2024.04 workshop

Before we start….

Get familiar with BerzeLiUs

Get yourself ready to log on to BerzeLiUs:

NSC Getting a login account

More detailed instructions on using ThinLinc remote desktop:

NSC Running graphical applications

We are going to use the Presto setup:

NSC CryoEM-PReSTO

Everything you need to know about how to get going on BerzeLiUs:

NSC Berzelius 🤩🥇

Berzelius: Following the link you'll get following info

- Introduction
- Getting Access to BerzeLiUs
- Login to BerzeLiUs
- Data Storage on BerzeLiUs
- Data Transfer from/to BerzeLiUs For data transfers between BerzeLiUs and your local computer, please use scp or rsync

- Modules and Build Environment NSC has a long list of software installed, and often in multiple versions to suit the needs of various user communities. The module system enables users to see what versions of what software packages are available, choose the ones they need for their work and have them set up correctly for their session, and not be bothered by all the rest of the software. In some cases, NSC also uses the module system to indicate what software versions are recommended to use, and which versions are recommended *not* to use.
- System Status
- User Support Mail any support issues to `BerzeLiUs-support@nsc.liu.se` or use the interface available in SUPR. Please report the following information when you encounter problems and obstacles:
  - A general description of the problems
  - Job IDs
  - Error messages
  - Commands to reproduce the error messages

  The support mail address is also the interface to make feature requests to add to BerzeLiUs, and we also have the possibility to bring in the BerzeLiUs vendor Atos or NVIDIA, should there be issues where extra support is needed.
- BerzeLiUs Events
- Research Projects on BerzeLiUs
- Frequently Asked Questions
- Acknowledgement

Information about SLURM job scheduler, the type of GPUs available on BerzeLiUs, the infamous BerzeLiUs GPU Usage Efficiency Policy, and the MIG lane:

NSC Berzelius GPU User Guide

Berzelius GPU User Guide: Following the link you'll get following info

1. CUDA: how to get info about GPU (including usage)
2. SLURM: is an open-source, highly configurable, and widely used workload manager and job scheduler for high-performance computing (HPC) clusters.
3. Interactive Sessions: An interactive session allows you to work directly on the cluster, interact with the compute nodes, and run commands in a real-time, interactive manner. Interactive sessions are useful for tasks like code development, testing, debugging, and exploring data.
4. Submitting Batch Jobs: In the context of HPC clusters, batch jobs are computational tasks that are submitted to a job scheduler for execution. Batch job submission is a common way to efficiently manage and execute a large number of computational tasks on HPC systems.
5. NSC boost-tools: to add more flexibility to the job scheduling
6. NVIDIA Multi-Instance GPU MIG: is a feature which allows a single GPU to be partitioned into multiple smaller GPU instances, each of which can be allocated to different tasks or users. This technology helps improve GPU utilization and resource allocation in multi-user and multi-workload environments.
7. Multi-node Jobs: Multi-node jobs for regular MPI-parallel applications
8. GPU Reservations: how to reserve GPUs/nodes for a specific time period.
9. Resource Allocations Costs: Depending on the type of resources allocated to a job the cost in GPUh will vary.
10. GPU Usage Efficiency Policy: As the demand for time on BerzeLiUs is high, we need to ensure that allocated time is efficiently used. The efficiency of running jobs is monitored continuously by automated systems.

CryoSPARC HPC software system architecture

CryoSPARC Architecture and System Requirements | CryoSPARC Guide

Transferring data to/from BerzeLiUs and checking your storage quotas

The instructions are provided under the link NSC Berzelius and NSC Berzelius.

- Quotas and your current usage can be checked with the command

```
nscquota
```

You can use `ncdu` to check which folders are taking the most space.

```
ncdu /home/username
ncdu /proj/your_proj/users/username
```

- For data transfer, use either scp, rsync or Filezilla.
  If you are a Windows and you chose to use MobaXTerm (MobaXterm) as a terminal emulator you can also use its built-in Scp/Sftp protocols for data browsing and transfer.

How to monitor Slurm jobs

NSC Introduction

Introduction to batch jobs: Following the link you'll get, among oter things, useful info about:

1. **Monitoring a batch job** (probably the most relevant from the CryoSparc data processing point of view)
   `squeue -u $USER`
2. Ending a queued or running job
3. What happens when a job starts?
4. Choosing a time limit for your job

Moving projects between different locations

It is possible to move your entire CryoSparc project to a different cluster or your local machine to continue the data processing there or to import a project that you already started somewhere else into the BerzeLiUs. In such a case you would need to Detach or Archive the project at the old location, transfer it to the new location, and then Attach or Unarchive it there.

The differences between the two, example scenarios and all the useful details on transferring, exporting, and importing data are described in CryoSparc's Guide: Data Management in CryoSPARC (v4.0+) | CryoSPARC Guide.

Prerequisites for the tutorial

To follow the tutorial, you must:

1. Set up an account in NAISS SUPR.
2. Once you have the account, you should request membership in the project `berzelius-2024-20`, "User workshop for cryo-EM data processing on BerzeLiUs".
3. Once you are granted membership in the tutorial project `berzelius-2024-20` you should also get an account on BerzeLiUs. Check the "Accounts" section on the NAISS SUPR portal and request a BerzeLiUs account if needed.
4. Get yourself a cryoSPARC license CryoSPARC

Step-by-step instructions for the tutorial

Logging in to the BerzeLiUs cluster

Log in to one of the two BerzeLiUs login nodes

```
ssh $USER@berzelius1.nsc.liu.se
```

or

```
ssh $USER@berzelius2.nsc.liu.se
```

Use the `Password` you got while setting up your BerzeLiUs account. If you have the two-step authentication activated you have to also install and set up the Google Authentication app on your phone which will show `Verification code` needed for logging in.

After successful login, you should see:

`**** Project storage directories available to you:`
`/proj/berzelius-2024-20`

There are two shared storage areas set up for your use:

- the home directory `/home/$USER`, nightly backed-up and small (20 GB quota per user)
- the project directory `/proj/berzelius-2024-20/users/$USER`

Get organized

Before running cryoSPARC it's a good idea to organize yourself and create a specific folder, for example for cryosparc's database and the workshop-related folder.

Go to your project folder

```
cd /proj/berzelius-2024-20/users/$USER
```

create folders for the cryosparc database

```
mkdir cryosparc_datadir
```

and a folder where you will process the workshop data

```
mkdir workshop_2024
```

go to your workshop folder

```
cd workshop_2024
```

create an image folder

```
mkdir movies
```

go to your movies folder

```
cd movies
```

link to the images of the dataset

```
ln -s /proj/berzelius-2024-20/datasets/workshop2024/data/*frameImage.tif .
```

go back to your workshop folder

```
cd ..
```

create a link to your gain file

```
ln -s /proj/berzelius-2024-20/datasets/workshop2024/data/K2-gain170629.mrc .
```

Setting up cryoSPARC at BerzeLiUs

> The bellow instructions are from NSC cryoSPARC on Berzelius

Go to your home directory at BerzeLiUs:

```
cd ~
```

The tilde sign (~) represents the home directory. You can check the exact path of the current directory using the `pwd` command, short for "print working directory".

We need to set up a file called .cryosparc-license with the license information. Type:

```
cat << EOF > .cryosparc-license
```

Then press the "Enter" key on your keyboard. Once you see the ">" symbol, please paste your CryoSparc license key immediately after it.

```
> cryoSPARC-license-key
```

Press “Enter” again. Once you see the ">" symbol, please paste your CryoSparc license key immediately after it.

```
> your-email-address
```

Press “Enter” and type `EOF`. Then press enter again.

```
>EOF
```

Your license file has been set successfully.

You only need to set up this file the first time you want to run cryoSPARC!

Make sure to replace "cryoSPARC-license-key" with your actual cryoSPARC license key, and replace "your-email-address" with your email address!

You can create the `.cryosparc-license` file using any text editor. This file should contain two lines: the first line with your cryoSPARC license key and the second line with your email address. It's important to remember that this file should be saved in your home directory at BerzeLiUs.

