Reproducible Scientific Software Release 2
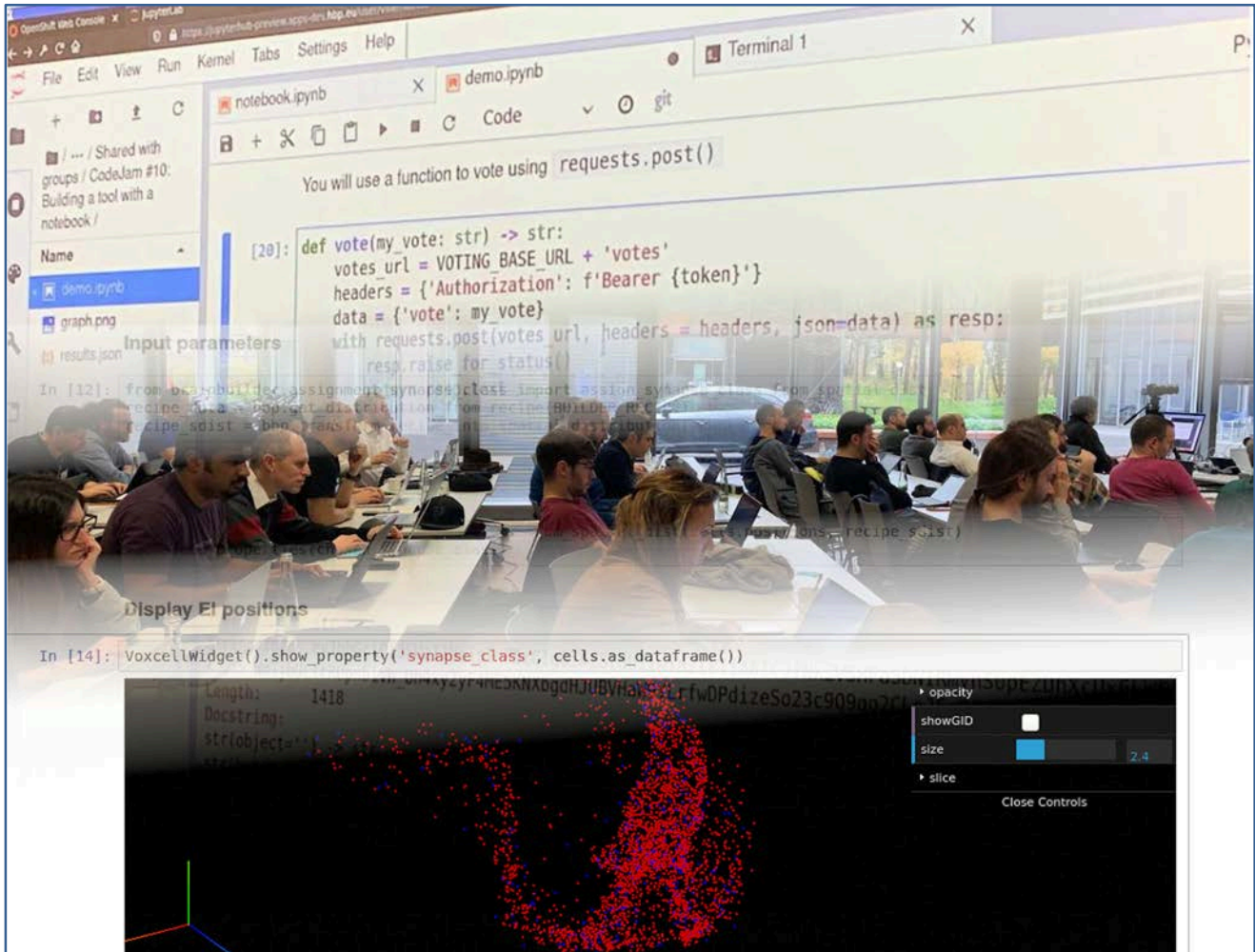# *(D5.5.1 - SGA2)*

Figure 1: JupyterLab in workshop and Jupyter Notebook for cell placement

A preview of the deployment featuring JupyterLab was used for a workshop at the CodeJam #10 in November 2019. The lower section is a screenshot of the Cell Placement Simulation from the Brain Simulation Platform. These teaching and exploration use cases are explained in more details in Section 3.

