PaNOSC

Photon and Neutron Open Science Cloud

H2020-INFRAEOSC-04-2018

Grant Agreement Number: 823852

D5.3: Repository of documented jupyter notebooks and Oasys canvases

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:	D5.3: Repository of documented jupyter notebooks and Oasys canvases
Deliverable Type:	Software
Dissemination Level:	Public
Contractual Delivery Date:	30/05/2022
Actual Delivery Date:	24/05/2022
EC project Officer:	Flavius Alexandru Pana

Document Control Sheet

Document	Title: D5.3: Repository of documented jupyter notebooks and Oasys canvases
	Version: 1
	Available at: https://github.com/panosc-eu/panosc
	Files: 1
Authorship	Written by: Carsten Fortmann-Grote
	Contributors: Mads Bertelsen, Stella d'Ambrumenil, Juncheng E, Aljoša Hafner, Gergely Norbert Nagy, Shervin Nourbakhsh, Mousumi Upadhyay Kahaly
	Reviewed by: Jordi Bodera Sempere
	Approved: Andy Götz

List of participants

Participant No.	**Participant organisation 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	Extreme Light Infrastructure Delivery Consortium (ELI-DC)	Belgium
6	Central European Research Infrastructure Consortium (CERIC-ERIC)	Italy
7	EGI Foundation (EGI.eu)	The Netherlands

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

Contents

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

1 Introduction

Deliverable D5.3 in PaNOSC is the release of jupyter notebooks and Oasys workspaces that demonstrate the utilization of simulation services in appropriate cloud services, i.e. jupyter hub instances or cloud based Oasys installations, respectively. This document lists the published notebooks and workspaces and gives a brief outline of their scientific content as well as instructions how to launch them. All notebooks and workspaces can be accessed through a common entry point, the ViNYL-notebooks repository at https://github.com/PaNOSC-ViNYL/ViNYL-notebooks. Installation instructions are provided in the README file of that repository. The repository is registered on zenodo at https://dx.doi.org/10.5281/zenodo.6562106/.

2 Simulation infrastructure

This Deliverable is accompagnied by Milestone MS5.3, the release of the simulation API library *libpyvinyl*. *libpyvinyl* ensures a common interface for simulations that choose to build upon this foundation. In addition, we built a instrument database to represent beamline and instrument components in neutron or x–ray RIs, their parameters and permissible parameter values. Instruments can be easily retrieved from an accompanying Python API, and expert users can update the instrument description directly through github where we host the database. By hosting the instrument database on github we ensure it is accessible to a broad user base without specialized knowledge of database technology such as SQL and we avoid having to implement and maintain web frontends. Another benefit is the option to run quality checks on proposed updates of instruments using Continuous Integration.

Our simulation services are backed by three distinct simulation frameworks: Simex for x–ray laser experiments, Oasys for x–ray optics, mainly focussing on synchrotron sources and McStasScript for neutron simulations. The underlying simulation codes are described in the respective framework's documentation. In the following, we document dedicated github repositories that host our jupyter notebooks and workspaces.

3 Oasys workspace repository

The main focus of OASYS is on providing a cohesive and easy-to-use graphical user interface (GUI) for a variety of X-ray optics simulation codes. It relies on the usage of Orange framework[1], with its atomic element being called a widget. During the standard usage pattern, different widgets (optical elements) are connected to each other, thus creating a workflow (beamline). This is particularly useful for X-ray optics simulations, as the wavefront propagation direction is well-defined and known in advance (from the source to the experimental station).

