XChemExplorer

Manual

Current version

Date

18/06/2018

Author

Tobias Krojer

tobias.krojer@sgc.ox.ac.uk

Table of Contents

Introduction

XChemExplorer (XCE) is a data management and workflow tool which supports large scale analysis of protein-ligand structures by X-ray crystallography. It is not an actual algorithm, but serves as a launch pad for batch submission and analysis of the essential steps in the structure determination of protein-ligand structures.

Reference

Krojer, T., Talon, R., Pearce, N., Collins, P., Douangamath, A., Brandao-Neto, J., Dias, A., Marsden, B., and von Delft, F. (2017). The XChemExplorer graphical workflow tool for routine or large-scale protein–ligand structure determination. *Acta Cryst D 73*, 267–278.

XChemExplorer makes extensive use of other people's software, therefore please cite their work accordingly:

- XIA2
- DIMPLE
- ACEDRG
- PanDDA
- COOT
- REFMAC
- PHENIX
- MolProbity

Getting started

Prerequisites

XCE works on any Mac OSX or Linux system, but it is essential that CCP4 (http://www.ccp4.ac.uk) version 7.0 or higher is installed correctly configured. XCE uses the python version that comes with it and will therefore not work if it does not exist. Additionally, it may be useful to also install PHENIX, since XCE uses several of its tools for validation purposes.

Installation

Download XChemExplorer from http://tkrojer.github.io/XChemExplorer

Put the gzipped tar archive to wherever you want XCE to be installed. In case you have no root privileges, put it somewhere into your home directory, e.g.:

/home/tkrojer/software

Then change to the respective directory and unpack the archive, e.g.:

cd /home/tkrojer/software

gunzip XChemExplorer-1.2.tar.gz

tar –xvf XChemExplorer-1.2.tar

This will create a new directory, i.e. the XChemExplorer directory. Change into this directory, e.g.:

cd XChemExplorer-1.2

The contents of the directory should look something like this when you type '**ls –l**':

-rwxr-xr-x 1 tkrojer users 238K Jun 18 13:58 XChemExplorer.py

-rwxr-xr-x 1 tkrojer users 269 Jun 18 13:58 XChemExplorer_local.sh

-rwxr-xr-x 1 tkrojer users 316 Jun 18 13:58 XChemExplorer_dmd.sh

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 13:58 web

-rwxr-xr-x 1 tkrojer users 553 Jun 18 13:58 setupssh.sh

drwxr-xr-x 2 tkrojer users 4.0K Jun 18 13:58 setup-scripts

-rwxr-xr-x 1 tkrojer users 2.8K Jun 18 13:58 README.md

drwxr-xr-x 2 tkrojer users 4.0K Jun 18 13:58 lib

drwxr-xr-x 2 tkrojer users 4.0K Jun 18 13:58 image

drwxr-xr-x 2 tkrojer users 4.0K Jun 18 13:58 icons

drwxr-xr-x 2 tkrojer users 4.0K Jun 18 13:58 helpers

drwxr-xr-x 2 tkrojer users 4.0K Jun 18 13:58 gui_scripts

-rwxr-xr-x 1 tkrojer users 182 Jun 18 13:58 Dockerfile

-rwxr-xr-x 1 tkrojer users 465 Jun 18 13:58 compile_test.py

lrwxrwxrwx 1 tkrojer users 20 Jun 18 14:33 XChemExplorer -> XChemExplorer_dmd.sh

The only thing left to do is to edit the XChemExplorer_dmd.sh file with a text editor. Change the line

export XChemExplorer_DIR='/usr/local/scripts/tobias/XChemExplorer'

to where you XChemExplorer is installed. In our example this would be

export XChemExplorer_DIR='/home/tkrojer/software/XChemExplorer-1.2'

That's it!

Usage

You can now run XCE by typing

/home/tkrojer/software/XChemExplorer-1.2/XChemExplorer_dmd.sh

It may however be easier if you insert an alias into your .bashrc or .cshrc file:

alias XChemExplorer='/home/tkrojer/software/XChemExplorer-1.2/XChemExplorer.sh

Settings

The settings tab contains information where XChemExplorer (XCE) will write files to and where it can find certain files. It also tells XCE the name and location of the SQLite database file.

If you use XCE at DLS as part of an XChem project somewhere in /dls/labxchem/… then you will usually not have to change anything. XCE will populate the directories and information about the SQLite database file with defaults.

The situation is different when you leave the labxchem environment; now you need to specify the information. Luckily however, there are really only 3 pieces of information required to get started:

Project directory

This is where XCE where all files that XCE creates will end up. The project directory contains a sub-directory for every crystal and the name of every sub-directory corresponds to the CrystalName field in the database. Please note that the names can be completely arbitrary, although it is highly recommended to choose some meaningful names. Please, check the XCE publication for more information about the structure and filename conventions of the project directory.

Reference Structure Directory

You can provide reference PDB, MTZ and CIF files in this directory. These files will be used for map calculation with DIMPLE or for selection of the best auto-processing result. The filenames can be arbitrary, however, if the files belong together then they need to have the same root.

Data Source

Here you need to specify the database file that you want to use. If you are starting from scratch, please check the next section which explains how to create a new database file

CCP4_SCR director

This is the same as your CCP4_SCR directory. It is essentially a scratch directory which XCE uses to save input scripts for processing, restraints generation and refinement to. Usually one can ignore this directory, however, it is a good place to start trouble-shooting in case a job did not behave as expected.

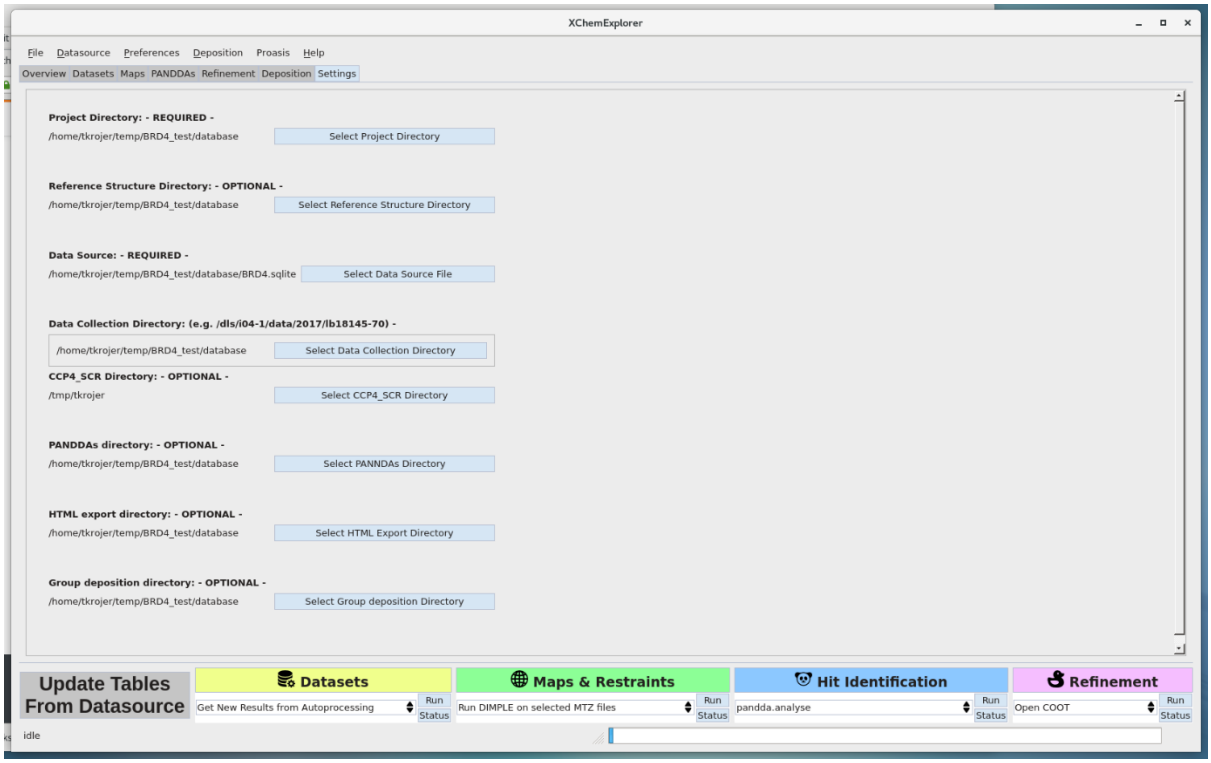


Figure 1. The settings tab in XChemExplorer.

Database

XCE uses a simple SQlite database to capture information, results and outcomes that are generated during the project. If your sample were collected at the XChem facility at the Diamond Light Source, then you will usually just take the soakDB file that was created during crystal preparation. Otherwise, just select 'Create New Datasource (SQLite)' from the Datasource menu and create a new database.

