The Coot User Manual

Paul Emsley
paul dot emsley at bioch dot ox dot ac dot uk

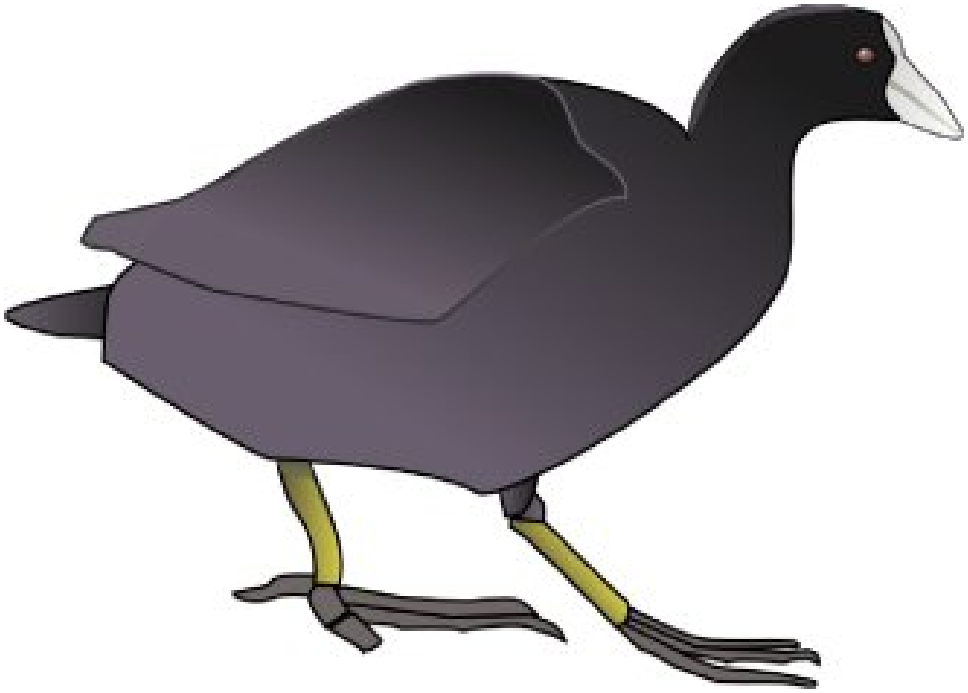

Table of Contents

1 Introduction

This document is the Coot User Manual, giving an overview of the interactive features. Other documentation includes the Coot Reference Manual and the Coot Tutorial. These documents should be distributed with the source code.

1.1 Citing Coot and Friends

If have found this software to be useful, you are requested (if appropriate) to cite:

> "Coot: model-building tools for molecular graphics" Emsley P, Cowtan K *Acta Crystallographica Section D-Biological Crystallography* **60**: 2126-2132 Part 12 Sp. Iss. 1 DEC 2004

The reference for the REFMAC5 Dictionary is:

> REFMAC5 dictionary: "Organization of Prior Chemical Knowledge and Guidelines for its Use" Vagin AA, Steiner RA, Lebedev AA, Potterton L, McNicholas S Long F, Murshudov GN *Acta Crystallographica Section D-Biological Crystallography* **60**: 2184-2195 Part 12 Sp. Iss. 1 DEC 2004"

If using "SSM Superposition", please cite:

> "Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions" Krissinel E, Henrick K *Acta Crystallographica Section D-Biological Crystallography* **60**: 2256-2268 Part 12 Sp. Iss. 1 DEC 2004

The reference for the the Electron Density Server is:

> GJ Kleywegt, MR Harris, JY Zou, TC Taylor, A Wählby, TA Jones (2004), "The Uppsala Electron-Density Server", *Acta Crystallographica Section D-Biological Crystallography* **60**, 2240-2249.
>
> Please also cite the primary literature for the received structures.

1.2 What is Coot?

Coot is a molecular graphics application. Its primary focus is crystallographic macromolecular model-building and manipulation rather than representation *i.e.* more like Frodo than Rasmol. Having said that, Coot can work with small molecule (SHELXL) and electron microscopy data, be used for homology modelling, make passably pretty pictures and display NMR structures.

Coot is Free Software. You can give it away. If you don't like the way it behaves, you can fix it yourself.

1.3 What Coot is Not

Coot is not:

- CCP4's official Molecular Graphics program[1]

[1] The official CCP4 graphics program (which contains parts of Coot (and Coot contains parts of CCP4MG)), CCP4MG is under the direct control of Liz Potterton and Stuart McNicholas.

- a program to do refinement[2]
- a protein crystallographic suite[3].

1.4 Hardware Requirements

The code is designed to be portable to any Unix-like operating system. Coot certainly runs on SGI IRIX64, RedHat Linux of various sorts, SuSe Linux[4] and MacOS X (10.2). The sgi Coot binaries should also work on IRIX.

If you want to port to some other operating system, you are welcome[5]. Note that your task will be eased by using GNU GCC to compile the programs components.

1.4.1 Mouse

Coot works best with a 3-button mouse and works better if it has a scroll-wheel too (see Chapter 2 for more details)[6].

1.5 Environment Variables

Coot responds to several environment variables that modify its behaviour.

- `COOT_STANDARD_RESIDUES` The filename of the pdb file containing the standard amino acid residues in "standard conformation"[7]
- `COOT_SCHEME_DIR` The directory containing standard (part of the distribution) scheme files
- `COOT_SCHEME_EXTRAS_DIR` The directory containing bespoke scheme files. This variable is not set by default. If you set it, Coot will test that it points to a directory, and if it does, Coot will load all the .scm files in that directory.
- `COOT_REF_STRUCTS` The directory containing a set of high resolution pdb files used as reference structures to build backbone atoms from Cα positions
- `COOT_REF_SEC_STRUCTS` The directory containing a set of high-quality structures to be used as templates for fitting beta strands. If this is not set, then the directory COOT_REF_SEC_STRUCTS will be used to find the reference pdb files.
- `COOT_REFMAC_LIB_DIR` Refmac's CIF directory containing the monomers and link descriptions. In the future this may simply be the same directory in which refmac looks to find the library dictionary.
- `COOT_RESOURCES_DIR` The directory that contains the splash screen image and the GTk application resources.
- `COOT_BACKUP_DIR` The directory to which backup are written (if it exists as a directory). If it is not, then backups are written to the current directory (the directory in which coot was started).

---

[2] although it does have a local refinement algorithm it is no substitute for refmac (a wrapper for refmac is available).

[3] that's the job of the CCP4 Program Suite.

[4] so far only 8.2 verified.

[5] it's Free Software after all and I could give you a hand.

[6] I can get by with a one button Macintosh - but it's not ideal.

[7] as it is known in Clipper.

And of course extension language environment variables are used too:

- `PYTHONPATH` (for python modules)
- `GUILE_LOAD_PATH` (for guile modules)

Normally, these environment variables will be set correctly in the coot shell script.

1.6 Command Line Arguments

Rather that using the GUI to read in information, you can use the following command line arguments:

- `--c` *`cmd`* to run a command *`cmd`* on start up
- `--script` *`filename`* to run a script on start up (but see Section Section 3.10 [Scripting], page 11)
- `--no-state-script` don't run the `0-coot.state.scm` script on start up. Don't save a state script on exit either.
- `--pdb` *`filename`* for pdb/coordinates file
- `--coords` *`filename`* for SHELX .ins/.res and CIF files
- `--data` *`filename`* for mtz, phs or mmCIF data file
- `--auto` *`filename`* for auto-reading mtz files (mtz file has the default labels FWT, PHWT)
- `--map` *`filename`* for a map (currently CCP4-format only)
- `--dictionary` *`filename`* read in a cif monomer dictionary
- `--help` print command line options
- `--stereo` start up in hardware stereo mode
- `--version` print the version of coot and exit
- `--code` *`accession-code`* on starting Coot, get the pdb file and mtz file (if it exists) from the EDS
- `--no-guano` don't leave "Coot droppings" i.e. don't write state and history files on exit.
- `--side-by-side` start in side-by-side stereo mode
- `--python` an argument with no parameters - used to tell Coot that the -c arguments should be process as python (rather than as scheme).

So, for example, one might use:

- `coot --pdb post-refinement.pdb --auto refmac-2.mtz --dictionary lig.cif`

1.7 Web Page

Coot has a web page:

- `http://www.biop.ox.ac.uk/coot`

There you can read more about the CCP4 molecular graphics project in general and other projects which are important for Coot[8].

[8] coot has several influences and dependencies, but these will not be discussed here in the User Manual.

1.8 Crash

Coot might crash on you - it shouldn't.

Whenever Coot manipulates the model, it saves a backup pdb file. There are backup files in the directory `coot-backup`[9]. You can recover the session (until the last edit) by reading in the pdb file that you started with last time and then use `File -> Recover Session...`.

I would like to know about coot crashing[10] so that I can fix it as soon as possible. If you want your problem fixed, this involves some work on your part sadly.

First please make sure that you are using the most recent version of coot. I will often need to know as much as possible about what you did to cause the bug. If you can reproduce the bug and send me the files that are needed to cause it, I can almost certainly fix it[11] - especially if you use the debugger (gdb) and send a backtrace too[12]. Note that you may have to source the contents of `bin/coot` so that the libraries are can be found when the executable dynamically links.

[9] COOT_BACKUP_DIR is used in preference if set

[10] The map-reading problem (documented in Section Section 6.1 [Maps in General], page 46) is already known.

[11] now there's a hostage to fortune.

[12] to do so, please send me the output of the following: `$ gdb ‘which coot-real‘ corefile` and then at the `(gdb)` prompt type: `where`, where *corefile* is the core dump file, ‘`core`’ or ‘`core.4536`’ or some such.

2 Mousing and Keyboarding

How do we move around and select things?

`Left-mouse Drag`
: Rotate view

`Ctrl Left-Mouse Drag`
: Translates view

`Shift Left-Mouse`
: Label Atom

`Right-Mouse Drag`
: Zoom in and out

`Ctrl Shift Right-Mouse Drag`
: Rotate View around Screen Z axis

`Middle-mouse`
: Centre on atom

`Scroll-wheel Forward`
: Increase map contour level

`Scroll-wheel Backward`
: Decrease map contour level

See also Chapter Section 8.5 [chap-hints], page 58 for more help.

2.1 Next Residue

`‘‘Space’’`
: Next Residue

`‘‘Shift’’ ‘‘Space’’`
: Previous Residue

See also “Recentring View” (Section Section 3.14 [Recentring View], page 15).

2.2 Keyboard Contouring

Use `+` or `-` on the keyboard if you don’t have a scroll-wheel.

2.3 Mouse Z Translation and Clipping

Here we can change the clipping and Translate in Screen Z

`Ctrl Right-Mouse Drag Up/Down`
: changes the slab (clipping planes)

`Ctrl Right-Mouse Drag Left/Right`
: translates the view in screen Z

2.4 Keyboard Translation

`Keypad 3` Push View (+Z translation)

`Keypad .` Pull View (-Z translation)

2.5 Keyboard Zoom and Clip

`N` Zoom out

`M` Zoom in

`D` Slim clip

`F` Fatten clip

2.6 Scrollwheel

When there is no map, using the scroll-wheel has no effect. If there is exactly one map displayed, the scroll-wheel will change the contour level of that map. If there are two or more maps, the map for which the contour level is changed can be set using either `HID -> Scrollwheel -> Attach scroll-wheel to which map?` and selecting a map number or clicking the "Scroll" radio button for the map in the Display Manager.

You can turn off the map contour level changing by the scroll wheel using:

```
(set-scroll-by-wheel-mouse 0)
```

(the default is 1 [on]).

2.7 Selecting Atoms

Several Coot functions require the selecting of atoms to specify a residue range (for example: Regularize, Refine (Section Section 5.1 [Regularization and Real Space Refinement], page 26) or Rigid Body Fit Zone (Section Section 5.4 [Rigid Body Refinement], page 29)). Select atoms with the Left-mouse. See also Picking (Section Section 8.9 [sec_picking], page 58).

Use the scripting function `(quanta-buttons)` to make the mouse functions more like other molecular graphics programs to which you may be more accustomed[1].

2.8 Virtual Trackball

You may not completely like the way the molecule is moved by the mouse movement[2]. To change this, try: `HID -> Virtual Trackball -> Flat`. To do this from the scripting interface: `(vt-surface 1)`[3].

If you *do* want `screen-z rotation` screen-z rotation, you can either use Shift Right-Mouse Drag or set the Virtual Trackball to Spherical Surface mode and move the mouse along the bottom edge of the screen.

---

[1] See also Section 2.9 [more on zooming], page 7

[2] Mouse movement in "Spherical Surface" mode generates a component of (often undesirable) screen z-rotation, particularly noticeable when the mouse is at the edge of the screen.

[3] `(vt-surface 0)` to turn it back to "Spherical" mode.

2.9 More on Zooming

The function `(quanta-like-zoom)` adds the ability to zoom the view using just Shift `+` Mouse movement[4].

There is also a Zoom slider (`Draw -> Zoom`) for those without a right-mouse button.

[4] this is off by default because I find it annoying.

3 General Features

The map-fitting and model-building tools can be accessed by using `Calculate -> Model/Fit/Refine...`. Many functions have tooltips[1] describing the particular features and are documented in Chapter Chapter 5 [Modelling and Building], page 26.

`F5:` posts the Model/Fit/Refine dialog

`F6:` posts the Go To Atom Window

`F7:` posts the Display Control Window

3.1 Version number

The version number of Coot can be found at the top of the "About" window (`Help -> About`).

This will return the version of coot:

```
$ coot --version
```

There is also a script function to return the version of coot:

```
(coot-version)
```

3.2 Anti-aliasing

The built-in antialiasing (for what it's worth) can be enabled using:

```
(set-do-anti-aliasing 1)
```

The default is `0` (off).

This can also be activated using `Edit Preferences -> Others -> Antialiasing -> Yes`.

If you have an nVidia graphics card, external antialiasing can be actived using nvidia-settings:

`Antialiasing Setting -> Override Application Settings` and slide the slider to the right.

On restarting Coot, it should be in antialias mode[2].

3.3 Molecule Number

Coot is based on the concept of molecules. Maps and coordinates are different representations of molecules. The access to the molecule is *via* the *molecule number*. It is often important therefore to know the molecule number of a particular molecule.

The Molecule Number of a molecule can be found by clicking on an atom of that molecule (if it has coordinates of course). The first number in brackets in the resulting text in the status bar and console is the Molecule Number. The Molecule Number can also be found in Display Control window (Section Section 3.7 [Display Manager], page 10). It is also displayed on the left-hand side of the molecule name in the option menus of the "Save Coordinates" and "Go To Atom" windows.

[1] Put your mouse over a widget for a couple of seconds, if that widget has a tooltip, it will pop-up in a yellow box (or a grey box for some reason if you are using Macintosh).

[2] that works for me, at least.

3.4 Display Issues

The "graphics" window is drawn using OpenGL. It is considerably smoother (i.e. more frames/sec) when using a 3D accelerated X server.

The view is orthographic (*i.e.* the back is the same size as the front). The default clipping is about right for viewing coordinate data, but is often a little too "thick" for viewing electron density. It is easily changed (see Section Section 3.15 [Clipping Manipulation], page 15).

Depth-cueing is linear and fixed on.

The graphics window can be resized, but it has a minimum size of 400x400 pixels.

3.4.1 Stereo

Hardware Stereo is an option for Coot (`Draw -> Stereo... -> Hardware Stereo -> OK`), side-by-side stereo is not an option.

The angle between the stereo pairs (the stereo separation) can be changed to suit your personal tastes using:

```
(set-hardware-stereo-angle-factor angle-factor)
```

where *angle-factor* would typically be between 1.0 and 2.0

3.4.2 Pick Cursor

When asked to pick a residue or atom, the cursor changes from the normal arrow shape to a "pick" cursor. Sometimes it is difficult to see the default pick cursor, so you can change it using the function

```
(set-pick-cursor-index i)
```

where *i* is an integer less than 256. The cursors can be viewed using an external X program:

```
xfd -fn cursor
```

3.4.3 Origin Marker

A yellow box called the "origin marker" marks the origin. It can be removed using:

```
(set-show-origin-marker 0)
```

Its state can be queried like this:

```
(show-origin-marker-state)
```

which returns an number (0 if it is not displayed, 1 if it is).

3.5 Screenshot

A simple screenshot (image dump) can be made using `Draw -> Screenshot -> Simple...`. Note that in side by side stereo mode you only get the left-hand image.

3.6 Raster3D output

Output suitable for use by Raster3D's "render" can be generated using the scripting function

`(raster3d` *`file-name`*`)`

where *`file-name`* is such as `"test.r3d"`[3].

There is a keyboard key to generate this file, run "render" and display the image: Function key F8.

You can also use the function

`(render-image)`

which will create a file '`coot.r3d`', from which "render" produces '`coot.png`'. This png file is displayed using ImageMagick's display program (by default). Use something like:

`(set! coot-png-display-program "gqview")`

to change that to different display program (`"gqview"` in this case).

`(set! coot-png-display-program "open")`

would use Preview (by default) on Macintosh.

To change the widths of the bonds and density "lines" use (for example):

`(set-raster3d-bond-thickness 0.1)`

and

`(set-raster3d-density-thickness 0.01)`

Similarly for bones:

`(set-raster3d-bone-thickness 0.05)`

To turn off the representations of the atoms (spheres):

`(set-renderer-show-atoms 0)`

3.7 Display Manager

This is also known as "Map and molecule (coordinates) display control". Here you can select which maps and molecules you can see and how they are drawn[4]. The "Display" and "Active" are toggle buttons, either depressed (active) or undepressed (inactive). The "Display" buttons control whether a molecule (or map) is drawn and the "Active" button controls if the molecule is clickable[5] (*i.e.* if the molecule's atoms can be labeled).

The `"Scroll"` radio buttons sets which map is has its contour level changed by scrolling the mouse scroll wheel.

By default, the path names of the files are not displayed in the Display Manager. To turn them on:

`(set-show-paths-in-display-manager 1)`

If you pull across the horizontal scrollbar in a Molecule view, you will see the "Render as" menu. You can use this to change between normal "Bonds (Colour by Atom)","Bonds (Colour by Chain)" and "Cα" representation There is also available "No Waters" and "Cα + ligands" representations.

[3] Povray support is only semi-working, there is a problem with the orientation of the image.

[4] to a limited extent.

[5] the substantial majority of the time you will want your the buttons to be both either depressed or undepressed, rarely one but not the other.

3.8 The Modelling Toolbar

You might not want to have the right-hand-side vertical toolbar that contains icons for some modelling operations[6] displayed:

`(hide-modelling-toolbar)`

to bring it back again:

`(show-modelling-toolbar)`

3.9 The file selector

3.9.1 File-name Filtering

The "Filter" button in the fileselection filters the filenames according to extension. For coordinates files the extensions are ".pdb" ".brk" ".mmcif" and others. For data: ".mtz", ".hkl", ".phs", ".cif" and for (CCP4) maps ".ext", ".msk" and ".map". If you want to add to the extensions, the following functions are available:

- `(add-coordinates-glob-extension` *`extension`*`)`
- `(add-data-glob-extension` *`extension`*`)`
- `(add-map-glob-extension` *`extension`*`)`
- `(add-dictionary-glob-extension` *`extension`*`)`

where *`extension`* is something like: `".mycif"`.

If you want the fileselection to be filtered without having to use the "Filter" button, use the scripting function

`(set-filter-fileselection-filenames 1)`

3.9.2 Filename Sorting

If you like your files initially sorted by date (rather than lexicographically, which is the default) use:

`(set-sticky-sort-by-date)`

3.9.3 Save Coordinates Directory

Some people prefer that the fileselection for saving coordinates starts in the original directory (rather than the directory from which they last imported coordinates). This option is for them:

`(set-save-coordinates-in-original-directory 1)`

3.10 Scripting

There is an compile-time option of adding a script interpreter. Currently the options are python and guile. It seems possible that in future you will be able to use both in the same executable. The binary distribution of Coot are linked with guile, others with python.

Hundreds of commands are made available for use in scripting by using SWIG, some of which are documented here. Other functions documented less well, but descriptions for them can be found at the end of this manual.

[6] British modelling, of course

Commands described throughout this manual (such as `(vt-surface 1))` can be evaluated directly by Coot by using the "Scripting Window" (`Calculate -> Scripting...`). Note that you type the commands in the upper entry widget and the command gets echoed (in red) and the return value and any output is displayed in the text widget lower (green). The typed command should be terminated with a carriage return[7]. Files[8] can be evaluated (executed) using `Calculate -> Run Script...`.

Note that in scheme (the usual scripting language of Coot), the parentheses are important.

To execute a script file from the command line use the `--script` *`filename`* arguments (except when also using the command line argument `--no-graphics`, in which case you should use `-s` *`filename`*).

After you have used the scripting window, you may have noticed that you can no longer kill Coot by using Ctrl-C in the console. To recover this ability:

`(exit)`

in the scripting window.

3.10.1 Python

Coot has an (optional) embedded python interpreter. Thus the full power of python is available to you. Coot will look for an initialization script (`$HOME/.coot.py`) and will execute it if found. This file should contain python commands that set your personal preferences.

3.10.1.1 Python Commands

The scripting functions described in this manual are formatted suitable for use with guile, *i.e.*:

`(`*`function arg1 arg2...`*`)`

If you are using Python instead: the format needs to be changed to:

`function``(`*`arg1,arg2...`*`)`

Note that dashes in guile function names become underscores for python, so that (for example) `(raster-screen-shot)` becomes `raster_screen_shot()`.

3.10.2 Scheme

The scheme interpreter is made available by embedding guile. The initialization script used by this interpreter is `$HOME/.coot`. This file should contain scheme commands that set your personal preferences.

3.10.3 Coot State

The "state" of Coot is saved on Exit and written to a file called `0-coot.state.scm` (scheme) `0-coot.state.py` (python). This state file contains information about the screen centre, the clipping, colour map rotation size, the symmetry radius, and other molecule related parameters such as filename, column labels, coordinate filename *etc.*.

[7] which causes the evaluation of the command.

[8] such as the Coot state file (Section Section 3.10.3 [Coot State], page 12).

Use `Calculate -> Run Script...` to use this file to re-create the loaded maps and models that you had when you finished using Coot[9] last time. A state file can be saved at any time using `(save-state)` which saves to file `0-coot.state.scm` or `(save-state-filename "thing.scm")` which saves to file `thing.scm`.

When Coot starts it can optionally run the commands in `0-coot.state.scm`.

Use `(set-run-state-file-status i)` to change the behaviour: `i` is `0` to never run this state file at startup, `i` is `1` to get a dialog option (this is the default) and `i` is `2` to run the commands without question.

3.10.4 Key Binding

"Power users" of Coot might like to write their own functions and bind that function to a keyboard key. How do they do that?

By using the add-key-binding function:

`(add-key-binding` *`function-name key function`*`)`

where *`key`* is a quoted string (note that upper case and lower case keys are distinguished - activate get upper case key binding you need to chord the shift key[10]).

for example:

```
(add-key-binding "Refine Active Residue with Auto-accept" "x" refine-
active-residue)
```

Have a look at the key bindings section on the Coot wiki for several more examples.

3.11 Backups and Undo

By default, each time a modification is made to a model, the old coordinates are written out[11]. The backups are kept in a backup directory and are tagged with the date and the history number (lower numbers are more ancient[12]). The "Undo" function discards the current molecule and loads itself from the most recent backup coordinates. Thus you do not have to remember to "Save Changes" - coot will do it for you[13].

If you have made changes to more than one molecule, Coot will pop-up a dialog box in which you should set the "Undo Molecule" *i.e.* the molecule to which the Undo operations will apply. Further Undo operations will continue to apply to this molecule until there are none left. If another Undo is requested Coot checks to see if there are other molecules that can be undone, if there is exactly one, then that molecule becomes the "Undo Molecule", if there are more than one, then another Undo selection dialog will be displayed.

You can set the undo molecule using the scripting function:

`(set-undo-molecule` *`imol`*`)`

---

[9] in that particular directory.

[10] funny that

[11] this might be initially surprising since this could chew up a lot of disk space. However, disk space is cheap compared to losing you molecule.

[12] The coordinates are written in pdb format - that's OK, isn't it?.

[13] unless you tell it not to, of course - use (*e.g.*) `(turn-off-backup 0)` to turn off the backup (for molecule 0 in this case).

If for reasons of strange system[14] requirements you want to remove the path components of the backup file name you can do so using:

```
(set-unpathed-backup-file-names 1)
```

3.11.1 Redo

The "undone" modifications can be re-done using this button. This is not available immediately after a modification[15].

3.11.2 Restoring from Backup

There may be certain circumstances[16] in which you wish to restore from a backup but can't get it by the "Undo" mechanism described above. In that case, start coot as normal and then open the (typically most recent) coordinates file in the directory `coot-backup` (or the directory pointed to the environment variable `COOT_BACKUP_DIR` if it was set) . This file should contain your most recent edits. In such a case, it is sensible for neatness purposes to immediately save the coordinates (probably to the current directory) so that you are not modifying a file in the backup directory.

See also Section Section 1.8 [Crash], page 4.

3.12 View Matrix

It is sometimes useful to use this to orient the view and export this orientation to other programs. The orientation matrix of the view can be displayed (in the console) using:

```
(view-matrix)
```

Also, the internal representation of the view can be returned and set using:

`(view-quaternion)` to return a 4-element list

`(set-view-quaternion i j k l)` which sets the view quaternion.

So the usage of these functions would be something like:

```
(let ((v (view-quaternion)))
   ;; manipulate v here, maybe
   (apply set-view-quaternion v))
```

3.13 Space Group and Symmetry

Occasionally you may want to know the space group of a particular molecule. Interactively (for maps) you can see it using the Map Properties button in the Molecule Display Control dialog.

There is a scripting interface function that returns the space group for a given molecule[17]:

`(show-spacegroup` *`imol`*`)`

You can force a space group onto a molecule using the following:

---

[14] or system manager.

[15] It works like the "Forwards" buttons in a web browser - which is not available immediately after viewing a new page.

[16] for example, if coot crashes.

[17] if no space group has been assigned it returns ``'No spacegroup for this molecule'`

`(set-space-group` *`imol space-group`*`)`

where *`space-group`* is one of the standard CCP4 space group names (*e.g.* `"P 21 21 21"`).

To show the symmetry operators of a particular molecule use: `(get-symmetry` *`imol`*`)` which will return a list of strings.

3.14 Recentring View

- Use Control + left-mouse to drag around the view
- or
- middle-mouse over an atom. In this case, you will often see "slide-recentring", the graphics smoothly changes between the current centre and the newly selected centre.
- or
- Use `Draw -> Go To Atom...` to select an atom using the keyboard. Note that you can subsequently use "Space" in the "graphics" window (OpenGL canvas) to recentre on the next Cα.
- or
- To centre on an arbitrary position (x,y,z), use the scripting function `(set-rotation-centre x y z)`.
- or
- Use the keyboard: [Ctrl G] then key in a residue number and (optionally) a chainid and press Return

If you don't want smooth recentring (sliding) `Edit -> Preferences -> Smooth Recentring -> Off`. You can also use this dialog to speed it up a bit (by decreasing the number of steps instead of turning it off).

3.15 Clipping Manipulation

The clipping planes (a.k.a. "slab") can be adjusted using `Edit -> Clipping` and adjusting the slider. There is only one parameter to change and it affects both the front and the back clipping planes[18]. The clipping can also be changed using keyboard "D" and "F".

One can "push" and "pull" the view in the screen-Z direction using keypad 3 and keypad "." (see Section Section 2.4 [Keyboard Z Translation], page 6).

3.16 Background colour

The background colour can be set either using a GUI dialog (`Edit$ -> Background Colour`) or the function `(set-background-colour 0.00 0.00 0.00)`, where the arguments are 3 numbers between 0.0 and 1.0, which respectively represent the red, green and blue components of the background colour. The default is (0.0, 0.0, 0.0) (black).

