Using Linux Software on Windows 10

David Dubbeldam[1,†] *, Sofía Calero[2], Thijs J.H. Vlugt[3] and Randall Q. Snurr[4]
[1] Van 't Hoff Institute of Molecular Sciences, University of Amsterdam, The Netherlands
[2] Department of Physical, Chemical and Natural Systems,
University Pablo de Olavide, Sevilla, Spain
[3] Delft University of Technology, Process & Energy Department, Delft, The Netherlands,
[4] Department of Chemical & Biological Engineering, Northwestern University, Evanston, USA

January 2, 2019

1 Requirements

- Minimum windows 10 version is the Anniversary Update (Version 1607),
- 64-bit version of Windows 10

2 Turn on *Developer Mode*

- Go to "Start menu"
- Goto "Settings"
- Search "for developer settings"
- Turn on "Developer Mode"

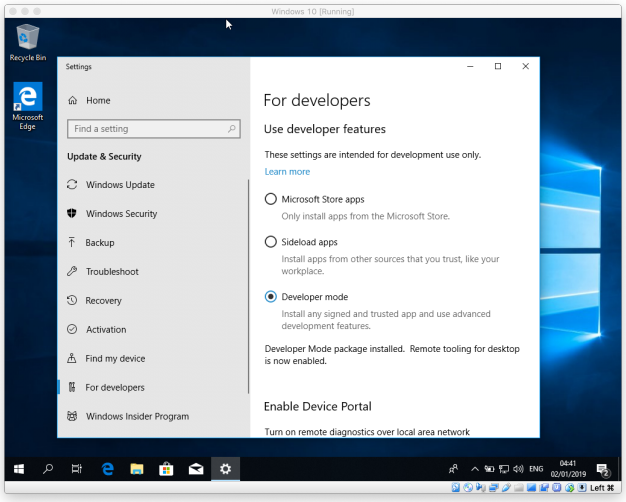
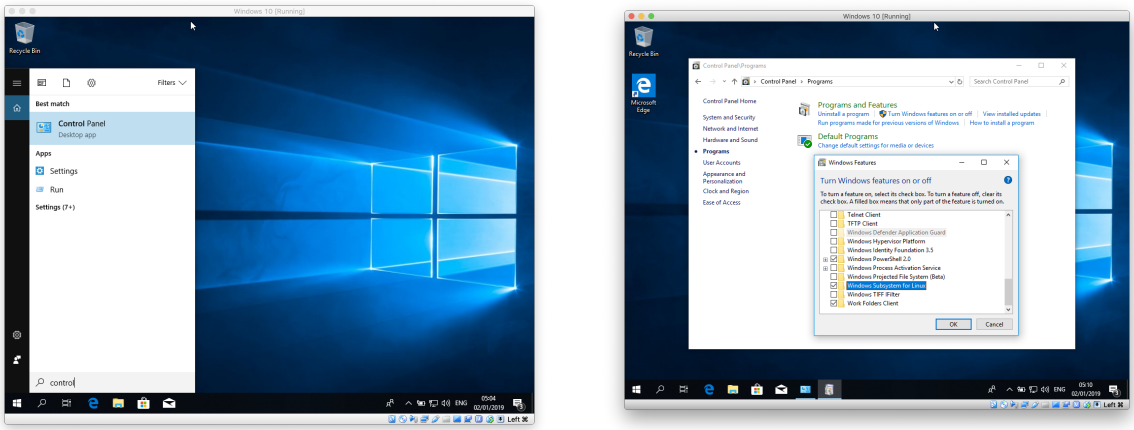


Figure 1: Turn on *Developer Mode*.

*† Email: D.Dubbeldam@uva.nl

3 Install *Windows Subsystem for Linux*

- Go to "Control Panel"
- Goto "Programs" and "Program and Features"
- Goto "Turn Windows features on or off"
- Select "Windows Subsystem for Linux"

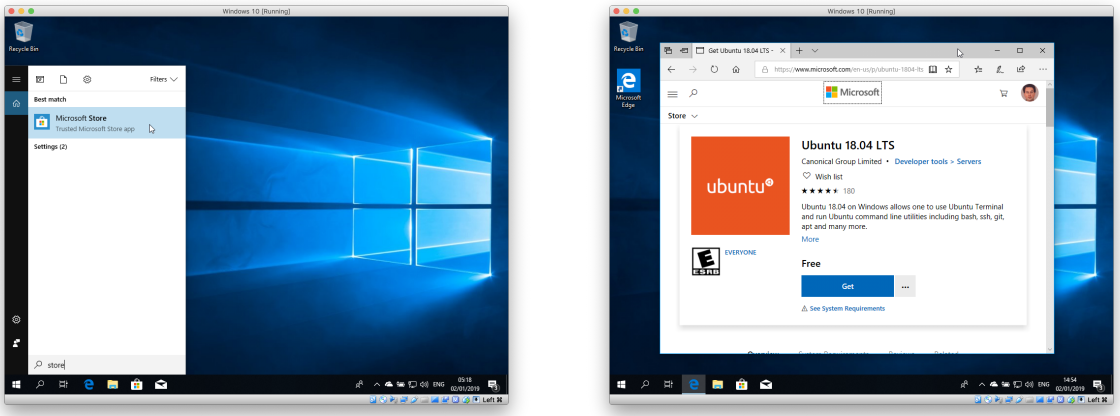


(a) (b)

