Jupyter * @ DKRZ

Interactive (Super) computing on Mistral

Dr. Sofiane Bendoukha

Deutsches Klimarechenzentrum (DKRZ)

* : hub/notebooks/lab/kernels

About me

- **Background**: Computer science
- **Department**: Application Support
- **Currently:**
  - backend developer –> interactive computing
  - containers

Introduction

```
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
```

What is, Why Jupyter? (1)

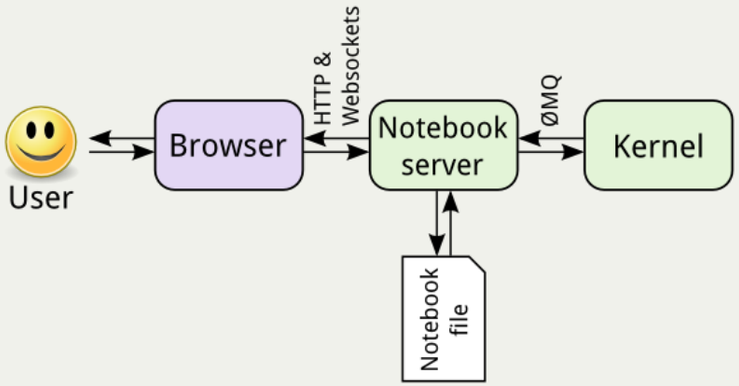
Python is popular

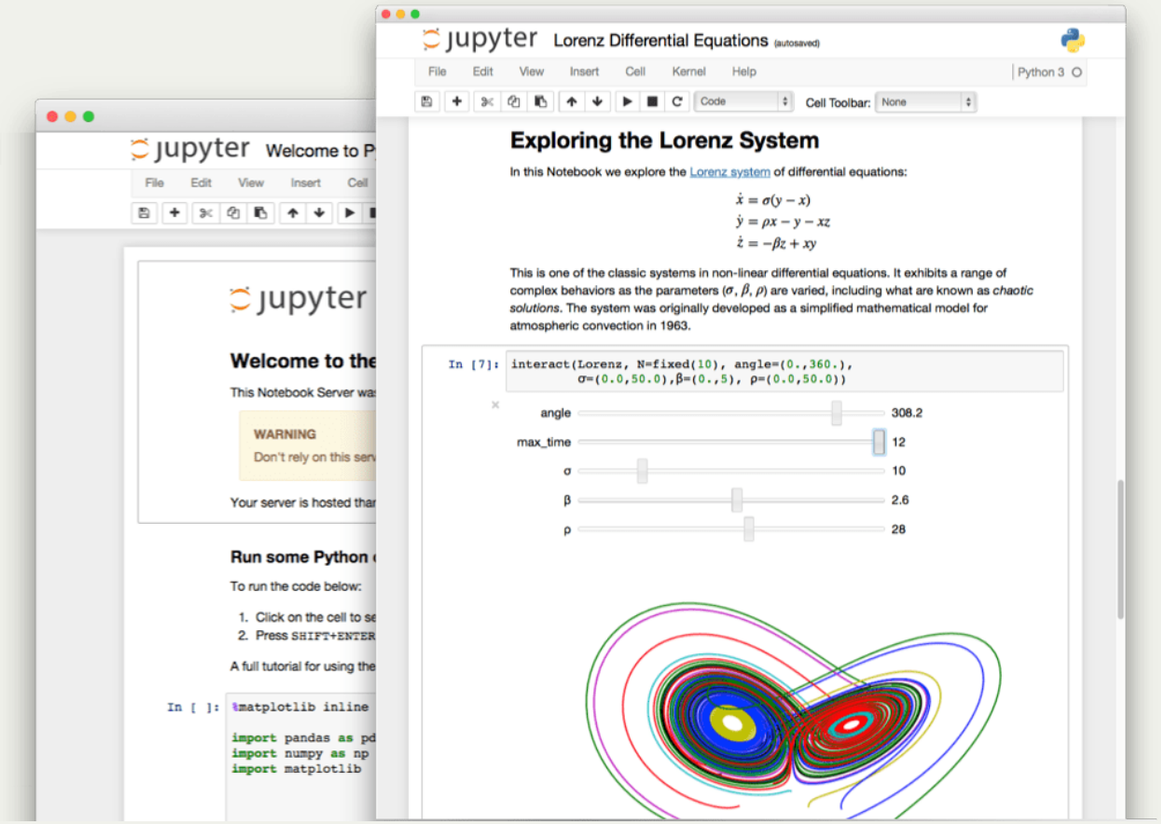
Aug 2020	Aug 2019	Change	Programming Language	Ratings	Change
1	2	▲	C	16.98%	+1.83%
2	1	▼	Java	14.43%	-1.60%
3	3		Python	9.69%	-0.33%
4	4		C++	6.84%	+0.78%

https://www.tiobe.com/tiobe-index/

Programming Language	2020	2015	2010	2005	2000	1995	1990	1985
Java	1	2	1	2	3	-	-	-
C	2	1	2	1	1	2	1	1
Python	3	7	6	6	22	20	-	-

What is, Why Jupyter? (2)





web application that allows you to **create** and **share** documents that contain **live code**, equations, **visualizations**.

Jupyter Notebook on HPC

- **Local**:
  Using Anaconda, anyone can install and run Jupyter Notebooks on their local computer.
- **HPC**: Infrastructure-specific
  - security concerns
  - shared file systems
  - metadata transactions

Jupyter Notebook on HPC

- **Local**:
  Using Anaconda, anyone can install and run Jupyter Notebooks on their local computer.
- **HPC**: Infrastructure-specific
  - security concerns
  - shared file systems
  - metadata transactions

```
1. SSH setup
2. Jupyter notebook setup
3. SSH to the remote system and start Jupyter notebook
4. Start Jupyter notebook with --no-browser and --port
5. Create an SSH "local port forward"
6. Open Jupyter notebook with your "Local" browser
```

What we provide @ DKRZ

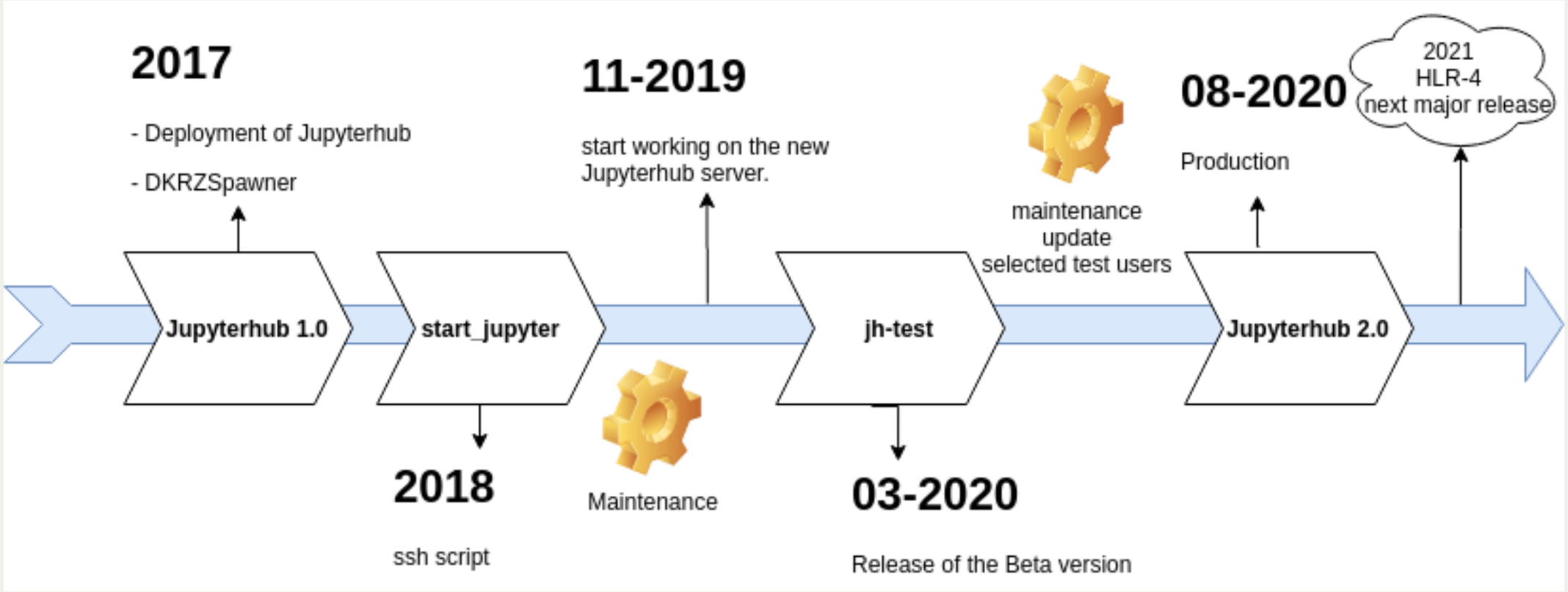
What we provide @ DKRZ

- **Convenient way**: *Jupyterhub*
  - user-friendly
  - full user support
  - continuous maintenance and update

What we provide @ DKRZ

- **Convenient way**: *Jupyterhub*
  - user-friendly
  - full user support
  - continuous maintenance and update
- **Old school:** *Single Jupyter notebooks* (ssh based)
  - `./start_jupyter [Options]`
  - limited support (?)
  - port forwarding can be annoying
  - **Jupyterhub advanced spawner?** 🤔

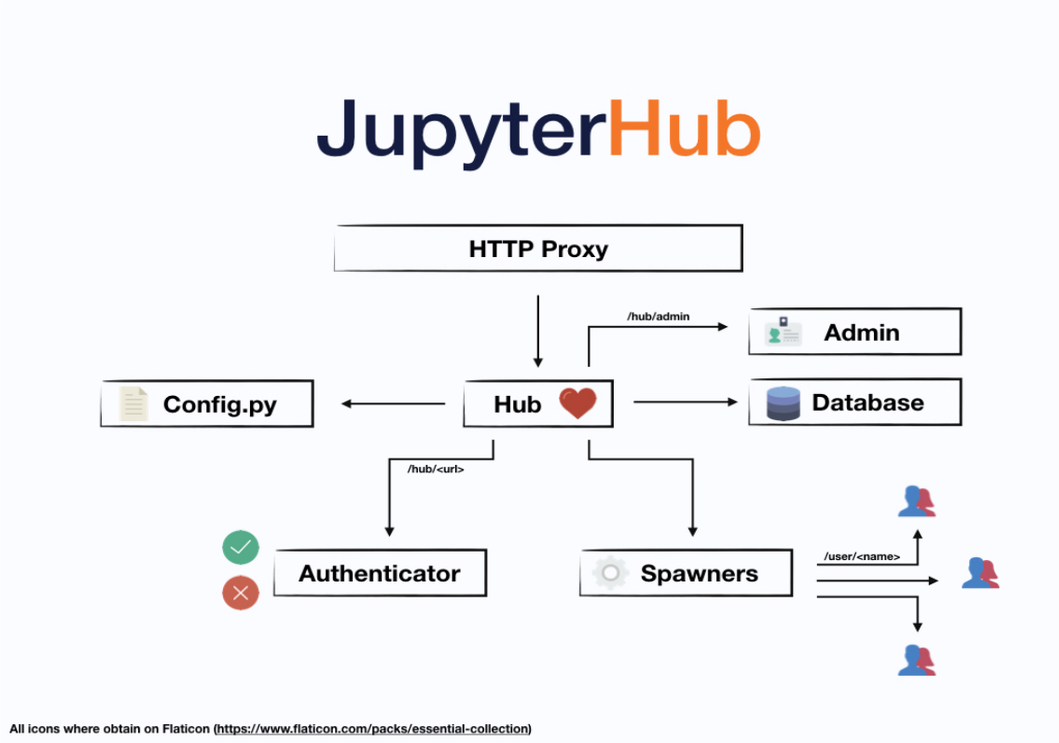
Roadmap


2017
- Deployment of Jupyterhub
- DKRZSpawner
Jupyterhub 1.0
start_jupyter
2018
ssh script
11-2019
start working on the new Jupyterhub server.
Maintenance
jh-test
03-2020
Release of the Beta version
maintenance update selected test users
08-2020
Production
Jupyterhub 2.0
2021
HLR-4
next major release

What's new?


