

Installing PLATON on a Mac and getting Olex² to use it

Anthony Linden, Department of Chemistry, University of Zurich, Switzerland, anthony.linden@chem.uzh.ch

Here is how to get PLATON running on a Mac. PLATON uses X11 for its GUI, so you need the X11 libraries and tools and you then need a Fortran compiler. Ton Spek does not create compiled versions of PLATON with static libraries, hence you must compile it yourself and the need for all this.

The instructions below should work with Mac OS 12 (Monterey), 13 (Ventura) and 14 (Sonoma) on both Intel and M1/M2 chipset Macs and has also worked on earlier OS versions. The procedure sometimes changes a little for each new Mac OS version, mainly because X11 libraries sometimes change location and a PLATON executable compiled under one OS does not always work under another. Once you get this set up for one OS, it works from then on at least until the next OS is installed, so you usually only need to do the actual compile step when updating PLATON. It is advisable to update and recompile PLATON regularly, because it is constantly being improved.

If you are familiar with Unix/Linux, the following is probably clear to you. If not, please simply try to follow the instructions below exactly. Please read and type carefully, as there are important spaces, dots and dashes within some commands. All line commands typed are case sensitive. Unix commands usually do not give any feedback when they work correctly. Messages often mean something might not have worked.

This looks long, but is quite quick if all goes well. Don't be discouraged! If you run into trouble, please reach out.

See also Ton Spek's readme at <https://www.platonsoft.nl/xraysoft/Mac-OSX/platon>. Note that it is unnecessary to install the full Xcode developer kit, just Command Line Tools (see below).

Note: In early 2024, some people experienced difficulties with compilation because of errors during the final link (ld) stage concerning missing library definitions, or similar. Apparently the installed version of Command Line Tools had become incompatible with the MacOS after an upgrade or update and the CLT needs to be deleted and reinstalled. See step 3a below.

If needed, there is a stripped Fortran only version of PLATON (without graphics) available (special) that usually compiles without problems. It allows for easy validation of jobs such as 'platon -U name.cif'.

1 – Although everything from here on can be done in a Terminal window, I like using XQuartz. Download the latest XQuartz (2.8.5 or later) from <https://xquartz.macosforge.org>

2 - Install XQuartz and start it (located in Applications/Utilities), or open Terminal. When the window opens, you are in the top level directory (folder) for your Mac user account. To return here at any time, type 'cd'. To find out which folder you are in at any time, type 'pwd'.

3 - Install Apple Command Line Tools; type in the XQuartz window: `xcode-select --install`

3a – If you want to update an existing version of Command Line Tools, type in the XQuartz window:
`cd /Library/Developer/`
`sudo rm -rf CommandLineTools`
 (enter your password, then give it a little time)
`xcode-select --install`

4 - Download gcc/gfortran

Here I use the download from <https://hpc.sourceforge.net>, but there are alternatives at <https://github.com/fxcoudert/gfortran-for-macOS/releases> or by installing Homebrew from <https://brew.sh>, then enter the command 'brew install gcc'.

For <https://hpc.sourceforge.net>:

The binaries are near the top of the page; be sure to choose the version (X.Y) appropriate for your Mac OS version and chipset: *i.e.* gcc-X.Y-bin.tar.gz (contains gcc and gfortran). It is recommended you use Safari for the download to avoid the tar file being completely unpacked, after which the instructions below will not work (Firefox unfortunately unpacks the downloads).

5 - Move the file to your Desktop for convenience. Your Mac might automatically uncompress it, so the .gz is removed, but you still must have a single file with the .tar or .tar.gz ending for the next step.

6 - In the XQuartz window, type the following lines. Uppercase, spaces and dots are part of the command and must be typed exactly as given; note that '-C /' has a space before the /):

```
cd
cd Desktop
sudo tar -xvf gcc-X.Y-bin.tar -C / (or sudo tar -xvzf gcc-X.Y-bin.tar.gz -C / if you still have the .gz file)
sudo asks for your login password the first time, assuming you have administrator rights. Otherwise the password for the admin account on your Mac.
```

7 - Create a folder named 'platon' at a convenient location. I use Desktop for now. Note that if you use uppercase letters for any part of the folder name, then you must type it exactly the same way from now on.

8 - Download the files: platon.f.gz, xdrv.c.gz, sucrose.spf from:
<https://www.platonsoft.nl/xraysoft/unix/platon> and move them into that folder.

