Deliverables

Deliverable Number	**D24.5**
Deliverable Title	**Report on test and deployment of mini demonstrator on at least six sites**
Lead Beneficiary	**ELETTRA**
Authors	**George Kourousias (ELETTRA)** **Ivan Adrian (ELETTRA)** **Daniel Salvat (ALBA)** **Alex Camps (ALBA)** **Aidan Campbell (ESRF)** **Andy Götz (ESRF)** **Gregory Viguier (SOLEIL)** **Tom Schoonjans (DIAMOND)** **Kai Kaminski (PSI)** **Johannes Reppin (DESY)**
Type	**Report**
Dissemination Level	**Public**
Due date of delivery	**Month 24**

Report on test and deployment of mini demonstrator on at least six sites

Test Survey structure and summary

The system as a whole is based on blueprint presented in D24.2 *Blueprint on implementing a DAAS platform,* and is made of multiple interconnected software components and services, many of which are still under development. Even if the project has common objectives, each facility may have slightly different needs from the portal and some have more advanced deployments than other. Note that the final deployment will be reported during the D25.7 Workshop. This document reports the testing of the system as of M24 by presenting an extensive survey and its summary. This survey has a formal section of 9 question (section A) and a general one where useful information regarding the testing and deployment is reported (section B; 7 questions). Section A is the examination of the system through *Functional* and *Non-functional Testing*. The outline of this examination is based on the following structure:

1. Functional and non-functional tests (A questions)

The first set of questions answered by the testers were based on functional tests:

1. Unit testing
    - Report on individual software components
2. Integration testing
    - example: login + portal + application + data
3. System testing
    - Does the complete infrastructure a whole satisfy the initial requirements?
4. Acceptance testing
    - Readiness for delivery and wide deployment

The following questions were on non-functional aspects:

5. Performance testing
    - Responsiveness, I/O, VNC, Jupyter performance
6. Security testing
    - Login, general concerns, etc
7. Usability testing
    - General feedback from beamline scientists and computing staff
8. Compatibility testing
    - Is it compatible with the classical workflow?

An additional set of questions (see below) concerned critical comments, technical observations and suggestions

Finally **seven** facilities participated in the deployment, testing and reporting of the results.

These facilities were ALBA, ELETTRA, ESRF, SOLEIL, PSI, DIAMOND and DESY.

2. Summary of results

The results can be summarised as follows:

- The system as a whole follows successfully the proposed Blueprint D24.2.
- Seven facilities deployed it and tested it; three of which have advanced installation and the rest have only limited integration.
- Three facilities involved beamline and laboratory scientists (end-user) during testing and reported their feedback. The feedback is mostly positive.

- Basic Installation is easy but the Integration to the existing ICT systems, services and databases is very challenging due to the complex infrastructure and requirements to have a tight integration.
- Documentation must be improved.
- The installation instructions have to be polished and formalised prior to wider deployment.
- Jupyter, PyMCA, Guacamole/VNC/RDP were among the tested software delivered through the platform as Docker containers.
- The connection to the Data Catalog has not been completed and the Data Catalog is identified as an open issue.
- Local authentication requires improvements.
- No performance results could be reported by the facilities since the deployment is in beta still it was reported that the system felt responsive.
- The portal is still under development (as planned). Its online community (using Slack) is active and responsive.
- Security issues need to be addressed in the future.
- Admin and monitoring tools need to be further developed.
- The system is suitable to serve as a starting-point and demonstrator for EOSC related projects like PaNOSC and EXPaNDS.
- The feedback from the tests with users is positive and see the portal playing a useful role.
- A number of the sites plan to extend the portal into a production ready service for users.
- The next version and results of this system will be reported during the Workshop of D24.7 in the future.

All systems were deployed starting with a basic installation which had to be altered in order to facilitate the integration to each facility’s ICT systems, services and databases.

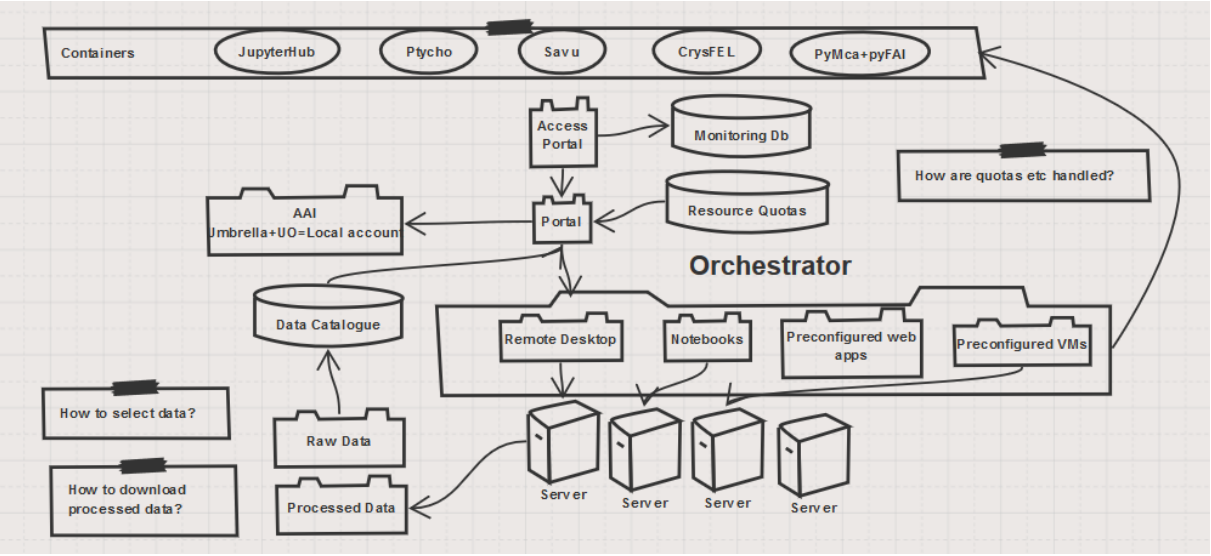


Figure 1 Architecture of DAAS Portal prototype (from Blueprint D24.2 deliverable)

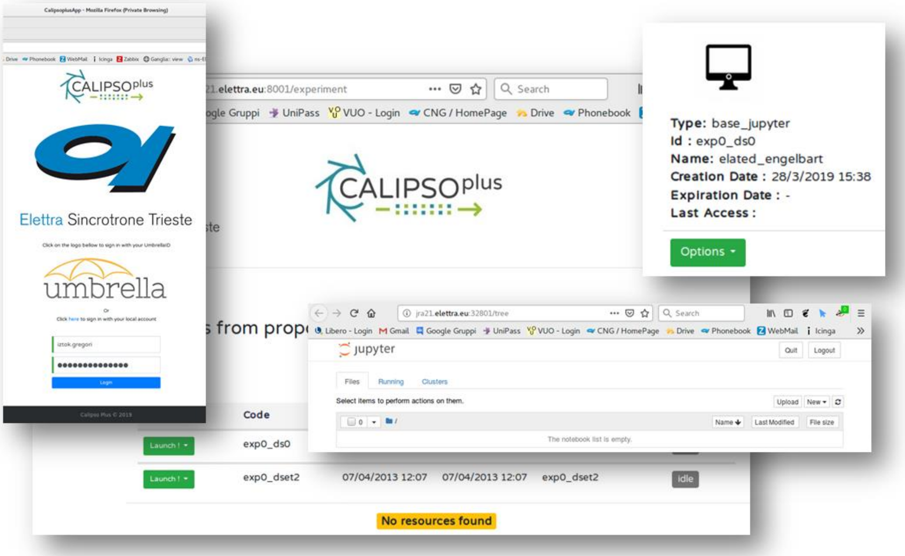
3. Instructions for basic installation

1. Install the backend - Complete instructions available at https://github.com/Calipsoplus/calipsoplus-backend

2. Install the frontend - complete instructions available at https://github.com/Calipsoplus/calipsoplus-backend

3. Configure the REST API on your facility's Users Office portal to connect the JRA2 portal to it for local authentication, authorisation, local data provider
4. Deploy the demo container available at https://github.com/Calipsoplus/calipso-docker-demo and check if the whole system is working
5. Connect the real data, according to the local data provider API responses, to the demo container
6. Deploy the use cases application containers and test them on real data and report about your experience by completing the survey

4. Screenshots from the deployments at facilities


CalipsoplusApp - Mozilla Firefox (Private Browsing)
CALIPSOplus
Elettra Sincrotrone Trieste
Click on the logo bellow to sign in with your UmbrellaID
umbrella
Click here to sign in with your local account
Login
Calipso Plus © 2019
jra21.elettra.eu:8001/experiment
Search
from propo
jra21.elettra.eu:32801/tree
jupyter
Quit
Logout
Files
Running
Clusters
Upload
New
Name
Last Modified
File size
Options
Launch !