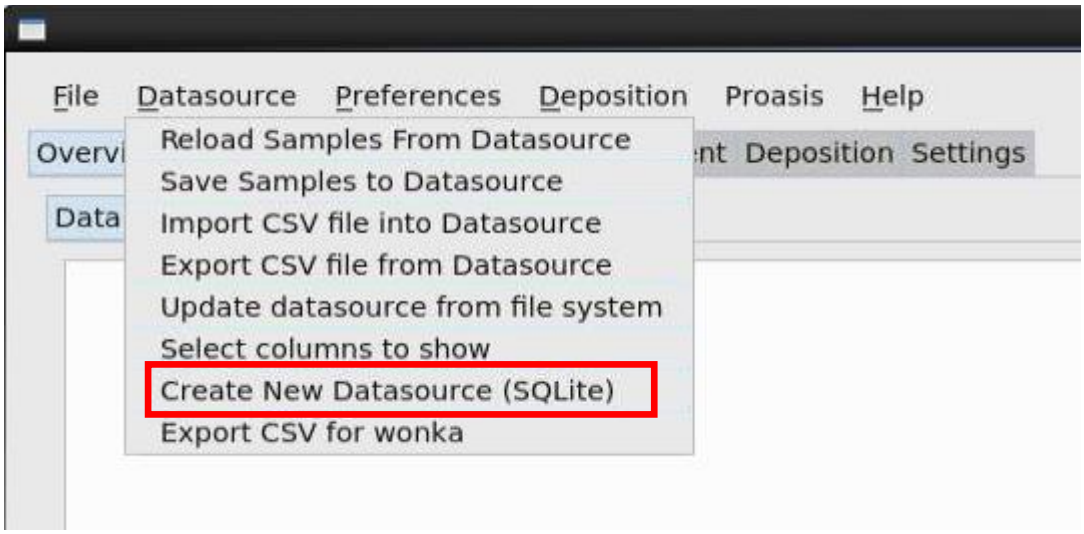


Figure 2. Datasource menu item 'Create New Datasource (SQLite)'.

Database update

Usually, there should be no need to make changes to the database and it generally not recommended to do so. One notable exception is the entry of information about the soaked/ co-crystallised compounds in case these compound are not part of the XChem fragment libraries. The initial ambition was to have this functionality available as part of XCE, but this approach will not be pursued for the time being. The DATASOURCE menu contains some rudimentary functionalities for database changes, but it is recommended to follow the instructions outlined here in case the need for database manipulation arises. Also, keep in mind that the tables in XCE only display the contents of the database. Changing the value of the fields, although it is possible, has not effect on the database! A single click on 'Update Table From Datasource' will bring back the actual content of the database.

Software

There are several programs for manipulation of SQLite files, the one we typically use is SQLite Browser (https://sqlitebrowser.org). It is available for Windows, Mac and Linux.

Instructions

It is easily possible to make all the required changes directly within SQLite Browser, but this really only make sense if you need to change a few fields. For large scale manipulations you should export the respective table (usually only the mainTable) into CSV format and then make the required changes in Microsoft Excel. Other programs like OpenOffice should work as well, however there have been anecdotal reports of import problems.

Before you start, make a copy of the current database in case something goes wrong!!!

Next, open the sqlite file and export the export mainTable to a CSV file:

File -> Export -> Table(s) as CSV file…

Save as mainTable.csv if asked.

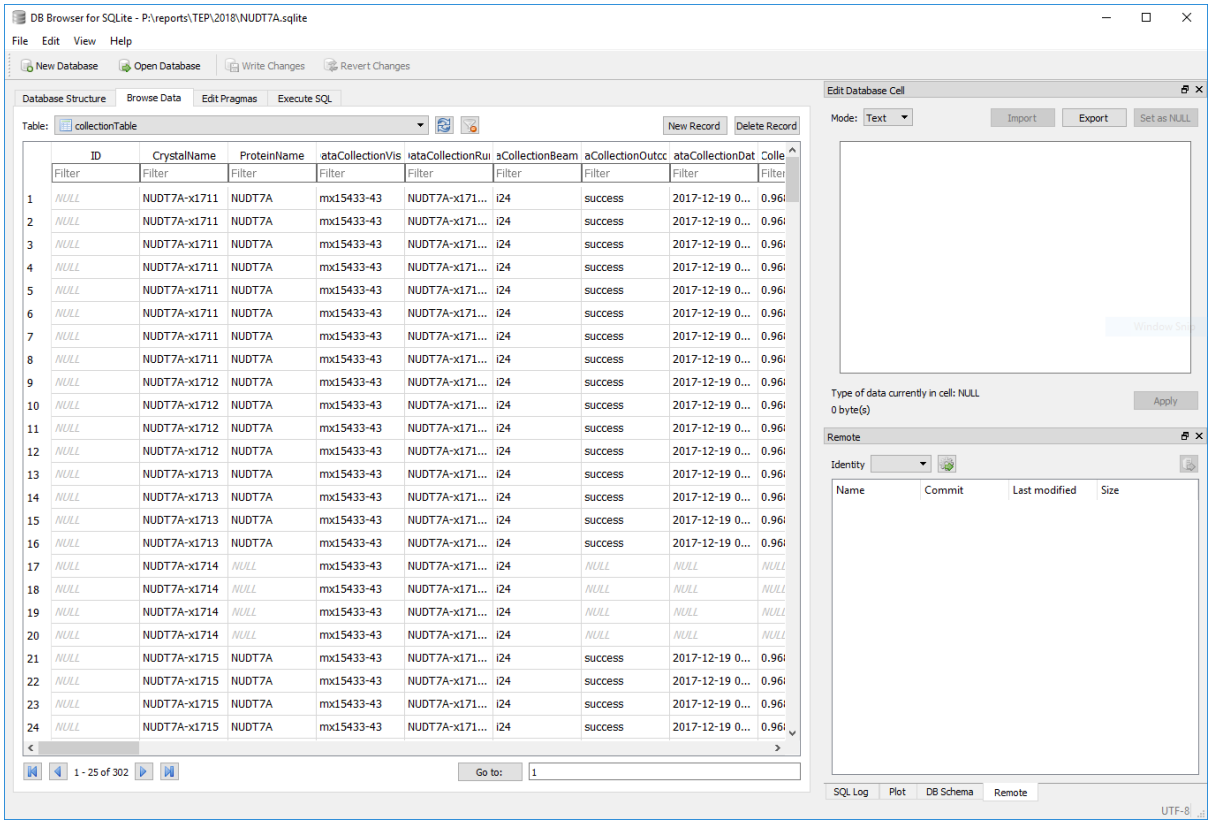


Figure 3. SQLite browser main window.

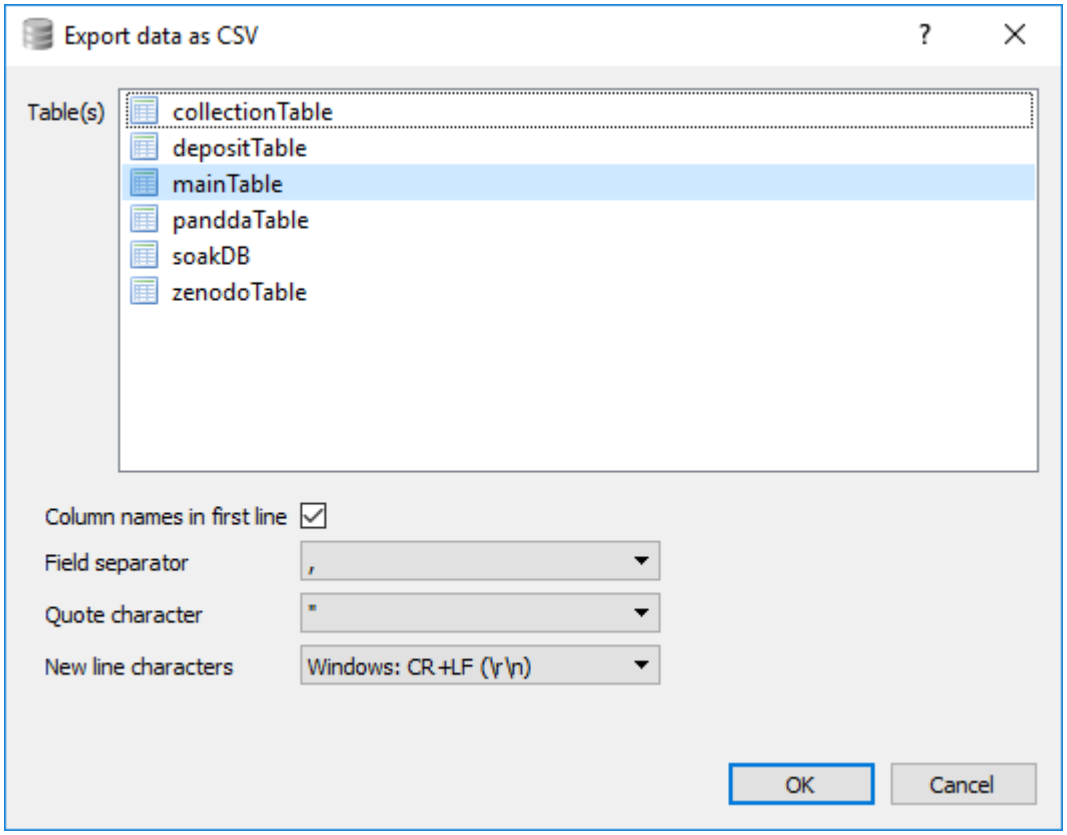


Figure 4. SQLite browser - export table to CSV file.