9 - In XQuartz, go to that folder. If already at Desktop from the above, then just type 'cd platon'. If lost, type:

```
cd
cd Desktop/platon
```

10 - gunzip the files platon.f.gz and xdrv.c.gz (the Mac probably has done that automatically). If not:

```
gunzip platon.f.gz
gunzip xdrv.c.gz
```

11 - Compile PLATON. This is the most critical and often troublesome step. Type carefully and exactly:

```
sudo gfortran -o platon platon.f xdrv.c -I/opt/X11/include -L/opt/X11/lib -lX11
```

Note that in the above, the character after the dash in `-I/opt` is an uppercase i and the character after the dash in `-lX11` is a lowercase L.

If nothing appears to be happening, just wait. The compiler takes a minute or two. If the command prompt reappears after a while without any messages, you have succeeded!

If this step fails with error messages after the long wait, either you did not type the above correctly, or the X11 libraries are not where they are expected to be, which might be the case if you are using an older Mac OS. Or see the earlier comment above about Command Line Tools.

If you have an earlier version of the Mac OS, e.g. Yosemite or Mavericks, one of the following might work instead (these will certainly not work if you use El Capitan or later):

```
gfortran -o platon platon.f xdrv.c -L/opt/X11/lib -lX11
gfortran -o platon platon.f xdrv.c -L/usr/X11/lib -lX11
```

If this also fails, try to find the path to the /X11/lib folders and use that with the `-L` switch.

12 - Copy the newly created file 'platon' (the executable) to /usr/local/bin or any other location you like your executables, such as shelxl, etc., to be. I recommend not putting these in the /Applications folder. If you do not have access to the folder /usr/local/bin prefix the lines below with 'sudo'. Alternatively, use any convenient folder within your user account and use the path to it instead of /usr/local/bin in the following.

In the following, the dot at the end is important.

```
cp platon /usr/local/bin/.
chmod -R 755 /usr/local/bin/platon
```

Quit XQuartz and start it again so the system learns about the new program (or type 'hash -r' if in the default bash shell, 'rehash' if in tsch or csh). Only needed the first ever time you install PLATON.

13 - Test: type 'platon sucrose.spf' in the XQuartz window (you have to be in the platon folder again if you moved away), click on ORTEP in the main PLATON menu for an ORTEP plot.

If all is well, you have a working version of PLATON that you can now use quite independently of any other program simply by using the command line in XQuartz or a Terminal window, such as 'platon x.cif'.

Similarly, if the shelx programs are on a known path, you can run them manually from there too. All you need to do is 'cd' to the folder with your current structure files in them.

To update PLATON at any time, repeat steps 8-12. The availability of a new version is indicated on the PLATON opening window (red text starting with WEB:) and you can download the new version by clicking on that information. The compilation and installation steps can be simplified by creating a makefile with the relevant instructions. See hints at <https://www.cs.colby.edu/maxwell/courses/tutorials/maketutor/>. A makefile for compiling PLATON is below. Remove the # from the first two lines if your Mac does not automatically unzip the downloads. <tab> means insert the tab character on those lines. Simply save this as a file called 'makefile' in the same place as the platon.f and xdrv.c files and run it in XQuartz with the command 'make' initially to compile and then, if no errors, 'make install' to move things to the right places.

```
platon:
#<tab>   gunzip -f platon.f.gz
#<tab>   gunzip -f xdrv.c.gz
<tab>    gfortran -o platon platon.f xdrv.c -I/usr/X11/include -L/opt/X11/lib -lX11

install:
<tab>    mv platon /usr/local/bin/.
<tab>    chmod 755 /usr/local/bin/platon
```

How to get Olex² to use PLATON on a Mac

To get Olex² to know that PLATON exists, and thus show the 'PTON' button at the top of the main GUI instruction panel, and to start PLATON, you need to do the steps below.

1 - Install PLATON as described above. Make a note of the full path to where you put the PLATON executable. For example, /usr/local/bin if you put it there. It is convenient if you also put the executables for SHELXT, SHELXL, SHREDCIF, etc. in the same location.

2 - Start Olex². In the main GUI instruction panel, select the HOME panel and then click on the SETTINGS tab. The second last item in the list that appears is "PATH:". Enter the full path to your PLATON executable here and then click out of the path definition box. Quit Olex² and restart it. Come back to the path definition box and ensure the path you typed is shown. If the box is empty try again.

3 - To test everything, run Olex². Select the sucrose demo structure on the opening screen. Go to the work tab then click on the down arrow beside the solve tab. The dropdown at the program line should show shelxt (and shelxs if installed). Click on the down arrow beside the refine tab and again you should see shelxl in the program dropdown. Olex²'s own routines are displayed by default, but it is better to use the shelx programs for these exercises. At the very top right of the Olex² screen, there should be a Pton button if Olex² knows that PLATON exists. If all this is OK, then you are ready to go!