[18] I find a clipping level of about 3.5 to 4 comfortable for viewing electron density maps - it is a little "thinner" than the default startup thickness.

3.17 Unit Cell

If coordinates have symmetry available then unit cells can be drawn for molecules (`Draw -> Cell & Symmetry -> Show Unit Cell?`).

3.18 Rotation Centre Pointer

There is a pink pointer at the centre of the screen that marks the rotation centre. The size of the pointer can be changed using `Edit -> Pink Pointer Size...` or using scripting commands: `(set-rotation-centre-size 0.3)`.

3.19 Pointer Distances

The Rotation Centre Pointer is sometimes called simply "Pointer". One can find distances to the pointer from any active set of atoms using "Pointer Distances" (under Measures). If you move the Pointer (*e.g.* by centering on an atom) and want to update the distances to it, you have to toggle off and on the "Show Pointer Distances" on the Pointer Distances dialog.

3.20 Crosshairs

Crosshairs can be drawn at the centre of the screen, using either the C key[19] in graphics window or `Draw -> Crosshairs...`. The ticks are at 1.54Å, 2.7Å and 3.8Å.

3.21 3D Annotations

Positions in 3D space can be annotated with 3D text. The mechanism to do this can be found under Extensions -> Representations -> 3D Annotations. 3D Annotations can be saved to and loaded from a file.

3.22 Frame Rate

Sometimes, you might as yourself "how fast is the computer?"[20]. Using `Calculate -> Frames/Sec` you can see how fast the molecule is rotating, giving an indication of graphics performance. It is often better to use a map that is more realistic and stop the picture whizzing round. The output is written to the status bar and the console, you need to give it a few seconds to "settle down". It is best not to have other widgets overlaying the GL canvas as you do this.

The contouring elapsed time[21] gives an indication of CPU performance.

3.23 Program Output

Due to its "in development" nature (at the moment), Coot produces a lot of "console"[22] output - much of it debugging or "informational". This will go away in due course. You are advised to run Coot so that you can see the console and the graphics window at the

[19] and C again to toggle them off.

[20] compared to some other one.

[21] prompted by changing the contour level.

[22] *i.e.* the terminal in which you started Coot.

same time, since feedback from atom clicking (for example) is often written there rather than displayed in the graphics window.

- Output that starts "ERROR..." is a programming problem (and ideally, you should never see it).
- Output that starts "WARNING..." means that something probably unintended happened due to the unexpected nature of your input or file(s).
- Output that starts "DEBUG..." has (obviously enough) been added to aid debugging. Most of them should have been cleaned up before release, but as Coot is constantly being developed, a few may slip through. Just ignore them.

4 Coordinate-Related Features

4.1 Reading coordinates

The format of coordinates that can be read by coot is either PDB or mmCIF. To read coordinates, choose `File -> Read Coordinates` from the menu-bar. Immediately after the coordinates have been read, the view is (by default) recentred to the centre of this new molecule and the molecule is displayed. The recentring of the view after the coordinates have been read can be turned off by unclicking the "Recentre?" radio-button.

To disable the recentring of the view on reading a coordinates file via scripting, use: `(set-recentre-on-read-pdb 0)`. However, when reading a coordinates file from a script it is just as good (if not better) to use `(handle-read-draw-molecule-with-recentre` *`filename`* `0)` - the additional `0` means "don't recentre". And that affects just the reading of *`filename`* and not subsequent files.

4.1.1 A Note on Space Groups Names

Coot uses the space group on the "CRYST1" line of the pdb file. The space group should be one of the xHM symbols listed (for example) in the CCP4 dictionary file '`syminfo.lib`'. So, for example, `"R 3 2 :H"` should be used in preference to `"H32"`.

4.1.2 Read multiple coordinate files

The reading multiple files using the GUI is not available (at the moment). However the following scripting functions are available:

`(read-pdb-all)`

which reads all the "*.pdb" files in the current directory

`(multi-read-pdb` *`glob-pattern dir`*`)`

which reads all the files matching *`glob-pattern`* in directory *`dir`*. Typical usage of this might be:

`(multi-read-pdb "a*.pdb" ".")`

Alternatively you can specify the files to be opened on the command line when you start coot (see Section Section 1.6 [Command Line Arguments], page 3).

4.1.3 SHELX .ins/.res files

SHELX ".res" (and ".ins" of course) files can be read into Coot, either using the GUI `File -> Open Coordinates...` or by the scripting function:

`(read-shelx-ins-file` *`file-name`*`)`

where *`file-name`* is quoted, such as `"thox.ins"`.

Although Coot should be able to read any SHELX ".res" file, it may currently have trouble displaying the bonds for centro-symmetric structures.

ShelxL atoms with negative `PART` numbers are given alternative configuration identifiers in lower case.

To write a SHELX ".ins" file:

`(write-shelx-ins-file` *`imol file-name`*`)`

where *imol* is the number of the molecule you wish to export.

This will be a rudimentary file if the coordinates were initially from a "PDB" file, but will contain substantial SHELX commands if the coordinates were initially generated from a SHELX `ins` file.

4.2 Atom Info

Information about about a particular atom is displayed in the text console when you click using middle-mouse. Information for all the atoms in a residue is available using `Info -> Residue Info...`.

The temperature factors and occupancy of the atoms in a residue can be set by using `Edit -> Residue Info...`.

4.3 Atom Labeling

Use Shift + left-mouse to label atom. Do the same to toggle off the label. The font size is changeable using `Edit -> Font Size...`. The newly centred atom is labelled by default. To turn this off use:

```
(set-label-on-recentre-flag 0)
```

Some people prefer to have atom labels that are shorter, without the slashes and residue name:

```
(set-brief-atom-labels 1)
```

To change the atom label colour, use:

```
(set-font-colour 0.9 0.9 0.9)
```

4.4 Atom Colouring

The atom colouring system in coot is unsophisticated. Typically, atoms are coloured by element: carbons are yellow, oxygens red, nitrogens blue, hydrogens white and everything else green (see Section Section 3.7 [Display Manager], page 10 for colour by chain). However, it is useful to be able to distinguish different molecules by colour, so by default coot rotates the colour map of the atoms (*i.e.* changes the H value in the HSV[1] colour system). The amount of the rotation depends on the molecule number and a user-settable parameter:

- `(set-colour-map-rotation-on-read-pdb 30)`.

The default value is 31°.

Also one is able to select only the Carbon atoms to change colour in this manner: `(set-colour-map-rotation-on-read-pdb-c-only-flag 1)`.

The colour map rotation can be set individually for each molecule by using the GUI: `Edit -> Bond Colours...`.

4.5 Bond Parameters

The various bond parameters can be set using the GUI dialog `Draw -> Bond Parameters` or *via* scripting functions.

[1] Hue Saturation Value (Intensity).

4.5.1 Bond Thickness

The thickness (width) of bonds of individual molecules can be changed. This can be done via the `Bond Parameters` dialog or the scripting interface:

`(set-bond-thickness` *`thickness imol`*`)`

where `imol` is the molecule number.

The default thickness is 3 pixels. The bond thickness also applies to the symmetry atoms of the molecule. The default bond thickness for new molecules can be set using:

`(set-default-bond-thickness` *`thick`*`)`

where *`thick`* is an integer.

There is no means to change the bond thickness of a residue selection within a molecule.

4.5.2 Display Hydrogens

Initially, hydrogens are displayed. They can be undisplayed using

`(set-draw-hydrogens` *`mol-no`* `0)`[2]

where *`mol-no`* is the molecule number.

There is a GUI to control this too, under "Edit -> Bond Parameters".

4.5.3 NCS Ghosts Coordinates

It is occasionally useful when analysing non-crystallographically related molecules to have "images" of the other related molecules appear matched onto the current coordinates. It is important to understand that these ghosts are for displaying differences of NCS-related molecules by structure superposition, not displaying neighbouring NCS related molecules. As you read in coordinates in Coot, they are checked for NCS relationships and clicking on "Edit -> Bond Parameters -> Show NCS Ghosts" -> "Yes" -> "Apply" will create "ghost" copies of them over the reference chain[3].

Sometimes SSM does not provide a good (or even useful) matrix. In that case, we can specify the residue range ourselves and let the LSQ algorithm provide the matrix. A gui dialog for this operation can be found under `Extensions -> NCS -> NCS Ghosts by Residue Range...`.

The scripting function is used like this:

`(manual-ncs-ghosts` *`imol resno-start resno-end ncs-chain-ids`*`)`

Typical usage: `(manual-ncs-ghosts 0 1 10 (list "A" "B" "C"))`

note that in *`ncs-chain-ids`*, the NCS master/reference chain-id goes first.

4.5.4 NCS Maps

Coot can use the relative transformations of the NCS-related molecules in a coordinates molecule to transform maps. Use `Calculate -> NCS Maps...` to do this (note the NCS maps only make sense in the region of the reference chain (see above).

Note also that the internal representation of the map is not transformed. If you try to export a NCS overlay map you will get an untransformed map. A transformed map only

[2] they can be redisplayed using `(set-draw-hydrogens` *`mol-no`* `1)`.

[3] the reference chain is, by default, the first chain of that type in the coordinates file. The reference (master) chain can be changed using the NCS Ghosts Control dialog.

makes sense around a given point (and when using transformed maps in Coot, this reference point is changed on the fly, thus allowing map transformations on the fly). [This applies to NCS overlap maps, NCS averaged maps are transformed].

This will also create an NCS averaged map[4].

4.5.5 Using Strict NCS

Coot can use a set of strict NCS matrices to specify NCS which means that NCS-related molecules can appear like convention symmetry-related molecules.

`(add-strict-ncs-matrix` *`imol ncs-chain-id ncs-target-chain-id m11 m12 m13 m21 m22 m23 m31 m32 m33 t1 t2 t3`*`)`

where *`ncs-chain-id`* might be "B", "C" "D" (etc.) and *`ncs-target-chain-id`* is "A", i.e. the B, C, D molecules are NCS copies of the A chain.

for icosahedral symmetry the translation components *`t1`*, *`t2`*, *`t3`* will be 0.

You need to turn on symmetry for molecule *`imol`* and set the displayed symmetry object type to "Display Near Chains".

4.6 Download coordinates

Coot provides the possibility to download coordinates from an OCA[5]. (*e.g.* EBI) server[6] (`File -> Get PDB Using Code...`). A pop-up entry box is displayed into which you can type a PDB accession code. Coot will then connect to the web server and transfer the file. Coot blocks as it does this (which is not ideal) but on a semi-decent internet connection, it's not too bad. The downloaded coordinates are saved into a directory called '`coot-download`'.

It is also possible to download mmCIF data and generate a map. This currently requires a properly formatted database structure factors mmCIF file[7].

4.7 Get Coordinates and Map from EDS

Using this function we have the ability to download coordinates and view the map from structures in the Electron Density Server (EDS) at Uppsala University. This is a much more robust and faster way to see maps from deposited structures. This function can be found under the File menu item.

This feature was added with the assistance of Gerard Kleywegt. If you use the EDS, please cite GJ Kleywegt, MR Harris, JY Zou, TC Taylor, A Wählby & TA Jones (2004), "*The Uppsala Electron-Density Server*", *Acta Cryst.* **D**60, 2240-2249.

4.8 Save Coordinates

On selecting from the menus `File -> Save Coordinates...` you are first presented with a list of molecules which have coordinates. As well as the molecule number, there is the molecule name - very frequently the name of the file that was read in to generate the

[4] that also only makes sense in the region of the reference chain.

[5] OCA is "goose" in Spanish (and Italian)

[6] the default is the Weizmann Institute - which for reasons I won't go into here is currently much faster than the EBI server.

[7] which (currently) only a fraction are.

coordinates in coot initially. However, this is only a *molecule* name and should not be confused with the filename to which the coordinates are saved. The coordinates *filename* can be selected using the `Select Filename...` button.

If your filename ends in `.cif`, `.mmcif` or `.mmCIF` then an mmCIF file will be written (not a "PDB" file).

4.9 Setting the Space Group

If for some reason, the pdb file that you read does not have a space group, or has the wrong space group, then you can set it using the following function:

`(set-space-group` *imol symbol*`)`

e.g.:

```
(set-space-group 0 "P 41 21 2")
```

4.10 Anisotropic Atoms

By default anisotropic atom information is not represented[8]. To turn them on, use `Draw -> Anisotropic Atoms -> Show Anisotropic Atoms? -> Yes`, or the command: `(set-show-aniso 1)`.

You cannot currently display thermal ellipsoids[9] for isotropic atoms.

4.11 Symmetry

Coordinates symmetry is "dynamic". Symmetry atoms can be labeled[10]. Every time you recentre, the symmetry coordinates are updated. The information shown contains the atom information and the symmetry operation number and translations needed to generate the atom in that position.

By default symmetry atoms are not displayed.

If you want coot to display symmetry coordinates without having to use the gui, add to your '`~/.coot`' the following:

```
(set-show-symmetry-master 1)
```

The symmetry can be represented as Cαs. This along with representation of the molecule as Cαs (Section Section 3.7 [Display Manager], page 10) allow the production of a packing diagram.

4.11.1 Missing symmetry

Sometimes (rarely) coot misses symmetry-related molecules that should be displayed. In that case you need to expand the shift search (the default is 1):

```
(set-symmetry-shift-search-size 2)
```

This is a hack, until the symmetry search algorithm is improved.

---

[8] using thermal ellipsoids

[9] in the case of isotropic atoms, ellipsoids are spherical, of course.

[10] symmetry labels are in pale blue and also provide the symmetry operator number and the translations along the a, b and c axes.

4.12 Sequence View

The protein is represented by one letter codes and coloured according to secondary structure. These one letter codes are active - if you click on them, they will change the centre of the graphics window - in much the same way as clicking on a residue in the Ramachandran plot.

4.13 Print Sequence

The single letter code (of the *imol*th molecule) is written out to the console in FASTA format. Use can use this to cut and paste into other applications:

`(print-sequence` *imol*`)`

4.14 Environment Distances

Environment distances are turned on using `Info -> Environment Distances...`. Contacts to other residues are shown and to symmetry-related atoms if symmetry is being displayed. The contacts are coloured by atom type[11].

4.15 Distances and Angles

The distance between atoms can be found using `Info -> Distance`[12]. The result is displayed graphically, and written to the console.

4.16 Zero Occupancy Marker

Atoms of zero occupancy are marked with a grey spot. To turn off these markers, use:

`(set-draw-zero-occ-markers 0)`

Use an argument of 1 to turn them on.

4.17 Atomic Dots

You can draw dots round arbitrary atom selections

`(dots` *imol atom-selection dot-density radius*`)` The function returns a handle.

e.g. put a sphere of dots around all atoms of the 0th molecule (it might be a set of heavy atom coordinates) at the default dot density and radius:

`(dots 0 "/1" 1 1)`

You can't change the colour of the dots.

There is no internal mechanism to change the radius according to atom type. With some cleverness you might be able to call this function several times and change the radius according to the atom selection.

There is a function to clear up the dots for a particular molecule *imol* and dots set identifier *dots-handle*

`(clear-dots` *imol dots-handle*`)`

[11] contacts not involving a carbon atom are yellow.

[12] Use `Angle` for an angle, of course.

There is a function to return how many dots sets there are for a particular molecule *imol*:

`(n-dots-set imol)`

4.18 Ball and Stick Representation

Fragments of the molecule can be rendered as a "ball and stick" molecule:

`(make-ball-and-stick imol atom-selection bond-thickness sphere-size draw-spheres-flag)`

e.g. `(make-ball-and-stick 0 "/1/A/10-20" 0.3 0.4 1)`

The ball-and-stick representation can be cleared using:

`(clear-ball-and-stick imol)`

4.19 Mean, Median Temperature Factors

Coot can be used to calculate the mean (average) and median temperatures factors:

`(average-temperature-factor imol)`

`(median-temperature-factor imol)`

−1 is returned if there was a problem[13].

4.20 Secondary Structure Matching (SSM)

The excellent SSM alogrithm[14] of Eugene Krissinel is available in Coot. The GUI interface is straight-forward and can be found under `Calculate -> SSM Superpose`. You can specify the specific chains that you wish to match using the "Use Specific Chain" check-button.

There is a scripting level function which gives even finer control:

`(superpose-with-atom-selection imol1 imol2 mmdb-atom-selection-string-1 mmdb-atom-selection-string-2 move-copy-flag )`

the *move-copy-flag* should be 1 if you want to apply the transformation to a copy of *imol2* (rather than *imol2* itself). Otherwise, *move-copy-flag* should be 0.

mmdb atom selection strings (Coordinate-IDs) are explained in detail in the mmdb manual.

Briefly, the string should be formed in this manner:

`/mdl/chn/seq(res).ic/atm[elm]:aloc`

e.g. `"/1/A/12-130/CA"`

4.21 Least-Squares Fitting

There is a simple GUI for this `Calculate -> LSQ Superpose...`

The scripting interface to LSQ fitting is as follows:

`(simple-lsq-match ref-start-resno ref-end-resno ref-chain-id imol-ref mov-start-resno mov-end-resno mov-chain-id imol-mov match-type)`

where:

[13] *e.g.* this molecule was a map or a closed molecule.

[14] the same one as in the CCP4 program SUPERPOSE

- *ref-start-resno* is the starting residue number of the reference molecule
- *ref-end-resno* is the last residue number of the reference molecule
- *mov-start-resno* is the starting residue number of the moving molecule
- *mov-end-resno* is the last residue number of the moving molecule
- *match-type* is one of `'CA`, `'main`, or `'all`.

e.g.: `(simple-lsq-match 940 950 "A" 0 940 950 "A" 1 'main)`

More sophisticated (match molecule number 1 chain "B" on to molecule number 0 chain "A"):

```
(define match1 (list 840 850 "A" 440 450 "B" 'all))
(define match2 (list 940 950 "A" 540 550 "B" 'main))
(clear-lsq-matches)
(set-match-element match1)
(set-match-element match2)
(lsq-match 0 1) ; match molecule number 1 onto molecule number 0.
```

4.22 Ligand Overlaying

The scripting function

`(overlap-ligands` *imol-ligand imol-ref chain-id-ref resno-ref*`)`

returns a rotation+translation operator which can be applied to other molecules (and maps). Here, *imol-ligand* is the molecule number of the ligand (which is presumed to be a a molecule on its own - Coot simply takes the first residue that it finds). *imol-ref chain-id-ref resno-ref* collectively describe the target position for the moving *imol-ligand* molecule.

The convenience function

`(overlay-my-ligands` *imol-mov chain-id-mov resno-mov imol-ref chain-id-ref resno-ref*`)`

wraps `overlap-ligands`. The is no GUI for this function yet.

4.23 Writing PDB files

As well as the GUI option `File -> Save Coordinates...` there is a scripting options available:

`(write-pdb-file` *imol pdb-file-name*`)`

which writes the *imol*th coordinates molecule to *filename*.

To write a specific residue range:

`(write-residue-range-to-pdb-file` *imol chain-id start-resno endresno pdb-file-name*`)`

5 Modelling and Building

The functions described in this chapter manipulate, extend or build molecules and can be found under `Calculate -> Model/Fit/Refine...`. When activated, the dialog "stays on top" of the main graphics window[1]. Some people think that this is not always desirable, so this behaviour can be undone using:

```
(set-model-fit-refine-dialog-stays-on-top 0)
```

5.1 Regularization and Real Space Refinement

Coot will read the geometry restraints for refmac and use them in fragment (zone) idealization - this is called "Regularization". The geometrical restraints are, by default, bonds, angles, planes and non-bonded contacts. You can additionally use torsion restraints by `Calculate -> Model/Fit/Refine... -> Refine/Regularize Control -> Use Torsion Restraints`. Truth to tell, this has not been successful in my hands (sadly).

"RS (Real Space) Refinement" (after Diamond, 1971[2]) in Coot is the use of the map in addition to geometry terms to improve the positions of the atoms. Select "Regularize" from the "Model/Fit/Refine" dialog and click on 2 atoms to define the zone (you can of course click on the same atom twice if you only want to regularize one residue). Coot then regularizes the residue range. At the end Coot, displays the intermediate atoms in white and also displays a dialog, in which you can accept or reject this regularization. In the console are displayed the χ^2 values of the various geometrical restraints for the zone before and after the regularization. Usually the χ^2 values are considerably decreased - structure idealization such as this should drive the χ^2 values toward zero.

The use of "Refinement" is similar - with the addition of using a map. The map used to refine the structure is set by using the "Refine/Regularize Control" dialog. If you have read/created only one map into Coot, then that map will be used (there is no need to set it explicitly).

Use, for example, `(set-matrix 20.0)`

to change the weight of the map gradients to geometric gradients. The higher the number the more weight that is given to the map terms[3]. The default is 60.0. This will be needed for maps generated from data not on (or close to) the absolute scale or maps that have been scaled (for example so that the sigma level has been scaled to 1.0).

For both "Regularize Zone" and "Refine Zone" one is able to use a single click to refine a residue range. Pressing A on the keyboard while selecting an atom in a residue will automatically create a residue range with that residue in the middle. By default the zone is extended one residue either size of the central residue. This can be changed to 2 either side using `(set-refine-auto-range-step 2)`.

Intermediate (white) atoms can be moved around with the mouse (click and drag with left-mouse, by default). Refinement will proceed from the new atom positions when the mouse button is released. It is possible to create incorrect atom nomenclature and/or chiral

[1] given a half-decent window manager

[2] Diamond, R. (1971). A Real-Space Refinement Procedure for Proteins. *Acta Crystallographica* **A**27, 436-452.

[3] but the resulting χ^2 values are higher.

volumes in this manner - so some care must be taken. Press the A key as you left-mouse click to move atoms more "locally" (rather than a linear shear) and CTRL key as you left-mouse click to move just one atom.

In more up to date versions, Coot will display colour patches (something like a traffic light system) representing the chi squared values of each of types of geometric feature refined. Typically "5 greens" is the thing to aim for, the colour changes occurring at chi squared values 2, 5 and 8 (8 being the most red).

To prevent the unintentional refinement of a large number of residues, there is a "heuristic fencepost" of 20 residues. A selection of than 20 residues will not be regularized or refined. The limit can be changed using the scripting function: *e.g.* `(set-refine-max-residues 30)`.

5.1.1 Dictionary

The geometry description for residues, monomers and links used by Coot are in the standard mmCIF format. Because this format alows multiple comp_ids (residue types) to be described within a cif loop, it is hard to tell when a dictionary entry needs to be overwritten when reading a new file. Therefore Coot makes this extra constraint: that the "chem_comp" loop should appear first in the comp list data item - if this is the case, then Coot can overwrite an old restraint table for a particular comp_id/residue-type when a new one is read.

By default, the geometry dictionary entries for only the standard residues are read in at the start[4]. It may be that your particular ligand is not amongst these. To interactively add a dictionary entry use `File -> Import CIF Dictionary`. Alternatively, you can use the function:

`(read-cif-dictionary` *`filename`*`)`

and add this to your `.coot` file (this may be the preferred method if you want to read the file on more than one occasion).

Note: the dictionary also provides the description of the ligand's torsions.

5.1.2 Sphere Refinement

Sphere refinement selects residues within a certain distance of the residue at the centre of the screen and includes them for real space refinement. In this way, one can select residues that are not in a linear range. This technique is useful for refining disulfide bonds and glycosidic linkages.

The following adds a key binding (Shift-R) that refines resides that are within 3.5Å of the residue at the centre of the screen:

```
(add-key-binding "Refine residues in a sphere" "R"
    (lambda ()
      (let ((active-atom (active-residue)))
        (if (list? active-atom))
            (let* ((centred-residue (list-head (cdr active-atom) 3))
                   (imol (car active-atom))
                   (other-residues (residues-near-residue
                                    imol centred-residue 3.5))
```

[4] And a few extras, such as phosphate

```
            (all-residues (if (list? other-residues)
                              (cons centred-residue other-residues)
                              (list centred-residue))))
      (refine-residues imol all-residues))))))
```

5.1.3 Refining Carbohydrates

Refining carbohydrates monomers should be as straightforward as refining a protein residue. Coot will look in the dictionary for the 3-letter code for the particular residue type, if it does not find it, Coot will try to search for dictionary files using "-b-D" or "-a-L" extensions.

When refining a group of carbohydrates, the situation needs a bit more explanation. For each residue pair with tandem residue numbers specified in the refinement range selection, Coot checks if these residue types are are furanose or pyranose in the dictionary, and if the are both one or the other, then it tries to see if there are any of the 11 link types (BETA1-4, BETA2-3, ALPHA1-2 and so on) specified in the dictionary. It does this by a distance check of the potentially bonding atoms. If the distance is less than 3.0Å, then a glycosidic bond is made and used in the refinement.

Bonds between protein and carbohydrate and branched carbohydrates can be refined using "Sphere Refinement".

LINK and LNKR cards are not used to determine the geometry of the restraints.

5.1.4 Planar Peptide Restraints

By default, Coot uses a 5 atom (CA-1, C-1, O-1, N-2, CA-2) planar peptide restraints. These restraints should help in low resolution fitting (the main-chains becomes less distorted), reduce accidental cis-peptides and may help "clean up" Ramachandran plots.

```
(add-planar-peptide-restraints)
```

And similarly they can be removed:

```
(remove-planar-peptide-restraints)
```

There is also a GUI to add and remove these restraints in `Extensions -> Refine... -> Peptide Restraints...`

5.1.5 The UNK residue type

The `UNK` residue type is a special residue type to Coot. It has been added for use with Buccaneer. Don't give you ligand (or anything else) the 3-letter-code `UNK` or confusion will result[5].

5.1.6 Moving Zero Occupancy Atoms

By default, atoms with zero occupancy are moved when refining and regularizing. This can sometimes be inconvenient. To turn of the movement of atoms with zero occupancy when refining and regularizing:

```
(set-refinement-move-atoms-with-zero-occupancy 0)
```

[5] unless you are using Buccaneer, of course

5.2 Changing the Map for Building/Refinement

You can change the map that is used for the fitting and refinement tools using the `Select Map...` button on the Model/Fit/Refine dialog.

5.3 Rotate/Translate Zone

"Rotate/Translate Zone" from the "Model/Fit/Refine" menu allows manual movement of a zone. After pressing the "Rotate/Translate Zone" button, select two atoms in the graphics canvas to define a residue range[6], the second atom that you click will be the local rotation centre for the zone. The atoms selected in the moving fragment have the same alternate conformation code as the first atom you click. To actuate a transformation, click and drag horizontally across the relevant button in the newly-created "Rotation & Translation" dialog. The axis system of the rotations and translations are the screen coordinates. Alternatively[7], you can click using left-mouse on an atom in the fragment and drag the fragment around. Use Control Left-mouse to move just one atom, rather than the whole fragment. If you click Control Left-mouse whilst *not* over an atom then you can rotate the fragment using mouse drag. Click "OK" (or press Return) when the transformation is complete.

To change the rotation point to the centre of the intermediate atoms (rather than the second clicked atom), use the setting:

```
(set-rotate-translate-zone-rotates-about-zone-centre 1)
```

5.4 Rigid Body Refinement

"Rigid Body Fit Zone" from the "Model/Fit/Refine" dialog provides rigid body refinement. The selection is zone-based[8]. So to refine just one residue, click on one atom twice.

Sometimes no results are displayed after Rigid Body Fit Zone. This is because the final model positions had too many final atom positions in negative density. If you want to over-rule the default fraction of atoms in the zone that have an acceptable fit (0.75), to be (say) 0.25:

```
(set-rigid-body-fit-acceptable-fit-fraction 0.25)
```

5.5 Simplex Refinement

Rigid body refinement via Nelder-Mead Simplex minimization is available in Coot. Simplex refinement has a larger radius of convergence and thus is useful in a position where simple rigid body refinement finds the wrong minimum. However the Simplex algorithm is much slower. Simplex refinement for a residue range *start-resno* to *end-resno* (inclusive) in chain *chain-id* can be accessed as follows:

```
(fit-residue-range-to-map-by-simplex start-resno end-resno alt-loc
chain-id imol imol-for-map)
```

There is currently no GUI interface to Simplex refinement.

