

Servers for bachelor students

The Bachelor servers will be incorporated into the [Virtual Desktop Cluster](#)

One by one the bach1, bach2 and bach3 servers will be added to the Virtual Desktop Cluster.

- bach3 will be migrated first, migration date 25 nov 2020.
- bach2 will be migrated second, migration date 27 nov 2020.
- bach1 will be migrated third, migration date 1 dec 2020.

Please start using vdesk.strw.leidenuniv.nl

Old access information

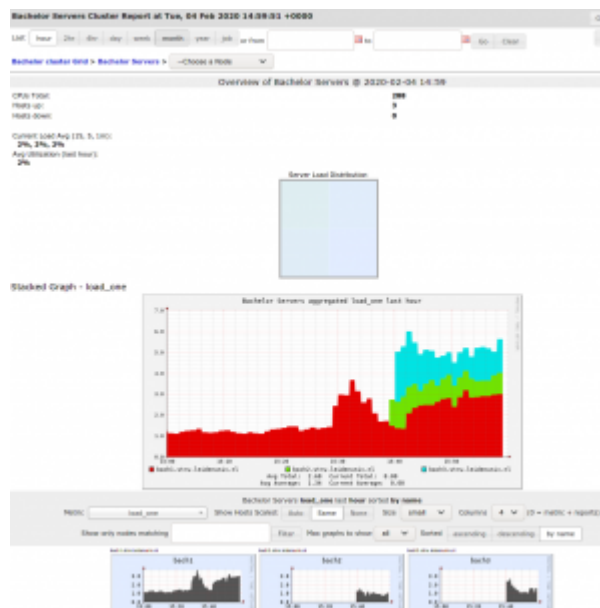
Leiden Observatory provides three powerful compute servers that bachelor students will use for their research projects. Each server has 96 CPU cores, 500 GB of internal memory, 50 TB of disk storage, and runs the Linux operating system RedHat version 7. Students are expected to use their own laptop or desktop computer (or possibly some other device) to connect to one of the servers via the internet. This page provides information on how to connect to the servers, and some basic information on how to use them. More general information about the computer environment and the software used by Leiden Observatory can be found [here](#). In particular, students are familiarise themselves with the [disk quota system](#) in place on the /home disk to avoid connection issues.



Note that students need a Leiden Observatory computer account (which is **not** the same as an ULCN account) to be able to connect to the servers.

Which bach(elor) server to connect to

The servers are located in a special server room, protected from overheating and power interruptions. User access to these machines is solely through an internet connection. The names of the three servers are:



- bach1.strw.leidenuniv.nl
- bach2.strw.leidenuniv.nl
- bach3.strw.leidenuniv.nl

As the servers are shared by many students, it is best to spread the load between the three servers. Before connecting (see below), one can look which machine has the most resources (free CPUs, memory, disk space) available by going to [this webpage](#) (log in with your STRW account). Note that there is a *fair use* policy in place, meaning that there are no hard limits on how much resources a single user or project can use. However, excessive use will be monitored, and limits may be put in place if excessive use leads to problems for other users. If large shares of resources are required for your project, please discuss this with your supervisor **and** the [Leiden Observatory IT department](#).

Connecting to the bach servers

The simplest way to connect to one of the bach servers is using ssh (also see [here](#)), e.g.:

```
ssh jklaassen@bach1.strw.leidenuniv.nl
```

followed by entering your password. The ssh command is available in the shell terminal / command prompt for most Windows, MacOS and Linux versions by default. An ssh connection to the bach servers can be set up both from inside and outside Leiden Observatory, and provides a text-based Linux shell from which commands can be issued. This does not allow for starting any programs with a graphical user interface (GUI). If your own computer is running X-server software (typically the case for MacOS and Linux), a connection which supports GUIs can be started as follows, e.g.:

```
ssh -XC jklaassen@bach1.strw.leidenuniv.nl
```

The C option enables compression which will greatly help interactive graphical work.

The major disadvantage of the approach given above is that any interruption in the connection between your computer and the server, either intentionally or unintentionally, will permanently break

the communication between your computer and the software that you started up. Below we describe two solutions to this problem, first a text-based solution using `screen` and then a GUI-based solution using `xrdp`.

Using a text-based screen session

If your work on the bach servers only involves text-based input and output, it is sufficient to run a simple `screen` session on the server to which it is possible to reconnect. After you connect with `ssh` to your bach server of choice, run

```
screen -S <session_name>
```

where `<session_name>` is a unique name of your choice (no spaces). This will start a new shell which is contained within the `screen` session. To detach from the `screen` session without killing it, type `<Ctrl>+a`, then `d`. To re-attach to the session at a later time (even after stopping and restarting the `ssh` connection), type

```
screen -r <session_name>
```

where `<session_name>` is the name used before. Of course, this only works when you `ssh` into the same server; e.g., you cannot reconnect to a `screen` session running on `bach3` when `ssh`-ing into `bach1`. This allows for students to continue working on their projects when and where they can, without the hassle of having to close down and restart the required software. This also enables running long computations without the need to having to keep the personal laptop or computer open. Students should keep in mind that disconnected but active sessions still use up some server resources. Note that you can start multiple `screen` sessions by using different session names. If you forgot the session names of your `screen` sessions, type:

```
screen -ls
```

This returns a list of running `screen` sessions, where each session name is given right after each number string plus dot `..` A `screen` session can be permanently closed from within by typing `exit` (which terminates the shell that was started when the `screen` session started).

