

## Basic details on Neurodesktop on the HPC cluster.

Neurodesktop is a plug-and-play, browser-accessible, containerised data analysis environment.

More information can be found on the homepage: <https://www.neurodesk.org/> 

The following versions of Neurodesktop are available on the HPC cluster:

Version	hpc-env	Architecture
2024-12	13.1 , 12.2	zen4 , zen3

Note that to access a specific version of Neurodesktop, you need to load one of `hpc-env` modules listed above and you must use a compute node with listed architectures.

You can get the most recent information on available versions of Neurodesktop using the command

```
1 | $ module spider Neurodesktop
```

however, it will only list versions with the architecture given by `$LMOD_SYSTEM_ARCH`. Please refer to the guide on [the organisation of software modules](#) for more details.

# Using Neurodesk on the HPC Cluster

---

Neurodesk is a flexible and scalable data analysis environment for reproducible neuroimaging. This guide explains how to use Neurodesk on our HPC cluster.

## Quick Start

---

A script has been provided to simplify the startup process of Neurodesk. Although the script works on the login nodes as well, it is primarily designed and encouraged to be used on compute nodes through the Slurm workload manager.

### Starting Neurodesk

---

1. Load the required module:

```
1 | module load Neurodesktop
```

2. Start Neurodesk:

```
1 | neurodesktop-start
```

### Running on Compute Nodes (Recommended)

---

To run Neurodesk on a compute node, create a Slurm job script:

```
1 | #!/bin/bash
2 | #SBATCH --nodes=1
3 | #SBATCH --ntasks=1
4 | #SBATCH --cpus-per-task=4
5 | #SBATCH --mem=16G
6 | #SBATCH --time=08:00:00
7 | #SBATCH --output=slurm.%A.log
8 |
9 | module load Neurodesktop/2024-12
10 | neurodesktop-start
```

The script will automatically detect that it's running within a Slurm job and adjust resources accordingly.

## Accessing Neurodesk

---

After starting the script, two steps are required to access Neurodesk:

1. The script will display an SSH command that needs to be executed in a terminal on your local machine. Copy and paste this command! In case you ran `neurodesk-start` in a job script, the command as well as the URL from step 2 will be written to jobs log file. The ssh command will look similar to:

```
1 | ssh -N -L 12345:localhost:54321 username@hpc1101.hpc.uni-oldenburg.de
```

2. Wait for the server to start, which can take up to a couple of minutes and will produce a lot of output which you don't have to worry about.

Eventually, the script will display a URL starting with `http://127.0.0.1` followed by a port number and authentication token. Copy this URL into your web browser.