---

[6] if you want to move only one residue, then click the same atom twice.

[7] like Refinement and Regularization

[8] like Regularization and Refinement.

5.6 Post-manipulation-hook

If you wanted automatically run a function after a model has been manipulated then you can do so using by creating a function that takes 2 arguments, such as:

`(post-manipulation-hook` *`imol manipulation-mode`*`)`

`manipulation-mode` is one of *`(DELETED)`*, *`(MUTATED)`* or *`(MOVINGATOMS)`*.

And of course *`imol`* is the model number of the maniplated molecule.

(It would of course be far more useful if this function was also passed a list of residues - that is something for the future).

5.7 Baton Building

Baton build is most useful if a skeleton is already calculated and displayed (see Section Section 6.14 [Skeletonization], page 50). When three or more atoms have been built in a chain, Coot will use a prior probability distribution for the next position based on the position of the previous three. The analysis is similar to that of Oldfield & Hubbard (1994)[9], however it is based on a more recent and considerably larger database.

Little crosses are drawn representing directions in which is is possible that the chain goes, and a baton is drawn from the current point to one of these new positions. If you don't like this particular direction[10], use `Try Another`. The list of directions is scored according to the above criterion and sorted so that the most likely is at the top of the list and displayed first as the baton direction.

When starting baton building, be sure to be about 3.8Å from the position of the first-placed Cα, this is because the next Cα is placed at the end of the baton, the baton root being at the centre of the screen. So, when trying to baton-build a chain starting at residue 1, centre the screen at about the position of residue 2.

It seems like a good idea to increase the map sampling to 2 or even 2.5 (before reading in your mtz file) [a grid sampling of about 0.5Å seems reasonable] when trying to baton-build a low resolution map. You can set the map sampling using `Edit -> Map Parameters -> Map Sampling`.

Occasionally, every point is not where you want to position the next atom. In that case you can either shorten or lengthen the baton, or position it yourself using the mouse. Use "b" on the keyboard to swap to baton mode for the mouse[11].

Baton-built atoms are placed into a molecule called "Baton Atom" and it is often sensible to save the coordinates of this molecule before quitting coot.

If you try to trace a high resolution map (1.5Å or better) you will need to increase the skeleton search depth from the default (10), for example:

`(set-max-skeleton-search-depth 20)`

Alternatively, you could generate a new map using data to a more moderate resolution (2Å), the map may be easier to interpret at that resolution anyhow[12].

---

[9] T. J. Oldfield & R. E. Hubbard (1994). "Analysis of Cα Geometry in Protein Structures" *Proteins-Structure Function and Genetics* **18(4)** 324 – 337.

[10] which is quite likely at first since coot has no knowledge of where the chain has been and cannot score according to geometric criteria.

[11] "b" again toggles the mode off.

[12] high-resolution map interpretation is planned.

The guide positions are updated every time the "Accept" button is clicked. The molecule name for these atoms is "Baton Build Guide Points" and is is not usually necessary to keep them.

5.7.1 Undo

There is also an "Undo" button for baton-building. Pressing this will delete the most recently placed Cα and the guide points will be recalculated for the previous position. The number of "Undo"s is unlimited. Note that you should use the "Undo" button in the Baton Build dialog, not the one in the "Model/Fit/Refine" dialog (Section Section 3.11 [Backups and Undo], page 13).

5.7.2 Missing Skeleton

Sometimes (especially at loops) you can see the direction in which the chain should go, but there is no skeleton (see Section Section 6.14 [Skeletonization], page 50) is displayed (and consequently no guide points) in that direction. In that case, "Undo" the previous atom and decrease the skeletonization level (`Edit -> Skeleton Parameters -> Skeletonization Level`). Accept the atom (in the same place as last time) and now when the new guide points are displayed, there should be an option to build in a new direction.

5.7.3 Building Backwards

The following scenario is not uncommon: you find a nice stretch of density and start baton building in it. After a while you come to a point where you stop (dismissing the baton build dialog). You want to go back to where you started and build the other way. How do you do that?

- Use the command:

  `(set-baton-build-params` *`start-resno chain-id`* `"backwards")`

  where *`start-resno`* would typically be 0[13] and *`chain-id`* would be `""` (default).
- Recentre the graphics window on the first atom of the just-build fragment
- Select "Ca Baton Mode" and select a baton direction that goes in the "opposite" direction to what is typically residue 2. This is slightly awkward because the initial baton atoms build in the "opposite" direction are not dependent on the first few atoms of the previously build fragment.

5.8 Reversing Direction of Fragment

After you've build a fragment, sometimes you might want to change the direction of that fragment (this function changes an already existing fragment, as opposed to Backwards Building which sets up Baton Building to place new points in reverse order).

The fragment is defined as a contiguous set of residues numbers. So that you should be sure that other partial fragments which have the same chain id and that are not connected to this fragment have residue numbers that are not contiguous with the fragment you are trying to reverse.

[13] *i.e.* one less than the starting residue in the forward direction (defaults to 1).

5.9 C\alpha -> Mainchain

Mainchain can be generated using a set of Cαs as guide-points (such as those from Baton-building) along the line of Esnouf[14] or Jones and coworkers[15]. Briefly, 6-residue fragments of are generated from a list of high-quality[16] structures. The Cα atoms of these fragments are matched against overlapping sets of the guide-point Cαs. The resulting matches are merged to provide positions for the mainchain (and Cβ) atoms. This procedure works well for helices and strands, but less well[17] for less common structural features.

This function is also available from the scripting interface:

`(db-mainchain imol chain-id resno-start resno-end direction)`

where direction is either `"backwards"` or `"forwards"`.

Recall that the *`chain-id`* needs to be quoted, *i.e.* use `"A"` not `A`. Note that *`chain-id`* is `""` when the Cαs have been built with Baton Mode in Coot.

5.10 Backbone Torsion Angles

It is possible to edit the backbone ϕ and ψ angles indirectly using an option in the Model/Fit/Refine's dialog: "Edit Backbone Torsions..". When clicked and an atom of a peptide is selected, this produces a new dialog that offers "Rotate Peptide" which changes this residues ψ and "Rotate Carbonyl" which changes ϕ. Click and drag across the button[18] to rotate the moving atoms in the graphics window. You should know, of course, that making these modifications alter the ϕ/ψ angles of more than one residue.

5.11 Docking Sidechains

Docking sidechains means adding sidechains to a model or fragment that has currently only poly-Ala, where the sequence assignment is unknown. The algorithm is basically the same as in Cowtan's Buccaneer, but with some corners cut to make things (more or less) interactive. The algorithm uses the shape of the density around the C-beta position to estimate the probability of each sidechain type at that position.

The function is accessed via the `Extensions -> Dock Sequence` menu item. First, a sequence should be assigned from a PIR file to a particular chain-id and model number. Secondly `Extensions -> Dock Sequence -> Dock Sequence on this fragment...`. Choose the model to build on and then `Dock Sequence!` If all goes well, the model will be updated with mutated residues and undergo rotamer seach for each of the new residues. If the sequence alignment is not sufficiently clear, then you will get a dialog suggesting that you extend or improve the fragment.

---

[14] R. M. Esnouf "Polyalanine Reconstruction from Cα Positions Using the Program *CALPHA* Can Aid Initial Phasing of Data by Molecular Replacement Procedures" *Acta Cryst.* , D**53**, 666-672 (1997).

[15] T.A. Jones & S. Thirup "Using known substructures in protein model building and crystallography" *EMBO J.* **5**, 819–822 (1986).

[16] and high resolution

[17] *i.e.* there are severely misplaced atoms

[18] as for Rotate/Translate Zone (Section Section 5.3 [Rotate/Translate Zone], page 29).

5.12 Rotamers

The rotamers are generated[19] from the backbone independent sidechain library of the Richardsons group[20].

The m, t and p stand for "minus (-60)", "trans (180)" and "plus (+60)". There is one letter per χ angle.

Use keyboard . and , to cycle round the rotamers.

5.12.1 Auto Fit Rotamer

"Auto Fit Rotamer" will try to fit the rotamer to the electron density. Each rotamer is generated, rigid body refined and scored according to the fit to the map. Fitting the second conformation of a dual conformation in this way will often fail - the algorithm will pick the best fit to the density - ignoring the position of the other atoms.

The algorithm doesn't know if the other atoms in the structure are in sensible positions. If they are, then it is sensible not to put this residue too close to them, if they are not then there should be no restriction from the other atoms as to the position of this residue - the default is "are sensible", which means that the algorithm is prevented from finding solutions that are too close to the atoms of other residues. `(set-rotamer-check-clashes 0)` will stop this.

By default, Auto Fit Rotamer will switch to "Backrub Rotamer"[21] mode when fitting against a map of worse than 2.7Å. This search mode moves the some atoms of the mainchain of the neighbouring residues. After rotation of the central residue and neighbouring atoms around the "backrub vector", the individual peptides are back-rotated (along the peptide axis) so that the carbonyl oxygen are placed as near as possible to their original position. The Ramachandran plot is not used in this fitting algorithm.

There is a scripting interface to auto-fitting rotamers:

`(auto-fit-best-rotamer` *`resno alt-loc ins-code chain-id imol-coords imol-map clash-flag lowest-rotamer-probability`*`)`

where:

`resno` is the residue number

`alt-loc` is the alternate/alternative location symbol (*e.g.* `"A"` or `"B"`, but most often `""`)

`ins-code` is the insertion code (usually `""`)

`imol-coords` is the molecule number of the coordinates molecule

`imol-map` is the molecule number of the map to which you wish to fit the side chains

`clash-flag` should the positions of other residues be included in the scoring of the rotamers (*i.e.* clashing with other other atoms gets marked as bad/unlikely)

---

[19] since version 0.4

[20] SC Lovell, JM Word, JS Richardson and DC Richardson (2000) "The Penultimate Rotamer Library" Proteins: Structure Function and Genetics 40: 389-408. You can get the paper from http://kinemage.biochem.duke.edu/databases/rotamer.php

[21] "The Backrub Motion: How Protein Backbone Shrugs When a Sidechain Dances" *Structure*, Volume **14**, Issue 2, Pages 265-274 I. Davis, W. Bryan Arendall III, D. Richardson, J. Richardson

lowest-rotamer-probability: some rotamers of some side chains are so unlikely that they shouldn't be considered - typically 0.01 (1%).

You can change the auto-fit rotamer fitting algorithms using

`(set-rotamer-search-mode` *mode*`)`

where *mode* is one of `(ROTAMERSEARCHAUTOMATIC)`, `(ROTAMERSEARCHLOWRES)` (*i.e.* "Backrub Rotamers") or `(ROTAMERSEARCHHIGHRES)` (the conventional/high-resolution method using rigid-body fitting).

By default, the auto-fit rotamer method is `(ROTAMERSEARCHAUTOMATIC)`.

5.12.2 De-clashing residues

Sometimes you don't have a map[22] but nevertheless there are clashing residues [23] (for example after mutation of a residue range) and you need to rotate side-chains to a non-clashing rotamer. There is a scripting interface:

`(de-clash` *imol chain-id start-resno end-resno*`)`

start-resno is the residue number of the first residue you wish to de-clash

end-resno is the residue number of the last residue you wish to de-clash

imol is the molecule number of the coordinates molecule

This interface will not change residues with insertion codes or alternate conformation. The *lowest-rotamer-probability* is set to 0.01.

5.13 Editing chi Angles

Instead of using Rotamers, one can instead change the χ angles (often called "torsions") "by hand" (using "Edit Chi Angles" from the "Model/Fit/Refine" dialog). To edit a residue's χ_1 press "1": to edit χ_2, "2": χ_3 "3" and χ_4 "4". Use left-mouse click and drag to change the χ value. Use keyboard "0"[24] to go back to ordinary view mode at any time during the editing. Alternatively, one can use the "View Rotation Mode" or use the CTRL key when moving the mouse in the graphics window. Use the Accept/Reject dialog when you have finished editing the χ angles.

For non-standard residues, the clicked atom defines the base of the atom tree, which defines the "head" of the molecule (it's the "tail" (twigs/leaves) that wags). To emphasise, then: it matters on which atom you click!

By default torsions for hydrogen atoms are turned off. To turn them on:

`(set-find-hydrogen-torsions 1)`

5.14 Torsion General

You need to click on the torsion-general button, then click 4 atoms that describe the torsion - the first atom will be the base (non moving) part of the atom tree, on clicking the 4th atom a dialog will pop up with a "Reverse" button. Move this dialog out of the way and

[22] for example, in preparation of a model for molecular replacement

[23] atoms of residues that are too close to each other

[24] that's "zero".

then left mouse click and drag in the main window will rotate the "top" part of the residue round the clicked atoms 2 and 3. When you are happy, click "Accept".

If you are torsion generaling a residue that has an alt conf, then the atoms of residue that are moved are those that have the same alt conf as the 4th clicked atom (or have an blank alt conf).

5.14.1 Ligand Torsion angles

For ligands, you will need to read the mmCIF file that contains a description of the ligand's geometry (see Section Section 5.1 [Regularization and Real Space Refinement], page 26). By default, torsions that move hydrogens are not included. Only 9 torsion angles are available from the keyboard torsion angle selection.

5.15 Pep-flip

Coot uses the same pepflip scheme as is used in `O` (*i.e.* the C, N and O atoms are rotated 180^o round a line joining the Cα atoms of the residues involved in the peptide). Flip the peptide again to return the atoms to their previous position.

5.16 Add Alternate Conformation

The allows the addition alternate (dual, triple *etc.*) conformations to the picked residue. By default, this provides a choice of rotamer (Section Section 5.12 [Rotamers], page 33). If there are not the correct main chain atoms a rotamer choice cannot be provided, and Coot falls back to providing intermediate atoms.

The default occupancy for new atoms is 0.5. This can be changed by using use slider on the rotamer selection window or by using the scripting function:

```
(set-add-alt-conf-new-atoms-occupancy 0.4)
```

The remaining occupancy of the atoms (after the new occupancy has been added) is split amongst the atoms that existed in the residue before the split. It is important therefore that the residues atoms have sane occupancies before adding an alternative conformation.

The default Split Type is to split the whole residue. If you want the default to be to split a residue after (and including) the CA, then add to your '`.coot`' file:

```
(set-add-alt-conf-split-type-number 0)
```

5.17 Mutation

Mutations are available on a 1-by-1 basis using the graphics. After selecting "Mutate. . ." from the "Model/Fit/Refine" dialog, click on an atom in the graphics. A "Residue Type" window will now appear. Select the new residue type you wish and the residue in the graphics is updated to the new residue type[25]. The initial position of the new rotamer is the *a priori* most likely rotamer. Note that in interactive mode, such as this, a residue type match[26] will not stop the mutation action occurring.

[25] Note that selecting a residue type that matches the residue in the graphics will also result in a mutation

[26] *i.e.* the current residue type matches the residue type to which you wish to mutate the residue

5.17.1 Mutating DNA/RNA

Mutation of DNA or RNA can be performed using "Simple Mutate" from the Model/Fit/Refine dialog. Residues need to be named `"Ad"`, `"Gr"`, `"Ur"` etc.

5.17.2 Multiple mutations

This dialog can be found under `Calculate -> Mutate Residue Range`. A residue range can be assigned a sequence and optionally fitted to the map. This is useful converting a poly-ALA model to the correct sequence[27].

Multiple mutations are also supported *via* the scripting interface. Unlike the single residue mutation function, a residue type match *will* prevent a modification of the residue[28]. Two functions are provided: To mutate a whole chain, use `(mutate-chain` *`imol chain-id sequence`*`)` where:

`chain-id` is the chain identifier of the chain that you wish to mutate (*e.g.* `"A"`) and

`imol` is molecule number.

`sequence` is a list of single-letter residue codes, such as `"GYRESDF"` (this should be a straight string with no additional spaces or carriage returns).

Note that the number of residues in the sequence chain and those in the chain of the protein must match exactly (*i.e.* the whole of the chain is mutated (except residues that have a matching residue type).)

To mutate a residue range, use

- `(mutate-residue-range` *`chain-id start-res-no stop-res-no sequence`*`)`

where

`start-res-no` is the starting residue for mutation

`stop-res-no` is the last residue for mutation, *i.e.* using values of 2 and 3 for *`start-res-no`* and *`stop-res-no`* respectively will mutate 2 residues.

Again, the length of the sequence must correspond to the residue range length. Note also that this is a protein sequence - not nucleic acid.

5.17.3 Mutating to a Non-Standard Residue

Sometimes one might like to model post-translational or other such modifications. How is that done, if the new residue type is not one of the standard residue types?

There is a scripting function:

`(mutate-by-overlap` *`imol chain-id resno new-three-letter-code`*`)`

This imports a model residue for the new residue type and overlays it on to the given residue by using graph-matching to determine the equivalent atoms.

The GUI for this can be found under `Extensions -> Modelling -> Replace Residue...` (for this to work, you need to be centred on the residue you wish to replace).

---

[27] *e.g.* after using Ca -> Mainchain.

[28] *i.e.* the residue atoms will remain untouched

5.17.4 Mutate and Autofit

The function combines Mutation and Auto Fit Rotamer and is the easiest way to make a mutation and then fit to the map. You can currently only "Mutate and Autofit" protein residues (*i.e.* things with a rotamer dictionary.

5.17.5 Renumbering

Renumbering is straightforward using the renumber dialog available under `Calculate -> Renumber Residue Range...`. There is also a scripting interface:

`(renumber-residue-range` *imol chain-id start-res-no last-resno offset*`)`

5.18 Importing Lignds/Monomers

You can import monomers (often ligands) using `File -> Get Monomer...`[29] by providing the 3-letter code of your monomer/ligand. The resulting molecule will be moved so that it placed at the current screen centre.

Typically, when you are happy about the placement of the ligand, you'd then use `Merge Molecules` to add the ligand/monomer to the main set of coordinates.

This procedure creates a pdb file '`monomer-XXX.pdb`' and a dictionary file '`libcheck_XXX.cif`' in the directory in which Coot was started.

A future invocation of Get Monomer uses these file so that the monomer appears quickly[30].

5.19 Ligand from SMILES strings

Similarly, you can generate ligands using `File -> SMILES...` and providing a SMILES string and a code for the residue name (this is your name for the residue type and a dictionary will be generated for the monomer of this type). This function is also a wrapper to LIBCHECK.

5.20 Find Ligands

You are offered a selection of maps to search (you can only choose one at a time) and a selection of molecules that act as a mask to this map. Finally you must choose which ligand types you are going to search for in this map[31]. Only molecules with less than 400 atoms are suggested as potential ligands.

If you do not have any molecules with less that 400 atoms loaded in Coot, you will get the message:

"Error: you must have at least one ligand to search for!"

New ligands are placed where the map density is and protein (mask) atoms are *not*). The masked map is searched for clusters using a default cut-off of 1.0σ. In weak density this cut-off may be too high and in such a case the cut-off value can be changed using something such as:

[29] this is a wrapper round LIBCHECK, so you must have CCP4 suite to installed for this function to work

[30] rather than running LIBCHECK again

[31] you can search for many different ligand types.

`(set-ligand-cluster-sigma-level 0.8)`

However, if the map to be searched for ligands is a difference map, a cluster level of 2.0 or 3.0 would probably be more appropriate (less likely to generate spurious sites).

Each ligand is fitted with rigid body refinement to each potential ligand site in the map and the best one for each site selected and written out as a pdb file. The clusters are sorted by size, the biggest one first (with an index of 0). The output placed ligands files have a prefix "best-overall" and are tagged by the cluster index and residue type of the best fit ligand in that site.

By default, the top 10 sites are tested for ligands - to increase this use:

`(set-ligand-n-top-ligands 20)`

5.20.1 Flexible Ligands

If the "Flexible?" checkbutton is activated, coot will generate a number of variable conformations (default 100) by rotating around the rotatable bonds (torsions). Each of these conformations will be fitted to each of the potential ligand sites in the map and the best one will be selected (again, if it passes the fitting criteria above).

Before you search for flexible ligands you must have read the mmCIF dictionary for that particular ligand residue type (`File -> Import CIF dictionary`).

Use:

`(set-ligand-flexible-ligand-n-samples` *`n-samples`*`)`

where *`n-samples`* is the number of samples of flexibility made for each ligand. Generally speaking, The more the number of rotatable bonds, the bigger this number should be.

By default the options to change these values are not in the GUI. To enable these GUI options, use the scripting function:

`(ligand-expert)`

5.20.2 Adding Ligands to Model

After successful ligand searching, one may well want to add that displayed ligand to the current model (the coordinates set that provided the map mask). To do so, use Merge Molecules (Section Section 5.28 [Merge Molecules], page 41).

5.21 Flip Ligand

Sometimes a ligand is placed more or less in the correct position, but the orientation is wrong - or at least you might want to explore other possible orientation. To do that easily a function has been provided:

`(flip-ligand` *`imol chain-id residue-number`*`)`

This will flip the orientation of the residue around the Eigen vector corresponding to the largest Eigen value, exploring 4 possible orientations.

This function has been further wrapped to provide flipping for the active residue:

`(flip-active-ligand)`

This function can easily be bound to a key.

5.22 Find Waters

As with finding ligands, you are given a chose of maps, protein (masking) atoms. A final selection has to be made for the cut-off level, note that this value is the number of standard deviation of the density of the map *before* the map has been masked. The default sigma level (water positions must have density above this level) is set for a "2Fo-Fc"-style map. If you want to use a difference map, you must change the sigma level (typically to 3 sigma) otherwise you run the risk of fitting waters to difference map noise peaks.

Then the map is masked by the masking atoms and a search is made of features in the map about the electron density cut-off value. Waters are added if the feature is approximately water-sized and can make sensible hydrogen bonds to the protein atoms. The new waters are optionally created in a new molecule called "Waters".

You have control over several parameters used in the water finding:

`(set-write-peaksearched-waters)`

which writes `ligand-waters-peaksearch-results.pdb`, which contains the water peaks (from the clusters) without any filtering and `ligand-waters.pdb` which are a disk copy filtered waters that have been either added to the molecule or from which a new molecule has been created.

`(set-ligand-water-to-protein-distance-limits` *`min-d max-d`*`)` sets the minimum and maximum allowable distances between new waters and the masking molecule (usually the protein).

`(set-ligand-water-spherical-variance-limit` *`varlim`*`)` sets the upper limit for the density variance around water atoms. The default is 0.12.

The map that is marked by the protein and is searched to find the waters is written out in CCP4 format as `"masked-for-waters.map"`.

5.22.1 Refinement Failure

Sometimes as a result of water fitting, you may see something like:

```
WARNING:: refinement failure
          start pos: xyz = (      17.1,     34.76,     60.42)
          final pos: xyz = (     17.19,     34.61,     60.59)
```

When Coot finds a blob, it does a crude positioning of an atom at the centre of the grid points. It then proceeds to move to the peak of the blob by a series of translations. There are a certain number of cycles, and if it doesn't reach convergence by the end of those cycles then you get the error message.

Often when you go to the position indicated, you can see why Coot had a problem in the refinement.

5.22.2 Blobs

After a water search, Coot will create a blobs dialog (see Section Section 7.3 [sec_blobs], page 54).

5.23 Add Terminal Residue

This creates a new residue at the C or N terminus by fitting to the map. ϕ/ψ angle pairs are selected at random based on the Ramachandran plot probability (for a generic residue).

By default there are 100 trials. It is possible that a wrong position will be selected for the terminal residue and if so, you can reject this fit and try again with Fit Terminal Residue[32]. Each of the trial positions are scored according to their fit to the map[33] and the best one selected. It is probably a good idea to run "Refine Zone" on these new residues.

If you use the Extensions (`Dock Sequence... -> Associate Sequence with Model`) to apply a PIR sequence file to a model then `Add Terminal Residue` will use the sequence alignment to determine the residue type of the added residue.

`(set-terminal-residue-do-rigid-body-refine 0)` will disable rigid body fitting of the terminal residue fragment for each trial residue position (the default is 1 (on)) - this may help if the search does not provide good results.

`(set-add-terminal-residue-n-phi-psi-trials 50)` will change the number of trials (default is 100).

5.24 Add OXT Atom to Residue

At the C-terminus of a chain of amino-acid residues, there is a "modification" so that the C-O becomes a carbonyl, *i.e.* an extra (terminal) oxygen (OXT) needs to be added. This atom is added so that it is in the plane of the Cα, C and O atoms of the residue.

Scripting usage:

`(add-OXT-to-residue imol residue-number insertion-code chain-id)`[34],

where `insertion-code` is typically `""`.

Note, in order to place OXT, the N, CA, C and O atoms must be present in the residue - if (for example) the existing carbonyl oxygen atom is called "OE1" then this function will not work.

5.25 Add Atom at Pointer

By default, "Add Atom At Pointer" will pop-up a dialog from which you can choose the atom type you wish to insert[35]. Using `(set-pointer-atom-is-dummy 1)` you can by-pass this dialog and immediately create a dummy atom at the pointer position. Use an argument of `0` to revert to using the atom type selection pop-up on a button press.

The atoms are added to a new molecule called "Pointer Atoms". They should be saved and merged with your coordinates outside of Coot.

5.26 Place Helix

The idea is to place a helix more or less "here" (the screen centre) by fitting to the electron density map. The algorithm is straightforward. First we move to the local centre of density, then examine the density for characteristic directions and fit ideal helices (of length 20 residues) to these directions. The helix is then extended if possible (by checking the fit to the map of residues added in ideal helix conformation) and chopped back if not. If the fit is

[32] usually if this still fails after two repetitions then it never seems to work.

[33] The map is selected using "Refine/Regularize Control"

[34] *e.g.* `(add-OXT-to-residue 0 428 "" "A")`

[35] including sulfate or phosphate ions (in such a case, it is probably useful to do a "Rigid Body Fit Zone" on that new residue).

successful, the helix is created in a new molecule called "Helix". If the fit is not successful, there is instead a message added to the status bar. You can build the majority of a helical protein in a few minutes using this method (you will of course have to assemble the helices and assign residue numbers and sequence later).

This is available as a scripting function `(place-helix-here)` and in the GUI (in the "Other Modelling Tools" dialog).

5.27 Building Ideal DNA and RNA

The interface to building ideal polynucleotides can be found by pressing the "Ideal RNA/DNA..." button on the "Other Modelling Tools" dialog.

For a given sequence, a choice of DNA or RNA, A or B form, single or double stranded is presented.

The interface may not gracefully handle uracils in DNA, thymines in RNA or B form RNA[36].

5.28 Merge Molecules

This dialog can be found under "Calculate" in the main menubar. This is typically used to add molecule fragments or residues that are in one molecule to the "working" coordinates[37].

5.29 Temperature Factor for New Atoms

The default temperature factor for new atoms is 30.0. This can be changed by the following

```
(set-default-temperature-factor-for-new-atoms 50.0)
```

5.30 Applying NCS Edits

Let's imagine that you have 3-fold NCS. You have molecule "A" as your master molecule and you make edits to that molecule. Now you want to apply the edits that you made to "A" (the NCS master chain ID) to the "B" and "C" molecules (i.e. you want the "B" and "C" molecules to be rotated/translated versions of the "A" molecule). How is that done?

There are now guis to NCS command to help you out (under Extensions). However, for completeness here are the scripting versions:

```
(copy-from-ncs-master-to-others imol master-chain-id)
```

If you have only a range of residues, rather than a whole chain to replace:

```
(copy-residue-range-from-ncs-master-to-others imol master-chain-id
start-resno end-resno)
```

e.g.

```
(copy-residue-range-from-ncs-master-to-others 0 "A" 1 5)
```

[36] But you don't want those things anyway, right?