Type: base_jupyter
Id : exp0_ds0
Name: elated_engelbart
Creation Date : 28/3/2019 15:38
Expiration Date : -
Last Access :

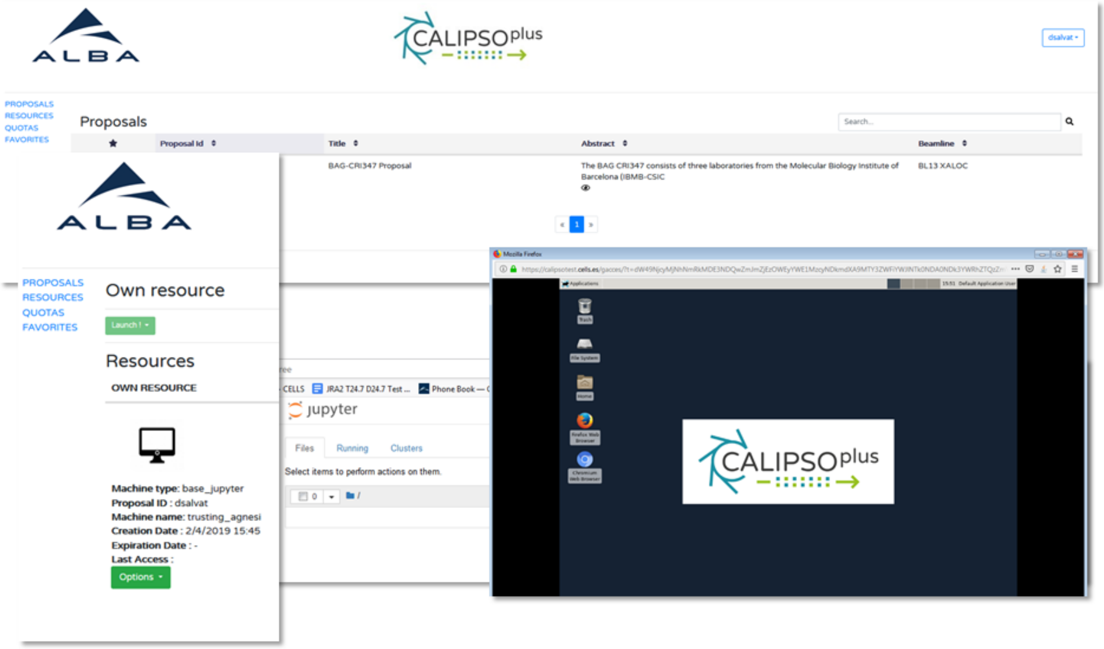
Select items to perform actions on them.

The notebook list is empty.

Code				
exp0_ds0				
exp0_dset2	07/04/2013 12:07	07/04/2013 12:07	exp0_dset2	idle

No resources found

Figure 2 DAAS portal @ ELETTRA



Proposals

★	Proposal Id	Title	Abstract	Beamline
		BAG-CRI347 Proposal	The BAG CRI347 consists of three laboratories from the Molecular Biology Institute of Barcelona (IBMB-CSIC	BL13 XALOC

Machine type: base_jupyter
Proposal ID : dsalvat
Machine name: trusting_agnesi
Creation Date : 2/4/2019 15:45
Expiration Date : -
Last Access :

Select items to perform actions on them.

Figure 3 DAAS portal @ ALBA

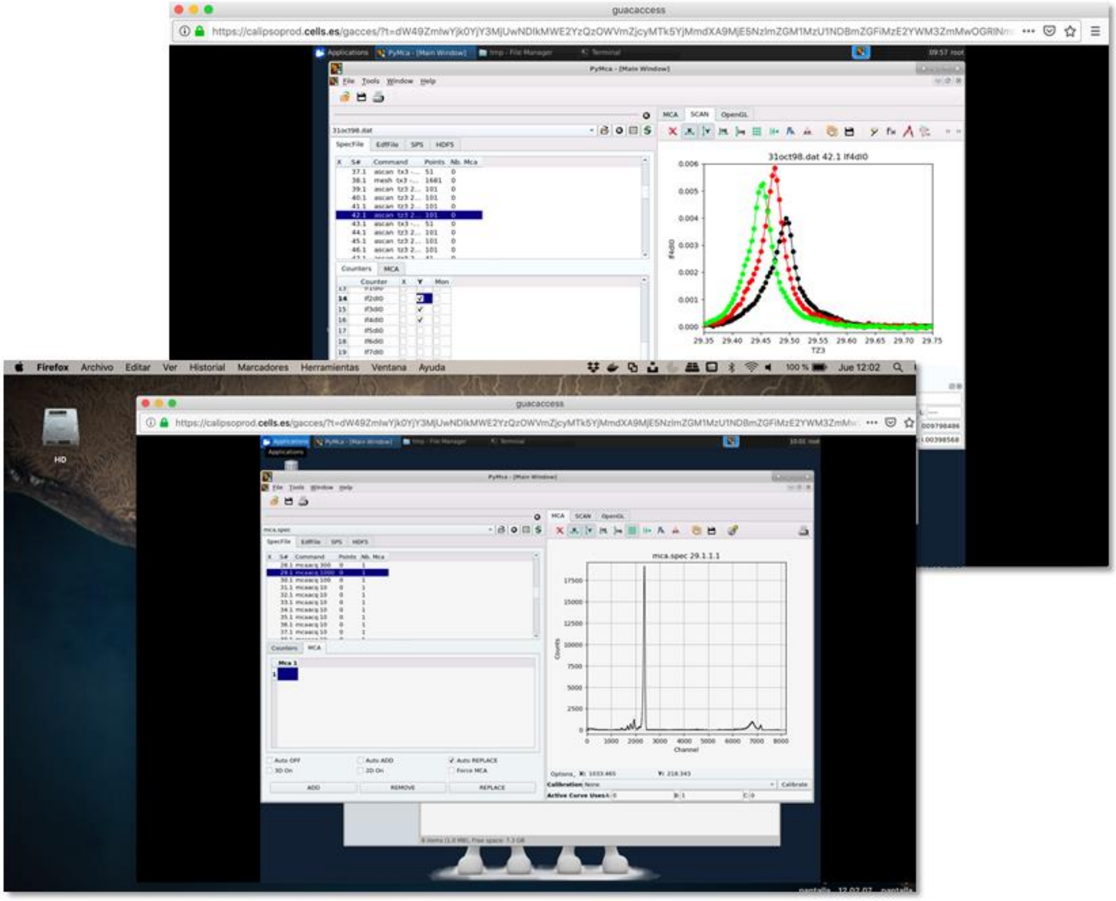
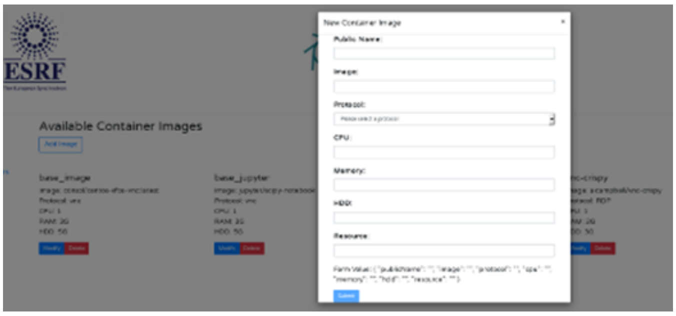

guacaccess
https://calipsoprod.cells.es/gacces/?t=dW49ZmlwYjk0YjY3MjUwNDlkMWE2YzQzOWVmZjcyMTk5YjMmdXA9MjE5NzlmZGM1MzU1NDBmZGFiMzE2YWM3ZmMwOGRlNm
PyMca - [Main Window]
File Tools Window Help
31oct98.dat
SpecFile EdfFile SPS HDF5
MCA SCAN OpenGL
31oct98.dat 42.1 If4dI0
TZ3
Firefox Archivo Editar Ver Historial Marcadores Herramientas Ventana Ayuda
Jue 12:02
mca.spec
mca.spec 29.1.1
Counts
Channel
Auto OFF
Auto ADD
Auto REPLACE
ADD
REMOVE
REPLACE
Calibrate

Figure 4 Example of applications running in DAAS portal @ ALBA



New Container Image

Public Name:

Image:

Protocol: Please select a protocol

CPU:

Memory:

HDD:

Resource:

