

Introduction

The "Scientific Computing" team at the LUIS starts supporting the [JupyterHub](#) platform for conducting classes in the LUIS compute cluster environment. The JupyterHub server also runs the system [NbGrader](#), which allows you to assign and (automatically) grade Jupyter notebooks. The JupyterHub platform is intended only for teaching activities that require the LUIS cluster resources. If you want to use Jupyter notebooks for your research, use the [Open OnDemand web portal](#) of the cluster instead.

JupyterHub is reachable under <https://jupyterhub.cluster.uni-hannover.de>. If you want to connect from the "outside" (e.g. from Home), you'll have to establish a VPN connection to the university's network first. See [VPN Service](#) for details.

Please make sure you have read and accepted the following terms of use of the JupyterHub service at the LUIS compute cluster.

Terms of use of the JupyterHub service

Follow [the link](#) for the German version.

Support

The JupyterHub service, deployed as part of the LUIS "Scientific Computing" service, directly accesses the resources of the LUH computing cluster. Therefore, [the rules for using the LUIS IT systems and services](#) as well as the description in [the cluster wiki](#) apply.

Availability

The JupyterHub service is available under the same general conditions as the other resources of the LUH computing cluster. However, the operation of the computing cluster takes priority over the operation of JupyterHub. **Neither is explicitly critical infrastructure.**

Due to scheduled cluster maintenance and some emergency conditions (e.g. kernel security bug of the cluster operating system), the JupyterHub service may experience downtime. **Continuous availability of the service cannot be guaranteed.** We recommend using JupyterHub exclusively for non-time-critical courses such as exercises, for example as a uniform working environment for students.

Usage

The JupyterHub service at the cluster, which is **only offered as a teaching or learning platform**, is not suitable for taking exams for the reasons mentioned above. Regardless of this, special legal regulations apply to examinations, for compliance with which the cluster team cannot assume any responsibility.

The JupyterHub service may only be used for courses that require resources of the LUH computing cluster.

After the end of a course, accounts in BIAS that are no longer required must be deleted.

Resources

Access to the computing resources is provided exclusively through the scheduling of the computing cluster. For each Jupyter Notebook session, one compute job must be submitted from JupyterHub. The time the job spends in the queue depends mainly on the job requirements and the current workload on the compute cluster.

If courses are to be set up on JupyterHub and the necessary cluster resources are to be made available, we require **early registration** and usually direct coordination with instructors, as the required computing resources must be prepared for reservation in order to shorten job waiting times. The compute resources are part of the "Scientific Computing" service and are therefore limited.

We welcome the use of the institute's own nodes in the [FCH](#) (research cluster housing) for JupyterHub sessions.

How to register and manage a course

To get started with the JupyterHub platform you would need to create cluster usernames using BIAS. Note that BIAS has recently made it possible to create bulk cluster accounts. The usernames are required to log into JupyterHub both for instructors of a course as well as for students this course is assigned to. Before the usernames are granted authentication on the platform, you would need to initially register your course on JupyterHub. To do this, please contact the cluster team at cluster-help@luis.uni-hannover.de and provide the following information about your course:

- Course name (the name should not contain a colon)
- Cluster username (and optionally first name, last name and email address) of at least one instructor of the course.

A course can have multiple instructors each with their own cluster username. Each registered course receives a unique identifier in the form of the string `course<4 digit number>`.

Once the course is registered, the course instructor can log into the JupyterHub platform at <https://jupyterhub.cluster.uni-hannover.de> (see [figure 1](#)) and

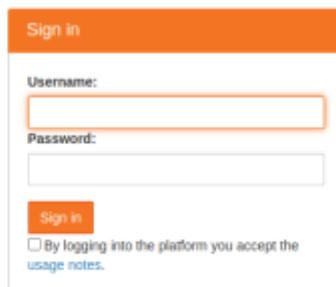
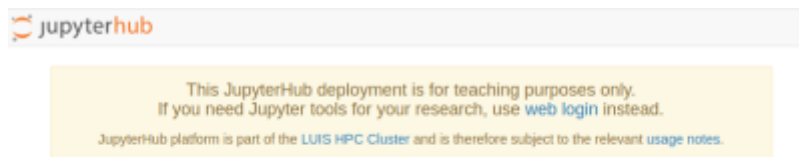
The image shows a "Sign in" form. It has an orange header with the text "Sign in". Below the header, there are two input fields: "Username:" and "Password:". Below the password field, there is an orange "Sign in" button. At the bottom, there is a checkbox with the text "By logging into the platform you accept the [usage notes](#)."