Dynamic Spawner Options
Jupyter NBExtensions
Enabled user Python env
Sphinx Documentation
Managed Jupyterhub services
Enabled user Python env
Dynamic Spawner Options
Jupyter NBExtensions
Managed Jupyterhub services
DKRZSpawner
Managed Jupyterhub services
New Partitions
New Partitions
Dynamic Spawner Options
Jupyter NBExtensions
New Partitions
Sphinx Documentation
Sphinx Documentation
Enabled user Python env
Jupyter NBExtensions
New Frontend
New Partitions
New Partitions
Jupyter NBExtensions
New Frontend
Enabled user Python env
Managed Jupyterhub services
Sphinx Documentation
Sphinx Documentation
New Partitions
Jupyter NBExtensions
New Frontend
DKRZSpawner
DKRZSpawner
New Frontend
DKRZSpawner
DKRZSpawner
Jupyter NBExtensions
Jupyter NBExtensions
New Frontend
New Frontend
DKRZSpawner
New Frontend
Enabled user Python env
New Frontend
Enabled user Python env
Dynamic Spawner Options
Jupyter NBExtensions
DKRZSpawner
Sphinx Documentation
DKRZSpawner
New Frontend
Enabled user Python env
New Frontend
Enabled user Python env
DKRZSpawner
New Partitions
DKRZSpawner
Sphinx Documentation
Enabled user Python env
New Partitions
Managed Jupyterhub services
New Frontend
Enabled user Python env
Dynamic Spawner Options
Jupyter NBExtensions
Dynamic Spawner Options
Sphinx Documentation
Dynamic Spawner Options
Jupyter NBExtensions
New Partitions
Managed Jupyterhub services
Jupyter NBExtensions
Enabled user Python env
Jupyter NBExtensions
Sphinx Documentation
Managed Jupyterhub services
DKRZSpawner
Sphinx Documentation
New Partitions
New Frontend
Dynamic Spawner Options
Sphinx Documentation

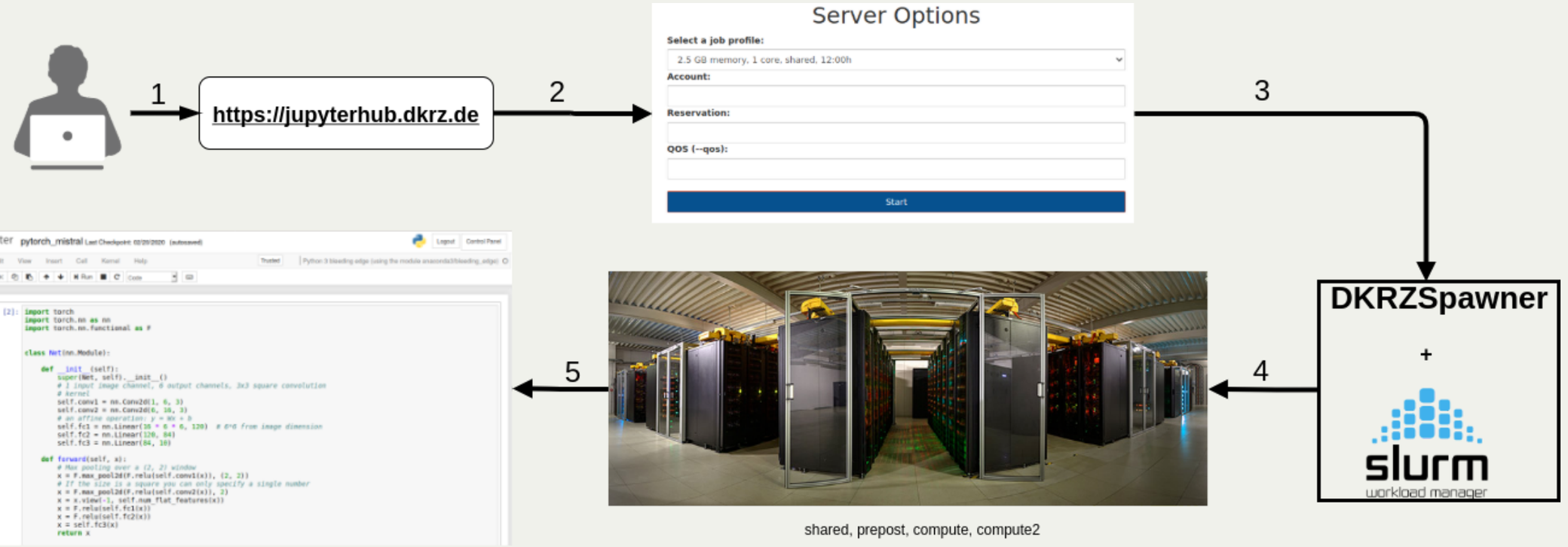
How it works?



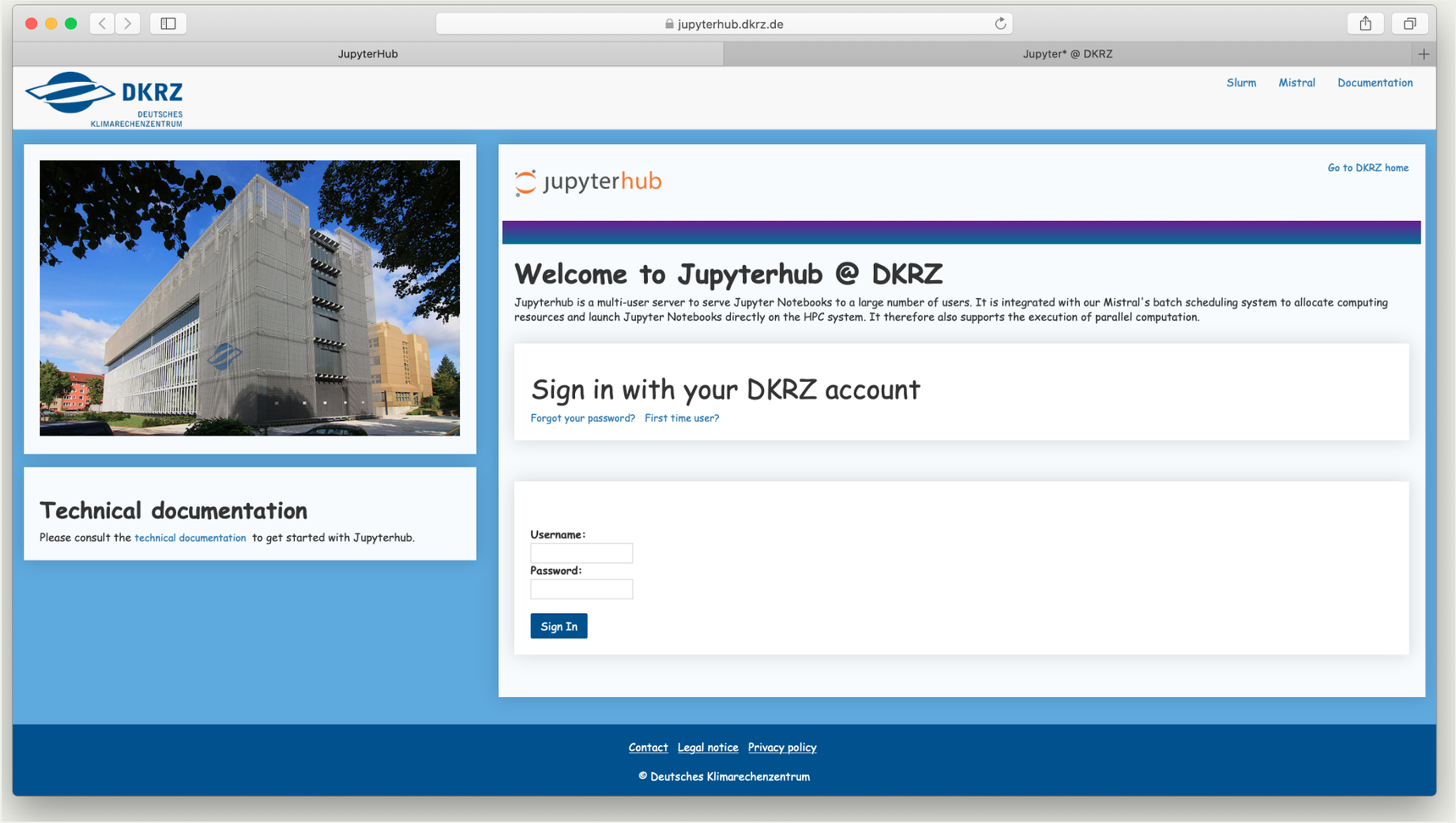
- *Manages authentication*
- *Spawns single-user notebook servers on-demand*
- *Gives each user a complete server*

How it works on Mistral?