You can check if the file exists in the folder by displaying the directory contents using the command below:

```
ls -la
```

To view the contents of a file, you can use a command such as:

```
cat ‘name_of_the_file’
```

A cryosparc user will be created for you automatically when starting cryosparc. This user is personal, and no other users should be created by you. Doing so could break the Terms of Service for the cluster.

As of cryosparc 4.2.1 and forwards, cryosparc is run on a login node (BerzeLiUs1 or BerzeLiUs2), from where jobs are scheduled to run on compute nodes. A few adaptions were made to cryosparc to get it to run OK on BerzeLiUs, described below. For general information about cryosparc, look at the official documentation at https://guide.cryosparc.com/

Starting cryoSPARC at BerzeLiUs

Briefly, once you are logged in to BerzeLiUs go to the folder where you intend to locate your cryoSPARC database

```
cd /proj/berzelius-2024-20/users/$USER/cryosparc_datadir
```

then load the current default cryoSPARC module by

```
module load cryosparc
```

now you can start cryoSPARC

```
cryosparc
```

```
...
app: started
app_api: started
----------------------------------------------------
CryoSPARC master started.
From this machine, access cryoSPARC and cryoSPARC Live at
http://localhost:39042

From other machines on the network, access cryoSPARC and cryoSPARC Live at
http://BerzeLiUs2.nsc.liu.se:39042
```

Note your access address (the port number `39042` in the example above might differ for your cryoSPARC instance).

In some cases, while starting cryosparc you may encounter the bellow problem:

```
echo found database .lock-file in /proj/berzeLiUs-XXXX-XX/users/$USER/cryosparc_datadir/database/ before t
```

To check if cryoSPARC is already ruining or to stop the process use `cryosparc status` and `cryosparc stop` commands, respectively. The command must be executed from the level of your cryosparc database directory (`cryosparc_datadir`).

An alternative way (or when you know you suspect an old instance of CryoSparc running but the above command does not report it) is to list CryoSparc-related processes using the command:

```
ps xww | grep -e cryosparc -e mongo
```

If you notice a process related to cryosparc in the system, please check if you have cryosparc running on the other BerzeLiUs login node. Log in to the other node and stop it if necessary.

If the `cryosparc stop` command does not work, you can try killing the cryosparc-related process using the following command:

```
kill ‘cryosparc_PID’
```

Please replace ‘cryosparc_PID’ with the cryosparc process ID that was reported by the `ps` command (refer to the instructions above).

Example:

```
(base) [x_piodr@berzelius2 cryosparc_datadir]$ ps xww | grep -e cryosparc -e mongo
1215136 pts/109  S+     0:00 grep --color=auto -e cryosparc -e mongo
2091831 ?        Ss     2:45 python /software/presto/e/9.6/software/cryosparc/4.4.1-foss-2021a-CUDA-11.3.1
2095346 ?        Sl   140:53 mongod --auth --nounixsocket --dbpath /proj/berzelius-2024-20/users/x_piodr/
2097452 ?        Sl    33:41 python -c import cryosparc_command.command_core as serv; serv.start(port=3905
2101195 ?        Sl     8:54 python -c import cryosparc_command.command_vis as serv; serv.start(port=39059
2101842 ?        Sl    40:11 python -c import cryosparc_command.command_rtp as serv; serv.start(port=39061
2104577 ?        Sl    14:02 /software/presto/e/9.6/software/cryosparc/4.4.1-foss-2021a-CUDA-11.3.1/cryosp
(base) [x_piodr@berzelius2 cryosparc_datadir]$ kill 2091831
(base) [x_piodr@berzelius2 cryosparc_datadir]$ ps xww | grep -e cryosparc -e mongo
1222458 pts/109  S+     0:00 grep --color=auto -e cryosparc -e mongo
```

After terminating the old CryoSPARC, you should be able to start it again without any issues.

If the cryoSPARC is already running but you forgot the port number used to access the cryoSPARC web interface then type

```
cryosparc status | grep CRYOSPARC_BASE_PORT
```

If you encounter the below error while calling the cryosparc command, ensure that you have loaded the cryosparc mode.

```
-bash: cryosparc: command not found
```

To check the modules currently loaded, use the command:

```
module list
```

If you don't see cryosparc in the list, then...

```
cryosparc module load
```

To start the CryoSparc with a specific port number use the command (replace the 39042 with your desired port number):

```
STARTING_PORT=39042 cryosparc
```

If CryoSparc cannot find an available base port number during startup, there are two solutions you can try.

1. First, you can log in to the other BerzeLiUs login node and attempt to start CryoSparc there.
2. Alternatively, you can increase the search range for the port numbers by starting CryoSparc using the following command:

```
STARTING_PORT=39000 MAX_RUN_COUNTER=300 cryosparc
```

You can also try a combination of both solutions.

Lanes

The information about different “lanes” and how to add new lanes on BerzeLiUs is available under the link:

NSC cryoSPARC on Berzelius

Each “lane” is a separate SLURM script that sends your jobs to be executed on the cluster. Lanes specify what type of computing node (aka computing resources) you need for your job, to which project this job belongs etc. By default, four lanes are created inside your `cryosparc_datadir`: Thin, Fat, MIG, and Safe,. Each “lane” has a separate folder with the `cluster_script.sh` SLURM submission script.

	Node Type	GPUs	CPUs	RAM	VRAM/GPU	Local SSD	GPU/h cost
1	Thin	8 x NVIDIA A100	2 x AMD Epyc 7742 / 16 threads	1 TB	40 GB	15 TB	1
2	Fat	8 x NVIDIA A100	2 x AMD Epyc 7742 / 32 threads	2 TB	80 GB	30 TB	2
3	MIG	1/7th of the A100s' compute capabilities	2 cores / 4 threads	32GB	10GB		0.25

4. Safe Lane - Jobs running within this reservation will be safe from automatic job termination (GPU Usage Efficiency Policy). However, this reservation will be intentionally underprovisioned, so expect longer queue times. The lane is accessed through an additional flag `--reservation=1g.10gb` in the `cluster_script.sh`.

More info on the topic: NSC Berzelius GPU User Guide

In case you are a member of more than one project at BerzeLiUs you have to specify which project's computational allocation you want to use when submitting your jobs to the cluster. Otherwise, your jobs will fail with an error:

```
    ==========================================================================
    ==========================================================================
> -------- Submission command:
    sbatch /proj/berzelius-2024-20/users/x_piodr/workshop_2024/CS-workshop-2024/J1/queue_sub_script.sh
> Cluster script submission for P1 J1 failed with exit code 1
> sbatch: error: You are a member of multiple accounts(projects) on berzelius.
    sbatch: error: You need to specify which one to use by adding the --account (-A)
    sbatch: error: option to your command line or job script.
    sbatch: error: Batch job submission failed: Invalid account or account/partition combination specified
```

In such a case you need to add `-A` parameter to your SLURM submission script (`cluster_script.sh`) to specify which project allocation you want to use to submit the jobs. To do this you can modify the existing lane by adding an extra text line

`#SBATCH -A Berzelius-2024-20` (where Berzelius-2024-20 is the name of the project)

at the end of the `cluster_script.sh` (just before the last line saying `{{ run_cmd }}`).

After editing the `cluster_script.sh` all you need to do is to restart the CryoSparc and the lanes will be reconfigured automatically.

If you want to create a new separate lane with different parameters, you can follow these steps. First, create a new directory (named eg. "lane_new") inside your CryoSparc database directory. Then, copy the four files from any of the existing lanes to this new directory. After that, modify the name of the lane in the "cluster_info.json" file and edit the "cluster_script.sh" file according to your requirements. Once you've made all the necessary changes, restart CryoSparc and the new lane should appear automatically.