Then delete mainTable from database. In the 'Database Structure' tab, highlight the table you want to delete, then go to Edit -> Delete Table. Then press 'Write Changes' and close the file.

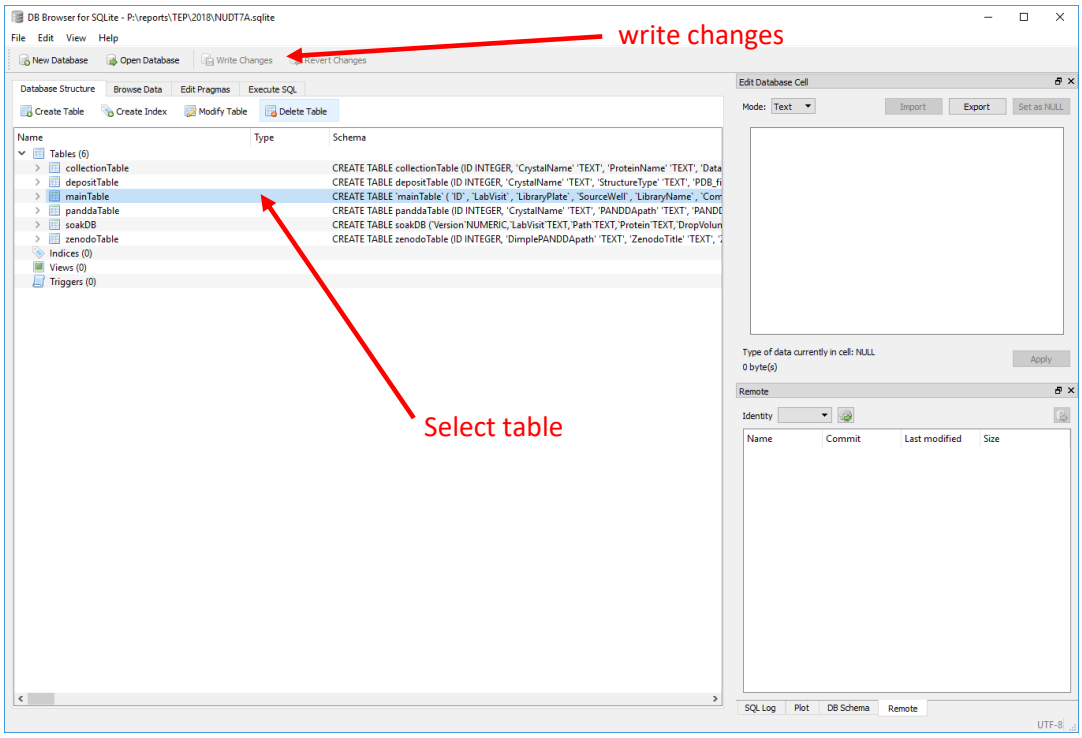


Figure 5. SQLite browser - select table for deletion.

Now open the CSV file with Microsoft EXCEL and make changes as necessary. The fields that most often will need changing are CompoundSMILES and CompoundCODE.

Open the SQLite file again and select File -> Import -> Table from CSV file…

Make sure the table name is *mainTable* and that '*Column names in first line*' is checked.

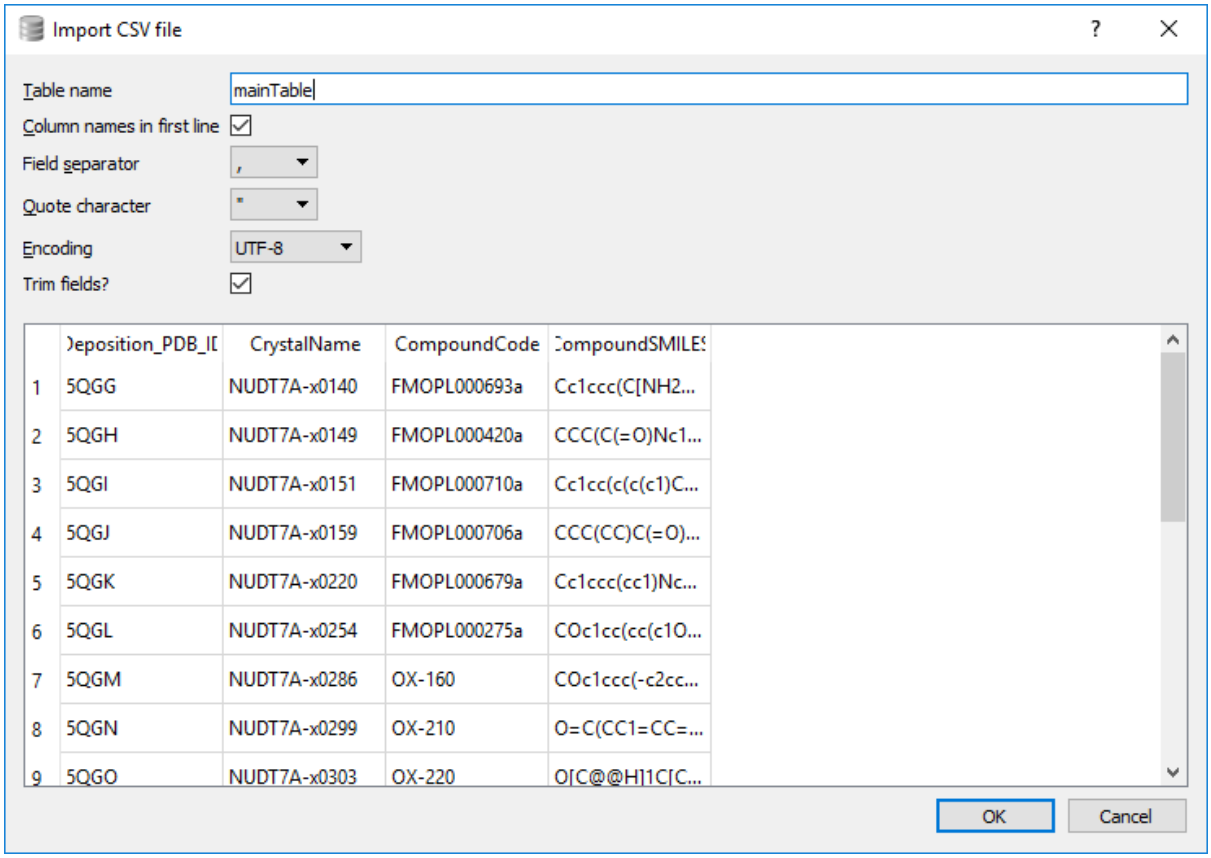


Figure 6. SQLite browser - import CSV file as table into SQLite database.

Dataset reprocessing

Running xia2 or dials

XCE can be used to reprocess datasets with either XIA2 or DIALS. This option is available in the Datasets/ Reprocess tab.

First select the datasets directory, then press search datasets. Note that if you want to process multiple-datasets then select the respective top-level directory under which the individual datasets are stored.

