

Executing Remote Applications with X11

Connection to a Linux Cluster with graphical output

Lichtenberg 2, TU Darmstadt:

```
clustername=lcluster<n>.hrz.tu-darmstadt.de  
<n>=13,15,17,19: System with GPU  
<n>=14,16,18: System without GPU
```

JustHPC, JLU Giessen:

```
clustername=justhpc.hpc.uni-giessen.de
```

Linux Cluster, Kassel University:

```
clustername=its-cs1.its.uni-kassel.de
```

MaRC3a Cluster, Philipps-University Marburg

```
clustername=marc3a.hrz.uni-marburg.de
```

Also needs -p 223!

Open an SSH connection to one of the login nodes:

- On Windows: Use MobaXTerm, or a current version of WSL 2, see below
- On Linux, MacOS and WSL 2 (on Mac, additionally install XQuartz, see below):

```
ssh -X -C [-p port] username@clustername
```

X11 forwarding Compression (speedup) Custom port (MaRC3 only)

Example for Lichtenberg 2 HPC Cluster:

```
ssh -X -C username@lcluster13.hrz.tu-darmstadt.de
```

Example for MaRC3a Cluster (custom SSH port):

```
ssh -X -C -p 223 username@marc3a.hrz.uni-marburg.de
```

Additional local requirements (e.g. connections only allowed from university intranet or VPN) still apply!

MobaXTerm

- Download page: <https://mobaxterm.mobatek.net/download-home-edition.html>
- Simplest way (no installation required): Choose the 'Portable Edition'
 - Unpack the downloaded zip file to a folder of choice
 - Double-click on the unpacked executable file to run
- The MobaXTerm website includes several demo videos that show how to use basic functions
- To connect, click on 'sessions' (top left), then 'SSH'.
- As 'Remote host', choose the right 'clustername' entry from above
- Check 'Specify username' box, then enter your cluster username
- MaRC3 only: Change 'Port' from '22' to '223'
- Click 'OK'
- You'll be prompted for your cluster account password
- You should now be able to open GUI programs on the cluster from your MobaXTerm session
- MobaXTerm also includes a file browser, so you do not need to additionally install e.g. FileZilla or WinSCP for that functionality

WSL 2

- The following should work with current Windows versions (Windows 10 Build 19044+ or Windows 11)
- Detailed information and step-by-step documentation is also available at <https://learn.microsoft.com/en-us/windows/wsl/tutorials/gui-apps>
- If WSL 2 is not yet installed, you can start the installation by opening a command prompt and typing

```
wsl --install
```

- This will take a while and require a reboot
- After rebooting, you will be prompted to enter a username and a password
 - This will be your local Linux account, it has nothing to do with your cluster credentials and can be freely chosen!
- After installation has finished, you can use it like linux commandline
 - Either issue linux commands from the windows command prompt by prefixing them with `wsl`, e.g.:

```
wsl ssh -XC <username>@lcluster13.hrz.tu-darmstadt.de
```

- Or open a Linux terminal window by searching and running

'Ubuntu' in the Windows start menu and using the linux commands,
e.g.:

```
ssh -XC <username>@lcluster13.hrz.tu-darmstadt.de
```

XQuartz

- Download page: <https://www.xquartz.org/>
- Download, install and run the program package that fits your macOS version
 - Everything else should work exactly as for Linux
- Alternatively, you can install XQuartz via homebrew:

```
brew install --cask xquartz
```

- The FAQ under <https://www.xquartz.org/FAQs.html> has lots of troubleshooting tips if something doesn't work