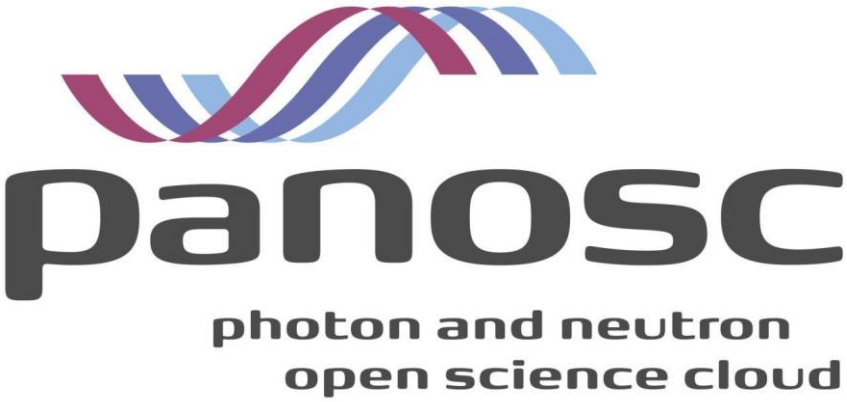
PaNOSC

Photon and Neutron Open Science Cloud

H2020-INFRAEOSC-04-2018

Grant Agreement Number: 823852



Deliverable:

Remote Desktop and Jupyter Service deployed at EOSC (4.3)

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

Project Deliverable Information Sheet

Project Reference No.	823852
Project acronym:	PaNOSC
Project full name:	Photon and Neutron Open Science Cloud
H2020 Call	INFRAEOSC-04-2018
Project Coordinator	Andy Götz (andy.gotz@esrf.fr)
Coordinating Organization:	ESRF
Project Website	www.panosc.eu
Deliverable No:	D4.3
Deliverable Type	Demonstrator
Dissemination Level	Confidential
Contractual Delivery Date:	31/05/2022
Actual Delivery Date:	04/07/2022
EU project officer	Flavius Alexandru Pana

Document Control Sheet

Document	Title: Remote Desktop and Jupyter Service deployed at EOSC
	Version: 1
	Available at: https://github.com/panosc-eu/panosc
	Files: 1
Authorship	Written by: Fabio Dall'Antonia
	Contributors: Stuart Caunt, Emiliano Coghetto, Marco De Simone, Jamie Hall, Teodor Ivanoaica, Giuseppe La Rocca, Jean-François Perrin, Thomas Vincent
	Reviewed by: Jordi Bodera
	Approved: Andy Götz

List of participants

Participant No.	Participant organization name	Country
1	European Synchrotron Radiation Facility (ESRF)	France
2	Institut Laue-Langevin (ILL)	France
3	European XFEL (XFEL.EU)	Germany
4	The European Spallation Source (ESS)	Sweden
5	ELI European Research Infrastructure Consortium (ELI-ERIC)	Belgium
6	Central European Research Infrastructure Consortium (CERIC-ERIC)	Italy
7	EGI Foundation (EGI.eu)	The Netherlands

Table of Content

1 Introduction

The Photon and Neutron Open Science Cloud (PaNOSC) project aims at providing a remote access infrastructure to enable and contain FAIR data services for users of the community and ideally scientists across domain borders, through the European Open Science cloud (EOSC). In the centre of this project, taken care of by work package 4, lies the actual cloud-based platform to (re-)use scientific data in terms of data analysis at the sites where the data were produced and are made accessible.

This major goal of PaNOSC is a reaction to the increase of data amounts produced by the facilities, which not only requires an increasing level of computer power and storage space in general, but makes a download and local computation for data analysis by single users partly infeasible. Enabling users to explore data through their web browser instead, after having identified a data set of interest (in case datasets from their own experiments before these have become open to the public), considerably lowers the barriers towards re-use of the data.

The Work Package deliverable 4.3 comprises the production-level provision of two principal types of data analysis services: the remote desktop for graphical software use and the Jupyter Notebook for programmatic data analysis. These services, as per deliverable, are available running in virtual machines, bundled with the necessary software tools respectively frameworks, as a remote service via the Internet. They are required to have been created and deployed at one partner site minimum.

This document reports on the production status of relevant implementations of remote data analysis services at all facilities concerned, where registration to the EOSC is a prerequisite to make them visible and available to a broader scientific community. As the deliverable concerned is closely based on the predecessor deliverable 4.2 (D4.2), defining the service prototypes, the report starts with a summary of D4.2, and explains the major deviation of service provision strategy upon transition from prototype to production. After presenting the separate solutions for both types of service and the VISA platform as integrated solution, with harmonised instances at all partner sites, the report concludes with a summary and outlook on sustainability of the data analysis services.

2 Context

2.1 Prototype status recapitulation (D 4.2)

Deliverable 4.2 introduced concepts and flavours of data analysis – online and offline analysis as well tutorials – and highlighted the improvement of user experience with remote tasks, in particular involving graphical result visualisations, thanks to the complementary approaches of remote desktop services running GUI tools and JupyterHub servers running notebooks – as compared to the lower performing ssh access (with X forwarding).

Concerning the status of remote desktops, there had been three production services and two pilot services present at the sites, whereas for JupyterHub two production services and four pilot services were reported. Individual solutions for each of these were presented, both from the usage perspective and with respect to technical aspects.

