

# Connect to RStudio using jupyterhub

Sara Correa Garcia

## Protocol to connect to RStudio using jupyterhub

To connect to RStudio from Jupyter Lab using the Compute Canada clusters, you can follow the steps below:

1. Open your browser and navigate to: [jupyterhub.sharcnet.ca](https://jupyterhub.sharcnet.ca). After filling the login page with your Compute Canada credentials, you will be redirected to a page to fill your server request.

2. Fill in the fields according to your needs: Account, Time, Number of cores, Memory and GPU configuration.
  - **Account:** At mECO:labs, we usually have resources allocated in the Graham cluster (that is why we choose sharcnet to start the session). But you may have resources allocated in other clusters. To find out how many resources have been allocated to the group at a given time, you can visit [ccdb.alliancecan.ca](https://ccdb.alliancecan.ca). Under the menu **My**

*Account*, you can explore the section *My Resources and Allocations*. At the moment of writing this, you will probably pick: rrg-profname\_cpu.

- **Time:** The minimum time is 15 minutes (0.25 hours). This should be enough for the purpose of testing your first connection. The more time you ask for, the higher your chances to see your request denied. I have seen difficulties to get a time slot of 4 hours.
- **Number of cores:** To test 1 is enough. Asking for more cores in combination with GPUs is specially useful if your code is optimized for parallel computing.
- **Memory:** The memory you need. Usually more than what the limit of your computer is. However, if you require too much resources your request may not spawn at that given time. If you really need to visualize your code in the cluster as you go with a lot of memory, you can try to ask for large resources at a later time (such as in the evening). For more than 32 Gb of memory, it may be difficult to get the allocation you ask for and you should aim to submit the task as a batch job.
- **GPU configuration:** To determine which type of GPU you need you need to understand the types available. You can read more about Graham's GPUs [here](#).

3. Click start. Depending on the resources that you asked for, the session may take some time to start. You land in this page, and may stay there for some time (A).

If the time runs out (B), it probably means that the resources you asked for were not available at the moment. You can try to change your number of cores and memory, or relaunch the petition for resources later.