[37] For example, after a ligand search has been performed.

5.31 Running Refmac

Use the "Run Refmac. . ." button to select the dataset and the coordinates on which you would like to run Refmac. Note that here Coot only allows the use of datasets which has Refmac parameters set as the MTZ file was read. By default, Coot displays the new coordinates and the new map generated from refmac's output MTZ file. Optionally, you can also display the difference map.

You can add extra parameters (data lines) to refmac's input by storing them in a file called `refmac-extra-params` in the directory in which you started coot.

You can also provide extra/replacement parameters for refmac by setting the variable `refmac-extra-params` to a list of strings, for example:

```
(set! refmac-extra-params (list "REFINE MATRIX 0.1" "MAKE HYDROGENS NO"))
```

Coot "blocks"[38] until Refmac has terminated[39].

The default refmac executable is `refmac5` it is presumed to be in the path. If you don't want this, it can be overridden using a re-definition either at the scripting interface or in one's `~/.coot` file *e.g.*:

- `(define refmac-exec "/e/refmac-new/bin/refmac5.6.3")`

After running refmac several times, you may find that you prefer if the new map that refmac creates (after refmac refinement) is the same colour as the previous one (from before this refmac refinement). If so, use:

```
(set-keep-map-colour-after-refmac 1)
```

which will swap the colours of then new and old refmac map so that the post-refmac map has the same colour as the pre-refmac map and the pre-refmac map is coloured with a different colour.

5.32 Running SHELXL

Coot can read shelx `.res` files and write `.ins` files, and thus one can refine using SHELXL in a convenient manner using the function

`(shelxl-refine` *`imol . hkl-file-name`*`)`

(the *`hkl-file-name`* is an optional argument)

e.g.

```
(shelxl-refine 0)
```

or

```
(shelxl-refine 0 "insulin.hkl")
```

In the former case, coot will presume that there is a SHELX `hkl` file corresponding to the `res` file that you read in; if there is not coot will print a warning and not try to run shelxl. In the latter case, you can specify the location of the `hkl` file.

After shelxl has finished, coot will automatically read in the resulting `res` coordinates, the `fcf` file, convert the data to `mmCIF` format and read that, which generates a σ_A map and a difference map.

---

[38] *i.e.* Coot is idle and ignores all input.

[39] This is not an ideal feature, of course and will be addressed in future. . . . Digressive Musing: If only computers were fast enough to run Refmac interactively. . .

Coot creates a time stamped `ins` file and a time-stamped sym-link to the `hkl` file in the `coot-shelxl` directory.

Please note that the output `ins` file will not be particularly useful (and thus shelxl will fail) if the input file was not in SHELX `ins` format.

There is a GUI for this operation under the "Extensions" menu item.

5.33 Clear Pending Picks

Sometimes one can click on a button[40] unintentionally. This button is there for such a case. It clears the expectation of an atom pick. This works not only for modelling functions, but also geometry functions (such as Distance and Angle).

5.34 Delete

Single atoms or residues can be deleted from the molecule using "Delete..." from the "Model/Fit/Refine"dialog. Pressing this button results in a new dialog, with the options of "Residue" (the default), "Atom" and "Hydrogen Atoms". Now click on an atom in the graphics - the deleted object will be the whole residue of the atom if "Residue" was selected and just that atom if "Atom" was selected. Note that if a residue has an alternative conformation, then "Delete Residue" will delete only the conformation that matches that alternative conformation specifier of the clicked atom.

Only waters are deletable if the `"Water"` check button is active and waters are not deletable if the `"Residue/Monomer"` check button is active. This is to reduce mis-clicking.

To rotate the view when in "Delete Mode", use Ctrl left-mouse.

If you want to delete multiple items you can use check the "Keep Delete Active" check-button on this dialog This will will keep the dialog open, ready for deletion of next item.

5.35 Sequence Assignment

You can assign a (FASTA format) sequence to a molecule using:

```
(assign-fasta-sequence imol chain-id fasta-seq)
```

This function has been provided as a precursor to functions that will (as automatically as possible) mutate your current coordinates to one that has the desired sequence. It will be used in automatic side-chain assignment (at some stage in the future).

5.36 Building Links and Loops

Coot can make an attempt to build missing linking regions or loops[41]. This is an area of Coot that needs to be improved, currently O does it much better. We will have several different loop tools here[42]. For now there is `Calculate -> Fit Gap` or the scripting function:

```
(fit-gap imol chain-id start-resno stop-resno)
```

and

```
(fit-gap imol chain-id start-resno stop-resno sequence)
```

[40] such that Coot would subsequently expect an atom selection "pick" in the graphics window.

[41] the current single function doesn't always perform very well in tests

[42] I suspect that there is not one tool that fits for all.

the second form will also mutate and try to rotamer fit the provided sequence.

Example usage: let's say for molecule number 0 in chain `"A"` we have residues up to 56 and then a gap after which we have residues 62 and beyond:

```
(fit-gap 0 "A" 57 61 "TYPWS")
```

5.37 Fill Partial Residues

After molecular replacement, the residues of your protein could well have the correct sequence but be chopped back to CG or CB atoms. There is a function to fill such partially-filled residues:

`(fill-partial-residues` *`imol`*`)`

This identifies residues with missing atoms, then fills them and does a rotamer fit and real-space refinement.

5.38 Setting Occupancies

As well as the editing "Residue Info" to change occupancies of individual atoms, one can use a scripting function to change occupancies of a whole residue range:

- `(zero-occupancy-residue-range` *`imol chain-id resno-start resno-last`*`)`

  example usage:

  ```
  (zero-occupancy-residue-range 0 "A" 23 28)
  ```

This is often useful to zero out a questionable loop before submitting for refinement. After refinement (with refmac) there should be relatively unbiased density in the resulting 2Fo-Fc-style and difference maps.

Similarly there is a function to reverse this operation:

- `(fill-occupancy-residue-range` *`imol chain-id resno-start resno-last`*`)`

5.39 Fix Nomenclature Errors

Currently this is available only in scripting form:

`(fix-nomenclature-errors` *`imol`*`)`

This will fix atoms nomenclature problems in molecule number *`imol`* according to the same criteria as WATCHECK[43] *e.g.* Chi-2 for Phe, Tyr, Asp, and Glu should be between -90 and 90 degrees. Note that Val and Leu nomenclature errors are also corrected.

5.40 Rotamer Fix Whole Protein

There is an experimental scripting function

`(fit-protein` *`imol`*`)`

which does a auto-fit rotamer and Real Space Refinement for each residue. The graphics follow the refinement.

[43] R.W.W. Hooft, G. Vriend, C. Sander, E.E. Abola, Errors in protein structures. *Nature* (1996) **381**, 272-272.

5.41 Refine All Waters

All the waters in a model can be refined (that is, moved to the local density peak) using

`(fit-waters` *`imol`*`)`

This is a non-interactive function (the waters are moved without user intervention).

5.42 Moving Molecules/Ligands

Often you want to move a ligand (or some such) from wherever it was read in to the position of interest in your molecule (*i.e.* the current view centre). There is a GUI to do this: `Calculate -> Move Molecule Here`.

There are scripting functions available for this sort of thing:

`(molecule-centre` *`imol`*`)`

will tell you the molecule centre of the *`imol`*th molecule.

`(translate-molecule-by imol x-shift y-shift z-shift)`

will translate all the atoms in molecule *`imol`* by the given amount (in Ångströms).

`(move-molecule-to-screen-centre` *`imol`*`)`

will move the *`imol`*th molecule to the current centre of the screen (sometimes useful for imported ligands). Note that this moves the atoms of the molecule - not just the view of the molecule.

5.43 Modifying the Labels on the Model/Fit/Refine dialog

If you don't like the labels "Rotate/Translate Zone" or "Place Atom at Pointer" and rather they said something else, you can change the button names using:

`(set-model-fit-refine-rotate-translate-zone-label "Move Zone")`

and

`(set-model-fit-refine-place-atom-at-pointer "Add Atom")`

6 Map-Related Features

6.1 Maps in General

Maps are "infinite," not limited to pre-calculated volume (the "Everywhere You Click - There Is Electron Density" (EYC-TIED) paradigm) symmetry-related electron density is generated automatically. Maps are easily re-contoured. Simply use the scroll wheel on you mouse to alter the contour level (or -/+ on the keyboard).

Maps follow the molecule. As you recentre or move about the crystal, the map quickly follows. If your computer is not up to re-contouring all the maps for every frame, then use `Draw -> Dragged Map...` to turn off this feature.

6.1.1 Map Reading Bug

Unfortunately, there is a bug in map-reading. If the map is not a bona-fide CCP4 map[1], then coot will crash. Sorry. A fix is in the works but "it's complicated". That's why maps are limited to the extension ".ext" and ".map", to make it less likely a non-CCP4 map is read.

6.2 Create a Map

From MTZ, mmCIF and .phs data use `File -> Open MTZ, CIF or phs...`. You can then choose the MTZ columns for the Fourier synthesis. The button "Expert mode" also adds to the options any anomalous columns you may have in the MTZ file. It also provides the option to apply resolution limits.

From a CCP4 map use `File -> Read Map`. After being generated/read, the map is immediately contoured and centred on the current rotation centre.

6.2.1 Auto-read MTZ file

This function allows Coot to read an MTZ file and make a map directly (without going through the column selection procedure). The default column labels for auto-reading are "FWT" and "PHWT" for the 2Fo-Fc-style map, "DELFWT" and "PHDELWT" for the difference map. You can change the column labels that Coot uses for auto-reading - here is an example of how to do that:

```
(set-auto-read-column-labels "2FOFCWT" "PHIWT" 0)  (set-auto-read-column-labels "FOFCWT" "DELPHIWT" 1)
```

By default the difference map is created in auto-reading the MTZ file. If you don't want a difference map, you can use the function:

```
(set-auto-read-do-difference-map-too 0)
```

6.2.2 Reading CIF data

There are several maps that can be generated from CIF files that contain observed Fs, calculated Fs and calculated phases:

- `(read-cif-data-with-phases-fo-alpha-calc` *`cif-file-name`*`)` Calculate an atom map using $F_o bs$ and $\alpha_c alc$

[1] *e.g.* it's a directory or a coordinate filename.

- `(read-cif-data-with-phases-2fo-fc` *`cif-file-name`*`)` Calculate an atom map using F_obs, F_calc and α_calc
- `(read-cif-data-with-phases-fo-fc` *`cif-file-name`*`)` Calculate an difference map using F_obs, F_calc and α_calc.

6.2.3 Reading PHS data

There are 2 ways to read data by scripting:

`(read-phs-and-make-map-using-cell-symm` *`phs-file-name space-group-name a b c alpha beta gamma`*`)`

`(read-pdb-and-make-map-with-reso-limits` *`imol-previous phs-file-name reso-limit-low reso-limit-high`*`)`

The first specifies the cell explicitly, and *`alpha, beta`* and *`gamma`* are specified in degrees.

The second form allows the specification of resolution limits and takes the cell and symmetry from a previous molecule (typically a pdb file).

6.3 Map Contouring

Maps can be re-contoured using the middle-mouse scroll-wheel (buttons 4 and 5 in X Window System(TM) terminology). Scrolling the mouse wheel will change the map contour level and the map it redrawn. If you have several maps displayed then the map that has its contour level changed can be set using `HID -> Scrollwheel -> Attach scroll-wheel to which map?`. If there is only one map displayed, then that is the map that has its contour level changed (no matter what the scroll-wheel is attached to in the menu). The level of the electron density is displayed in the top right hand corner of the OpenGL canvas.

Use keyboard `+` or `-` to change the contour level if you don't have a scroll-wheel[2].

If you are creating your map from an MTZ file, you can choose to click on the "is difference map" button on the Column Label selection widget (after a data set filename has been selected) then this map will be displayed in 2 colours corresponding to `+` and `-` the map contour level.

If you read in a map and it is a difference map then there is a checkbutton to tell Coot that.

If you want to tell Coot that a map is a difference map after it has been read, use:

`(set-map-is-difference-map` *`imol`*`)`

where *`imol`* is the molecule number.

By default the change of the contour level is determined from the sigma of the map. You can change this in the map properties dialog or by using the scripting function:

`(set-contour-by-sigma-step-by-mol` *`step on/off? imol`*`)`

where

`step` is the difference in sigma from one level to the next (typically 0.2)

`on/off?` is either 0 (sigma stepping off) or 1 (sigma stepping on)

[2] like I don't on my Mac.

By default the map radius[3] is 10Å. The default increment to the electron density depends on whether or not this is a difference map (0.05 $e^-/\mathring{A}^3$ for a "2Fo-Fc" style map and 0.005 $e^-/\mathring{A}^3$ for a difference map). You can change these using `Edit -> Map Parameters` or by using the "Properties" button of a particular map in the Display Control (Display Manager) window.

6.4 Map Extent

The extent of the map can be set using the GUI (`Edit -> Map Parameters -> Map Radius`) or by using the scripting function, *e.g.*:

```
(set-map-radius 13.2)
```

6.5 Map contour "scrolling" limits

Usually one doesn't want to look at negative contour levels of a map[4], so Coot has by default a limit that stops the contour level going beyond (less than) 0. To remove the limit:

`(set-stop-scroll-iso-map 0)` for a 2Fo-Fc style map

`(set-stop-scroll-diff-map 0)` for a difference map

To set the limits to negative (*e.g.* -0.6) levels:

```
(set-stop-scroll-iso-map-level -0.6)
```

and similarly:

```
(set-stop-scroll-diff-map-level -0.6)
```

where the level is specified in $e^-/\mathring{A}^3$.

6.6 Map Line Width

The width of the lines that describe the density can be changed like this:

```
(set-map-line-width 2)
```

The default line width is 1.

6.7 "Dynamic" Map colouring

By default, maps get coloured according to their molecule number. The starting colour (*i.e.* for molecule 0) is blue. The colour of a map can be changed by `Edit -> Map Colour...` The map colour gets updated as you change the value in the colour selector[5]. Use "OK" to fix that colour.

As subsequent maps are read, they are coloured by rotation round a colour wheel. The default colour map step is 31 degrees. You can change this using:

```
(set-colour-map-rotation-for-map step)
```

---

[3] actually, it's a box.

[4] in a coot difference map you will get to see the negative level contoured at the inverted level of the positive level, what I mean is that you don't want to see the "positive" level going less than 0.

[5] takes you right back to the good old Frodo days, no?

6.8 Difference Map Colouring

For some strange reason, some crystallographers[6] like to have their difference maps coloured with red as positive and green as negative, this option is for them:

```
(set-swap-difference-map-colours 1)
```

This option will allow the "blue is positive, red is negative" colour scheme on "Edit -> Map Colour".

6.9 Make a Difference Map

Using the "Make a Difference Map" function in the Extensions menu, one can make a difference from two arbitrary maps. The maps need not be on the same griding, or in the same space group even. The resulting map will be on the same griding and space group as the "Reference" map.

6.10 Make an Averaged Map

There is a scripting interface to the generation of map averages. As above, the maps need not be on the same grid or in the same space group. The resulting map will have the same gridding and space group as the first map in the list. Typical usage:

```
(average-map '((1 1.0) (2 1.0)))
```

The argument to `(average-map` is a list of lists, each list element is a list of the map number and a weighting factor (1.0 in this case).

6.11 Map Sampling

By default, the Shannon sampling factor is the conventional 1.5. Use larger values (`Edit -> Map Parameters -> Sampling Rate`) for smoother maps[7].

This value can be set by the scripting command

```
(set-map-sampling-rate 2.5)
```

6.12 Dragged Map

By default, the map is re-contoured at every frame during a drag (Ctrl Left-mouse). Sometimes this can be annoyingly slow and jerky so it is possible to turn it off: `Draw -> Dragged Map -> No`.

To change this by scripting:

```
(set-active-map-drag-flag 0)
```

6.13 Dynamic Map Sampling and Display Size

If activated (`Edit -> Map Parameters -> Dynamic Map Sampling`) the map will be re-sampled on a more coarse grid when the view is zoomed out. If "Display Size" is also activated, the box of electron density will be increased in size also. In this way, you can see electron density for big maps (many unit cells) and the graphics still remain rotatable.

[6] Jan Dohnalek, for instance.

[7] a value of 2.5 is often sufficient.

If you want to have these functions active for all maps, add the following to your initialization file Section 3.10.2 [Scheme], page 12:

```
(set-dynamic-map-sampling-on) (set-dynamic-map-size-display-on)
```

6.14 Skeletonization

The skeleton (also known as "Bones"[8]) can be displayed for any map. A map can be skeletonized using `Calculate -> Map Skeleton...`. Use the option menu to choose the map and click "On" then "OK" to the generate the map (the skeleton is off by default).

The level of the skeleton can be changed by using `Edit -> Skeleton Parameters... -> Skeletonization Level...` and corresponds to the electron density level in the map. By default this value is 1.2 map standard deviations. The amount of map can be changed using `Edit -> Skeleton Parameters... -> Skeleton Box Radius...`[9]. The units are in Ångströms, with 40 as the default value.

The skeleton is often recalculated as the screen centre changes - but not always since it can be an irritatingly slow calculation. If you want to force a regeneration of the displayed skeleton, simply centre on an atom (using the middle mouse button) or press the S key.

6.15 Map Sharpening

It can be educational (even useful at lower resolutions) to sharpen or blur a map. This can be achieved with the sharpening tool `Calculate -> Map Sharpening...`. By default, the maximum and minimum sharpness is +/- 30$\mathring{A}^2$, this can be changed (in this case to 80) using:

```
(set-map-sharpening-scale-limit 80)
```

6.16 Masks

A map can be masked by a set of coordinates. Use the scripting function:

```
(mask-map-by-molecule imol-map imol-model invert-mask?)
```

If *invert-mask?* is 0, this will create a new map that has density only where there are no (close) coordinates. If *invert-mask?* is 1 then the map density values will be set to zero everywhere *except* close to the atoms of molecule number *imol-model*.

The radius of the mask around each atom is 2.0Å by default. You can change this using:

```
(set-map-mask-atom-radius radius)
```

There is a GUI interface to Map Masking under the Extensions menu.

6.16.1 Example

If one wanted to show just the density around a ligand:

1. Make a pdb file the contains just the ligand and read it in to Coot - let's say it is molecule 1 and the ligand is residue 3 of chain "L".
2. Get a map that covers the ligand (*e.g.* from refmac). Let's say this map is molecule number 2.

[8] If you're living in Sweden... or Captain Kirk, that is.

[9] you may think it strange that a box has a radius, this is an idiosyncrasy of Coot.

3. Mask the map:

   `(mask-map-by-molecule 2 1 1)`

   This creates a new map. Turn the other maps off, leaving only the masked map.

To get a nice rendered image, press F8 (see Section Section 3.6 [Raster3D], page 10).

6.17 Trimming

If you want to remove all the atoms[10] that lie "outside the map" (*i.e.* in low density) you can use

`(trim-molecule-by-map` *`imol-coords imol-map density-level delete/zero-occ?`*`)`

where *`delete/zero-occ?`* is `0` to remove the atoms and `1` to set their occupancy to zero.

There is a GUI interface for this feature under the "Extensions" menu item.

6.18 Map Transformation

If you want to transform a map, you can do it thusly:

`(transform-map` *`imol rotation-matrix trans point radius`*`)`

where:

`rotation-matrix` is a 9-membered list of numbers for an orthogonal rotation matrix.

`trans` is a 3-membered list of numbers (distances in Ångstöms).

`point` is a 3-membered list of numbers (centre point in Ångstöms).

`radius` is a single number (also in Ångstöms).

This applies the rotation *`rotation-matrix`* and a translation *`trans`* to a map fragment, so that when the transformation is applied the centre of the new map is at *`point`*.

Example usage:

`(transform-map 2 '(1 0 0 0 1 0 0 0 1) '(0 0 1) (rotation-centre) 10)`

which transforms map number 2 by a translation of 1Å along the Z axis, centred at the screen centre for 10Å around that centre.

Here's a more real-world example:

Let's say we want to tranform the density over the "B" molecule to a position over the "A" molecule. First we do a LSQ transformation to get the rotation and translation that moves the "B" coordinates over the "A" coordinates:

In the terminal output we get:

```
|    0.9707,    0.2351,   0.05033|
|  -0.04676,      0.39,   -0.9196|
|   -0.2358,    0.8903,    0.3896|
(    -33.34,     21.14,     18.82)
```

The centre of the "A" molecule is at (58.456, 5.65, 11.108). So we do:

`(transform-map 3 (list 0.9707 0.2351 0.05033 -0.04676 0.39 -0.9196 -0.2358 0.8903 0.3896) (list -33.34 21.14 18.82) (list 58.456 5.65 11.108) 8)`

[10] or set their occupancy to zero

Which creates a map over the middle of the "A" molecule. Note that using a too high *radius* can cause overlap problems, so try with a small *radius* (*e.g.* 5.0) if the resulting map looks problematic.

Alternatively, instead of typing the whole matrix, you can use a coordinates least-squares fit to generate the matrix for you. `(transform-map-using-lsq-matrix)` does just that.

Heres how to use it:

`(transform-map-using-lsq-matrix` *imol-ref ref-chain ref-resno-start ref-resno-end imol-mov mov-chain mov-resno-start mov-resno-end imol-map about-pt radius*`)`

Hopefully the arguments are self explanatory (*ref* refers to the reference molecule, of course and *about-pt* is a 3-number list such as is returned by `(rotation-centre)`).

We can now export that map, if we want.

6.19 Export Map

You can write out a map from Coot (e.g. one from NCS averaging, or masking or general transformation) using the export map function:

`(export-map` *imol filename*`)`

e.g.

```
(export-map 4 "ncs-averaged.map")
```

7 Validation

The validation functions are still being added to from time to time. In future there will be more functions, particularly those that will interface to other programs.

7.1 Ramachandran Plots

Ramachandran plots are "dynamic". When you edit the molecule (*i.e.* move the coordinates of some of atoms) the Ramachandran plot gets updated to reflect those changes. Also the underlying ϕ/ψ probability density changes according to the selected residue type (*i.e.* the residue under the mouse in the plot). There are 3 different residue types: GLY, PRO, and not-GLY-or-PRO[1].

When you mouse over a representation of a residue (a little square or triangle[2]) the residue label pops up. The residue is "active" *i.e.* it can be clicked. The "graphics" view changes so that the Cα of the selected residue is centred. In the Ramachandran plot window, the current residue is highlighted by a green square.

The underlying distributions are taken from the Richardson's Top500 structures http://kinemage.biochem.duke.edu/databases/top500.php.

The probability levels for acceptable (yellow) and preferred (red) are 0.2% and 2% respectively.

You can change the contour levels:

```
(set-ramachandran-plot-contour-levels 0.025 0.003)
```

You can change the "blocksize" (the default is 10 degrees) of the contours using

```
(set-ramachandran-plot-background-block-size 5)
```

These comes into effect when a new plot is created (it doesn't change plots currently displayed).

7.2 Chiral Volumes

The dictionary is used to identify the chiral atoms of each of the model's residues. A clickable list is created of atoms whose chiral volume in the model is of a different sign to that in the dictionary.

During refinement and regularization, Coot will pop-up dialogs warning about chiral volume errors - if you have them. This can be annoying[3]. You can inhibit this dialog like this:

```
(set-show-chiral-volume-errors-dialog 0)
```

7.2.1 Fixing Chiral Volume Errors

There are two obvious ways:

1) mutate and auto-fit rotamer (mutate it to the residue type that it is)

[1] the not-GLY-or-PRO is the most familiar Ramachandran plot.

[2] prolines have a grey outline rather than a black one, triangles are glycines.

[3] but that's partly the idea, I suppose.

2) RS Refine the residue and invert the chiral centre by pulling an atom. Usually you can pull the CA to the other side of the plane made by the chiral neighbouring atoms (using ctrl left-click). Sometimes giving the CB a good old tweak is the easier way.

Inverting the CB of THR is easier, just move the OG so that the plane of the neighbours is on the other side of the CB (again with ctrl left-click).

7.3 Blobs: a.k.a. Unmodelled density

This is an interface to the Blobs dialog. A map and a set of coordinates that model the protein are required.

A blob is region of relatively high residual election density that cannot be explained by a simple water. So, for example, sulfates, ligands, mis-placed sidechains or unbuilt terminal residues might appear as blobs. The blobs are in order, the biggest[4] at the top.

7.4 Check Waters by Difference Map

Sometimes waters can be misplaced - taking the place of sidechains or ligands or crystallization agents such as phosphate for example[5]. In such cases the variance of the difference map can be used to identify these problems.

This function is also useful to check anomalous maps. Often waters are placed in density that is really a cation. If such an atom diffracts anomalously this can be identified and corrected.

By default the waters with a map variance greater than 3.5σ are listed. One can be more rigorous by using a lower cut-off:

```
(set-check-waters-by-difference-map-sigma-level 3.0)
```

7.5 Check Waters via Difference Map

Another check of waters that one can perform is the following:

`(check-waters-by-difference-map` *`imol-coords imol-diff-map`*`)`

where *`imol-coords`* is the molecule number of the coordinates that contain the waters to be checked

`imol-diff-map` is the molecule number of the difference map (it must be a difference map, not an "ordinary" map). This difference map must have been calculated using the waters. So there is no point in doing this check immediately after "Find Waters". You will need to run Refmac first[6].

This analysis will return a list of water atoms that have outstandingly high local variance of the difference map (by default a sphere of 1.5Å centred about the atom position). This analysis might find waters that are actually something else, for example: part of a ligand, a sulfate, an anion or cation, only partially occupied or should be deleted entirely. Coot doesn't decide what should be done about these atoms[7], it merely brings them to your attention. It may be interesting to use an anomalous map to do this analysis.

[4] and therefore most interesting

[5] or the water should be more properly modelled as anistrotropic or a split partial site

[6] and remember to check the difference map button in the "Run Refmac" dialog

[7] as yet

7.6 Molprobity Tools Interface

The molprobity tools '`probe`' and '`reduce`' have been interfaced into Coot (currently, the interface is not as slick as it might be). However, the tools are useful and can be used in the following way:

first we need to tell Coot where to find the relevant executables (typically you would add the following lines to you '`~/.coot`' file):

```
(define *probe-command* "/path/to/probe/executable")
(define *reduce-command* "/path/to/reduce/executable")
```

now the probe hydrogens and probe dots can be generated using `Validate -> Probe Clashes` (or in the Scripting Window):

```
(probe imol)
```

where *imol* is the molecule number of coordinates to be probed. A new molecule with Hydrogens is created (by '`reduce`') and read in.

By default Coot creates a new molecule for the molecule that now has hydrogens. To change this:

```
(set! reduce-molecule-updates-current #t)
```

and that, as you can guess, replaces, rather than adds to the "probed" molecule.

This gives a "static" view of the molecule's interactions.

To get a dynamic view (which is currently only enabled on rotating chi angles) add these to your '`~/.coot`' file:

```
(set-do-probe-dots-on-rotamers-and-chis 1)
```

To get a semi-static view (dots are regenerated in the region of zone after a "Real Space Refinement"):

```
(set-do-probe-dots-post-refine 1)
```

7.7 GLN and ASN B-factor Outliers

It is often difficult to detect by eye the correct orientation of the amino-carbonylo group of GLN and ASNs. However, we can use (properly refined) temperature factors to detect outliers. We take the Z value as half the difference between the B-factor of the NE2 and OE1 divided by the standard deviation of the B-factors of the rest of the residue. An analysis of GLNs and ASNs of high resolutions structures indicates that a Z value of greater than 2.25 indicates a potential (if not probable) flip. A "Fix" button is provided in the resultant dialog make this easy to do.

This analysis was added after discussions with Atsushi Nakagawa and so is called "Nakagawa's Bees".