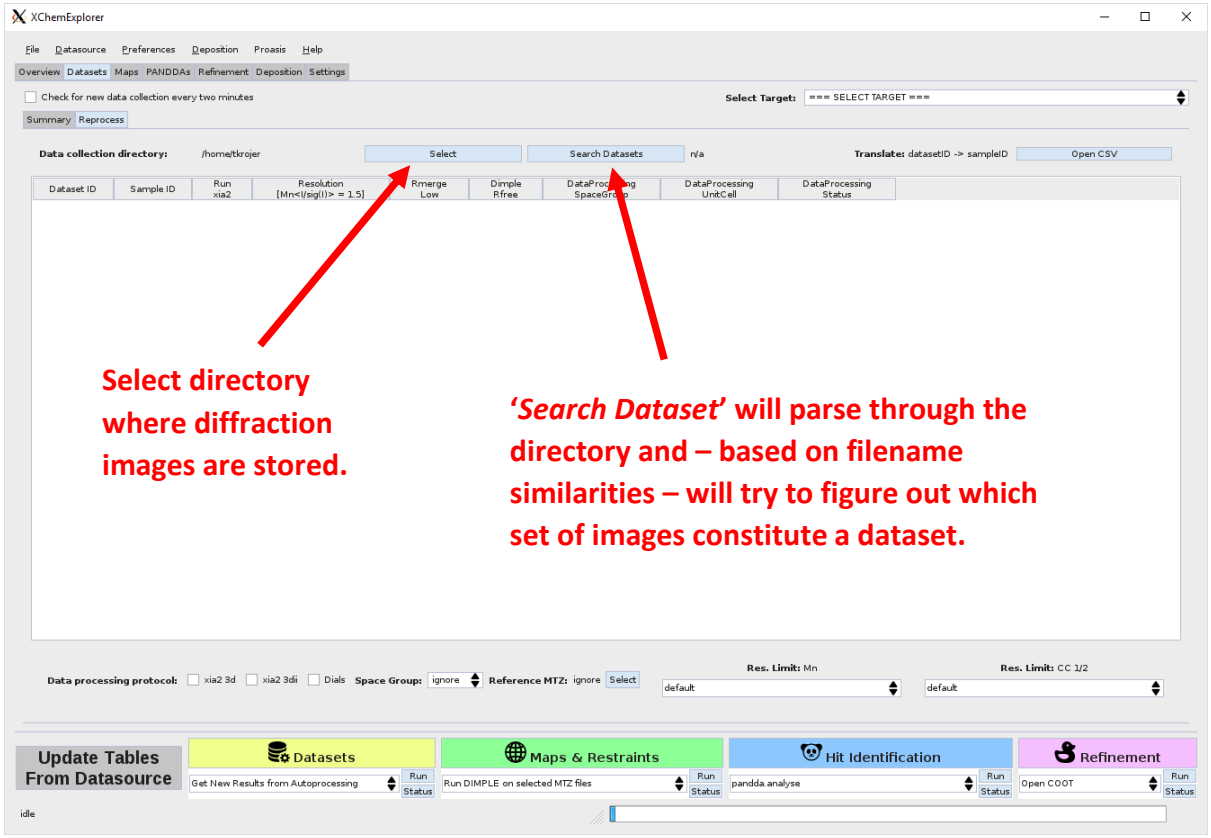


Figure 7. (Re-) processing of diffraction data in XCE.

Depending on the number of datasets, parsing of the respective directory may take a while. However, the progress bar will indicate how far along the program is and finally the table should get populated as in the exemplary screenshot below. There are a few things one should keep in mind:

- the datasetID is the name of each folder in the 'Data collection directory'; e.g. all subfolders in /xdata/BRD4A

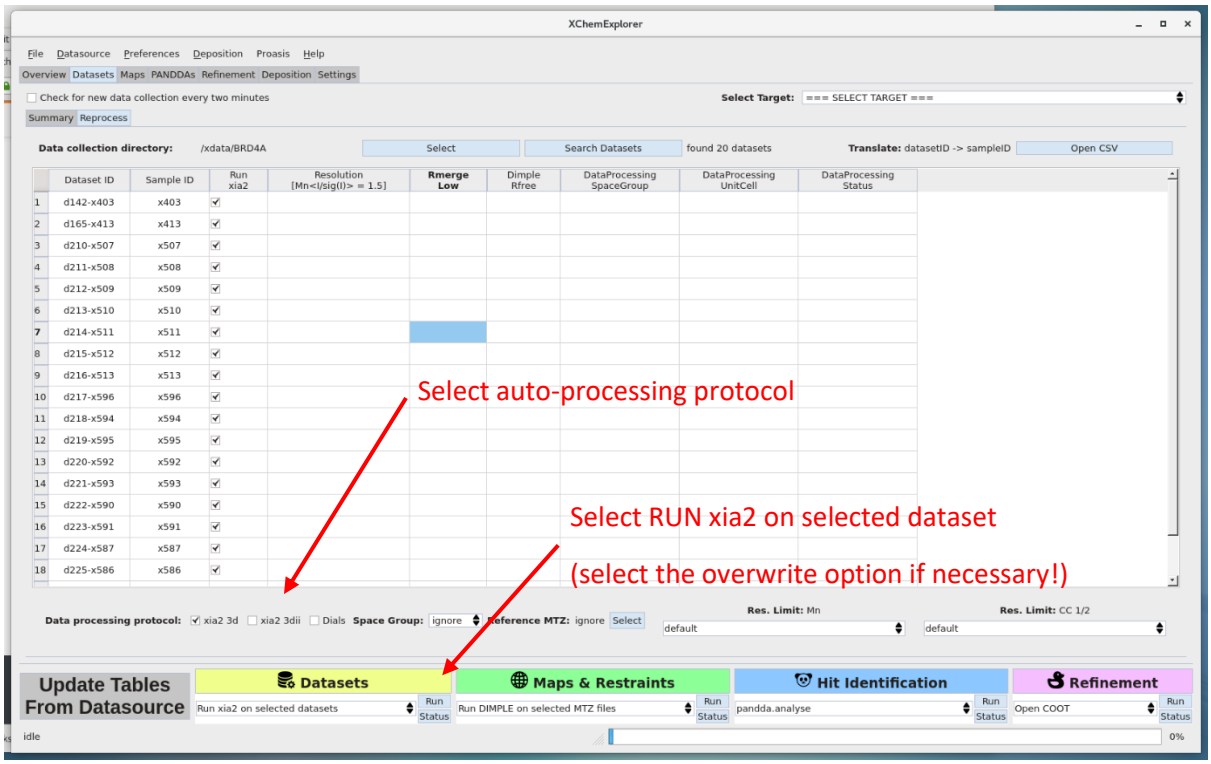
- only folders with more than 20 diffraction images in them will be listed

- XCE assumes that datasetID=sampleID; and if there is already an entry in the datasource for a particular sampleID then it will display things like Rmerge etc.

- you can change the sampleID simply by entering another value in the cell

- if you want to user different sample IDs for many datasets, then you can use the 'Translate dataset ID -> sample ID' option.

- you can either select all the samples you want to run by clicking on the checkbox or you select a range of rows with the mouse, right-click, click on 'mark selected for reprocessing'



Next, select the data processing protocol(s) you want to use (red arrow above), then select 'Run xia2 on selected datasets' from the yellow Datasets combobox and press 'RUN'. Note that all data processing results will end up in the project directory.

If you want to provided reference files then put the respective files into the reference directory.

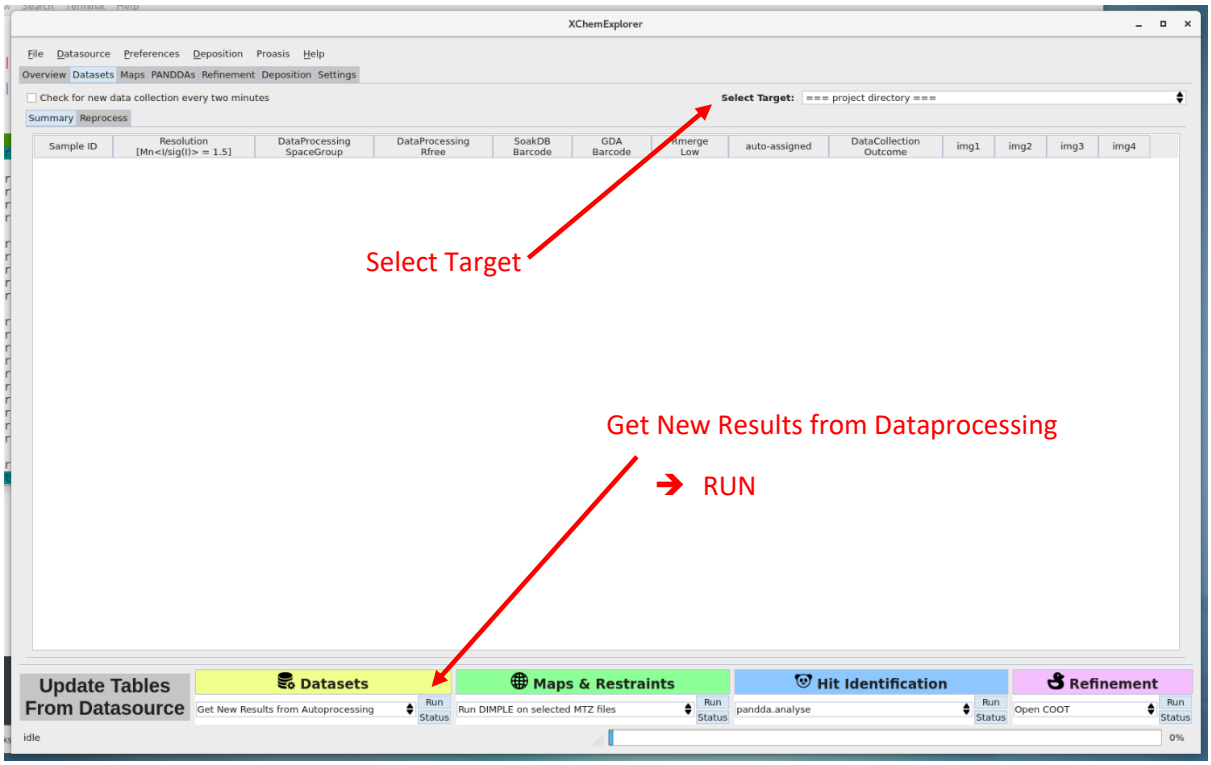
Please note that if your local machine is not automatically able to submit jobs to a computer cluster via qsub, it will process them sequentially on the local machine. Needless to say that this may keep the machine busy for a while in case you want to process tens or even hundreds of datasets. In this case it is advisable to use only one data processing protocol.