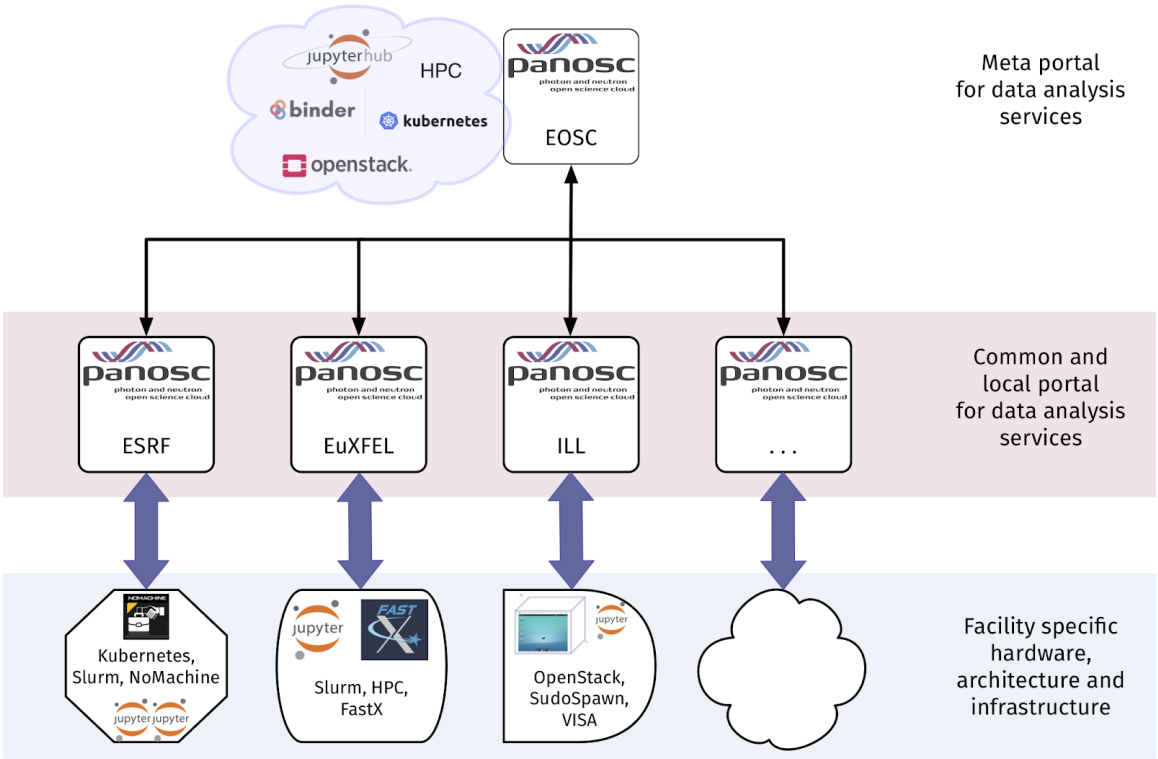


Figure 1: Concept of data analysis services harmonisation through local, but common-design portal implementations and a centrally (EOSC) hosted meta portal.

Finally, an outlook on the harmonisation of services was given, presenting the concept of a portal – being an integrated web-service solution with front-end user interface and a back-end of microservices connecting to virtual machines running the actual and diverse service implementations on the cloud-cluster infrastructures of the facilities.

2.2 Strategy revision

Based on the portal concept, a demonstrator was developed and presented by the end of 2020. It was planned to follow this approach of a central platform for cloud-based data analysis services, and developments had started to advance to a production-ready version. The aimed at design was to combine a search module (web-based frontend applications and back-end API for searches against metadata catalogues) with micro-services for user authentication, virtual machine (VM) management and embedding of both a remote desktop solution and a Jupyter server for data analysis. It was planned to support VMs running on different cloud infrastructures, in particular OpenStack and Proxmox were aimed to be supported out-of-the-box.

Meanwhile, with the outbreak of the COVID-19 pandemic and the ban on travel in 2020, the initial plans to develop a new solution had to be adapted in order to accommodate the urgent need to allow remote experiments. Therefore the cloud-based VISA (Virtual Infrastructure for Scientific Analysis) platform was put in production at ILL, based on OpenStack. ILL colleagues who were also driving the development of the PaN portal, performed a thorough implementation- and maintenance-effort analysis. Their comparison of a completely microservice-based PaN portal, partly ported from VISA code but largely rewritten, versus the adoption of VISA by other PaNOSC partners showed the latter to be the most efficient use of the available resources within PaNOSC to achieve the objectives of PaNOSC. The decision to improve the existing VISA codebase, package and document it therefore constitutes the best approach for the scope of the project, as well as for1 the sustainability beyond the lifetime of PaNOSC. The pros and cons of the change of plans were discussed within WP4 and all partners agreed that adapting VISA was the best solution. The VISA code-base has been made open- source1 and synchronised to the PaNOSC repository2.

As of mid 2021, the WP4 colleagues have thus decided for a new development-, deployment- and usage strategy for the PaN portal based on the ILL-VISA codebase. Deployment details are described in Task 4.5 (see below). Integration to EOSC will first be in terms of facility instances of VISA, which will be registered in EOSC separately. On top of this, ELI and EGI have started a pilot project (linked to WP6) for a federated VISA instance, hosted at EGI. The deployment task is expected to be ready by the first quarter of 2022. One aim of the pilot project with EGI is to estimate the sustainability of a federated service, for WP7.

The decision of adopting the existing VISA codebase (as was proposed in the initial proposal) has the consequence that cross-facility searching for open data needs to be implemented separately. The separate, federated, PaN search portal will make use of existing service components developed in WP3 and WP4. This development is not part of the deliverable 4.3.

[1] https://github.com/ILLGrenoble
[2] https://github.com/panosc-portal

3 Remote Jupyter services

3.1 Jupyter notebook concept

The Jupyter Notebook is a web application that allows users to create and share documents containing live code, equations, visualisations and descriptive text. A notebook server hosts notebook documents, which are JSON files, and provides the interface to computation kernels (interpreters with environments for Julia, Python or R).