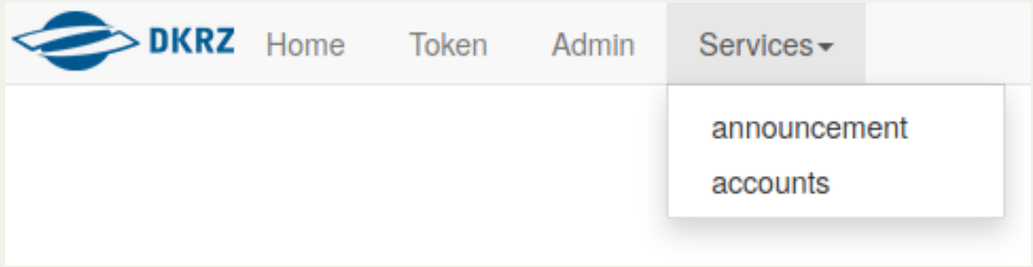
Spawning workflow



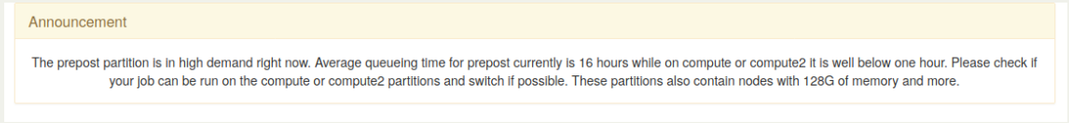
GUI


jupyterhub.dkrz.de
JupyterHub
Jupyter* @ DKRZ
DKRZ
DEUTSCHES KLIMARECHENZENTRUM
Slurm
Mistral
Documentation
jupyterhub
Go to DKRZ home
Welcome to Jupyterhub @ DKRZ
Jupyterhub is a multi-user server to serve Jupyter Notebooks to a large number of users. It is integrated with our Mistral's batch scheduling system to allocate computing resources and launch Jupyter Notebooks directly on the HPC system. It therefore also supports the execution of parallel computation.
Sign in with your DKRZ account
Forgot your password?
First time user?
Username:
Password:
Sign In
Technical documentation
Please consult the technical documentation to get started with Jupyterhub.
Contact
Legal notice
Privacy policy
© Deutsches Klimarechenzentrum

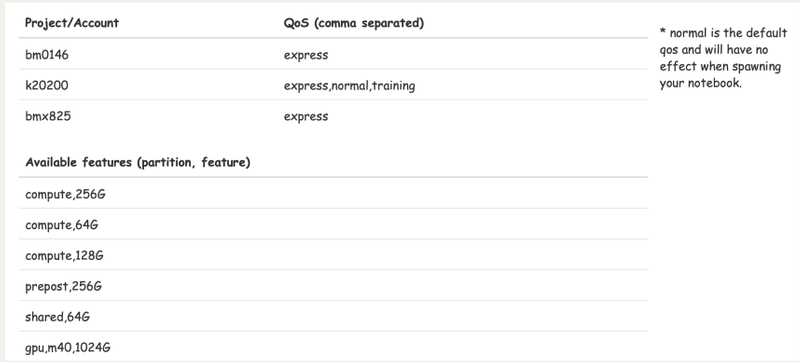
REST Web services



Announcements (Admin)



Accounts



Project/Account	QoS (comma separated)
bm0146	express
k20200	express,normal,training
bmx825	express

Available features (partition, feature)
compute,256G
compute,64G
compute,128G
prepost,256G
shared,64G
gpu,m40,1024G

* normal is the default qos and will have no effect when spawning your notebook.

Documentation

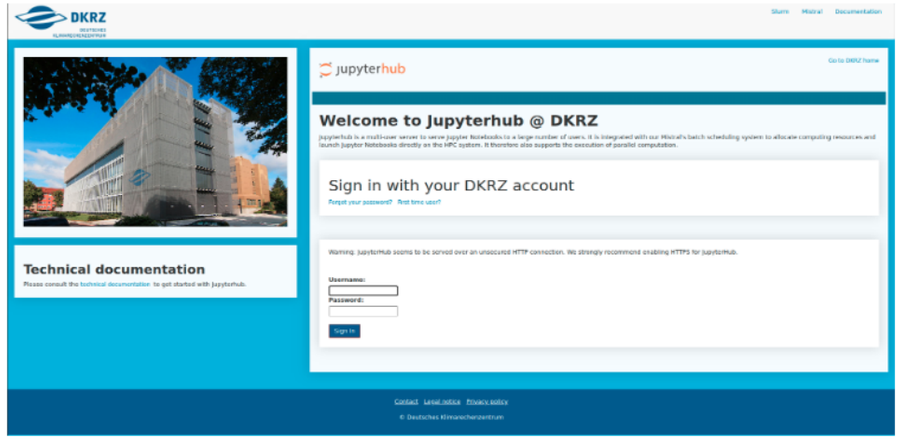
DKRZ

Overview Quick Start Spawner options Kernels Systems Tutorials Changelog DKRZ User Portal

Search the docs ...

Welcome to Jupyterhub @ DKRZ!

Jupyterhub is a multi-user server for Jupyter notebooks that allows to execute the notebook directly on the DKRZ HPC system Mistral. It therefore also supports the execution of parallel computation.



Note

JupyterHub is available at https://jupyterhub.dkrz.de for all DKRZ users who have access to Mistral and who are allowed to submit batch jobs.

Warning

DKRZ operates the jupyterhub server in extended testing phase, which means that we do not guarantee fulltime availabilty of the system and some more limitations documented here.

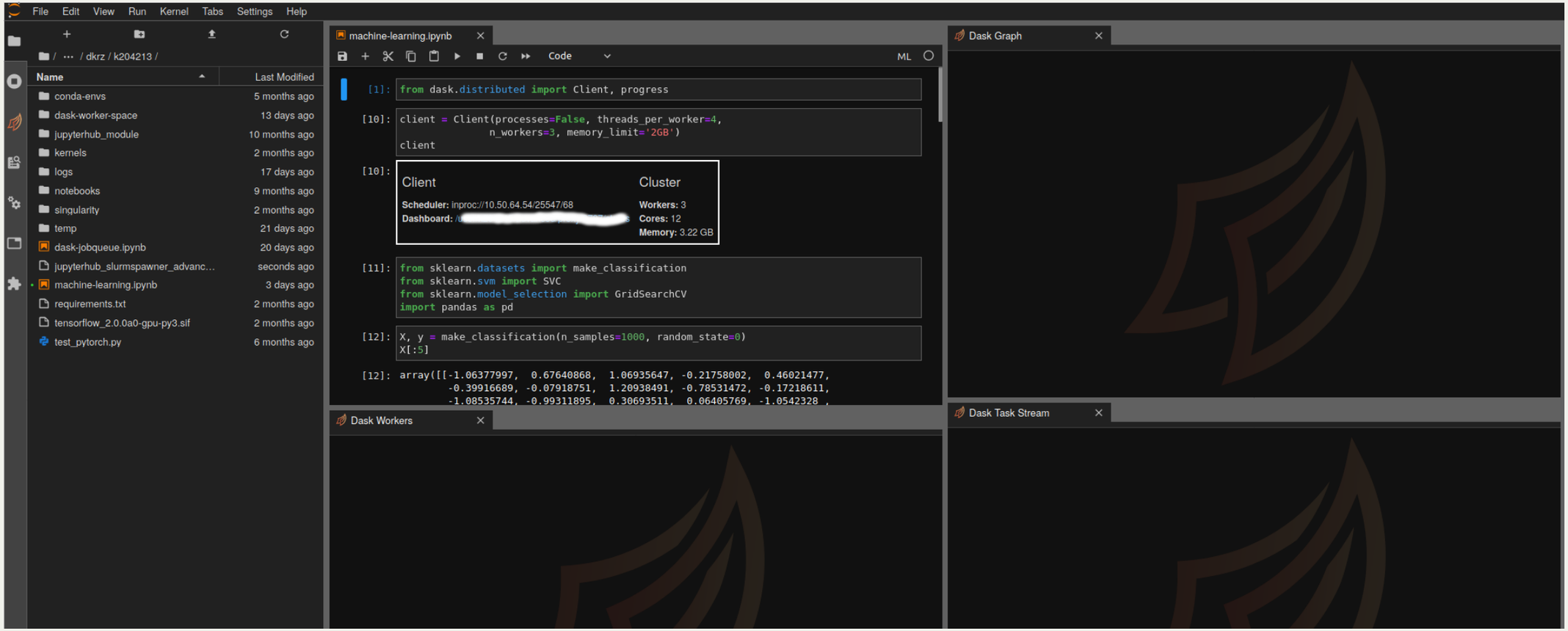
On this page

Welcome to Jupyterhub @ DKRZ!

Indices and tables

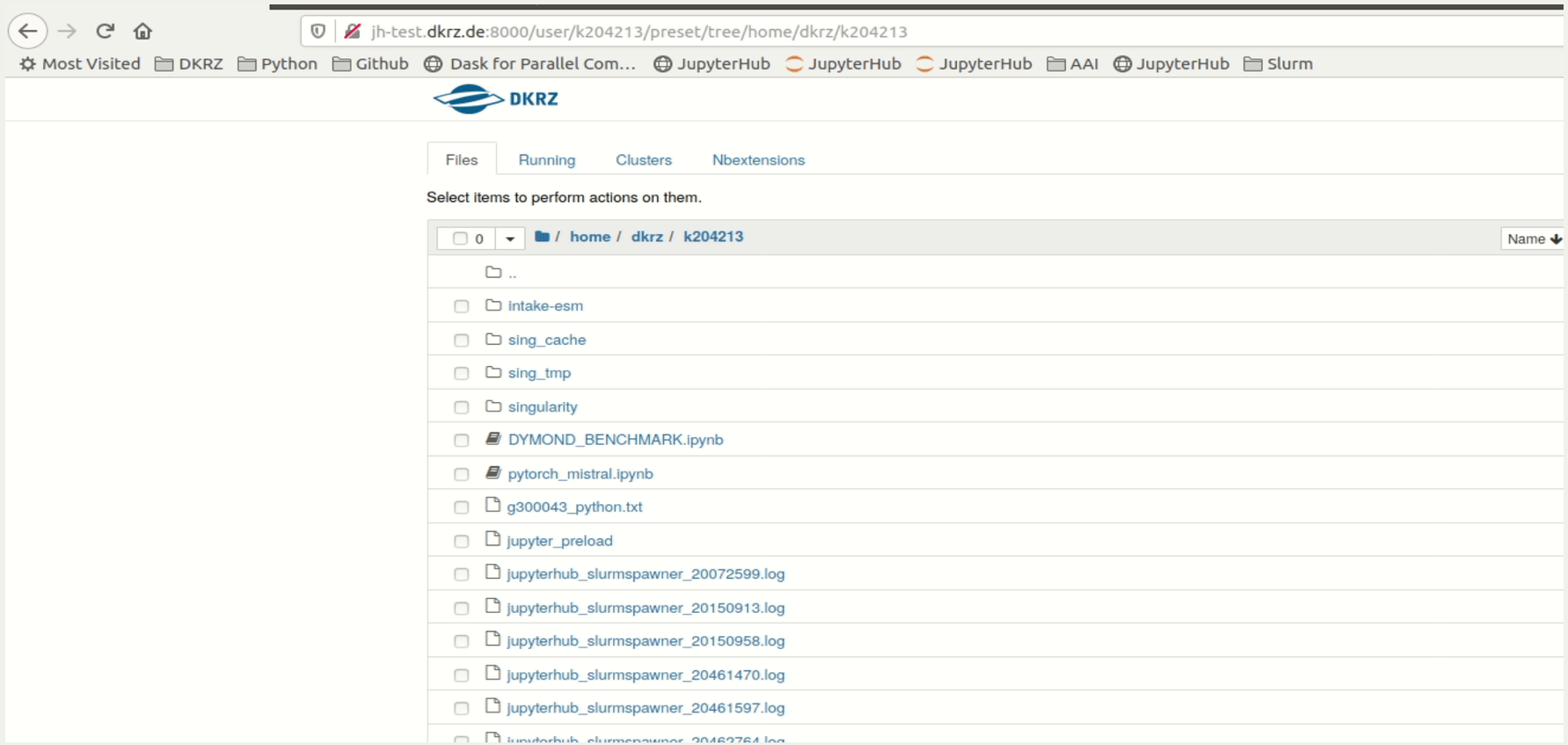
Any input/request is welcome!