Project Number:	785907	Project Title:	Human Brain Project SGA2

Document Title:	Reproducible Scientific Software Release 2
Document Filename:	D5.5.1 (D32.1 D76) SGA2 M22 ACCEPTED 201005.docx
Deliverable Number:	SGA2 D5.5.1 (D32.1, D76)
Deliverable Type:	Report
Work Packages:	WP5.5
Key Result(s):	KR5.9
Dissemination Level:	PU
Planned Delivery Date:	SGA2 M22 / 31 Jan 2020
Actual Delivery Date:	SGA2 M23 / 20 Feb 2020; resubmitted: 31 Aug 2020 and 2 Oct 2020; accepted 5 Oct 2020
Author(s):	Jonathan VILLEMAIRE-KRAJDEN, EPFL (P1)
Compiled by:	Jonathan VILLEMAIRE-KRAJDEN, EPFL (P1)
Contributor(s):	Marc MORGAN, EPFL (P1)
SciTechCoord Review:	Allan FRANCANI, EPFL (P1)
Editorial Review:	Annemieke MICHELS, EPFL (P1)
Description in GA:	This production release will make publicly available JupyterLab functionality optimized and integrated with HBP data management solutions and data service APIs (T5.5.2)
Abstract:	
Keywords:	JupyterHub, Jupyter notebooks, JupyterLab, Reproducible science, Collaboratory
Target Users/Readers:	Scientific community, scientific infrastructure providers, education platform providers

Table of Contents

Table of Figures

History of Changes made to this Deliverable (post Submission)

Date	Change Requested / Change Made / Other Action
20 Feb 2020	Deliverable submitted to EC
29 Jul 2020	Resubmission with specified changes requested in Review Report Main changes requested: • Integration of Jupyter notebooks with collaboratory, help function, links.
27 Aug 2020	Revised draft sent by SP/CDP to PCO. Main changes made, with indication where each change was made: • Addendum added to address issues identified in Review Report.
31 Aug 2020	Revised version resubmitted to EC by PCO via SyGMa
2 Oct 2020	Minor editorial change by PCO
2 Oct 2020	Revised version resubmitted to EC by PCO via SyGMa

1.   Introduction

Jupyter Notebooks are already widely used in the Human Brain Project (HBP), as well as by a wider community of scientists. This document covers the release of the Jupyter environment in the Collaboratory 2.0 release. In the Collaboratory 1.0 release during SGA1, there was a first implementation that served as a baseline for the implementation in the Collaboratory 2.0. The integration provided with the rest of the environment assists the users with collaboration and with access to other services provided by the HBP, such as high-performance computing (HPC). This new release updates the Jupyter stack and integrates with the Collaboratory 2.0 ecosystem. The use cases at the HBP can be defined in these broad categories: education, dissemination, documentation, exploration and automation.

Scientists have many roles in society, including research and education. Jupyter notebook is a tool that supports them in both of these tasks. In the early stages of research, Jupyter notebooks are a convenient way to explore data and run simulations in an interactive trial and error process. In the later stages, they can be used to coordinate workflows and document processes. Finally, when the scientists have reached their results, notebooks are well adapted for replication, dissemination and education.

1.1   History

Jupyter notebooks were designed for the workflows of scientific computing (Thomas *et al.*, 2016). It rests on the foundation of the IPython project (Perez and Granger, 2007) which began in 2001 as enhancements to the Python ecosystem to improve visualisation and the interactive shell. It added features for distributed computing. Notebooks are a rich version of a REPL (read-evaluate-print-loop) interactive shell. They allow a narrative to be embedded with code, and the code's output. Jupyter notebooks provide a platform for running it and communicating with a kernel, the execution environment for the code. JupyterLab is an interface to interact with the Jupyter notebooks.

At the HBP, Jupyter notebooks have been integrated in the Collaboratory since 2016. The implementation provided integration with the Collaboratory 1.0 infrastructure, such as the storage service. This infrastructure has been widely used (see SGA2 M12 Deliverable D5.8.1 (D35.1, D19)[1]) and has been maintained and upgraded to deal with increasing use. However, many components are deprecated and the current infrastructure has limitations in both performance and user experience. In order to provide a smooth transition to a newer service, we have handled the transition by providing a new deployment in parallel with the Collaboratory 1.0.

1.2   Motivation

The integration of Jupyter Notebooks needed to be updated to tie into the new infrastructure. Moreover, the technology to deploy distributed services has evolved rapidly in the last years, and the Jupyter community has shifted away from the Docker Swarm approach used in the Collaboratory 1.0 release, focusing instead on Kubernetes as a platform for deployment. The new JupyterHub is also integrated with Collaboratory 2.0 Drive featuring a FUSE mount, making data access much more convenient than the previous, deprecated storage system. Also, the quota management has been greatly improved with this deployment, which solves issues from the previous platform regarding resource utilisation and the stability of the cluster. Finally, some security issues were also fixed with respect to the authentication flow.

