

Hackathon - BIOMERO 101 - prep

How to set up a local BIOMERO instance on your **Windows** laptop (or **Windows** remote computer) ?

It consists basically of the following steps:

1. Prerequisite: Install Docker Desktop & Git
2. Prerequisite: Set up SSH
3. Set up (local) Slurm
4. Set up NL-BIOMERO

Note: We will explain setup with **Docker Desktop for Windows**, as it's the most user-friendly. But all these containers should also work under any Docker or Podman installation on any platform, or at least Windows and Linux.

Tip of the day: *You can get auto-complete in the terminal when typing folder or filenames by hitting **tab** a few times after typing a few letters. Very useful.*

1. Prerequisite: Install Docker Desktop & Git

1. Install Docker Desktop: <https://docs.docker.com/desktop/>
 - a. Windows: <https://docs.docker.com/desktop/setup/install/windows-install/>
 - i. Use the Windows Subsystem for Linux **WSL** backend preferably (check <https://learn.microsoft.com/en-us/windows/wsl/install#install-wsl-command> to setup WSL, then make sure to enable this in Docker Desktop-Settings-General-'Use the WSL2 based engine')
 - b. Start Docker Desktop (you should have an App installed now)
2. Install Git (for Win): <https://gitforwindows.org/> (or <https://git-scm.com/downloads/win>)
3. Create a new folder to house your workspace (e.g. *D:/workspace*), for all the code etc.
4. Let's get ready to download our required code and containers:
 - a. Open a *PowerShell* terminal.
 - b. I will provide commands below to copy and paste into the terminal in **blue**.
 - c. To start, navigate to your workspace:
cd D:/workspace
 - d. Clone the git repository for Slurm containers
git clone <https://github.com/TorecLuik/slurm-docker-cluster>
 - e. Clone the git repository for NL-BIOMERO containers
git clone <https://github.com/Cellular-Imaging-Amsterdam-UMC/NL-BIOMERO.git>

2. Prerequisite: Set up SSH

5. We have to set up SSH access between Slurm and BIOMERO, this will mimic the real thing. For that, you need a SSH key. Do we have it already?

```
ls ~/.ssh/
```

If this shows some files including `id_rsa` and `id_rsa.pub`, you can already have nice SSH keys. If it shows different encoding (not rsa), perhaps just create new rsa keys anyway.

6. If not, you need to generate some:

```
ssh-keygen -t rsa
```

Press enter a few times to accept the defaults.

```
ls ~/.ssh/
```

We should see `id_rsa` and the `id_rsa.pub` files now.

7. We also have to provide an `alias` inside our SSH config file, so BIOMERO can use it (for now the alias should be `localslurm`, to match the `slurm-config.ini` we will use).

- a. Do you have a config file yet?

```
ls ~/.ssh/config
```

That shows an error (don't have it), or a file named `config`

- b. If you don't have a SSH config file yet, just copy our example file from NL-BIOMERO directory:

```
cp NL-BIOMERO/ssh.config.example ~/.ssh/config
```

- c. If you already have a ssh config file (`~/.ssh/config`), you need to edit this file and add this new alias to it as shown below:

```
Host localslurm
    HostName host.docker.internal
    User slurm
    Port 2222
    IdentityFile ~/.ssh/id_rsa
    StrictHostKeyChecking no
```

3. Set up (local) Slurm

8. We will install a Slurm setup onto our own computer:

- a. You should have a subdirectory called `slurm-docker-cluster` after cloning, let's verify:

```
ls slurm-docker-cluster
```

This should show a list of files, including `docker-compose.yml`.

- b. Now let's set up, go into the folder:

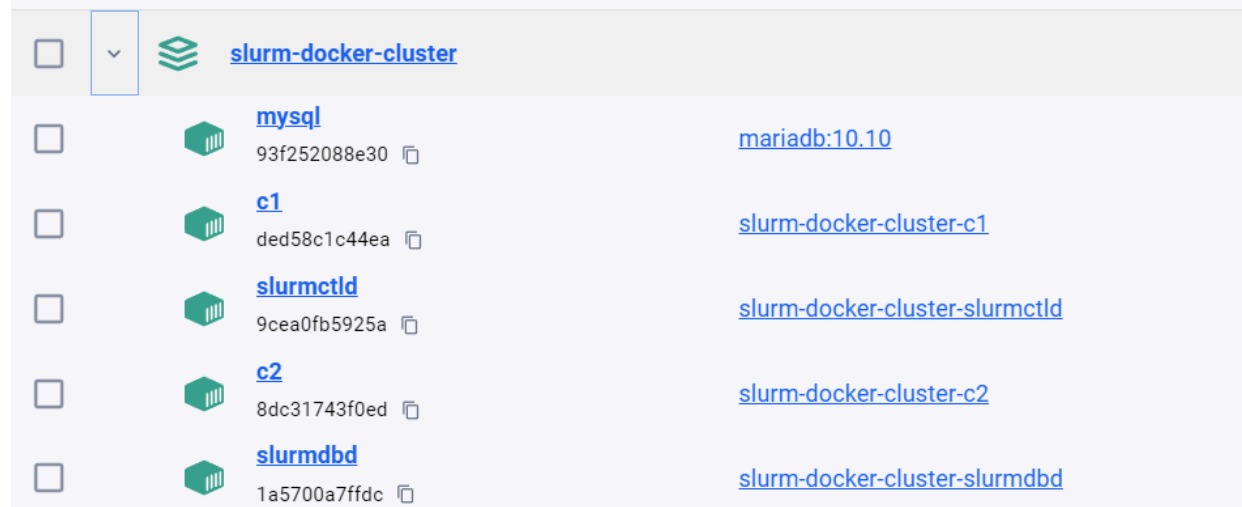
```
cd slurm-docker-cluster
```

- c. First, we'll copy the public SSH key we created before to our current folder for the Slurm cluster access:

```
cp ~/.ssh/id_rsa.pub .
```

Don't miss the '.' at the end, it is part of the command and means "this directory" (which is still `/slurm-docker-cluster` since we didn't `cd` anywhere else yet)

- d. Now we build and run our Slurm containers using our favorite `docker-compose`
`docker-compose up -d --build`
*This tells Docker to **build** all containers that are part of this Slurm cluster, and then to **up** (turn on the containers) the cluster, in a detached **-d** style so that it doesn't take over our terminal. You could also separate the commands into `docker-compose build` followed by a `docker-compose up -d`.*
- e. After that is done (building will take a while the first time) you should see a lot of containers active after the `up -d`. E.g. check Docker Desktop:

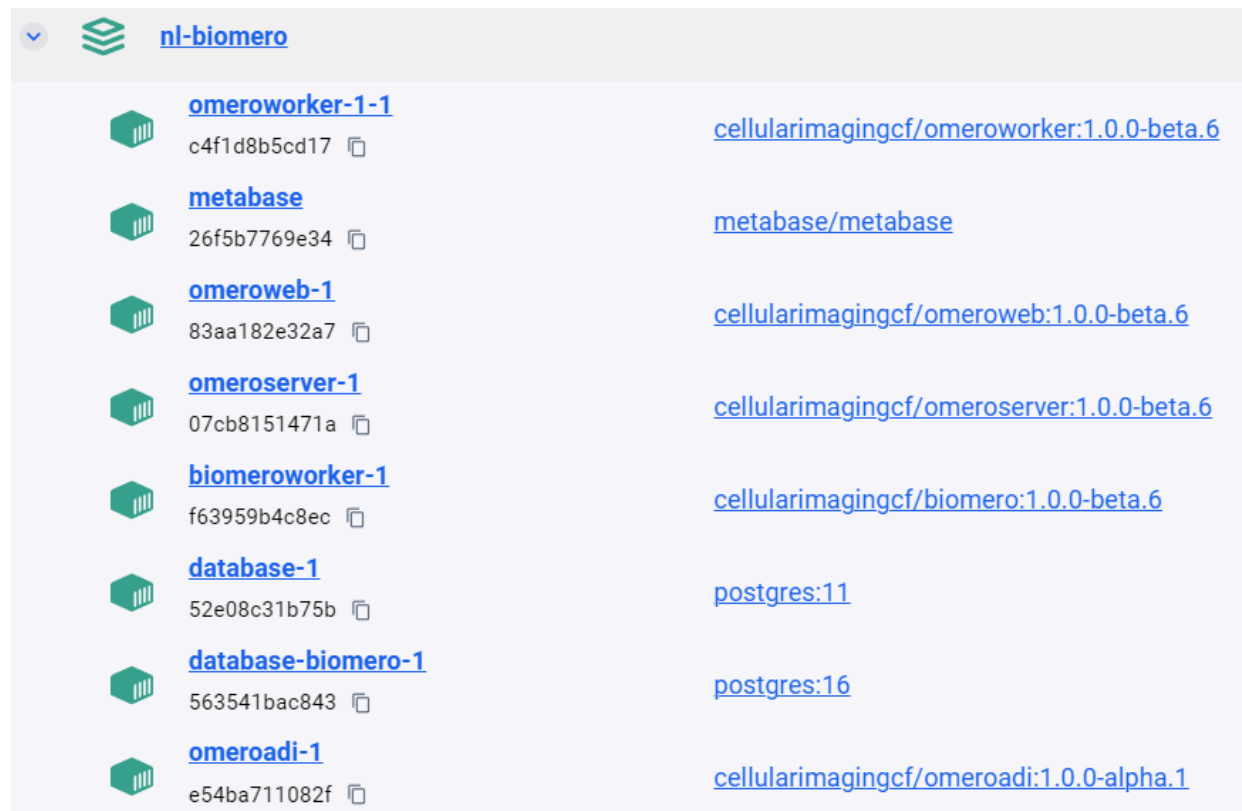


- f. Finally, we want to test that we can access it with our SSH:
`ssh -i ~/.ssh/id_rsa -p 2222 -o StrictHostKeyChecking=no slurm@localhost`
 This should open a terminal inside the `slurmctld` container showing
`[slurm@slurmctld data]$`.
Great, SSH is working from your host computer, and you have a Slurm cluster!
- g. Now quit the SSH session again
`exit`

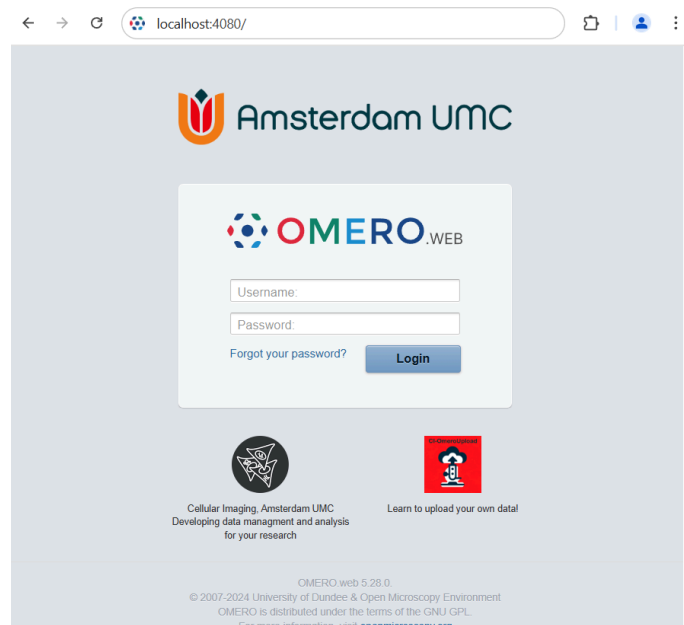
4. Set up NL-BIOMERO

9. Next, let's setup OMERO w/ BIOMERO, ADI & CANVAS: the **NL-BIOMERO**
 - a. First, go back to your workspace:
`cd D:/workspace`
 - b. You should have a subdirectory called NL-BIOMERO, after the clone, let's verify:
`ls NL-BIOMERO`
This should show a list of files, including
`docker-compose-from-dockerhub.yml`.
 - c. Get in there
`cd NL-BIOMERO`
 - d. Now let's spin up NL-BIOMERO containers, but slightly different than before (*it will take a while again; this time it's downloading from the internet*):
`docker-compose -f .\docker-compose-from-dockerhub.yml up -d --build`

