Cloud-basierte Web-Anwendungen und Jupyter Notebooks im Zusammenhang mit globalen Luftqualitätsdaten

Martin G. Schultz, Jülich Supercomputing Centre (m.schultz@fz-juelich.de)

With a lot of material from

JUPYTERLAB - SUPERCOMPUTING IN YOUR BROWSER

Training course "Introduction to the usage and programming of supercomputer resources in Jülich"

2022-05-16 | JENS H. GÖBBERT (J.GOEBBERT@FZ-JUELICH.DE)
TIM KREUZER (T.KREUZER@FZ-JUELICH.DE)
ALICE GROSCH (A.GROSCH@FZ-JUELICH.DE

TOAR-I 2014-2019

Tropospheric Ozone Assessment Report (TOAR)

Global metrics for climate change, human health and crop/ecosystem research

Mission:

To provide the research community with an up-to-date scientific assessment of tropospheric ozone's global distribution and trends from the surface to the tropopause.

Deliverables:

1) The first tropospheric ozone assessment report based on all available surface observations, the peer-reviewed literature and new analyses.
2) A database containing ozone exposure metrics at thousands of measurement sites around the world, freely accessible for research on the global-scale impact of ozone on climate, human health and crop/ecosystem productivity.

Stakeholders:

Member of the Helmholtz A

WMO

GAW

LRTAP Task Force on Hemispheric Transport of Air Pollution

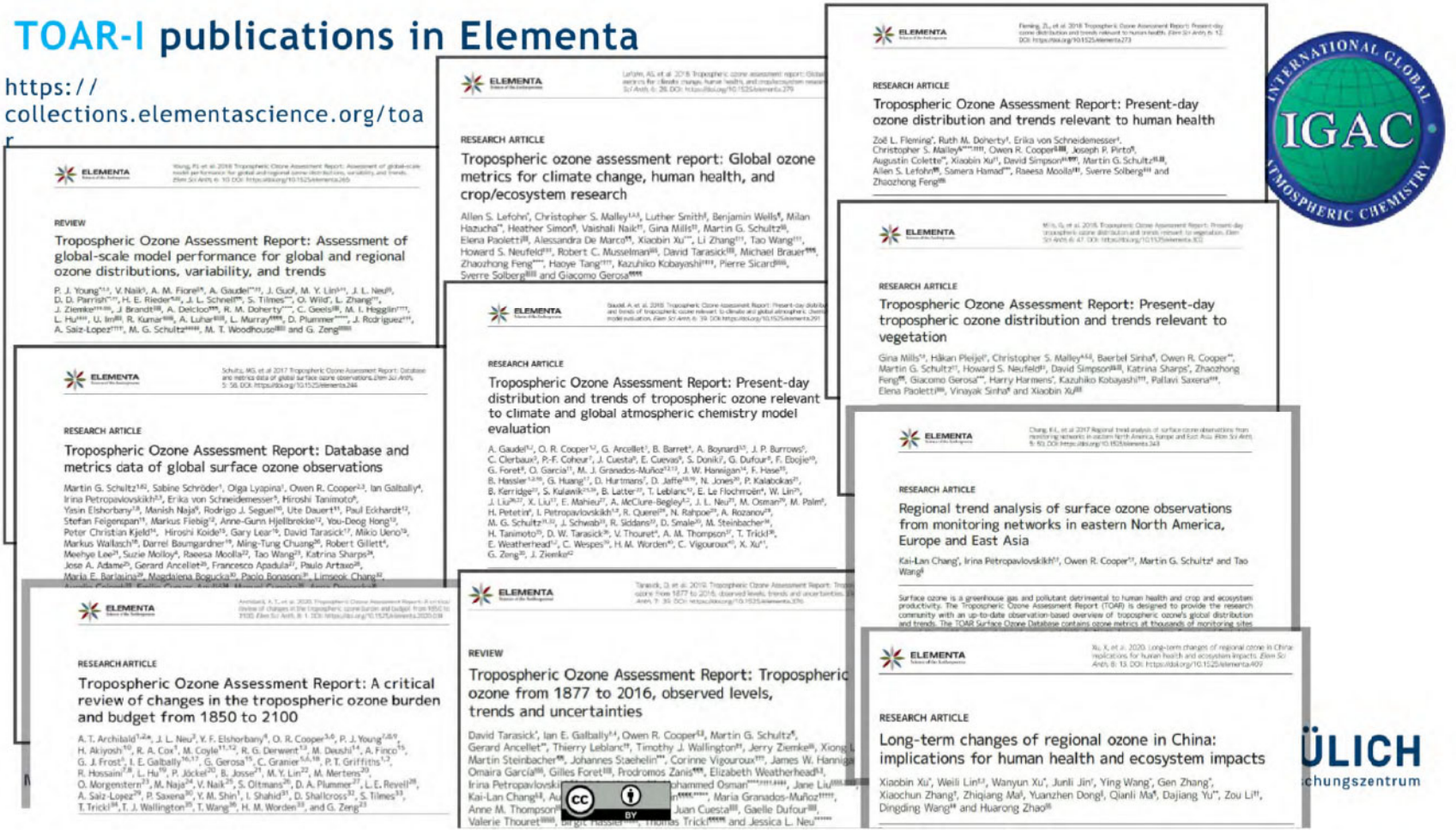
World Health Organization

NOAA

JÜLICH Forschungszentrum

TOAR-I publications in Elementa

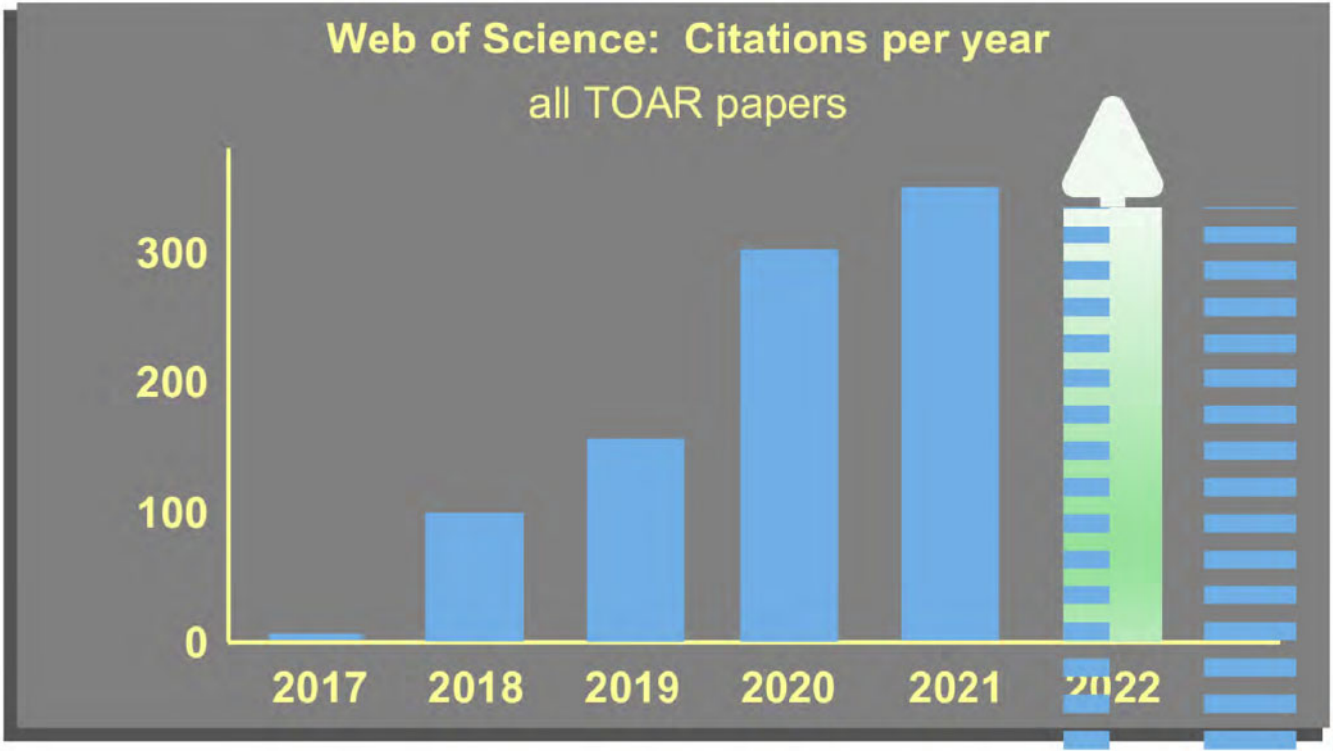
https://collections.elementascience.org/toar