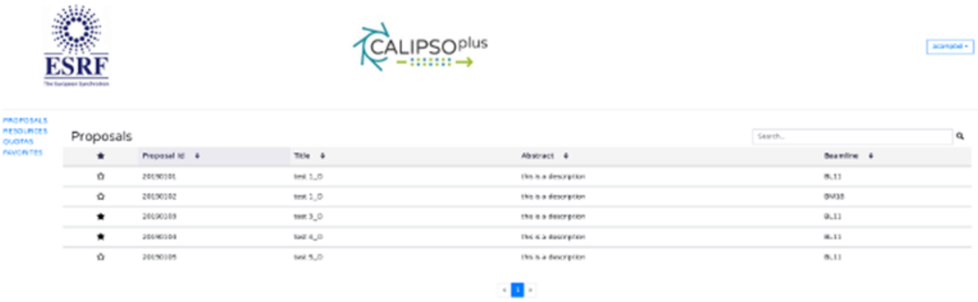
Form Value: { "publicName": "", "image": "", "protocol": "", "cpu": "", "memory": "", "hdd": "", "resource": "" }

base_image
image: ccesid/centos-xfce-vnclarad
Protocol: vnc
CPU: 1
RAM: 2G
HDD: 50

base_jupyter
image: jupyter/scipy-notebook
Protocol: vnc
CPU: 1
RAM: 2G
HDD: 50

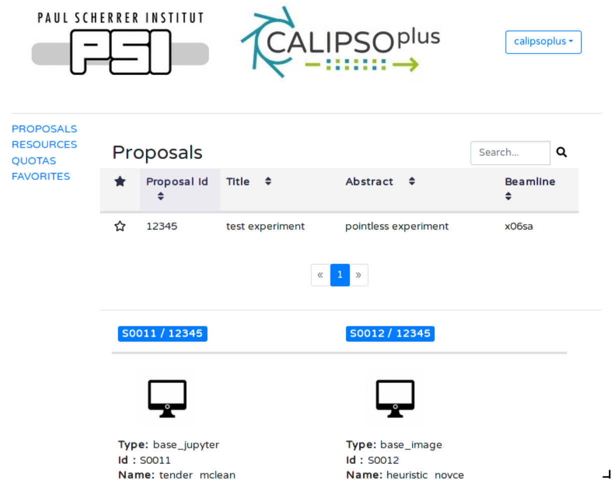
vnc-crispy
image: acampbal/vnc-crispy
Protocol: RDP
CPU: 1
RAM: 2G
HDD: 50

Figure 5 DAAS portal @ ESRF



Proposals

★	Proposal Id	Title	Abstract	Beamline
☆	20190101	test 1_0	this is a description	BL11
☆	20190102	test 1_0	this is a description	BM38
★	20190103	test 3_0	this is a description	BL11
★	20190104	test 4_0	this is a description	BL11
☆	20190105	test 5_0	this is a description	BL11



Proposals

Search...

★	Proposal Id	Title	Abstract	Beamline
☆	12345	test experiment	pointless experiment	x06sa

« 1 »

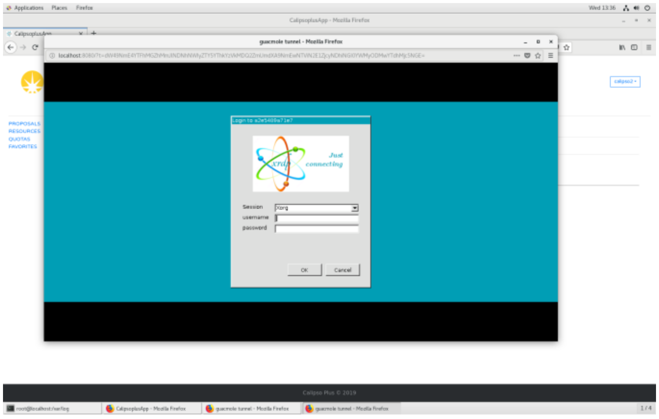
S0011 / 12345

S0012 / 12345

Type: base_jupyter
Id : S0011
Name: tender mclean

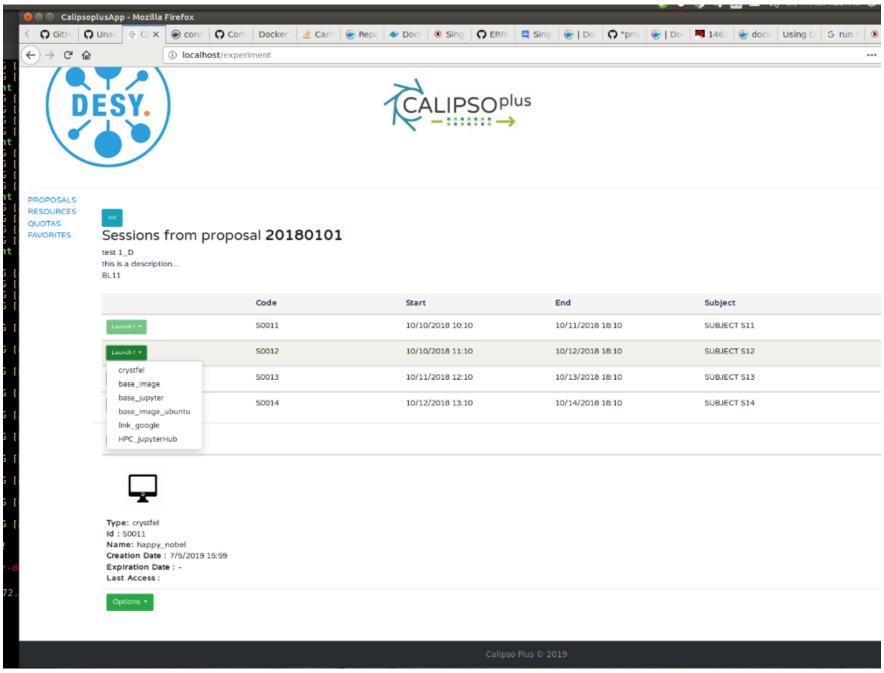
Type: base_image
Id : S0012
Name: heuristic novce

Figure 6 DAAS portal @ PSI



Applications
Places
Firefox
Wed 13:36
CalipsoplusApp - Mozilla Firefox
guacmole tunnel - Mozilla Firefox
localhost:8080/?t=dW49NmE4YTFhMGZhMmJlNDNhNWIyZTY5YThkYzVkMDQ2ZmUmdXA9NmEwNTVlN2E1ZjcyNDhiNGI0YWMyODMwYTdhMjc5NGE=
calipso2 ▾
PROPOSALS
RESOURCES
QUOTAS
FAVORITES
Login to a2e5409a71e7
xrdp
Just connecting
Session
Xorg
username
password
OK
Cancel
Calipso Plus © 2019
root@localhost:/var/log
CalipsoplusApp - Mozilla Firefox
guacmole tunnel - Mozilla Firefox
guacmole tunnel - Mozilla Firefox
1 / 4

