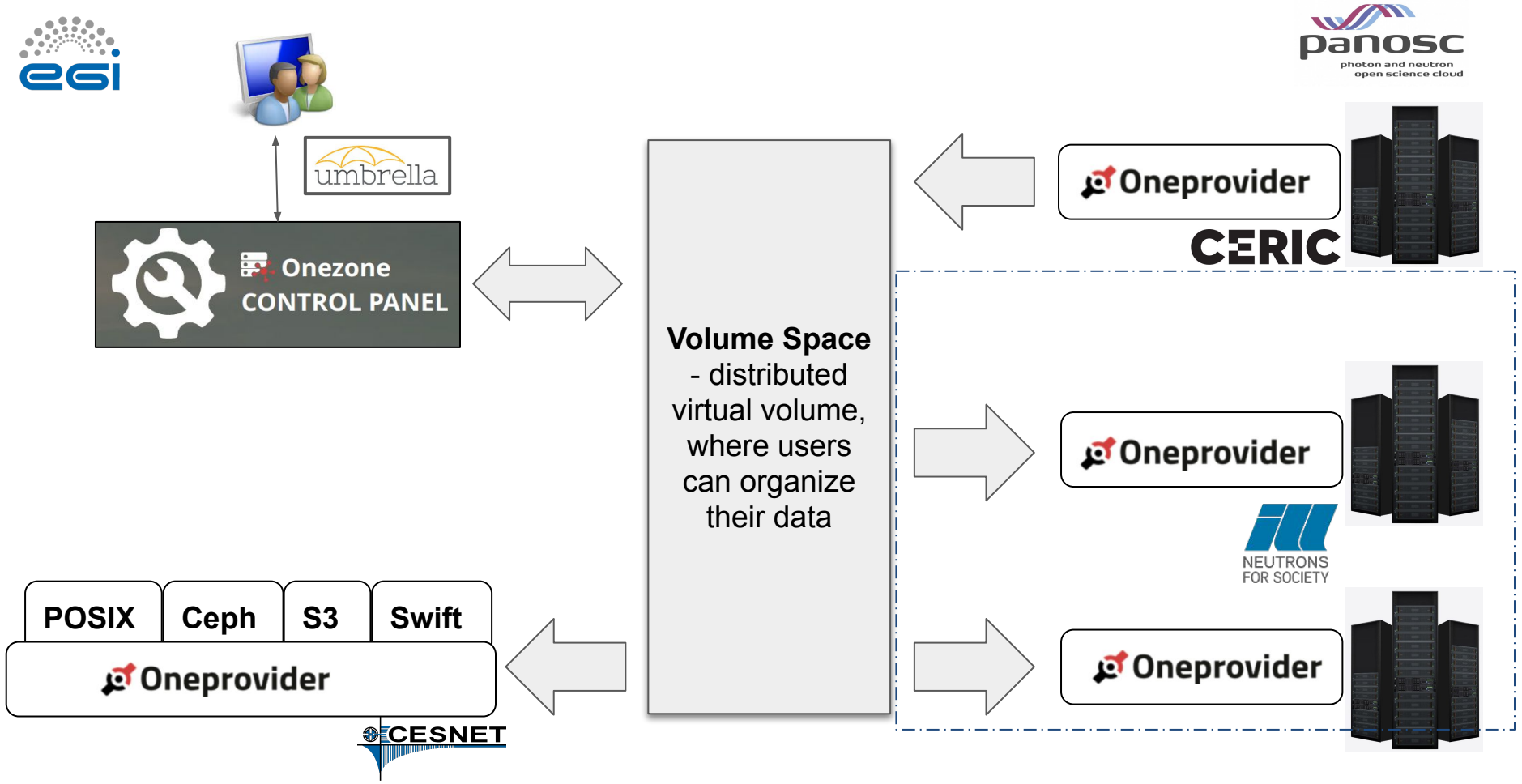
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