JupyterLab (1)



JupyterLab (2)

switching between classic/lab

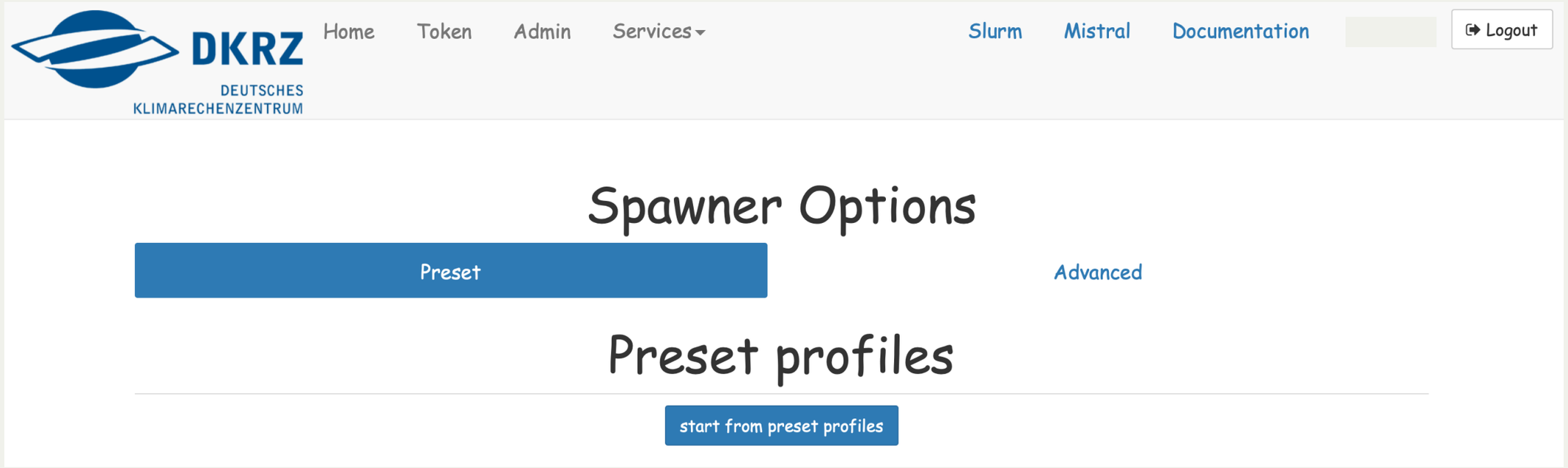


Agenda

- ~~Introduction~~
- Spawning process
- Kernels
- Extensions
- Q & A

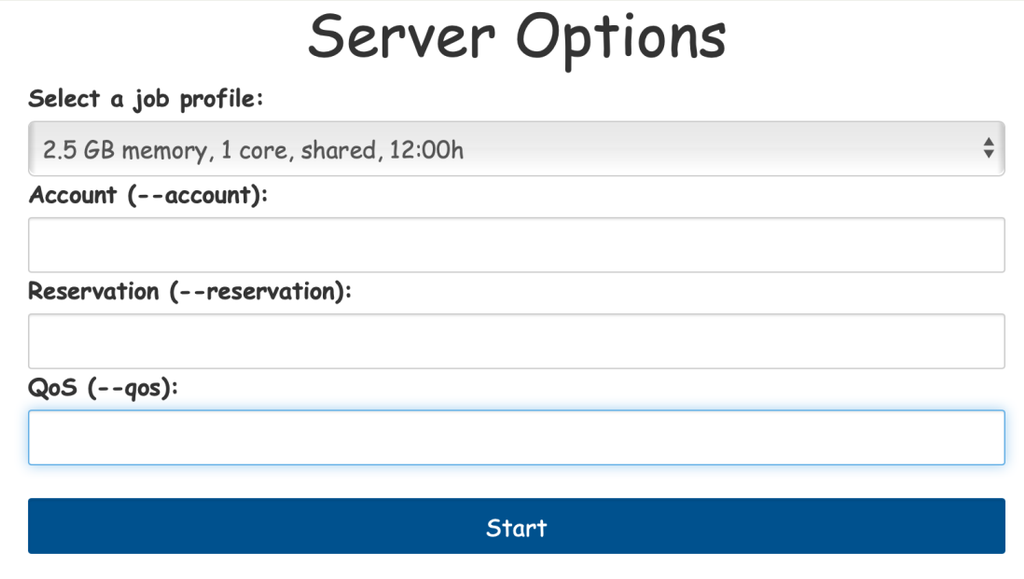
Spawner options

Overview

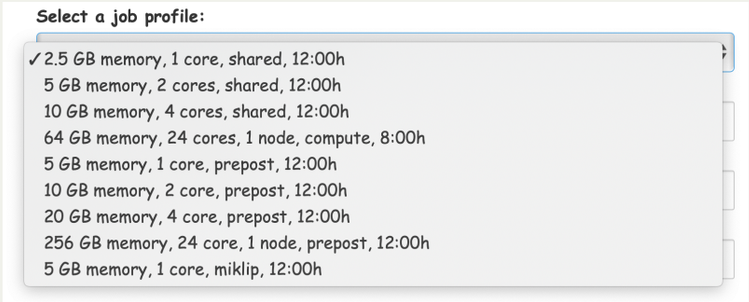

DKRZ
DEUTSCHES
KLIMARECHENZENTRUM
Home
Token
Admin
Services
Slurm
Mistral
Documentation
Logout
Spawner Options
Preset
Advanced
Preset profiles
start from preset profiles

Preset options form

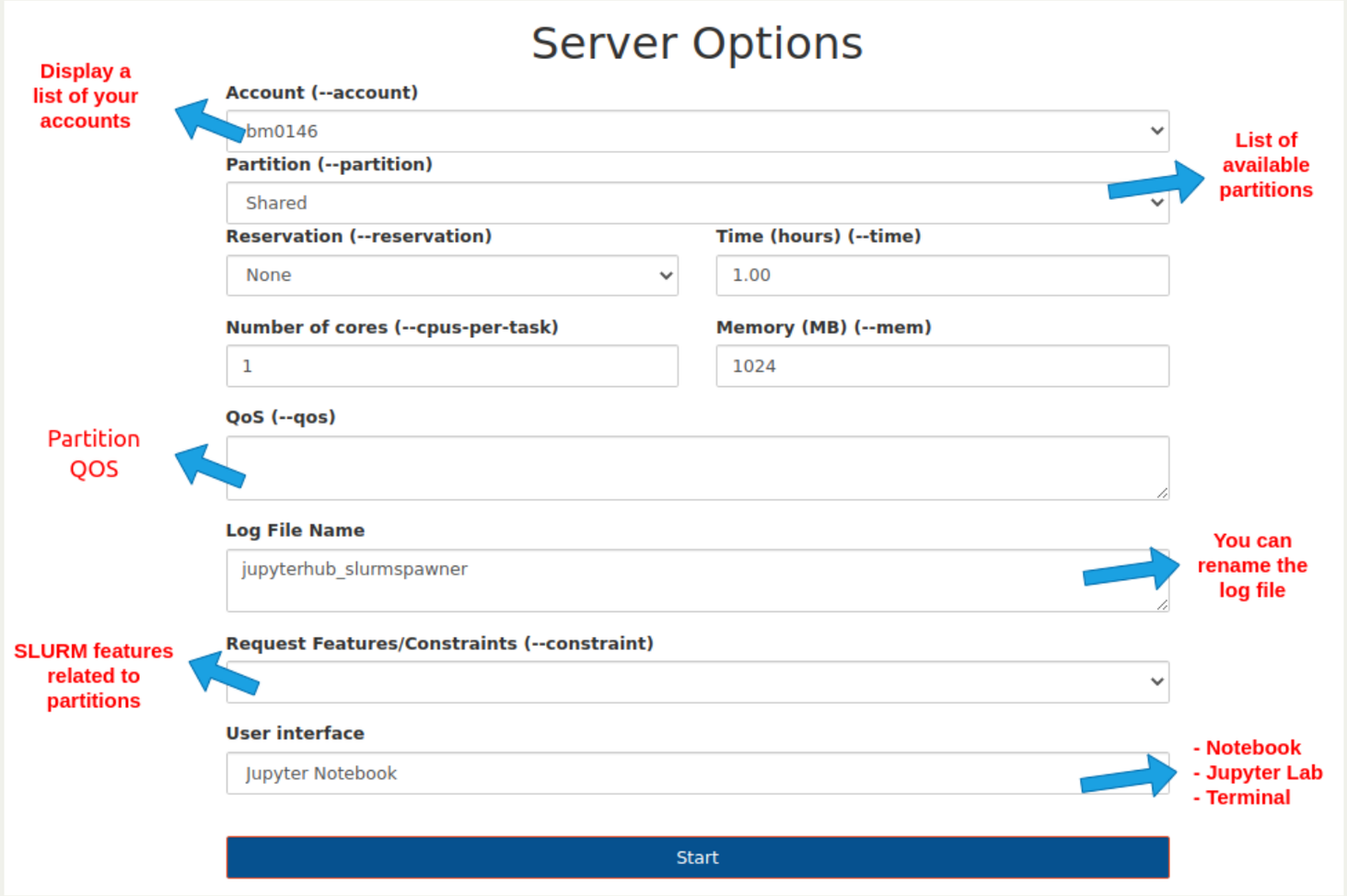
Choose from the list



Available profiles



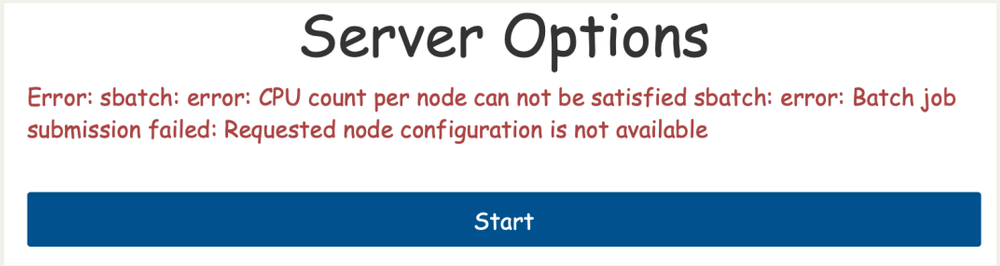
Advanced options form (1)