Since Orange is built with Qt, a suitable way of interacting with the environment in the cloud had to be found. Docker containers had been selected through a thorough evaluation. The recipes to make

[1] https://orangedatamining.com/

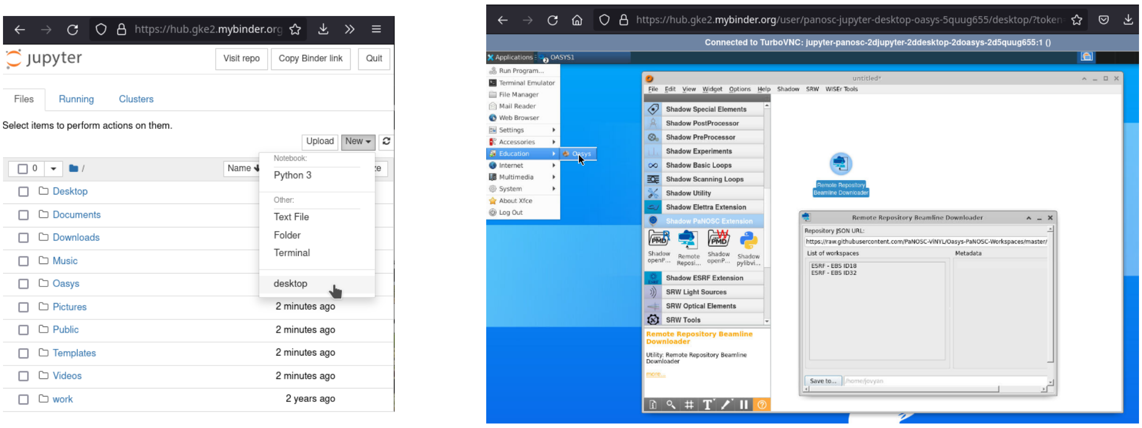
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

and run the containers have been provided for the deliverable. We provide two different Docker containers, one being suitable for local (workstation) installation and another one being suitable for cloud (server) installation. The latter comes in the form of a Jupyter hub plugin, essentially linking the OASYS package with the rest of the WP5 simulation codes and being able to be installed into any existing Jupyter hub instance. A screenshot of OASYS running remotely in Jupyter hub is shown in Figure 1 and the code repositories (including documentation on how to build and run) of the two aforementioned Docker containers are available through the following links:

Table 1: Oasys docker containers

OASYS local Docker container	`https://gitlab.elettra.eu/panosc/ceric/oasys-local-docker`
OASYS Jupyter Docker container	`https://gitlab.elettra.eu/panosc/jupyter-desktop-oasys`
Dockerhub release	`https://hub.docker.com/r/ceric/panosc-oasys-local`

OASYS projects are internally saved into the so-called OASYS *Workspaces* (OWS). OWS files have an XML-like structure with the layout and some general properties being saved in easy-to-access ASCII format, while the actual content/parameters of individual widgets (optical elements) are in binary format (pickled Numpy objects). A beamline/instrument database therefore consists of a set of previously prepared and curated OWS files corresponding to an actual beamline. Each of the entries in the database contains some basic metadata and a pointer to the location of the respective OWS file. The user interacts with the database through a dedicated widget that is a part of the *Shadow Panosc toolbox* extension. The database itself consists of the actual OWS files and an index file and is hosted on Github. Since OWS files contain binary parts it is desirable not to change their content. Therefore an index file in JSON format is provided, serving two purposes:


https://hub.gke2.mybinder.org

Select items to perform actions on them.

Name	
Desktop	
Documents	
Downloads	
Music	
Oasys	
Pictures	2 minutes ago
Public	2 minutes ago
Templates	2 minutes ago
Videos	2 minutes ago
work	2 years ago

(a)

https://hub.gke2.mybinder.org/user/panosc-jupyter-desktop-oasys-5quug655/desktop/?token=

Connected to TurboVNC: jupyter-panosc-2djupyter-2ddesktop-2doasys-2d5quug655:1 ()

Remote Repository Beamline Downloader

Utility: Remote Repository Beamline Downloader

Repository JSON URL: https://raw.githubusercontent.com/PaNOSC-ViNYL/Oasys-PaNOSC-Workspaces/master/

List of workspaces:
ESRF - EBS ID18
ESRF - EBS ID32

Save to... /home/jovyan

(b)

Figure 1: (a) Running the XFCE desktop environment from within Jupyter hub. (b) A new browser tab opens with XFCE desktop, providing a way to run OASYS inside the web browser while sharing the computing resources with the Jupyter hub instance.