Figure 7 DAAS portal @ DIAMOND



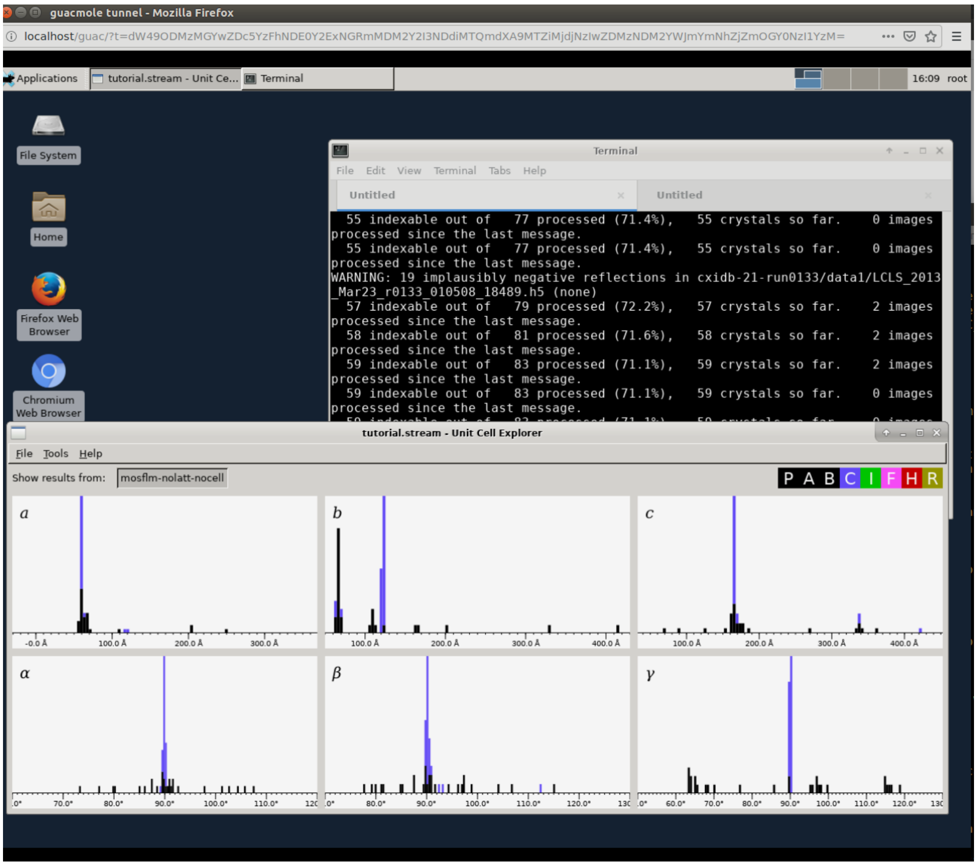
Sessions from proposal **20180101**

test 1_D
this is a description...
BL11

	Code	Start	End	Subject
Launch ! ▾	S0011	10/10/2018 10:10	10/11/2018 18:10	SUBJECT S11
Launch ! ▾	S0012	10/10/2018 11:10	10/12/2018 18:10	SUBJECT S12
	S0013	10/11/2018 12:10	10/13/2018 18:10	SUBJECT S13
	S0014	10/12/2018 13:10	10/14/2018 18:10	SUBJECT S14

Type: crystfel
Id : S0011
Name: happy_nobel
Creation Date : 7/5/2019 15:59
Expiration Date : -
Last Access :

Figure 8 DAAS portal @ DESY


guacmole tunnel - Mozilla Firefox
localhost/guac/?t=dW49ODMzMGYwZDc5YzFhNDE0Y2ExNGRmMDM2Y2I3NDdiMTQmdXA9MTZiMjdjNzIwZDMzNDM2YWJmYmNhZjZmOGY0NzI1YzM=
Applications
tutorial.stream - Unit Ce...
Terminal
16:09 root
File System
Home
Firefox Web Browser
Chromium Web Browser
Terminal
File Edit View Terminal Tabs Help
Untitled
Untitled
tutorial.stream - Unit Cell Explorer
File Tools Help
Show results from:
mosflm-nolatt-nocell
P A B C I F H R
a
b
c
α
β
γ
100.0 Å
200.0 Å
300.0 Å
400.0 Å
70.0°
80.0°
90.0°
100.0°
110.0°
120.0°

```
  55 indexable out of   77 processed (71.4%),   55 crystals so far.    0 images
processed since the last message.
  55 indexable out of   77 processed (71.4%),   55 crystals so far.    0 images
processed since the last message.
WARNING: 19 implausibly negative reflections in cxidb-21-run0133/data1/LCLS_2013
_Mar23_r0133_010508_18489.h5 (none)
  57 indexable out of   79 processed (72.2%),   57 crystals so far.    2 images
processed since the last message.
  58 indexable out of   81 processed (71.6%),   58 crystals so far.    2 images
processed since the last message.
  59 indexable out of   83 processed (71.1%),   59 crystals so far.    2 images
processed since the last message.
  59 indexable out of   83 processed (71.1%),   59 crystals so far.    0 images
processed since the last message.
```

5. Feedback on Functional and Non-Functional Testing

Question A.1. Please state technical issues and observations concerning the following components: Portal back-end, Portal front-end, Data browsing, Jupyter, Data analysis Application containers (PyMCA,CrysFEL,Savu, Ptycho,pyFAI (update list)), Guacamole / VNC

ALBA	*The portal front-end and back-end were deployed with no major issues, although the front-end was easier to have it running, as the back-end needed more integration from the differents local services (eg. Users and Experiments/Proposals from User Office Portal).* *Guacamole was successfully deployed but no performance tests have been done so far with it, so we are not sure about its capabilities to run intensive applications.* *Umbrella was also successfully installed. Having the User Office already integrated helped on boosting the configuration for this project.*
ELETTRA	*The portal front-end was deployed easily and has a clear interface. It's responsive and easily customisable but there are still new features added in the latest versions. The data browsing is what expected from basic data browsing on a portal with some context menus that may not be clear enough. We successfully deployed Guacamole but we found its performance and features inferior to other remote desktop solutions (ie. NX). We also have Jupyter which is a standard Jupyter interface. Finally we deployed a full featured PyMCA v.5.4.3 but its container (created by Elettra) was rather large and not optimised.*
ESRF	*For the front-end, we were never able to successfully get the remote desktop viewer (Guacamole) to work. We had to overcome this by modifying the code and inserting our in browser remote desktop viewer (Guacamole too). By doing this, we also removed the Guacamole database as a necessity.* *For the software such as PyMCA and Crispy, they worked perfectly as intended. The scientist noticed that the latency was very small and the container was perfectly usable to interact with the software. Note that this was tested only using RDP and not VNC. At present, we have no intentions to use VNC as RDP is supposed to give a superior performance but more benchmarking from us is needed.* *We didn't use the Jupyter Notebook containers already provided in the portal as we had a Jupyterhub instance running on a Kubernetes cluster. We were able to modify the code to redirect to the Jupyterhub instance rather than creating a new container which worked just as expected.* *At the beginning, the containers running the software didn't support our UID and GID which meant we couldn't mount the home directories or the data catalogue. We were able to find the solution to this and submitted the code to the repository.* *As of today, the data browsing does not work as we are using iCAT which is not supported by the portal just yet. We are working on creating a plugin which supports iCAT but this is not yet ready. We should be able to provide the same response expected which can be added to the experiment table.*

SOLEIL	*The tests are not yet completed. Here are our comments based on what we have seen:* *- Portal front-end is working flawlessly* *- Jupyter was not tested* *- Data Analysis containers were not tested* *- Guacamole/VNC was tested and is running well.*
PSI	*Backend:* *Once it was running, the backend worked well. Deployment would have been aided by more extensive documentation and more detailed logging.* *It should be noted that we currently run the backend in debug mode with local authentication using calipso-local-login-mock.* *Frontend:* *As for the backend, deployment would have been easier with better documentation and more detailed logging. One particular issue we ran into was that at least one of the URLs was sensitive regarding trailing slashes, ie 'http://.../' vs 'http://...'.* *We also noticed that the frontend is pulling files directly from external systems, eg. googleapis.com, which may not be reachable from clients.* *Data Browsing:* *We have not tested this.* *Jupyter:* *We can launch and connect to Jupyter instances using the frontend.* *Data Analysis Application Containers:* *Due to time constraints we did not try any of these.* *Guacamole/VNC:* *On its own Guacamole works, but using it from the frontend does not. When trying to connect to a container, a new browser window opens with the Guacamole login form. Authentication then fails. This is probably a misconfiguration, but from the documentation it is not clear what might be missing.*

DIAMOND

- *Portal back-end:*
  - *Instead of downloading the Docker images with the latest tag, all tags for that image were downloaded (see https://github.com/Calipsoplus/calipsoplus-backend/pull/59)*
  - *I also noticed that if the container is stopped through other means than the resources page (e.g. docker container stop), the container remains marked as running on the webpage, and issuing stop results in:*

    ```
    [10/Apr/2019 12:10:12] DEBUG [apprest.services.resources:54]
    stop_resource (mystifying_diffie)
    [10/Apr/2019 12:10:12] DEBUG [apprest.services.resource_docker:202]
    CalipsoResourceDockerContainerService stop_resource
    (mystifying_diffie)
    [10/Apr/2019 12:10:12] ERROR [apprest.services.resource_docker:221]
    Stop
     container error
    [10/Apr/2019 12:10:12] ERROR [apprest.services.resource_docker:222]
    404 Client Error: Not Found ("No such container: mystifying_diffie")
    [10/Apr/2019 12:10:12] DEBUG [apprest.views.resource:84] 404 Client
    Error:
    Not Found ("No such container: mystifying_diffie")
    ```
  - *Whenever I log out while the Docker images are still downloading, I noticed that I cannot use these afterwards when attempting to launch as I get the error:*

    ```
    [11/Apr/2019 06:59:00] DEBUG [apprest.services.quota:84] container
    with public_name=base_jupyter
    [11/Apr/2019 06:59:00] DEBUG [apprest.views.resource:119] Error after
    run_container : Resource already launched
    ```

    Even though the container isn't actually running.