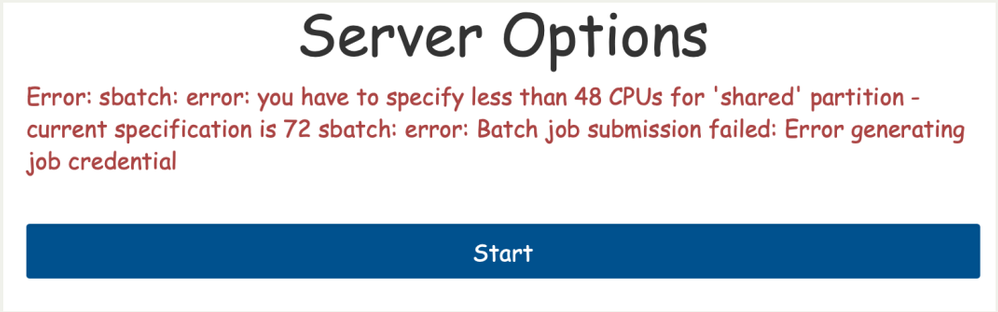


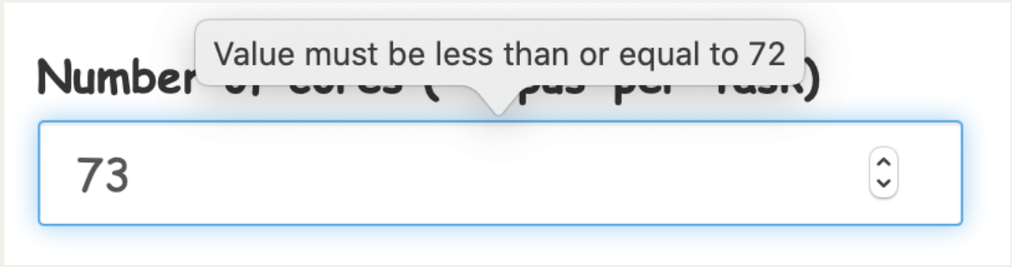
Advanced options form (2)

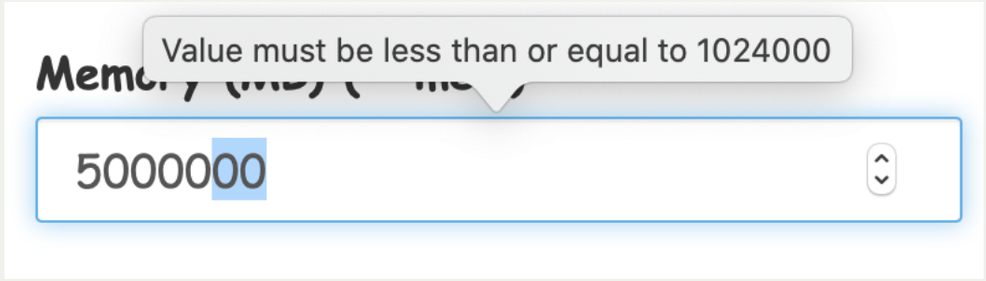
Parameter	Mandatory	Description
Account	Yes	Project that should be charged
Partition	Yes	Partition to run the job
Reservation	No	Resources reserved for certain time/accounts
Time	No (Default: 1 Hour)	The maximum amount of time your job can take before Slurm forcefully kills it.
Number of cores	No (Default: 1 core)	Number of threads (logical cores) per task.
Memory	No (Default: 1024 MB)	The total amount of RAM to allocate.
QoS	No	Quality of Service often puts the job in high priority queue (e.g. training).
Log File Name	No (Default: `jupyterhub_slurmspawner_advanced`)	Notebook log
Request Features	No	Node-features requested for the job.
User interfaces	No (Default: notebook)	Notebook/Lab/Terminal

Advanced options form (3)

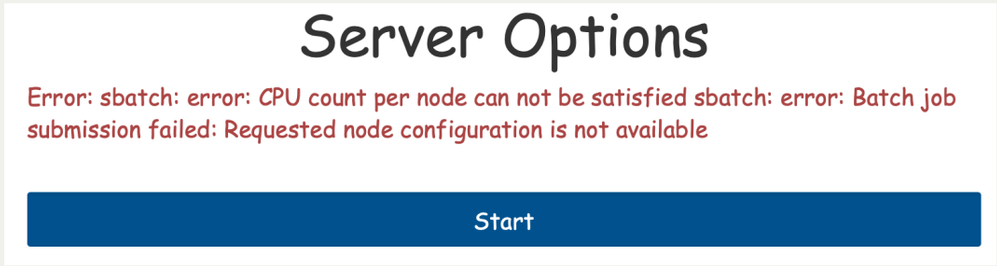


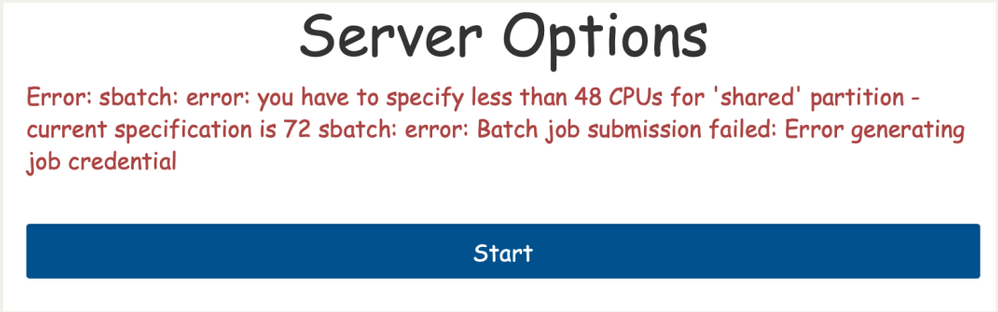


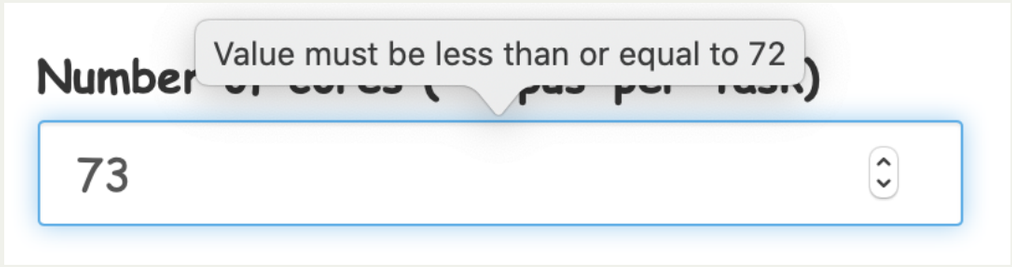


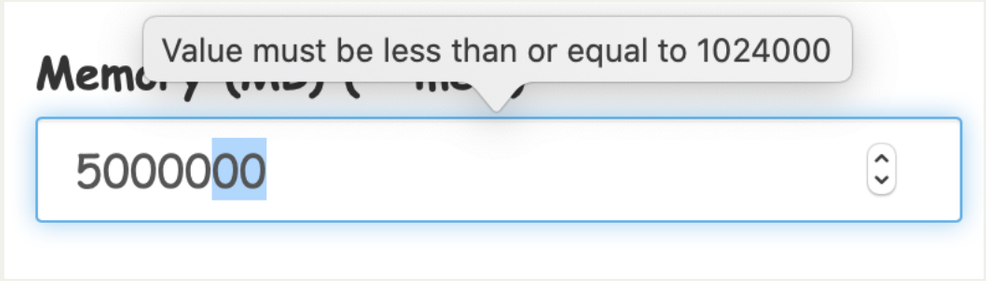


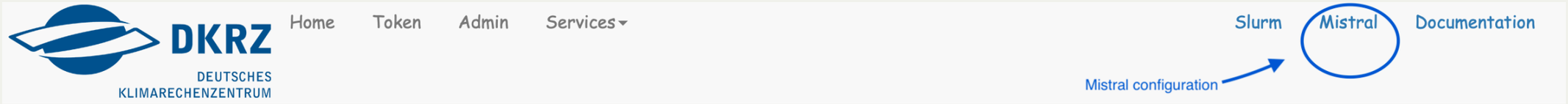
Advanced options form (3)