Figure 2: Install Windows Subsystem for Linux: (a) start Control Panel, (b) select "Windows Subsystem for Linux"

4 Download *Ubuntu 18* from the Microsoft Store

- Go to "Microsoft Store"
- Search "Ubuntu 18"
- Install "Ubuntu 18.04 LTS"



(a) (b)

Figure 3: Install linux for windows: (a) start "Microsoft Store", (b) install "Ubuntu 18.04 LTS"

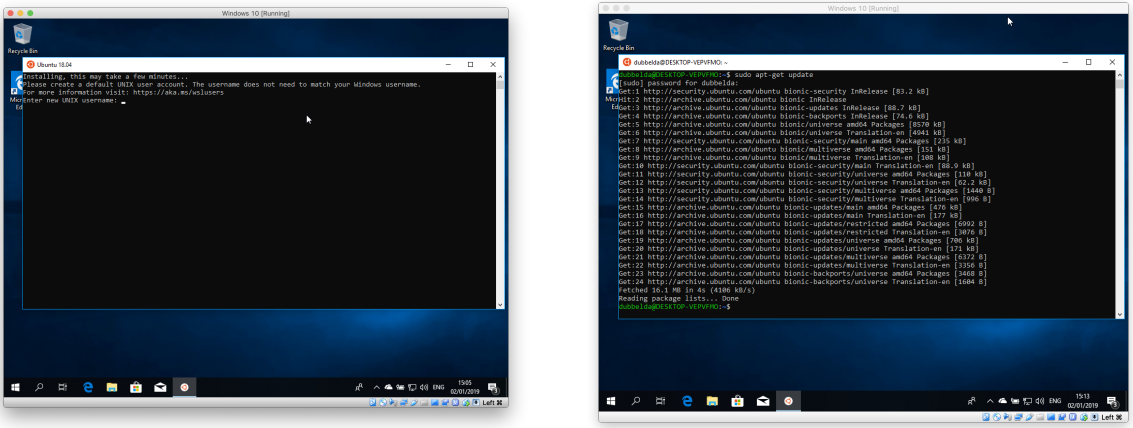
Alternatively, download the AppX-package from

https://docs.microsoft.com/en-us/windows/wsl/install-manual

5 Create a linux account

Start the "Ubuntu" app (i.e. the bash terminal). First, a linux account need to be created. This account does *not* have to be the same as your Windows account.

- Enter a username in the required field and press Enter (you can not use the username "admin")
- Enter a password



(a) (b)

Figure 4: Setup (a) the default linux user account (you can't use the username "admin"), (b) the linux system.

6 Install required linux packages

- sudo apt-get update
- sudo apt-get upgrade
- sudo apt-get install build-essential
- sudo apt-get install gfortran
- sudo apt-get install gnuplot

7 Where to put your files?

To use your files from both windows and linux, you should use your windows files in your windows account accessible from linux at `\mnt\c\Users`. Any change made from the linux side will immediately show up in windows. You can then use installed window software to read and visualize output files.

When you access files on your Windows filesystem from within Bash, it will honors the NT filesystem behaviors (e.g. case-insensitivity), permissions, etc. so you can easily access the same files using both Windows tools and Bash tools without having to copy files back and forth between filesystems.

If you want to access a system folder your user account does not have permission to access, you wouldd need to right-click the Bash shell shortcut and select "Run as Administrator" to launch the Bash shell with Windows Administrator privileges.

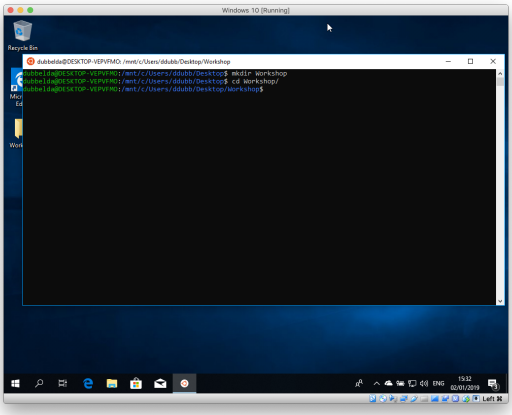


Figure 5: Your windows files are accesible from `\mnt\c\`.