IGAC International Global Atmospheric Chemistry

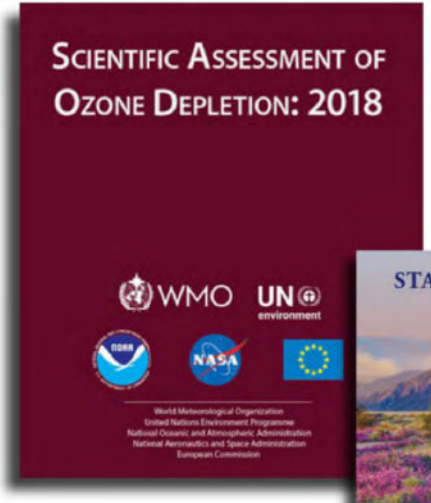
ÜLICH
chungszentrum

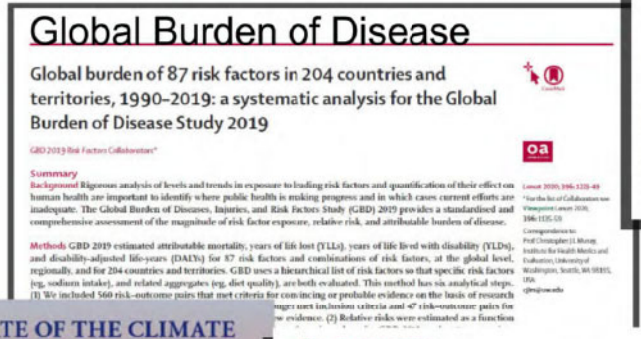
TOAR-I publications are highly cited

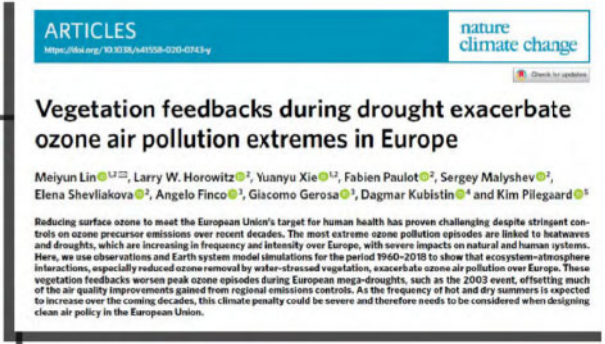


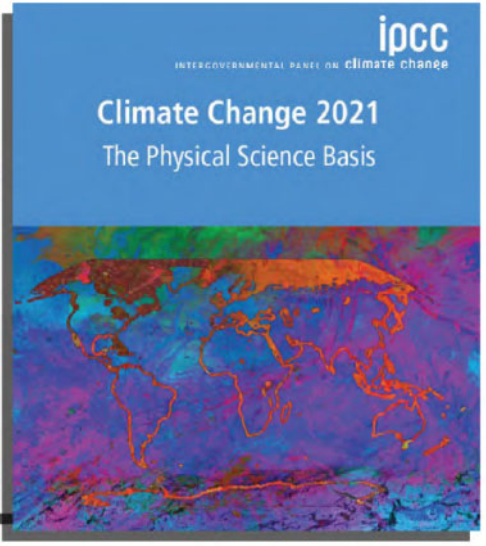
TOAR-I Fostering independent research

Dozens of independent studies have utilized the TOAR database









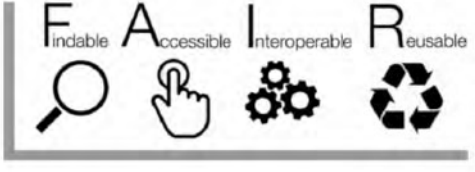
TOAR-II Primary Goal and Scientific Scope

Final Product: An observation-based, up-to-date assessment of tropospheric ozone's distribution and trends on regional, hemispheric and global scales.

TOAR-II will assess the physical science basis for tropospheric ozone's global distribution and trends *(similar to IPCC Working Group I)*

TOAR-II will also explore and quantify the ***impacts*** of tropospheric ozone on human health, crop and ecosystem productivity and climate change *(similar to IPCC Working Group II)*

Foundation: An enhanced data infrastructure based on open data and FAIR principles



JÜLICH
Forschungszentrum

Purpose of TOAR data:
To provide globally consistent metrics for analyses of health, vegetation, and climate impacts from ozone air pollution

We provide data + analysis services

The TOAR Data Infrastructure

… comprises of

The TOAR database of global ground-level air quality observations

Web services for accessing and processing of TOAR data and metadata

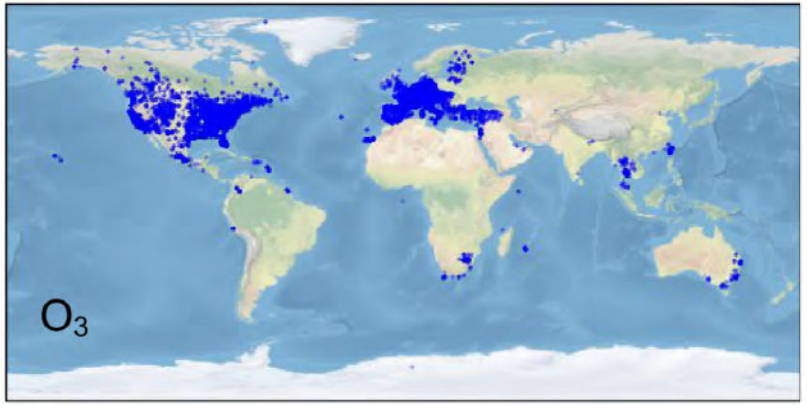
Publication services for original TOAR datasets

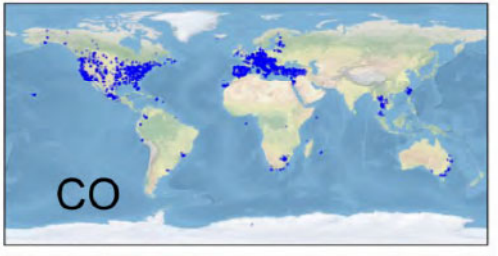
Tools, source code, and documentation

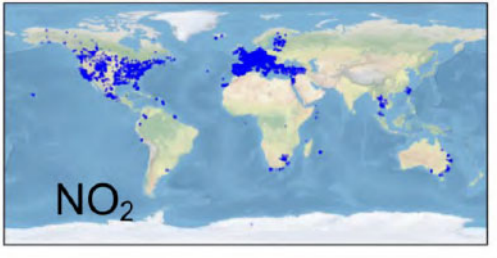
The TOAR Database

Current status as of 01 Sept 2022

stations: 20,100; time series: 81,231;
data records: 3,403,601,152







You can also find and submit data of PM_x, NO, BC, various VOCs and meteorological variables.