The Collaboratory 2.0 has been released earlier this year with improvements in the user experience. The migration of content is expected to happen in the first months of SGA3.

---

[1] https://sos-ch-dk-2.exo.io/public-website-production/filer_public/99/cd/99cd9654-c822-4c08-b8b1-d176b30d8ecc/d581_d351_d19_sga2_m12_accept_190905.pdf

1.3 Wider community

The wider Jupyter community is very active and the project is evolving: the Project Jupyter Steering Council received the 2017 ACM Software System Award[2]; the new JupyterLab interface is more flexible; the JupyterHub Spawners built onto Kubernetes are robust and flexible; and the authenticator architecture is also improved. Moving to a more recent platform brings us all the benefits above and keeps us abreast with the development lead by a large scientific community.

2. Release

The JupyterHub release is available at https://lab.humanbrainproject.eu/ until final integration into EBRAINS at https://lab.ebrains.eu. Access require a user to have an HBP account. Documentation can be found in the "Jupyter User Group" Collab at https://wiki.humanbrainproject.eu/bin/view/Collabs/jupyter-user-group/Documentation/. A user account can be requested via support@humanbrainproject.eu.

2.1 Interface

The new deployment of Jupyter in the Collaboratory 2.0 environment provides the JupyterLab interface, which offers split views and a more integrated development environment (IDE) feel. The file browser allows users to view their folders from Collabs and is synchronised with the Drive. The interface can be customised by adding extensions. A screenshot of the JupyterLab interface in action can be seen in Figure 2.

---

[2] ACM Recognizes Innovators Who Have Shaped the Digital Revolution, https://awards.acm.org/about/2017-technical-awards
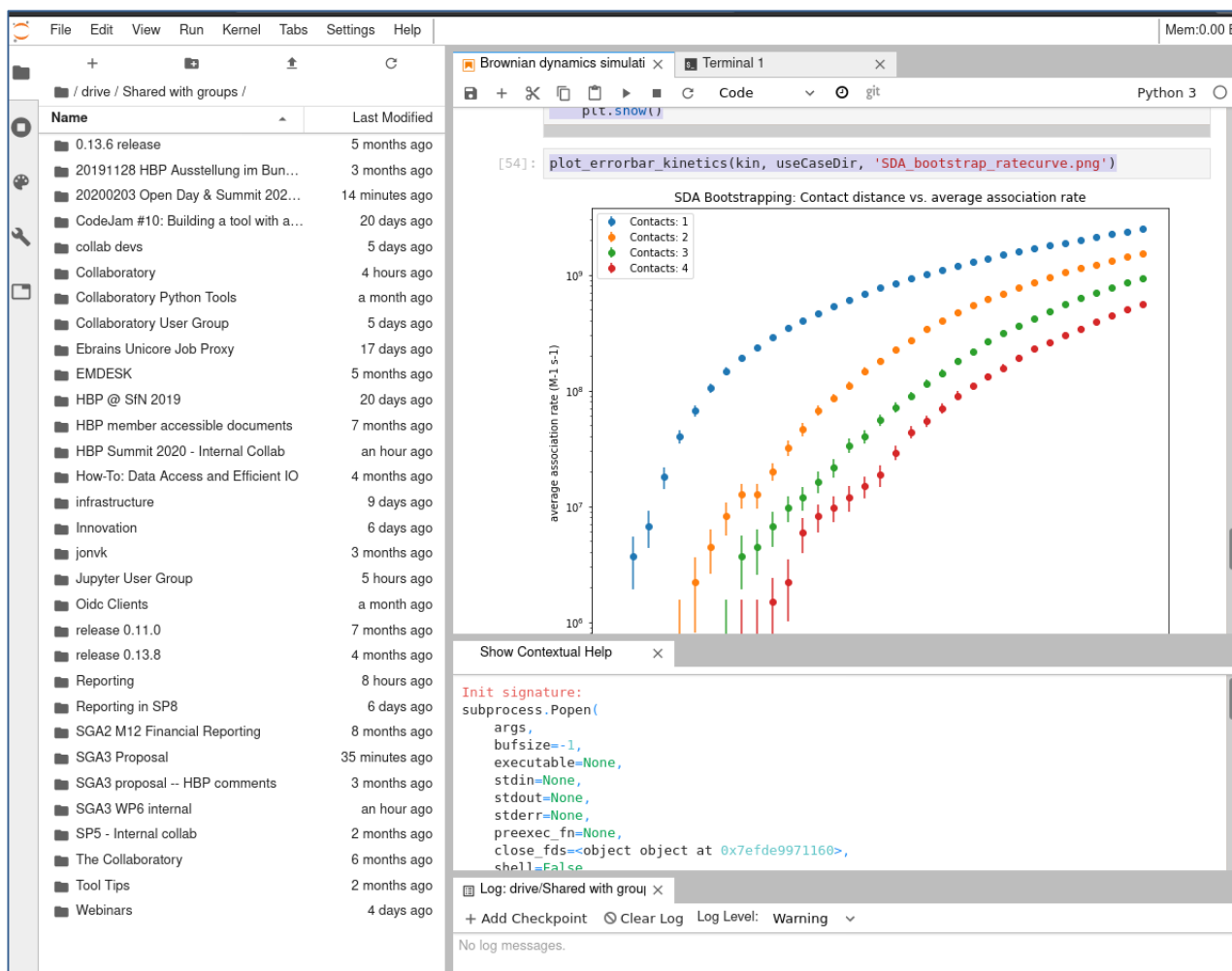
Figure 2: JupyterLab interface as seen in the Collaboratory 2.0