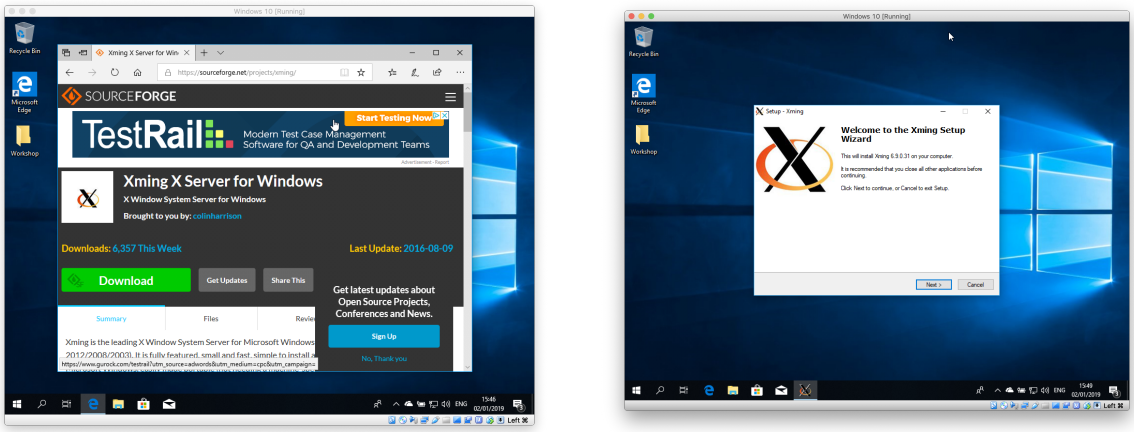
8 Run graphical programs in linux

Installing an X server will allow graphical linux applications to appear on your Windows desktop. The Xming X Server for Windows works well for this, although there are other X servers for Windows and they should also work.

Download the Xming X-server from

https://sourceforge.net/projects/xming

and install Xming using the default settings. Launch Xming and it will appear in your system tray, running in the background and waiting for you to launch a graphical Linux program.

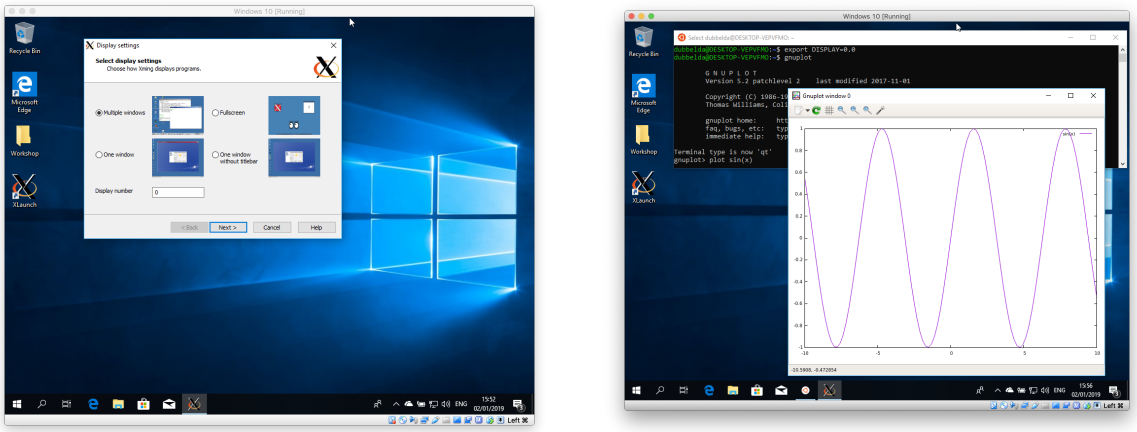


(a) (b)

Figure 6: Download the Xming X-server and install it.

Once you've installed the program, you can specify the X server and then launch the application. For example, to launch gnuplot, run the below commands. (To launch another graphical application, just specify its name instead of "gnuplot.)"

```
export DISPLAY=:0
gnuplot
```