4. When your sever launches, you will get in this page:

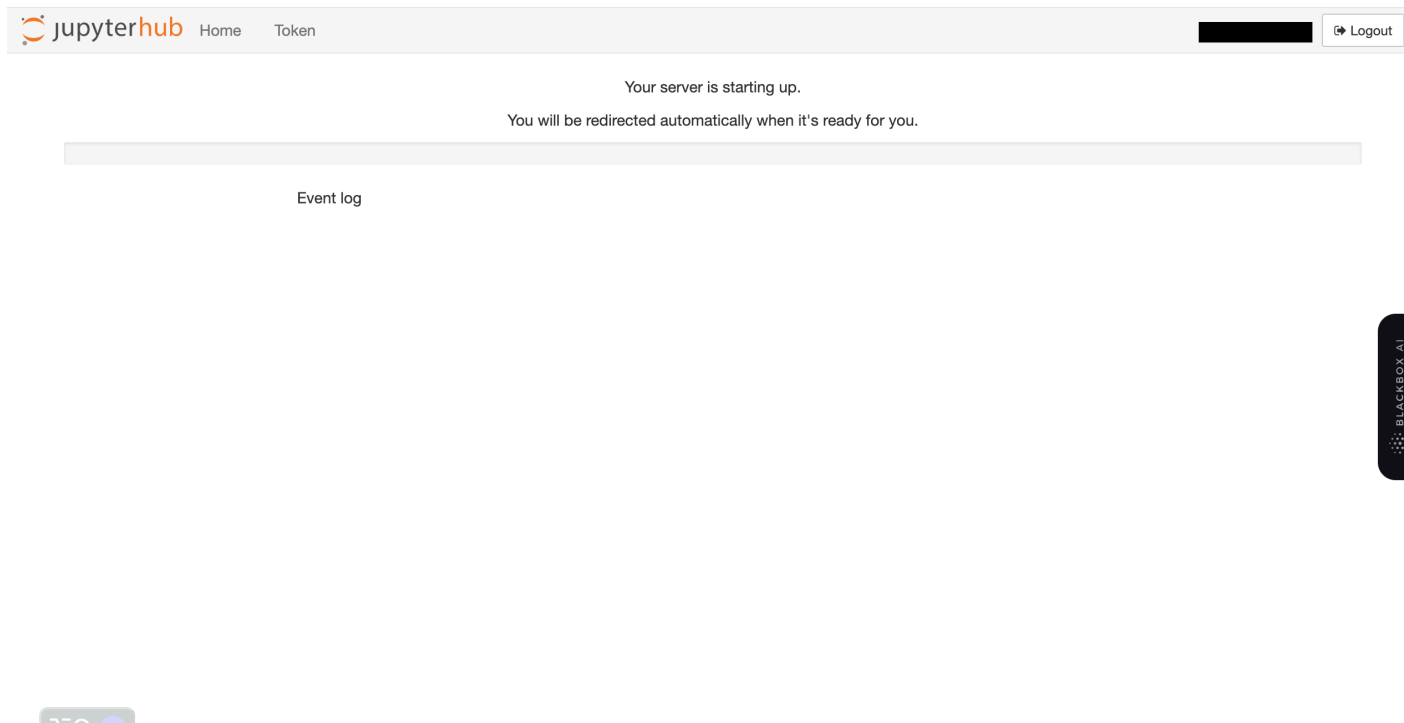


Figure 1: A. Server allocation request is pending in queue

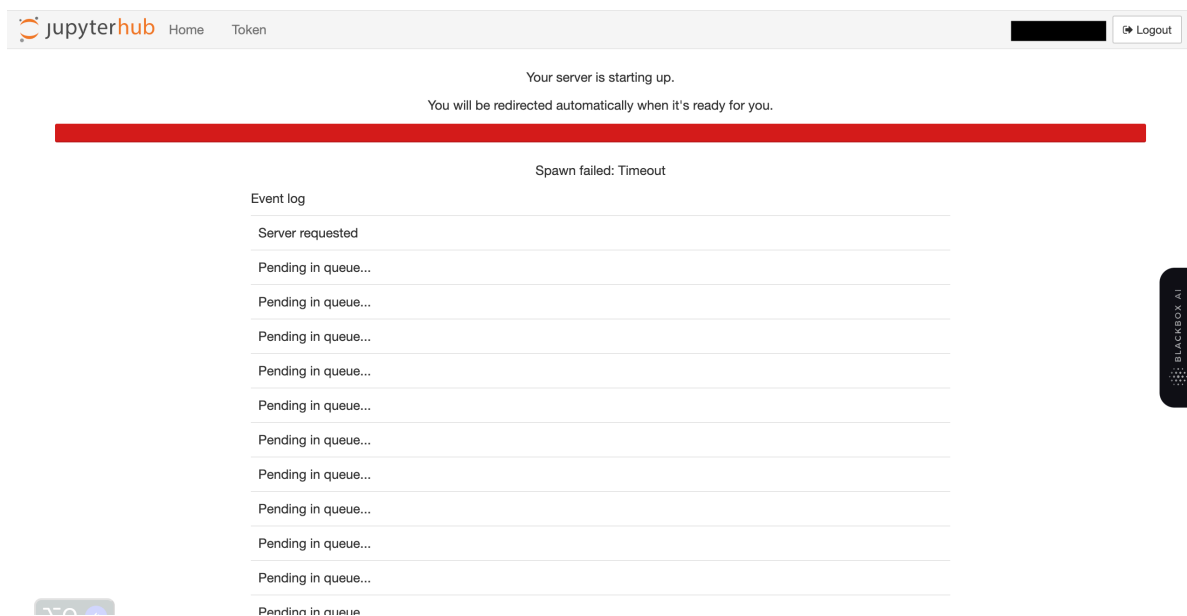