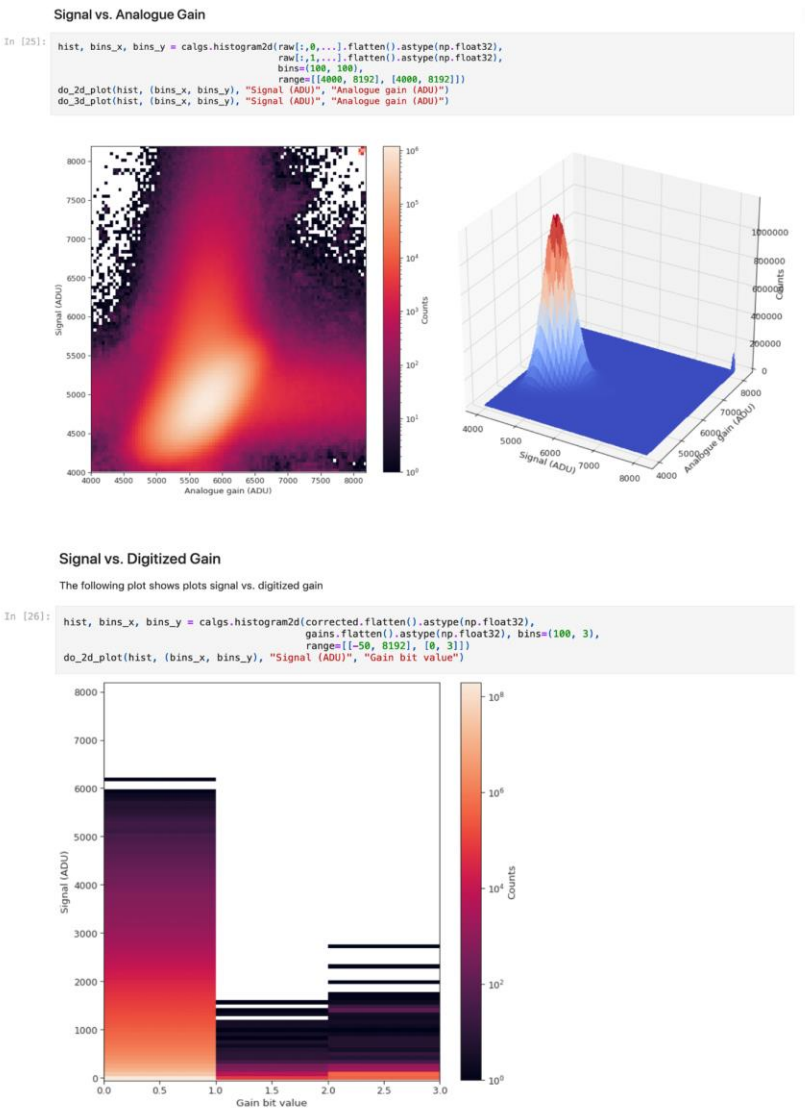


Figure 2: Jupyter notebook used for detector calibration at European XFEL (PaNOSC use case 5)

Jupyter notebooks are popular within the scientific community because they present data analysis tasks as workflows with a clearly defined order (code cells) thus enhancing the ease of sharing data analysis workflows.

3.2 Jupyter Hub

JupyterHub is an infrastructure that provides a multiuser implementation of the Jupyter Notebook, enabling users to run notebook servers on shared hardware. JupyterHub allows for user and resource management by system administrators and can run using a variety of backends.

Pre-existing resource managers at facilities can be used for spawning new users notebook servers – such as Kubernetes, Docker Swarm or Slurm. In this case users' Jupyter instances can be distributed across multiple servers, and it is possible to take advantage of already existing Graphics Processing Unit (GPU) or High-Performance Computing (HPC) resources.

3.3 Jupyter service implementations and registration status

Table 1 Jupyter service instances at PaNOSC RIs and their EOSC registration

Facility	**Service**	**Prototype**	**Production**	**URL**	**EOSC registration**	
					In progress	done
ESRF	JupyterHub		✔	https://jupyter-slurm.esrf.fr		✔ [1]
ILL	VISA		✔	https://visa.ill.fr		✔ [2]
EuXFEL	JupyterHub		✔	https://max-jhub.desy.de	(✔ *)	
ESS	Jupyter Hub		✔	https://e-learning.pan-training.eu		✔ [3]
ELI-ERIC	Jupyter Hub		✔	notebooks.eli-laser.eu	✔	
CERIC-ERIC	Jupyter		✔	https://vuo.elettra.eu	✔	
EGI	Jupyter Hub		✔	https://notebooks.egi.eu		✔ [4]

* Registration will be done by DESY

[1] https://marketplace.eosc-portal.eu/services/jupyter-notebook-can-be-used-to-create-and-share-documents-that-contain-live-code-equations-visualizations-and-text
[2] https://marketplace.eosc-portal.eu/services/visa-virtual-infrastructure-for-scientific-analysis
[3] https://marketplace.eosc-portal.eu/services/pan-learning-org-f68a8ee0-ed50-4eb1-b51a-b28516df7966
[4] https://marketplace.eosc-portal.eu/services/egi-notebooks

Jupyter ecosystem services are present either as stand-alone web-service or as part of a larger service infrastructure:

3.3.1 ESRF

The service running under https://jupyter-slurm.esrf.fr provides access to two compute cluster partitions: one based on X86 computers and a second one based on Power9 computers with GPUs. The landing page of this service where users can select the computing resources they want to use is based on jupyterhub_moss (version 3.0.0). This service provides JupyterLab version 2 and version 3. The JupyterLab HDF5 file viewer jupyterlab-h5web (version 3.0.0) is available with this service.