Fig. 1: JupyterHub login page

access the registered courses from the menu **Available Courses**, see [figure 3](#). To get to the menu page, click the link **To Available Courses** after logging in, [figure 2](#).



Fig. 2: JupyterHub home page



Fig. 3: JupyterHub courses menu page

By selecting a course from the menu, the page displays the course settings, such as course SLURM job options, a list of instructors and students assigned to the course, the location of course data in the cluster, etc., [figure 4](#). The information shown varies depending on whether the authenticated username is registered as an instructor or student to the selected course.



Fig. 4: Settings of the selected course

To start Jupyter Notebook for the selected course, click the link **Start**, see [figure 4](#). This submits a SLURM batch job to the cluster. The page opened will show the state of the job in the queue including the ID of the job, [figure 5](#). The time a course job waits in the queue depends on the job requirements and the current workload of the cluster.

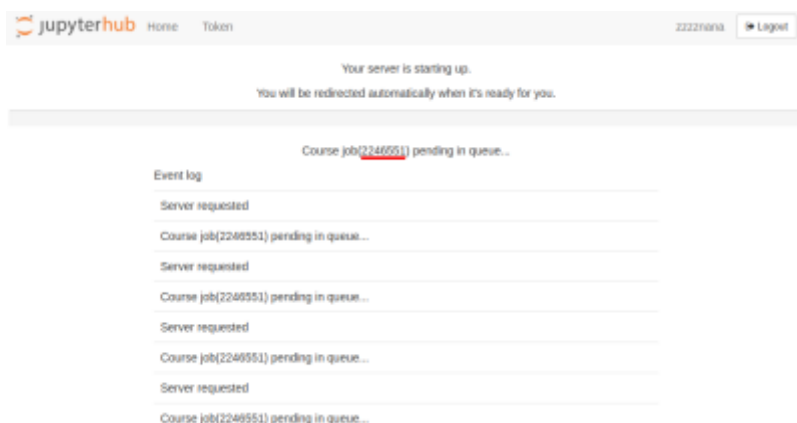


Fig. 5: Course job in queue

Note: if a course job fails, please include the job ID in your support request.

Note: If you want to reduce the wait time for a course job, the appropriate cluster computing resources must be reserved for the course. To do so, please provide us with the required amount of resources (number of CPUs, memory, walltime and other options per course job, the number of course participants) as well as the start and end dates of the course at least a few weeks before the course starts.

To manage the registered courses use the links > **Export course config** and > **Import course config** on the course menu page, [figure 4](#). The link > **Export course config** exports the current configuration of the selected course as a JSON file. The file name is the course ID with a JSON extension. Open the JSON file and make appropriate changes. To activate the changes, use the link > **Import course config** to upload the JSON file to the JupyterHub server. Note that the JSON file name must match the ID of the course whose configuration you want to change. The following course options can be modified:

- List of instructors
- List of students
- Course SLURM job options (long form of the options must be entered)
- The name and location of the Conda environment used for the course
- Course start and end dates
- Course comment

The following is an example JSON configuration for the course with the ID course010 (save the lines to the file course010.json):