If you want to remove a lane from the CryoSparc instance, go to your CryoSparc database directory and type:

```
cryosparc cli "remove_scheduler_lane('<lane_name>')"
```

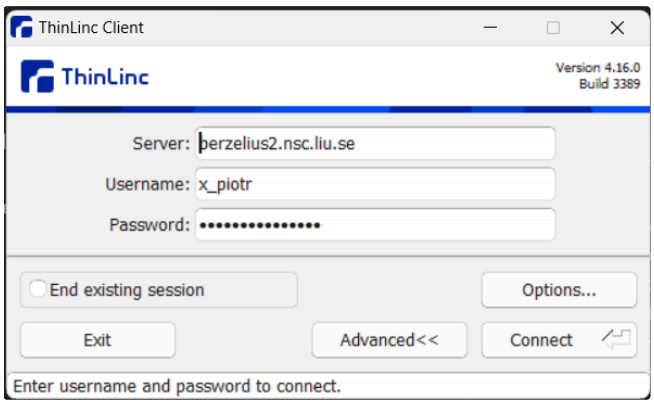
Once you restart CryoSparc, the lane should disappear.

More about adding new lanes to your cryoSPARC instance at NSC cryoSPARC on Berzelius and .

Accessing cryoSPARC @berzelius from your local computer

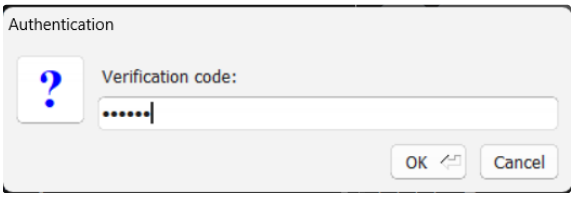
ThinLink remote desktop

After installing the ThinLink Client (ThinLinc downloads | ThinLinc by Cendio) on your computer, open the app and provide the BerzeLiUs server address and your login credentials



If you connect for the first time, you will see the "The server's host key is not cached ..." dialog. Verify that the fingerprint shown on your screen matches the one listed below! If it does not match, press Abort and then contact NSC Support!

It will then ask for a verification code from your two-step verification app (like Google Authenticator which you should have installed on your phone).



After a few seconds, a window with a simple desktop session in it will appear. From the Applications menu, start a Terminal Window. You are now logged in to BerzeLiUs and can submit jobs, start interactive sessions, and start graphical interfaces as usual.

SSH port forwarding

Windows users can either use the native PowerShell/Command Prompt or install MobaXterm (MobaXterm).
To open Windows PowerShell/Command Prompt open the "Start" menu and type "cmd". Click "Command Prompt". Alternatively, press "Windows" + "R" to open the Run program. Type "cmd" and press "Enter".

In addition to command prompt functionality MobaXterm also allows predefined connection sessions (so that you don't have to type ssh and the server address every time you connect), provides X-window forwarding (it can forward the graphical interface of the program you run remotely eg. Relion or Chimera), provides SFTP functionality (so that you can easily browse the file system on the remote server, and copy/transfer data).

To connect to BerzeLiUs and access cryoSPARC from your local web browser, follow these instructions:

Accessing the CryoSPARC User Interface | CryoSPARC Guide

1. In a fresh terminal type as below, but instead of `39042` put the port number from the step “Starting cryoSPARC at BerzeLiUs”.

```
ssh -N -L localhost:39042:localhost:39042 remote_hostname
```

with `remote_hostname`, depending on which of the two nodes you were logged in when starting cryoSPARC.

```
user_name@berzelius1.nsc.liu.se
```

```
user_name@berzelius2.nsc.liu.se
```

The shh port forwarding command should be executed on your local machine. Do not run this command in the terminal where you are currently logged in to BerzeLiUs.

2. Open the web browser and type

```
localhost:39042
```

(remember to change the port number to your value!)

3. Login to cryoSPARC using the email address you used to get your CryoSPARC licence and the licence ID you obtained from info@structura.bio as a password

During the workshop, some users ran into a problem while trying to log in to the CryoSparc web interface. Despite correct login details, the CryoSparc replied 'user not found'. Here is the solution provided by one of the users (thanks Tarvi!):

In case

- you have the correct `.cryosparc-license` file in your home directory
- you can start Cryosparc without any issues
- you can successfully forward the CryoSparc Web interface port using the `ssh -N -L` command
- and you can see the CryoSPARC login page in your browser

but still can't log in because of the 'user not found' error, then you may need to create a new user yourself. Before this, existing users can be checked with the command after you have started Cryosparc:

```
cryosparc listusers
```

If the list is empty, it may indicate that you do not have an existing user. To create a new user, use this command:

```
cryosparc createuser --email "$CRYOSPARC_EMAIL" --password "$CRYOSPARC_LICENSE_ID" --username "$USER" --fi
```

If this is successful, a new user should be visible using the `cryosparc listusers` command.

Restart Cryosparc (`cryosparc restart` command) and create a port forwarding connection using the previously mentioned `ssh -N -L` command.
Now you should be able to log in on the CryoSparc web interface.

Processing in cryoSPARC

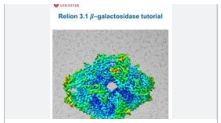
We are going to use the following dataset

EMPIAR-10204 The first reconstruction of beta-galactosidase solved by cryoARM200

that is available at this location:

`/proj/berzelius-2024-20/datasets/workshop2024/`

If you are interested the tutorial for Relion 3.1 is available:



Relion 31 Tutori…rs.pdf
12 Jan 2024, 03:07 PM

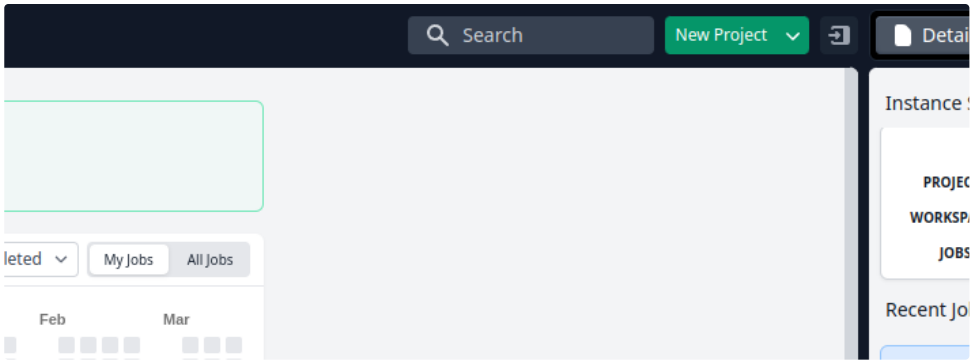
but we are going to process this first in cryoSPARC as BerzeLiUs is better suited to run this software package (for Relion you should consider using Tetralith cluster).

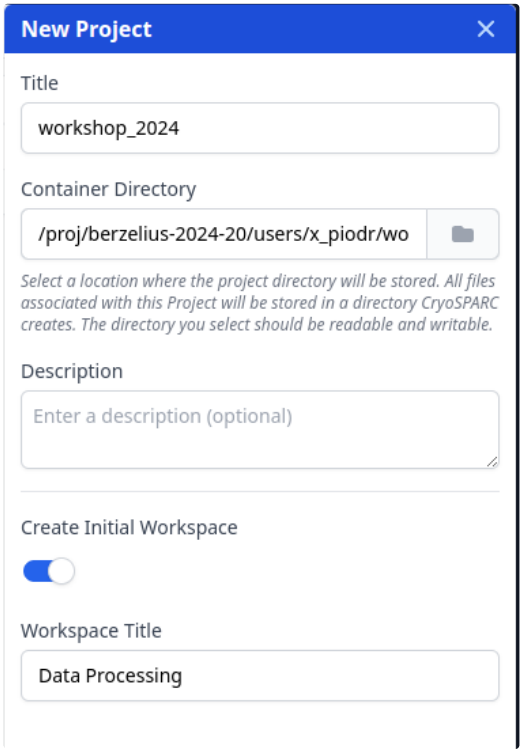
> The tutorial dataset consists of
>
> 1338 movies
>
> 49 frames per movie
>
> 0.885 Å/pixel size

more experimental details about the dataset collection here:

Electron Microscopy Data Bank

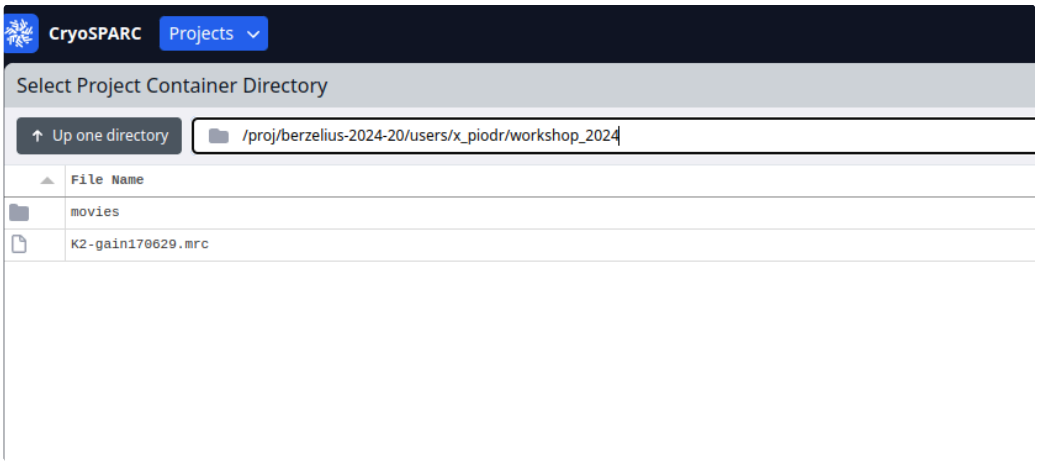
Create a New Project





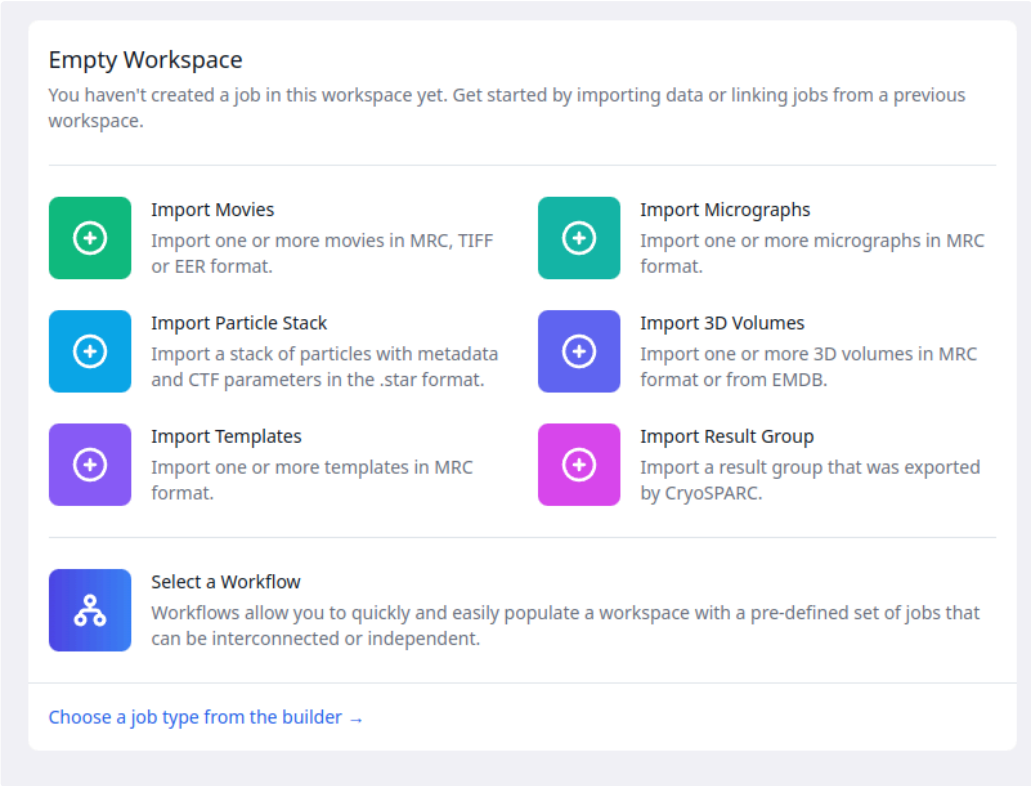
Toggle on the “Create Initial Workspace” option. (you can add more work spaces later by using the “New Workspace” button)

As the “Container Directory” use your `/proj/berzelius-2024-20/users/$USER/workshop_2024` folder



Finally click “Create” button at the bottom of the “New Project” panel

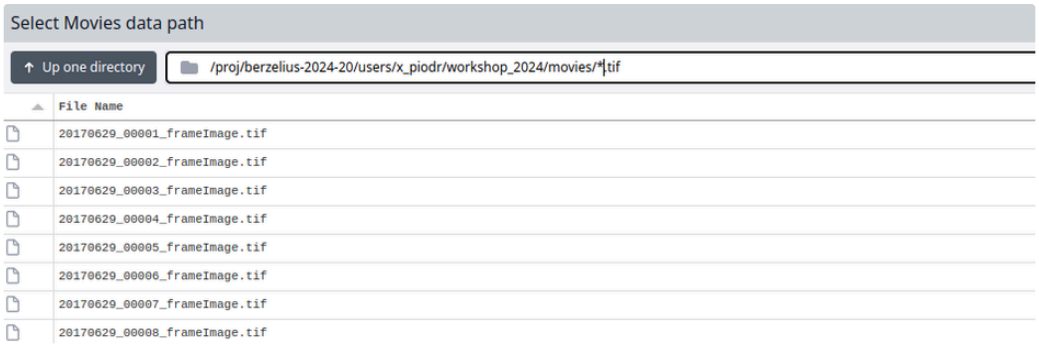
Import raw data (movies)



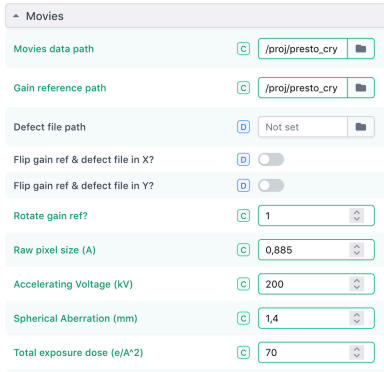
Click on the folder icon next to the “Movies data path” and go to the folder withe the raw movie files that we prepared during the “Get organized” step. select one of the movie .tif files. The full path to this file will appear in the path field but then substitute the name of the file with `*` wild card, and just leave the `.tif` extension.

`/proj/berzelius-2024-20/users/$USER/workshop_2024/movies/*.tif`

In this way all `tif` files from this folder will be imported.

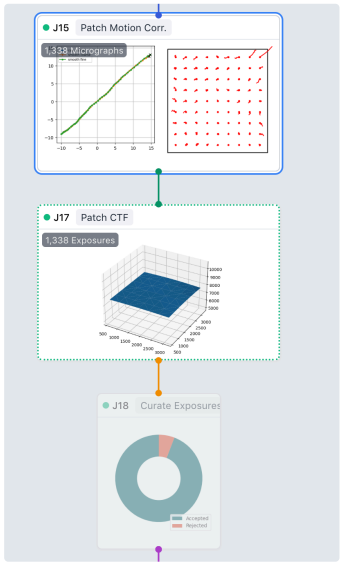


import your data: 7s



settings for importing the data

beware the rotate gain reference: the setting (1) rotates the gain image by 90°



workflow to pre-process micrographs

- Times indicated in the following processing steps will differ in your case. We have tested the processing previously without using any MIG flags.
- Your particle-picking strategy may vary, and your results could differ slightly. Explore which picking options work best for you!
- Particle downsampling may not be necessary

Motion correct and CTF

Patch Motion Correct: ~1h 23m (MIG: 2h 41min)

standard settings - nothing particular here.

