## I cannot see my model in COOT after pandda.export or refinement!!!

If you expect to see a refined model, but the nothing shows up when you select the respective crystal in the COOT interface, or if COOT automatically jumps to the next crystal, then the first thing to do is to go into the respective sample directory to find out what went wrong. If you work in PanDDA refinement mode (which is the default), COOT expects two PDB files:

**refine.split.bound-state.pdb**

**refine.split.ground-state.pdb**

However, if the files are not present, COOT will immediately jump to the next crystal in the list until it finds a set of valid files. Be careful, the files are in fact symbolic links, which point to the files in the folder for the latest refinement cycle. XCE will create the links regardless if refinement was successful, i.e. in case refinement did not work the links will be broken. Type the following command in the sample directory to see if everything is OK:

**ls -lh --color=tty**

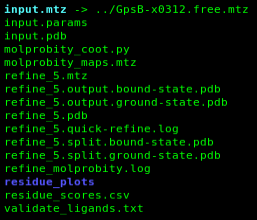
It should look like this:



However, the links are broken if it looks like this:



Something went wrong during refinement cycle 6 (Refine\_0006) in the example above. This folder is the first thing to check in order to find out what might have gone wrong. The folder should contain the following files and subfolders:



Start with checking the respective the quick-refine logfile in the respective folder (refine\_6.quick-refine.log). However, this may not always be informative, because the actual error might have happened earlier.

Hence, before we continue, we need to explain what actually happens under the hood when you press ‘REFINE’:

First, XCE determines what the last refinement cycle was and then it creates a new folder in **cootOut/Refine\_<current\_cycle>** accordingly. It then merges the current ground-state and bound-state model and saves the resulting model as **multi-state-model.pdb** into this folder. Next, XCE runs **giant.make\_restraints multi-state-model.pdb** in order to generate the required occupancy restraints. Afterwards, XCE creates a shell script (refmac.csh) which contains a list of instructions regarding refinement and validation and which is usually (at least if you process your data at Diamond) submitted to the cluster. Hence, the current refinement folder in cootOut is usually the place to start troubleshooting: check if all the files are there and look at the logfiles. Sometimes the logs can be quite clear, but they may also contain a long list of cryptic python errors. Although this may not look very useful, it at least gives us an idea where to start.

What happens next?

After the script is submitted to the cluster, **giant.quick\_refine** will create another **Refine\_<current\_cycle>** folder (but with *four* digits) and this folder contains another set of logfiles, as well as the final (split) PDB and validation files. At the very end of each refinement cycle, the script will create the above mentioned symbolic links. Again, please keep in mind that the links are created regardless if refinement was successful or not, which is why they can be broken, but might still give you the illusion that everything is OK. Hence, this folder is the next place to look for potential problems.

How to continue?

Now might be the time to post to xchembb or talk to your local contact. However, you can also try mending the problem yourself, or in case the problem is due to an XCE bug and a new version is available you need to manually reset the affected sample directories. There are different options available, but first, delete all the broken symbolic links and temporary files:

**rm -f refine.pdb refine.mtz refine\_molprobity.log refine.output.bound-state.pdb refine.output.ground-state.pdb refine.split.bound-state.pdb refine.split.ground-state.pdb validate\_ligands.txt validation\_summary.txt**

One option is to revert back to the last successful refinement cycle and then try again. This is the way forward in case a new XCE version became available that addressed a specific bug. For example, let’s assume that something went wrong in refinement cycle 6. After you have removed the links, go back to cycle 5 (or any other successful cycle) by recreating the links:

**ln -s Refine\_0005/refine\_5.split.bound-state.pdb refine.split.bound-state.pdb**

**ln -s Refine\_0005/refine\_5.split.ground-state.pdb refine.split.ground-state.pdb**

**ln -s Refine\_0005/refine\_5.pdb refine.pdb**

**ln -s Refine\_0005/refine\_5.mtz refine.mtz**

You will now see the models again in COOT and will be able to launch refinement.

Another possibility is to remove the links *and* the failed refinement folder (Refine\_0006 in our example). Then change into **cootOut/Refine\_6.** You can try modifying the refmac.csh file and run it locally or on the cluster and see what happens. Or, launch the **giant.quick\_refine** command as present in the script from the command line. The latter option gives you immediate feedback, which can be very helpful for trouble-shooting.