## Accessing and Using the Graphical Desktop

---

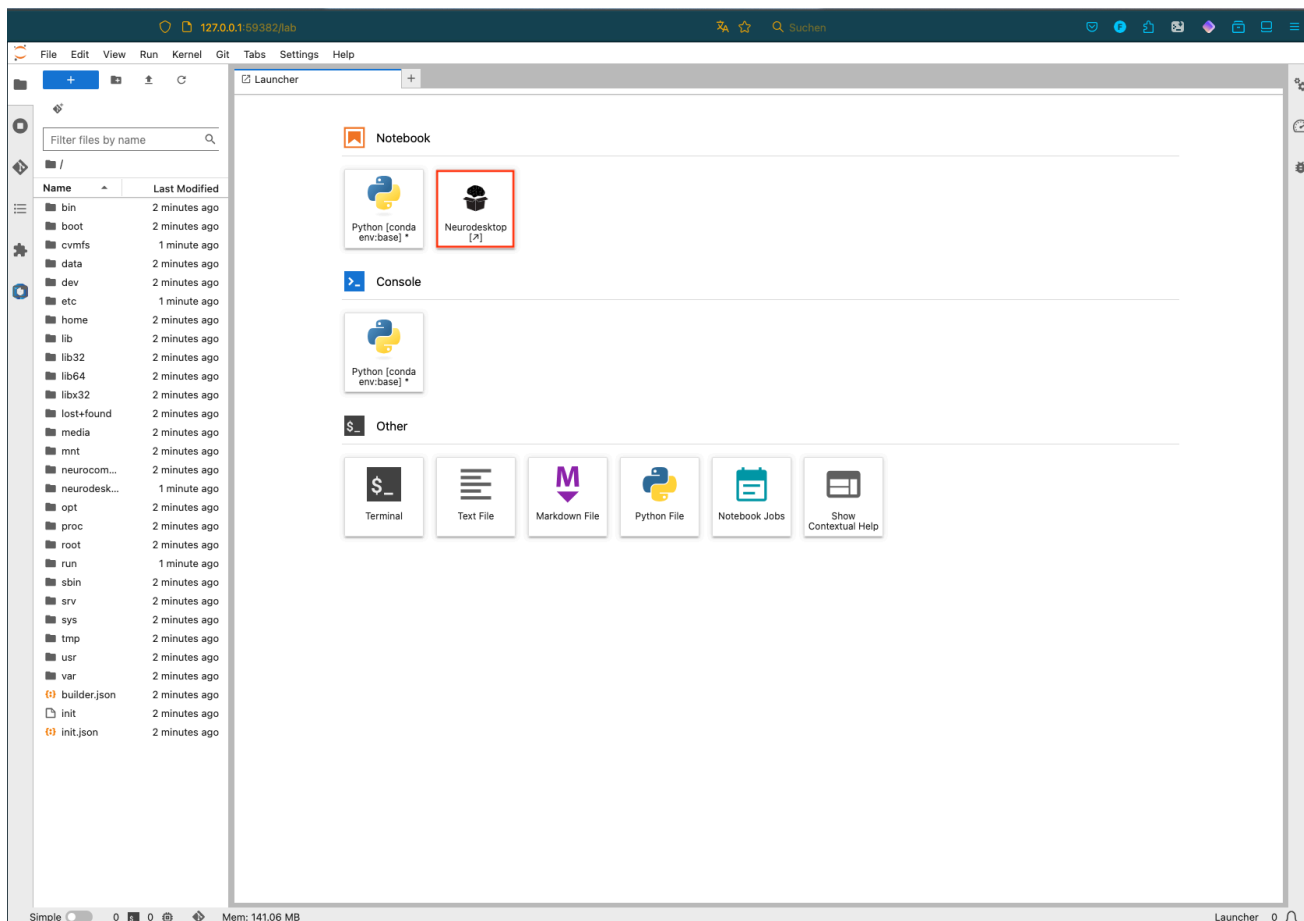
### How to Enter the Neurodesktop Interface

---


After setting up the connection and clicking the link, a JupyterLab page should open. Since this installation is specifically set up to run `Neurodesktop`, you will not be able to create new Jupyter notebooks here. To proceed, click on the `Neurodesktop [↗]` icon instead.



In case you can't see the `Neurodesktop [↗]` icon, please wait a few seconds and refresh the browser window.



This will lead to a new page in which you need to select the option **Desktop RDP**, which will take you to the actual Neurodesktop environment.