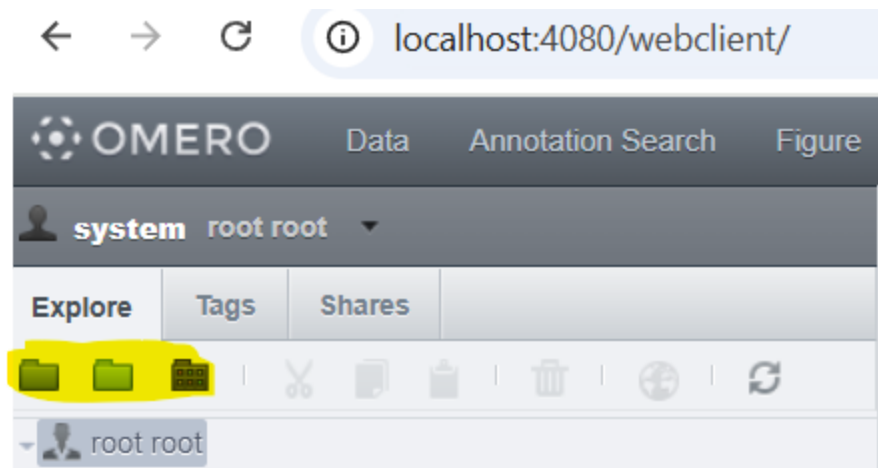
- e. You should see a lot of containers active after the `up -d`, e.g. in Docker Desktop:



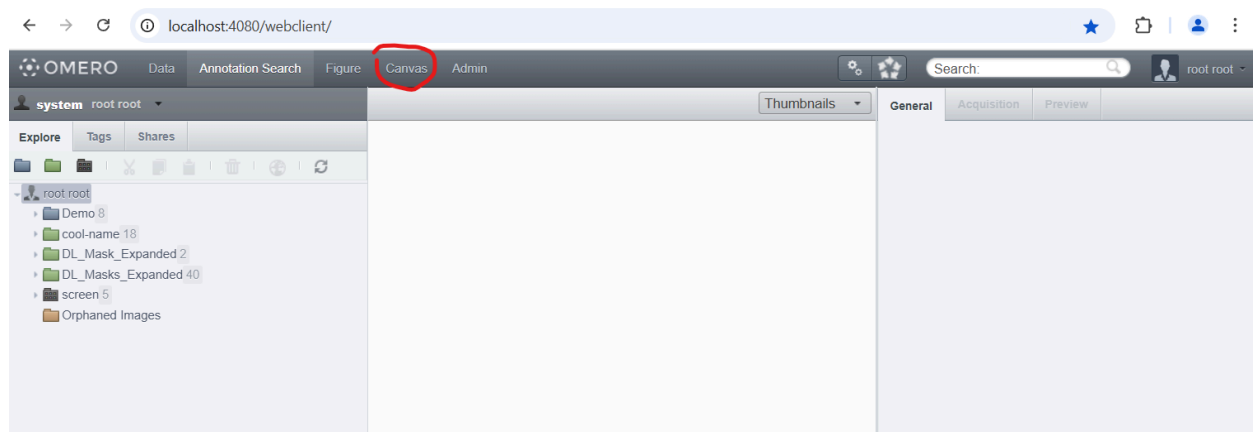
- f. Now, you can access OMERO at `localhost:4080` with user `root` pw `omero`
Note that OMERO server takes a bit before accepting login requests from web (max 1 minute). Try again later.



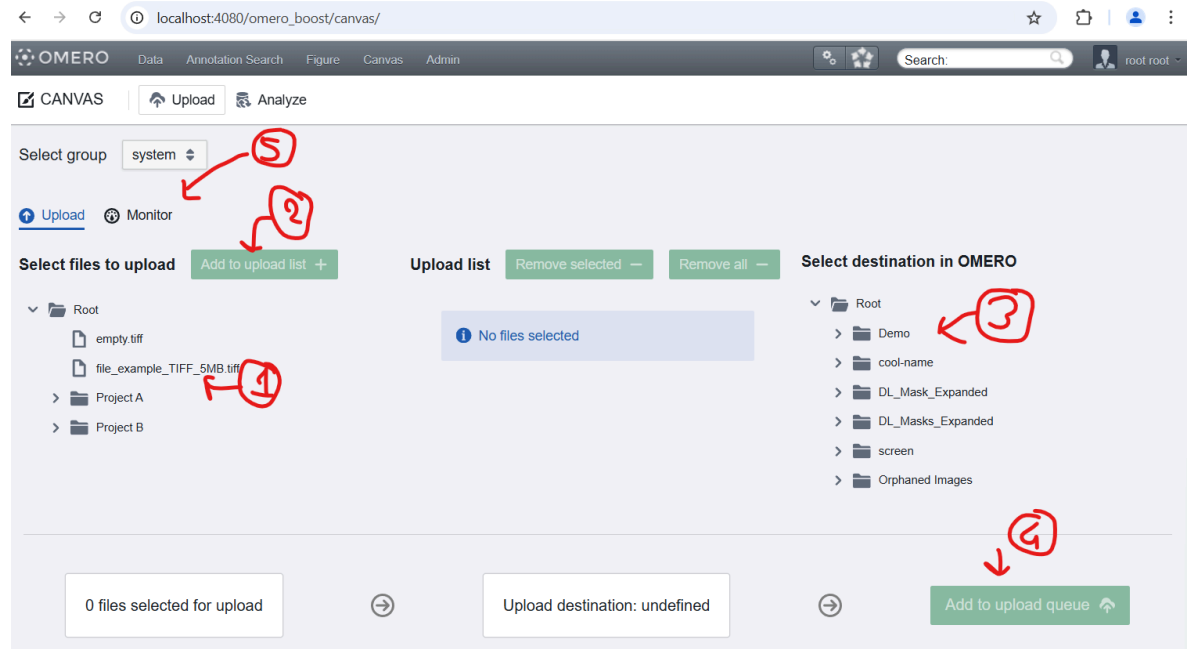
- g. Create some new Projects, Datasets and Screens