The analysis does not check residues with multiple conformations.

7.8 Validation Graphs

Coot provides several graphs that are useful for model validation (on a residue by residue basis): residue density fit, geometry distortion, temperature factor variance, peptide distortion and rotamer analysis.

7.8.1 Residue Density Fit

The density fit graph shows the density fit for residues. The score is the average electron density level at the atom centres of the atoms in the residue. The height of the blocks is inversely proportional to the density average.

The residue density fit is by default scaled to a map that is calculated on the absolute scale. Sometimes you might be using a map with density levels considerably different to this, which makes the residue density fit graph less useful. To correct for this you can use the scripting function:

`(set-residue-density-fit-scale-factor` *`factor`*`)`

where *`factor`* would be $1/(4\sigma_m ap)$ (as a rule of thumb).

`(residue-density-fit-scale-factor)` returns the current scale factor (default 1.0).

7.8.2 Rotamer Analysis

Residue rotamers are scored according to the prior likelihood. Note that when CD1 and CD2 of a PHE residue are exchanged (simply a nomenclature error) this can lead to large red blocks in the graph (apparently due to very unlikely rotamers). There are several other residues that can have nomenclature errors like this. To fix these problems use

`(fix-nomenclature-errors` *`imol`*`)`

7.8.3 Temperature Factor Variance

This idea is from Eleanor Dodson, who liked to use the standard deviation of a residue's temperature factors to highlight regions of questionable structure.

Note that Hydrogens are ignored in this analysis.

7.8.4 Peptide *omega* Distortion

Peptide distortions. Some variability of the ω is to be expected in the peptide bond. But not too much. Anything more than 13 degrees is suspicicous. Unexpected CIS peptide bonds show up red.

8 Hints and Usage Tips

8.1 Documentation

This manual is on the web where it can be searched:

- http://www.biop.ox.ac.uk/coot/doc/user-manual.html monolithic version
- http://www.biop.ox.ac.uk/coot/doc/chapters/user-manual_toc.html which is split into sections

In the Menu item "About", under "Online Docs URL..." there is a entry bar that can be used to search the Coot documentation via Google. The results are returned as a web page in web browser. The browser type can be specified as in this example:

```
(set-browser-interface "firefox")
```

Example usage can be found in '`xxx/share/coot/scheme/group-settings.scm`'

8.2 Low Resolution

Building structures using low resolution data is a pain. We hope to make it less of a pain in future, but there are some things that you can do now.

- [Add Planar Peptide Restraints] Add restraints via scripting command
- [Use Secondary Structure Restraints] where appropriate under Refinement Control
- [Check Chirals] Check Chiral Volumes regularly
- [Change the Weighing Scheme] `(set-matrix 20.0)` [Default is 60, the lower the number the more the geometry is idealised]

8.3 Coot Droppings

This describes the files and directory that coot leaves behind after it has been fed (sorry, I mean "used"). Everything except the `0-coot.state.scm` state file can comfortably be deleted if needed after coot has finished.

You can stop the state and history files being written if you start coot with the `--no-guano` option.

- `0-coot.state.scm` The most important file. This contains the state of coot when you last exited. It contains things like which molecules were read, the maps, the colours of the molecules and map, the screen centre, map size and so on. When restarting a coot session, this file should usually be used.
- `0-coot-history.scm` The history of coot commands you used in your last coot session in scheme format. Incomplete history. One day this will be a complete history of the session suitable for uploading into a database describing the model modification.
- `0-coot-history.py` The history of coot commands you used in your last coot session in python format.
- `coot-download` directory where the files downloaded from the network (e.g. from the EBI and EDS) go.
- `coot-backup` Each model modification generates the saving of coordinates as a pdb file in this directory.

- `coot-refmac` When running REFMAC using the Coot interface, the input to refmac and the output go in this directory.
- `coot-molprobity` When running Molprobity's Probe and Reduce using the Coot interface, the input and output go in this directory.

8.4 Clearing Backups

Coot will occasionally ask you to clear up the '`coot-backup`' directory. You can adjust the behaviour in a number of ways:

- `(define *clear-out-backup-run-n-days* 3)` will run the backup clearance every 3 days (the default is every 7).
- `(define *clear-out-backup-old-days* 1)` will clear out files older then 1 day (rather than the default 7 days).
- You can create your own version of the function that is run on exiting Coot: `(clear-backups-maybe)`

So, if you wanted to clear out everything more than 1 day old, every time, without Coot asking you about it:

```
(define *clear-out-backup-run-n-days* 0)
(define *clear-out-backup-old-days* 1)
(define (clear-backups-maybe)
  (delete-coot-backup-files 'delete)
  (coot-real-exit 0))
```

8.5 Getting out of "Translate" Mode

If you get stuck in "translate" mode in the GL canvas (*i.e.* mouse does not rotate the view as you would expect) simply press and release the Ctrl key to return to "rotate" mode.

8.6 Getting out of "Continuous Rotation" Mode

The keyboard I key toggles the "continuous rotation" mode. The menu item `Draw -> Spin View On/Off` does the same thing.

8.7 Getting out of "Label Atom Only" Mode

Similarly, if you are stuck in a mode where the "Model/Fit/Refine" buttons don't work (the atoms are not selected, only the atom gets labelled), press and release the Shift key.

8.8 Button Labels

Button labels ending in "..." mean that a new dialog will pop-up when this button is pressed.

8.9 Picking

Note that left-mouse in the graphics window is used for both atom picking and rotating the view, so try not to click over an atom when trying to rotate the view when in atom selection mode.

8.10 Resizing View

Click and drag using right-mouse (up and down or left and right) to zoom in and out.

8.11 Scroll-wheel

To change the map to which the scroll-wheel is attached, use the scroll check button in the Display Manager or use `HID -> Scrollwheel -> Attach Scrollwheel to which map?`

8.12 Slow Computer Configuration

Several of the parameters of Coot are chosen because they are reasonable on my "middle-ground" development machine. However, these parameters can be tweaked so that slower computers perform better:

- `(set-smooth-scroll-steps 4) ; default 8`
- `(set-smooth-scroll-limit 30) ; Angstroms`
- `(set-residue-selection-flash-frames-number 3);`
- `(set-skeleton-box-size 20.0) ; A (default 40).`
- `(set-active-map-drag-flag 0) ; turn off recontouring every step`
- `(set-idle-function-rotate-angle 1.5) ; continuous spin speed`

9 Other Programs

9.1 findligand

`findligand` is a stand-alone command-line program that uses the libraries of Coot.

`findligand` provides a number of command line arguments for increased flexibility:

- `--pdbin` *`pdb-in-filename`*
  where *`pdb-in-filename`* is the protein (typically)
- `--hklin` *`mtz-filename`*
- `--f` *`f_col_label`*
- `--phi` *`phi_col_label`*
- `--clusters` *`nclust`*
  where *`nclust`* is the number of density clusters (potential ligand sites) to search for
- `--sigma` *`sigma-level`*
  where *`sigma-level`* the density level (in sigma) above which the map is searched for ligands
- `--fit-fraction` *`frac`*
  where *`frac`* is the minimum fraction of atoms in density allowed after fit [default 0.75]
- `--flexible`
  means use torsional conformation ligand search
- `--samples` *`nsamples`*
  `nsamples` is the number of flexible conformation samples [default 30]
- `--dictionary` *`cif-dictionary-name`*
  the file containing the CIF ligand dictionary description

One uses `findligand` like this:

```
$ findligand various-args ligand-pdb-file-name(s)
```

i.e. the example ligand pdb files that you wish to search for are given at the end of the command line.

10 Scripting Functions

10.1 The Virtual Trackball

10.1.1 VT-FLAT

`VT-FLAT` [define]

10.1.2 VT-SPHERICAL

`VT-SPHERICAL` [define]

10.1.3 vt-surface

`vt-surface` *mode* [function]

Where *mode* is an integer number

How should the mouse move the view?

mode=1 for "Flat", mode=2 for "Spherical Surface"

10.1.4 vt-surface-status

`vt-surface-status` [function]

return the mouse view status mode

mode=1 for "Flat", mode=2 for "Spherical Surface"

10.2 File System Functions

10.2.1 make-directory-maybe

`make-directory-maybe` *dir* [function]

Where *dir* is a string

make a directory dir (if it doesn't exist) and return error code

If it can be created, create the directory dir, return the success status like mkdir: mkdir

Returns: zero on success, or -1 if an error occurred. If dir already exists as a directory, return 0 of course.

10.2.2 set-show-paths-in-display-manager

`set-show-paths-in-display-manager` *i* [function]

Where *i* is an integer number

Show Paths in Display Manager?

Some people don't like to see the full path names in the display manager here is the way to turn them off, with an argument of 1.

10.2.3 show-paths-in-display-manager-state

`show-paths-in-display-manager-state` [function]
return the internal state

What is the internal flag?

Returns: 1 for "yes, display paths" , 0 for not

10.2.4 add-coordinates-glob-extension

`add-coordinates-glob-extension` *ext* [function]
Where *ext* is a string

add an extension to be treated as coordinate files

10.2.5 add-data-glob-extension

`add-data-glob-extension` *ext* [function]
Where *ext* is a string

add an extension to be treated as data (reflection) files

10.2.6 add-dictionary-glob-extension

`add-dictionary-glob-extension` *ext* [function]
Where *ext* is a string

add an extension to be treated as geometry dictionary files

10.2.7 add-map-glob-extension

`add-map-glob-extension` *ext* [function]
Where *ext* is a string

add an extension to be treated as geometry map files

10.2.8 remove-coordinates-glob-extension

`remove-coordinates-glob-extension` *ext* [function]
Where *ext* is a string

remove an extension to be treated as coordinate files

10.2.9 remove-data-glob-extension

`remove-data-glob-extension` *ext* [function]
Where *ext* is a string

remove an extension to be treated as data (reflection) files

10.2.10 remove-dictionary-glob-extension

`remove-dictionary-glob-extension` *ext* [function]
Where *ext* is a string

remove an extension to be treated as geometry dictionary files

10.2.11 remove-map-glob-extension

`remove-map-glob-extension` *ext* [function]
Where *ext* is a string

remove an extension to be treated as geometry map files

10.2.12 set-sticky-sort-by-date

`set-sticky-sort-by-date` [function]
sort files in the file selection by date?

some people like to have their files sorted by date by default

10.2.13 unset-sticky-sort-by-date

`unset-sticky-sort-by-date` [function]
do not sort files in the file selection by date?

removes the sorting of files by date

10.2.14 set-filter-fileselection-filenames

`set-filter-fileselection-filenames` *istate* [function]
Where *istate* is an integer number

on opening a file selection dialog, pre-filter the files.

set to 1 to pre-filter, [0 (off, non-pre-filtering) is the default

10.2.15 set-file-selection-dialog-size

`set-file-selection-dialog-size` *w* [function]
Where *w* is a GtkWidget

10.2.16 filter-fileselection-filenames-state

`filter-fileselection-filenames-state` [function]
, return the state of the above variable

10.2.17 on-filename-filter-toggle-button-toggled-gtk1

`on-filename-filter-toggle-button-toggled-gtk1` *button user_data* [function]
Where:

- *button* is a GtkButton
- *user_data* is an integer number

10.2.18 add-filename-filter

`add-filename-filter` *fileselection* [function]
Where *fileselection* is a GtkWidget

10.2.19 add-filename-filter-button

`add-filename-filter-button` *fileselection type* [function]

Where:

- *fileselection* is a GtkWidget
- *type* is an integer number

10.2.20 on-filename-filter-key-press-event

`on-filename-filter-key-press-event` *widget event user_data* [function]

Where:

- *widget* is a GtkWidget
- *event* is a GdkEventKey
- *user_data* is an integer number

10.2.21 fill-option-menu-with-coordinates-options

`fill-option-menu-with-coordinates-options` *option_menu signal_func imol_active_position* [function]

Where:

- *option_menu* is a GtkWidget
- *signal_func* is a GtkSignalFunc
- *imol_active_position* is an integer number

10.2.22 coot-file-chooser

`coot-file-chooser` [function]

10.2.23 coot-dataset-chooser

`coot-dataset-chooser` [function]

10.2.24 coot-map-name-chooser

`coot-map-name-chooser` [function]

10.2.25 coot-save-coords-chooser

`coot-save-coords-chooser` [function]

10.2.26 coot-cif-dictionary-chooser

`coot-cif-dictionary-chooser` [function]

10.2.27 coot-run-script-chooser

`coot-run-script-chooser` [function]

10.2.28 coot-save-state-chooser

`coot-save-state-chooser` [function]

10.2.29 coot-save-symmetry-chooser

`coot-save-symmetry-chooser` [function]

10.2.30 coot-screendump-chooser

`coot-screendump-chooser` [function]

10.2.31 set-directory-for-coot-file-chooser

`set-directory-for-coot-file-chooser` *w* [function]

Where *w* is a GtkWidget

10.2.32 coot-file-chooser-file-name

`coot-file-chooser-file-name` *widget* [function]

Where *widget* is a GtkWidget

10.2.33 set-filename-for-filechooserselection

`set-filename-for-filechooserselection` *widget name* [function]

Where:

- *widget* is a GtkWidget
- *name* is a string

10.3 Widget Utilities

10.3.1 get-positive-float-from-entry

`get-positive-float-from-entry` *w* [function]

Where *w* is a GtkEntry

10.3.2 handle-filename-filter-gtk1

`handle-filename-filter-gtk1` *widget* [function]

Where *widget* is a GtkWidget

10.3.3 set-transient-and-position

`set-transient-and-position` *window_type window* [function]

Where:

- *window_type* is an integer number
- *window* is a GtkWidget

10.3.4 info-dialog

`info-dialog` *txt* [function]

Where *txt* is a string

create a dialog with information

create a dialog with information string txt. User has to click to dismiss it, but it is not modal (nothing in coot is modal).

10.3.5 main-menubar

`main-menubar` [function]

10.3.6 main-statusbar

`main-statusbar` [function]

10.3.7 main-toolbar

`main-toolbar` [function]

10.4 MTZ and data handling utilities

10.4.1 manage-column-selector

`manage-column-selector` *filename* [function]

Where *filename* is a string

given a filename, try to read it as a data file

We try as .phs and .cif files first

10.4.2 manage-refmac-column-selection

`manage-refmac-column-selection` *w* [function]

Where *w* is a GtkWidget

10.4.3 fill-f-optionmenu-with-expert-options

`fill-f-optionmenu-with-expert-options` *f_optionmenu* [function]

Where *f_optionmenu* is a GtkWidget

10.4.4 handle-column-label-make-fourier

`handle-column-label-make-fourier` *column_label_window* [function]

Where *column_label_window* is a GtkWidget

10.4.5 wrapped-create-run-refmac-dialog

`wrapped-create-run-refmac-dialog` [function]

10.5 Molecule Info Functions

10.5.1 chain-n-residues

`chain-n-residues` *chain_id imol* [function]

Where:

- *chain_id* is a string
- *imol* is an integer number

the number of residues in chain chain_id and molecule number imol

Returns: the number of residues

10.5.2 molecule-centre-internal

`molecule-centre-internal` *imol iaxis* [function]

Where:

- *imol* is an integer number
- *iaxis* is an integer number

10.5.3 resname-from-serial-number

`resname-from-serial-number` *imol chain_id serial_num* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *serial_num* is an integer number

return the rename from a residue serial number

Returns: NULL (f) on failure.

10.5.4 seqnum-from-serial-number

`seqnum-from-serial-number` *imol chain_id serial_num* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *serial_num* is an integer number

a residue seqnum (normal residue number) from a residue serial number

Returns: < -9999 on failure

10.5.5 insertion-code-from-serial-number

`insertion-code-from-serial-number` *imol chain_id serial_num* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *serial_num* is an integer number

the insertion code of the residue.

Returns: NULL (f) on failure.

10.5.6 chain-id

`chain-id` *imol ichain* [function]

Where:

- *imol* is an integer number
- *ichain* is an integer number

the chain_id (string) of the ichain-th chain molecule number imol

Returns: the chain-id

10.5.7 n-chains

`n-chains` *imol* [function]

Where *imol* is an integer number

number of chains in molecule number imol

Returns: the number of chains

10.5.8 is-solvent-chain-p

`is-solvent-chain-p` *imol chain_id* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string

is this a solvent chain? [Raw function]

This is a raw interface function, you should generally not use this, but instead use (is-solvent-chain? imol chain-id)

Returns: -1 on error, 0 for no, 1 for is "a solvent chain". We wouldn't want to be doing rotamer searches and the like on such a chain.

10.5.9 sort-chains

`sort-chains` *imol* [function]

Where *imol* is an integer number

xxbrief sort the chain ids of the imol-th molecule in lexographical order

10.5.10 copy-molecule

`copy-molecule` *imol* [function]

Where *imol* is an integer number

copy molecule imol

Returns: the new molecule number. Return -1 on failure to copy molecule (out of range, or molecule is closed)

10.5.11 exchange-chain-ids-for-seg-ids

`exchange-chain-ids-for-seg-ids` *imol* [function]

Where *imol* is an integer number

Experimental interface for Ribosome People.

Ribosome People have many chains in their pdb file, they prefer segids to chainids (chainids are only 1 character). But coot uses the concept of chain ids and not seg-ids. mmdb allow us to use more than one char in the chainid, so after we read in a pdb, let's replace the chain ids with the segids. Will that help?

10.6 Library and Utility Functions

10.6.1 coot-version

`coot-version` [function]

the coot version string

Returns: something like "coot-0.1.3". New versions of coot will always be lexographically greater than previous versions.

10.6.2 molecule-name

`molecule-name` *imol* [function]

Where *imol* is an integer number

return the name of molecule number imol

Returns: 0 if not a valid name (-> f in scheme) e.g. "/a/b/c.pdb" for "d/e/f.mtz FWT PHWT"

10.6.3 set-molecule-name

`set-molecule-name` *imol new_name* [function]

Where:

- *imol* is an integer number
- *new_name* is a string

set the molecule name of the imol-th molecule

10.6.4 main-window

`main-window` [function]

10.6.5 coot-checked-exit

`coot-checked-exit` *retval* [function]

Where *retval* is an integer number

10.6.6 coot-real-exit

`coot-real-exit` *retval* [function]

Where *retval* is an integer number

exit from coot, give return value retval back to invoking process.

10.6.7 coot-clear-backup-or-real-exit

`coot-clear-backup-or-real-exit` *retval* [function]

Where *retval* is an integer number

10.6.8 fill-about-window

`fill-about-window` *widget* [function]

Where *widget* is a GtkWidget

10.6.9 add-coot-references-button

`add-coot-references-button` *widget* [function]
Where *widget* is a GtkWidget

10.6.10 wrapped-create-coot-references-dialog

`wrapped-create-coot-references-dialog` [function]

10.6.11 first-coords-imol

`first-coords-imol` [function]
What is the molecule number of first coordinates molecule?

return -1 when there is none.

10.7 Graphics Utility Functions

10.7.1 set-do-anti-aliasing

`set-do-anti-aliasing` *state* [function]
Where *state* is an integer number

set the bond lines to be antialiased

10.7.2 do-anti-aliasing-state

`do-anti-aliasing-state` [function]
return the flag for antialiasing the bond lines

10.7.3 set-do-GL-lighting

`set-do-GL-lighting` *state* [function]
Where *state* is an integer number

turn the GL lighting on (state = 1) or off (state = 0)

slows down the display of simple lines

10.7.4 do-GL-lighting-state

`do-GL-lighting-state` [function]
return the flag for GL lighting

10.7.5 use-graphics-interface-state

`use-graphics-interface-state` [function]
shall we start up the Gtk and the graphics window?

if passed the command line argument –no-graphics, coot will not start up gtk itself.

An interface function for Ralf.

10.7.6 start-graphics-interface

`start-graphics-interface` [function]
start Gtk (and graphics)

This function is useful if it was not started already (which can be achieved by using the command line argument –no-graphics).

An interface for Ralf

10.7.7 reset-view

`reset-view` [function]
"Reset" the view

return 1 if we moved, else return 0.

centre on last-read molecule with zoom 100. If we are there, then go to the previous molecule, if we are there, then go to the origin.

10.7.8 graphics-n-molecules

`graphics-n-molecules` [function]
return the number of molecules (coordinates molecules and map molecules combined) that are currently in coot

Returns: the number of molecules (closed molecules are not counted)

10.7.9 own-molecule-number

`own-molecule-number` *imol* [function]
Where *imol* is an integer number

10.7.10 next-map-for-molecule

`next-map-for-molecule` *imol* [function]
Where *imol* is an integer number

10.7.11 toggle-idle-spin-function

`toggle-idle-spin-function` [function]
Spin spin spin (or not).

10.7.12 set-idle-function-rotate-angle

`set-idle-function-rotate-angle` *f* [function]
Where *f* is a number

how far should we rotate when (auto) spinning? Fast computer? set this to 0.1

10.7.13 idle-function-rotate-angle

`idle-function-rotate-angle` [function]

10.7.14 handle-read-draw-molecule

`handle-read-draw-molecule` *filename* [function]

Where *filename* is a string

a synonym for read-pdb. Read the coordinates from filename (can be pdb, cif or shelx format)

10.7.15 handle-read-draw-molecule-with-recentre

`handle-read-draw-molecule-with-recentre` *filename* *recentre_on_read_pdb_flag* [function]

Where:

- *filename* is a string
- *recentre_on_read_pdb_flag* is an integer number

read coordinates from filename with option to not recentre.

set recentre_on_read_pdb_flag to 0 if you don't want the view to recentre on the new coordinates.

10.7.16 handle-read-draw-molecule-and-move-molecule-here

`handle-read-draw-molecule-and-move-molecule-here` *filename* [function]

Where *filename* is a string

read coordinates from filename and recentre the new molecule at the screen rotation centre.

10.7.17 read-pdb

`read-pdb` *filename* [function]

Where *filename* is a string

read coordinates from filename

10.7.18 assign-hetatms

`assign-hetatms` *imol* [function]

Where *imol* is an integer number

some programs produce PDB files with ATOMs where there should be HETATMs. This is a function to assign HETATMs as per the PDB definition.

10.7.19 replace-fragment

`replace-fragment` *imol_target imol_fragment atom_selection* [function]

Where:

- *imol_target* is an integer number
- *imol_fragment* is an integer number
- *atom_selection* is a string

replace the parts of molecule number imol that are duplicated in molecule number imol_frag

10.7.20 clear-and-update-model-molecule-from-file

`clear-and-update-model-molecule-from-file` *molecule_number file_name* [function]

Where:

- *molecule_number* is an integer number
- *file_name* is a string

replace pdb. Fail if molecule_number is not a valid model molecule. Return -1 on failure. Else return molecule_number

10.7.21 screendump-image

`screendump-image` *filename* [function]

Where *filename* is a string

dump the current screen image to a file. Format ppm

You can use this, in conjunction with spinning and view moving functions to make movies

10.7.22 add-is-difference-map-checkbutton

`add-is-difference-map-checkbutton` *fileselection* [function]

Where *fileselection* is a GtkWidget

10.7.23 on-read-map-difference-map-toggle-button-toggled

`on-read-map-difference-map-toggle-button-toggled` *button user_data* [function]

Where:

- *button* is a GtkButton
- *user_data* is an integer number

10.7.24 add-recentre-on-read-pdb-checkbutton

`add-recentre-on-read-pdb-checkbutton` *fileselection* [function]

Where *fileselection* is a GtkWidget

10.7.25 on-recentre-on-read-pdb-toggle-button-toggled

`on-recentre-on-read-pdb-toggle-button-toggled` *button user_data* [function]

Where:

- *button* is a GtkButton
- *user_data* is an integer number

10.8 Interface Preferences

10.8.1 set-scroll-by-wheel-mouse

`set-scroll-by-wheel-mouse` *istate* [function]

Where *istate* is an integer number

Some people (like Phil Evans) don't want to scroll their map with the mouse-wheel.

To turn off mouse wheel recontouring call this with istate value of 0

10.8.2 scroll-by-wheel-mouse-state

`scroll-by-wheel-mouse-state` [function]

return the internal state of the scroll-wheel map contouring

10.8.3 set-default-initial-contour-level-for-map

`set-default-initial-contour-level-for-map` *n_sigma* [function]

Where *n_sigma* is a number

set the default inital contour for 2FoFc-style map

in sigma

10.8.4 set-default-initial-contour-level-for-difference-map

`set-default-initial-contour-level-for-difference-map` [function]
n_sigma

Where *n_sigma* is a number

set the default inital contour for FoFc-style map

in sigma

10.8.5 print-view-matrix

`print-view-matrix` [function]

print the view matrix to the console, useful for molscript, perhaps

10.8.6 get-view-matrix-element

`get-view-matrix-element` *row col* [function]

Where:

- *row* is an integer number
- *col* is an integer number

10.8.7 get-view-quaternion-internal

`get-view-quaternion-internal` *element* [function]

Where *element* is an integer number

internal function to get an element of the view quaternion. The whole quaternion is returned by the scheme function view-quaternion

10.8.8 set-view-quaternion

`set-view-quaternion` *i j k l* [function]

Where:

- *i* is a number
- *j* is a number
- *k* is a number
- *l* is a number

Set the view quaternion.

10.8.9 apply-ncs-to-view-orientation

`apply-ncs-to-view-orientation` *imol current_chain next_ncs_chain* [function]

Where:

- *imol* is an integer number
- *current_chain* is a string
- *next_ncs_chain* is a string

Given that we are in chain current_chain, apply the NCS operator that maps current_chain on to next_ncs_chain, so that the relative view is preserved. For NCS skipping.

10.8.10 apply-ncs-to-view-orientation-and-screen-centre

`apply-ncs-to-view-orientation-and-screen-centre` *imol current_chain next_ncs_chain* [function]

Where:

- *imol* is an integer number
- *current_chain* is a string
- *next_ncs_chain* is a string

10.8.11 set-fps-flag

`set-fps-flag` *t* [function]

Where *t* is an integer number

10.8.12 get-fps-flag

`get-fps-flag` [function]

10.8.13 set-show-origin-marker

`set-show-origin-marker` *istate* [function]

Where *istate* is an integer number

set a flag: is the origin marker to be shown? 1 for yes, 0 for no.

10.8.14 show-origin-marker-state

`show-origin-marker-state` [function]
return the origin marker shown? state

10.8.15 hide-modelling-toolbar

`hide-modelling-toolbar` [function]
hide the vertical modelling toolbar in the GTK2 version

10.8.16 show-modelling-toolbar

`show-modelling-toolbar` [function]
show the vertical modelling toolbar in the GTK2 version (the toolbar is shown by default)

10.8.17 show-model-toolbar-all-icons

`show-model-toolbar-all-icons` [function]
show all available icons in the modelling toolbar (same as MFR dialog)

10.8.18 show-model-toolbar-main-icons

`show-model-toolbar-main-icons` [function]
show only a selection of icons in the modelling toolbar

10.8.19 toolbar-popup-menu

`toolbar-popup-menu` *toolbar event_button user_data* [function]
Where:

- *toolbar* is a GtkToolbar
- *event_button* is a GdkEventButton
- *user_data* is an integer number

10.8.20 set-model-toolbar-docked-position-callback

`set-model-toolbar-docked-position-callback` *w user_data* [function]
Where:

- *w* is a GtkWidget
- *user_data* is an integer number

10.8.21 reattach-modelling-toolbar

`reattach-modelling-toolbar` [function]
reattach the modelling toolbar to the last attached position

10.8.22 set-model-toolbar-docked-position

`set-model-toolbar-docked-position` *state* [function]

Where *state* is an integer number

to swap sides of the Model/Fit/Refine toolbar 0 (default) is right, 1 is left, 2 is top, 3 is bottom

10.8.23 suck-model-fit-dialog

`suck-model-fit-dialog` [function]

reparent the Model/Fit/Refine dialog so that it becomes part of the main window, next to the GL graphics context

10.8.24 suck-model-fit-dialog-bl

`suck-model-fit-dialog-bl` [function]

10.8.25 close-model-fit-dialog

`close-model-fit-dialog` *dialog_hbox* [function]

Where *dialog_hbox* is a GtkWidget

10.8.26 popup-window

`popup-window` *s* [function]

Where *s* is a string

10.8.27 add-status-bar-text

`add-status-bar-text` *s* [function]

Where *s* is a string

Put text s into the status bar.

use this to put info for the user in the statusbar (less intrusive than popup).