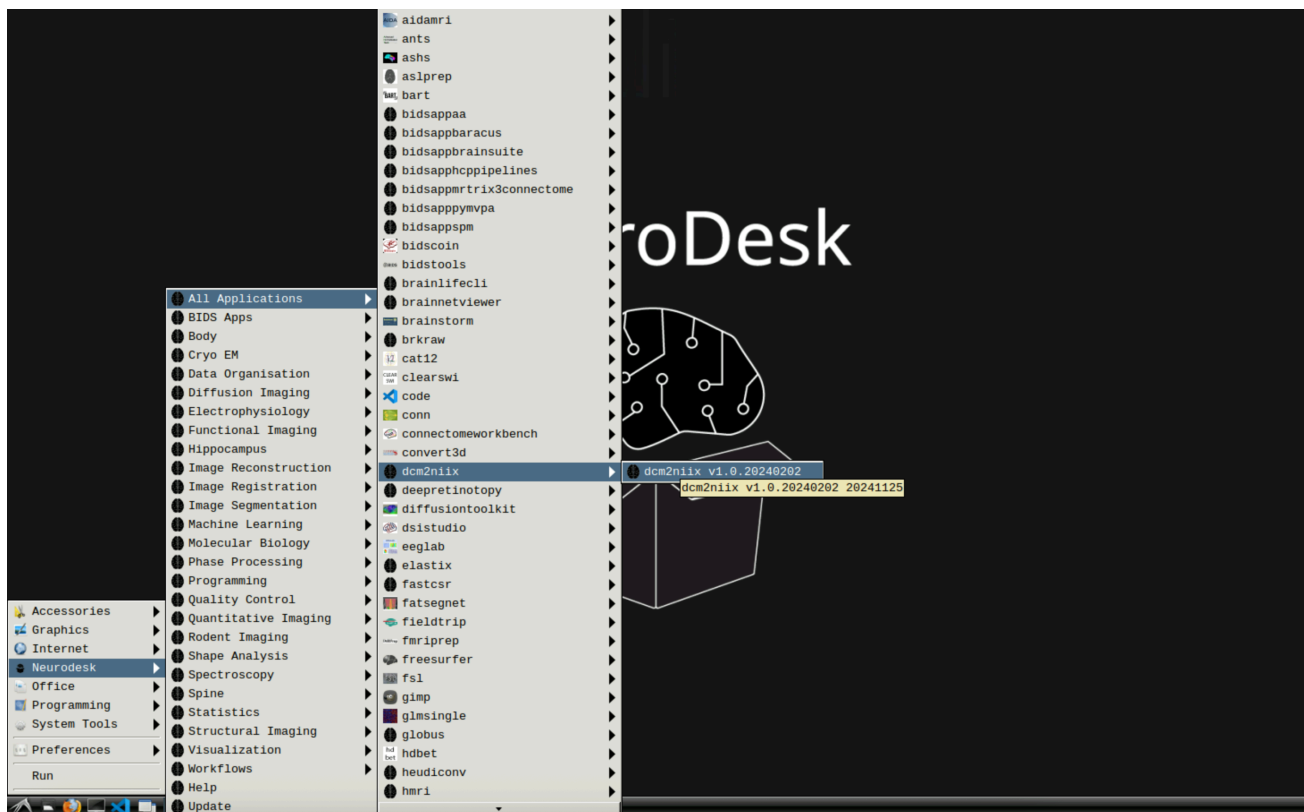
 **Desktop RDP** - changes resolution by resizing window and waiting for refresh

 **Desktop VNC** - changes resolution by running `lxrandr` on a terminal

## How to Start Applications from Within Neurodesktop

As Neurodesktop is a virtualised desktop environment, it should feel similar to using and navigating a Windows or Linux environment. Clicking on the button at the bottom right (similar to the Windows Start button) opens a menu where you can select the software you want to start.