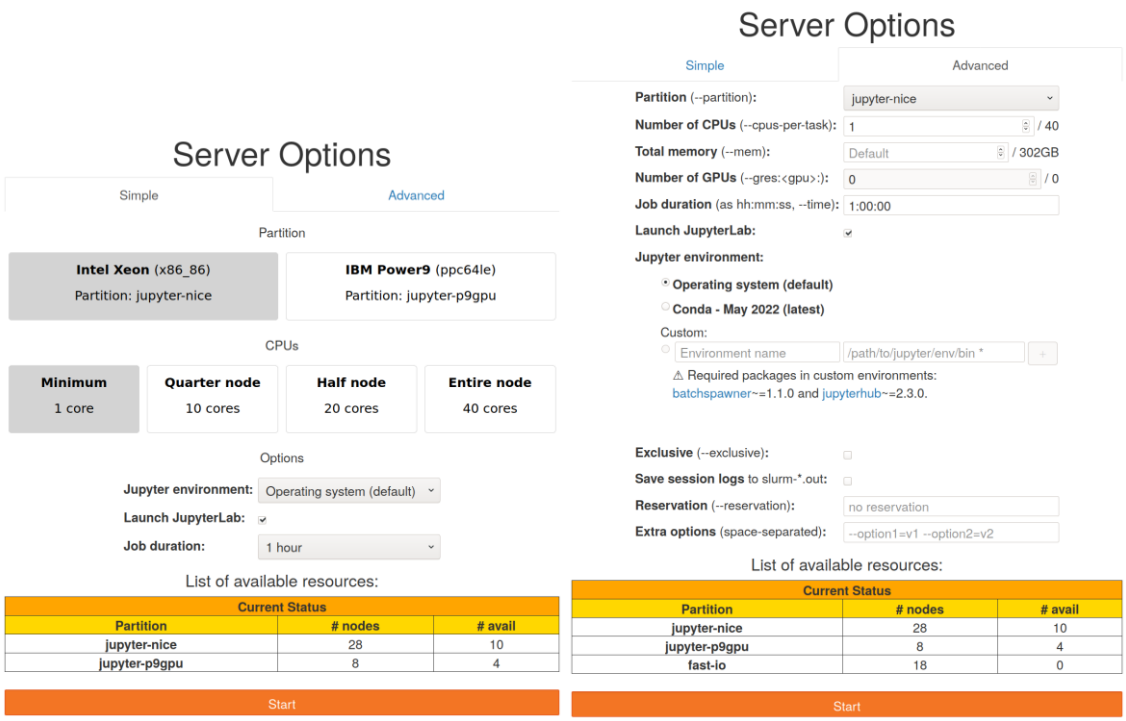


Figure 3: Interface of https://jupyter-slurm.esrf.fr for selecting computing resources

3.3.2 ILL

JupyterLab has been integrated inside the VISA platform (https://visa.ill.fr). All ILL users are free to create a data analysis virtual machine which will give them access to a remote desktop or Jupyter lab environment. Access to VISA is publically available to the internet (an ILL account is required to login to the platform). The user can currently obtain resources of up to 16vCPUs / 128GB memory.

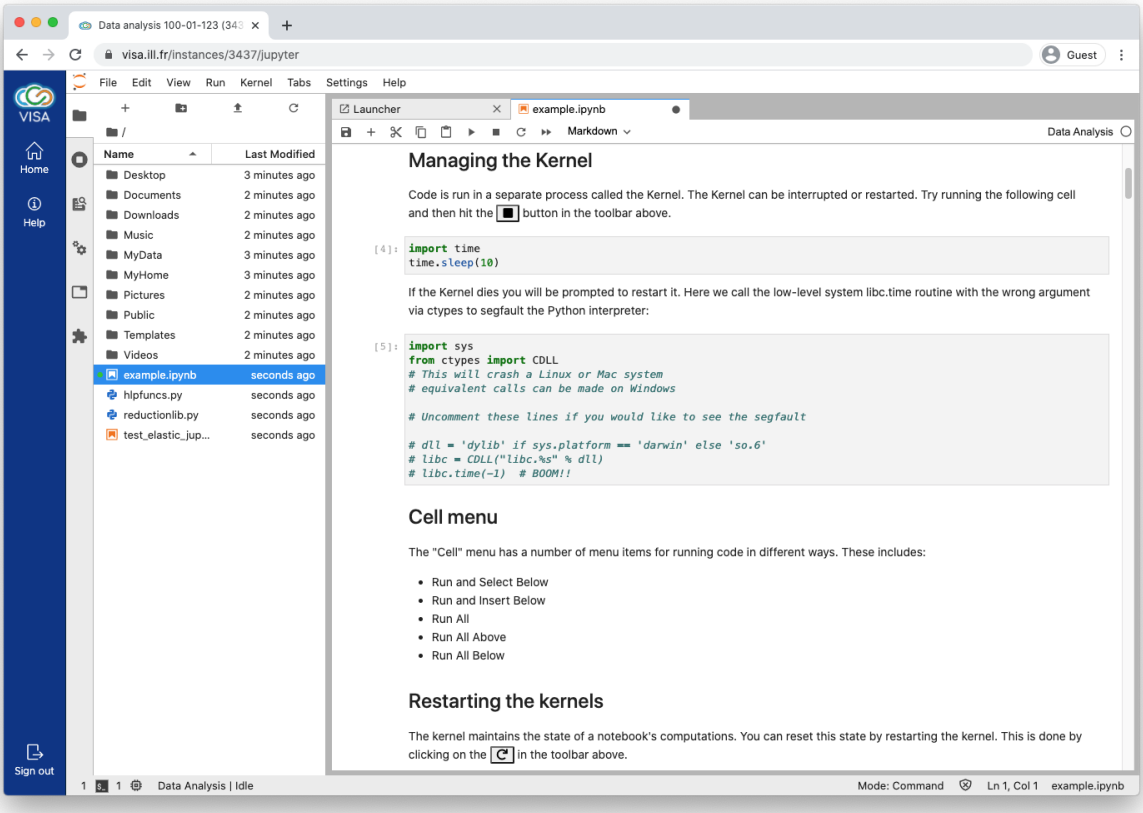


Figure 4: Jupyter notebook from the JupyterLab service within a VISA VM instance

3.3.3 European XFEL

The service running under https://max-jhub.desy.de is provided by Deutsches Elektronen-Synchrotron (DESY). European XFEL uses two dedicated partitions on the DESY HPC cluster, one for staff and one for beamtime users.

3.3.4 ESS

The service running under https://e-learning.pan-training.eu provides, through a slurm frontend, access to JupyterLab dedicated compute resources on the ESS HPC cluster for use by e.g. neutron physics workshops, experiment preparation or student training, and are available for users registered with e-learning.pan-training.org.

3.3.5 ELI-ERIC

A JupyterHub Infrastructure service is under development will be hosted by ELI ERIC and made available to ELI users and personnel. This deployment was massively delayed by the delays in procuring the hardware dedicated to ELI ERIC Computing Infrastructure. A temporary solution is already in place and the development and implementation processes are started.