JÜLICH Forschungszentrum

The TOAR Data Commitment

TOAR is fully committed to

Open Data: all data collected in TOAR must be freely available with no use constraints

- CC-BY 4 license; citations&acknowledgeme

We won a 2nd price in the German **Open Data Impact Award** 2021

FAIRness: make it easy to find, access, interoperate and re-use TOAR data, also in other contexts

- elaborated metadata schema; REST-API; coordination with other data centres

Reproducibility: TOAR analyses shall be fully reproducible, even when the database grows

- use of PIDs, versioning, QC flags, FDOs

The TOAR Data REST API

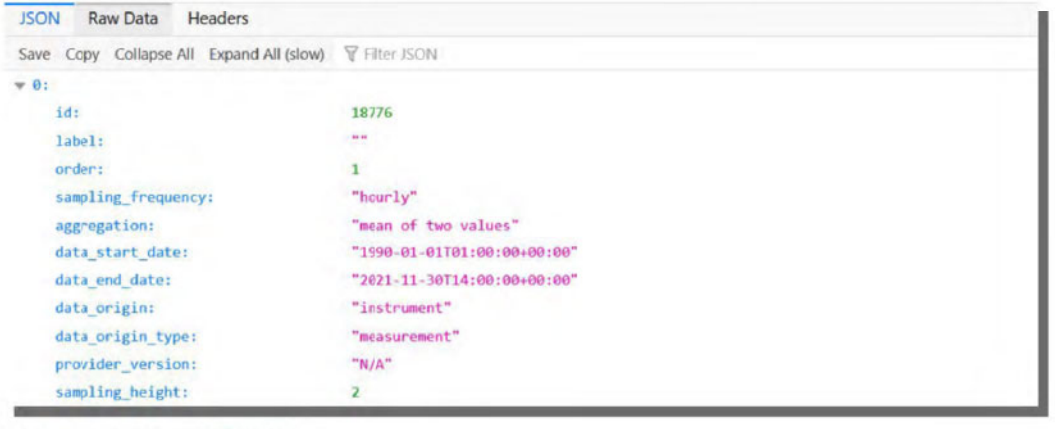
https://toar-data.fz-juelich.de/api/v2

2.6 Search (combined endpoint of stations and timeseries)

This endpoint enables queries that filter for a combination of metadata of stations and timeseries.
Any combination of query options (except for id) from both 2.4 Stationmeta and 2.5 Timeseries is possible here.

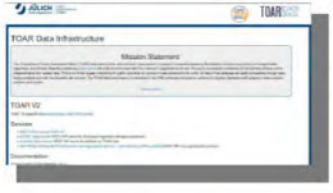
Example from the TOAR Quick Start Guide:
https://toar-data.fz-juelich.de/api/v2/search/?bounding_box=49,7,50,8&variable_id=5,4&limit=None

The above command will get you all the time series within an area between 49°N 7°E and 50°N 8°E that record ozone or pm1.



Summary of TOAR data resources

Data portal: https://toar-data.org



Access to data & documentation:
https://toar-data.fz-juelich.de

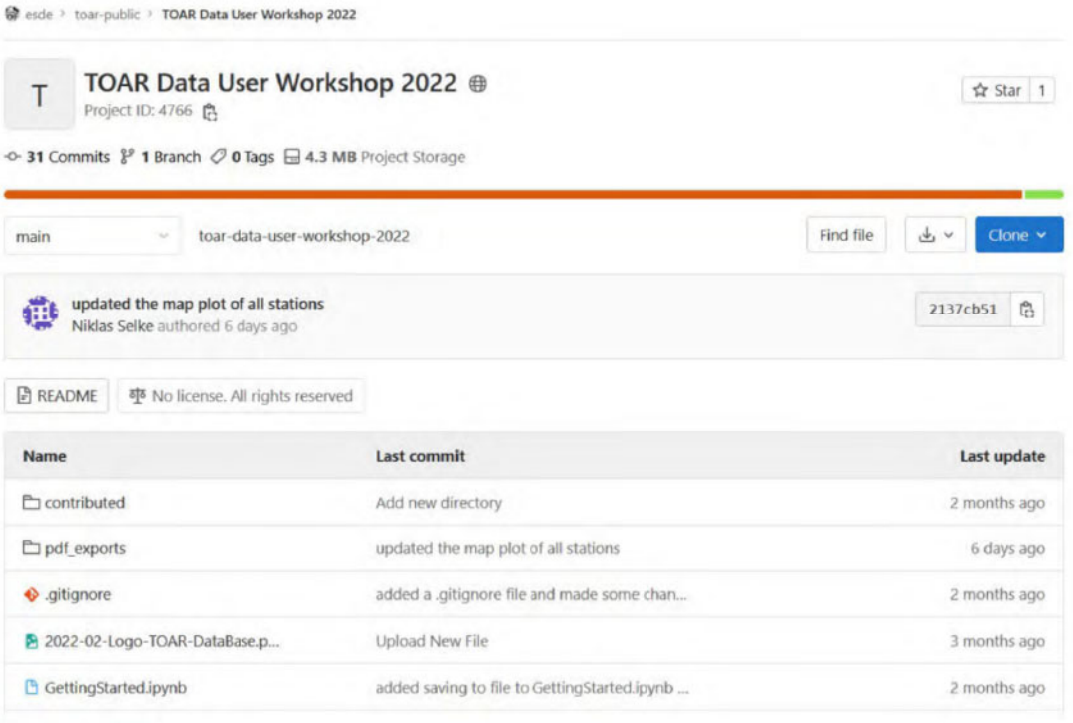
Training material:
https://go.fzj.de/toar-data-training
https://go.fzj.de/toar-user-git

TOAR tropospheric ozone assessment report Phase II

INTERNATIONAL GLOBAL ATMOSPHERIC CHEMISTRY IGAC

Jupyter notebooks for TOAR analyses

https://go.fzj.de/toar-user-git



BENEFITS

Why Jupyter is so popular among Data Scientists

Some of the reasons …

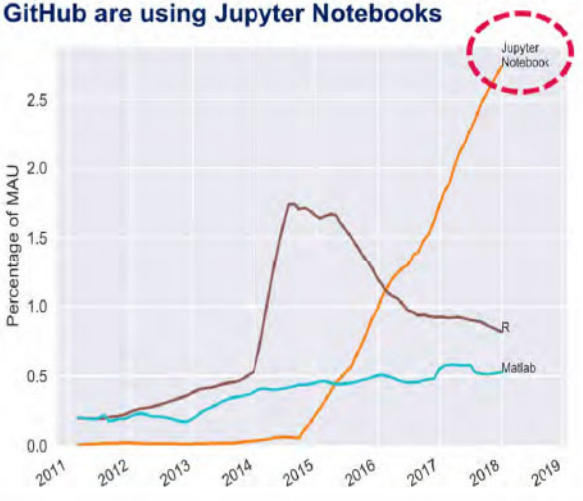
- Jupyter allows to **view the results of the code in-line** without the dependency of other parts of the code.
- Jupyter mixes easy for users who extend their code **line-by-line with feedback** attached all along the way
- Jupyter Notebooks support visualization and include rendering data in **live-graphics and charts**.
- Jupyter is maintaining the **state of execution of each cell** automatically.
- Supports IPyWidget packages, which provide **standard user interface** for exploring code and data interactively.
- Platform and language **independent** because of its representation in JSON format.

MOTIVATION

Rise of Jupyter´s popularity

- In 2007, Fernando Pérez and Brian Granger announced
  „**Ipython**: a system for interactive scientific computing“ [1]
- In 2014, Fernando Pérez announced
  a spin-off project from IPython called **Project Jupyter**.
  - IPython continued to exist as a Python shell and a kernel for Jupyter, while the Jupyter notebook moved under the Jupyter name.
- In 2015, GitHub and the Jupyter Project announced
  native rendering of Jupyter notebooks file format (.ipynb files) on the **GitHub**
- In 2017, the **first JupyterCon** was organized by O'Reilly in New York City.
  Fernando Pérez opened the conference with an inspiring talk. [2]
- In 2018, **JupyterLab** was announced
  as the next-generation web-based interface for Project Jupyter.
- In 2019, JupyterLab 1.0 ...
  In 2020, JupyterLab 2.0 ...
  In 2021, JupyterLab 3.0 ...