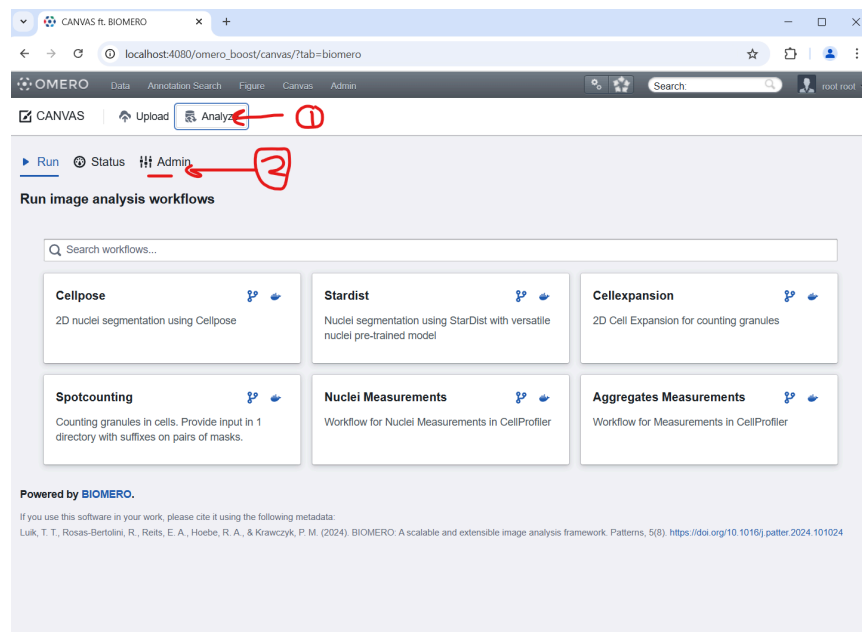
- h. Then, you can now access the **CANVAS** plugin at the top:



- i. When you want to import your own files using **ADI**, place the files in (a subfolder of) the folder **D:\Workspace\NL-BIOMERO\web\L-Drive**
- j. Now use the **CANVAS** plugin in OMERO.web to initiate some imports with **ADI**



- k. Next, we need to initialize BIOMERO once; Go to **Analyze** within **CANVAS** and then to the **Admin** tab



- l. Here, we start the **Slurm Init** script to download all the workflow containers that we have configured to our Slurm cluster. Click **Run script** (also in the pop-up).

← → ↻ localhost:4080/omero_boost/canvas/?tab=biomero ☆ 🔍 root:root

OMERO Data Annotation Search Figure Canvas Admin

CANVAS Upload Analyze

Admin

Settings

View or edit your settings for BIOMERO here!

Note that some settings will apply immediately (like a model's Additional Slurm Parameters), but others might require setup.

I would recommend running the **Slurm Init** script after changing these settings. You can also use **Slurm Check Setup** to see if its needed.

Please check the **BIOMERO** documentation for more info.