Access for staff shall be managed via local identities, while access for users shall be granted using UmbreallaID and ORCID, the platform will be reached through https://notebooks.eli-laser.eu.

3.3.6 CERIC-ERIC

JupyterLab instances are available via VUO/Rafec inside singularity containers running on edge devices available to each beamline. They can be reached by users through https://vuo.elettra.eu

3.3.7 EGI

On top of the Jupyter notebook production service as listed in the above table, the service https://notebooks-panosc.fedcloud-tf.fedcloud.eu/ and a Binder instance https://binder-panosc.fedcloud-tf.fedcloud.eu/ are deployed on the CESNET-MCC cloud infrastructure. From a technical perspective, these two services are not in production even if production recipes were used. A Kubernetes cluster composed by one master and 3 worker nodes is used to spawn jupyter notebooks on demand. No GPU resources are currently available. Access to the resources granted using UmbrellaID.

The h5web plugin for JupyterLab (https://github.com/silx-kit/jupyterlab-h5web) has been installed at all facilities.

4 Remote desktop services

4.1 Introduction

A special graphical forwarding mechanism has to be employed if connections to remote computing resources shall include the export of graphical components to the local computer display of a user. A secure-shell (ssh) connection with X-forwarding is a basic solution, but dedicated services for that purpose have a better performance and offer a more comprehensive user experience, providing an entire graphical desktop environment including operating system aspects, hence they are called remote desktop services.

A strong argument for data analysis through remote desktop services is the coverage of software that can only be run with graphical user interfaces (GUIs). While Jupyter services support graphics as well, in particular for data visualisation, users may prefer GUI software because they are used to it, or are beginners that benefit from the low bar of entry that a well-designed GUI offers.

At the time of the prototype deliverable (D4.2) there were several solutions either tested or in production at the sites of PaNOSC partners.

Table 2 Individual desktop solutions in the early phase of PaNOSC

Facility	**Remote desktop**	
	Status	Technology
ESRF	Production	NoMachine
ILL	Production	VISA (XRDP, Guacamole)
EuXFEL	Production	FastX
ESS	Pilot	VNC
ELI-ERIC	NA	NA
CERIC-ERIC	Pilot	VNC, Guacamole, XPRA

4.2 Software requiring a remote desktop service

Table 3 Example of software used in the PaN community that require, or benefit from, a GUI

Application	Description	GUI requirement	Jupyter notebook alternative
Sasview	Small angle scattering analysis	essential	partly
Crispy	GUI for core level spectra calculation	essential	no
Mantid	Neutron scattering analysis framework	essential *	partly
PyMca	X-ray powder diffraction analysis	optional	yes
McStas	Neutron scattering experiment simulation	optional	partly
PyFAI	Fast Azimuthal Integration	optional	yes

A complete list of relevant PaN software is available at the PaNOSC software catalogue, which is available under https://software.pan-data.eu/ and registered in EOSC at https://marketplace.eosc-portal.eu/services/panosc-software-catalogue?q=PaNOSC+Software+Catalogue.

4.3 Remote desktop service implementations and registration status incl. VISA

Table 4 Overview of remote desktop services to be used in production at the end of PaNOSC

Facility	Service	Prototype	Production	URL	EOSC registration	
					In progress	done
ESRF	Guacamole		✔	https://remote.esrf.fr	✔	
ESRF	VISA		✔	https://visa.esrf.fr	✔	
ILL	VISA		✔	https://visa.ill.fr		✔
EuXFEL	FastX		✔	https://max-exfl-display.desy.de:3443	✔	
EuXFEL	VISA	✔		https://visa.xfel.eu	✔	
ESS	VISA	✔		https://visa-prod.esss.dk		
ELI-ERIC	VISA	✔		https://visa.eli-laser.eu		
CERIC-ERIC	VISA	✔		https://visa.ceric-eric.eu **		
EGI	VISA			N/A		

* Registration will be done by DESY

** currently available only internally

[1] https://marketplace.eosc-portal.eu/services/visa-virtual-infrastructure-for-scientific-analysis

5 VISA Platform

5.1 Description of the service

VISA, as deployed at ILL (https://visa.ill.fr), is a service in production for more than 18 months, which has been used by more than 1,100 facility users so far.

Users can select their experiment and resource options (memory, cpu, display) for the virtual machine to be used, and the type of analysis service to use within the VM: a Jupyter notebook or a remote desktop with access to the software stack contained in the VM image.