Patch CTF: ~51 min

standard settings

Curate Exposures: interactive

To select micrographs based on certain thresholds.

```
Exposures accepted          : 1260
Exposures rejected          : 78
Exposures manually rejected : 0
Thresholds set:
    Average defocus (A) - Min: 1030.15 -> Max: 18366.15
    CTF fit resolution (A) - Min: 2.461 -> Max: 8.604
    Relative Ice Thickness - Min: 0.98 -> Max: 1.069
    Full-frame motion curvature - Min: 2.02 -> Max: 21.55
```

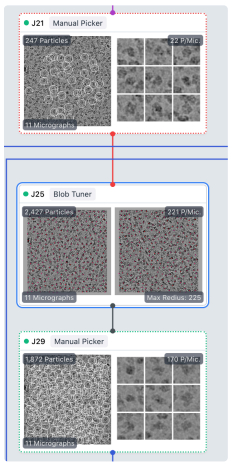
thresholds set - copy from logfile

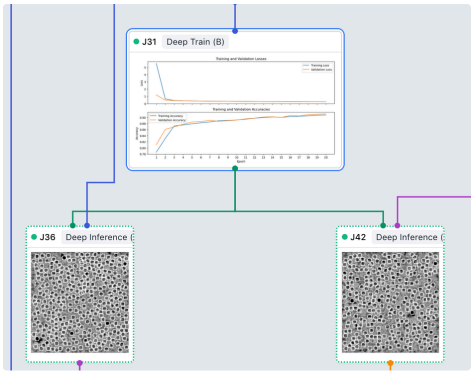
Particle picking

> ⚠ If you decide to use the Topaz convolutional neural network as the method to pick particles, please do not specify the path to the Topaz executable in CryoSparc. The connection between Topaz and CryoSparc@berzelius is preconfigured. Once you start your Topaz job, CryoSparc will automatically call the correct Topaz executable.

For the tutorial, we will use DeepPicker, which is a CryoSparc particle-picking method based on a convolutional neural network.

Workflow to pick particles:





after training check the picks with a small subset
(J36, left) before picking all (J42, right)

If you follow this tutorial, you realize that we created a smaller image subset only after we had already picked manually particles from a few images that were selected from the entire dataset.

When you process the data,

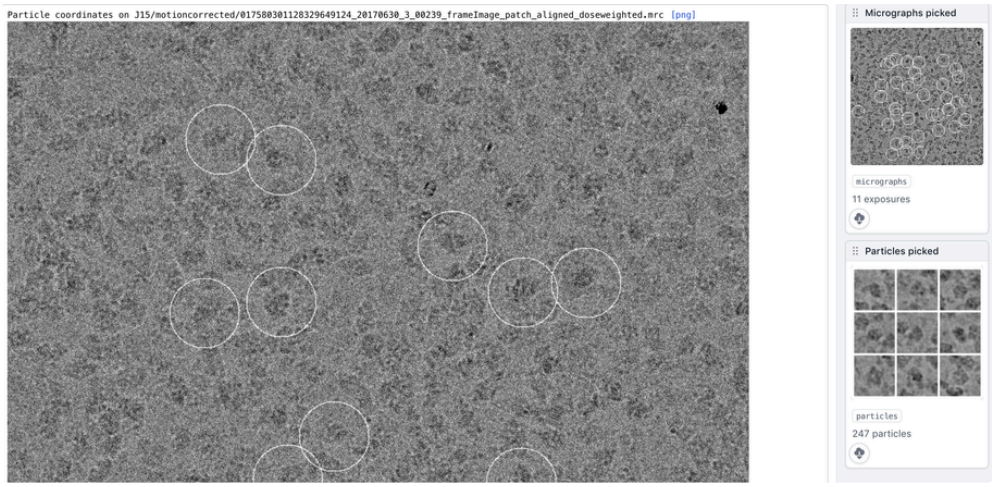
(1) first create smaller image subsets (somewhat between 10-50 micrographs)

(2) apply your favourite picking strategy on the first subset (training)

(3) apply your model/parameters as a test on the second small subset that has not been used for training (inference)

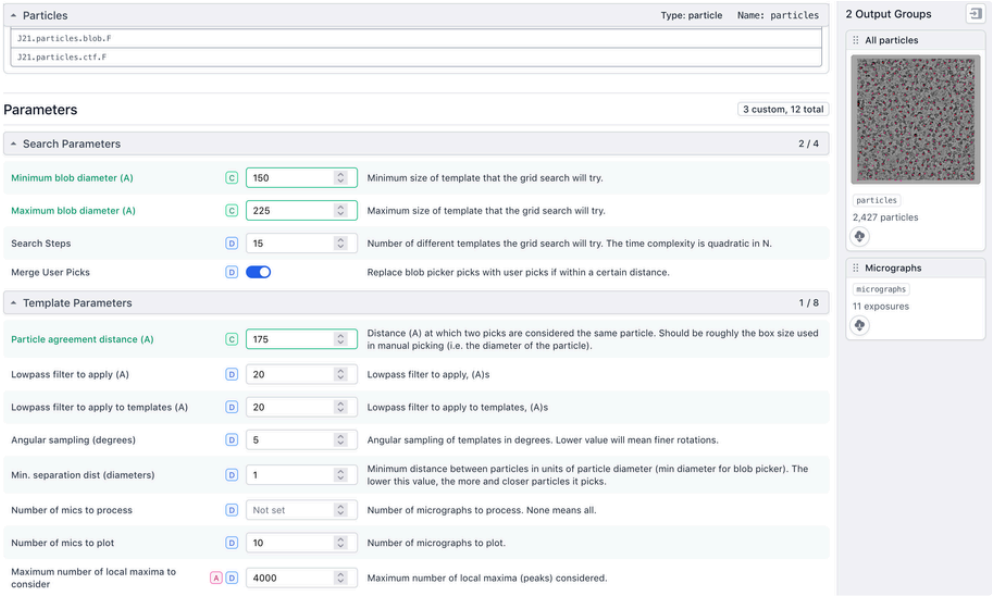
(4) then pick particles from the entire dataset

J21.Manual Picker: Interactive



picked about 250 particles from 11 micrographs (with varying defocus);
particle size of 175 Å

J25.Blob Tuner

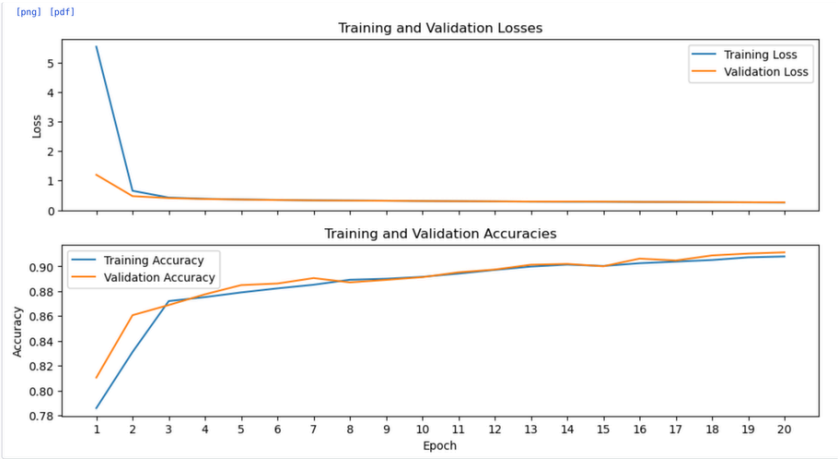


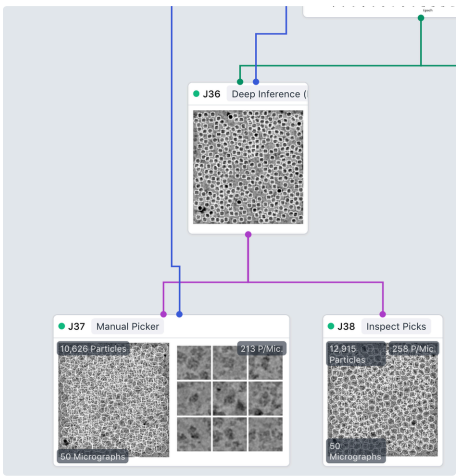
settings for tuning

J29: Manual Picker

To deselect a few bad picks

J31: Deep Train; ~3min





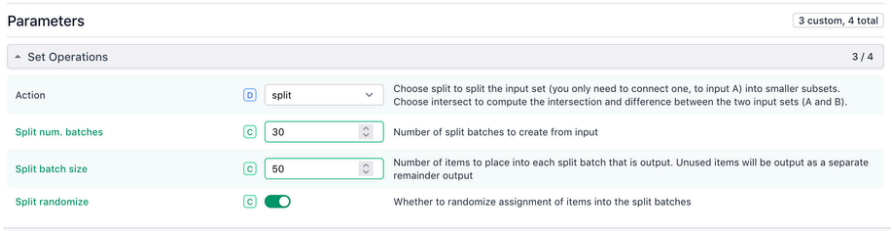
Check your picks with a small subset

Check your picks

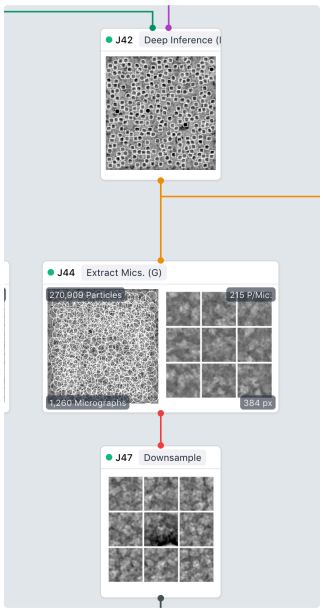
created before smaller subsets using 50 micrographs that were subset using the job type