- > SSH Settings
- > Slurm Settings
- > Analytics Settings
- > Converters Settings
- > Models Settings

Note on saving BIOMERO settings

Note that there are possibly **multiple** config files that BIOMERO reads from and combines into 1 final configuration.

By default (in this order):

- (1) /etc/slurm-config.ini
- (2) and ~/slurm-config.ini
- (3) and environment variables that you set

We write these values in (2) the local ~/slurm-config.ini, but read also from (1) the system-wide /etc/slurm-config.ini. So it could be that **removing** some setting here doesn't work because they are set in /etc/slurm-config.ini: if so, please contact your system administrator to change that file. **Adding** and/or **overwriting** values should always work, because ~/slurm-config.ini is read and applied last (but before environment variables).

[Save Settings](#) [Undo All Changes](#)

Search scripts...

[Upload Script](#)

biomero 10 omero 16

Data

Admin

Minimal Slurm Script

Example script to run on slurm cluster Runs a script remotely on SLURM.
Connection ready? True

Authors: Torec Luik

Version: 2.0.0-alpha.4

[Run script](#)

Slurm Check Setup

Check Slurm setup, e.g. available workflows.

Authors: Torec Luik

Version: 2.0.0-alpha.4

[Run script](#)

Slurm Init

Will initiate the Slurm environment for workflow execution. You can provide a config file location, and/or it will look for default locations: /etc/slurm-config.ini ~/slurm-config.ini

Authors: Torec Luik

Version: 2.0.0-alpha.4

[Run script](#)

Run Slurm Init - Google Chrome

localhost:4080/webclient/script_ui/41153/

Slurm Init

Will initiate the Slurm environment for workflow execution.

You can provide a config file location, and/or it will look for default locations:

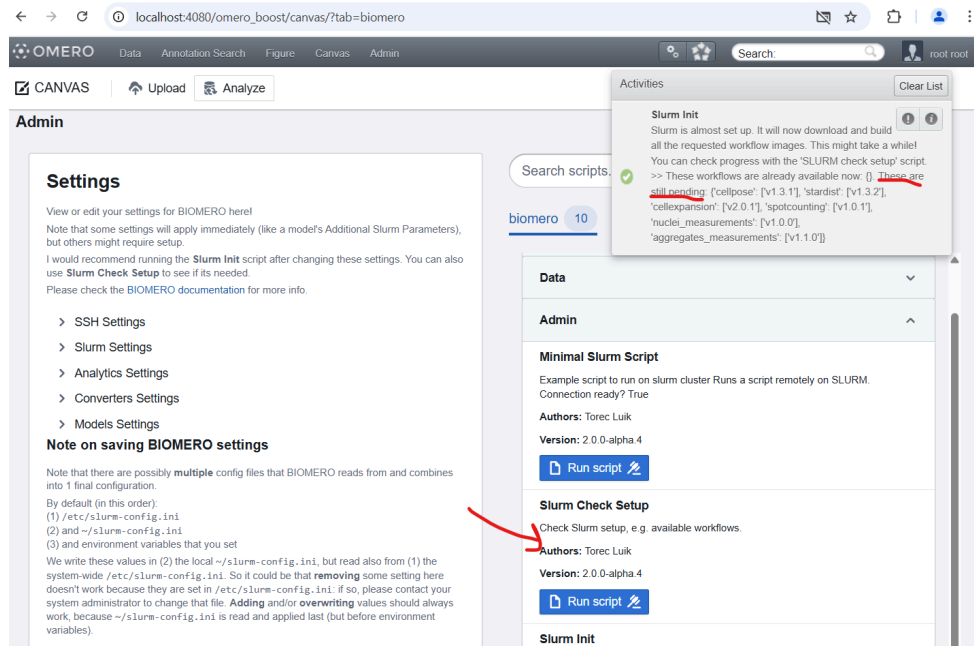
- /etc/slurm-config.ini
- ~/slurm-config.ini

Authors: Torec Luik
Contact: cellularimaging@amsterdamumc.nl
Version: 2.0.0-alpha.4

Init Slurm ☒

Extra Config file (optional):

[View Script](#) [Cancel](#) [Run Script](#)



- m. The script tells us that it is downloading the containers (These are still pending). This will take a while (> 10 minutes); you can check if there is any progress with the **Slurm Check Setup** script (look for **Available Models** instead of **Pending Models**).

Now you've got the whole NL-BIOMERO setup on your own computer!