The pilot architecture @ CERIC-ERIC



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

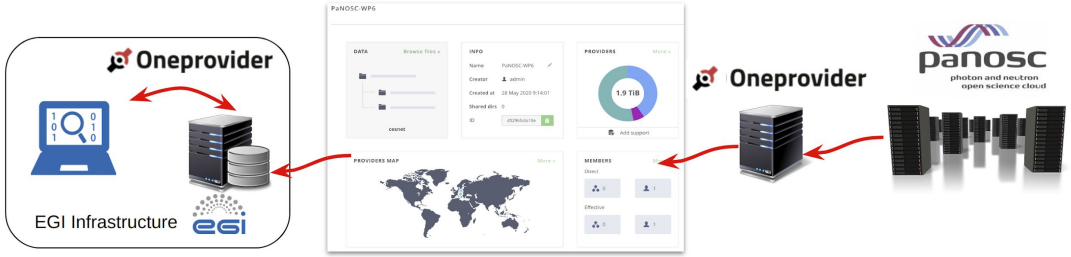
The PaNOSC Space for the pilot



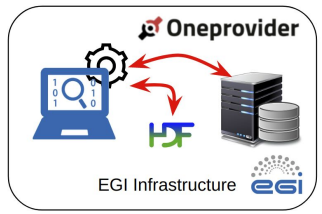
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

The Pilot in action

1. Reading data in the EGI cloud

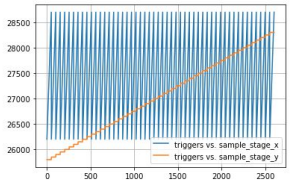


2. Computing in the cloud

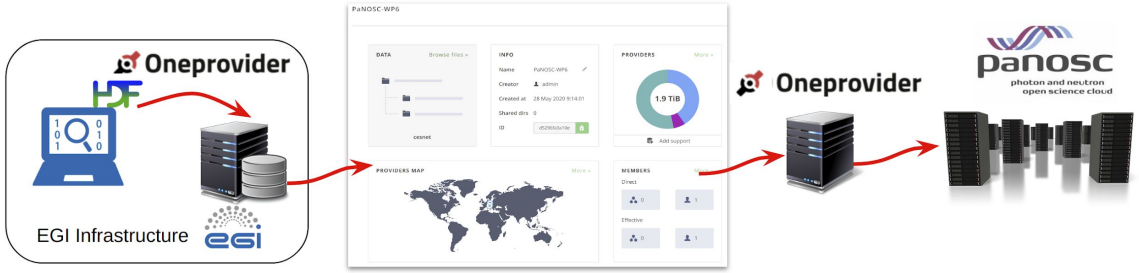



```
In [33]:
# This code snippet simulates the processing of datasets read from the volume space.
# As result of this simulation, the PNG file will be uploaded from the notebooks space
# in the default OneProvider (CESNET-MCC).

get_millisecs = lambda: int(round(time.time() * 1000))
get_millisecs()

print("Available keys: %s" % fh5py.keys())

plt_file="result_%s.png" %get_millisecs()

if ("triggers" in fh5py.keys()):

    # Getting datasets
    triggers = fh5py['triggers']
    timestamp = fh5py['triggers_timestamp']
    sample_stage_x = fh5py['sample_motors/sample_stage_x']
    sample_x_pos = fh5py['sample_motors/sample_x_pos']
    sample_y_pos = fh5py['sample_motors/sample_y_pos']
    sample_stage_y = fh5py['sample_motors/sample_stage_y']

    print("- triggers: %s" %triggers[:])
    print("- timestamp: %s" %timestamp[:])
    print("- sample_stage_x: %s" %sample_stage_x[:])
    print("- sample_x_pos: %s" %sample_x_pos[:])
    print("- sample_y_pos: %s" %sample_y_pos[:])
    print("- sample_stage_y: %s" %sample_stage_y[:])

    # Define data, grid, legend, and save the plot
    plt.plot(triggers[:], sample_stage_x[:], label='triggers vs. sample_stage_x')
    plt.plot(triggers[:], sample_stage_y[:], label='triggers vs. sample_stage_y')
    plt.grid()
    plt.legend()
    plt.savefig(plt_file, dpi = (100))
else:
    print ("Working in progress..")

#fh5py.close()
#f.close()
```

```
Available keys: <KeysViewHDF5 ['andor', 'dante', 'sample_motors', 'triggers', 'triggers_timestamp']>
- triggers: [   1    2    3 ... 2599 2600 2601]
- timestamp: [1.57496154e+09 1.57496154e+09 1.57496154e+09 ... 1.57496179e+09
 1.57496179e+09 1.57496179e+09]
- sample_stage_x: [26200. 26250. 26300. ... 28600. 28650. 28700.]
- sample_x_pos: [0. 0. 0. ... 0. 0. 0.]
- sample_y_pos: [0. 0. 0. ... 0. 0. 0.]
- sample_stage_y: [25800. 25800. 25800. ... 28300. 28300. 28300.]
```