- *Portal front-end: no comments*
- *Data browsing: this was not tested as the DIAMOND data mounts were not accessible from the VM.*
- *Jupyter: this was only briefly tested. It appears to work, but had problems with the Docker image as all tags of jupyter/scipy-notebook were being downloaded, instead of just latest, and resulted in my hard drive slowly filling up…*
- *Data analysis application containers: these were not tested*
- *Guacamole/VNC/RDP: this was tested with the Ubuntu image and was found to be working well, though getting guacamole properly configured was tricky.*

DESY

Portal back-end:
Our preferred environment is based on openstack, kubernetes scalable application deployment. Attempts to deploy the back-end in this environment were not successful. The demo deployment of the portal is lacking some functionalities required for kubernetes deployment, so we finally had to revert to the docker-compose based tests. The portal back-end could readily be deployed using the docker-compose based demo installation.

Portal front-end:
The front-end was deployed as tested without major problems. Due to security constraints we would need to run the front-end behind a suitable proxy like nginx. This

was not successful so far. The documentation could be improved to guide proxied front-end deployment.

Use cases:
We tested the singularity images for crystfel, pymca, savu on the back-end without major issues. ptycho-shelves was tested outside the back-end (lacking access to the license server due to security reasons).

Jupyter:
So far we did not try the jupyter server from the portal. Launching a notebook server directly from the backend is however always possible. We are running JupyterHubs to access different compute infrastructures like a HPC cluster and the openstack/kubernetes environment. Preferred mode would be to launch the back-end plus notebook from the JupyterHub rather than the other way round. This would however require the kubernetes integration, which hasn't been completed yet.

Guacamole/VNC/RDP:
Guacamole was very briefly tested. The solution appears suitable for "non-demanding" users/use cases. Some applications require however GPU hardware acceleration. For such cases we use a commercial RDP based solution (starnet fastx) which also offers an extensible API. Integration of this solution has not yet been tested.

Question A.2. Comment on the available APIs resulting to the integration of the individual components to a complete system. These may include: Local authentication APIs (local,Umbrella), APIs Authorisation / Privileges, Data Access / Local data provider, General observations

ALBA

Local authentication does not look strong enough to be used in a production environment. Other authentication strategies shall be put in place before going live.

Authorization depends on the information retrieved from external services (e.g Data Catalogue API or User Office Experiments retrieving given a user). As ALBA doesn't not provide any Data Catalogue yet.

ELETTRA

The available APIs are sufficient to cover the integration of all the components of the system. Certain of them are not well document and miss functions. The Data Catalog and AA APIs should be clearer for wider deployments. The consortium was very responsive and added the required functionality for our facility (eg. Owner's UserID binded to datasets)

ESRF

The initial APIs available were not usable for the ESRF. The local authentication method was completely incompatible with our systems as we didn't support how it was implemented. We had to add OpenID Connect to the portal to handle authentication. By

	doing this, we also added Umbrella authentication too which then maps to our ESRF accounts. The Umbrella URL provided in the code was not used. The privileges such as admin/staff or user in the portal were not mappable to our systems. We therefore have to do all of that manually or else update our Keycloak configuration to support this. These features were used for creating a single container “not related to an experiment” and for accessing the admin section. The data provider was also unusable as iCAT returns the data in a completely different format to what was available. We also do not agree with duplication of data whereby we have our data catalog stored in a database at the ESRF and then almost an exact duplication for the Calipsoplus project. General observations: The initial portal design (ALBA) did not take into account the differences between facilities such as authentication, access to data, access to computing resources etc which made initial setup very difficult. We have been adding to the portal some methods to support other sites and making the APIs as general as possible. In the case of any facility using OpenID Connect, they will immediately be able to login by inserting some values into the configuration file which they can get from Keycloak. This will require almost no configuration on their end part from authorising the portal to connect to Keycloak. We hope to have a similar result with the iCAT plugin.,
SOLEIL	We only used the local authentication and authorization (calipsoplus-local-login-mock). We didn’t have the opportunity to test Umbrella and OpenID (tests are in progress). However the available APIs are well designed and sufficient for integration in our systems.
PSI	We only implemented local authentication and did not use or review any of the APIs.
DIAMOND	Only local authentication and authorization was attempted using the local login mock Django backend (https://github.com/Calipsoplus/calipsoplus-local-login-mock). For this to work, I had to modify CalipsoExperimentsServices.get_external_is_authorized to not send authentication parameters along with the request, as my dummy endpoint was not written to deal with that. Umbrella and OpenID were not tested.
DESY	We only tested local authentication for the demo and did not use or review any of the APIs. The openstack/kubernetes environment support the EOSC EGI OpenID connect solution, which also supports UmbrellaID as one of the EduGain partners, so we foresee to support this solution in the local portal deployment.

Question A.3. Does the resulting integration (as deployed in your site) complie with the general requirements as described in JRA2 (D24.2 Blueprint)? Please comment on the degree and the main points for improvement.

ALBA

Although the portal is still under development, other implementations are expected (such as Kubernetes integration) to be used at ALBA. As a first step, using Docker seems quite convenient, as it is perceived as the simplest solution whilst the right infrastructure is put in place (Kubernetes or OpenStack) at ALBA.

ELETTRA

Our current deployment confirms that at an architectural level the system complies with the D24.2 Blueprint. There is still though work required for better data access and easier inclusion of new applications/containers.

ESRF

The blueprint and implementation are very close but some details such as a common authentication portal seem to have been added directly to the portal rather than connect to it.

The biggest implementation problems are the differences in security at our facility as well as the database/way that we store and access our data.

Unfortunately, every facility is going to experience this problem and we will have to work together to overcome it.

We need to make the APIs as general as possible to support this and remove the need for facilities to change their methods to fit the portal. This will likely result in the portal becoming a little "heavier" but this is needed for reliability and allowing for integration with more systems.

SOLEIL

The deployment has only been tested on a test server but it seems that the project is built around proven and functional technologies. The only downside is the documentation which is not detailed enough.

PSI

We could not integrate our CALIPSOplus instance with our existing systems yet.

DIAMOND

No deployment has been attempted at this point, only testing within a VM. Considerable effort will be required for integration within the Diamond infrastructure.

DESY

So far only the demo deployment has been tested successfully. Integration into the DESY cloud infrastructure and access to experimental data has not yet been achieved.

Question A.4. Please comment on how ready is the system for delivery. According to your experience what is necessary to achieve an even wider cross-facility deployment? What were its strengths and weaknesses? How was the collaboration and deployment assistance? Were platforms like Slack useful in the deployment phase? What is the state of documentation?

ALBA	*README.md document is not complete yet, and needs to be updated. Other facilities have given useful feedback in order to improve the information provided in it.*
ELETTRA	*The system is still on development so we don't consider it ready for wider cross-facility deployment. The are many components and APIs in non-stable versions. The effort for integrating it with the existing facility databases (eg. VUO) is substantial. Nevertheless this is a necessary step for improving this great system. The system has limited documentation. The Slack has been very active and a crucial tool for the development and deployment.*
ESRF	*We feel that the system has a good foundation but it is nowhere close to being ready for deployment facility-wide or cross-facility wide.* *Each facility is responsible for making sure that it is able to make use of their resources such as infrastructure but at the ESRF, we are not yet ready to support something like this on a large scale.* *At present, the containers are only created on a single machine which makes scaling impossible. Our goal is to add Kubernetes support which will allow us to spread the containers among multiple machines. This is to remove the single failing point and allow us to support more containers simultaneously.* *There are also more features that we want to add to the portal such as virtual machines which was defined in the blueprint but has not yet been implemented.* *The collaboration was shakey at the beginning in terms of adding new features, changing existing code etc. These issues seem to be mostly resolved but we need to be open to making the portal a little heavier rather than going for the leanest implementation in order to support more than one facility.* *Slack was vital in addressing any problems with both development and deployment as email was generally too slow. Slack allowed for group discussions in almost real-time, ability to share code snippets, errors, images etc.*
SOLEIL	*We should check that it does not contain any security vulnerabilities but from our point of view, the system is ready for system delivery. This project is a big step forward for us. This kind of service is really a key feature for our users. Slack is the only source of information since the documentation is poor (as mentioned in the previous point). The collaboration is very helpful.*
PSI	*We believe that there is still a significant amount of work to be done:*

- *The documentation needs to be more detailed and cover more topics, eg. operations and security.*
- *It would be good to have command-line tools and monitoring plugins/interfaces to query the state of the system.*
- *Logging should be more detailed.*
- *It is not clear how to integrate existing storage, especially in a secure manner.*
- *Support for other compute backends (eg Kubernetes).*

Slack was very useful, especially thanks to the archived discussions in #test-setups.

DIAMOND

During the short testing at DIAMOND, several bugs were encountered that need fixing (see 1.) before delivery can be considered. However he project already looks quite formidable and will certainly be of great benefit to users across the participating facilities. Documentation appears to be lagging behind the development and some additional effort appears to be necessary. Slack has been extremely useful during testing, as I would not have managed to get the portal up and running without the help of Aidan, Daniel and Alex, to who I am deeply indebted.

