How to run . NET 8 MaxQuant on Linux

System Requirements

Since version 2.6.3.0, MaxQuant runs on .NET 8. Please make sure that you have installed .NET 8.0 (best is 8.0.302 or newer)!!!

To find out whether you already have it, type in the command line:

```
dotnet --version
```

If you see the version over 8.0, then everything is ready to start MaxQuant.

Otherwise please follow the instructions below to install .NET 8.

Install .NET 8 on Linux

We recommend two ways to install .NET 8 on linux, **you can choose one of them but please don't do both**:

(1) Install .NET 8 by running automated_installation.sh:

```
source ./automated_installation.sh
```

The shell script will:

(a) Ask the user for an installation path as a required argument
(b) Download the official installation script from Microsoft
(c) Run this script to install dotnet 8.0.302 in the supplied installation path
(d) Modify the .bashrc file to permanently add the path to it
(e) Soruce .bashrc so the changes are in effect immediately

(2) Install .NET8 manually:

Go to the local path (/home/user/local_path, please replace it with your own path!) that you would like to install dotnet and download the zipped file Linux binaries of .Net8 by running the following command:

```
wget https://download.visualstudio.microsoft.com/download/pr/dd6ee0c0-6287-4fca-85d0-1023fc52444b/874148c23613c594fc8f711fc0330298/dotnet-sdk-8.0.302-linux-x64.tar.gz
```

Run the following command to extract the SDK

```
mkdir -p /home/user/local_path/dotnet && tar zxf dotnet-sdk-8.0.302-linux-x64.tar.gz -C /home/user/local_path/dotnet
```

Then the SDK will be available under the path /home/user/local_path/dotnet. You can add the path

```
export PATH=/home/user/local_path/dotnet:$PATH
```

After you've installed the .NET 8 successfully, when you run

```
dotnet --version
```

you should see

```
8.0.302
```

Let’s start!

Display the help screen

Before you start your very first MaxQuant run on Linux, we recommend you to display the help screen and get an overview of the functions and corresponding commands.

```
dotnet MaxQuant_version/bin/MaxQuantCmd.dll --help
```

After you type in the command line in above, you should see something see the help screen like this:

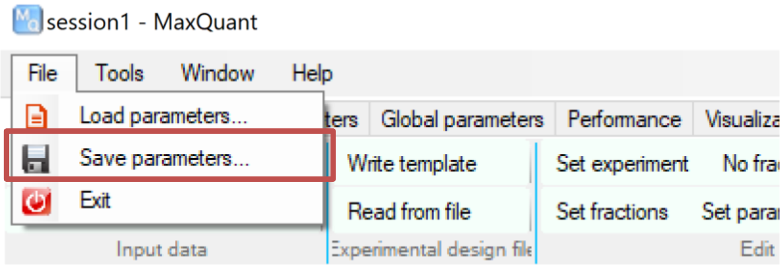
```
MaxQuantCmd version

Copyright © Max-Planck-Institute of Biochemistry 2021

USAGE:

Complete run of an existing mqpar.xml file:

  MaxQuantCmd.dll mqpar.xml

Print job ids/names table:

  MaxQuantCmd.dll mqpar.xml --dryrun

Rerunning second and third parts of the analysis:

  MaxQuantCmd.dll mqpar.xml --partial-processing-end 3 --partial-processing 1

Changing folder location for fasta files and raw files for a given mqpar file:

  MaxQuantCmd.dll mqpar.xml --changeFolder "<new mqpar.xml>" "<new folder with fasta
files>" "<new folder with raw files>" "<new folder with libraries (only for DIA runs
using predicted libraries)>"



  -p, --partial-processing        (Default: 1) Start processing from the specified job
id. Can be used to continue/redo

                                  parts of the processing. Job id's can be obtained in
the MaxQuant GUI partial

                                  processing view or from --dryrun option. The first
job id is 1.
```

```
  -e, --partial-processing-end      (Default: 2147483647) Finish processing at the
specified job id. 1-based indexing is

                                  used.



  -n, --dryrun                    Print job ids and job names table.



  -c, --create                    Create a template of MaxQuant parameter file.



  -f, --changeFolder                 Change folder location for fasta and raw files
(presuming all raw files are in the

                                  same folder). Expecting three or four locations
separated by space. 1) path to the

                                  mqpar file 2) path to the fasta file(s) 3) path to
the raw files. 4) path to the

                                  library files (only for DIA runs)



  --help                          Display this help screen.



  --version                       Display version information.



  mqpar (pos. 0)                  Required. Path to the mqpar.xml file. If you do not
have an mqpar.xml, you can

                                  generate one using the MaxQuant GUI or use --create
option.
```

Create the mqpar.xml file

If you have never used MaxQuant, you may have noticed that each MaxQuant run creates a mqpar.xml file in the folder where the raw files are stored. The mqpar.xml file is the MaxQuant parameter file that stores all the parameters for the database search (such as: raw files, fasta files, enzymes, threads, etc.). To run MaxQuant on Linux, you need to call the mqpar.xml file (of course you can rename it).

Currently we highly recommend to preconfigure the mqpar.xml file in MaxQuant GUI on Windows. Just specify all the parameters and click "File-Save parameters", the mqpar file will be saved.



After transferring the mqpar file on Linux machine, you'll need to modify the path of the raw files and fasta files coming from the Windows machine. You can use this command to change them, given all raw files are in the same folder:

```
dotnet  MaxQuant_version/bin/MaxQuantCmd.dll  "<mqpar.xml>"  --changeFolder  "<new mqpar.xml>" "<new folder with fasta files>" "<new folder with raw files>" "<new folder with libraries (only for DIA runs using predicted libraries)>"
```

Then you should see a message:

```
ChangeFolder: The new MaxQuant parameter file 'path_to_new_mqpar_file' is created
```

As you might have a very powerful Linux machine, you can increase the number of threads by modify `<numThreads> <numThreads>` in your new mqpar.xml file accordingly.

For experienced users, who are very familiar with the layout of mqpar.xml file, you can also create a template mqpar file with the command below and fill in all the information on Linux.

```
dotnet MaxQuant_version/bin/MaxQuantCmd.dll --create template_mqpar_file.xml
```

The template mqpar.xml file already has already has most of the parameters set to default values, you must at least specify the raw files and fasta files by replacing the example.fasta and file.example.RAW as shown below:

```
<fastaFilePath>example.fasta</fastaFilePath>
```

```
<filePaths>
      <string>file.example.RAW</string>
</filePaths>
```

Run MaxQuant

To run MaxQuant on Linux, simply use the command below to call the mqpar file:

```
dotnet MaxQuant_version/bin/MaxQuantCmd.dll mqpar.xml
```

However, MaxQuant run will stop as soon as you close the terminal. Therefore, we highly recommend you to run MaxQuant in the background, you can use the command below:

```
nohup dotnet MaxQuant_version/bin/MaxQuantCmd.dll mqpar.xml > nohup.out &
```

So you can close the terminal without interrupting the MaxQuant run and the progress will be written to the nohup.out file.

Another possibility is to run the command in a screen. More about the screen command you can find here