Exposure Sets Tool

to split the dataset into batches with a size of 50 micrographs



Picks looked very fine and picking was performed on all micrographs in the following steps. After picking, particles were extracted and downsampled (optional; speeds up calculations).



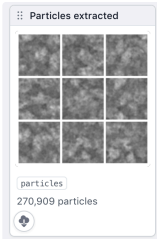
J42. Deep Picker Inference; ~11 min

picked 330k particles from 1260 micrographs

What are good box sizes?

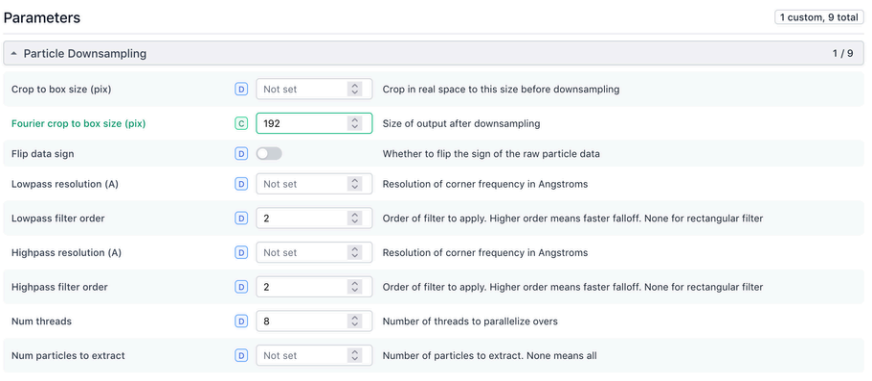
EMAN2/BoxSize - EMAN Wiki

J44.Extract from Micrographs ~10 min



extracted 270k particles - particles near edges were removed

J47. Downsample



Parameters — 1 custom, 9 total

Particle Downsampling — 1 / 9

Crop to box size (pix)	Not set	Crop in real space to this size before downsampling
Fourier crop to box size (pix)	192	Size of output after downsampling
Flip data sign		Whether to flip the sign of the raw particle data
Lowpass resolution (A)	Not set	Resolution of corner frequency in Angstroms
Lowpass filter order	2	Order of filter to apply. Higher order means faster falloff. None for rectangular filter
Highpass resolution (A)	Not set	Resolution of corner frequency in Angstroms
Highpass filter order	2	Order of filter to apply. Higher order means faster falloff. None for rectangular filter
Num threads	8	Number of threads to parallelize overs
Num particles to extract	Not set	Number of particles to extract. None means all

from 384 → 192 px

thus the px-size changes from 0.885 Å to 1.77 Å,

and the max achievable resolution for reconstruction from 1.77 Å to 3.54 Å

2D classes to filter particles

Two classification jobs with different mask settings - continued with the larger mask.

What are good settings for the mask? from NCBI - WWW Error Blocked Diagnostic

“The diameter of this mask is set slightly larger than the largest dimension of the complex.” typically 10% larger than the particle diameter.

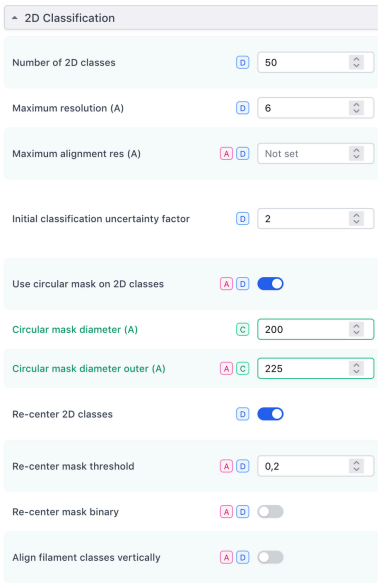
However, these authors

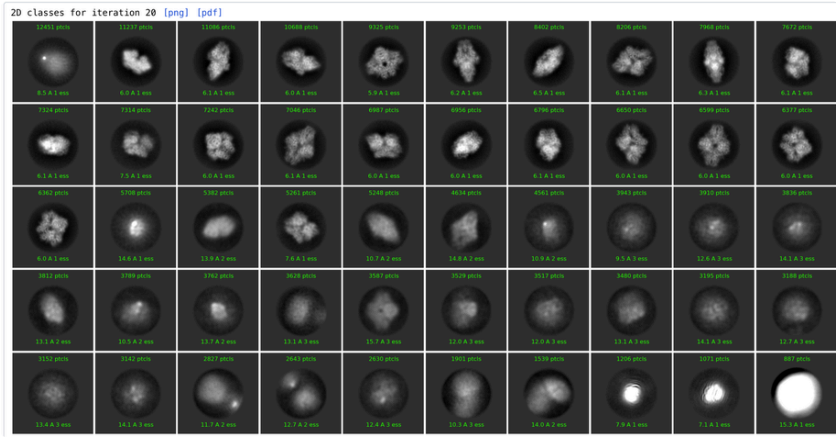
Size matters: optimal mask diameter and box size for single-particle cryogenic electron microscopy

propose to use larger masks (1.5 times the particle diameter). Both settings yielded good-looking classes.

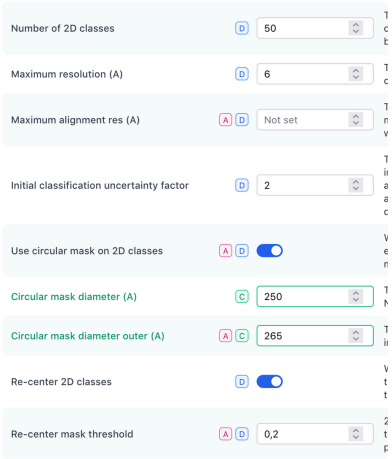
2D classes with smaller (200 Å) and larger (250 Å) masks. Both yielded reasonable classes.