DESY

The documentation needs additions and clarification. It would in particular be very helpful to elaborate on security issues like reverse proxies, non-standard deployments like kubernetes. We haven't tried integration of (remote) data storage infrastructures. Best practices how to achieve that in a secure manner would be helpful.

Question A.5. Report on performance specific observations. State deployment setup specs (OS, HW, Network, versions). The observations may include: general responsiveness, I/O if available, stability, Guacamole, and data analysis software.

ALBA

No performance tests have been done yet.

ELETTRA

Our deployment was on KVM virtual machines running on our local cluster (CentOS 7) over 40Gbps Ethernet for both intecommunication and shared filesystem (Ceph). The overall system felt responsive. No quantitative I/O tests have been performed. Guacamole seemed not good/performant enough to replace standard VNC clients (over VPN) or NX. The analysis software (PyMCA container) had slightly reduced interaction performance due to remote graphics.

ESRF

No performance tests have been carried out.

SOLEIL

We cannot evaluate the performance since the deployment is only on a test server and not fully completed.

PSI

Our setup is still experimental, so we cannot offer any observations regarding performance.

DIAMOND

Testing was done on a Centos 7 VirtualBox VM that runs on an iMac host system. Docker was installed on the host instead of the guest, which turned out to work fine, as the guest has access to all ports exposed by the host.

Performance in general was not good, due to limited number of resources allocated to the guest, but this cannot really be considered a good indicator as the deployment will eventually occur on a far more powerful setup.

DESY

No performance tests have been done yet.

Question A.6. Discuss security related issues that cover login (authentication & authorisation), Umbrella topics and general concerns.

ALBA

Authentication on local facilities is still to be standardized, although there are already interesting solutions in place. In ALBA's case, the Umbrella implementation was successfully done, but allowing local authentication as well, which uses authentication against the User Office portal.

In order to make it fully secure, local authentication should be disabled. The counterpart of this solution is forcing all our users to have an Umbrella account, which is also an objective, but may cause some resistance to this change.

ELETTRA

We didn't manage to use Umbrella even if it is supported as a login from our user office VUO. A general issue is that the beamline scientists and personnel with elevated user rights, connecting from public computers anywhere on the internet, will be using a web interface that will be granting them access to systems that are traditionally accessed from local facility computers that tend to be well maintained/patched and behind the facility's firewall.

ESRF

The security of the portal is a large concern for our infrastructure team who manage authentication as well as access to other systems such as our data and creating container/virtual machines on our hardware.

The security of the portal was/is extremely lacking as we do not support authentication by sending POST requests with the username and password without a secure connection. There are also numerous potential problems such as storing the username in an unencrypted local storage which could be changed by a user and a large duplication of data which is sensitive.

The portal has a huge duplication of sensitive data issue that one development team is very supportive of while the other development team strongly disagrees.
Regarding security, the UID and GID of a user who has carried out an experiment is stored in a database which seems to be unsalted and unencrypted. If anyone including an admin was able to gain access to these values, they would be able to access any files stored on NFS by that user as well as fully impersonate them as a docker container could be run as that user thus tricking the system into who owns the process.
The other problem is that there is a large duplication of data such as data on experiments (not experiment data) and all of this is unencrypted/unsalted in a database.

SOLEIL	*We didn't have the time to setup the authentication against the SOLEIL LDAP. We are working on this particular point. Umbrella was not tested either.*

PSI	*Due to time constraints we could only implement local authentication.*

DIAMOND	*Authentication and authorization was only tested using the mock login portal. We do plan to set up authentication using Diamonds own systems.*

DESY	*Authentication and authorization was only tested using the mock login portal. We plan to use the EOSC EGI OpenID connect solution for the portal.*

Question A.7. Involve one or more beamline scientists or data analysis personnel and report their experience with the system. This should include general observations but also specific comments on the use of data analysis applications with real data. State the name, field/beamline and application used.

ALBA	*V., ALBA's Data Management project leader:* ● *The portal is very attractive and the user experience is good as well.* ● *The login page should be improved in order to make it more user friendly.* ● *Current images, using Desktop interfaces, allow ssh connection to any of the machines within ALBA's network. This may not be desirable, or at least it should be controlled.* ● *Resources should be monitored and controlled, not only at the individual user level, but also at the portal level. If the number of users connected reaches a certain level, new users should be added to a certain queue in order to avoid any service overload.*

- *Read-only mounting of the data directory certainly looks like the right approach. Apparently, there are limitations to do that, and the root directory of the data is mounted. That means that all folders are visible but not accessible, except for the experiment we want access to, which is indeed readable.*
- *Read-write mounting for the results of the data analysis should be put somewhere it can be reachable by any of the machines that are launched. If the results of the analysis can be used for by another machine, that should be possible.*
- *Having a Beamline-specific Docker container seems to make sense in ALBA's case. As a data analyst, I'd want to use the same machine to process data and use one software package after another.*
- *Own Resources (machines with no Experiments attached) is very interesting for us. After very few tests with small data, the results are very promising.*

S., IT SysAdmin:

- *Although this is not a high priority being this portal a demonstrator, it has come to my knowledge the risk of allowing a ssh connection from any of the Docker containers to any of the internal machines. It would be recommended to put the container host behind a firewall and only allow communications to the indispensable servers that the service needs.*
- *Access to the Docker Daemon should require authentication or at the very least be firewalled so only the CALIPSOplus machine can launch new containers.*
- *Ideally the dockerd process should not run as root, although that may not be possible with the current Docker versions.*
- *For scaling and being able to run much more containers than the current host allows, longer term we should probably move the running of containers to a Kubernetes cluster.*
- *It would be very desirable to devise a better way to manage the passwords for the proposal accounts (u2xxxxxxx). For instance, instead of having default passwords, the User Office could a assign a random password to each proposal that would be picked up by our scripts when creating the corresponding LDAP account. Another long-requested feature by the Beamline scientists would be the ability reset proposal passwords from the User Office portal, or even better that the main proposer could reset it by himself.*
- *I was pleasantly surprised to see how fast it is to start a container for the platform and to be able to log in (in 5 seconds I already had a desktop), and also that, as much as you already know Guacamole and know what it does, the fact of being able to launch a remote desktop for HTML5 without having to install any client is still nice*

E., Beamline scientist:

- *I've found to portal very attractive and responsive and, having this kind of tool, it feels very useful to our users.*
- *I couldn't get my real data because the file system of my Beamline wasn't mounted, but I would be willing to use this tool and test the software packages we usually provide (pyFAI, Dawn, Origin, SASView, SASFit, Fit2D, ATSAS, Matlab, BornAgain,...)*
- *Providing a machine with only Python installed, and some libraries (such as numpy) would also be interesting to us.*

C., Scientific Software Group Leader:

- *The general impression is quite good, considering that this is a PoC.*
- *The login page is misleading with the relative location of the logos and text.*
- *As already pointed out, the access to the original data in the proposal should be simpler from the application containers:the user should see a folder called "data" (or something similar) in its own home or in the root, and this folder should only contain the user proposal.*
- *I like the current approach of having one image per application ("microservice-like approach"). I prefer it over the other two possible alternatives (i.e. having a single image with "everything" in it, or building a custom image with a user-selected list of programs) because:*
    - *The microservice approach is easier to extend (add new applications),*
    - *The effort of defining and maintaining the images can be shared more easily (it does not need coordination).*
    - *It even opens the door to accept contributed images from users/groups of users, where they provide specific tools.*
- *But for this "microservice" approach to work it is crucial to have a single shared writable volume for processed data.(i.e. The volume of the original data from the proposal can -and should- be kept read-only, but the volume of "processed data" must be the same for all the containers, and must persist along sessions.*

G., Controls System Section Head:

- *The meaning of "Own Resources" is unclear to me. Perhaps there should be a clearer explanation of its purpose.*
- *Regarding the "running" Resources area, the information is not updated automatically. It works well when you start (it appears) and when you stop it (it disappears), but not when you close an access (i.e. the "Last access" field is not updated). It does so when you switch views but not if you stay on the Resources view.*
- *Also, I think it would be really useful to let the user comment on that "Running resources state". I guess that the reason why it is not stopped (i.e. "Pending to analyze feature X on dataset Y") may be key to access between N resources that the scientist may have left running.*
- *When launching a resource, you have to go to that specific resource, and select "Enter" from the options drop-down menu. I think that this second action could be skipped and automatically show the remote window when launching.*
- *I'd appreciate if in addition to the card-view when visualizing the running resources, having a table-view would also be appreciated.*
- *In the Quotas view, I see there are some machines available. Nevertheless, I cannot start another resources while I have one running. This could be also allowed.*
- *Input and output data names are not very intuitive for the mounted folders.*

ELETTRA

AC, beamline TwinMic, head of beamline: The portal looks nice. Accessing the data from home is very useful and I didn't have to use the usual VPN+VNC. PyMCA was a bit slow and I had difficulty in browsing the data. I haven't used Jupyter in the past but it looks like a good alternative to a Matlab tool I've been using. I would like also to be able to easily add other analysis apps on this system.