Counting how many Monthly Active Users (MAU) on GitHub are using Jupyter Notebooks

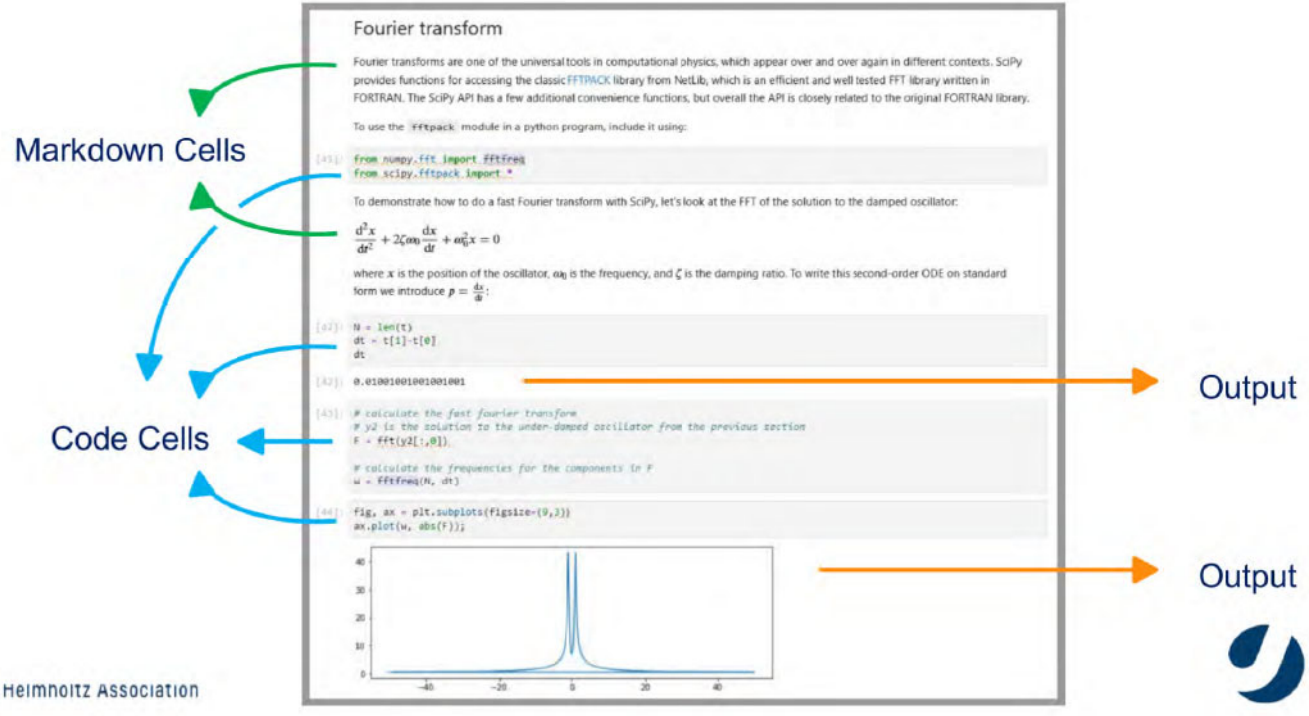


https://www.benfrederickson.com/ranking-programming-languages-by-github-users/
https://github.com/benfred/github-analysis

[1] Pérez F, Granger BE (2007) Ipython: a system for interactive scientific computing. Comput Sci Eng 9(3):21–29
[2] Pérez F, Project Jupyter: From interactive Python to open science -> https://www.youtube.com/watch?v=xuNj5paMuow

JUPYTER NOTEBOOK

creating reproducible computational narratives



TERMINOLOGY

What is JupyterLab

JupyterLab

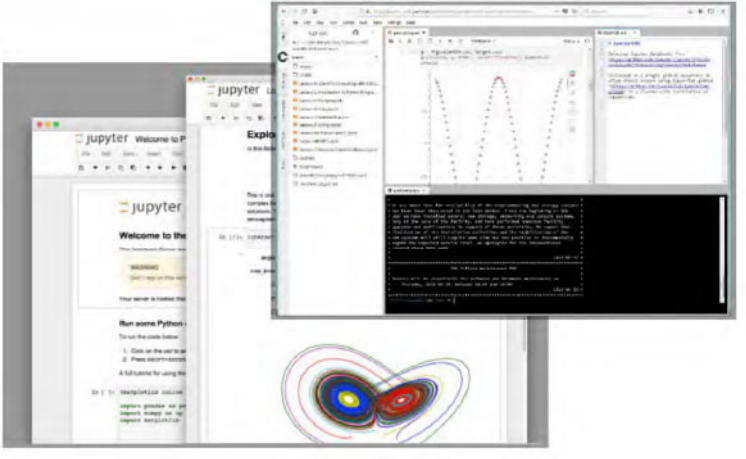
- **Interactive** working environment in the web browser
- For the creation of **reproducible** computer-aided narratives
- Very **popular** with researchers from all fields
- Jupyter = Julia + Python + R

Multi-purpose working environment

- Language agnostic
- Supports execution environments ("*kernels*")
  - For dozens of languages: Python, R, Julia, C++, ...
- Extensible software design („*extensions*")
  - many server/client plug-ins available
  - Eg. in-browser-terminal and file-browsing

Document-Centered Computing ("*notebooks*")

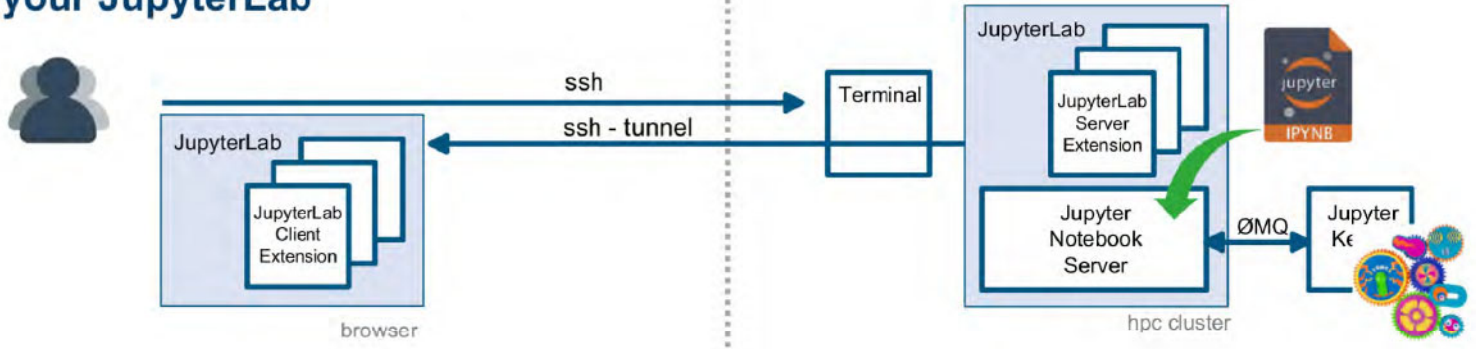
- Combines code execution,
  rich text, math, plots and rich media.
- All-in-one document called Jupyter Notebook