10.8.28 set-model-fit-refine-dialog-stays-on-top

`set-model-fit-refine-dialog-stays-on-top` *istate* [function]

Where *istate* is an integer number

10.8.29 model-fit-refine-dialog-stays-on-top-state

`model-fit-refine-dialog-stays-on-top-state` [function]

10.8.30 save-accept-reject-dialog-window-position

`save-accept-reject-dialog-window-position` *acc_reg_dialog* [function]

Where *acc_reg_dialog* is a GtkWidget

10.8.31 set-accept-reject-dialog

`set-accept-reject-dialog` *w* [function]

Where *w* is a GtkWidget

10.8.32 set-accept-reject-dialog-docked

`set-accept-reject-dialog-docked` *state* [function]
Where *state* is an integer number

10.8.33 accept-reject-dialog-docked-state

`accept-reject-dialog-docked-state` [function]

10.8.34 set-accept-reject-dialog-docked-show

`set-accept-reject-dialog-docked-show` *state* [function]
Where *state* is an integer number

10.8.35 accept-reject-dialog-docked-show-state

`accept-reject-dialog-docked-show-state` [function]

10.8.36 set-model-toolbar-style

`set-model-toolbar-style` *state* [function]
Where *state* is an integer number

10.8.37 model-toolbar-style-state

`model-toolbar-style-state` [function]

10.9 Mouse Buttons

10.9.1 quanta-buttons

`quanta-buttons` [function]

10.9.2 quanta-like-zoom

`quanta-like-zoom` [function]

10.9.3 set-control-key-for-rotate

`set-control-key-for-rotate` *state* [function]
Where *state* is an integer number

Alternate mode for rotation.

Prefered by some, including Dirk Kostrewa. I don't think this mode works properly yet

10.9.4 control-key-for-rotate-state

`control-key-for-rotate-state` [function]
return the control key rotate state

10.9.5 blob-under-pointer-to-screen-centre

`blob-under-pointer-to-screen-centre` [function]

10.10 Cursor Function

10.10.1 normal-cursor

`normal-cursor` [function]

10.10.2 fleur-cursor

`fleur-cursor` [function]

10.10.3 pick-cursor-maybe

`pick-cursor-maybe` [function]

10.10.4 rotate-cursor

`rotate-cursor` [function]

10.10.5 set-pick-cursor-index

`set-pick-cursor-index` *icursor_index* [function]

Where *icursor_index* is an integer number

let the user have a different pick cursor

sometimes (the default) GDK_CROSSHAIR is hard to see, let the user set their own

10.11 Model/Fit/Refine Functions

10.11.1 post-model-fit-refine-dialog

`post-model-fit-refine-dialog` [function]

display the Model/Fit/Refine dialog

10.11.2 wrapped-create-model-fit-refine-dialog

`wrapped-create-model-fit-refine-dialog` [function]

10.11.3 update-model-fit-refine-dialog-menu

`update-model-fit-refine-dialog-menu` *widget* [function]

Where *widget* is a GtkWidget

10.11.4 update-model-fit-refine-dialog-buttons

`update-model-fit-refine-dialog-buttons` *widget* [function]

Where *widget* is a GtkWidget

10.11.5 unset-model-fit-refine-dialog

`unset-model-fit-refine-dialog` [function]

10.11.6 unset-refine-params-dialog

`unset-refine-params-dialog` [function]

10.11.7 show-select-map-dialog

`show-select-map-dialog` [function]
display the Display Manager dialog

10.11.8 set-model-fit-refine-rotate-translate-zone-label

`set-model-fit-refine-rotate-translate-zone-label` *txt* [function]
Where *txt* is a string
Allow the changing of Model/Fit/Refine button label from "Rotate/Translate Zone".

10.11.9 set-model-fit-refine-place-atom-at-pointer-label

`set-model-fit-refine-place-atom-at-pointer-label` *txt* [function]
Where *txt* is a string
Allow the changing of Model/Fit/Refine button label from "Place Atom at Pointer".

10.11.10 wrapped-create-other-model-tools-dialog

`wrapped-create-other-model-tools-dialog` [function]

10.11.11 unset-other-modelling-tools-dialog

`unset-other-modelling-tools-dialog` [function]

10.11.12 post-other-modelling-tools-dialog

`post-other-modelling-tools-dialog` [function]
display the Other Modelling Tools dialog

10.11.13 set-refinement-move-atoms-with-zero-occupancy

`set-refinement-move-atoms-with-zero-occupancy` *state* [function]
Where *state* is an integer number
shall atoms with zero occupancy be moved when refining? (default 1, yes)

10.11.14 refinement-move-atoms-with-zero-occupancy-state

`refinement-move-atoms-with-zero-occupancy-state` [function]
return the state of "shall atoms with zero occupancy be moved when refining?"

10.11.15 wrapped-create-fast-ss-search-dialog

`wrapped-create-fast-ss-search-dialog` [function]

10.12 Backup Functions

10.12.1 make-backup

`make-backup` *imol* [function]
Where *imol* is an integer number
make backup for molecule number imol

10.12.2 turn-off-backup

`turn-off-backup` *imol* [function]

Where *imol* is an integer number

turn off backups for molecule number imol

10.12.3 turn-on-backup

`turn-on-backup` *imol* [function]

Where *imol* is an integer number

turn on backups for molecule number imol

10.12.4 backup-state

`backup-state` *imol* [function]

Where *imol* is an integer number

return the backup state for molecule number imol

return 0 for backups off, 1 for backups on, -1 for unknown

10.12.5 apply-undo

`apply-undo` [function]

10.12.6 apply-redo

`apply-redo` [function]

10.12.7 set-have-unsaved-changes

`set-have-unsaved-changes` *imol* [function]

Where *imol* is an integer number

set the molecule number imol to be marked as having unsaved changes

10.12.8 have-unsaved-changes-p

`have-unsaved-changes-p` *imol* [function]

Where *imol* is an integer number

10.12.9 set-undo-molecule

`set-undo-molecule` *imol* [function]

Where *imol* is an integer number

set the molecule to which undo operations are done to molecule number imol

10.12.10 show-set-undo-molecule-chooser

`show-set-undo-molecule-chooser` [function]

show the Undo Molecule chooser - i.e. choose the molecule to which the "Undo" button applies.

10.12.11 wrapped-create-undo-molecule-chooser-dialog

`wrapped-create-undo-molecule-chooser-dialog` [function]

10.12.12 set-unpathed-backup-file-names

`set-unpathed-backup-file-names` *state* [function]

Where *state* is an integer number

set the state for adding paths to backup file names

by default directories names are added into the filename for backup (with / to _ mapping). call this with state=1 to turn off directory names

10.12.13 unpathed-backup-file-names-state

`unpathed-backup-file-names-state` [function]

return the state for adding paths to backup file names

10.13 Recover Session Function

10.13.1 recover-session

`recover-session` [function]

recover session

After a crash (shock horror!) we provide this convenient interface to restore the session. It runs through all the molecules with models and looks at the coot backup directory looking for related backup files that are more recent that the read file.

10.13.2 execute-recover-session

`execute-recover-session` *w* [function]

Where *w* is a GtkWidget

10.14 Map Functions

10.14.1 calc-phases-generic

`calc-phases-generic` *mtz_file_name* [function]

Where *mtz_file_name* is a string

fire up a GUI, which asks us which model molecule we want to calc phases from. On "OK" button there, we call map_from_mtz_by_refmac_calc_phases()

10.14.2 map-from-mtz-by-refmac-calc-phases

`map-from-mtz-by-refmac-calc-phases` *mtz_file_name f_col sigf_col imol_coords* [function]

Where:

- *mtz_file_name* is a string
- *f_col* is a string

- *sigf_col* is a string
- *imol_coords* is an integer number

Calculate SFs (using refmac optionally) from an MTZ file and generate a map. Get F and SIGF automatically (first of their type) from the mtz file.

Returns: the new molecule number, -1 on a problem.

10.14.3 map-from-mtz-by-calc-phases

`map-from-mtz-by-calc-phases` *mtz_file_name f_col sigf_col imol_coords* [function]

Where:

- *mtz_file_name* is a string
- *f_col* is a string
- *sigf_col* is a string
- *imol_coords* is an integer number

Calculate SFs from an MTZ file and generate a map.

Returns: the new molecule number.

10.14.4 get-map-colour

`get-map-colour` *imol* [function]

Where *imol* is an integer number

10.14.5 add-on-map-colour-choices

`add-on-map-colour-choices` *w* [function]

Where *w* is a GtkWidget

10.14.6 map-colour-mol-selector-activate

`map-colour-mol-selector-activate` *menuitem user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.14.7 my-delete-menu-items

`my-delete-menu-items` *widget data* [function]

Where:

- *widget* is a GtkWidget
- *data* is a void

10.14.8 add-on-map-scroll-whell-choices

`add-on-map-scroll-whell-choices` *menu* [function]

Where *menu* is a GtkWidget

10.14.9 map-scroll-wheel-mol-selector-activate

`map-scroll-wheel-mol-selector-activate` *menuitem user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.14.10 set-scroll-wheel-map

`set-scroll-wheel-map` *imap* [function]

Where *imap* is an integer number

set the map that is moved by changing the scroll wheel and change_contour_level().

10.14.11 scroll-wheel-map

`scroll-wheel-map` [function]

return the molecule number to which the mouse scroll wheel is attached

10.14.12 save-previous-map-colour

`save-previous-map-colour` *imol* [function]

Where *imol* is an integer number

save previous colour map for molecule number imol

10.14.13 restore-previous-map-colour

`restore-previous-map-colour` *imol* [function]

Where *imol* is an integer number

restore previous colour map for molecule number imol

10.14.14 set-active-map-drag-flag

`set-active-map-drag-flag` *t* [function]

Where *t* is an integer number

set the state of immediate map upate on map drag.

By default, it is on (t=1). On slower computers it might be better to set t=0.

10.14.15 get-active-map-drag-flag

`get-active-map-drag-flag` [function]

return the state of the dragged map flag

10.14.16 set-last-map-colour

`set-last-map-colour` *f1 f2 f3* [function]

Where:

- *f1* is a number
- *f2* is a number
- *f3* is a number

set the colour of the last (highest molecule number) map

10.14.17 set-map-colour

`set-map-colour` *imol red green blue* [function]

Where:

- *imol* is an integer number
- *red* is a number
- *green* is a number
- *blue* is a number

set the colour of the imolth map

10.14.18 handle-map-colour-change

`handle-map-colour-change` *map_no* [function]

Where:

- *map_no* is an integer number
- is a number

10.14.19 handle-symmetry-colour-change

`handle-symmetry-colour-change` *mol* [function]

Where:

- *mol* is an integer number
- is a number

10.14.20 fill-single-map-properties-dialog

`fill-single-map-properties-dialog` *window imol* [function]

Where:

- *window* is a GtkWidget
- *imol* is an integer number

10.14.21 set-contour-level-absolute

`set-contour-level-absolute` *imol_map level* [function]

Where:

- *imol_map* is an integer number
- *level* is a number

10.14.22 set-contour-level-in-sigma

`set-contour-level-in-sigma` *imol_map level* [function]

Where:

- *imol_map* is an integer number
- *level* is a number

10.14.23 set-last-map-sigma-step

`set-last-map-sigma-step` *f* [function]

Where *f* is a number

set the sigma step of the last map to f sigma

10.14.24 set-contour-sigma-button-and-entry

`set-contour-sigma-button-and-entry` *window imol* [function]

Where:

- *window* is a GtkWidget
- *imol* is an integer number

10.14.25 set-contour-by-sigma-step-maybe

`set-contour-by-sigma-step-maybe` *window imol* [function]

Where:

- *window* is a GtkWidget
- *imol* is an integer number

10.14.26 set-contour-by-sigma-step-by-mol

`set-contour-by-sigma-step-by-mol` *f state imol* [function]

Where:

- *f* is a number
- *state* is an integer number
- *imol* is an integer number

set the contour level step

set the contour level step of molecule number imol to f and variable state (setting state to 0 turns off contouring by sigma level)

10.14.27 data-resolution

`data-resolution` *imol* [function]

Where *imol* is an integer number

return the resolution of the data for molecule number imol. Return negative number on error, otherwise resolution in A (eg. 2.0)

10.14.28 solid-surface

`solid-surface` *imap on_off_flag* [function]

Where:

- *imap* is an integer number
- *on_off_flag* is an integer number

10.14.29 export-map

`export-map` *imol filename* [function]

Where:

- *imol* is an integer number
- *filename* is a string

export (write to disk) the map of molecule number imol to filename.

Return 0 on failure, 1 on success.

10.14.30 transform-map-raw

`transform-map-raw` *imol r00 r01 r02 r10 r11 r12 r20 r21 r22 t0 t1 t2 pt0 pt1 pt2 box_half_size* [function]

Where:

- *imol* is an integer number
- *r00* is a number
- *r01* is a number
- *r02* is a number
- *r10* is a number
- *r11* is a number
- *r12* is a number
- *r20* is a number
- *r21* is a number
- *r22* is a number
- *t0* is a number
- *t1* is a number
- *t2* is a number
- *pt0* is a number
- *pt1* is a number
- *pt2* is a number
- *box_half_size* is a number

10.14.31 rotate-map-round-screen-axis-x

`rotate-map-round-screen-axis-x` *r_degrees* [function]

Where *r_degrees* is a number

10.14.32 rotate-map-round-screen-axis-y

`rotate-map-round-screen-axis-y` *r_degrees* [function]

Where *r_degrees* is a number

10.14.33 rotate-map-round-screen-axis-z

`rotate-map-round-screen-axis-z` *r_degrees* [function]
Where *r_degrees* is a number

10.14.34 difference-map

`difference-map` *imol1 imol2 map_scale* [function]
Where:

- *imol1* is an integer number
- *imol2* is an integer number
- *map_scale* is a number

make a difference map, taking map_scale * imap2 from imap1, on the grid of imap1. Return the new molecule number. Return -1 on failure.

10.15 Density Increment

10.15.1 get-text-for-iso-level-increment-entry

`get-text-for-iso-level-increment-entry` *imol* [function]
Where *imol* is an integer number

10.15.2 get-text-for-diff-map-iso-level-increment-entry

`get-text-for-diff-map-iso-level-increment-entry` *imol* [function]
Where *imol* is an integer number

10.15.3 set-iso-level-increment

`set-iso-level-increment` *val* [function]
Where *val* is a number

set the contour scroll step (in absolute e/A3) for 2Fo-Fc-style maps to val

The is only activated when scrolling by sigma is turned off

10.15.4 get-iso-level-increment

`get-iso-level-increment` [function]

10.15.5 set-iso-level-increment-from-text

`set-iso-level-increment-from-text` *text imol* [function]
Where:

- *text* is a string
- *imol* is an integer number

10.15.6 set-diff-map-iso-level-increment

`set-diff-map-iso-level-increment` *val* [function]

Where *val* is a number

set the contour scroll step for difference map (in absolute e/A3) to val

The is only activated when scrolling by sigma is turned off

10.15.7 get-diff-map-iso-level-increment

`get-diff-map-iso-level-increment` [function]

10.15.8 set-diff-map-iso-level-increment-from-text

`set-diff-map-iso-level-increment-from-text` *text imol* [function]

Where:

- *text* is a string
- *imol* is an integer number

10.15.9 single-map-properties-apply-contour-level-to-map

`single-map-properties-apply-contour-level-to-map` *w* [function]

Where *w* is a GtkWidget

10.15.10 set-map-sampling-rate-text

`set-map-sampling-rate-text` *text* [function]

Where *text* is a string

10.15.11 set-map-sampling-rate

`set-map-sampling-rate` *r* [function]

Where *r* is a number

set the map sampling rate (default 1.5)

Set to something like 2.0 or 2.5 for more finely sampled maps. Useful for baton-building low resolution maps.

10.15.12 get-text-for-map-sampling-rate-text

`get-text-for-map-sampling-rate-text` [function]

10.15.13 get-map-sampling-rate

`get-map-sampling-rate` [function]

return the map sampling rate

10.15.14 set-scrollable-map

`set-scrollable-map` *imol* [function]

Where *imol* is an integer number

set the map that has its contour level changed by the scrolling the mouse wheel to molecule number imol

10.15.15 change-contour-level

`change-contour-level` *is_increment* [function]

Where *is_increment* is an integer number

change the contour level of the current map by a step

if is_increment=1 the contour level is increased. If is_increment=0 the map contour level is decreased.

10.15.16 set-last-map-contour-level

`set-last-map-contour-level` *level* [function]

Where *level* is a number

set the contour level of the map with the highest molecule number to level

10.15.17 set-last-map-contour-level-by-sigma

`set-last-map-contour-level-by-sigma` *n_sigma* [function]

Where *n_sigma* is a number

set the contour level of the map with the highest molecule number to n_sigma sigma

10.15.18 set-stop-scroll-diff-map

`set-stop-scroll-diff-map` *i* [function]

Where *i* is an integer number

create a lower limit to the "2Fo-Fc-style" map contour level changing

(default 1 on)

10.15.19 set-stop-scroll-iso-map

`set-stop-scroll-iso-map` *i* [function]

Where *i* is an integer number

create a lower limit to the difference map contour level changing

(default 1 on)

10.15.20 set-stop-scroll-iso-map-level

`set-stop-scroll-iso-map-level` *f* [function]

Where *f* is a number

set the actual map level changing limit

(default 0.0)

10.15.21 set-stop-scroll-diff-map-level

`set-stop-scroll-diff-map-level` *f* [function]

Where *f* is a number

set the actual difference map level changing limit

(default 0.0)

10.15.22 set-residue-density-fit-scale-factor

`set-residue-density-fit-scale-factor` *f* [function]

Where *f* is a number

set the scale factor for the Residue Density fit analysis

10.16 Density Functions

10.16.1 set-map-line-width

`set-map-line-width` *w* [function]

Where *w* is an integer number

draw the lines of the chickenwire density in width w

10.16.2 map-line-width-state

`map-line-width-state` [function]

return the width in which density contours are drawn

10.16.3 make-and-draw-map

`make-and-draw-map` *mtz_file_name f_col phi_col weight use_weights is_diff_map* [function]

Where:

- *mtz_file_name* is a string
- *f_col* is a string
- *phi_col* is a string
- *weight* is a string
- *use_weights* is an integer number
- *is_diff_map* is an integer number

make a map from an mtz file (simple interface)

given mtz file mtz_file_name and F column f_col and phases column phi_col and optional weight column weight_col (pass use_weights=0 if weights are not to be used). Also mark the map as a difference map (is_diff_map=1) or not (is_diff_map=0) because they are handled differently inside coot.

Returns: -1 on error, else return the new molecule number

10.16.4 make-and-draw-map-with-refmac-params

`make-and-draw-map-with-refmac-params` *mtz_file_name a b weight use_weights is_diff_map have_refmac_params fobs_col sigfobs_col r_free_col sensible_f_free_col* [function]

Where:

- *mtz_file_name* is a string
- *a* is a string

- *b* is a string
- *weight* is a string
- *use_weights* is an integer number
- *is_diff_map* is an integer number
- *have_refmac_params* is an integer number
- *fobs_col* is a string
- *sigfobs_col* is a string
- *r_free_col* is a string
- *sensible_f_free_col* is an integer number

as the above function, execpt set refmac parameters too

pass along the refmac column labels for storage (not used in the creation of the map)

Returns: -1 on error, else return imol

10.16.5 make-and-draw-map-with-reso-with-refmac-params

`make-and-draw-map-with-reso-with-refmac-params` *mtz_file_name* [function]
a b weight use_weights is_diff_map have_refmac_params fobs_col sigfobs_col r_free_col sensible_f_free_col is_anomalous use_reso_limits low_reso_limit high_reso_lim

Where:

- *mtz_file_name* is a string
- *a* is a string
- *b* is a string
- *weight* is a string
- *use_weights* is an integer number
- *is_diff_map* is an integer number
- *have_refmac_params* is an integer number
- *fobs_col* is a string
- *sigfobs_col* is a string
- *r_free_col* is a string
- *sensible_f_free_col* is an integer number
- *is_anomalous* is an integer number
- *use_reso_limits* is an integer number
- *low_reso_limit* is a number
- *high_reso_lim* is a number

as the above function, except set expert options too.

10.16.6 valid-labels

`valid-labels` *mtz_file_name f_col phi_col weight_col use_weights* [function]

Where:

- *mtz_file_name* is a string
- *f_col* is a string
- *phi_col* is a string
- *weight_col* is a string
- *use_weights* is an integer number

does the mtz file have the columms that we want it to have?

10.16.7 mtz-file-has-phases-p

`mtz-file-has-phases-p` *mtz_file_name* [function]

Where *mtz_file_name* is a string

10.16.8 is-mtz-file-p

`is-mtz-file-p` *filename* [function]

Where *filename* is a string

10.16.9 cns-file-has-phases-p

`cns-file-has-phases-p` *cns_file_name* [function]

Where *cns_file_name* is a string

10.16.10 auto-read-make-and-draw-maps

`auto-read-make-and-draw-maps` *filename* [function]

Where *filename* is a string

read MTZ file filename and from it try to make maps

Useful for reading the output of refmac. The default labels (FWT/PHWT and DELFWT/PHDELFWT) can be changed using ...[something]

Returns: the molecule number for the new map

10.16.11 set-auto-read-do-difference-map-too

`set-auto-read-do-difference-map-too` *i* [function]

Where *i* is an integer number

set the flag to do a difference map (too) on auto-read MTZ

10.16.12 auto-read-do-difference-map-too-state

`auto-read-do-difference-map-too-state` [function]

return the flag to do a difference map (too) on auto-read MTZ

Returns: 0 means no, 1 means yes.

10.16.13 set-auto-read-column-labels

`set-auto-read-column-labels` *fwt phwt is_for_diff_map_flag* [function]

Where:

- *fwt* is a string
- *phwt* is a string
- *is_for_diff_map_flag* is an integer number

set the exected MTZ columns for Auto-reading MTZ file.

Not every program uses the default refmac labels (FWT/PHWT) for its MTZ file. Here we can tell coot to expect other labels,

e.g. (set-auto-read-column-labels "2FOFCWT" "PH2FOFCWT" 0)

10.16.14 get-text-for-density-size-widget

`get-text-for-density-size-widget` [function]

10.16.15 set-density-size-from-widget

`set-density-size-from-widget` *text* [function]

Where *text* is a string

10.16.16 set-map-radius

`set-map-radius` *f* [function]

Where *f* is a number

set the extent of the box/radius of electron density contours

10.16.17 set-density-size

`set-density-size` *f* [function]

Where *f* is a number

another (old) way of setting the radius of the map

10.16.18 set-map-radius-slider-max

`set-map-radius-slider-max` *f* [function]

Where *f* is a number

10.16.19 set-display-intro-string

`set-display-intro-string` *str* [function]

Where *str* is a string

Give me this nice message str when I start coot.

10.16.20 get-map-radius

`get-map-radius` [function]

return the extent of the box/radius of electron density contours

10.16.21 set-esoteric-depth-cue

`set-esoteric-depth-cue` *istate* [function]

Where *istate* is an integer number

not everone likes coot's esoteric depth cueing system

Pass an argument istate=1 to turn it off

(this function is currently disabled).

10.16.22 esoteric-depth-cue-state

`esoteric-depth-cue-state` [function]

native depth cueing system

return the state of the esoteric depth cueing flag

10.16.23 set-swap-difference-map-colours

`set-swap-difference-map-colours` *i* [function]

Where *i* is an integer number

not everone lies coot's default difference map colouring.

Pass an argument i=1 to swap the difference map colouring so that red is positve and green is negative.

10.16.24 swap-difference-map-colours-state

`swap-difference-map-colours-state` [function]

10.16.25 set-map-is-difference-map

`set-map-is-difference-map` *imol* [function]

Where *imol* is an integer number

post-hoc set the map of molecule number imol to be a difference map

Returns: success status, 0 -> failure (imol does not have a map)

10.16.26 map-is-difference-map

`map-is-difference-map` *imol* [function]

Where *imol* is an integer number

10.16.27 another-level

`another-level` [function]

Add another contour level for the last added map.

Currently, the map must have been generated from an MTZ file.

Returns: the molecule number of the new molecule or -1 on failure

10.16.28 another-level-from-map-molecule-number

`another-level-from-map-molecule-number` *imap* [function]

Where *imap* is an integer number

Add another contour level for the given map.

Currently, the map must have been generated from an MTZ file.

Returns: the molecule number of the new molecule or -1 on failure

10.16.29 residue-density-fit-scale-factor

`residue-density-fit-scale-factor` [function]

return the scale factor for the Residue Density fit analysis

10.16.30 density-at-point

`density-at-point` *imol x y z* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number

return the density at the given point for the given map. Return 0 for bad imol

10.17 Parameters from map

10.17.1 mtz-hklin-for-map

`mtz-hklin-for-map` *imol_map* [function]

Where *imol_map* is an integer number

return the mtz file that was use to generate the map

return 0 when there is no mtz file associated with that map (it was generated from a CCP4 map file say).

10.17.2 mtz-fp-for-map

`mtz-fp-for-map` *imol_map* [function]

Where *imol_map* is an integer number

return the FP column in the file that was use to generate the map

return 0 when there is no mtz file associated with that map (it was generated from a CCP4 map file say).

10.17.3 mtz-phi-for-map

`mtz-phi-for-map` *imol_map* [function]

Where *imol_map* is an integer number

return the phases column in mtz file that was use to generate the map

return 0 when there is no mtz file associated with that map (it was generated from a CCP4 map file say).

10.17.4 mtz-weight-for-map

`mtz-weight-for-map` *imol_map* [function]

Where *imol_map* is an integer number

return the weight column in the mtz file that was use to generate the map

return 0 when there is no mtz file associated with that map (it was generated from a CCP4 map file say) or no weights were used.

10.17.5 mtz-use-weight-for-map

`mtz-use-weight-for-map` *imol_map* [function]

Where *imol_map* is an integer number

return flag for whether weights were used that was use to generate the map

return 0 when no weights were used or there is no mtz file associated with that map.