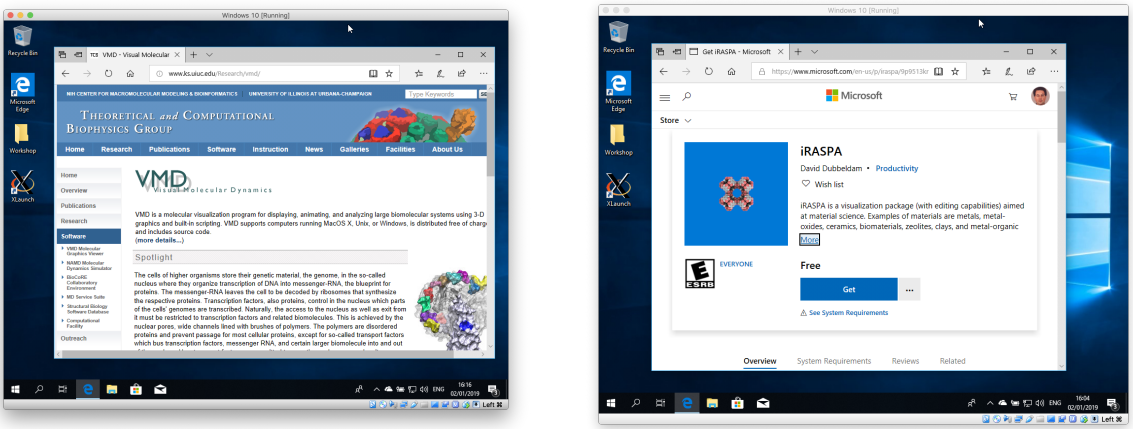
(a) (b)

Figure 7: Using gnuplot in linux, (a) start Xming, (b) connect to the X-server display `export DISPLAY=:0` and start gnuplot.

9 Use molecular viewers in windows

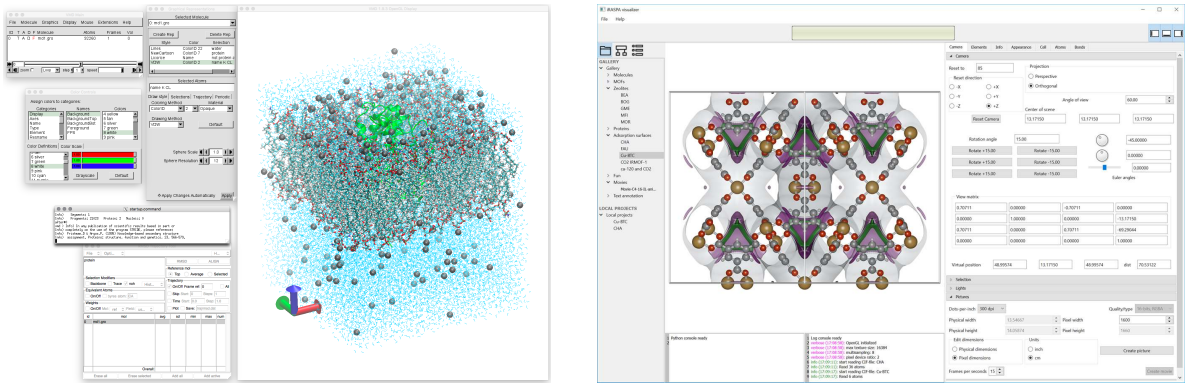
VMD is designed for viewing and analyzing molecular dynamics data of biological systems such as proteins, nucleic acids, lipid bilayer assemblies, etc. It also includes tools for working with volumetric data and sequence data. The functionality can be easily extended using python and Tcl scripts as VMD includes embedded Tcl and Python interpreters. Figure 9a shows a screen shot of VMD while visualizing an MD simulation of a protein embedded in a membrane. Download VMD from the following link to install it: `https://www.ks.uiuc.edu/Research/vmd/`.

iRASPA is a visualization package (with editing capabilities) aimed at material science. Figure 9b shows a screenshot of the program. iRASPA supports crystallographic operations like space group detection and finding the primitive cell, and extensively utilizes GPU computing. For example, void-fractions and surface areas can be computed in a fraction of a second for small/medium structures and in a few seconds for very large unit cells. It can handle large structures (hundreds of thousands of atoms), including ambient occlusion, with high frame rates. iRASPA is available from the Windows 10 store. It requires OpenGL and OpenCL drivers to be installed for your video card.



(a) (b)

Figure 8: Download visualization packages (a) VMD from `https://www.ks.uiuc.edu/Research/vmd/`, or (b) iRASPA from the Windows 10 Store.



(a) (b)

Figure 9: Windows 10 screenshots of (a) VMD showing a protein embedded in a membrane with water and ions at both sides, (b) iRASPA showing the primitive unit cell of a CHA-type zeolite with three adsorption surfaces showing the shape of the cavity, the diffusion paths, and the adsorption sites.

10 Useful additional linux packages

```
sudo apt-get install build-essential gfortran git bzr
sudo apt-get install gnuplot xfig emacs vim
sudo apt-get install vtk6 libvtk6.3 libvtk6-dev vtk6-examples vtk6-doc cmake
sudo apt-get install mesa-common-dev xorg-dev libglu1-mesa-dev freeglut3-dev mesa-utils
sudo apt-get install opencl-dev clinfo
sudo apt-get install qt5-default qtcreator qt5*-dev
sudo apt-get install texmaker grace gimp automake curl chemical-mime-data

sudo apt-get install openbabel avogadro jmol cp2k pymol
sudo apt-get install tcsh vim  speedcrunch ftnchek
```