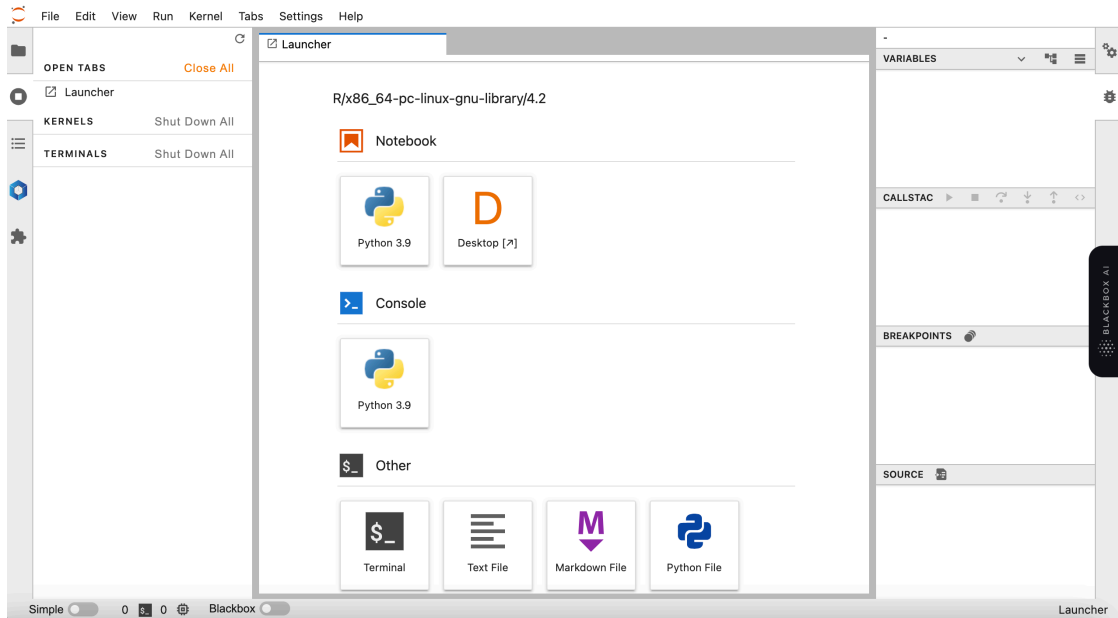
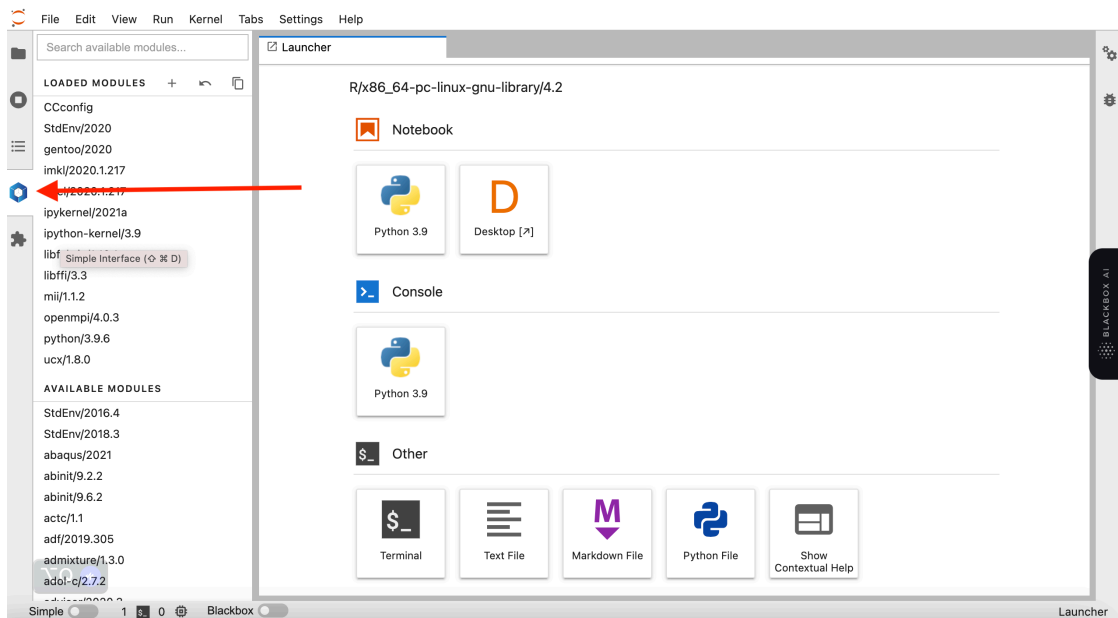


Figure 2: B. Server allocation failed: try at a later time, or reduce your request for resources