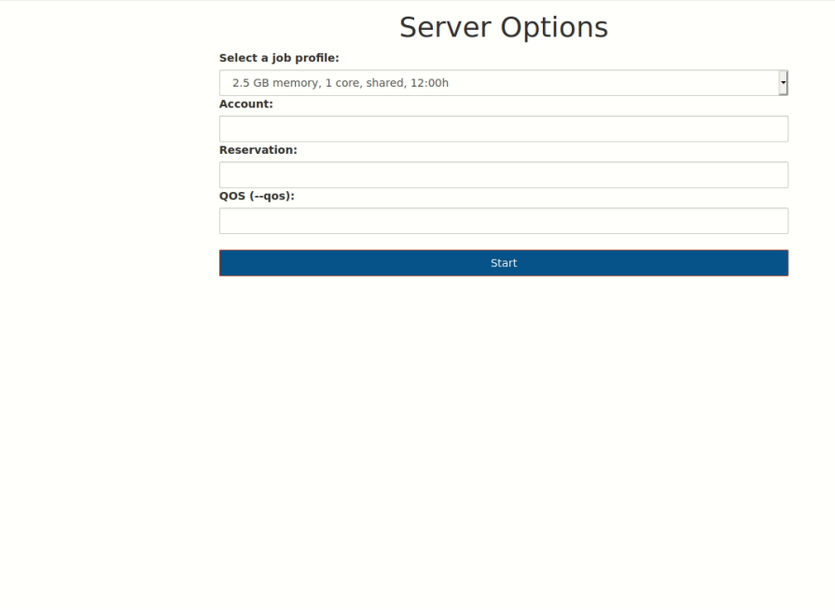






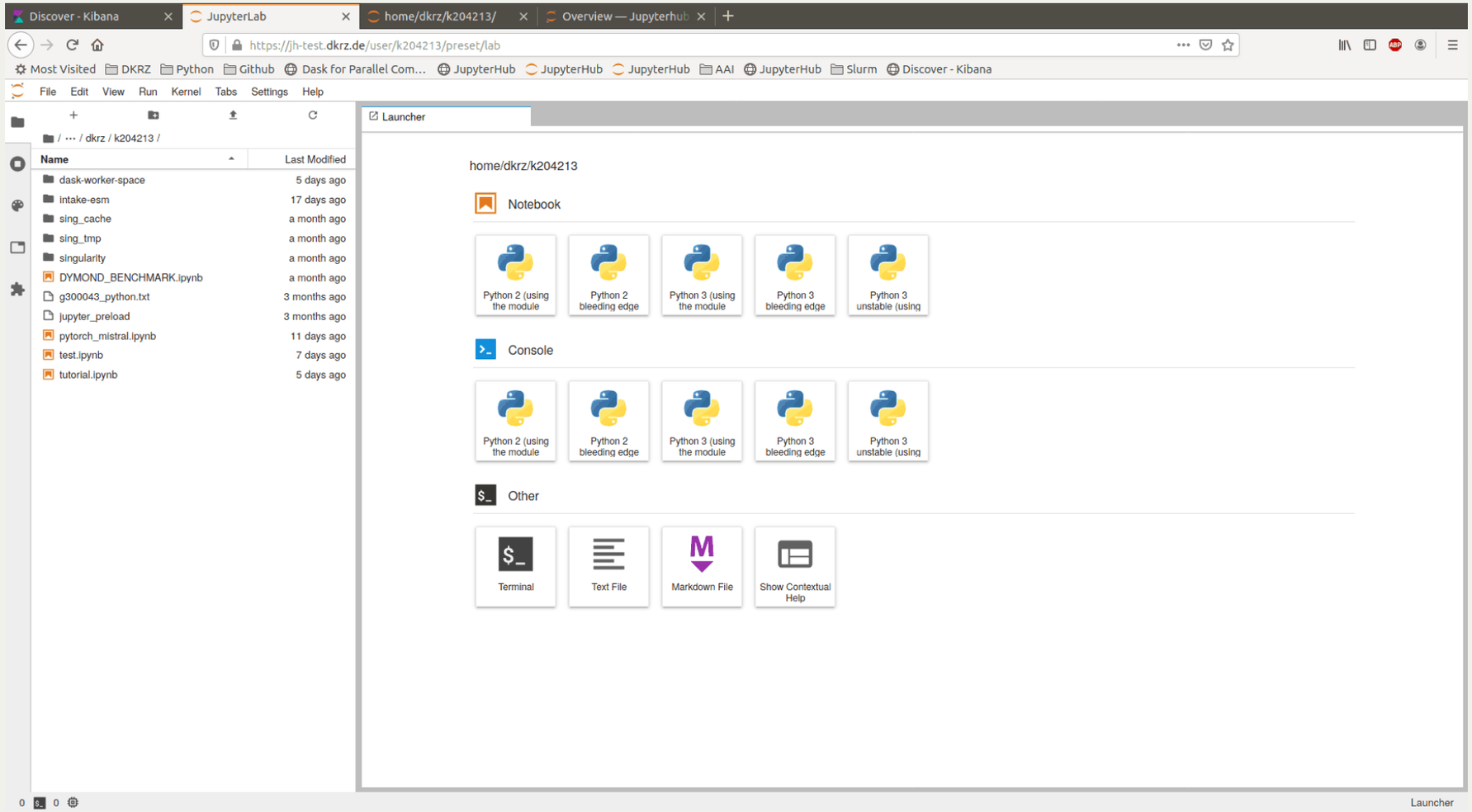


Named servers

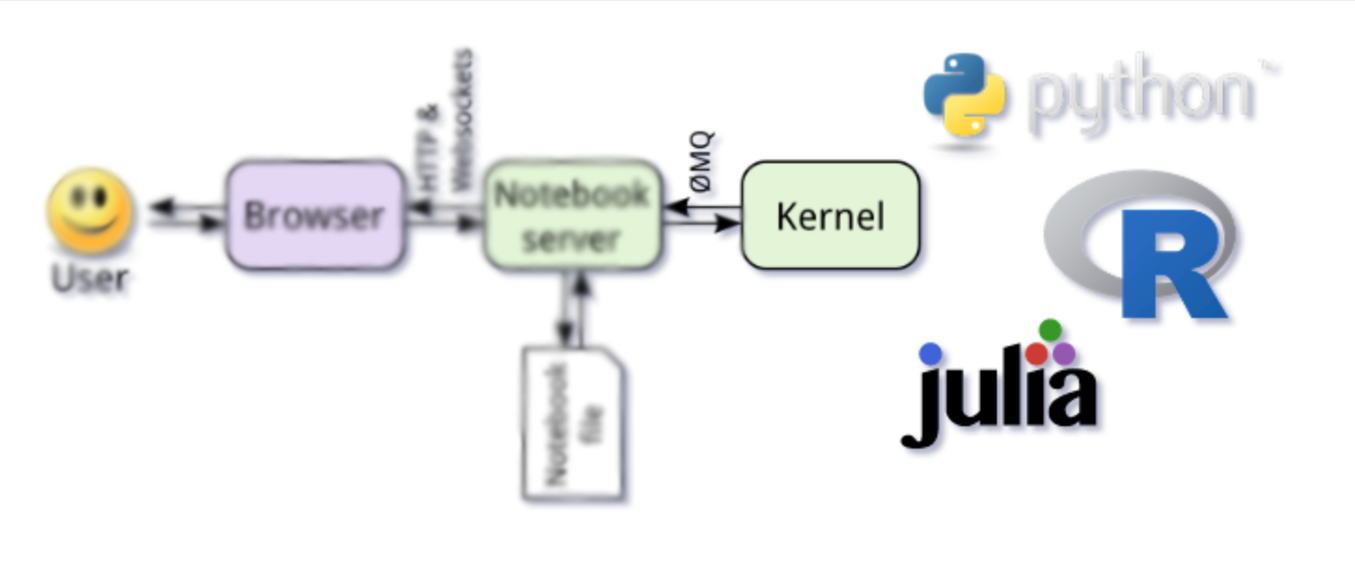


- *Named servers allow you to have more than one server running in the same time*
- *Currently 2 allowed:* ***preset*** *and* ***advanced***
- *Extendable --> HLRE 4 --> more*

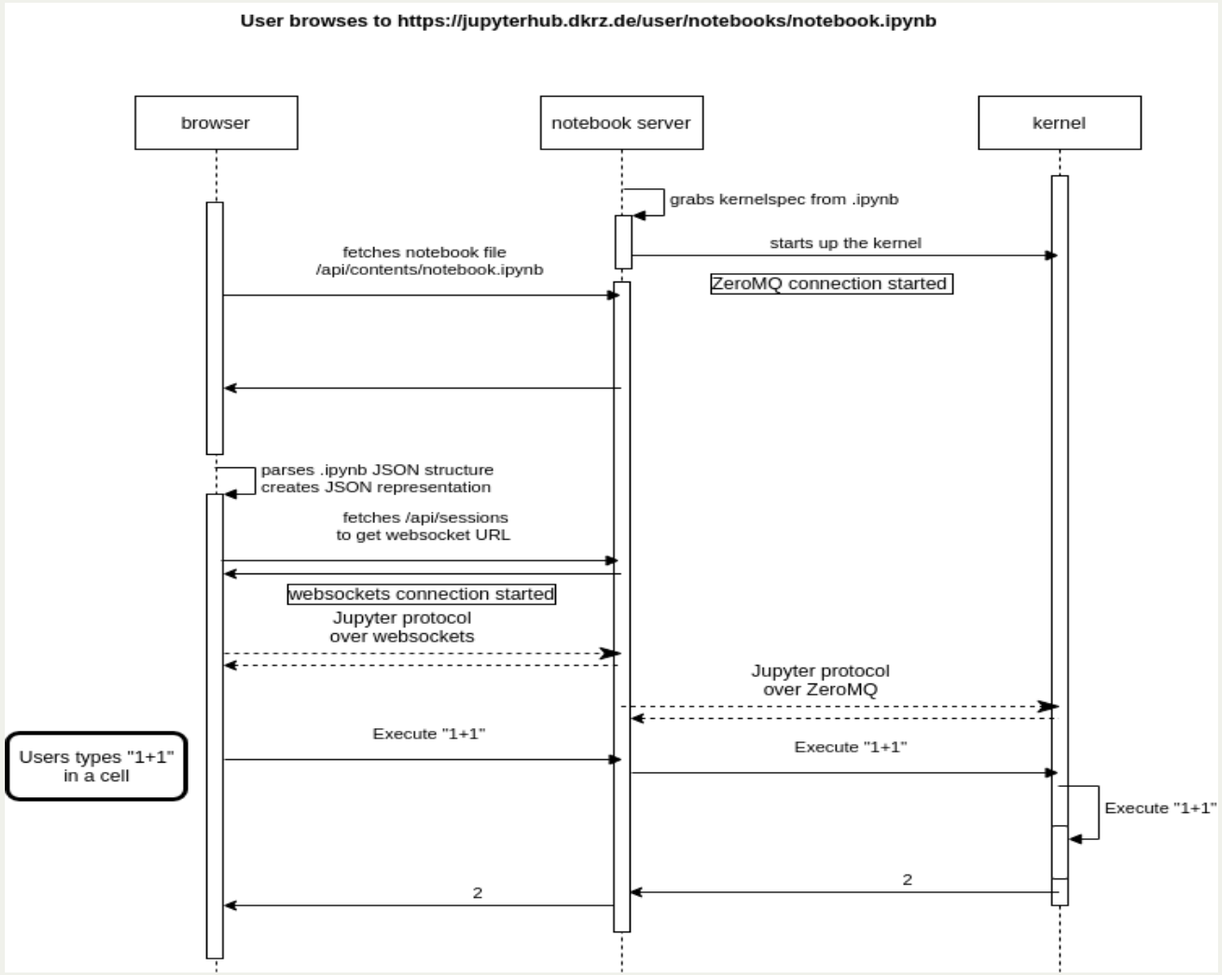
Stop your server



Kernels


User
Browser
HTTP &
WebSockets
Notebook
server
ØMQ
Kernel
Notebook
file
python
R
julia

What is happening in the background



Default system kernels

Kernel	Source module
Python 2	python/2.7.12
Python 2	anaconda2/bleeding_edge
Python 3	anaconda3/bleeding_edge
Python 3 unstable	*python3/unstable*

"We will have a new Python about every half year, which is going to be called 'python3/YYYY.MM-compilerversion'. The first one is 'python3/2020.02-gcc-9.1.0'".

Bring your own environment (1)

- Conda

```
% mkdir $HOME/kernels
% conda create --prefix $HOME/kernels/tensorflow ipykernel python=3.x
% source activate $HOME/kernels/tensorflow
% python -m ipykernel install --user --name tensorflow --display-name="ten
% conda deactivate
```

- Virtualenv

```
% python -m pip install --user virtualenv
% python -m virtualenv --system-site-packages /path/to/new-kernel
% source /path/to/new-kernel/bin/activate
% pip install ipykernel
% python -m ipykernel install --user --name new-kernel --display-name="new
```

Bring your own environment (2)

Default

```
{
  "argv": [
      "/path/to/kernel/bin/python",
      "-m",
      "ipykernel_launcher",
      "-f",
      "{connection_file}"
  ],
  "display_name": "new-kernel",
  "language": "python"
}
```

Customized

```
{
    "argv": [
        "/path/to/kernel/bin/python",
        "-m",
        "ipykernel_launcher",
        "-f",
        "{connection_file}"
    ],
    "display_name": "new-kernel",
    "language": "python",
    "env": {
        "variable": "value",
    }
   }
```

Bring your own environment (2)