https://jupyterlab.readthedocs.io

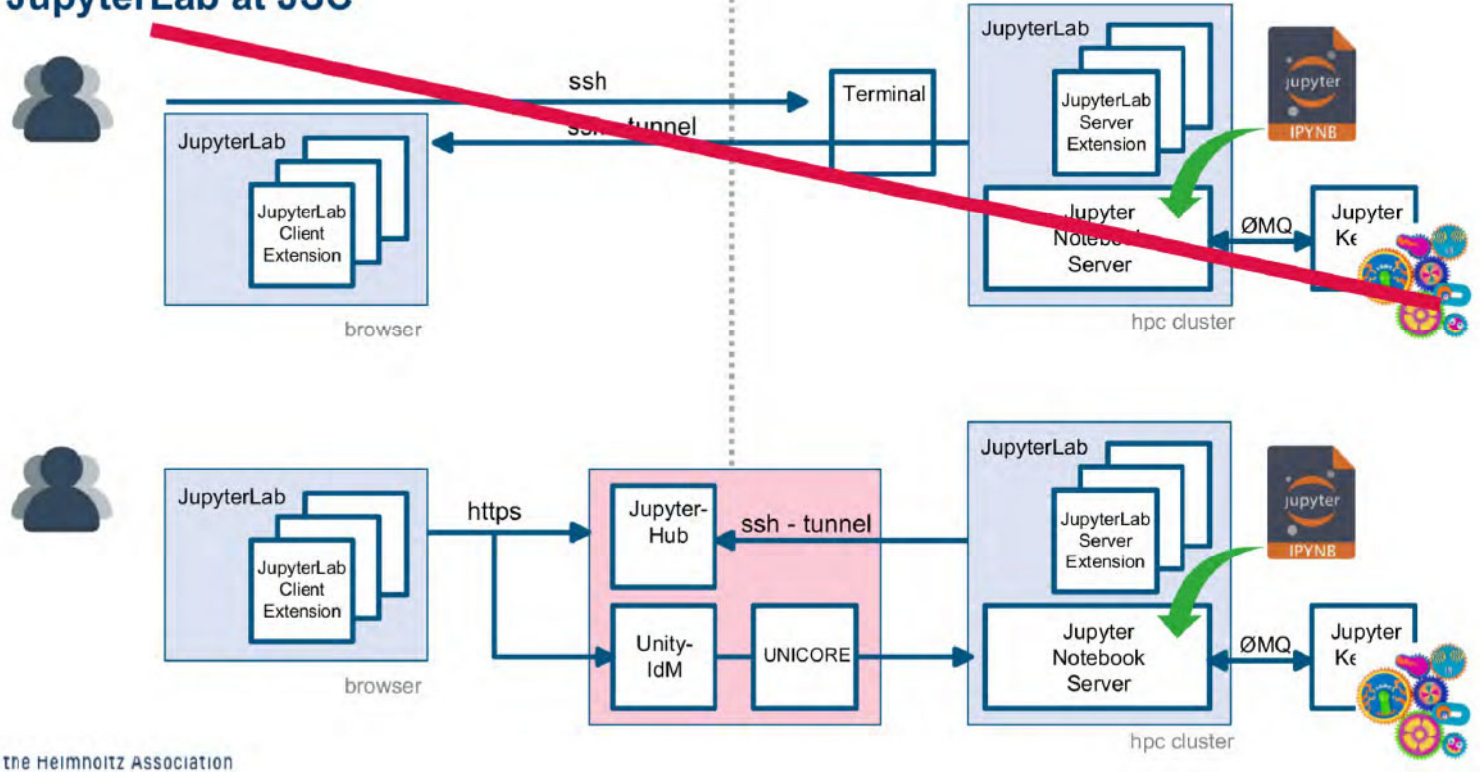
JUPYTER-JSC WEBSERVICE

Start your JupyterLab



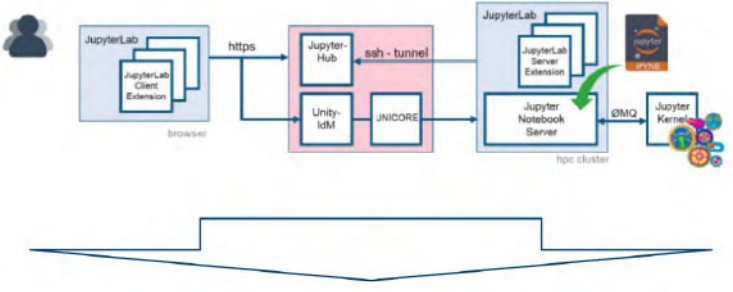
JUPYTER-JSC WEBSERVICE

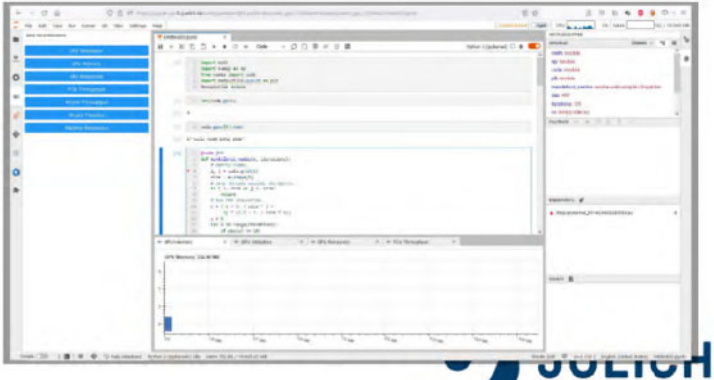
Start JupyterLab at JSC



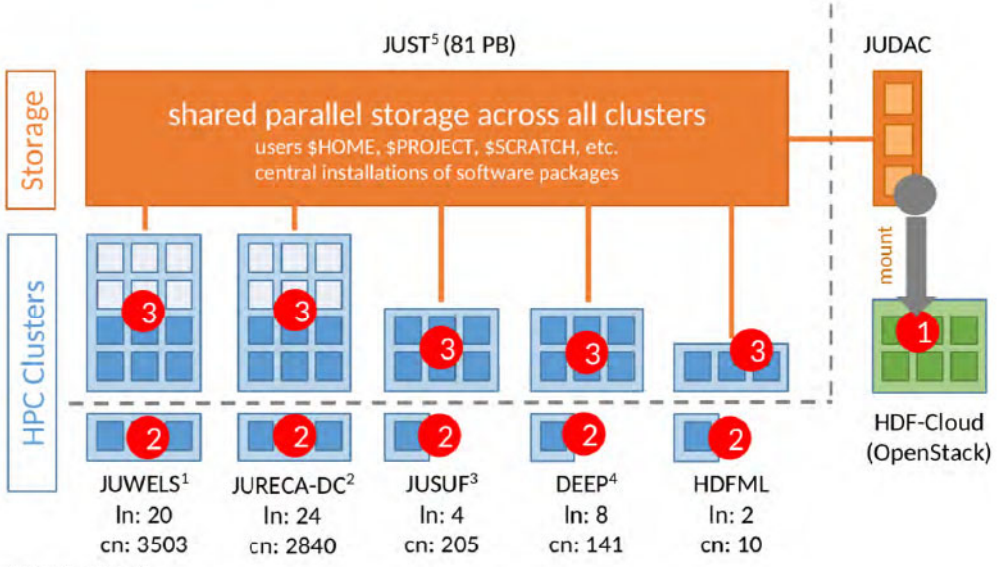
JUPYTER-JSC WEBSERVICE

Start JupyterLab at JSC





JUPYTERLAB EVERYWHERE

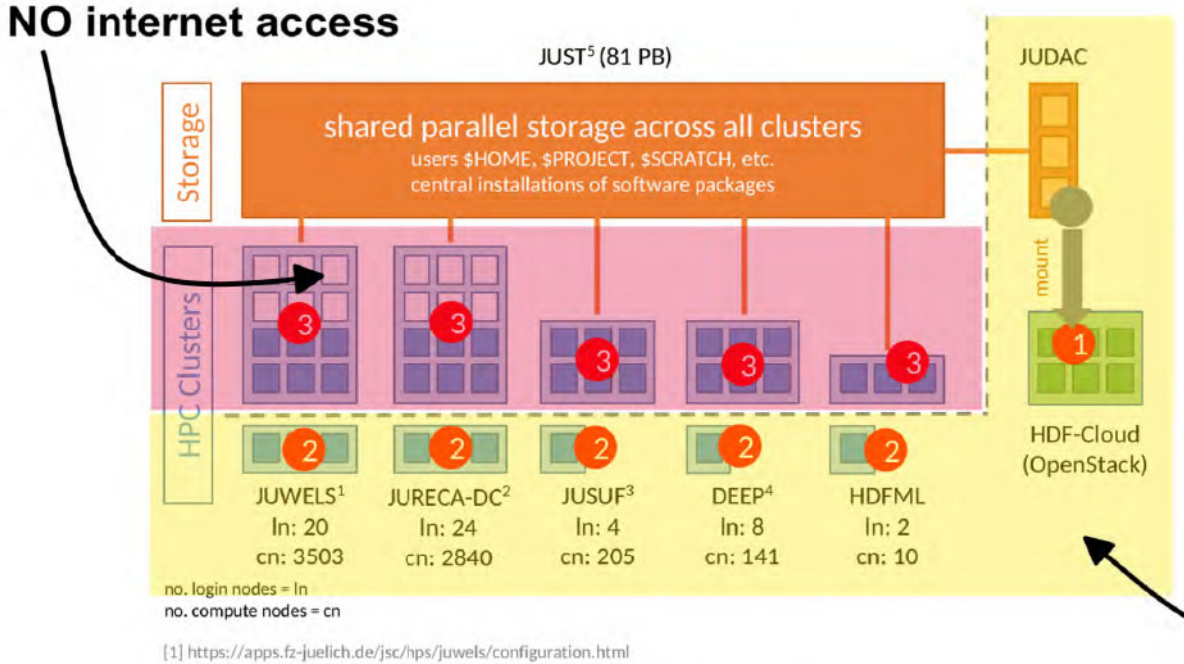


JupyterLab everywhere

1. JupyterLab on HDF-Cloud
2. JupyterLab on login nodes
3. JupyterLab on compute nodes

[1] https://apps.fz-juelich.de/jsc/hps/juwels/configuration.html
[2] https://apps.fz-juelich.de/jsc/hps/jureca/configuration.html
[3] https://apps.fz-juelich.de/jsc/hps/jusuf/cluster/configuration.html
[4] https://www.fz-juelich.de/ias/jsc/EN/Expertise/Supercomputers/DEEP-EST/_node.html
[5] https://www.fz-juelich.de/ias/jsc/EN/Expertise/Datamanagement/OnlineStorage/JUST/Configuration/Configuration_node.html