	FB, Scientific Computing, leader: The portal was responsive. I didn't find a link on how to create an account but someone else did it for me. I could log in and browse my data. I would had appreciated a terminal/console but there was not available. The Jupyter I used was responsive and I got the expected behaviour from it. Overall it is a nice system.
ESRF	*"As it is now the portal provides all that is required to run a virtual machine for Crispy. There are however a few things that could be changed to improve the user experience. The page displayed following the login could make better use of the available space. Instead of presenting a combo box with a list of the available virtual machines, it would be easier to have big buttons with the name or a screenshot of the machines arranged in the center of the page. If I am not mistaken such an arrangement is used by ILL in their Guacamole portal.* *Once the portal is connected to the data storage, all proposals of the current user should be accessible, as it is very common to need data from different proposals. The mounting point should be linked to the desktop under an obvious folder name like "Proposals".* *Regarding specifically the Docker image for Crispy, after the image is started the login to the VNC server should be done in the background, and if possible Crispy should be started automatically. The path to Quanty should be set automatically and the installation of the Python related packages should be done using apt package manager. These issues are all solved in the Docker image I've made for Crispy https://github.com/mretegan/dockerfile-crispy/blob/master/Dockerfile.* *I've only briefly tested the Jupyter notebook, but it worked as expected."* *(MR, ESRF, Scientist)*
SOLEIL	*We plan to involve scientists in few weeks when the deployment is properly completed.*
PSI	*Due to time constraints and the limitations of our setup we could not involve a beamline scientist.*
DIAMOND	*Due to lack of time and small scale of the test no beamline staff or data analysis personnel could be involved in testing.* *Also the DIAMOND file shares could not be mounted into the Docker images during this test.*
DESY	*Our setup was not mature enough to allow testing by beamline staff or external users.*

Question A.8. Comment on compatibility and general comparison to the existing/classical workflow of data analysis. Advantages and disadvantages.

ALBA	*It certainly makes sense to provide Dockerfile recipes containing all software packages needed for a certain Beamline. It reflects pretty much the current use of the data analysis service provided at ALBA. This approach would facilitate the use for the scientists, but may not be very efficient in term of resources optimization.* *On the other side, having one image for each software package would be optimal, but would require integration implementations in order to allow scientists to connect outputs from one sides to inputs from another.*
ELETTRA	*Our deployment relies on well tested components including PyMCA and Jupyter accessing real data stored in our facility's storage using existing user credentials. This ensured absolute compatibility. The classical workflow requires login in different systems through ssh and various portals. This system allowed through a single sign it to its portal to launch various services. On the other hand there is no option for multiple tools (ie. only one way to browse the data). The remote desktop solution is also lacking in terms of performance.*
ESRF	*It is a great start to providing this service but there is still a lot of work to do. We also have to decide how to best structure the container images. We could create many containers with only a few software packages installed or we could create a container for a specific beamline/scientific method e.g spectroscopy*
SOLEIL	*The big advantage for our users is that they don't have to copy the data and they don't have to install data analysis program on their computer. We already have automatic workflows that provide data analysis results, this system is really efficient since it's running on our HPC but is not interactive like Calipsoplus where the user can choose what to do.*
PSI	*Due to time constraints and the limitations of our setup we could not involve a beamline scientist.*
DIAMOND	*Any comparison will depend to a large extent on the amount of data that has been collected during an experiment. If the amount is small (couple of GBs) then the users will not be able to benefit much from this service as it will be easier for them to analyze the data at their home institutes. However if the amount of data is too large to copy home, then this portal will become an essential tool to analyze it, and will become an appropriate alternative to the NX remote session capabilities we currently offer to users.*

DESY

The main advantage would be the availability of identical tools at different facilities for occasional users with moderate compute requirements without the need to register at each of the facilities.
For users with very high compute demands we will presumably require local registration and authorization. For such users and users with the need for GPU hardware acceleration the existing system will provide a more suitable environment. The back-end would however be available in any case providing harmonized environments.

Question A.9. Report any additional Testing related observations that were not covered from the previous questions.

ALBA

pyMCA was tested within ALBA's network (not real remote access, although using wi-fi and not eth) and no much difference perceived between the Guacamole interface or any remote access, when describing the ease of use and performance.

Load tests should be done in order to identify limitations of the system and how to control load and mitigate any service interruption.

ELETTRA

There was a substantial amount of work required to interface the APIs with the local resources and databases (both for login and data access). This work was not taken into serious consideration well in advance and was underestimated.

ESRF

The user experience could be greatly improved as the pages the user sees are not always clear and it is difficult to know how to create a container or access Jupyter Notebooks if there is no data present.
One user would like to see a page which lists all of the services available e.g Container, virtual machines, Jupyterhub and the experiment page.

SOLEIL

None

PSI

None

DIAMOND

None

DESY	*None*

6. General Comments, Observations and Suggestions

Question B.1 Installation - report on the installation procedure and the difficulties encountered, required developments, and other issues.

ALBA

Being part of the development team, the installation was done with no major problems besides the inherent complexity of having 5 components to take care of: Back-End, Front-End, Docker Server, Guacamole Server and Umbrella.
Documentation on Github shall be updated while receiving feedback from other facilities that are deploying the portal.

We were able to use basic Docker images very fast, but passing user id's and groups id's was not so trivial when we wanted to mount our facility storage, and grant permissions to specific users.

ELETTRA

The installation process was straightforward for our infrastructure. However the were missing parts in the documentation that resulted in delays and required communication and interaction with the developers.

ESRF

Overall the portal was deployed with some difficulty but this was mostly due to our network configuration as well as some of our systems and the need to use MySQL databases. We were able to overcome most of these difficulties by changing the databases to SQLite and storing them locally instead of on a different server. We also had to use the mock authentication server at the beginning as we didn't support the local authentication method provided in the portal with our authentication systems.
We never got the Guacamole application working within the portal. This was mostly due to lack of documentation on where to download the Apache Guacamole code and how to install it.
We able to get another version of Guacamole running entirely in the browser without using Apache's interface and with more features. By doing this, we were also able to remove the Guacamole database entirely.
The Jupyter Notebooks provided as a Docker container were easy to deploy as they didn't require Guacamole and worked immediately.
After some time, we were able to get the portal installed and almost fully working. We also added some documentation to the Github page for any difficulties that we encountered.

SOLEIL

Installation is not really easy. The documentation is basic and a lot of information is missing. However the community is very responsive and helpful.

PSI

Running the backend on RHEL 7 failed due to problems with the versions of *different components (eg the Python MySQL library and the libmysql included with RHEL). In the end we deployed the backend in a Fedora 29 container running on a RHEL7 host.*

The frontend only works when using http://.../login/. Otherwise only the CALIPSOplus logo is displayed.

DIAMOND

Installation was rather cumbersome as documentation was rather limited on how the dependencies had to be installed. A particularly difficult one was guacamole and considerable effort and time was spent on getting a working configuration.

DESY

The deployment in our kubernetes environment was not successful, partially due to the somewhat limited documentation.

Question B.2 Integration - how the DAAS was integrated with your facility's infrastructure (e.g. accessing data, users accounts, authentication, local hardware)

ALBA

Storage user accounts at ALBA, once the proposal is accepted, are not organized per person but per experiment. Integration in this case had to take into account this fact and, while the individual authentication is done on the Portal (using the User Office Portal Database), the user/proposal mapping had to be prepared in order to mount volumes with the right group (proposal) user.

At ALBA there is still no Data Catalogue in place, even though there is an ongoing project which aims to have it ready by the end of 2019. The endpoint that retrieves information about the proposal is currently getting the data from a User Office portal web service. Once the Data Catalogue will be in place, changing the endpoint will be strongly considered.

ELETTRA

The integration was challenging. Even after a successful installation the system was lacking the necessary integration with local resources and services like Authentication, Authorisation, and access to the data catalogue. It was necessary to develop dedicated web services for our digital user office (Elettra VUO).

ESRF

Initially very difficult to integrate with ESRF infrastructure.
At the time, the portal only supported Umbrella and login through POST requests. As Umbrella didn't work with some of our internal platforms and we do not support POST requests for security reasons, we had to add OpenID Connect to the portal.
By handling authentication with OpenID Connect, the user instantly had full access to all of our internal platforms.