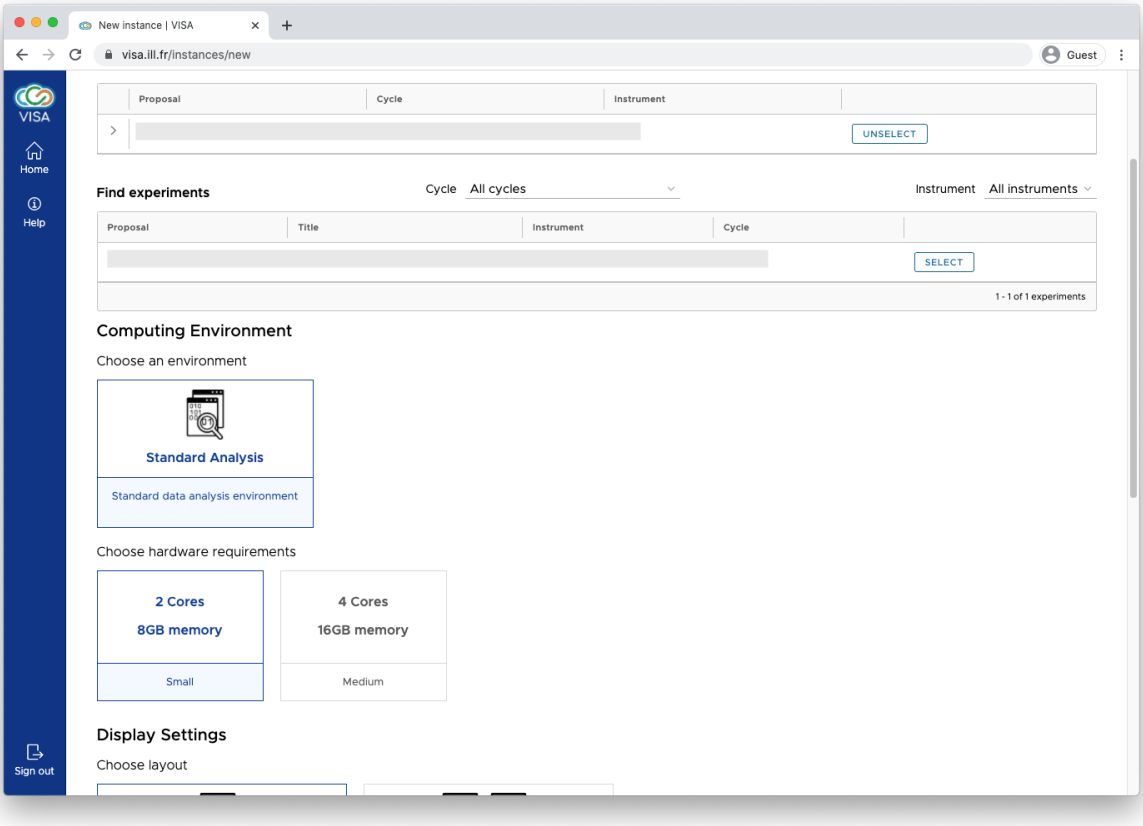


Figure 5: VISA instance preparation

Often data analysis is performed as a team and as such VISA has been designed with real-time collaborations at its core. The platform makes it easy to share analysis environments between proposal team members, ILL scientists and computing experts. In parallel to doing data analysis, VISA provides an ideal platform for performing remote experiments. Accessible anywhere in the world where there is an internet connection, a scientist has the freedom to remotely pilot an experiment via the instrument control software, be it from the comfort of their own home or at their laboratory.

VISA also takes care of the lifecycle and creation of the Virtual Machines and resources are managed depending on the type of experiment a user has selected.

The VMs have direct access to the ILL's experimental storage, analysis procedures and scientific software (i.e. MANTID, MatLab, FullProf, LAMP etc.) which are ready to be used from the get go. The platform

can also be used for training users such as learning about Mantid or Python programming for performing data analysis.

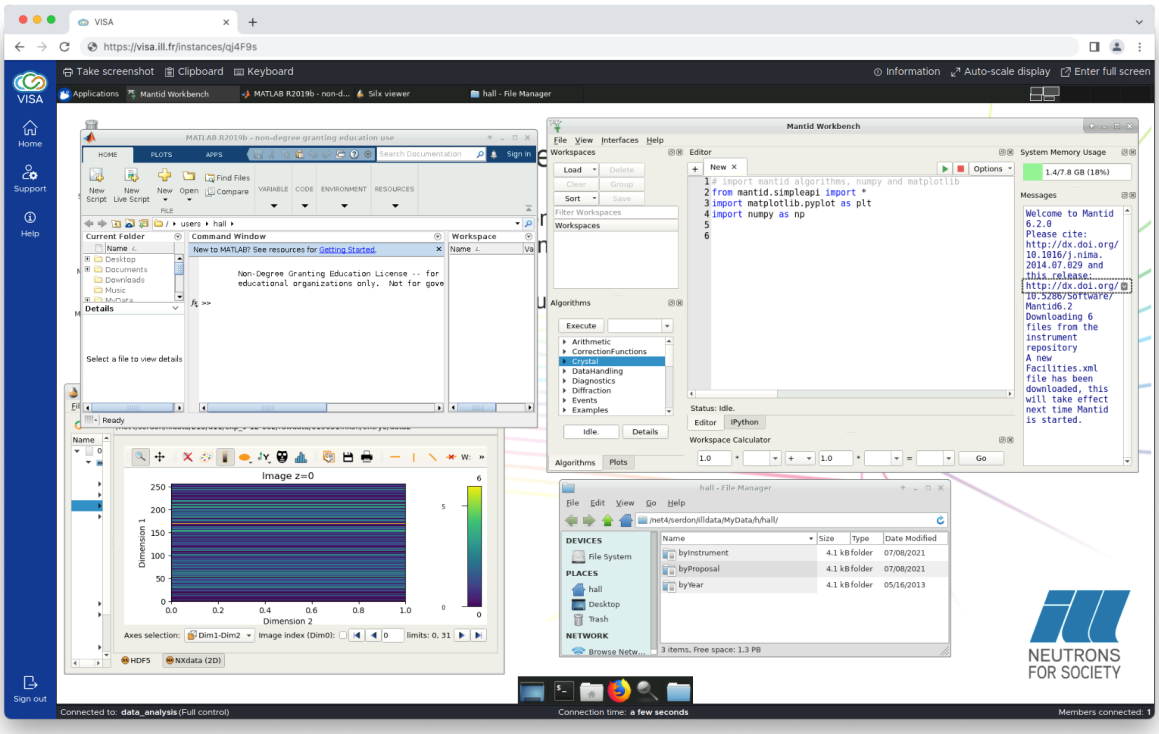


Figure 6: Running VISA instance with several GUI software opened in the remote desktop

The VISA database provides the link between user and experiment information. In order to fill the database of VISA instances, an ETL (Extract, transform, load) process needs to be performed. For adoption of VISA at partner RIs, the establishment of an ETL mapping from facility-specific experiment/user information and structure of their user office database to VISA requires some effort and information on the roles of different users. Some site-specific adaptation of the VISA Accounts service (the backend authentication process based on the OpenID Connect protocol) is also necessary to allow partner RIs to link the authenticated user to the VISA database. As well as these necessary software developments, each site must also set up their own cloud infrastructure. The following schema depicts the architecture of VISA:

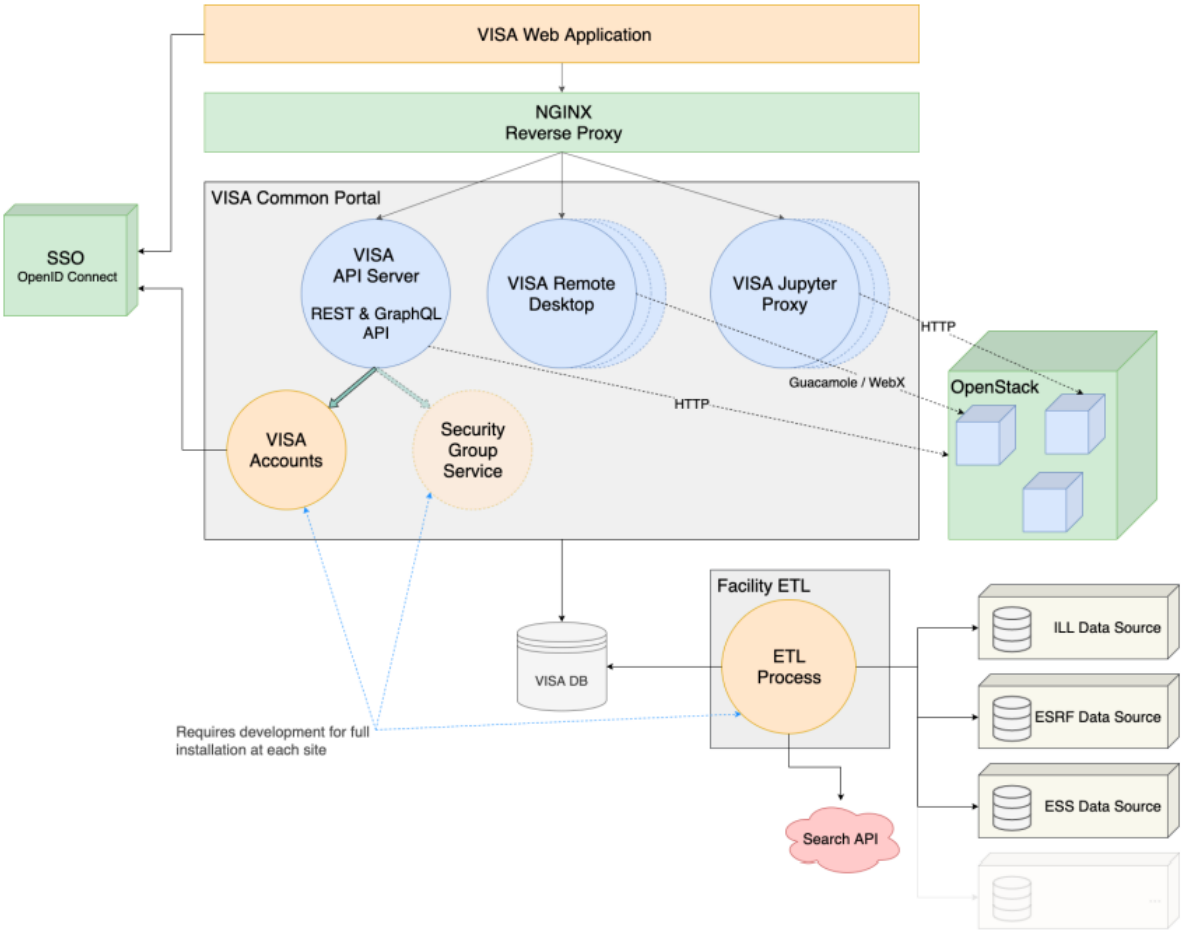


Figure 7: Schematic diagram of VISA architecture

VISA is currently at version 2.2.1. Latest developments include features for the VISA administration user interface, implementation of a more generic user authentication management and improvements for the Jupyter service.

Documentation of VISA can be found at: https://visa.readthedocs.io/en/latest

5.2 VISA Deployment and usage status

The deployment phase started in September 2021. Good deployment progress has been made at CERIC, ESS, ESRF and EuXFEL (with the help of DESY experts) who succeeded with proof-of-concept VISA deployments. Other partner RIs have set up or are on the way to set up OpenStack to start demonstrating VISA to their users.

5.2.1 ESRF

At the ESRF synchrotron, the project is running with 2 different beamlines acting as pilot, there users will be the first to benefits from the production VISA service. These pilots have expressed the need to get GPUs available for their users as well as access to HPC cluster. The microservice deployment has been accomplished, GPU resources added and an HPC cluster using SLURM as job scheduler added. Instead of installing the scientific software inside the virtual machines, software is provisioned using CernVM-FS2 (CVMFS) and packaged inside singularity3 containers.

5.2.2 ILL

At ILL, VISA is in production. For details, please refer to the previous section.

5.2.3 European XFEL

European XFEL have a deployed VISA instance (visa.xfel.eu) interfacing with the DESY OpenStack cloud infrastructure, where microservice (re-)deployment is managed by a Gitlab/Kubernetes continuous integration workflow. The installation is coupled to OpenID Connect authentication with Keycloak and is currently providing three optional virtual machine images in demonstration state, one providing crystallographic software.