JUPYTERLAB EVERYWHERE

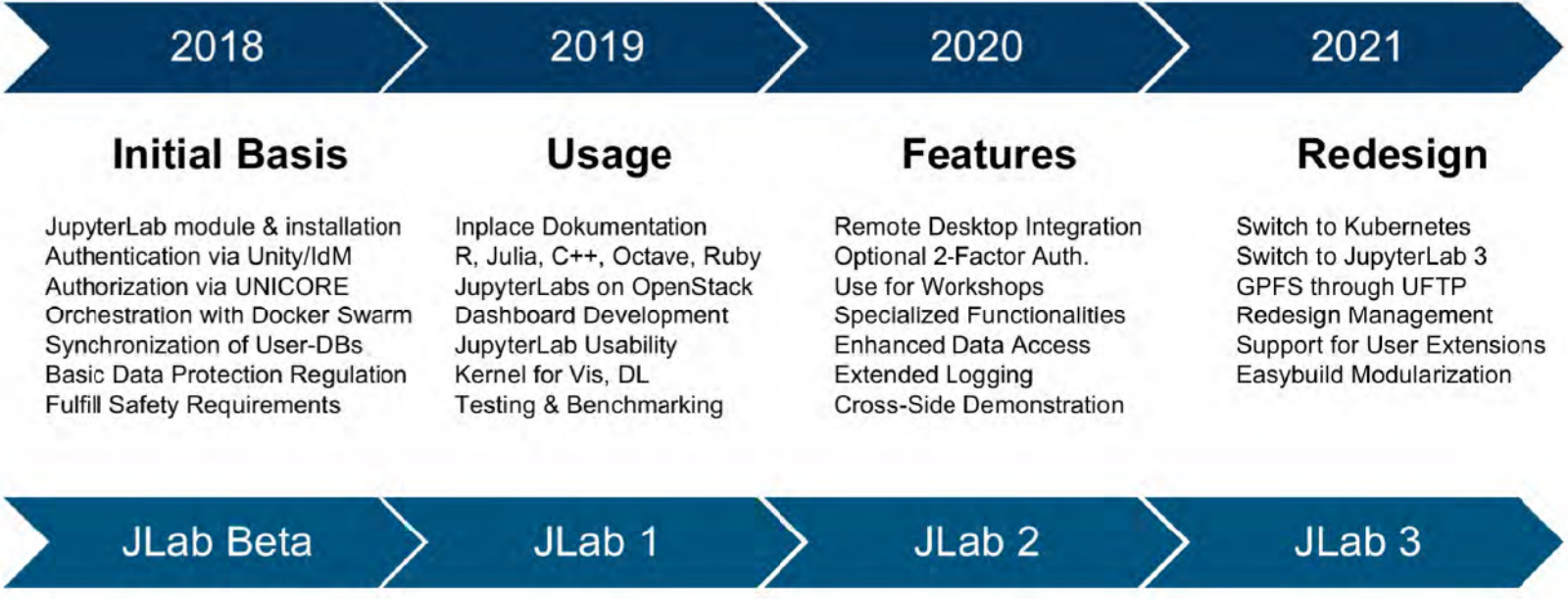


JupyterLab everywhere

1. JupyterLab on HDF-Cloud
2. JupyterLab on login nodes
3. JupyterLab on compute nodes

[1] https://apps.fz-juelich.de/jsc/hps/juwels/configuration.html
[2] https://apps.fz-juelich.de/jsc/hps/jureca/configuration.html
[3] https://apps.fz-juelich.de/jsc/hps/jusuf/cluster/configuration.html
[4] https://www.fz-juelich.de/ias/jsc/EN/Expertise/Supercomputers/DEEP-EST/_node.html
[5] https://www.fz-juelich.de/ias/jsc/EN/Expertise/Datamanagement/OnlineStorage/JUST/Configuration/Configuration_node.html

HISTORY OF JUPYTERLAB AT JSC

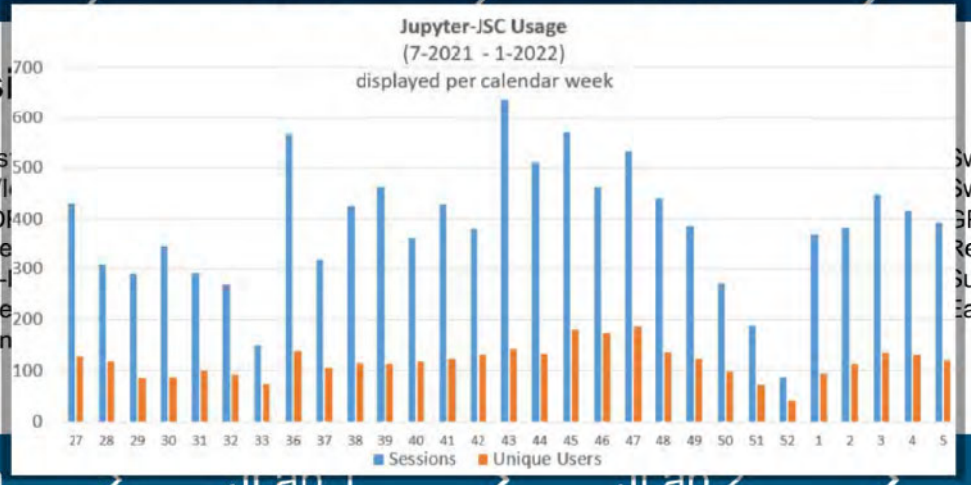


HISTORY OF JUPYTERLAB AT JSC

2018 | 2019 | 2020 | 2021

Initial Basi

JupyterLab module & ins
Authentication via Unity/I
Authorization via UNICOI
Orchestration with Docke
Synchronization of User-I
Basic Data Protection Re
Fulfill Safety Requiremen



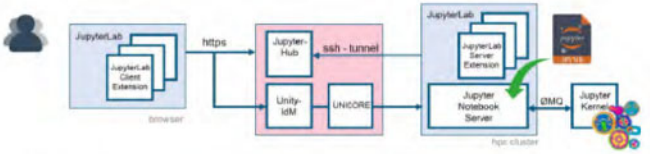
Redesign

Switch to Kubernetes
Switch to JupyterLab 3
GPFS through UFTP
Redesign Management
Support for User Extensions
Easybuild Modularization