2.2   Implementation

The JupyterHub installation in the Collaboratory 2.0 provides users with Jupyter notebooks running on the Fenix research infrastructure. The notebook server image comes with popular science libraries and neuroscience-specific libraries pre-installed. It also provides an integration with the Collaboratory 2.0 Drive, allowing users to share notebooks and data easily.

JupyterHub is designed for extensibility. We override the authenticator, the component which manages authentication, and the spawner, the component which creates a notebook server for each connected user. These components are described below in more details. The architecture is illustrated in Figure 3. JupyterHub runs in Kubernetes and schedules pods for each user. The user first connects to JupyterHub for authentication, then is proxied to a notebook server running in the pod which has been spawned for them by Kubernetes. Their folders from the Drive are mounted in their notebook container and visible in the JupyterHub interface (Figure 4).
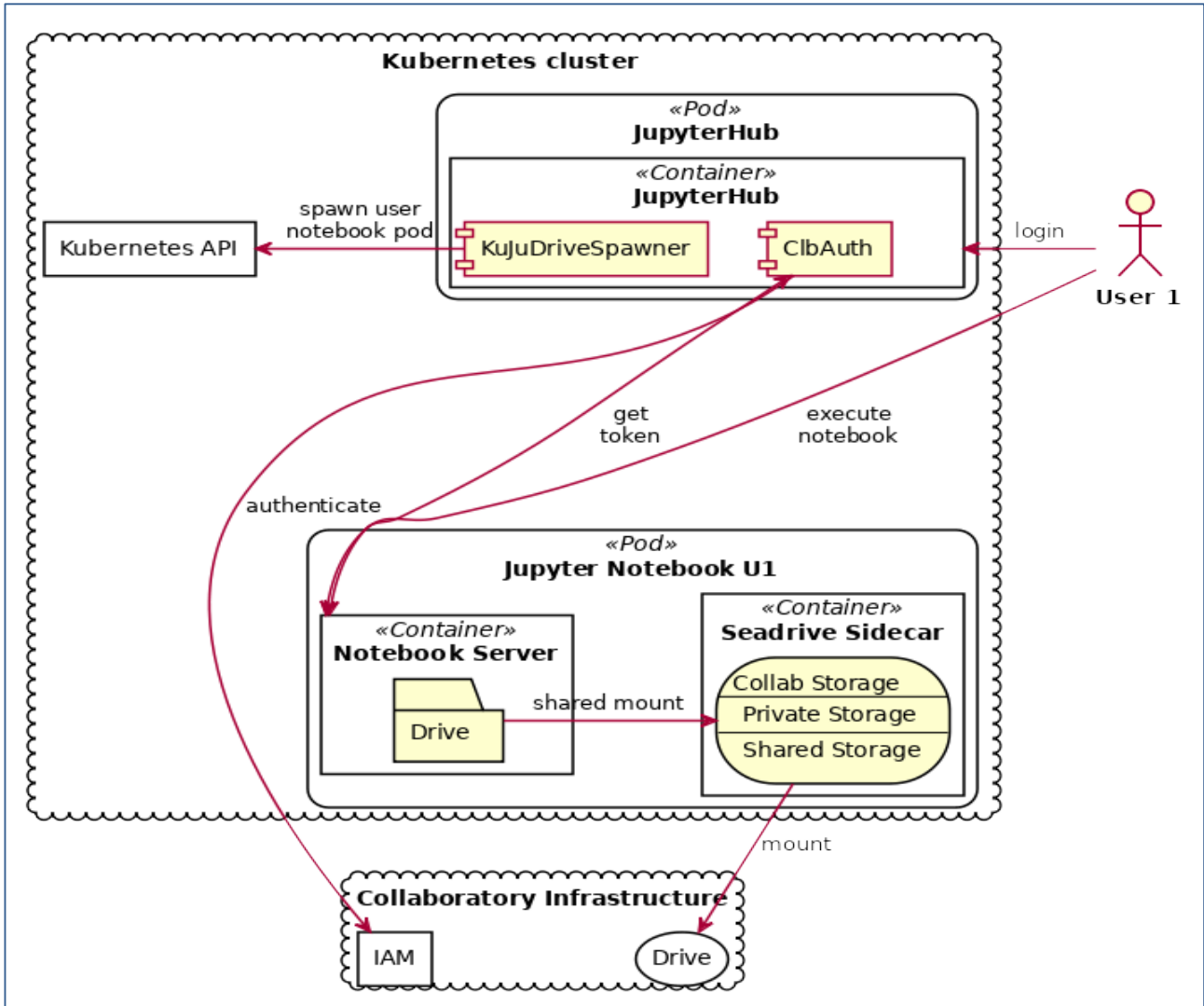
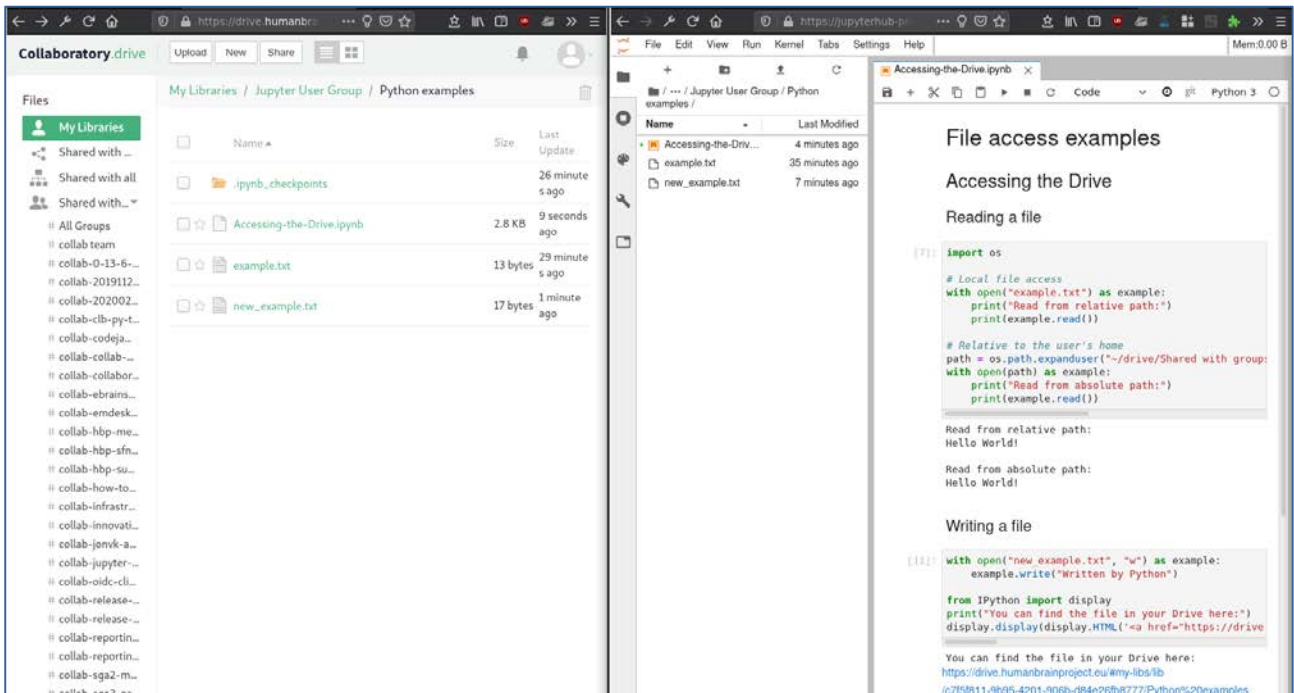**Figure 3: JupyterHub Architecture in the Collaboratory 2.0**

Figure 4: Documents are synchronised between the Collaboratory Drive and Jupyter Notebook.

2.2.1   Authentication and Authorization integration

The Collaboratory uses OpenID Connect[3] as the backbone for its authentication and authorisation. In JupyterHub, this is managed through a custom authenticator extending the OAuthenticator package provided by the JupyterHub team. The HBP authenticator connects to our identity and account management (IAM) service. It keeps a persistent session which the user can revoke. So long as this session is active, the user can obtain a token which will authenticate them and authorise them to connect to third party applications from a notebook.