Default

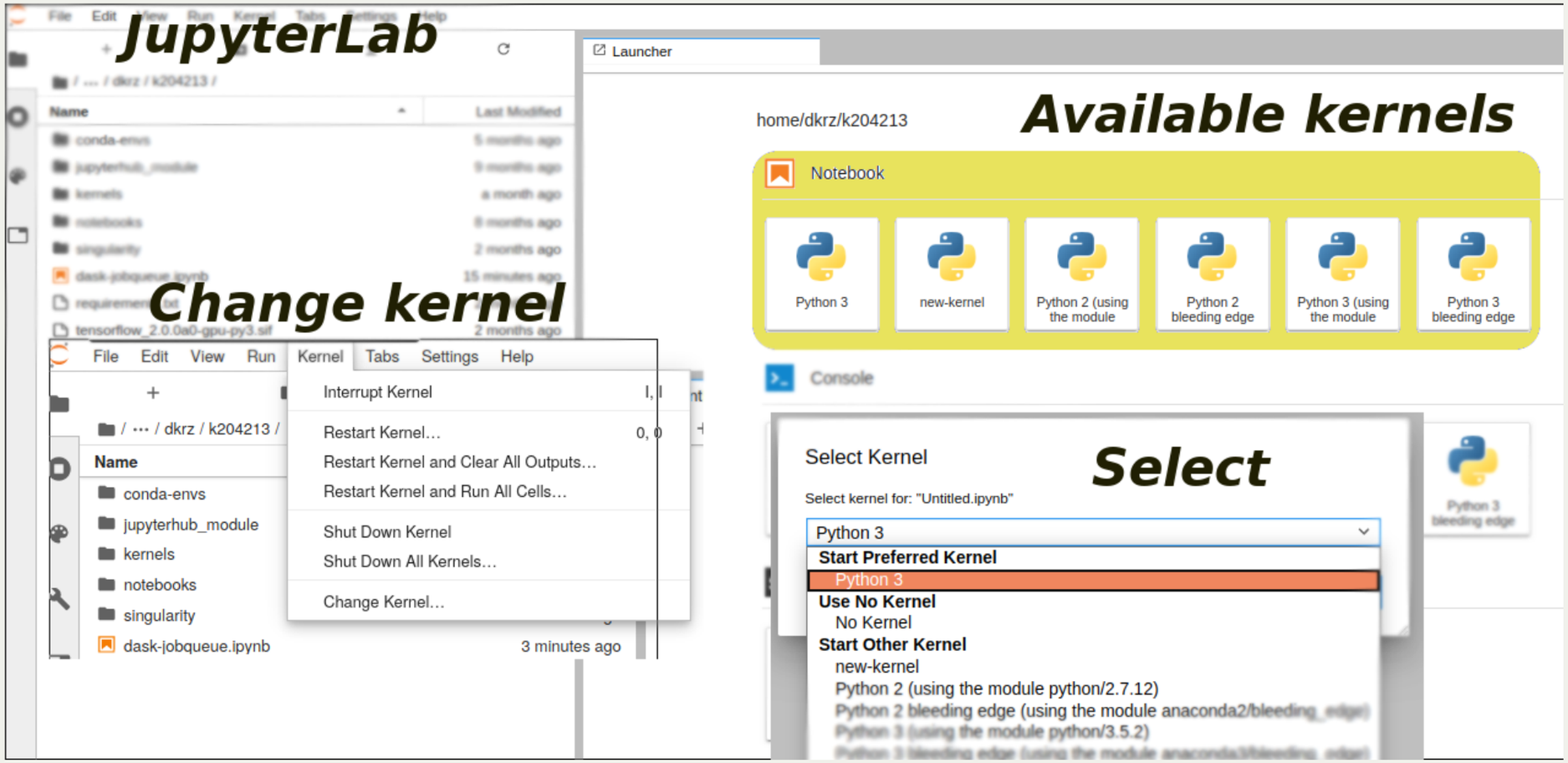
```
{
  "argv": [
      "/path/to/kernel/bin/python",
      "-m",
      "ipykernel_launcher",
      "-f",
      "{connection_file}"
  ],
  "display_name": "new-kernel",
  "language": "python"
}
```

Customized

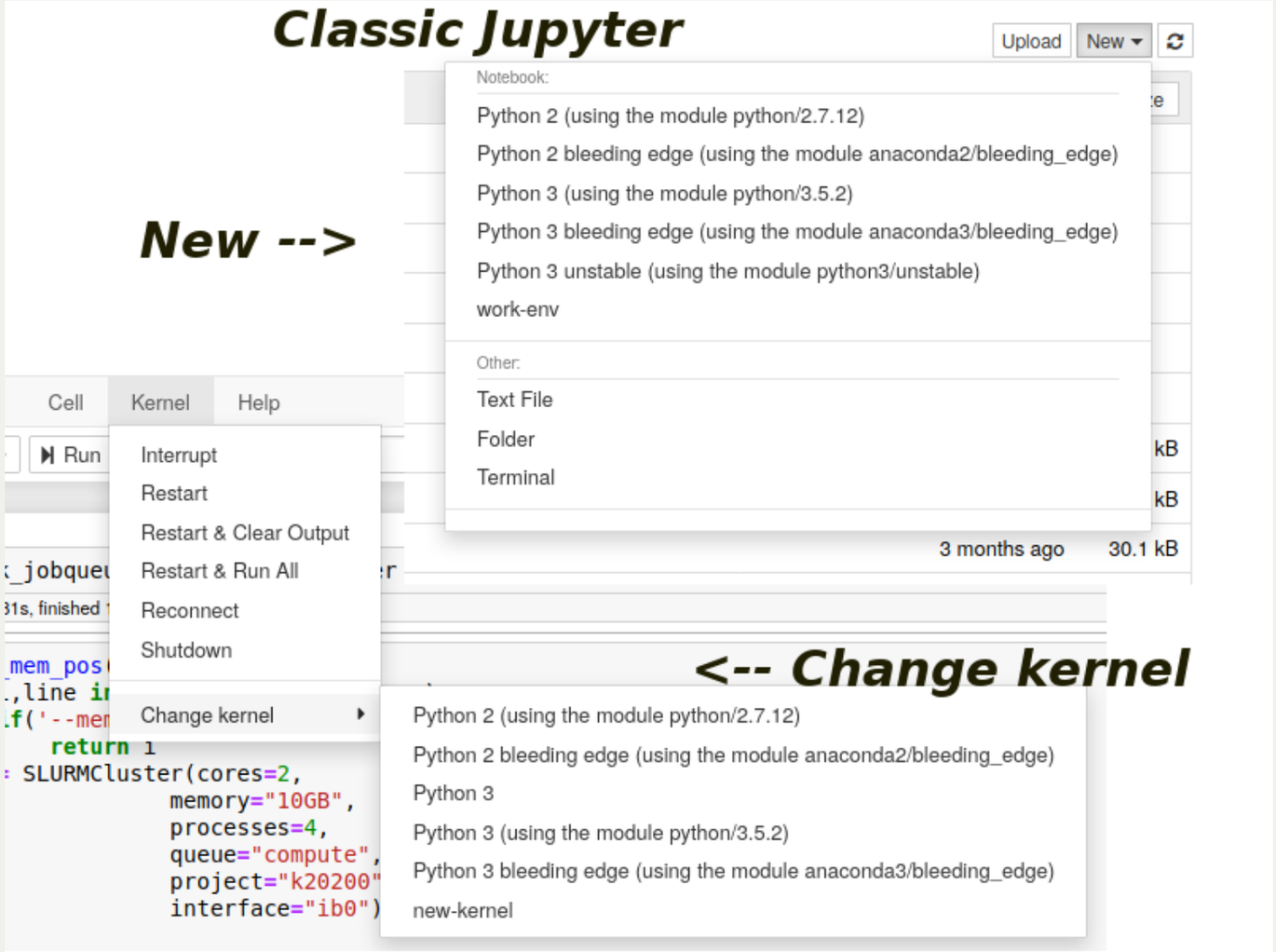
```
{
    "argv": [
        "/path/to/kernel/bin/python",
        "-m",
        "ipykernel_launcher",
        "-f",
        "{connection_file}"
    ],
    "display_name": "new-kernel",
    "language": "python",
    "env": {
        "variable": "value",
    }
   }
```

more advanced: use executable scripts! --> check the doc

Using/Changing the kernels



Using/Changing the kernels



Debugging

Debugging

- preset spawner
  - **jupyterhub_slurmspawner_preset_{slurm_job_id}.log**

Debugging

- preset spawner
  - **jupyterhub_slurmspawner_preset_{slurm_job_id}.log**
- advanced spawner
  - default: **jupyterhub_slurmspawner_advanced_{slurm_job_id}.log**
  - customized: **you_name_it.log**

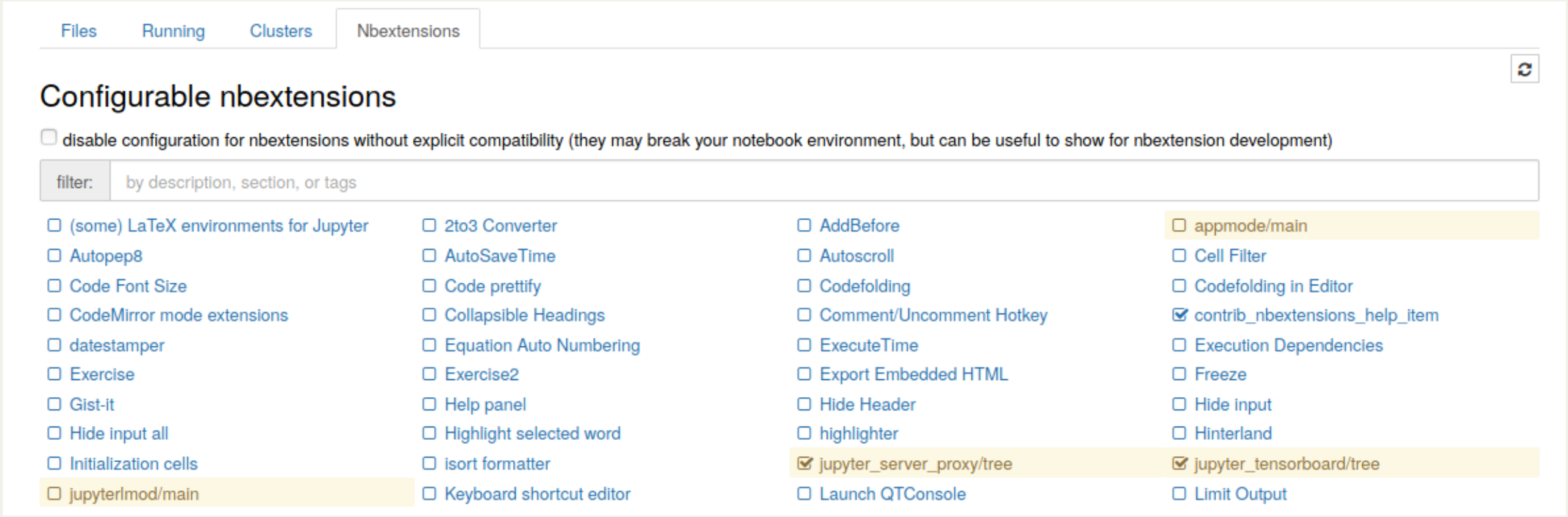
Debugging

- preset spawner
  - **jupyterhub_slurmspawner_preset_{slurm_job_id}.log**
- advanced spawner
  - default: **jupyterhub_slurmspawner_advanced_{slurm_job_id}.log**
  - customized: **you_name_it.log**