JLab Beta | JLab 1 | JLab 2 | JLab 3

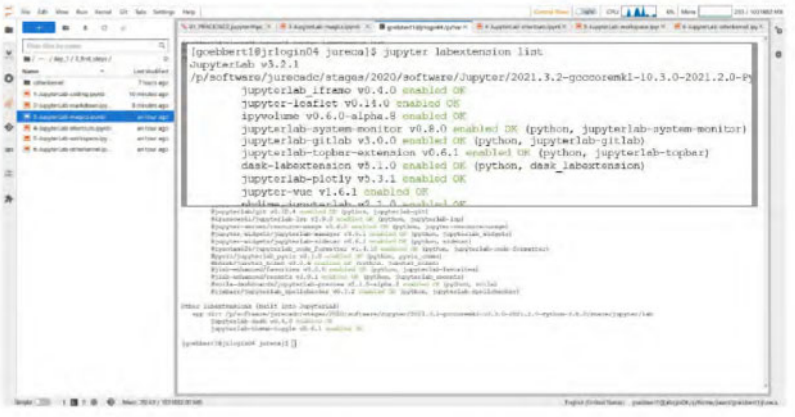
JUPYTER EXTENSIONS

Some general information



List the installed JupyterLab extensions

- Open the Launcher
- Start a Terminal
- Run command `jupyter labextension list`



https://jupyterlab.readthedocs.io/en/stable/user/extensions.html

Extensions are installed in JupyterLab´s Application Directory, which

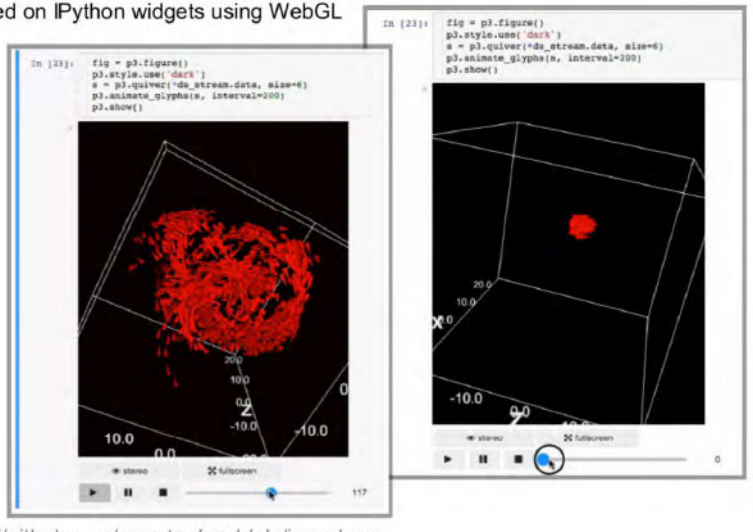
- stores any information that JupyterLab persists
  - including settings and built assets of extensions
- default location is `<sys-prefix>/share/jupyter/lab`
- can be relocated by setting $JUPYTERLAB_DIR
  - contains the JupyterLab static assets
    - (e.g. `static/index.html`)
  - **JupyterLab < 3:**
    any change requires a rebuild of the whole JupyterLab to take effect!
  - **JupyterLab >= 3:**
    introduced prebuild extensions, which are loaded at startup time

JUPYTER-JSC EXTENSIONS

Installed by default

IPyVolume

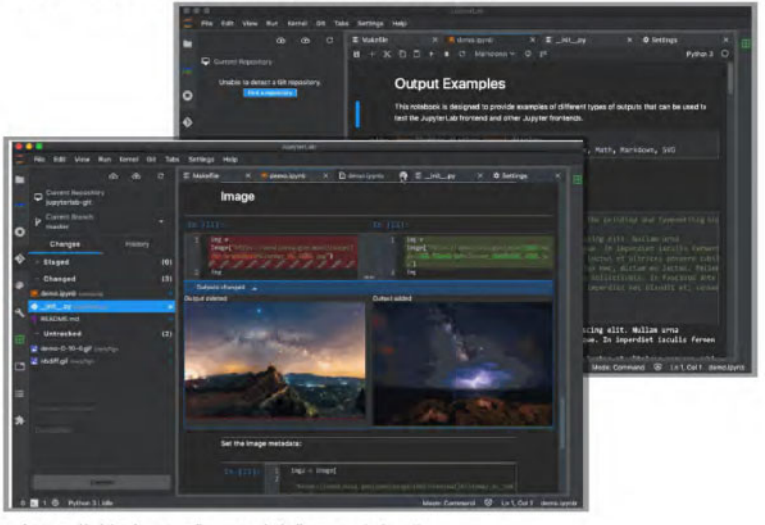
3d plotting for Python in the Jupyter notebook based on IPython widgets using WebGL



https://github.com/maartenbreddels/ipyvolume

JupyterLab-Git

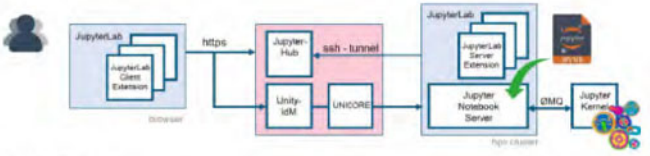
JupyterLab extension for version control using Git



https://github.com/jupyterlab/jupyterlab-git

JUPYTER-JSC EXTENSIONS

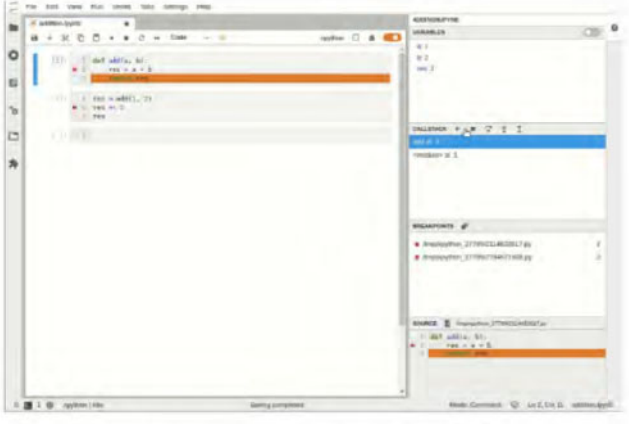
Installed by default



JupyterLab - Visual Debugger

JupyterLab 3.0 now ships with a Debugger front-end by default.

This means that notebooks, code consoles and files can now be debugged from JupyterLab directly! For the debugger to be enabled and visible, a kernel with support for debugging is required.

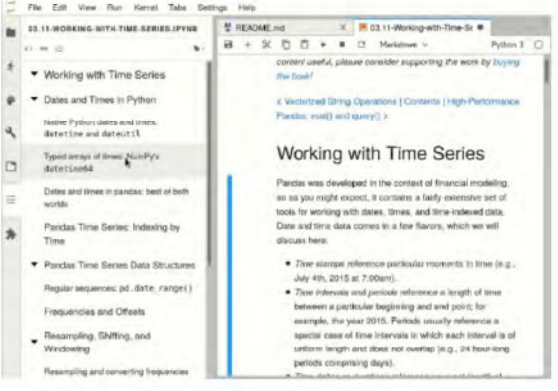


https://jupyterlab.readthedocs.io/en/stable/user/debugger.html

JupyterLab-toc

A Table of Contents extension for JupyterLab.

This auto-generates a table of contents in the left area when you have a notebook or markdown document open.The entries are clickable, and scroll the document to the heading in question.

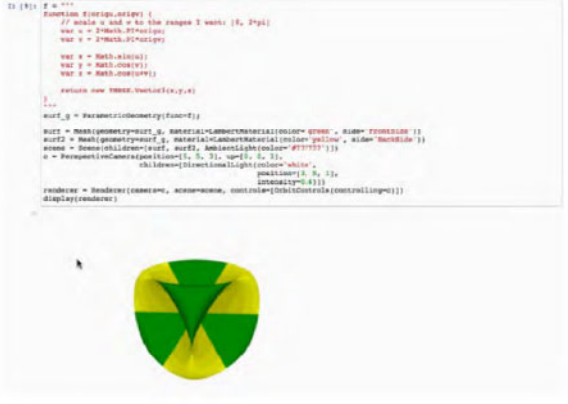


https://github.com/jupyterlab/jupyterlab-toc

JUPYTER-JSC EXTENSIONS

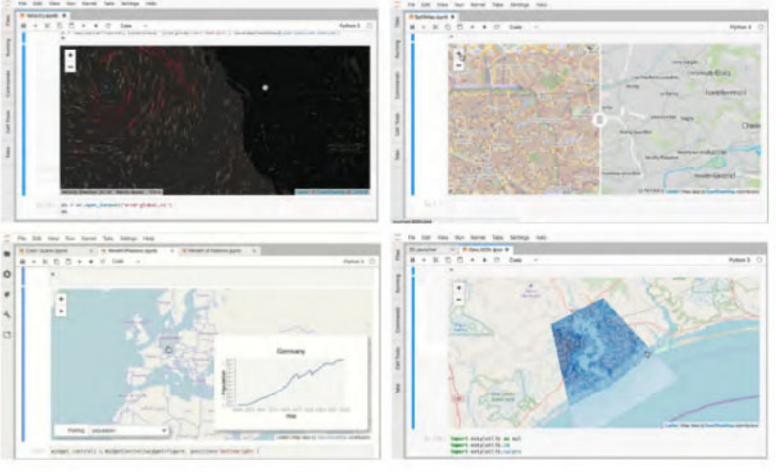
Installed by default

PyThreeJS
A Python / ThreeJS bridge utilizing the Jupyter widget infrastructure.
https://threejs.org - lightweight, 3D library with a default WebGL renderer.



https://github.com/jupyter-widgets/pythreejs

IPyLeaflet
A Jupyter / Leaflet bridge enabling interactive maps in the Jupyter notebook.