Using a graphics-based RDP session

If your work on the bach servers involves GUI-based input and output, you can set up an RDP session (Remote Desktop Protocol by Microsoft). All three bach servers are running `xrdp`, which is an open source RDP service. You can connect to your server of choice using an RDP client program. Depending on the operating system (OS) on the computer you are using to connect, the RDP client will look and work slightly differently. Below there is some guidance for the most common cases (Windows 10, MacOS, Linux). It is important to know that direct RDP connections from outside the Leiden Observatory Linux network are not allowed for security reasons. For this reason we apply a technique called *ssh tunneling* to allow for RDP communication between your computer and the bach servers. The descriptions below per OS explain how to connect via RDP using `ssh` tunneling.

When a student connects to a particular server using RDP for the first time, it will start a new desktop session unique for each user (the default lightweight desktop is called MATE). Similar to `screen`, this

session can stay active, even when the student disconnects from the session. When the student connects again to the same server, this will reconnect to the same session. Students should keep in mind that disconnected but active sessions still use up some server resources.

Note that it is (in principle) possible to connect to a single session from multiple computers, hence allowing for remote collaborative work. However, **do not share your account information (username, password) to give others access to your session!** If multiple connections to a single session are required for your project, please discuss this with your supervisor **and** the [Leiden Observatory IT department](#).

Below is some guidance on how to connect to a server from different operating systems.

[Connecting from Windows](#)

[Connecting from MacOS](#)

[Connecting from Linux](#)

Disconnecting or finishing a session

To keep a session running on the server, just close the window of the RDP client program running on your computer, or disconnect in your xrdp client window. Next time you connect to the same server, you will re-enter the running session (making sure that the display size is the same). Again, please keep in mind that disconnected but active sessions still use up some server resources.

If you are done with a session, you can Log out in your session on the server (e.g., under System menu in MATE). This will close all open windows and terminate the session. Next time you connect to the same server, a new session will be started.

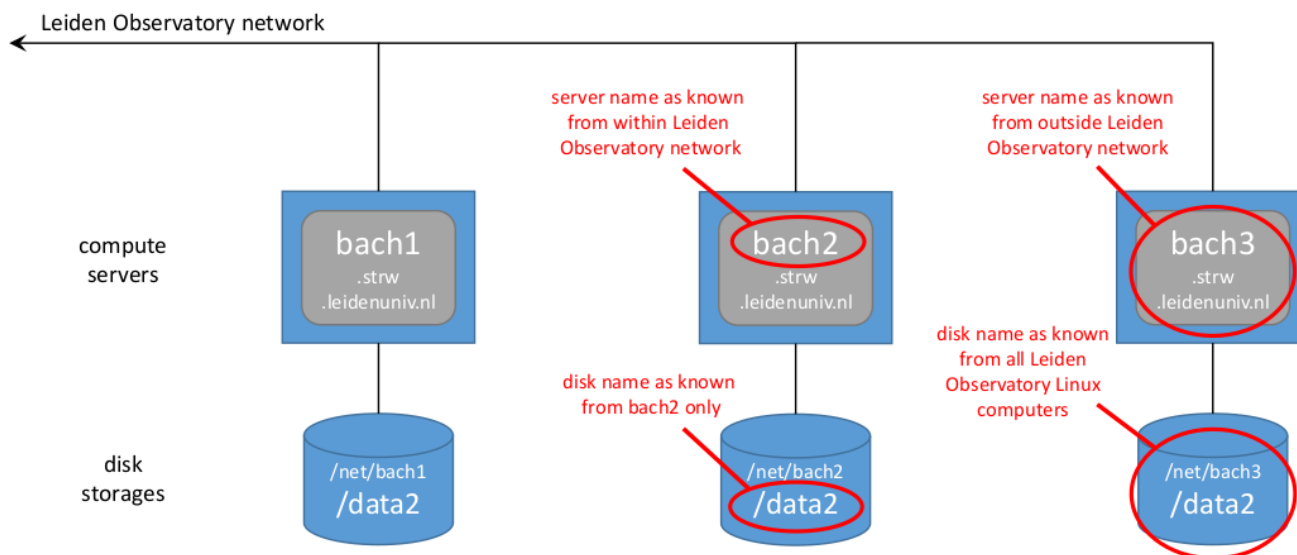
Storing and accessing your project data

Project data can be stored on one of the three 50 TB disk storages connected to the bach servers (see schematic drawing below). **It is important that you avoid storing project data in your /home directory as the amount of storage you have on the /home disk is limited.** If you exceed your quota on the /home disk, you will be unable to connect to the Sterrewacht computers. From any of the bach servers (or any Linux computer on the Leiden Observatory network), you can access any of the three storages connected to the bach servers via the /net/<server>/data2 path, where <server> is the name of the server that the storage is connected to (bach1, bach2, or bach3, without the .strw.leidenuniv.nl part). For example, you can have a session running on bach1 and store/access your data on the storage connected to bach3 using the /net/bach3/data2 path. To keep the data from many users organized, please open a terminal and create a sub-directory with your Leiden Observatory username to store your data in. For example:

```
mkdir /net/bach3/data2/jklaassen
cd /net/bach3/data2/jklaassen
... etc ...
```

Please note that the data storages connected to the bach servers are not meant for long-term

storage. In principle, all data areas will be cleared before next year's bachelor students start. At the end of the project, the students and their supervisors are responsible for copying away any data that needs to be stored long-term, or discuss possible data retention options with the [Leiden Observatory IT department](#). Also, please refrain from storing lots of personal data (vacation pictures, cat videos, etc.) on these storages.



From:

<https://helpdesk.strw.leidenuniv.nl/wiki/> - **Computer Documentation Wiki**

Permanent link:

<https://helpdesk.strw.leidenuniv.nl/wiki/doku.php?id=manuals:bachelorservers&rev=1605902384>

Last update: **2020/11/20 19:59**