- user needs no Github account for read access,

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

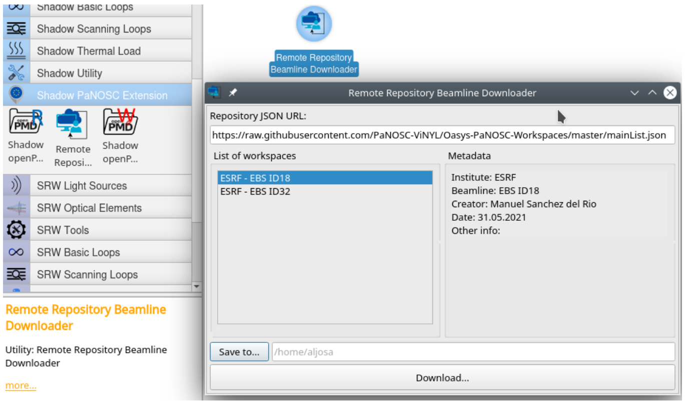
- additional metadata can be provided.

Table 2: Oasys workspaces

OASYS workspaces database	https://github.com/PaNOSC-ViNYL/Oasys-PaNOSC-Workspaces
OASYS Panosc toolbox code	https://github.com/PaNOSC-ViNYL/OASYS1-PaNOSC

The index file[2] needs to be curated by the database admin and its changes can be easily tracked through the git commits. Furthermore, this way the widget offers the possibility of interacting with a database of arbitrary file formats and arbitrary locations, while still retaining the metadata. Figure 2 shows a screenshot of the widget. The links to the respective repositories are listed in Table 2.

Currently, two ESRF beamlines are ingested into the database and serve as the template for growing the repository.



Remote Repository Beamline Downloader

Utility: Remote Repository Beamline Downloader

more...

Repository JSON URL: https://raw.githubusercontent.com/PaNOSC-ViNYL/Oasys-PaNOSC-Workspaces/master/mainList.json

List of workspaces

ESRF - EBS ID18
ESRF - EBS ID32

Metadata

Institute: ESRF
Beamline: EBS ID18
Creator: Manuel Sanchez del Rio
Date: 31.05.2021
Other info:

/home/aljosa

Figure 2: OASYS widget for interaction with the online instrument database.

4 McStasScript notebooks

The McStasScript notebook repository and the code repository are listed in Table 3. The McStasScript notebook repository currently contains a full 11 notebook tutorial on McStas using the McStasScript Python interface developed under WP5. This tutorial is also available through the online McStasScript documentation. The repository also contains an example on using the widget interface for McStasScript, and an example on using a cryostat construction tool included in McStasScript.

Table 3: McStasScript resources

McStasScript code repo	https://github.com/PaNOSC-ViNYL/McStasScript
McStasScript notebooks	https://github.com/PaNOSC-ViNYL/McStasScript-notebooks
McStasScript documentation	https://mads-bertelsen.github.io

[2] https://github.com/PaNOSC-ViNYL/Oasys-PaNOSC-Workspaces/blob/master/mainList.json

5 SimEx notebooks

Numerous notebooks have been developed that showcase the use of the SIMEX library for simulation of photon experiments at x–ray laser facilities. The following table gives an overview and links directly to a static version of the notebooks on github. To run these notebooks, a working SIMEX installation is needed.

Table 4: SIMEX notebooks

Notebook title (link)	**Description**
Jungfrau-ESRF.ipynb	Crystallograhy at ESRF with 4M Jungfrau detector
crystFEL.ipynb	Nanocrystal diffraction at Eu.XFEL SFX instrument
diffrChaperonins.ipynby	Singe particle imaging at Eu.XFEL SPB instrument
start_to_end_demo.ipynb	Start–to–end simulation for single particle imaging at Eu.XFEL SPB instrument.
EuXFEL_SASE1_SPB_KBfocus.ipynb	KB mirror focusing of SASE1 beam at Eu.XFEL
crl_focus.ipynb	CRL focusing of SASE1 beam at Eu.XFEL
edge.ipynb	Coherent x–ray diffraction from a solid edge
spheres.ipynb	Coherent x–ray diffraction from solid spheres
wire.ipynb	Coherent x–ray diffraction from a thin wire

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