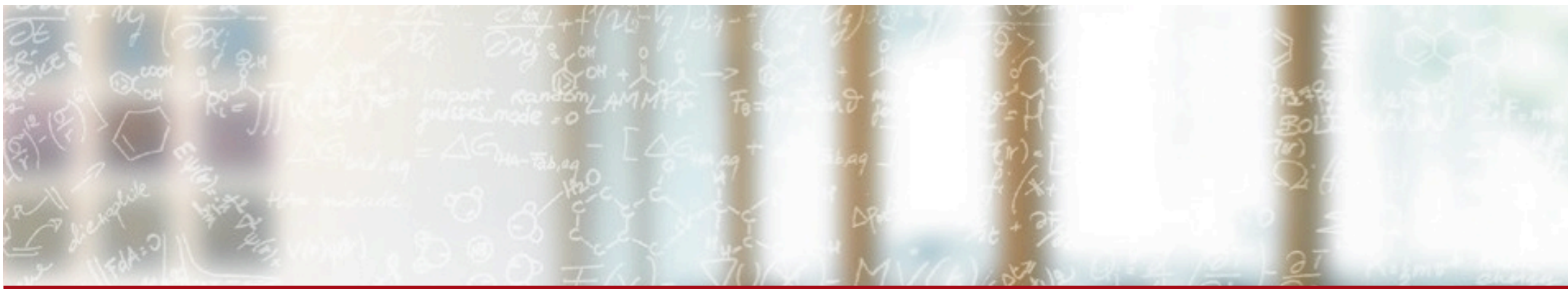




CSCS

Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



Interactive Computing with Jupyter Notebooks

CSCS User Lab Day, Luzern

Tim Robinson and Rafael Sarmiento, CSCS

September 11, 2018

Agenda

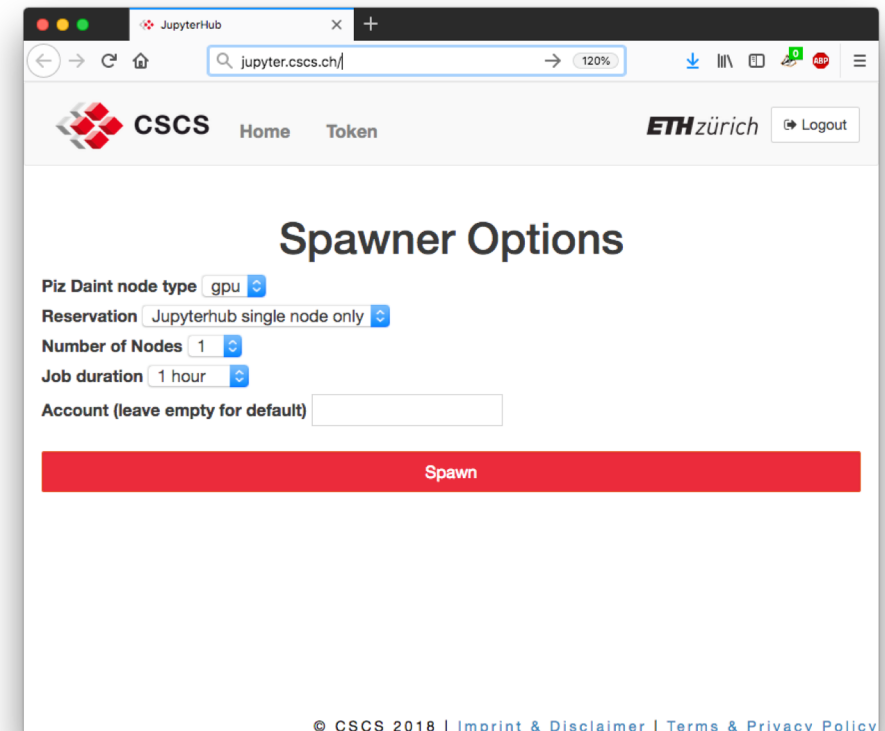
1. Introduction to Jupyter Notebooks (10 min)
2. Live Demonstrations (40 min)
3. Q and A (10 min)

Jupyter Notebooks

- Project Jupyter (<https://jupyter.org>) is an open-source project that develops software, open standards, and services for interactive and reproducible computing
- The name is derived from the core languages supported: Python, Julia, R
 - Kernels for many other languages are made available at:
<https://github.com/jupyter/jupyter/wiki/Jupyter-kernels>
- The main application developed by the project is the **Jupyter Notebook**
 - Jupyter Notebook is a web-app that allows users to create documents that combine **live code** with **narrative text**, **mathematical equations**, and **visualizations**
 - Notebooks are great for developing, sharing, communicating, and reproducing computational work in scientific computing and data science
- **JupyterHub** is a multi-user environment for managing multiple instances of the single-user Jupyter Notebook server