course010.json

```
{
  "state": "active",
  "name": "Python for scientific computing",
  "comment": "Python for scientific computing",
  "teachers": [
    "username1,John,Chase,j.chase@uni-hannover.de",
    "username2,Juliet,Williams,williams@uni-hannover.de",
    "username3,Callan,Barry,c.barry@uni-hannover.de"
  ],
  "students": [
    "stud0001,Armaan,Nunez,nunez@stud.uni-hannover.de",
```

```

    "stud0002,Angelica,Vargas,angelica.vergas@stud.uni-hannover.de",
    "stud0003,Jago,Clements,clements@stud.uni-hannover.de",
    "stud0004,Suzanne,Barker,barker@stud.uni-hannover.de",
    "stud0005,Faris,Hodges,faris.hodges@stud.uni-hannover.de",
    "stud0006,Rajan,Jensen,jensen@stud.uni-hannover.de",
    "stud0007,Kezia,Copeland,k.copeland@stud.uni-hannover.de",
    "stud0008,Kyron,Petersen,petersem@stud.uni-hannover.de",
    "stud0009,Ann,Edwards,ann.edwards@stud.uni-hannover.de",
    "stud0010,Benjamin,Fletcher,fletcher@stud.uni-hannover.de"
  ],
  "location": "/jupyterhub/courses/course010/course_dir",
  "start": "03.09.2023",
  "end": "08.12.2023",
  "job_options": " --time=04:00:00 --nodes=1 --cpus-per-task=4 --mem-per-cpu=4G --gres=gpu:1",
  "jupyter_notebook": {
    "autosave_interval": 10,
    "kernel_default": {
      "name": "Python Computing 2023",
      "conda_env":
"/jupyterhub/courses/course010/conda_envs/python_computing_2023"
    }
  }
}

```

Each student and instructor entry is a comma-separated string with the following content: "`<username>,<first name>,<last name>,<email address>`". Where all except `<username>` are optional.

The course data directory ("`location`": in JSON file) is located at `/jupyterhub/courses/<course id>/course_dir` in the cluster. Where `<course id>` should be replaced with the course ID. The data directory is where the NbGrader database and config files, Jupyter notebooks and other working directories are placed. The path is fixed and cannot be modified.

The path to the course Conda environment ("`conda_env`":) should be prefixed with `/jupyterhub/courses/<course id>/conda_envs`. See the example configuration file above.

Instructors can also access their course locations from their home directory in the cluster as `~/MY_COURSE/<course name>`.

The following SLURM options are set for a course job by default: "`job_options`": "`--time=04:00:00 --nodes=1 --cpus-per-task=1 --mem-per-cpu=2G`"

Note: if you want to remove all student usernames from your course config, set "`students`": "`"`".

Note: The maximum number of student usernames per course is currently limited to 1,500.

To deactivate a course, set "`state`": "`inactive`" in the course JSON file. Inactive courses do not appear in the course menu, but their data are not deleted and remain at the same location.

Once the course job starts, the Jupyter notebook server is launched on the allocated compute node.

Instructors can manage (generate, release, collect, grade, etc.) assignments via the **Formgrader** menu in a Jupyter notebook session, see [figure 6](#).

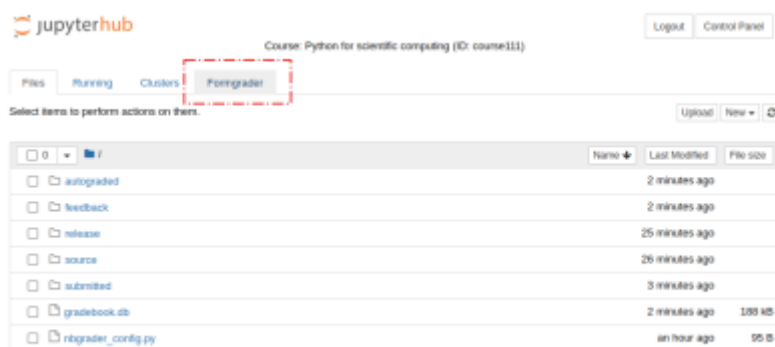


Fig. 6: NbGrader for instructors

Using the **Assignments** menu, [figure 7](#), in a Jupyter notebook session, students can retrieve and submit course assignments,



Fig. 7: NbGrader for students

Currently running course jobs are shown at the bottom of the main page, see the section **Running Course Jobs**, [figure 2](#). You can connect to the running Jupyter session using the link **connect**. The link **stop** terminates a session (i.e. cancels the corresponding SLURM job).

Details on how to create Jupyter notebooks for course assignments in NbGrader format and how to use NbGrader interfaces can be found [here](#). Note that NbGrader supports managing course assignments both interactively via a Jupyter notebook session as well as using a command line tool.

From:
<https://docs.cluster.uni-hannover.de/> - **Cluster Docs**

Permanent link:
<https://docs.cluster.uni-hannover.de/doku.php/resources/gizojhub>

Last update: **2023/10/11 07:09**