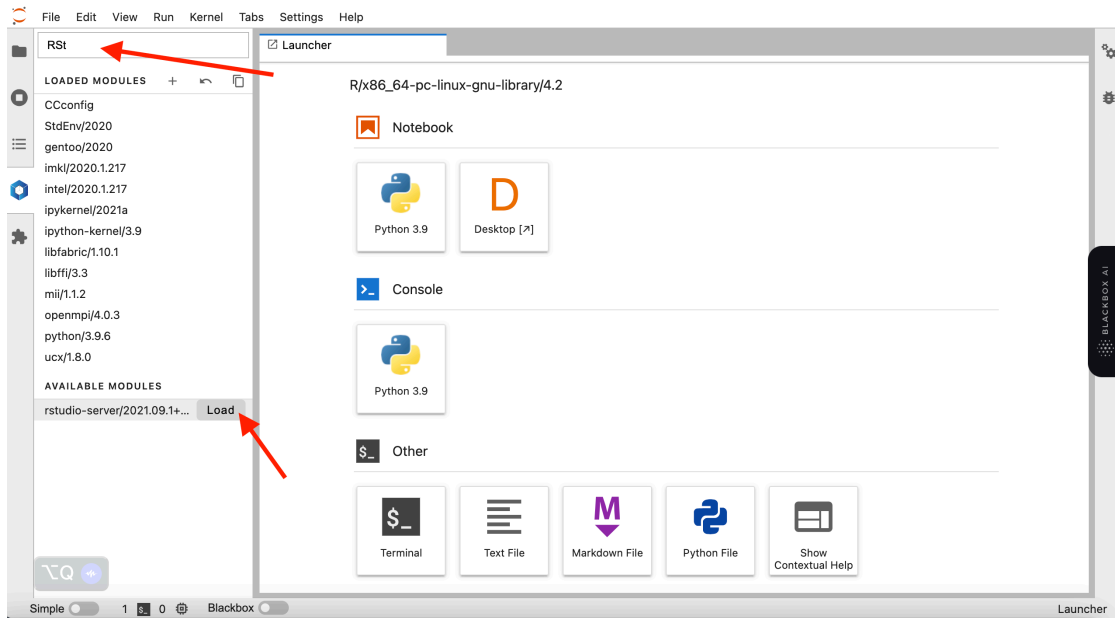


Here, you can open a python notebook automatically, or a terminal, as needed. We need to load a module to start RStudio.

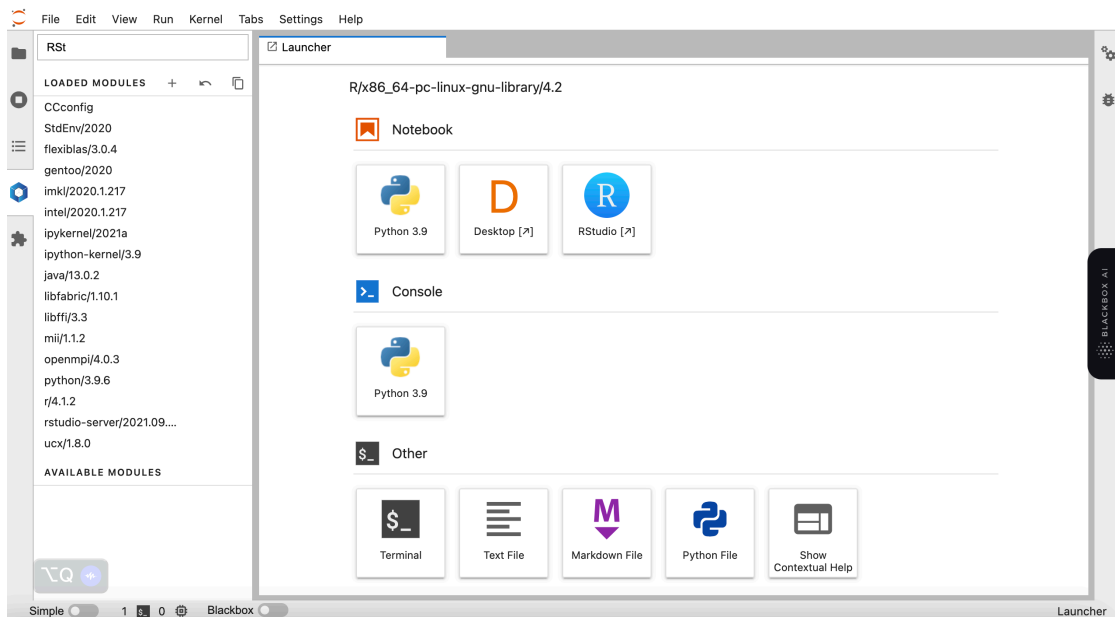
5. To start RStudio, you will need to navigate to the modules tab, on the left menu:



6. In the search box, you can start typing RStudio, until the `rstudio-server` module becomes available at the bottom section. Click on the load button.