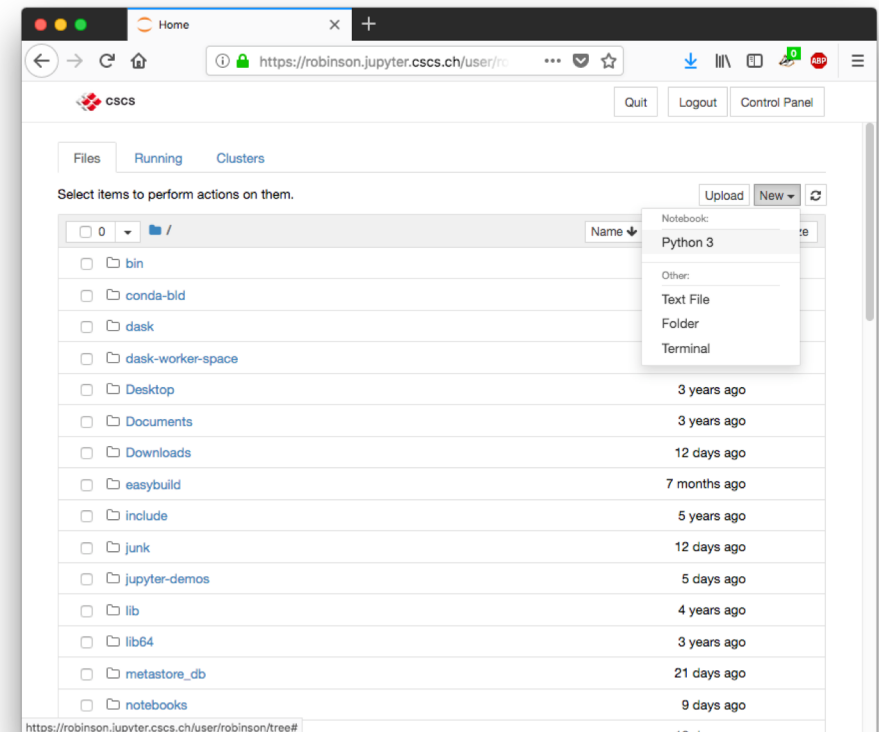
JupyterHub

- Users of Piz Daint can launch Notebooks at <https://jupyter.cscs.ch>
- Log in with your CSCS credentials
- There is a dedicated queue for single-node Notebooks
 - Maximum 16 concurrent users on each of the gpu and mc partitions
- For multi-node Notebooks, requests go to the “normal” queue
- Requests are cancelled if the allocation is not granted within 5 minutes
- A log file is created at `$SCRATCH/jupyterhub_slurmspawn_<jobid>`



Jupyter Notebook Dashboard

- From “New” you can create a new Jupyter Notebook, text file, folder or launch a terminal directly on a compute node
- To have your scratch folder appear in your dashboard execute the following command in a terminal on Daint (from \$HOME):
 - `ln -s $SCRATCH ./scratch`
- Remember to click the “Quit” button once you have finished your session – the “Logout” button will keep your server running and thus you will continue to consume your compute quota!



Custom Python Environments

- The “daint-gpu” or “daint-mc” modules are automatically loaded depending on which one was selected from the Spawner Options on the launch page
- An environment file “.jupyterhub.env” (located in your \$HOME folder) is sourced every time that a Jupyter server is launched
- Additional modules can thus be loaded by adding “module load” commands to the “.jupyterhub.env” file
 - `module load TensorFlow/1.7.0-CrayGNU-17.08-cuda-8.0-python3`
- Custom virtual environments can be activated by adding the corresponding command to the “.jupyterhub.env” file
 - `source my-venv/bin/activate`

Python 2

- Python 2 is not officially supported by CSCS
- If you would like to try it (at your own risk!), create a file “.local/share/jupyter/kernels/python2/kernel.json” with the following:

```
{
  "display_name": "Python 2",
  "language": "python",
  "argv": [
    "/apps/daint/UES/jenkins/6.0.UP04/gpu/easybuild/software/IPython/5.7.0-CrayGNU-17.08-python2/py27launcher",
    "-f",
    "{connection_file}"
  ]
}
```

Further Information

- CSCS documentation: <https://user.cscs.ch/tools/interactive/>
- Project Jupyter: <http://jupyter.org/>



CSCS

Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich

Live Demonstrations

If you would like access to the notebooks demonstrated live at the User Lab Day please contact the authors of this presentation. Email addresses can be found at <https://www.cscs.ch/about/staff/>