The token mechanism consists of a service which obtains a valid access token by using the user's refresh token obtained at login. This service runs on JupyterHub. The implementation can be found on Github[4]. Helper functions are injected into the notebook kernel in order for users to be able to access this token. This access token can then be used to make requests to other services, such as the HPC platform using Unicore.

2.2.2   Drive Integration

The Collaboratory 2.0 Drive integration is provided by the custom spawner, KuJuDriveSpawner[5], based on JuyterHub's Kubespawner. This is a key component of the JupyterHub architecture, which launches a process to handle the user's environment. In this case, it is requesting a pod running two containers from Kubernetes. Before initiating the launch, the spawner obtains an access token from the drive and sets up a sidecar[6] container to mount the users' libraries (folders) from the drive server. The mount requires elevated privileges. For this reason, it is run in a sidecar container, which is not directly accessible by the user. The Drive mount is done in a second shared mount which is also available from the Jupyter Notebook container to also access the files from the sidecar. This setup provides isolation for the Drive mount from the user. As a result, there is a secure and simple way to share documents between collaborators which are accessible from the notebooks.

2.3   Migration

The migration to the new platform is expected to occur gradually. Both platforms will be run in parallel up to September 2021 to give users time to migrate notebooks to the new platform. The major change is the access to files in the HBP storage, since previously files needed to be downloaded and uploaded specifically. The new platform simplifies the design by offering a local mount of the drive, making remote file operations transparent. Python 2 has also reached its end of life and will no longer be supported. There will be "migrathons" with the Collaboratory team helping users to migrate before the end of SGA2 and during the beginning of SGA3. Support will be offered throughout the transition phase.

3.   Use Cases

The Jupyter Notebook use cases are centred around reproducible science and dissemination. The core feature of Jupyter Notebooks in a deployment such as the Collaboratory's JupyterHub are the ease of use, the collaborative environment and capability to tie a narrative to an environment which allows users to interact with the content. This makes them well adapted to various workloads where scientists can execute a workload without having to install software on their laptops and then share the work they do with a group. The integration in the Collaboratory also allows them to increase the visibility of their work.

---

[3] "Welcome to OpenID Connect", https://openid.net/connect/
[4] https://github.com/HumanBrainProject/jupyterhub-access-token-service
[5] https://github.com/HumanBrainProject/kujudrivespawner
[6] "Sidecar Pattern", https://docs.microsoft.com/en-us/azure/architecture/patterns/sidecar

3.1 Education

The Collaboratory 1.0's Jupyter infrastructure has been used to teach online courses (MOOC)[7] and courses during workshops. As a platform, it allows users to run code without installing any software on their machines and it scales sufficiently to allow a classroom of students to run the material simultaneously.

With respect to this release, the preview environment has already been used at the HBP CodeJam #10 in Heidelberg to give a workshop presenting the new infrastructure. The material can be found in the Collaboratory 2.0[8]. HBP members have been given access to the preview environment to begin migrating existing material from the Collaboratory 1.0.

3.2 Dissemination

A large part of work during the scientific process is the dissemination. Notebooks can be used alongside publications to improve the reproducibility and provide detailed descriptions of the analysis, workflow and simulations. Tutorials can also be made available for teaching. Jupyter has been embraced by the community for both.

3.2.1 Live Papers

The result of research is generally published in journals with a description of the process. Jupyter Notebooks allow for a more detailed publication for data analysis and simulations. Interactive tables and charts can be included and readers can review the analysis and make changes. In the last few years, papers have been published with accompanying materials including Jupyter notebooks.

Live papers have been produced as part of the HBP funded research, such as the work of Mandge *et al* (Mandge and Manchanda, 2018) with the accompanying Collab[9]. This Collab contains a notebook with code to access the model and generate the figures used in the paper. A list of live papers from the Brain Simulation Platform can be found in the Collaboratory 1.0[10], and on GitHub to get a more general idea of the concept[11].

3.2.2 Tutorials

Platforms can create Collabs with tutorials that can be run within a notebook. This practice was well established in the Collaboratory 1.0. The tutorials can be used in workshops, but also for dissemination if they are sufficiently clear to be run independently.

---

[7] "HBP MOOC How to Study the Multi-Scale Brain", https://education.humanbrainproject.eu/web/hbp-mooc
[8] "CodeJam #10: Building a tool with a notebook",
https://wiki.humanbrainproject.eu/bin/view/Collabs/codejam-10-building-a-tool-with-a-notebo/
[9] "A biophysically detailed computational model of urinary bladder small DRG neuron soma",
https://collab.humanbrainproject.eu/#/collab/54924/nav/375596
[10] "The Brain Simulation Platform Live Papers",
https://collab.humanbrainproject.eu/#/collab/1655/nav/306845
[11] "A Gallery of Interesting Jupyter Notebooks", https://github.com/jupyter/jupyter/wiki/A-gallery-of-interesting-Jupyter-Notebooks#reproducible-academic-publications

3.3 Documentation

Jupyter Notebooks can accompany documentation for libraries built for the HBP. In the first edition of the Collaboratory, tutorials documenting the use of multiple packages can be found [12] [13]. These tutorials will be migrated to the new Collaboratory during SGA3.