Please also note that XCE does currently not indicate when the jobs are finished (v1.2). You need to check the workload on your machine to find out.

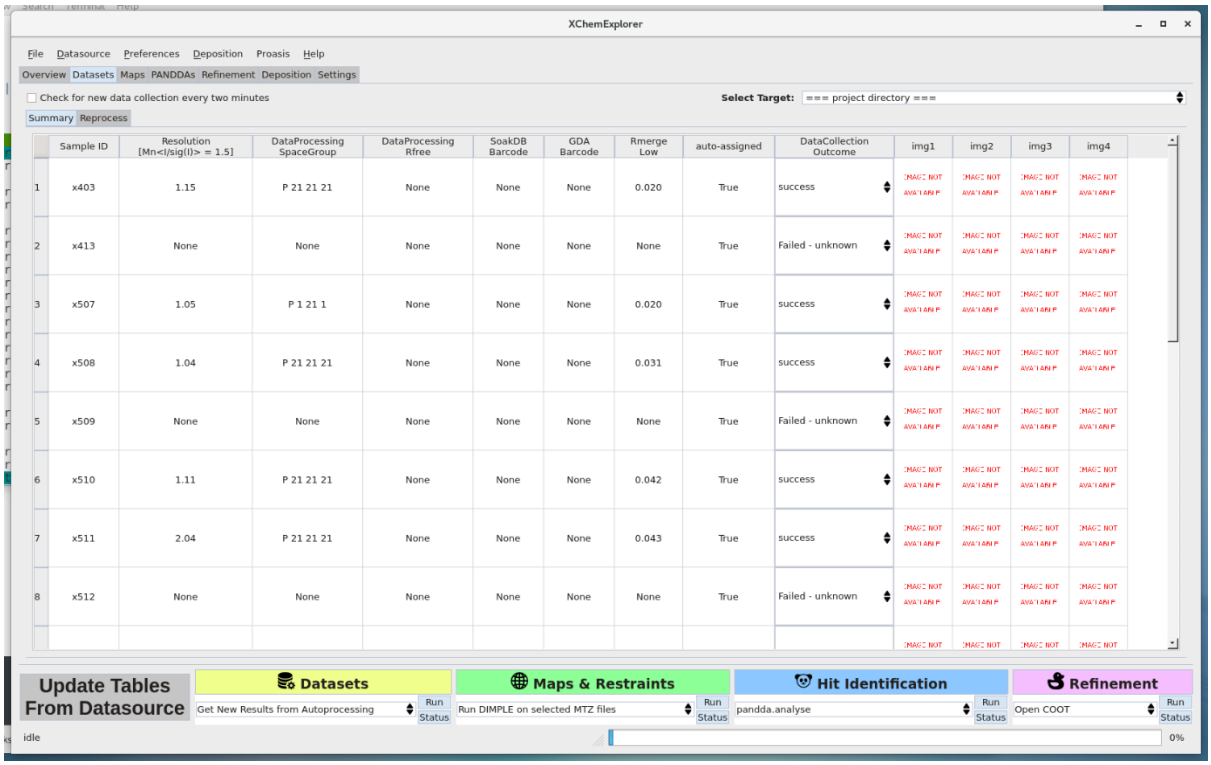
Getting the results into the database

Now you need to get the data processing results/ outcomes back into the database. Stay in the main Datasets tab, but switch to the Summary sub-tab. Go to the select targets dropdown and select '===

project directory ===’. Then select ‘Get New results from Autoprocessing’ from the yellow action box and press RUN.



If everything worked well, each sample directory with a successful data processing outcome should look something like this:



As you can see, you have symbolic links to the MTZ and AIMLESS logfile from xia2; and the files have the same root as the respective sample directory.

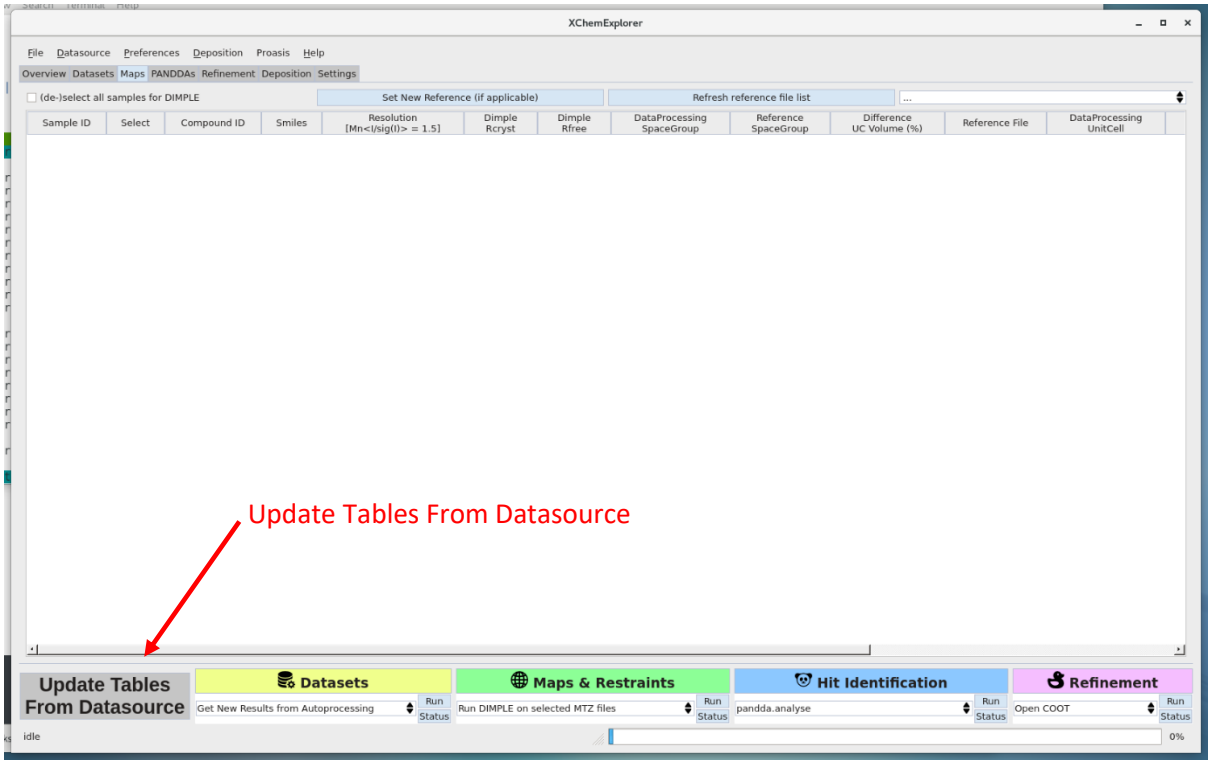
```
drwxr-xr-x 3 tkrojer users 4.0K Jun 18 13:37 diffraction_images

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 14:42 processed

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 15:50 jpg

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 15:50 autoprocessing

lrwxrwxrwx 1 tkrojer users   58 Jun 18 15:50 x596.mtz -> autoprocessing/database-run_13d/AUTOMATIC_DEFAULT_free.mtz

lrwxrwxrwx 1 tkrojer users   61 Jun 18 15:50 x596.log -> autoprocessing/database-run_13d/AUTOMATIC_DEFAULT_aimless.log
```