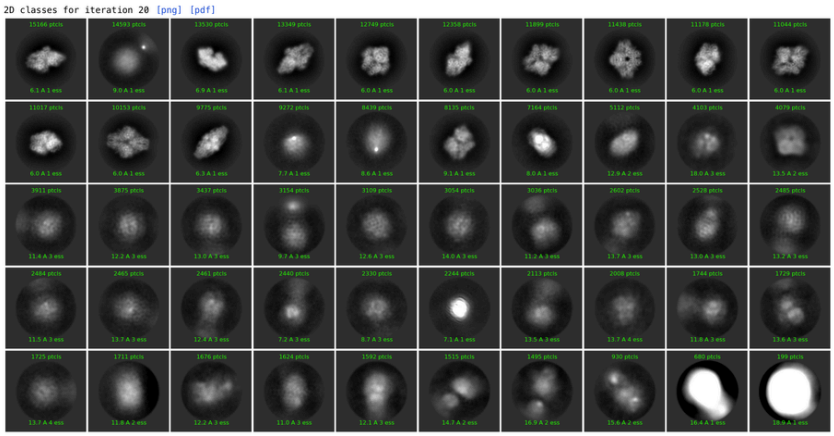
J51.2D Class; ~7 min





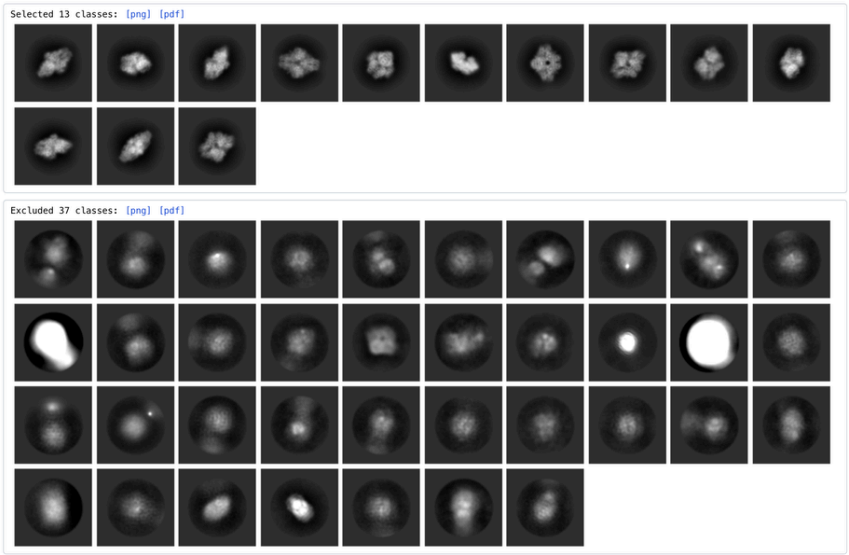
J52.2D Class; ~7 min





J53. Select 2D; interactive

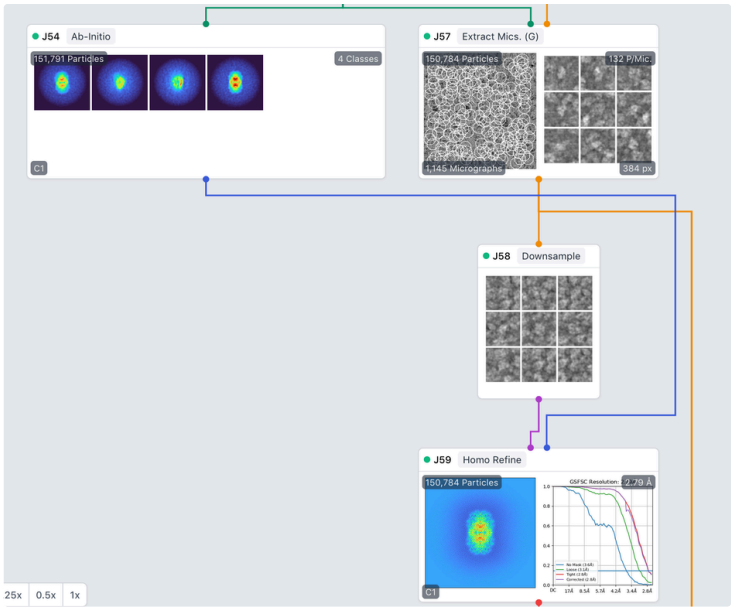
To de-select junk particles:



selected ~150k particles

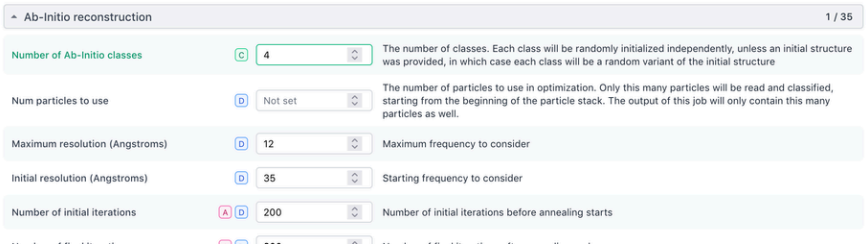
3D reconstruction and refinement

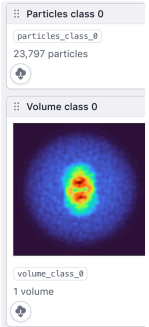
Selected particles were reconstructed ab initio asking for four volumes. The fourth volume was used for homogenous refinement, feeding particles that were re-extracted and downsampled to 256 px → 1.3275 Å (2.655 Å limit).

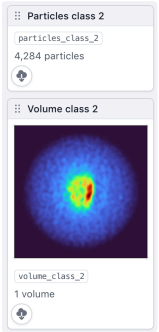


Initial 3D reconstruction steps

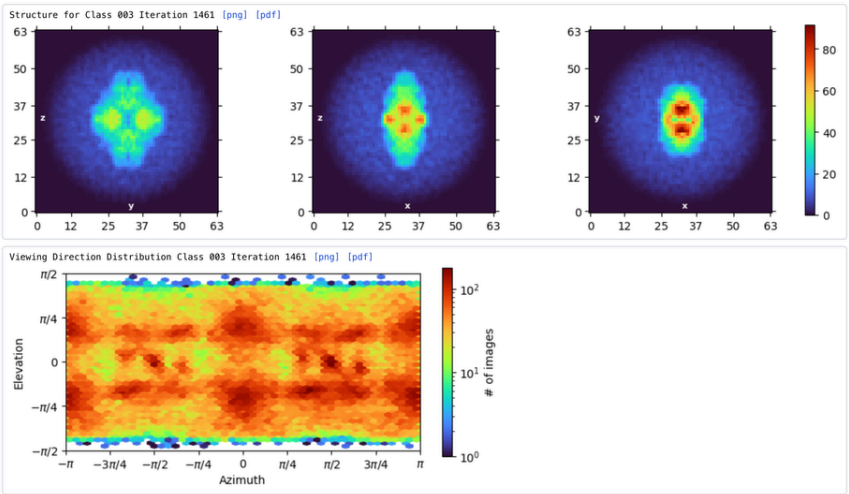
J54.Ab initio reconstruction







Class 3 had the most particles (118k particles) and looked promising:



J57.Extract Mics; ~13min

Micrographs — Type: exposure — Name: micrographs

J42.micrographs

Particles — Type: particle — Name: particles

J53.particles_selected

Parameters — 1 custom, 9 total

Compute settings — 0 / 1

Number of GPUs to parallelize (0 for CPU-only)	1	Micrographs will be split between the GPUs.

Particle Extraction — 1 / 8

Extraction box size (pix)	384	Size of box to be extracted from micrograph.
Fourier crop to box size (pix)	Not set	Size of particle boxes after they have been extracted. None means use the same as the extraction box size
Force re-extract CTFs from micrographs	off	Force the job to re-extract CTFs from the connected micrographs. Without this option, the job will retain CTFs that are attached to the input particles if present.
Recenter using aligned shifts	on	Whether or not to recenter the picks based on the alignment shifts. Particles will only be recentered upon extraction if this is true, and alignments3D or alignments2D are connected.
Number of mics to extract	Not set	Only extract a certain number of micrographs. None means all
Flip mic. in x before extract?	off	Flip the micrograph in the x axis before extraction
Flip mic. in y before extract?	off	Flip the micrograph in the y axis before extraction
Scale constant (override)	Not set	Override the scale constant used to scale the micrographs

J58.Downsample; ~3min

256 px → 1.3275 Å (2.655 Å limit)