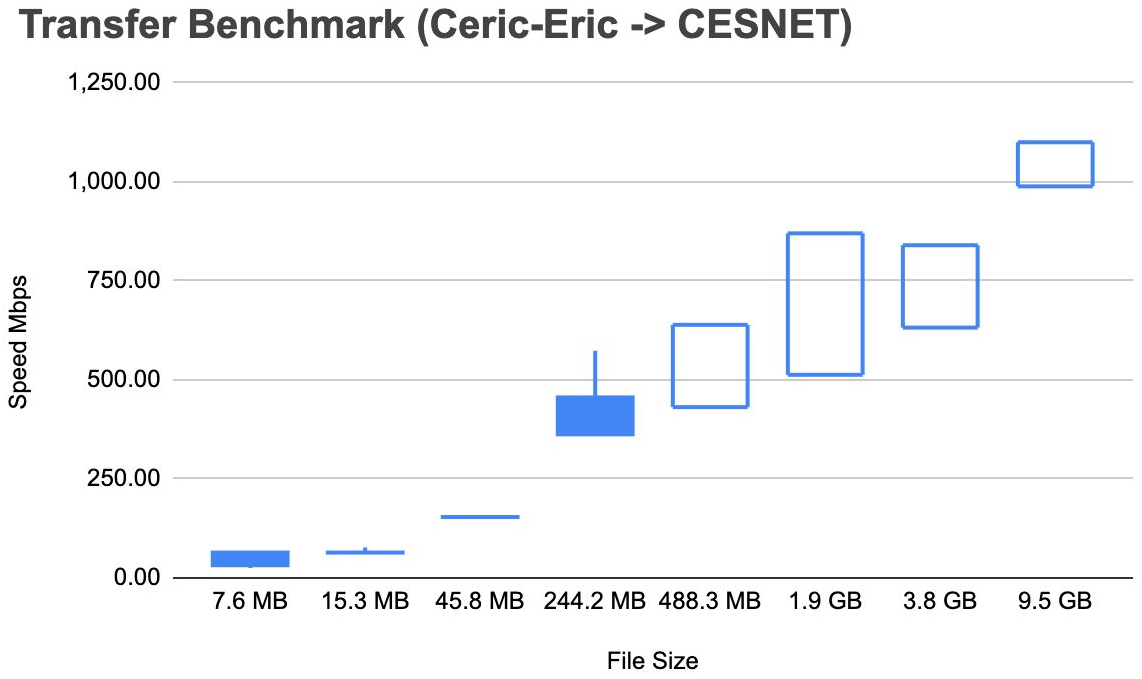
3. Transferring data back to local RI storage



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Results & conclusion

- Data transferred up to 100 GB
- HDF5 data files > 1GB can reach the maximum bandwidth available.
- Smaller files have a much smaller throughputs (⅕)
- Files < 50MB in size take the same amount of time of 250MB files, probably to the overhead required by the onedata transfer protocol, a standard pattern of network file transfer protocols.



We could extend the pilot deploying Oneprovider & a PaNOSC Space for each RI, so we will move data between RIs not only towards EGI cloud, once some issue with Luma will be solved

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Resources

1. Data Transfer Pilot on github
2. PaNOSC D6.1 Data-Hub
3. PaNOSC data transfer pilot setup
4. EGI DataHub
5. EGI Notebooks service
6. Onedata documentation
7. UmbrellaID
8. OnedataFS

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Thanks

The Pilot Team:

- Giuseppe La Rocca EGI Foundation
- Enol Fernández EGI Foundation
- Andrea Manzi EGI Foundation
- Marco De Simone CERIC-ERIC
- Michal Orzechowski CYFRONET
- Łukasz Opioła CYFRONET
- Bartosz Kryza CYFRONET
- Miroslav Ruda CESNET-MCC
- Andrei Kirushchanka CESNET-MCC
- Christos Kanellopoulos GEANT
- Grégory Fanjas (ILL)
- William Turner (ILL)
- Philippe Le Brouster (ILL)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852