Initial Map Calculation

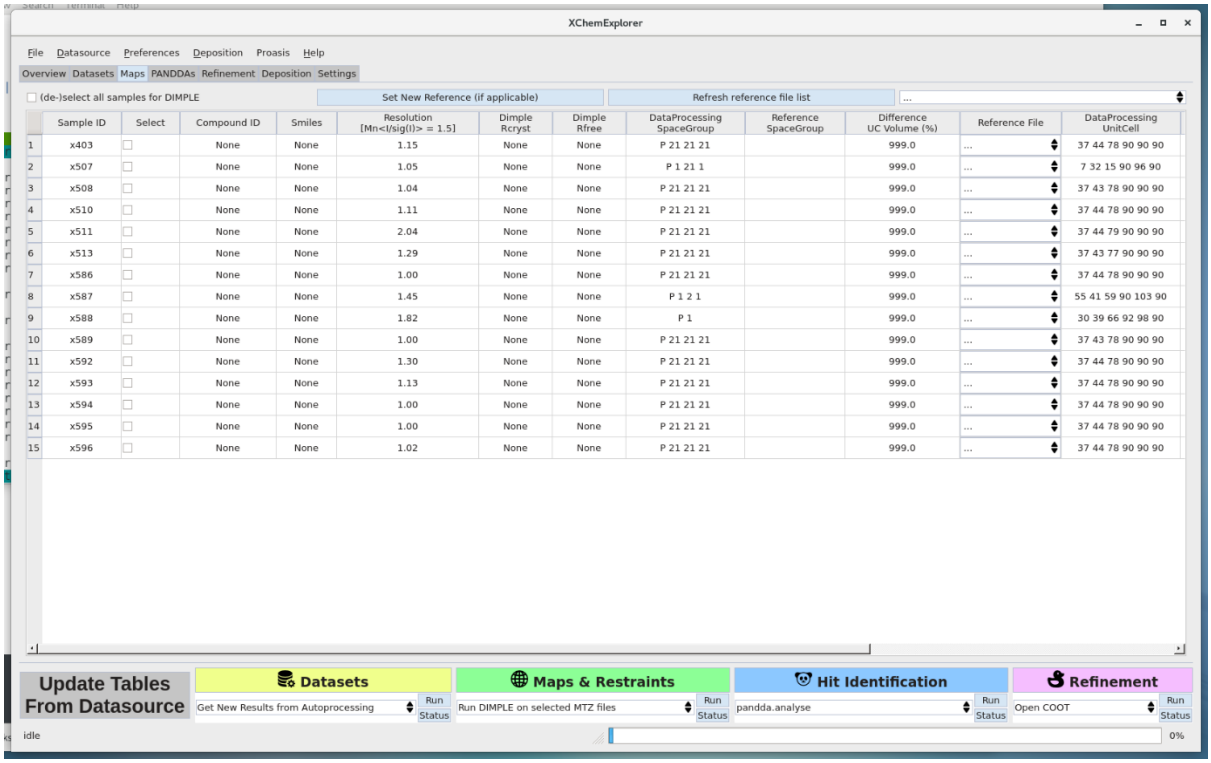
XCE uses DIMPLE to perform an initial round of refinement and to calculate the resulting 2fofc and fofc maps.

Please note that the tables in XCE do no update automatically! It does so during startup, but they will not refresh automatically. Also, there is currently a bug which means that the status indicators in the MAPS table are not always up to date (v1.2).

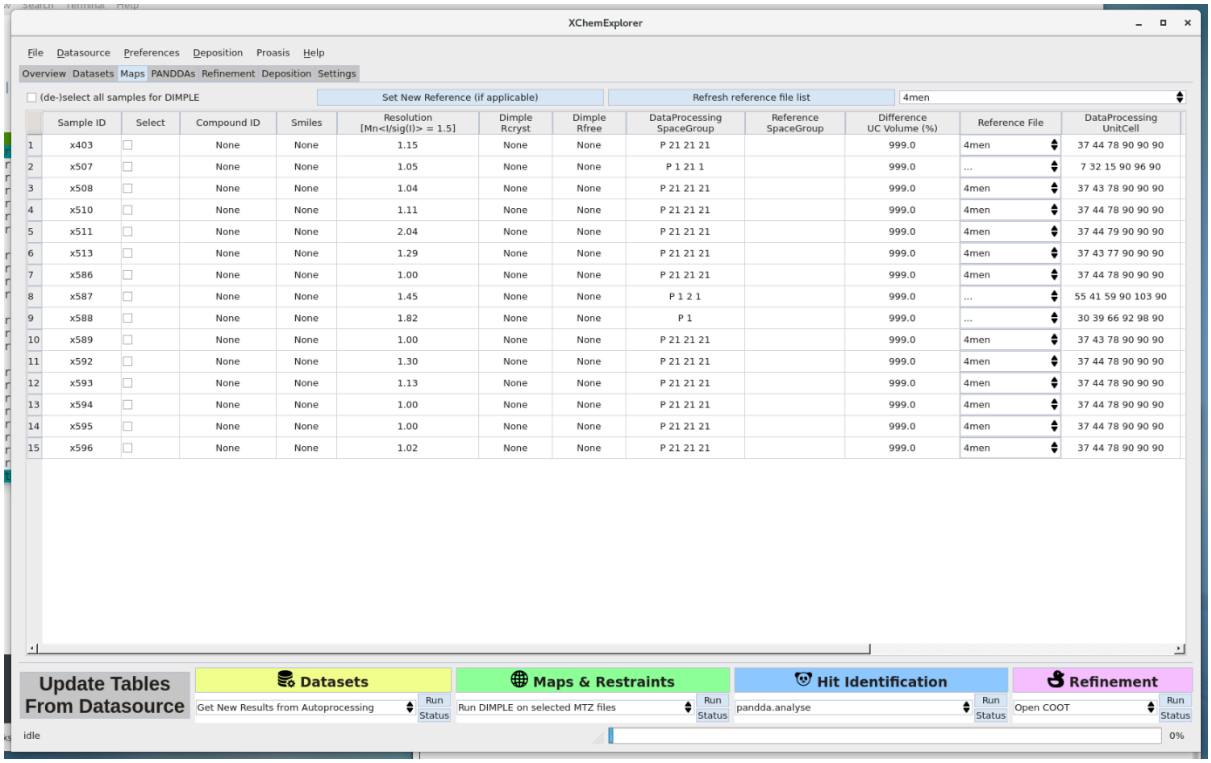
If you have successfully processed your datasets or read in the results from DLS auto-processing, then press on the grey ‘update tables from datasource’ button at the lower left hand corner. You can press this button as often as you want and it will update all tables with the exception being the Datasets Summary table with the latest information from the database.



In our current example, the table will look like this:

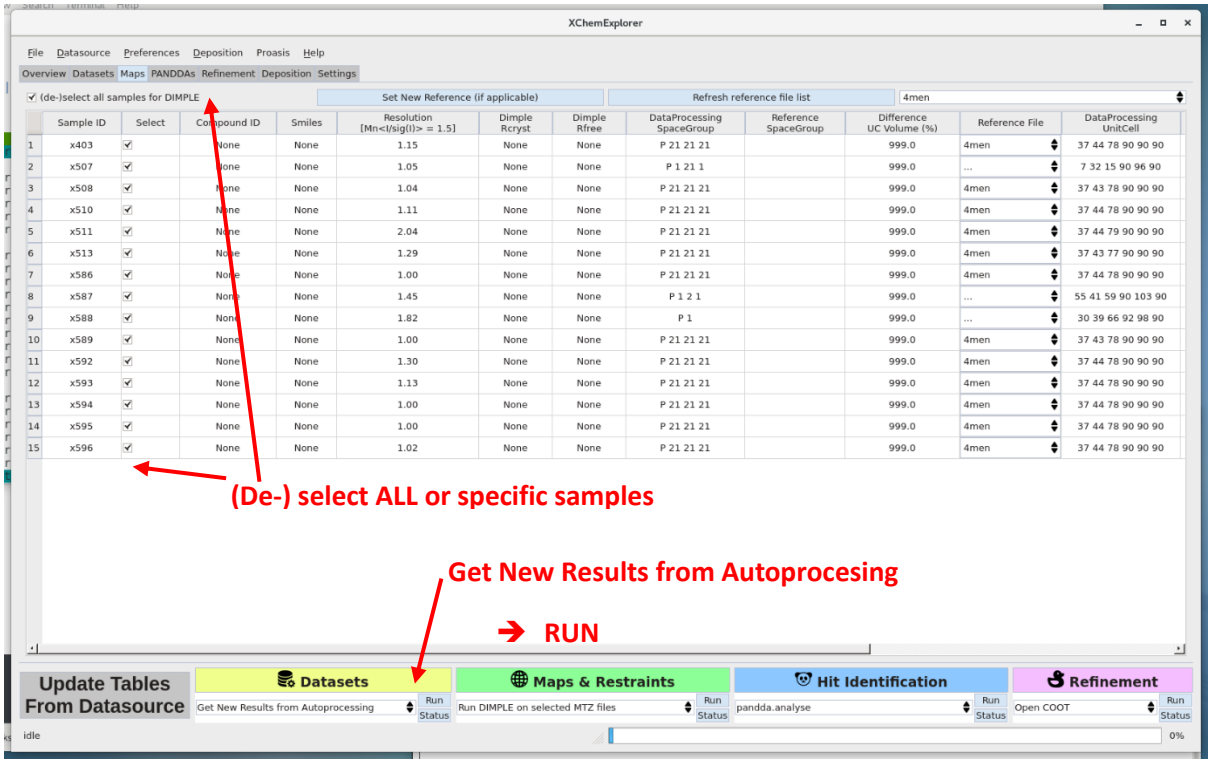


We can now see information about the high resolution limit and space group for each crystal in the table, but the information about which reference PDB file to use is still empty. This is because in the example I have not yet provided a suitable reference PDB file in the reference directory. After I have done so, you need to press ‘Refresh reference file list’, then select the reference file that you want to use, and then press ‘Set New Reference (if applicable)’.



You can check the XCE publication for more information about how XCE selects which reference file to use. Briefly, XCE will go through all PDB files that you have provided in the reference directory and read the CRYST card in the header. From the CRYST card, it will calculate the unit cell volume and determine the point group. Then it will compare this to the unit cell volume and point group of your crystal. If the unit cell volume differs by less then 12% (this can be adjusted in the Preferences menu) and the two have the same point group, then XCE will consider this to be a suitable input file for Dimple. This selection mechanism works well as long as you have either different point groups or the unit cell volume differs significantly between different space groups. It will most likely struggle in cases where you have different crystal forms that are not detectable by the rather coarse selection mechanism! However, I have often found that if XCE does not find a suitable reference file it is recommended to manually check what is going on. It may be that the presence of a certain ligand triggered some change that is best investigated outside the XCE workflow. If it really is a different crystal form, solve the structure as you usually would, then add the resulting structure to your reference file directory.

Now select which crystals you want to process (or simply choose *select all samples for DIMPLE*) and then choose '*Run DIMPLE on selected MTZ files*' from the green action box and press '*RUN*'.



If everything worked as expected, then then respective sample directory should now look like this:

```
drwxr-xr-x 3 tkrojer users 4.0K Jun 18 13:37 diffraction_images

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 15:15 processed

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 15:50 jpg

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 15:50 autoprocessing

lrwxrwxrwx 1 tkrojer users   58 Jun 18 15:50 x403.mtz -> autoprocessing/database-run_13d/AUTOMATIC_DEFAULT_free.mtz

lrwxrwxrwx 1 tkrojer users   61 Jun 18 15:50 x403.log -> autoprocessing/database-run_13d/AUTOMATIC_DEFAULT_aimless.log

drwxr-xr-x 3 tkrojer users 4.0K Jun 18 16:18 dimple

lrwxrwxrwx 1 tkrojer users   53 Jun 18 16:21 dimple.pdb -> dimple/dimple_rerun_on_selected_file/dimple/final.pdb

lrwxrwxrwx 1 tkrojer users   53 Jun 18 16:21 dimple.mtz -> dimple/dimple_rerun_on_selected_file/dimple/final.mtz

lrwxrwxrwx 1 tkrojer users   53 Jun 18 16:21 2fofc.map -> dimple/dimple_rerun_on_selected_file/dimple/2fofc.map

lrwxrwxrwx 1 tkrojer users   52 Jun 18 16:21 fofc.map -> dimple/dimple_rerun_on_selected_file/dimple/fofc.map

-rw-r--r-- 1 tkrojer users 1.3M Jun 18 16:21 x403.free.mtz
```

You can now either run PanDDA or move straight to the Refinement tab and look at your electron density maps after initial refinement.

Refinement

Refinement stages

Figure 8 Refinement progress model of XCE

The figure summarises the refinement stage model that XCE uses to track the progress of each sample and which is also used to triage samples. You need at least do some initial refinement to be able to look at samples in the refinement interface.

Overview

A list of all models that are currently ‘in refinement’ can be viewed in the ‘Refinement’ tab.

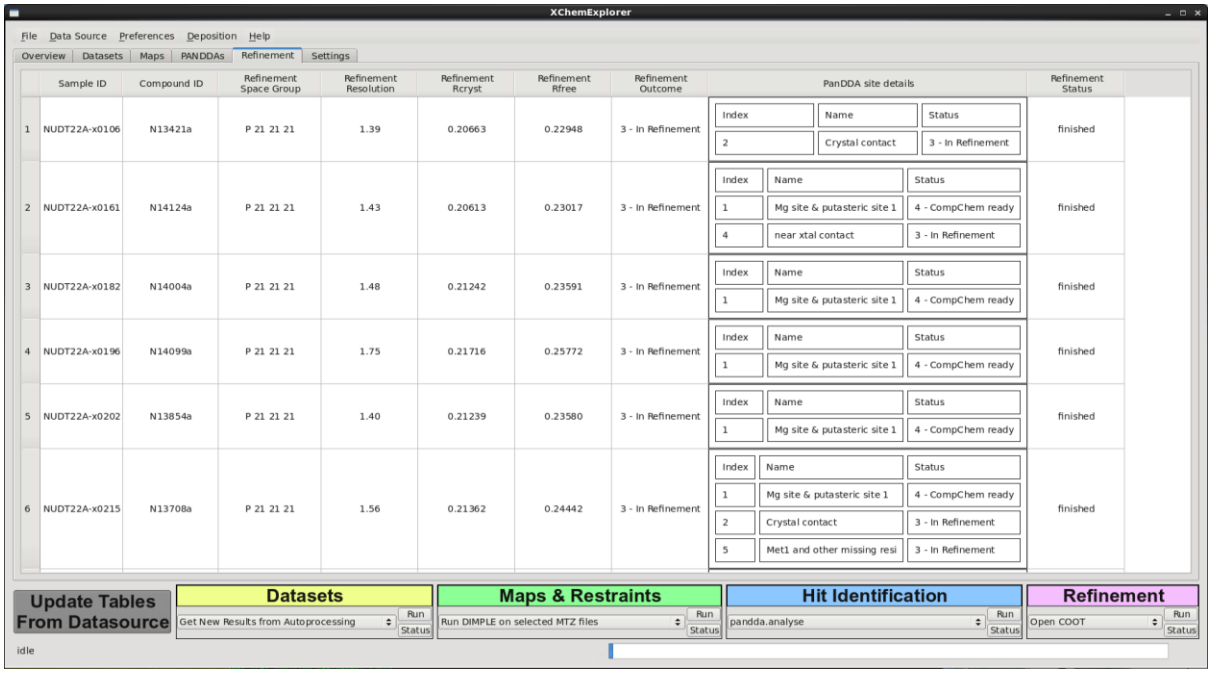


Figure 9 Refinement tab in XCE

If you want to inspect them in COOT and if necessary want to refine them further, choose ‘*Open COOT*’ from the magenta action box and press RUN. F

Figure 10 Refinement task box

This will launch COOT and the respective XCE interface (Figure 11).