10.18 PDB Functions

10.18.1 write-pdb-file

`write-pdb-file` *imol file_name* [function]

Where:

- *imol* is an integer number
- *file_name* is a string

write molecule number imol as a PDB to file file_name

10.18.2 write-residue-range-to-pdb-file

`write-residue-range-to-pdb-file` *imol chainid resno_start resno_end filename* [function]

Where:

- *imol* is an integer number
- *chainid* is a string
- *resno_start* is an integer number
- *resno_end* is an integer number
- *filename* is a string

write molecule number imol's residue range as a PDB to file file_name

10.19 Refmac Functions

10.19.1 execute-refmac

`execute-refmac` *window* [function]

Where *window* is a GtkWidget

10.19.2 refmac-molecule-button-select

`refmac-molecule-button-select` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.19.3 set-refmac-molecule

`set-refmac-molecule` *imol* [function]

Where *imol* is an integer number

10.19.4 fill-option-menu-with-refmac-options

`fill-option-menu-with-refmac-options` *optionmenu* [function]

Where *optionmenu* is a GtkWidget

10.19.5 fill-option-menu-with-refmac-methods-options

`fill-option-menu-with-refmac-methods-options` *optionmenu* [function]

Where *optionmenu* is a GtkWidget

10.19.6 fill-option-menu-with-refmac-phase-input-options

`fill-option-menu-with-refmac-phase-input-options` *optionmenu* [function]

Where *optionmenu* is a GtkWidget

10.19.7 fill-option-menu-with-refmac-labels-options

`fill-option-menu-with-refmac-labels-options` *optionmenu* [function]

Where *optionmenu* is a GtkWidget

10.19.8 fill-option-menu-with-refmac-file-labels-options

`fill-option-menu-with-refmac-file-labels-options` *optionmenu* [function]

Where *optionmenu* is a GtkWidget

10.19.9 fill-option-menu-with-refmac-ncycle-options

`fill-option-menu-with-refmac-ncycle-options` *optionmenu* [function]

Where *optionmenu* is a GtkWidget

10.19.10 update-refmac-column-labels-frame

`update-refmac-column-labels-frame` *optionmenu fobs_menu fiobs_menu fpm_menu f_free_menu phases_menu fom_menu hl_menu* [function]

Where:

- *optionmenu* is a GtkWidget
- *fobs_menu* is a GtkWidget

- *fiobs_menu* is a GtkWidget
- *fpm_menu* is a GtkWidget
- *f_free_menu* is a GtkWidget
- *phases_menu* is a GtkWidget
- *fom_menu* is a GtkWidget
- *hl_menu* is a GtkWidget

10.19.11 free-memory-run-refmac

`free-memory-run-refmac` *window* [function]
Where *window* is a GtkWidget

10.19.12 set-refmac-counter

`set-refmac-counter` *imol refmac_count* [function]
Where:

- *imol* is an integer number
- *refmac_count* is an integer number

set counter for runs of refmac so that this can be used to construct a unique filename for new output

10.19.13 refmac-name

`refmac-name` *imol* [function]
Where *imol* is an integer number

the name for refmac

Returns: a stub name used in the construction of filename for refmac output

10.19.14 get-refmac-refinement-method

`get-refmac-refinement-method` [function]
Where is a void

10.19.15 set-refmac-refinement-method

`set-refmac-refinement-method` *method* [function]
Where *method* is an integer number

10.19.16 get-refmac-phase-input

`get-refmac-phase-input` [function]
Where is a void

10.19.17 set-refmac-phase-input

`set-refmac-phase-input` *phase_flag* [function]
Where *phase_flag* is an integer number

10.19.18 set-refmac-use-tls

`set-refmac-use-tls` *state* [function]
Where *state* is an integer number

10.19.19 refmac-use-tls-state

`refmac-use-tls-state` [function]
Where is a void

10.19.20 set-refmac-use-twin

`set-refmac-use-twin` *state* [function]
Where *state* is an integer number

10.19.21 refmac-use-twin-state

`refmac-use-twin-state` [function]
Where is a void

10.19.22 set-refmac-use-sad

`set-refmac-use-sad` *state* [function]
Where *state* is an integer number

10.19.23 refmac-use-sad-state

`refmac-use-sad-state` [function]
Where is a void

10.19.24 get-refmac-ncycles

`get-refmac-ncycles` [function]
Where is a void

10.19.25 set-refmac-ncycles

`set-refmac-ncycles` *no_cycles* [function]
Where *no_cycles* is an integer number

10.19.26 add-refmac-ncycle-no

`add-refmac-ncycle-no` *cycle* [function]
Where *cycle* is an integer number

10.19.27 set-refmac-use-ncs

`set-refmac-use-ncs` *state* [function]
Where *state* is an integer number

10.19.28 refmac-use-ncs-state

`refmac-use-ncs-state` [function]
Where is a void

10.19.29 set-refmac-use-intensities

`set-refmac-use-intensities` *state* [function]
Where *state* is an integer number

10.19.30 refmac-use-intensities-state

`refmac-use-intensities-state` [function]
Where is a void

10.19.31 refmac-imol-coords

`refmac-imol-coords` [function]
Where is a void

10.19.32 add-refmac-sad-atom

`add-refmac-sad-atom` *atom_name fp fpp lambda* [function]
Where:

- *atom_name* is a string
- *fp* is a number
- *fpp* is a number
- *lambda* is a number

10.19.33 add-refmac-sad-atom-fp

`add-refmac-sad-atom-fp` *atom_name fp fpp* [function]
Where:

- *atom_name* is a string
- *fp* is a number
- *fpp* is a number

10.19.34 add-refmac-sad-atom-lambda

`add-refmac-sad-atom-lambda` *atom_name lambda* [function]
Where:

- *atom_name* is a string
- *lambda* is a number

10.19.35 clear-refmac-sad-atoms

`clear-refmac-sad-atoms` [function]

10.19.36 store-refmac-mtz-file-label

`store-refmac-mtz-file-label` *label* [function]
Where *label* is a GtkWidget

10.19.37 get-refmac-mtz-file-label

`get-refmac-mtz-file-label` [function]
Where is a void

10.19.38 fill-refmac-sad-atom-entry

`fill-refmac-sad-atom-entry` *widget* [function]
Where *widget* is a GtkWidget

10.19.39 get-refmac-used-mtz-file-state

`get-refmac-used-mtz-file-state` [function]

10.19.40 set-refmac-used-mtz-file

`set-refmac-used-mtz-file` *state* [function]
Where *state* is an integer number

10.19.41 get-saved-refmac-file-filename

`get-saved-refmac-file-filename` [function]
Where is a void

10.19.42 set-stored-refmac-file-mtz-filename

`set-stored-refmac-file-mtz-filename` *imol mtz_filename* [function]
Where:

- *imol* is an integer number
- *mtz_filename* is a string

10.19.43 save-refmac-params-to-map

`save-refmac-params-to-map` *imol_map mtz_filename fobs_col sigfobs_col r_free_col r_free_flag_sensible* [function]
Where:

- *imol_map* is an integer number
- *mtz_filename* is a string
- *fobs_col* is a string
- *sigfobs_col* is a string
- *r_free_col* is a string
- *r_free_flag_sensible* is an integer number

10.19.44 save-refmac-phase-params-to-map

`save-refmac-phase-params-to-map` *imol_map phi fom hla hlb hlc hld* [function]

Where:

- *imol_map* is an integer number
- *phi* is a string
- *fom* is a string
- *hla* is a string
- *hlb* is a string
- *hlc* is a string
- *hld* is a string

10.19.45 swap-map-colours

`swap-map-colours` *imol1 imol2* [function]

Where:

- *imol1* is an integer number
- *imol2* is an integer number

swap the colours of maps

swap the colour of maps imol1 and imol2. Useful to some after running refmac, so that the map to be build into is always the same colour

10.19.46 set-keep-map-colour-after-refmac

`set-keep-map-colour-after-refmac` *istate* [function]

Where *istate* is an integer number

flag to enable above

call this with istate=1

10.19.47 keep-map-colour-after-refmac-state

`keep-map-colour-after-refmac-state` [function]

the keep-map-colour-after-refmac internal state

Returns: 1 for "yes", 0 for "no"

10.19.48 refmac-runs-with-nolabels

`refmac-runs-with-nolabels` [function]

Where is a void

10.20 Symmetry Functions

10.20.1 get-text-for-symmetry-size-widget

`get-text-for-symmetry-size-widget` [function]

10.20.2 set-symmetry-size-from-widget

`set-symmetry-size-from-widget` *text* [function]
Where *text* is a string

10.20.3 set-symmetry-size

`set-symmetry-size` *f* [function]
Where *f* is a number

set the size of the displayed symmetry

10.20.4 get-symmetry-bonds-colour

`get-symmetry-bonds-colour` *imol* [function]
Where *imol* is an integer number

10.20.5 get-show-symmetry

`get-show-symmetry` [function]
is symmetry master display control on?

10.20.6 set-show-symmetry-master

`set-show-symmetry-master` *state* [function]
Where *state* is an integer number

set display symmetry, master controller

10.20.7 set-show-symmetry-molecule

`set-show-symmetry-molecule` *mol_no state* [function]
Where:

- *mol_no* is an integer number
- *state* is an integer number

set display symmetry for molecule number mol_no

pass with state=0 for off, state=1 for on

10.20.8 symmetry-as-calphas

`symmetry-as-calphas` *mol_no state* [function]
Where:

- *mol_no* is an integer number
- *state* is an integer number

display symmetry as CAs?

pass with state=0 for off, state=1 for on

10.20.9 get-symmetry-as-calphas-state

`get-symmetry-as-calphas-state` *imol* [function]

Where *imol* is an integer number

what is state of display CAs for molecule number mol_no?

return state=0 for off, state=1 for on

10.20.10 set-symmetry-molecule-rotate-colour-map

`set-symmetry-molecule-rotate-colour-map` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

set the colour map rotation (i.e. the hue) for the symmetry atoms of molecule number imol

10.20.11 symmetry-molecule-rotate-colour-map-state

`symmetry-molecule-rotate-colour-map-state` *imol* [function]

Where *imol* is an integer number

should there be colour map rotation (i.e. the hue) change for the symmetry atoms of molecule number imol?

return state=0 for off, state=1 for on

10.20.12 set-symmetry-colour-by-symop

`set-symmetry-colour-by-symop` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

10.20.13 set-symmetry-whole-chain

`set-symmetry-whole-chain` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

10.20.14 set-symmetry-atom-labels-expanded

`set-symmetry-atom-labels-expanded` *state* [function]

Where *state* is an integer number

10.20.15 wrapped-create-show-symmetry-window

`wrapped-create-show-symmetry-window` [function]

10.20.16 symmetry-colour-adjustment-changed

`symmetry-colour-adjustment-changed` *adj window* [function]

Where:

- *adj* is a GtkAdjustment
- *window* is a GtkWidget

10.20.17 symmetry-molecule-controller-dialog

`symmetry-molecule-controller-dialog` [function]

10.20.18 has-unit-cell-state

`has-unit-cell-state` *imol* [function]

Where *imol* is an integer number

molecule number imol has a unit cell?

Returns: 1 on "yes, it has a cell", 0 for "no"

10.20.19 save-symmetry-coords

`save-symmetry-coords` *imol filename symop_no shift_a shift_b shift_c pre_shift_to_origin_na pre_shift_to_origin_nb pre_shift_to_origin_nc* [function]

Where:

- *imol* is an integer number
- *filename* is a string
- *symop_no* is an integer number
- *shift_a* is an integer number
- *shift_b* is an integer number
- *shift_c* is an integer number
- *pre_shift_to_origin_na* is an integer number
- *pre_shift_to_origin_nb* is an integer number
- *pre_shift_to_origin_nc* is an integer number

save the symmetry coordinates of molecule number imol to filename

Allow a shift of the coordinates to the origin before symmetry expansion is apllied (this is how symmetry works in Coot internals).

10.20.20 new-molecule-by-symmetry

`new-molecule-by-symmetry` *imol m11 m12 m13 m21 m22 m23 m31 m32 m33 tx ty tz pre_shift_to_origin_na pre_shift_to_origin_nb pre_shift_to_origin_nc* [function]

Where:

- *imol* is an integer number
- *m11* is a number
- *m12* is a number
- *m13* is a number

- *m21* is a number
- *m22* is a number
- *m23* is a number
- *m31* is a number
- *m32* is a number
- *m33* is a number
- *tx* is a number
- *ty* is a number
- *tz* is a number
- *pre_shift_to_origin_na* is an integer number
- *pre_shift_to_origin_nb* is an integer number
- *pre_shift_to_origin_nc* is an integer number

create a new molecule (molecule number is the return value) from imol.

The rotation/translation matrix components are given in *fractional* coordinates.

Allow a shift of the coordinates to the origin before symmetry expansion is aplied.

Return -1 on failure.

10.20.21 setup-save-symmetry-coords

`setup-save-symmetry-coords` [function]

10.20.22 save-symmetry-coords-from-fileselection

`save-symmetry-coords-from-fileselection` *fileselection* [function]

Where *fileselection* is a GtkWidget

10.20.23 set-space-group

`set-space-group` *imol spg* [function]

Where:

- *imol* is an integer number
- *spg* is a string

set the space group for a coordinates molecule

for shelx FA pdb files, there is no space group. So allow the user to set it. This can be initted with a HM symbol or a symm list for clipper

10.20.24 set-symmetry-shift-search-size

`set-symmetry-shift-search-size` *shift* [function]

Where *shift* is an integer number

set the cell shift search size for symmetry searching.

When the coordinates for one (or some) symmetry operator are missing (which happens sometimes, but rarely), try changing setting this to 2 (default is 1). It slows symmetry searching, which is why it is not set to 2 by default.

10.21 File Selection Functions

10.21.1 set-directory-for-fileselection

`set-directory-for-fileselection` *coords_fileselection1* [function]
Where *coords_fileselection1* is a GtkWidget

10.21.2 save-directory-from-fileselection

`save-directory-from-fileselection` *fileselection* [function]
Where *fileselection* is a const GtkWidget

10.21.3 save-directory-for-saving-from-fileselection

`save-directory-for-saving-from-fileselection` *fileselection* [function]
Where *fileselection* is a const GtkWidget

10.21.4 set-file-for-save-fileselection

`set-file-for-save-fileselection` *fileselection* [function]
Where *fileselection* is a GtkWidget

10.21.5 add-sort-button-fileselection

`add-sort-button-fileselection` *fileselection* [function]
Where *fileselection* is a GtkWidget

10.21.6 add-ccp4i-project-optionmenu

`add-ccp4i-project-optionmenu` *fileselection file_selector_type* [function]
Where:

- *fileselection* is a GtkWidget
- *file_selector_type* is an integer number

10.21.7 add-ccp4i-projects-to-optionmenu

`add-ccp4i-projects-to-optionmenu` *optionmenu file_selector_type func* [function]
Where:

- *optionmenu* is a GtkWidget
- *file_selector_type* is an integer number
- *func* is a GtkSignalFunc

10.21.8 add-ccp4i-project-shortcut

`add-ccp4i-project-shortcut` *fileselection* [function]
Where *fileselection* is a GtkWidget

10.21.9 option-menu-refmac-ccp4i-project-signal-func

`option-menu-refmac-ccp4i-project-signal-func` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.21.10 run-refmac-ccp4i-option-menu-signal-func

`run-refmac-ccp4i-option-menu-signal-func` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.21.11 clear-refmac-ccp4i-project

`clear-refmac-ccp4i-project` [function]

10.21.12 lookup-file-selection-widgets

`lookup-file-selection-widgets` *item file_selector_type* [function]

Where:

- *item* is a GtkWidget
- *file_selector_type* is an integer number

10.21.13 fileselection-sort-button-clicked-gtk1

`fileselection-sort-button-clicked-gtk1` *sort_button file_list* [function]

Where:

- *sort_button* is a GtkWidget
- *file_list* is a GtkCList

10.21.14 push-the-buttons-on-fileselection

`push-the-buttons-on-fileselection` *filter_button sort_button fileselection* [function]

Where:

- *filter_button* is a GtkWidget
- *sort_button* is a GtkWidget
- *fileselection* is a GtkWidget

10.22 History Functions

10.22.1 print-all-history-in-scheme

`print-all-history-in-scheme` [function]

print the history in scheme format

10.22.2 print-all-history-in-python

`print-all-history-in-python` [function]
print the history in python format

10.22.3 set-console-display-commands-state

`set-console-display-commands-state` *istate* [function]
Where *istate* is an integer number

set a flag to show the text command equivalent of gui commands in the console as they happen.

1 for on, 0 for off.

10.22.4 set-console-display-commands-hilights

`set-console-display-commands-hilights` *bold_flag colour_flag colour_index* [function]
Where:

- *bold_flag* is an integer number
- *colour_flag* is an integer number
- *colour_index* is an integer number

10.23 State Functions

10.23.1 save-state

`save-state` [function]
save the current state to the default filename

10.23.2 save-state-file

`save-state-file` *filename* [function]
Where *filename* is a string

save the current state to file filename

10.23.3 set-save-state-file-name

`set-save-state-file-name` *filename* [function]
Where *filename* is a string

set the default state file name (default 0-coot.state.scm)

10.23.4 save-state-file-name-raw

`save-state-file-name-raw` [function]

10.23.5 set-run-state-file-status

`set-run-state-file-status` *istat* [function]

Where *istat* is an integer number

set run state file status

0: never run it 1: ask to run it 2: run it, no questions

10.23.6 run-state-file

`run-state-file` [function]

run the state file (reading from default filenname)

10.23.7 run-state-file-maybe

`run-state-file-maybe` [function]

run the state file depending on the state variables

10.23.8 wrapped-create-run-state-file-dialog

`wrapped-create-run-state-file-dialog` [function]

10.24 Clipping Functions

10.24.1 do-clipping1-activate

`do-clipping1-activate` [function]

10.24.2 clipping-adjustment-changed

`clipping-adjustment-changed` *adj window* [function]

Where:

- *adj* is a GtkAdjustment
- *window* is a GtkWidget

10.24.3 set-clipping-back

`set-clipping-back` *v* [function]

Where *v* is a number

10.24.4 set-clipping-front

`set-clipping-front` *v* [function]

Where *v* is a number

10.25 Unit Cell interface

10.25.1 get-show-unit-cell

`get-show-unit-cell` *imol* [function]

Where *imol* is an integer number

return the stage of show unit cell for molecule number imol

10.25.2 set-show-unit-cells-all

`set-show-unit-cells-all` *istate* [function]

Where *istate* is an integer number

set the state of show unit cell for all molecules

1 for displayed 0 for undisplayed

10.25.3 set-show-unit-cell

`set-show-unit-cell` *imol istate* [function]

Where:

- *imol* is an integer number
- *istate* is an integer number

set the state of show unit cell for the particular molecule number imol

1 for displayed 0 for undisplayed

10.25.4 set-unit-cell-colour

`set-unit-cell-colour` *red green blue* [function]

Where:

- *red* is a number
- *green* is a number
- *blue* is a number

10.26 Colour

10.26.1 set-symmetry-colour-merge

`set-symmetry-colour-merge` *v* [function]

Where *v* is a number

10.26.2 set-colour-map-rotation-on-read-pdb

`set-colour-map-rotation-on-read-pdb` *f* [function]

Where *f* is a number

set the hue change step on reading a new molecule

10.26.3 set-colour-map-rotation-on-read-pdb-flag

`set-colour-map-rotation-on-read-pdb-flag` *i* [function]

Where *i* is an integer number

shall the hue change step be used?

10.26.4 set-colour-map-rotation-on-read-pdb-c-only-flag

`set-colour-map-rotation-on-read-pdb-c-only-flag` *i* [function]

Where *i* is an integer number

shall the colour map rotation apply only to C atoms?

10.26.5 set-colour-by-chain

`set-colour-by-chain` *imol* [function]

Where *imol* is an integer number

colour molecule number imol by chain type

10.26.6 set-colour-by-molecule

`set-colour-by-molecule` *imol* [function]

Where *imol* is an integer number

colour molecule number imol by molecule

10.26.7 get-colour-map-rotation-on-read-pdb-c-only-flag

`get-colour-map-rotation-on-read-pdb-c-only-flag` [function]

10.26.8 set-symmetry-colour

`set-symmetry-colour` *r g b* [function]

Where:

- *r* is a number
- *g* is a number
- *b* is a number

set the symmetry colour base

10.27 Map colour

10.27.1 set-colour-map-rotation-for-map

`set-colour-map-rotation-for-map` *f* [function]

Where *f* is a number

set the colour map rotation (hue change) for maps

default: for maps is 14 degrees.

10.27.2 wrapped-create-coords-colour-control-dialog

`wrapped-create-coords-colour-control-dialog` [function]

10.27.3 set-molecule-bonds-colour-map-rotation

`set-molecule-bonds-colour-map-rotation` *imol theta* [function]

Where:

- *imol* is an integer number
- *theta* is a number

set the colour map rotation for molecule number imol

theta is in degrees

10.27.4 get-molecule-bonds-colour-map-rotation

`get-molecule-bonds-colour-map-rotation` *imol* [function]
Where *imol* is an integer number

Get the colour map rotation for molecule number imol.

10.28 Anisotropic Atoms Interface

10.28.1 get-limit-aniso

`get-limit-aniso` [function]

10.28.2 get-show-limit-aniso

`get-show-limit-aniso` [function]

10.28.3 get-show-aniso

`get-show-aniso` [function]

10.28.4 set-limit-aniso

`set-limit-aniso` *state* [function]
Where *state* is an integer number

10.28.5 set-aniso-limit-size-from-widget

`set-aniso-limit-size-from-widget` *text* [function]
Where *text* is a string

10.28.6 set-show-aniso

`set-show-aniso` *state* [function]
Where *state* is an integer number

10.28.7 get-text-for-aniso-limit-radius-entry

`get-text-for-aniso-limit-radius-entry` [function]

10.28.8 set-aniso-probability

`set-aniso-probability` *f* [function]
Where *f* is a number

10.28.9 get-aniso-probability

`get-aniso-probability` [function]

10.29 Display Functions

10.29.1 set-graphics-window-size

`set-graphics-window-size` *x_size y_size* [function]

Where:

- *x_size* is an integer number
- *y_size* is an integer number

set the window size

10.29.2 set-graphics-window-position

`set-graphics-window-position` *x_pos y_pos* [function]

Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set the window position

10.29.3 store-graphics-window-position

`store-graphics-window-position` *x_pos y_pos* [function]

Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

10.29.4 store-window-position

`store-window-position` *window_type w* [function]

Where:

- *window_type* is an integer number
- *w* is a GtkWidget

10.29.5 store-window-size

`store-window-size` *window_type w* [function]

Where:

- *window_type* is an integer number
- *w* is a GtkWidget

10.29.6 graphics-draw

`graphics-draw` [function]

draw a frame

10.29.7 hardware-stereo-mode

`hardware-stereo-mode` [function]

try to turn on stereo mode

10.29.8 stereo-mode-state

`stereo-mode-state` [function]
what is the stero state?

Returns: 1 for in hardware stereo, 2 for side by side stereo, else return 0.

10.29.9 mono-mode

`mono-mode` [function]
try to turn on mono mode

10.29.10 side-by-side-stereo-mode

`side-by-side-stereo-mode` *use_wall_eye_mode* [function]
Where *use_wall_eye_mode* is an integer number

turn on side bye side stereo mode

10.29.11 set-hardware-stereo-angle-factor

`set-hardware-stereo-angle-factor` *f* [function]
Where *f* is a number

how much should the eyes be separated in stereo mode?

10.29.12 hardware-stereo-angle-factor-state

`hardware-stereo-angle-factor-state` [function]
return the hardware stereo angle factor

10.29.13 set-model-fit-refine-dialog-position

`set-model-fit-refine-dialog-position` *x_pos y_pos* [function]
Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set position of Model/Fit/Refine dialog

10.29.14 set-display-control-dialog-position

`set-display-control-dialog-position` *x_pos y_pos* [function]
Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set position of Display Control dialog

10.29.15 set-go-to-atom-window-position

`set-go-to-atom-window-position` *x_pos y_pos* [function]

Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set position of Go To Atom dialog

10.29.16 set-delete-dialog-position

`set-delete-dialog-position` *x_pos y_pos* [function]

Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set position of Delete dialog

10.29.17 set-rotate-translate-dialog-position

`set-rotate-translate-dialog-position` *x_pos y_pos* [function]

Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set position of the Rotate/Translate Residue Range dialog

10.29.18 set-accept-reject-dialog-position

`set-accept-reject-dialog-position` *x_pos y_pos* [function]

Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set position of the Accept/Reject dialog

10.29.19 set-ramachandran-plot-dialog-position

`set-ramachandran-plot-dialog-position` *x_pos y_pos* [function]

Where:

- *x_pos* is an integer number
- *y_pos* is an integer number

set position of the Ramachadran Plot dialog

10.30 Smooth Scrolling

10.30.1 set-smooth-scroll-flag

`set-smooth-scroll-flag` *v* [function]

Where *v* is an integer number

set smooth scrolling

10.30.2 get-smooth-scroll

`get-smooth-scroll` [function]

return the smooth scrolling state

10.30.3 set-smooth-scroll-steps-str

`set-smooth-scroll-steps-str` *t* [function]

Where *t* is a string

10.30.4 set-smooth-scroll-steps

`set-smooth-scroll-steps` *i* [function]

Where *i* is an integer number

set the number of steps in the smooth scroll

Set more steps (e.g. 50) for more smoothness (default 10).

10.30.5 get-text-for-smooth-scroll-steps

`get-text-for-smooth-scroll-steps` [function]

10.30.6 set-smooth-scroll-limit-str

`set-smooth-scroll-limit-str` *t* [function]

Where *t* is a string

10.30.7 set-smooth-scroll-limit

`set-smooth-scroll-limit` *lim* [function]

Where *lim* is a number

do not scroll for distances greater this limit

10.30.8 get-text-for-smooth-scroll-limit

`get-text-for-smooth-scroll-limit` [function]

10.31 Font Size

10.31.1 set-font-size

`set-font-size` *i* [function]

Where *i* is an integer number

set the font size

10.31.2 get-font-size

`get-font-size` [function]

return the font size

Returns: 1 (small) 2 (medium, default) 3 (large)

10.31.3 set-font-colour

`set-font-colour` *red green blue* [function]

Where:

- *red* is a number
- *green* is a number
- *blue* is a number

set the colour of the atom label font - the arguments are in the range 0->1

10.32 Rotation Centre

10.32.1 set-rotation-centre-size-from-widget

`set-rotation-centre-size-from-widget` *text* [function]

Where *text* is a string

10.32.2 set-rotation-centre-size

`set-rotation-centre-size` *f* [function]

Where *f* is a number

10.32.3 get-text-for-rotation-centre-cube-size

`get-text-for-rotation-centre-cube-size` [function]

10.32.4 recentre-on-read-pdb

`recentre-on-read-pdb` [function]

10.32.5 set-recentre-on-read-pdb

`set-recentre-on-read-pdb` *int* [function]

Where *int* is a short

10.32.6 set-rotation-centre

`set-rotation-centre` *x y z* [function]

Where:

- *x* is a number
- *y* is a number
- *z* is a number

10.32.7 set-rotation-centre-internal

`set-rotation-centre-internal` *x y z* [function]

Where:

- *x* is a number
- *y* is a number
- *z* is a number

10.32.8 rotation-centre-position

`rotation-centre-position` *axis* [function]

Where *axis* is an integer number

10.33 Orthogonal Axes

10.33.1 set-draw-axes

`set-draw-axes` *i* [function]

Where *i* is an integer number

10.34 Atom Selection Utilities

10.34.1 atom-index

`atom-index` *imol chain_id iresno atom_id* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *iresno* is an integer number
- *atom_id* is a string

10.34.2 atom-index-first-atom-in-residue

`atom-index-first-atom-in-residue` *imol chain_id iresno ins_code* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *iresno* is an integer number
- *ins_code* is a string

10.34.3 atom-index-first-atom-in-residue-with-altconf

`atom-index-first-atom-in-residue-with-altconf` *imol chain_id iresno ins_code alt_conf* [function]

Where:

- *imol* is an integer number

- *chain_id* is a string
- *iresno* is an integer number
- *ins_code* is a string
- *alt_conf* is a string

10.34.4 median-temperature-factor

`median-temperature-factor` *imol* [function]
Where *imol* is an integer number

10.34.5 average-temperature-factor

`average-temperature-factor` *imol* [function]
Where *imol* is an integer number

10.34.6 clear-pending-picks

`clear-pending-picks` [function]

10.34.7 centre-of-mass-string

`centre-of-mass-string` *imol* [function]
Where *imol* is an integer number

10.34.8 set-default-temperature-factor-for-new-atoms

`set-default-temperature-factor-for-new-atoms` *new_b* [function]
Where *new_b* is a number

set the default temperature factor for newly created atoms (initial default 20)

10.34.9 default-new-atoms-b-factor

`default-new-atoms-b-factor` [function]
return the default temperature factor for newly created atoms

10.34.10 set-reset-b-factor-moved-atoms

`set-reset-b-factor-moved-atoms` *state* [function]
Where *state* is an integer number

reset temperature factor for all moved atoms to the default for new atoms (usually 30)

10.34.11 get-reset-b-factor-moved-atoms-state

`get-reset-b-factor-moved-atoms-state` [function]
return the state if temperature factors shoudl be reset for moved atoms

10.34.12 set-atom-attribute

`set-atom-attribute` *imol chain_id resno ins_code atom_name alt_conf attribute_name val* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string
- *atom_name* is a string
- *alt_conf* is a string
- *attribute_name* is a string
- *val* is a number

set a numberical attibute to the atom with the given specifier.