[2] https://cvmfs.readthedocs.io/en/stable/

[3] https://sylabs.io/guides/3.0/user-guide/

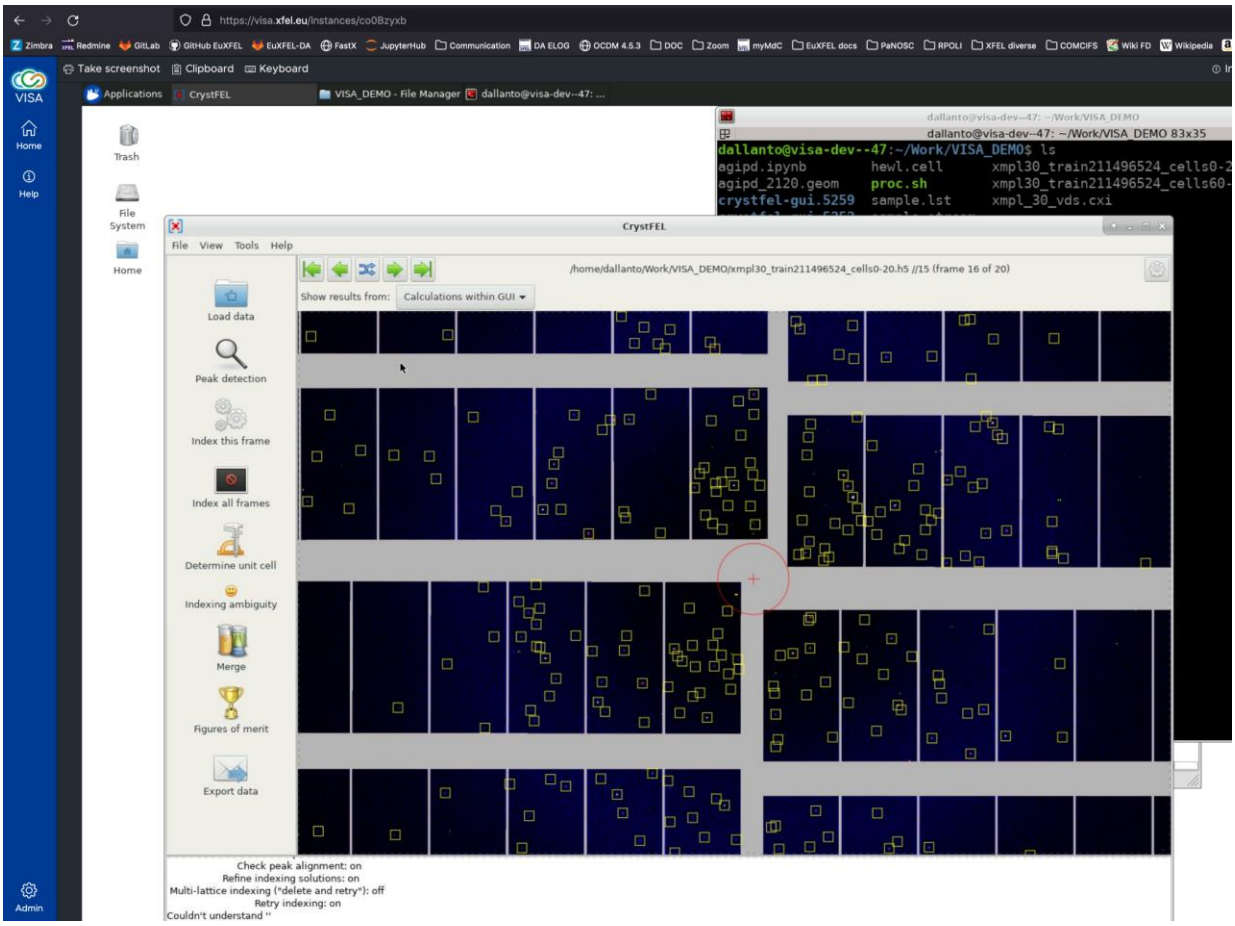


Figure 8: The European XFEL instance of VISA running the remote desktop service with the CrystFEL GUI.

In the first production phase, VISA will provide experiments and associated data from a selected set of open "example" data that is also exposed in the open section of the metadata catalogue for request by the PaN federated open data search portal (via search API). These are datasets with corrected multi-gain detector data that does not require the European XFEL-specific correction/pre-processing step and can be readily used by software tools, such as CrystFEL in the case of serial femtosecond crystallography data.

5.2.4 ESS

The ESS has VISA ready as service using virtual machine instances based on the demo image. As such, it is already used by the ESS instrument data scientists and software developers.

5.2.5 CERIC-ERIC

The core VISA microservices have been deployed on a production virtual machine managed by Proxmox https://www.proxmox.com, the virtualization management platform in use at CERIC. Some site-specific adaptations have been implemented on the authentication, database data injection and cloud infrastructure access side.

The VISA Accounts service has been configured to use the OpenID Connect authentication provided by a Keycloak instance in use at CERIC.

An ETL (Extract, Transform, Load) process to fill the database of VISA has been developed. The process regularly synchronises the VISA database with the relevant portion of CERIC's data source (the Proposal Management and User Office System in use at CERIC).

Lastly, on the access to the cloud infrastructure, a couple of minor adaptations have been implemented on the VISA Cloud Provider service to let VISA interface with Proxmox using a specific resource pool (a set of virtual machines, containers, and storage devices).

5.2.6 ELI-ERIC

VISA is being deployed using the hardware infrastructure provided by ELI Beamlines and shall be in production and registered in EOSC by the end of PaNOSC.

For this initial setup the VISA Applications will run either in a Kubernetes cluster or in an existing VSphere virtualization infrastructure provided by ELI ERIC. For virtualization ELI ERIC is now exploring the possibility of having in production a Proxmox Cluster. Keycloak will be used to serve the OpenID Connect endpoint, as it is already providing support for other ELI ERIC services.

At the same time the ETL services required by the VISA setup are not developed and they depend on the final architecture of the ELI ERIC IT and Data Infrastructure.

6 Summary and Outlook

Almost all PaNOSC partners have a production instance for Jupyter notebook services in place. The VISA deployment has reached a state where nearly all partners can use a close-to-production demo or an actual production instance, so that a cloud solution for data analysis can be offered to users.

The availability of both aspects - Jupyter notebook and remote desktop service (VISA as integrated platform and in case an alternative remote desktop tool) - will be extremely valuable for the last phase of the project where the added value of such developments to the PaN user community has to be demonstrated. The more usage the services get, the more easily a continuation of development and maintenance can be motivated at the institutions, respectively the more likely will funding for additional personnel and hardware resources be.

The registration of the services to EOSC is a means to promote them to the PaN users, whether this addresses a RI's own user base in the first place, or already aims at a more open model of all-PaN community usage. The registration process of services at EOSC has been started or even been completed by most of the partners. A complete registration of all services including VISA at all partner RIs is achievable until the end of the project.