We also added the code and documentation for each facility on how to enable OpenID Connect if they support it. Many of the facilities that use Keycloak for authentication will be able to make use of this.
We still do not have integration between the portal and our Datahub(ICAT) but we are in the process of writing a plugin which will interact with the REST API. The current portal implementation does not work due to different fields, JSON responses etc.
Until recently, we were not able to access our home directories on the containers due to the UID and GID being different to our NFS systems. We have been able to fix this and added the code to the repository for the other facilities to use.
We are able to create containers using Docker but we are still not able to create them on our Kubernetes cluster. We are in the process of adding this functionality to the portal.
Despite some of the limitations, we were able to get the portal fully integrated with our systems and we have a working version of the portal.

SOLEIL

We didn't have the time to test the integration yet. Accessing the data should not be a big problem, however we need to work on the authentication and implement the proposed API.

PSI

We have not yet integrated the software with our infrastructure.

DIAMOND

This was not done: deployment was limited to a single VM, running on an iMac that was attached to the visitor network.

DESY

Currently we don't have any integration with existing infrastructures or services.

Question B.3 Deployment - results on how this was deployed and issues encountered

ALBA

As part of the Demonstrator development team, no major issues were encountered during deployment, as the procedure was very well known by the team, and has been even automatized using Continuous Integration and Deployment tools, such as Jenkins, when new developments have been integrated.

ELETTRA

The deployment followed a rolling model including additional developments. This test reports on our deployment as of 10/04/2019 but our facility will continue improving its deployment and integrating new features (e.g. Umbrella).

ESRF

The portal has only been deployed on a development machine as we are still waiting on one of our sysadmins to allow OpenID Connect to work on a certain virtual machine.

We have the portal deployed on a development machine and a virtual machine acting as a Docker hypervisor for all of the containers.

Our hope is to migrate this from the development machine to a virtual machine which will allow other members of the team to access it and test it more frequently.

For the Kubernetes cluster running the Jupyterhub instance, we created five virtual machines on an OpenStack cluster and installed Kubernetes on those.

SOLEIL

As mentioned above, the project has only been deployed on a test server.

PSI

We have not yet integrated the software with our infrastructure.

DIAMOND

See 2.

DESY

As mentioned we were limited to the demo test deployment without further integration. Within this deployment we didn't see major problems on the back-end.

Question B.4 Use cases - results for the tested use cases in your facility (in conjunction with previous Question A.7)

ALBA

Docker images adapted on use cases were not easily testable until the very end, and we finally used another version using an Ubuntu distribution, which turned out to be simpler to test.

ELETTRA

We have deployed an in-house prepared container of PyMCA, and the project's one for Jupyterhub. The paths to the data are part of a special automount.

ESRF

We have been able to create a container with PyMca v.5.4.3 and Crispy installed with access to our home directories and the data directories. These directories were mounted to the container through NFS.

	These container images were custom made and took a long time to develop in order to support running them with a non-root user in order to support NFS. We have since created a template and shared this on the Calipsoplus Github page which other facilities can extend and improve. *We have also tested accessing the Jupyterhub service and creating Jupyter Notebooks. We are able to read/write our home directories and the data directories*
SOLEIL	*We didn't have the opportunity to test the use cases until now.*
PSI	*A.7*
DIAMOND	*No actual data analysis or processing has been attempted so far as this would require access to the data shares as well as authentication against the DIAMOND account management system.*
DESY	*The use cases could be deployed and tested. For all applications we need ready to use test datasets and some recipes to actually run the applications in an auto-play mode.*

Question B.5 User experience - general feedback ; which users tried the prototype, what was their feedback, how did match to their expectations / needs, etc.

ALBA	*Section A - Question 7.*
ELETTRA	*See Q.a.7. If you have any further information to report please write them here*
ESRF	*Like Section B – Question 4*
SOLEIL	*Up to now, no users have seen this project.*
PSI	*Like Section B – Question 4*
DIAMOND	*No users were involved in the test.*

DESY	*Up to now no user/staff involvement.*

Question B.6 Improvements - what is missing to make this a production ready service useful for the facility's users, what other solutions or features to be added/explored

ALBA	● *Resources Management: in order to make sure not to overload the system, there should be some sort of queuing system in order to limit access to users once a certain load is reached. And inform the next user in the queue when there is room for her/him to use a new machine.* ● *Demonstrator as a Docker Container. That worked very well during the hands-on meeting held in October 2018, and it is a good first contact to the project.*

ELETTRA	*We can list certain points that we consider potential improvements not in order of importance:* ● *The DAAS Portal (back-end & front-end) delivered as a Docker container* ● *Better documentation for the integration process (not only for the installation)* ● *More Docker containers for analysis applications* ● *Critical study on alternative remote desktop solutions* ● *Benchmarking*

ESRF	*At present, the portal only supports Docker which means there will be a small limit on the number of containers which can be run concurrently. We will need the portal to support Kubernetes which will allow us to scale the hardware to support many more containers concurrently with more resources (CPU, RAM) and remove the single point of failure (single Docker hypervisor).* *We also want to support virtual machines as an alternative to containers. This will require an OpenStack plugin to create virtual machines.* *As we still can't access our data, we need to finish the ICAT plugin which will allow the portal and our datahub to interact and see the list of experiments/data.* *One of the use cases for the portal was to have the ability to create a container (still linked to the quota) without the need for the user to have an experiment. This is ideal for IT staff, scientists and engineers who want to create containers for learning, testing or evaluating software. This also provides the ability to provide a large training service as containers/jupyterhub would be available to everyone.*

SOLEIL	*The link between our User Office portal (sunset) and calipsoplus is missing. We will have to work on this particular point before opening the service to the users.*

PSI

A.4

DIAMOND

DIAMOND uses SynchWeb as a web frontend to access the ISPyB database. Integration between both SynchWeb and the portal would probably be appreciated by users: accessing a visit in SynchWeb should present the user with the option to redirect to the corresponding proposal and visit in the portal, where data processing or visualization can then be started.

Access will also need to be provided to our HPC systems for data analysis that can benefit from it.

DESY

As mentioned the cloud/kubernetes integration would already solve a number of open issues. It would facilitate integration of authentication services, scalable deployment and utilization of (arbitrary) cloud resources.

We see a strong demand from users as well as beamline staff to support Jupyter notebooks launched from a JupyterHub, thereby enabling access to different compute resources ranging from HPC to kubernetes deployed single-user server. The rapid development of Jupyter-based services enabling for example interactive widgets or webgl support might leverage some of the integrations required for the portal. Combining Jupyter services with the portal might be achieved in different ways, either by making Jupyter an integral part of the back-end, or by loosely coupling the front-end to partially independent Jupyter-services. We need to keep an eye on the developments on both sides and possibly re-evaluate options based on these developments.

We also see an increasing demand for "anonymous" binder-like services, which would allow reproduction of processing workflows in a non-persistent manner. Such services are best implemented in a cloud/kubernetes environment (e.g. utilizing the repo2docker toolset to create instances on the fly).

Question B.7 Conclusion - what is the overall impression of the prototype and its suitability for adoption and further development in new projects like PaNSOC, ExPaNds, LEAPS, EOSC, etc

ALBA

The portal is a good starting point for future EOSC projects, such as PaNoSC or ExPaNDS.

Many new developments are expected, and integration. We might need to establish how to include new plugins without affecting current development or other facilities that may not need them.

We might need establish a change committee and contribution principles.

ELETTRA

We consider the portal and excellent demonstrator for projects like EOSC related projects like PaNSOC and ExPaNds. It will serve as a pivotal point for further developments and inspiration. Moreover it is the first system of its kind that the end-users have some limited experience with. The tests showed that there is plenty of space for improvements.

ESRF	*The portal works well as an initial prototype for PaNOSC however we need to be more flexible on adding different technologies such as OpenID Connect, OpenStack etc. If other project become involved, we will need to write many more plugins to support the platforms at the other facilities.* *We also need to address the coding architecture to make sure that it is scalable and easier to add to.*
SOLEIL	*This prototype is really impressive. Once everything will be tested and validated, it could be deployed in production because it provides a lot of useful services for our users. It's a good precursor of PaNOSC.*
PSI	*Due to our limited experience with the prototype, we don't feel we can give a useful assessment.*
DIAMOND	*When implemented completely at our facility according to the JRA2 blueprint, the portal will provide users with a powerful and new way of processing their data, without having to rely on copying the data back to their home institutes.*
DESY	*Our deployment was too basic to allow a fair assessment and comparison with existing solutions.*