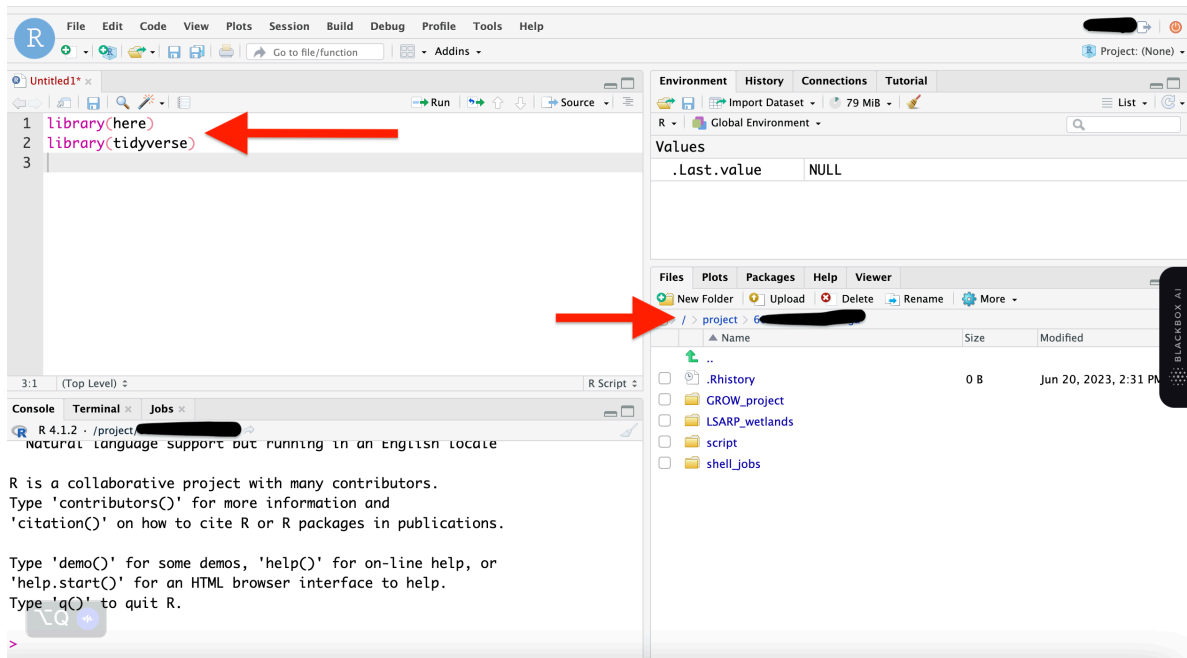


- After a couple of seconds, a new RStudio icon will appear in the Launcher page. Click on it to start RStudio in a new navigation tab.



- Notice that your current working directory is now located at your personal Compute Canada project's directory. All the actions you perform here: deleting, renaming or creating files or folders, will be reflected in your project directory: be mindful of your clicking to avoid altering or removing important files. You can now create `.Rproj` as usual, following your typical project architecture. Also, this RStudio session won't have

access to the internet, so all the packages must be preinstalled in an interactive node session (check out the [Compute Canada tutorial](#) to guide you through it). You can then load them as usual with the `library()` function. If you need to submit your data and your script to the cluster, you can do so following the instructions provided in the previously mentioned tutorial.



And that's it! Now you are ready to code in RStudio using the Compute Canada HPC resources.

## References and resources

- <https://jupyterhub.sharcnet.ca/hub/spawn>
- <https://ccdb.alliancecan.ca/>
- [https://docs.alliancecan.ca/wiki/JupyterHub#JupyterHub\\_on\\_clusters](https://docs.alliancecan.ca/wiki/JupyterHub#JupyterHub_on_clusters)
- [https://github.com/le-labo-yergeau/Coding-club-Compute\\_Canada](https://github.com/le-labo-yergeau/Coding-club-Compute_Canada)
- <https://docs.alliancecan.ca/wiki/Graham>