Once you have selected the tool of your choice, a terminal will open where some settings will be set automatically and, if necessary, the software container will be extracted. If it is a CLI tool, the prompt will be released for use after initial configuration. If it is a GUI tool, however, the prompt will remain busy and the graphical user interface of the launched tool should appear after some time.



```
FILE EDIT INSUR MAP
Neuroimaging tools are accessible via the Neurodesktop Applications menu and running them through the menu will provide help and setup instructions. If you are
familiar with the tools and you want to combine multiple tools in one script, you can run "ml av" to see which tools are available and then use "ml <tool><vers
ion>" to load them.
[DEBUG] fetch_and_run.sh: Script name : /neurocommand/local/fetch_and_run.sh
[DEBUG] fetch_and_run.sh: Current working dir : /home/jovyan
[DEBUG] fetch_and_run.sh: Script location path (dir) : /neurocommand/local
[DEBUG] fetch_and_run.sh: SINGULARITY_BINDPATH :
[DEBUG] fetch_and_run.sh: SINGULARITY_BINDPATH is not set. Trying to set it
Reading config from /neurocommand/local/config.ini
CVMFS detected and Container seems to be available
Mounting containers from CVMFS directly.
[DEBUG] fetch_and_run.sh: Fetching containers done.
[DEBUG] fetch_and_run.sh: MOD_NAME: dcm2niix
[DEBUG] fetch_and_run.sh: MOD_VERS: v1.0.20240202
[DEBUG] fetch_and_run.sh: Module 'dcm2niix/v1.0.20240202' is installed. Use the command 'module load dcm2niix/v1.0.20240202' outside of this shell to use it.
[DEBUG] fetch_and_run.sh: looking for /cvmfs/neurodesk.arcd.edu.au/containers/dcm2niix_v1.0.20240202_20241125/dcm2niix_v1.0.20240202_20241125.simg
[DEBUG] fetch_and_run.sh: Module loading the container to set environment variables.
[DEBUG] fetch_and_run.sh: Attempting to launch container dcm2niix_v1.0.20240202_20241125

-----
## dcm2niix/v1.0.20240202' ##
dcm2niix is designed to convert neuroimaging data from the DICOM format to the NIFTI format. This web page hosts the developmental source code - a compiled vers
ion for Linux, MacOS, and Windows of the most recent stable release is included with MRICroGL. A full manual for this software is available in the form of a NIT
RC wiki.

Example:
...
./dcm2niix ./test-dicom-dir -b y
...

More documentation can be found here: (https://github.com/rordenlab/dcm2niix?tab=readme-ov-file)

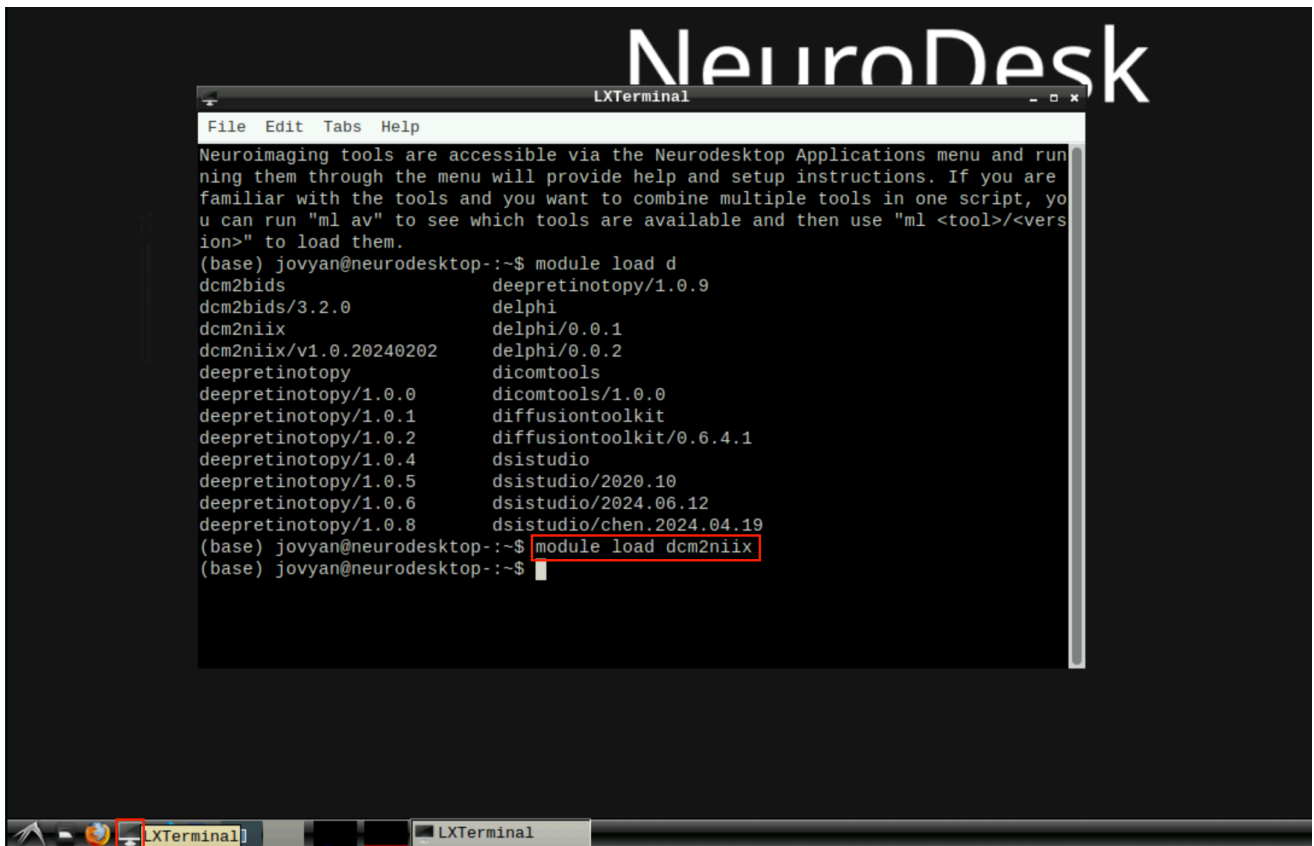
To make the executables and scripts inside this container transparently available in the command line of environments where Neurocommand is installed: ml dcm2ni
ix/v1.0.20240202

Citation:
...
Li X, Morgan PS, Ashburner J, Smith J, Rorden C (2016) The first step for neuroimaging data analysis: DICOM to NIFTI conversion. J Neurosci Methods. 264:47-56.
doi: 10.1016/j.jneumeth.2016.03.001. PMID: 26945974
...

License: This software is open source. The bulk of the code is covered by the BSD license. Some units are either public domain (nifti*.*, miniz.c) or use the MI
T license (jpeg.cpp). See the license.txt file for more details.
https://github.com/rordenlab/dcm2niix/blob/master/license.txt

-----
dcm2niix-v1.0.20240202:~$
```