J59.Homogeneous Refinement; ~13 min

Particle stacks — Type: particle — Name: particles

J58.particles

blob
J58.particles.blob.F

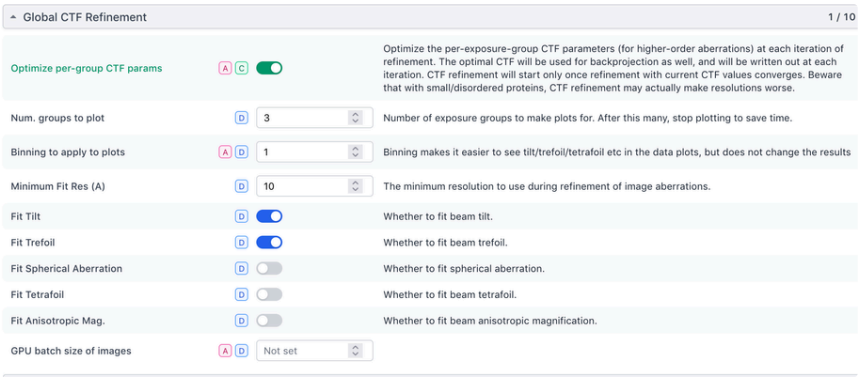
ctf
J58.particles.ctf.F

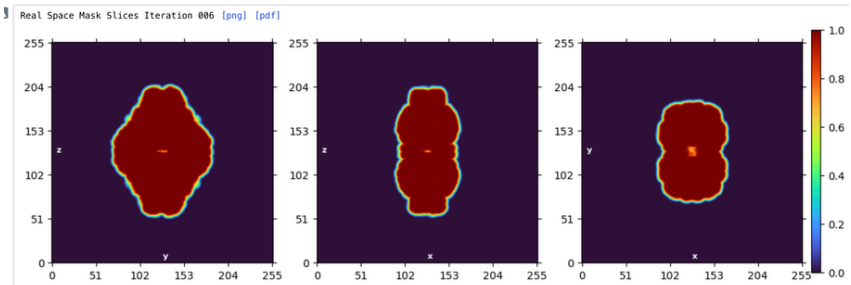
Passthrough
J58.particles.alignments2D.F
J58.particles.location.F
J58.particles.pick_stats.F
J58.particles.ml_properties.F

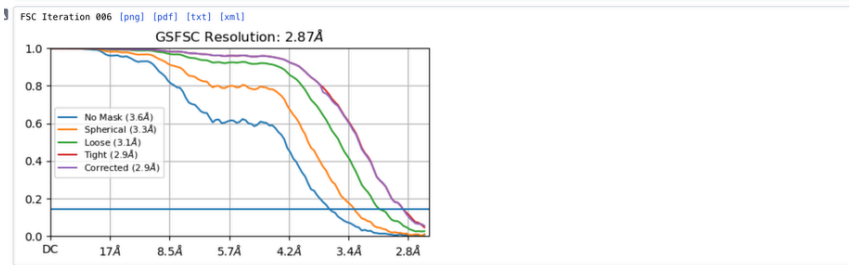
Initial volume — Type: volume — Name: volume

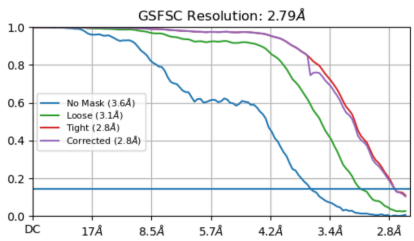
J54.volume_class_3

map
J54.volume_class_3.map.F









Due to downsampling, we hit the Nyquist limit; and we should also apply the symmetry D2 to the particles.

J60.Homogeneous Refinement; full resolution and D2 symmetry ~29 min

Particle stacks

blob

J57.particles.blob.F

ctf

J59.particles.ctf.F

alignments3D

J59.particles.alignments3D.F

Passthrough

J57.particles.location.F

J57.particles.alignments2D.F

J57.particles.pick_stats.F

J57.particles.ml_properties.F

Initial volume

J59.volume

map

J59.volume.map.F

Passthrough

J59.volume.map_sharp.F

J59.volume.map_half_A.F

J59.volume.map_half_B.F

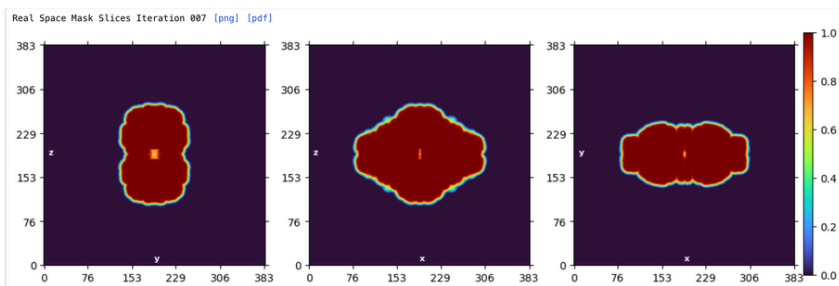
J59.volume.mask_refine.F

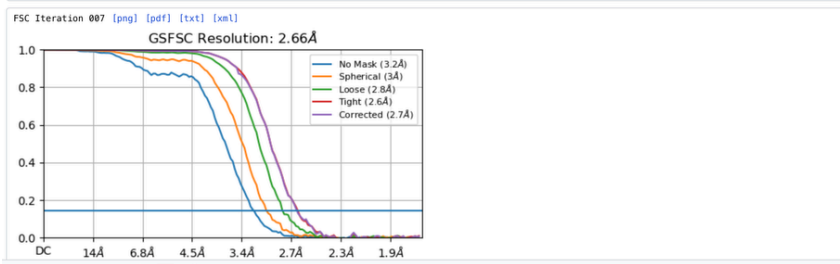
J59.volume.mask_fsc.F

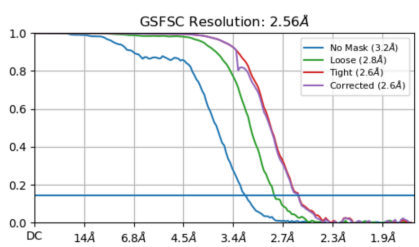
J59.volume.mask_fsc_auto.F

J59.volume.precision.F

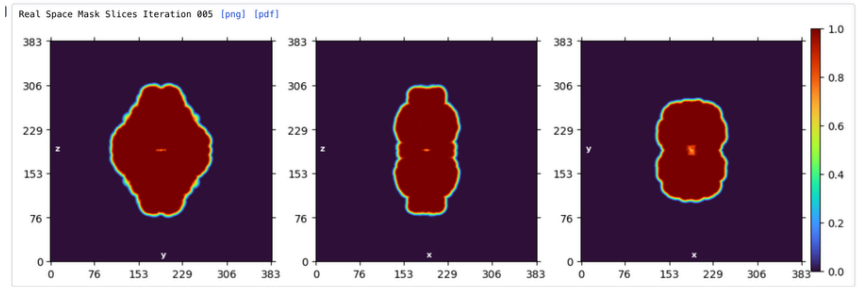
Note that we fed the alignment3D and ctf parameters from the previous refinement job. (not sure if this improved anything)

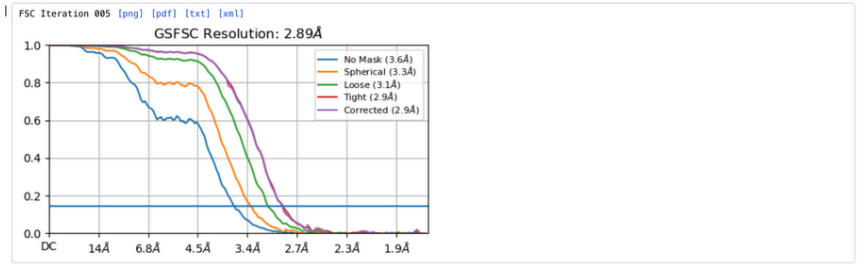






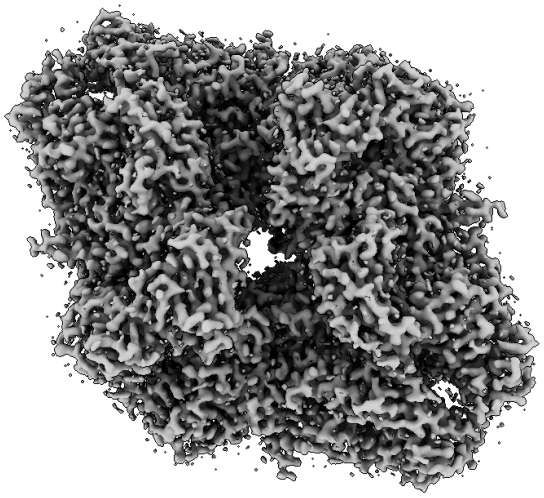
J61.Homogeneous Refinment; full resolution, no symmetry ~23 min





It's probably best to look at the map locally, after transferring the final volume via

```
scp
```



and the final map J60 at ~2.6 Å