3.4 Exploration

Exploration is a key phase in hypothesis building. IPython, on which Jupyter Notebooks are built, was first developed by Fernando Perez in 2001 to provide better tools for validation. The modern Jupyter environment is well suited to building dashboards and interactive analysis pipelines.

In the HBP, there are examples of notebooks providing dashboards and analysis workflows[14] [15]. The Drive provides simpler access to files and an easier solution for output to be saved. The JupyterLab interface also is more flexible than the original Jupyter Notebook interface for dashboard creation.

Finally, the use of Jupyter Notebooks, in a collaborative environment, such as the Collaboratory with its integrated Drive, allows users to easily share notebooks and data with selected team members. The Collaboratory provides an environment with team management, a wiki, file storage and access to compute resources whose setup and coordination is simple. This allows multiple users to contribute to a notebook without requiring expertise in development or a complex setup.

3.5 Automation

Jupyter notebooks can be used to run HPC tasks using UNICORE[16]. Such an integration can be seen in the Collaboratory 1.0 (Senk *et al.*, 2017) and documentation for the Collaboratory 2.0 can be found in the tutorials[17].

4. Conclusion

Jupyter notebooks are a popular platform for scientists to collaborate and work on. The Collaboratory environment includes a JupyterHub platform where scientists have convenient access to notebooks with libraries already setup. Moreover, the environment allows them to easily share their work with others and access the other resources of the platform. Powerful compute resources from Fenix are available with a click. The Drive allows them to easily access and share documents and notebooks.

The new environment increases the performance of the system, simplifies and improve the user experience, and brings the release of JupyterLab up to date. The Kubernetes platform provides better scalability than the previous installation. The Drive integration has been improved compared to the previous storage service and client, which removes most of the complexity in setting up notebooks. The environment is fully up to date with the evolution of the Jupyter ecosystem.

---

[12] "Elephant Tutorial", https://collab.humanbrainproject.eu/#/collab/651/nav/6056
[13] "Using UNICORE for HPAC Access", https://collab.humanbrainproject.eu/#/collab/3656/nav/28988
[14] "HippoUnit Validation Dashboard", https://collab.humanbrainproject.eu/#/collab/8123/nav/401176
[15] "Online Use Cases – Molecular Level", https://collab.humanbrainproject.eu/#/collab/1655/nav/362934
[16] "UNICORE": https://www.unicore.eu/
[17] "HPAC tasks", https://wiki.humanbrainproject.eu/bin/view/Collabs/jupyter-notebooks-collaboratory/Tutorials/Running%20HPAC%20Tasks/

5.   References

Mandge, D., and R. Manchanda, 'A Biophysically Detailed Computational Model of Urinary Bladder Small DRG Neuron Soma', PLOS Computational Biology, Vol. 14, No. 7, July 18, 2018, p. e1006293.

Perez, F., and B.E. Granger, 'IPython: A System for Interactive Scientific Computing', Computing in Science & Engineering, Vol. 9, No. 3, 2007, pp. 21–29.

Senk, J., A. Yegenoglu, O. Amblet, Y. Brukau, A. Davison, D.R. Lester, A. Lührs, et al., 'A Collaborative Simulation-Analysis Workflow for Computational Neuroscience Using HPC', in E. Di Napoli, M.-A. Hermanns, H. Iliev, A. Lintermann, and A. Peyser (eds.), High-Performance Scientific Computing, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2017, pp. 243–256. (P673)

Thomas, K., R.-K. Benjamin, P. Fernando, G. Brian, B. Matthias, F. Jonathan, K. Kyle, et al., 'Jupyter Notebooks - a Publishing Format for Reproducible Computational Workflows', Stand Alone, 2016, pp. 87–90.