In order to re-use the started application at a later time, you can make use of the module system. To do so, open a terminal in the Neurodesktop session, and use the `module load` command as shown below.



Please note that Neurodesktops software, and module system is entirely detached from the clusters module environment, as each Neurodesktop instance is entirely isolated from the HPC environment.

## Important Notes

### Data Exchange Directory

- ▶ The script creates a directory called `neurodesktop_share` in your home directory
- ▶ This directory serves as a bridge between the Neurodesktop environment and the cluster filesystem
- ▶ To copy data into this directory from within Neurodesktop, use the `sudo` command, e.g.:

```
1 | sudo cp /path/to/source /home/jovyan/share/
```

- ▶ If you want to make files from the cluster accessible in the Neurodesk environment, use the share folder as you normally would, e.g.:

1 | `cp /path/to/source $HOME/neurodesktop_share`

## Web Interface Access

---

- ▶ After opening the provided URL, you'll see the JupyterHub interface
- ▶ If the Neurodesktop icon doesn't appear immediately, wait a few seconds, and refresh the page (F5 on Windows/Linux, Command+R on macOS)
- ▶ When clicking on the Neurodesktop icon, you'll be presented with two options:
  - ▶ Desktop VNC
  - ▶ Desktop RDP
- ▶ **Important:** Select "Desktop RDP" as this is the only functioning option

## Resource Allocation

---

When running inside a Slurm job, the script automatically:

- ▶ Uses the allocated CPU cores and memory
- ▶ Sets up port forwarding to make the service accessible
- ▶ Creates a job-specific configuration file in your `neurodesktop_share` directory

## Troubleshooting

---

- ▶ If the JupyterHub interface doesn't show the Neurodesktop icon after the first start, refresh the page as Neurodesktop might still be initializing
- ▶ Ensure you're using the RDP option when connecting through the web interface
- ▶ Check that the SSH port forwarding command is running on your local machine

Powered by [Wiki.js](#)