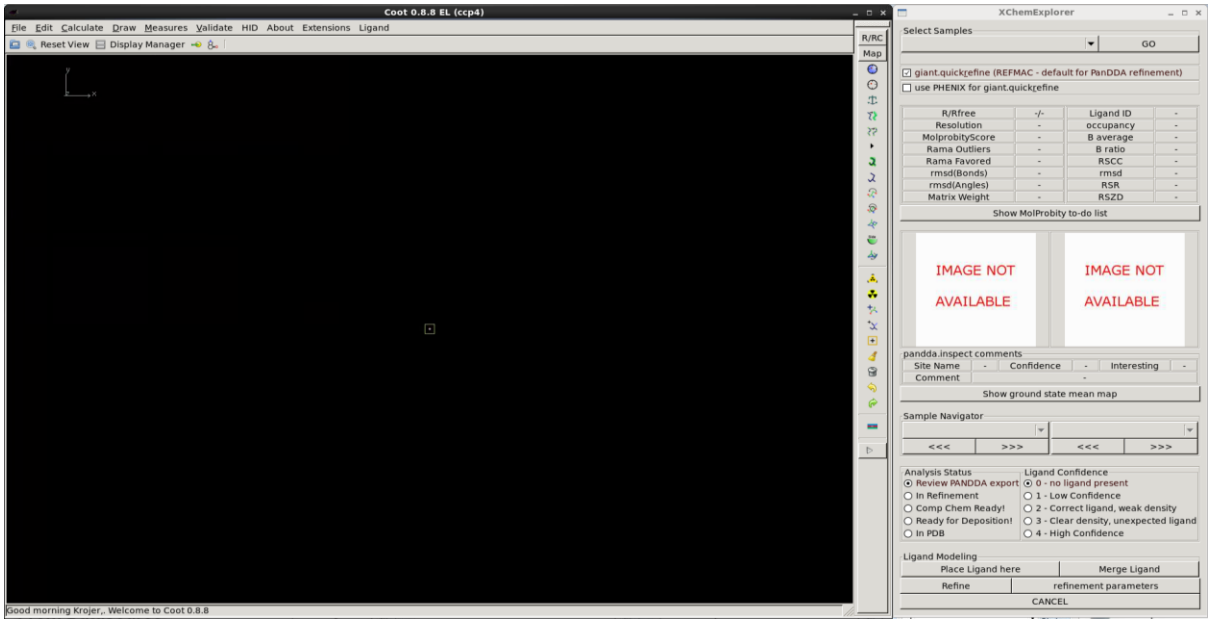


Figure 11 XCE Refinement interface for COOT

First you need to select the subset of samples you want to review/ refine from the drop-down menu at the top of the XCE interface (Figure 11). The drop-down lets you choose the Refinement Stage and once you press GO will load all samples that are in the respective Refinement stage. The image below shows you all the available categories

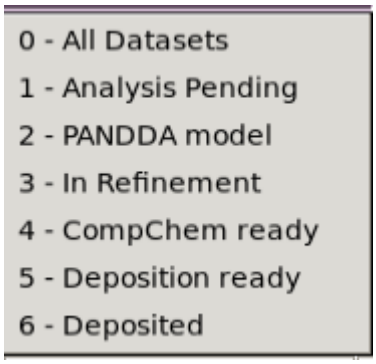


Figure 12 Sample selection criteria.

For example: if you want to load all models which are already in refinement, then select category 3 and press GO. In the example below, 79 structures are currently being refined (Figure 13).

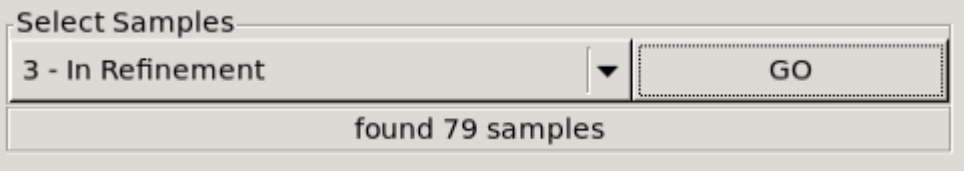
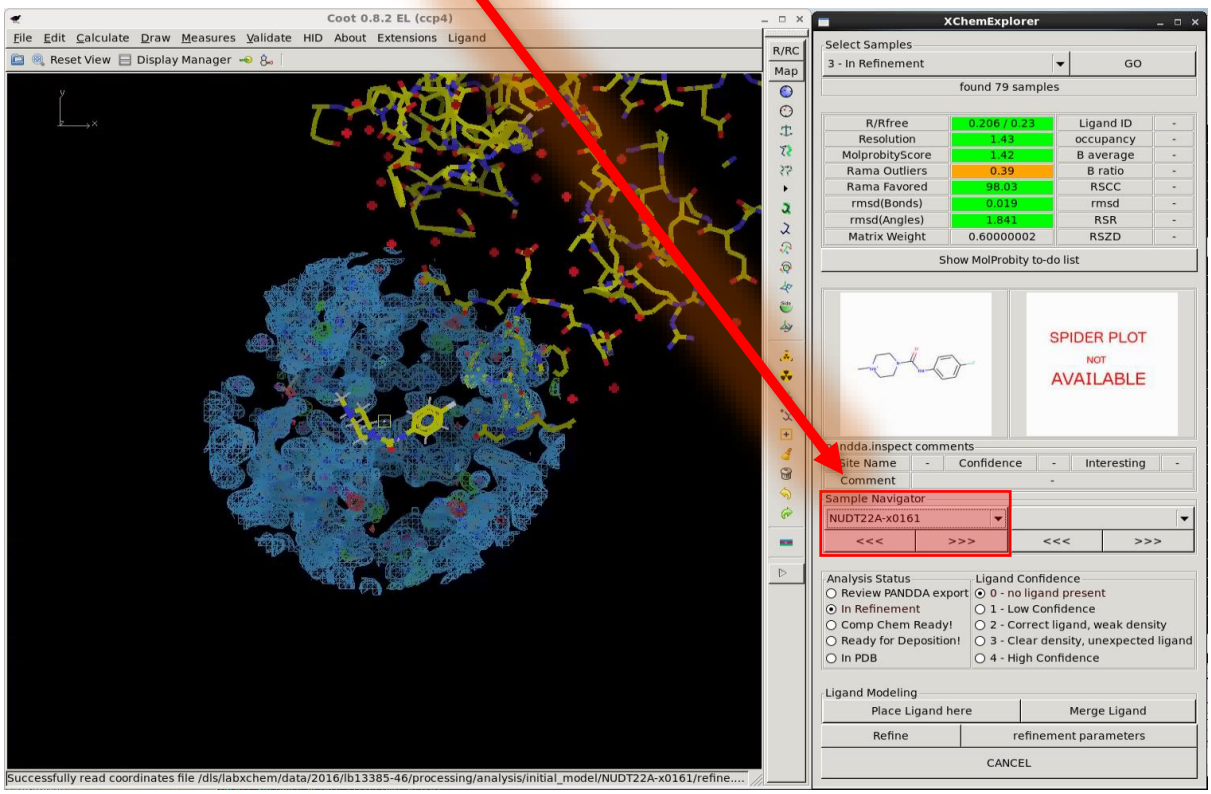


Figure 13 Exemplary reponse after sample selection.

In the left column of the 'Sample Navigator' Section, use the arrow buttons or the drop-down menu to select the structure of interest:



XCE will load the structure[1], 2fofc map (blue), fofc (green/red) map and a pdb file (+ dictionary) of the ligand if the latter was created and specified in the database. This ligand molecule may be slightly confusing because it may seem to just float in space. However, the molecule is completely

[1] Structure refers to the file called refine.pdb in the respective sample directory, or if no refinement has been carried out so far and category 0 or 1 are selected, then it will try to load dimple.pdb. Note: if you use XCE throughout the process, then refine.pdb as well as dimple.pdb are not actual files but symbolic links that point to the most recently refined file.

ignored as long as it is not merged into the main structure! It is only loaded to enable quick modelling if the ligand is not already part of the structure.

Frequently asked questions

What happens when you step through the models?

XCE will load a file called refine.pdb (or dimple.pdb in case no refinement has been carried out so far) from the sample directory and if available a pdb file of the ligand and the respective restraints. Additionally, 2fofc as well as fofc maps are loaded (or they are calculated on the fly from refine.mtz/ dimple.mtz if the map files were for whatever reason no pre-calculated). Note that refine.pdb/mtz and dimple.pdb/mtz are therefore reserved file names. Hence, if you wanted to manipulate your model outside the XCE environment you can easily do so and XCE will read the manipulated model in as long as it's called refine.pdb and present in the respective sample directory. One thing to keep in mind though: XCE carries out the actual refinement in a subfolder called Refine_<cycle number> and only links the resulting pdb/mtz files as refine.pdb/mtz into the project directory. Every time a new refinement is launched, it will first delete the respective symbolic links. But it will also delete it if it is an actual file. So if you want to keep the original, better create a symbolic link called refine.mtz. When you go through the models, it will remove all currently loaded models and load the aforementioned files from the next sample directory.

What happens when I make changes to the model?

All changes that you make to the model called refine.pdb will be preserved if you launch refinement (see next point). They will however be lost if you go to the next dataset. XCE will currently not ask if you want to keep the changes, the pdb file will be deleted from the list of molecules in COOT and it will be lost forever! Also, be careful if you read in additional molecules with the same name, for example if you want to analyse something. COOT does not mind if molecules have the same name since every molecule that is read in get a unique internal identifier. XCE however recognises molecules by filename it may get confused in case of duplicates.

What happens when I refine the model?

When you press refine, XCE will take the model called refine.pdb and save it to a subfolder called Refine_<cycle number + 1> as in.pdb together with the shell script that will be used for refinement. If something goes wrong and the reason for failure is not clear it is usually a good starting point to

look at the logfiles in the respective folder. Keep in mind that if you added for example water or other solvent molecules and did not merge them into refine.pdb, then they will not be included. XCE does not do any automatic merging!