6. Addendum

While the SGA2 Report was being reviewed, the final step of integration of the Jupyter Notebook feature was deployed to production in Collaboratory 2. The Collaboratory.drive feature is now accessible directly from the main access point to collabs, i.e. the https://wiki.humanbrainproject.eu URL as shown in the figure below.
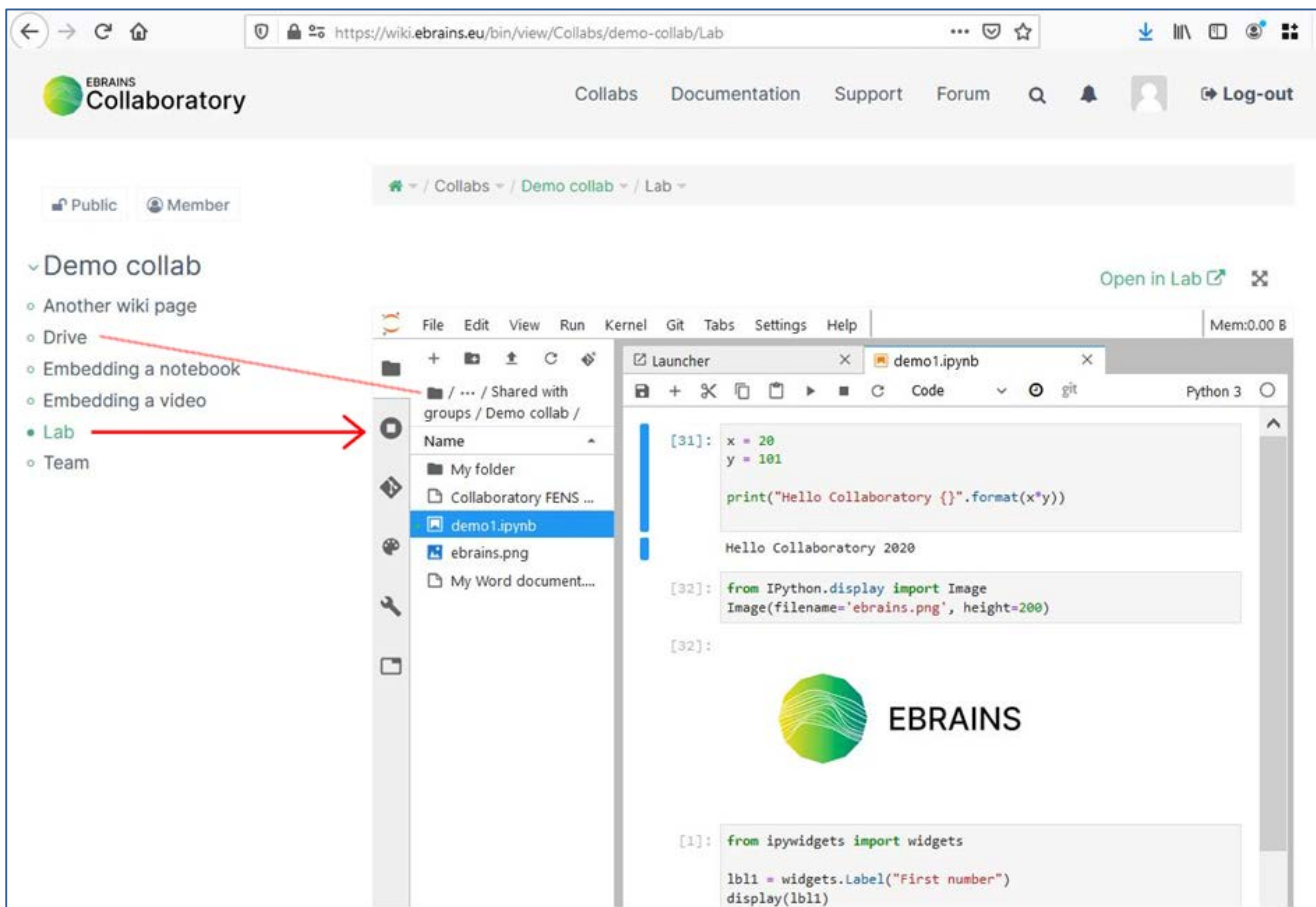

Figure 5: All collabs now have a Lab link that opens JupyterLab directly from the collab

The navigation menu of all collabs now includes a "Lab" navigation item (alongside Drive and Team). Lab opens JupyterLab in the main iframe of the window. JupyterLab is opened directly at the folder corresponding to the Drive of the present collab. As for the Drive, the top of this frame offers the possibility of maximizing JupyterLab in the window, or of opening JupyterLab in a separate tab.

Users don't need to start JupyterLab to view a notebook. The collab users with editor permission can publish a notebook in a notebook viewer directly in a wiki page of the collab as shown in the figure below.

The new Documentation link in the menu at the top of the window takes the user to the "The Collaboratory" collab which regroups or points to all the documentation for the Collaboratory. Some additional effort is being done to improve the organization of the documentation ahead of the first Collaboratory Migrathon event organized to support users in migrating their Collaboratory 1 content to Collaboratory 2.