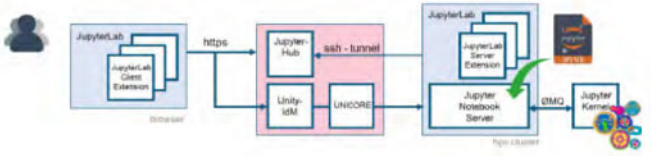


https://github.com/jupyter-widgets/ipyleaflet

… and more

JUPYTER KERNEL

How to create your own Juypter Kernel

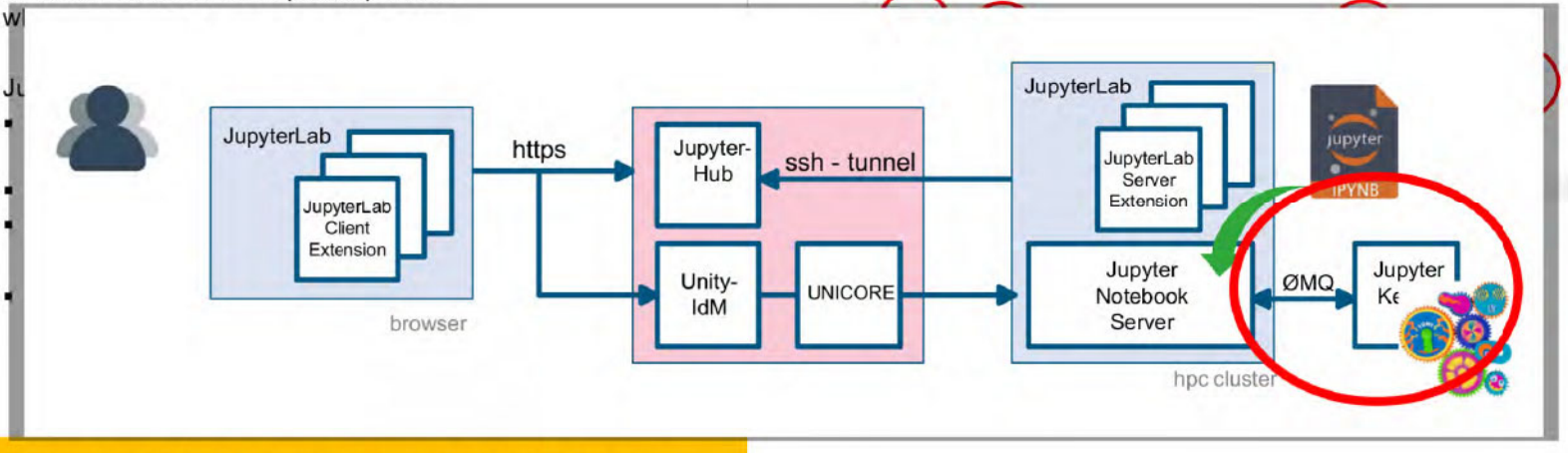


Jupyter Kernel

A “kernel” refers to the separate process

w

Ju

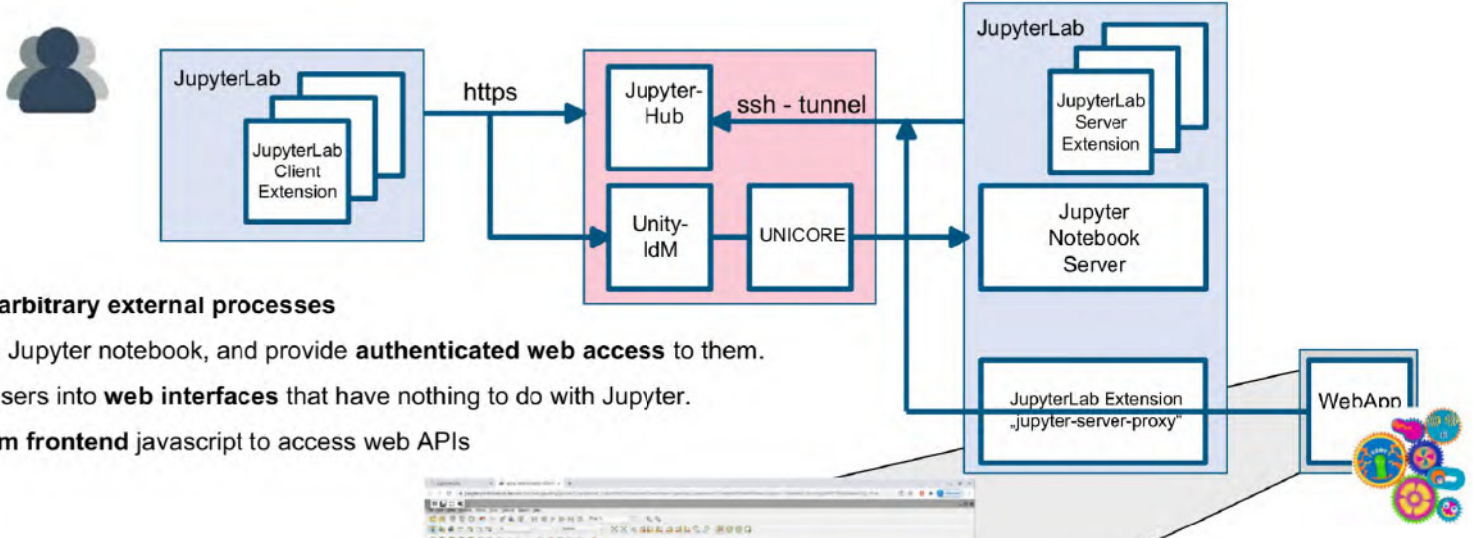


You can easily **create your own kernel** which for example runs your specialized virtual Python environment.

https://github.com/jupyter/jupyter/wiki/Jupyter-kernels

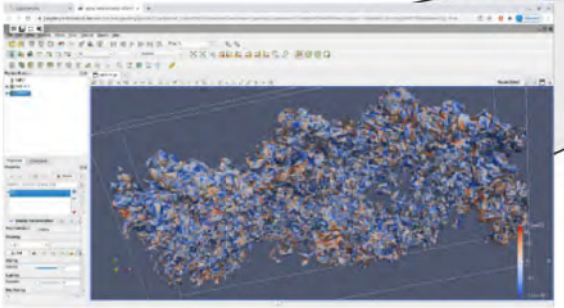
JUPYTERLAB – WEBSERVICE PROXY

Extension: jupyter-server-proxy



Allows to run **arbitrary external processes**

- alongside a Jupyter notebook, and provide **authenticated web access** to them.
- launching users into **web interfaces** that have nothing to do with Jupyter.
- **access from frontend** javascript to access web APIs



Turbulent mixing with variable density,
subset of 1939x600x3584 grid points, Michael Gauding, CORIA

https://github.com/jupyterhub/jupyter-server-proxy

JÜLICH Forschungszentrum

QUESTIONS?

https://jupyter-jsc.fz-juelich.de