Extensions

Jupyter extensions (1)

Nbextensions



Jupyter extensions (2)

Enabling/Disabling extensions

Enable:

```
jupyter nbextension enable <nbextension require path>
```

Example:

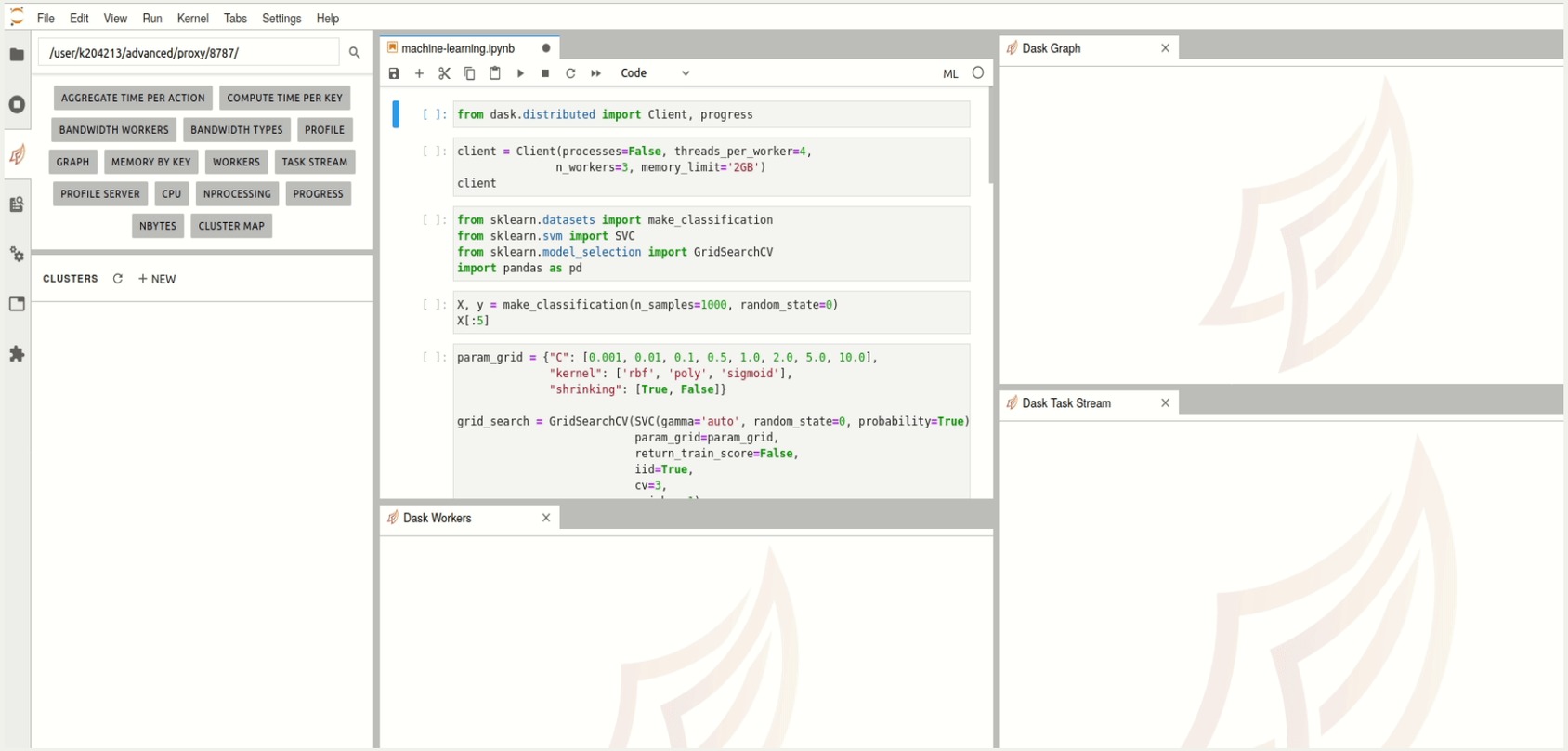
```
jupyter nbextension enable appmode/main
```

Disable:

```
jupyter nbextension disable <nbextension require path>
```

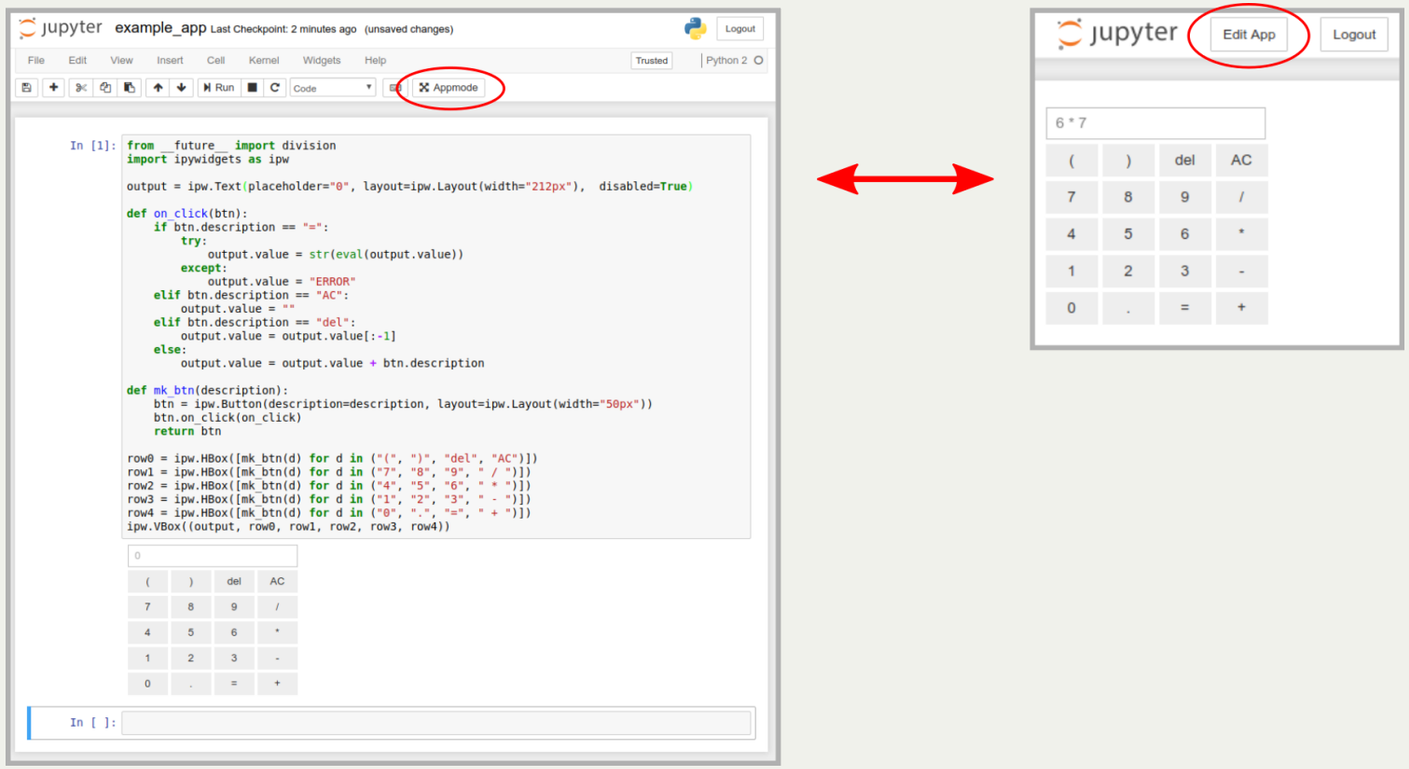
Jupyter extensions (3)

Dask Labextension



Jupyter extensions (4)

https://github.com/oschuett/appmode

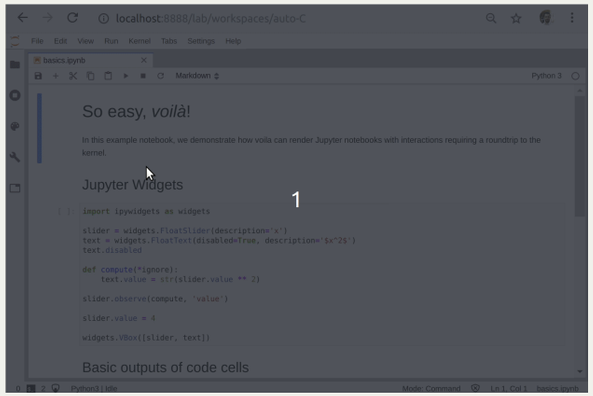


A Jupyter extensions that turns notebooks into web applications.

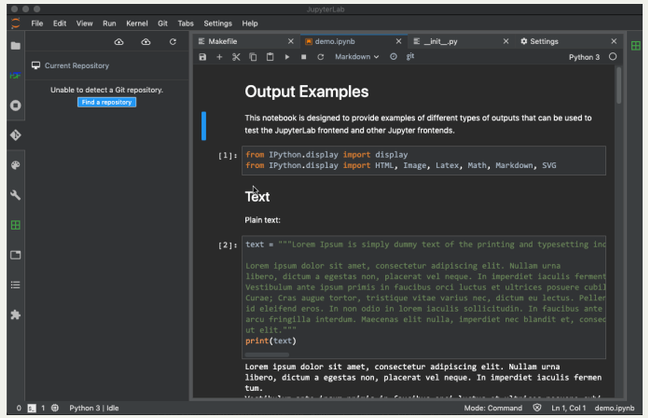
Jupyter extensions (5)

Do you need more?

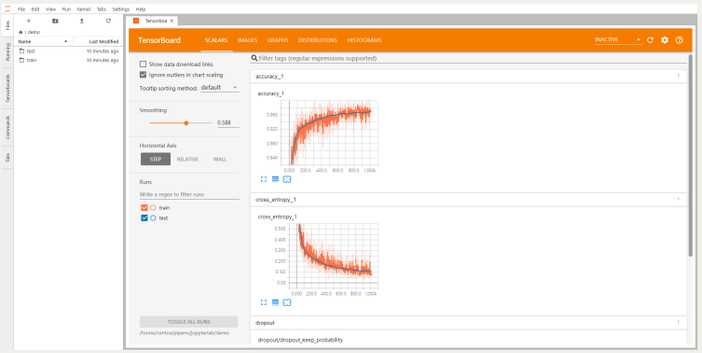
Voila



JupyterLab Git



TensorBoard



… support@dkrz.de

"The JupyterLab development team is excited to have a robust third-party extension community. However, we do not review third-party extensions, and some extensions may introduce security risks or contain malicious code that runs on your machine."

Future work

1. Sharing services/extensions
   - shared notebooks repositories (Git/hub/lab)
   - external sharing services
   - WPS
2. Enhanced spawning queue for Jupyterhub
3. More dedicated system kernels and extensions (e.g. ML)
4. Speed up loading Python packages (HLR4)
5. ContainerIze Jupyterhub
6. Binder for Mistral?

Feedback? Questions?

- Jupyterhub @ DKRZ
- Technical documentation
- support@dkrz.de