Attributes can be "x", "y","z", "B", "occ" and the attribute val is a floating point number

10.34.13 set-atom-string-attribute

`set-atom-string-attribute` *imol chain_id resno ins_code atom_name alt_conf attribute_name attribute_value* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string
- *atom_name* is a string
- *alt_conf* is a string
- *attribute_name* is a string
- *attribute_value* is a string

set a string attibute to the atom with the given specifier.

Attributes can be "atom-name", "alt-conf", "element" or "segid".

10.35 Skeletonization Interface

10.35.1 skel-greer-on

`skel-greer-on` [function]

10.35.2 skel-greer-off

`skel-greer-off` [function]

10.35.3 skel-foadi-on

`skel-foadi-on` [function]

10.35.4 skel-foadi-off

`skel-foadi-off` [function]

10.35.5 skeletonize-map-by-optionmenu

`skeletonize-map-by-optionmenu` *optionmenu* [function]

Where *optionmenu* is a GtkWidget

10.35.6 skeletonize-map-single-map-maybe

`skeletonize-map-single-map-maybe` *window imol* [function]

Where:

- *window* is a GtkWidget
- *imol* is an integer number

10.35.7 skeletonize-map

`skeletonize-map` *prune_flag imol* [function]

Where:

- *prune_flag* is an integer number
- *imol* is an integer number

skeletonize molecule number imol

the prune_flag should almost always be 0.

10.35.8 unskeletonize-map

`unskeletonize-map` *imol* [function]

Where *imol* is an integer number

undisplay the skeleton on molecule number imol

10.35.9 fill-option-menu-with-skeleton-options

`fill-option-menu-with-skeleton-options` *option_menu* [function]

Where *option_menu* is a GtkWidget

10.35.10 set-initial-map-for-skeletonize

`set-initial-map-for-skeletonize` [function]

10.35.11 set-max-skeleton-search-depth

`set-max-skeleton-search-depth` *v* [function]

Where *v* is an integer number

set the skeleton search depth, used in baton building

For high resolution maps, you need to search deeper down the skeleton tree. This limit needs to be increased to 20 or so for high res maps (it is 10 by default)

10.35.12 set-on-off-skeleton-radio-buttons

`set-on-off-skeleton-radio-buttons` *skeleton_frame* [function]
Where *skeleton_frame* is a GtkWidget

10.35.13 set-on-off-single-map-skeleton-radio-buttons

`set-on-off-single-map-skeleton-radio-buttons` *skeleton_frame imol* [function]
Where:

- *skeleton_frame* is a GtkWidget
- *imol* is an integer number

10.35.14 get-text-for-skeletonization-level-entry

`get-text-for-skeletonization-level-entry` [function]

10.35.15 set-skeletonization-level-from-widget

`set-skeletonization-level-from-widget` *txt* [function]
Where *txt* is a string

10.35.16 get-text-for-skeleton-box-size-entry

`get-text-for-skeleton-box-size-entry` [function]

10.35.17 set-skeleton-box-size-from-widget

`set-skeleton-box-size-from-widget` *txt* [function]
Where *txt* is a string

10.35.18 set-skeleton-box-size

`set-skeleton-box-size` *f* [function]
Where *f* is a number

the box size (in Angstroms) for which the skeleton is displayed

10.36 Skeleton Colour

10.36.1 handle-skeleton-colour-change

`handle-skeleton-colour-change` *mol map_col* [function]
Where:

- *mol* is an integer number
- *map_col* is a number

10.36.2 set-skeleton-colour

`set-skeleton-colour` *imol r g b* [function]

Where:

- *imol* is an integer number
- *r* is a number
- *g* is a number
- *b* is a number

10.36.3 get-skeleton-colour

`get-skeleton-colour` [function]

10.37 Read Maps

10.37.1 handle-read-ccp4-map

`handle-read-ccp4-map` *filename is_diff_map_flag* [function]

Where:

- *filename* is a string
- *is_diff_map_flag* is an integer number

read a CCP4 map or a CNS map (despite the name).

10.38 Save Coordinates

10.38.1 save-coordinates-using-widget

`save-coordinates-using-widget` *widget* [function]

Where *widget* is a GtkWidget

10.38.2 save-coordinates

`save-coordinates` *imol filename* [function]

Where:

- *imol* is an integer number
- *filename* is a string

save coordinates of molecule number imol in filename

Returns: status 1 is good (success), 0 is fail.

10.38.3 set-save-coordinates-in-original-directory

`set-save-coordinates-in-original-directory` *i* [function]

Where *i* is an integer number

10.38.4 save-molecule-coords-button-select

`save-molecule-coords-button-select` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.38.5 save-molecule-number-from-option-menu

`save-molecule-number-from-option-menu` [function]

10.38.6 set-save-molecule-number

`set-save-molecule-number` *imol* [function]

Where *imol* is an integer number

10.39 Read Phases File Functions

10.39.1 read-phs-and-coords-and-make-map

`read-phs-and-coords-and-make-map` *pdb_filename* [function]

Where *pdb_filename* is a string

read phs file use coords to get cell and symm to make map

uses pending data to make the map.

10.39.2 read-phs-and-make-map-using-cell-symm-from-previous-mol

`read-phs-and-make-map-using-cell-symm-from-previous-mol` [function]
phs_filename

Where *phs_filename* is a string

read a phs file, the cell and symm information is from previously read (most recently read) coordinates file

For use with phs data filename provided on the command line

10.39.3 read-phs-and-make-map-using-cell-symm-from-mol

`read-phs-and-make-map-using-cell-symm-from-mol` *phs_filename* [function]
imol

Where:

- *phs_filename* is a string
- *imol* is an integer number

read phs file and use a previously read molecule to provide the cell and symmetry information

Returns: the new molecule number, return -1 if problem creating the map (e.g. not phs data, file not found etc).

10.39.4 read-phs-and-make-map-using-cell-symm-from-mol-using-implicit-phs-filename

`read-phs-and-make-map-using-cell-symm-from-mol-using-implicit-phs-filename` *imol* [function]

Where *imol* is an integer number

10.39.5 read-phs-and-make-map-using-cell-symm

`read-phs-and-make-map-using-cell-symm` *phs_file_name hm_spacegroup a b c alpha beta gamma* [function]

Where:

- *phs_file_name* is a string
- *hm_spacegroup* is a string
- *a* is a number
- *b* is a number
- *c* is a number
- *alpha* is a number
- *beta* is a number
- *gamma* is a number

read phs file use coords to use cell and symm to make map

in degrees

10.39.6 read-phs-and-make-map-with-reso-limits

`read-phs-and-make-map-with-reso-limits` *imol phs_file_name reso_lim_low reso_lim_high* [function]

Where:

- *imol* is an integer number
- *phs_file_name* is a string
- *reso_lim_low* is a number
- *reso_lim_high* is a number

read a phs file and use the cell and symm in molecule number imol and use the resolution limits reso_lim_high (in Angstroems).

10.39.7 graphics-store-phs-filename

`graphics-store-phs-filename` *phs_filename* [function]

Where *phs_filename* is a string

10.39.8 graphics-get-phs-filename

`graphics-get-phs-filename` [function]

10.39.9 possible-cell-symm-for-phs-file

`possible-cell-symm-for-phs-file` [function]

10.39.10 get-text-for-phs-cell-chooser

`get-text-for-phs-cell-chooser` *imol field* [function]

Where:

- *imol* is an integer number
- *field* is a string

10.40 Graphics Move

10.40.1 undo-last-move

`undo-last-move` [function]

undo last move

10.40.2 translate-molecule-by

`translate-molecule-by` *imol x y z* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number

translate molecule number imol by (x,y,z) in Angstroms

10.40.3 transform-molecule-by

`transform-molecule-by` *imol m11 m12 m13 m21 m22 m23 m31 m32 m33 x y z* [function]

Where:

- *imol* is an integer number
- *m11* is a number
- *m12* is a number
- *m13* is a number
- *m21* is a number
- *m22* is a number
- *m23* is a number
- *m31* is a number
- *m32* is a number
- *m33* is a number
- *x* is a number
- *y* is a number
- *z* is a number

transform molecule number imol by the given rotation matrix, then translate by (x,y,z) in Angstroms

10.41 Go To Atom Widget Functions

10.41.1 wrapped-create-goto-atom-window

`wrapped-create-goto-atom-window` [function]

10.41.2 post-go-to-atom-window

`post-go-to-atom-window` [function]

Post the Go To Atom Window.

10.41.3 fill-go-to-atom-window

`fill-go-to-atom-window` *widget* [function]

Where *widget* is a GtkWidget

10.41.4 go-to-atom-molecule-number

`go-to-atom-molecule-number` [function]

10.41.5 go-to-atom-chain-id

`go-to-atom-chain-id` [function]

10.41.6 go-to-atom-atom-name

`go-to-atom-atom-name` [function]

10.41.7 go-to-atom-residue-number

`go-to-atom-residue-number` [function]

10.41.8 go-to-atom-ins-code

`go-to-atom-ins-code` [function]

10.41.9 go-to-atom-alt-conf

`go-to-atom-alt-conf` [function]

10.41.10 set-go-to-atom-chain-residue-atom-name

`set-go-to-atom-chain-residue-atom-name` *t1_chain_id iresno t3_atom_name* [function]

Where:

- *t1_chain_id* is a string
- *iresno* is an integer number
- *t3_atom_name* is a string

set the go to atom specification

It seems important for swig that the char * arguments are const char *, not const gchar * (or else we get wrong type of argument error on (say) "A"

10.41.11 set-go-to-atom-chain-residue-atom-name-no-redraw

`set-go-to-atom-chain-residue-atom-name-no-redraw` *t1 iresno t3 make_the_move_flag* [function]

Where:

- *t1* is a string
- *iresno* is an integer number
- *t3* is a string
- *make_the_move_flag* is an integer number

10.41.12 set-go-to-atom-chain-residue-atom-name-strings

`set-go-to-atom-chain-residue-atom-name-strings` *t1 t2 txt* [function]

Where:

- *t1* is a string
- *t2* is a string
- *txt* is a string

10.41.13 goto-next-atom-maybe-new

`goto-next-atom-maybe-new` *window* [function]

Where *window* is a GtkWidget

10.41.14 goto-previous-atom-maybe-new

`goto-previous-atom-maybe-new` *window* [function]

Where *window* is a GtkWidget

10.41.15 update-go-to-atom-from-current-position

`update-go-to-atom-from-current-position` [function]

update the Go To Atom widget entries to atom closest to screen centre.

10.41.16 apply-go-to-atom-values

`apply-go-to-atom-values` *window* [function]

Where *window* is a GtkWidget

10.41.17 update-go-to-atom-residue-list

`update-go-to-atom-residue-list` *imol* [function]

Where *imol* is an integer number

10.41.18 atom-spec-to-atom-index

`atom-spec-to-atom-index` *mol chain resno atom_name* [function]

Where:

- *mol* is an integer number

- *chain* is a string
- *resno* is an integer number
- *atom_name* is a string

what is the atom index of the given atom?

10.41.19 full-atom-spec-to-atom-index

`full-atom-spec-to-atom-index` *imol chain resno inscode atom_name altloc* [function]

Where:

- *imol* is an integer number
- *chain* is a string
- *resno* is an integer number
- *inscode* is a string
- *atom_name* is a string
- *altloc* is a string

what is the atom index of the given atom?

10.41.20 update-go-to-atom-window-on-changed-mol

`update-go-to-atom-window-on-changed-mol` *imol* [function]

Where *imol* is an integer number

update the Go To Atom window

10.41.21 update-go-to-atom-window-on-new-mol

`update-go-to-atom-window-on-new-mol` [function]

update the Go To Atom window. This updates the option menu for the molecules.

10.41.22 update-go-to-atom-window-on-other-molecule-chosen

`update-go-to-atom-window-on-other-molecule-chosen` *imol* [function]

Where *imol* is an integer number

10.41.23 set-go-to-atom-molecule

`set-go-to-atom-molecule` *imol* [function]

Where *imol* is an integer number

set the molecule for the Go To Atom

For dynarama callback sake. The widget/class knows which mapview molecule that it was generated from, so in order to go to the molecule from dynarama, we first need to the the molecule - because

does not mention the molecule (see "Next/Previous Residue" for reasons for that). This function simply calls the graphics_info_t function of the same name.

Also used in scripting, where go-to-atom-chain-residue-atom-name does not mention the molecule number.

10.41.24 go-to-atom-molecule-optionmenu-active-molecule

`go-to-atom-molecule-optionmenu-active-molecule` *widget* [function]
Where *widget* is a GtkWidget

10.41.25 save-go-to-atom-widget

`save-go-to-atom-widget` *widget* [function]
Where *widget* is a GtkWidget

10.41.26 unset-go-to-atom-widget

`unset-go-to-atom-widget` [function]

10.41.27 clear-atom-list

`clear-atom-list` *atom_gtklist* [function]
Where *atom_gtklist* is a GtkWidget

10.41.28 apply-go-to-atom-from-widget

`apply-go-to-atom-from-widget` *widget* [function]
Where *widget* is a GtkWidget

10.41.29 on-go-to-atom-residue-list-select-child

`on-go-to-atom-residue-list-select-child` *list widget user_data* [function]
Where:

- *list* is a GtkList
- *widget* is a GtkWidget
- *user_data* is an integer number

10.41.30 on-go-to-atom-residue-tree-selection-changed-gtk1

`on-go-to-atom-residue-tree-selection-changed-gtk1` *gtktree user_data* [function]
Where:

- *gtktree* is a GtkList
- *user_data* is an integer number

10.41.31 on-go-to-atom-atom-list-selection-changed-gtk1

`on-go-to-atom-atom-list-selection-changed-gtk1` *list user_data* [function]
Where:

- *list* is a GtkList
- *user_data* is an integer number

10.41.32 on-go-to-atom-residue-list-unselect-child

`on-go-to-atom-residue-list-unselect-child` *list widget user_data* [function]
Where:

- *list* is a GtkList
- *widget* is a GtkWidget
- *user_data* is an integer number

10.42 Map and Molecule Control

10.42.1 save-display-control-widget-in-graphics

`save-display-control-widget-in-graphics` *widget* [function]
Where *widget* is a GtkWidget

10.42.2 wrapped-create-display-control-window

`wrapped-create-display-control-window` [function]

10.42.3 post-display-control-window

`post-display-control-window` [function]
display the Display Constrol window

10.42.4 add-map-display-control-widgets

`add-map-display-control-widgets` [function]

10.42.5 add-mol-display-control-widgets

`add-mol-display-control-widgets` [function]

10.42.6 add-map-and-mol-display-control-widgets

`add-map-and-mol-display-control-widgets` [function]

10.42.7 reset-graphics-display-control-window

`reset-graphics-display-control-window` [function]

10.42.8 close-graphics-display-control-window

`close-graphics-display-control-window` [function]

10.42.9 set-map-displayed

`set-map-displayed` *imol state* [function]
Where:

- *imol* is an integer number
- *state* is an integer number

make the map displayed/undisplayed, 0 for off, 1 for on

10.42.10 set-mol-displayed

`set-mol-displayed` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

make the coordinates molecule displayed/undisplayed, 0 for off, 1 for on

10.42.11 set-mol-active

`set-mol-active` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

make the coordinates molecule active/inactve (clickable), 0 for off, 1 for on

10.42.12 mol-is-displayed

`mol-is-displayed` *imol* [function]

Where *imol* is an integer number

return the display state of molecule number imol

Returns: 1 for on, 0 for off

10.42.13 mol-is-active

`mol-is-active` *imol* [function]

Where *imol* is an integer number

return the active state of molecule number imol

Returns: 1 for on, 0 for off

10.42.14 map-is-displayed

`map-is-displayed` *imol* [function]

Where *imol* is an integer number

return the display state of molecule number imol

Returns: 1 for on, 0 for off

10.42.15 set-all-maps-displayed

`set-all-maps-displayed` *on_or_off* [function]

Where *on_or_off* is an integer number

if on_or_off is 0 turn off all maps displayed, for other values of on_or_off turn on all maps

10.42.16 set-all-models-displayed-and-active

`set-all-models-displayed-and-active` *on_or_off* [function]

Where *on_or_off* is an integer number

if on_or_off is 0 turn off all models displayed and active, for other values of on_or_off turn on all models.

10.42.17 show-spacegroup

`show-spacegroup` *imol* [function]

Where *imol* is an integer number

return the spacegroup of molecule number imol

Returns: "No Spacegroup" when the spacegroup of a molecule has not been set.

10.43 Align and Mutate

10.43.1 wrapped-create-align-and-mutate-dialog

`wrapped-create-align-and-mutate-dialog` [function]

10.43.2 do-align-mutate-sequence

`do-align-mutate-sequence` *w* [function]

Where *w* is a GtkWidget

10.43.3 align-and-mutate-molecule-menu-item-activate

`align-and-mutate-molecule-menu-item-activate` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.43.4 align-and-mutate-chain-option-menu-item-activate

`align-and-mutate-chain-option-menu-item-activate` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.43.5 align-and-mutate

`align-and-mutate` *imol chain_id fasta_maybe* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *fasta_maybe* is a string

10.44 Renumber Residue Range

10.44.1 renumber-residue-range

`renumber-residue-range` *imol chain_id start_res last_res offset* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *start_res* is an integer number
- *last_res* is an integer number
- *offset* is an integer number

renumber the given residue range by offset residues

10.44.2 wrapped-create-renumber-residue-range-dialog

`wrapped-create-renumber-residue-range-dialog` [function]

10.44.3 renumber-residues-from-widget

`renumber-residues-from-widget` *window* [function]

Where *window* is a GtkWidget

10.44.4 change-residue-number

`change-residue-number` *imol chain_id current_resno current_inscode new_resno new_inscode* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *current_resno* is an integer number
- *current_inscode* is a string
- *new_resno* is an integer number
- *new_inscode* is a string

change chain id, residue number or insertion code for given residue

10.45 Scripting Interface

10.45.1 post-scripting-window

`post-scripting-window` [function]

do nothing - compatibility function

10.45.2 post-scheme-scripting-window

`post-scheme-scripting-window` [function]

pop-up a scripting window for scheming

10.45.3 post-python-scripting-window

`post-python-scripting-window` [function]
pop-up a scripting window for pythoning

10.45.4 run-command-line-scripts

`run-command-line-scripts` [function]

10.45.5 setup-guile-window-entry

`setup-guile-window-entry` *entry* [function]
Where *entry* is a GtkWidget

10.45.6 setup-python-window-entry

`setup-python-window-entry` *entry* [function]
Where *entry* is a GtkWidget

10.45.7 set-guile-gui-loaded-flag

`set-guile-gui-loaded-flag` [function]

10.45.8 set-python-gui-loaded-flag

`set-python-gui-loaded-flag` [function]

10.45.9 set-found-coot-gui

`set-found-coot-gui` [function]

10.45.10 set-found-coot-python-gui

`set-found-coot-python-gui` [function]

10.46 Monomer

10.46.1 handle-get-accession-code

`handle-get-accession-code` *widget* [function]
Where *widget* is a GtkWidget

10.46.2 wrapped-create-libcheck-monomer-dialog

`wrapped-create-libcheck-monomer-dialog` [function]

10.46.3 handle-get-libcheck-monomer-code

`handle-get-libcheck-monomer-code` *widget* [function]
Where *widget* is a GtkWidget

10.46.4 get-monomer

`get-monomer` *three_letter_code* [function]

Where *three_letter_code* is a string

import libcheck monomer give the 3-letter code.

Returns: the new molecule number, if not -1 (error).

10.46.5 get-monomer-from-dictionary

`get-monomer-from-dictionary` *three_letter_code idealised_flag* [function]

Where:

- *three_letter_code* is a string
- *idealised_flag* is an integer number

10.46.6 handle-make-monomer-search

`handle-make-monomer-search` *text viewport* [function]

Where:

- *text* is a string
- *viewport* is a GtkWidget

10.46.7 run-script

`run-script` *filename* [function]

Where *filename* is a string

run script file

10.46.8 run-guile-script

`run-guile-script` *filename* [function]

Where *filename* is a string

10.46.9 run-python-script

`run-python-script` *filename* [function]

Where *filename* is a string

10.47 Regularization and Refinement

10.47.1 do-regularize

`do-regularize` *state* [function]

Where *state* is an integer number

10.47.2 do-refine

`do-refine` *state* [function]

Where *state* is an integer number

10.47.3 do-regularize-kill-delete-dialog

`do-regularize-kill-delete-dialog` [function]

10.47.4 add-planar-peptide-restraints

`add-planar-peptide-restraints` [function]

add a restraint on peptides to make them planar

This adds a 5 atom restraint that includes both CA atoms of the peptide. Use this rather than editting the mon_lib_list.cif file.

10.47.5 remove-planar-peptide-restraints

`remove-planar-peptide-restraints` [function]

remove restraints on peptides to make them planar.

10.47.6 planar-peptide-restraints-state

`planar-peptide-restraints-state` [function]

10.47.7 add-omega-torsion-restriants

`add-omega-torsion-restriants` [function]

add restraints on the omega angle of the peptides

(that is the torsion round the peptide bond). Omega angles that are closer to 0 than to 180 will be refined as cis peptides (and of course if omega is greater than 90 then the peptide will be refined as a trans peptide (this is the normal case).

10.47.8 remove-omega-torsion-restriants

`remove-omega-torsion-restriants` [function]

remove omega restraints on CIS and TRANS linked residues.

10.47.9 set-refinement-immediate-replacement

`set-refinement-immediate-replacement` *istate* [function]

Where *istate* is an integer number

set immediate replacement mode for refinement and regularization. You need this (call with istate=1) if you are scripting refinement/regularization

10.47.10 refinement-immediate-replacement-state

`refinement-immediate-replacement-state` [function]

query the state of the immediate replacement mode

10.47.11 set-residue-selection-flash-frames-number

`set-residue-selection-flash-frames-number` *i* [function]

Where *i* is an integer number

set the number of frames for which the selected residue range flashes

On fast computers, this can be set to higher than the default for more aesthetic appeal.

10.47.12 accept-regularizement

`accept-regularizement` [function]

accept the new positions of the regularized or refined residues

If you are scripting refinement and/or regularization, this is the function that you need to call after refine-zone or regularize-zone.

10.47.13 clear-up-moving-atoms

`clear-up-moving-atoms` [function]

10.47.14 clear-moving-atoms-object

`clear-moving-atoms-object` [function]

10.47.15 fill-option-menu-with-refine-options

`fill-option-menu-with-refine-options` *option_menu* [function]

Where *option_menu* is a GtkWidget

10.47.16 wrapped-create-refine-params-dialog

`wrapped-create-refine-params-dialog` [function]

10.47.17 do-torsions-toggle

`do-torsions-toggle` *button* [function]

Where *button* is a GtkWidget

10.47.18 set-refine-with-torsion-restraints

`set-refine-with-torsion-restraints` *istate* [function]

Where *istate* is an integer number

turn on (or off) torsion restraints

Pass with istate=1 for on, istate=0 for off.

10.47.19 refine-with-torsion-restraints-state

`refine-with-torsion-restraints-state` [function]

return the state of above

10.47.20 set-refine-params-toggle-buttons

`set-refine-params-toggle-buttons` *button* [function]

Where *button* is a GtkWidget

10.47.21 set-matrix

`set-matrix` *f* [function]

Where *f* is a number

set the relative weight of the geometric terms to the map terms

The default is 60.

The higher the number the more weight that is given to the map terms but the resulting chi squared values are higher). This will be needed for maps generated from data not on (or close to) the absolute scale or maps that have been scaled (for example so that the sigma level has been scaled to 1.0).

10.47.22 matrix-state

`matrix-state` [function]

return the relative weight of the geometric terms to the map terms.

10.47.23 set-refine-auto-range-step

`set-refine-auto-range-step` *i* [function]

Where *i* is an integer number

change the +/- step for autoranging (default is 1)

Auto-ranging alow you to select a range from one button press, this allows you to set the number of residues either side of the clicked residue that becomes the selected zone

10.47.24 set-refine-max-residues

`set-refine-max-residues` *n* [function]

Where *n* is an integer number

set the heuristic fencepost for the maximum number of residues in the refinement/regularization residue range

Default is 20

10.47.25 refine-zone-atom-index-define

`refine-zone-atom-index-define` *imol ind1 ind2* [function]

Where:

- *imol* is an integer number
- *ind1* is an integer number
- *ind2* is an integer number

refine a zone based on atom indexing

10.47.26 refine-zone

`refine-zone` *imol chain_id resno1 resno2 altconf* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno1* is an integer number
- *resno2* is an integer number
- *altconf* is a string

refine a zone

presumes that imol_Refinement_Map has been set

10.47.27 refine-zone-with-full-residue-spec

`refine-zone-with-full-residue-spec` *imol chain_id resno1 inscode_1 resno2 inscode_2 altconf* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno1* is an integer number
- *inscode_1* is a string
- *resno2* is an integer number
- *inscode_2* is a string
- *altconf* is a string

refine a zone, allowing the specification of insertion codes for the residues too.

presumes that imol_Refinement_Map has been set

10.47.28 refine-auto-range

`refine-auto-range` *imol chain_id resno1 altconf* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno1* is an integer number
- *altconf* is a string

refine a zone using auto-range

presumes that imol_Refinement_Map has been set

10.47.29 regularize-zone

`regularize-zone` *imol chain_id resno1 resno2 altconf* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno1* is an integer number
- *resno2* is an integer number
- *altconf* is a string

regularize a zone

10.47.30 set-dragged-refinement-steps-per-frame

`set-dragged-refinement-steps-per-frame` *v* [function]

Where *v* is an integer number

set the number of refinement steps applied to the intermediate atoms each frame of graphics.

smaller numbers make the movement of the intermediate atoms slower, smoother, more elegant.

Default: 50.

10.47.31 dragged-refinement-steps-per-frame

`dragged-refinement-steps-per-frame` [function]

return the number of steps per frame in dragged refinement

10.47.32 set-refinement-refine-per-frame

`set-refinement-refine-per-frame` *istate* [function]

Where *istate* is an integer number

allow refinement of intermediate atoms after dragging, before displaying (default: 0, off).

An attempt to do something like xfit does, at the request of Frank von Delft.

Pass with istate=1 to enable this option.

10.47.33 refinement-refine-per-frame-state

`refinement-refine-per-frame-state` [function]

query the state of the above option

10.47.34 set-refine-ramachandran-angles

`set-refine-ramachandran-angles` *state* [function]

Where *state* is an integer number

turn on Ramachandran angles refinement in refinement and regularization

10.47.35 refine-ramachandran-angles-state

`refine-ramachandran-angles-state` [function]

10.47.36 set-numerical-gradients

`set-numerical-gradients` *istate* [function]

Where *istate* is an integer number

10.47.37 set-fix-chiral-volumes-before-refinement

`set-fix-chiral-volumes-before-refinement` *istate* [function]

Where *istate* is an integer number

correct the sign of chiral volumes before commencing refinement?

Do we want to fix chiral volumes (by moving the chiral atom to the other side of the chiral plane if necessary). Default yes (1). Note: doesn't work currently.

10.47.38 check-chiral-volumes

`check-chiral-volumes` *imol* [function]

Where *imol* is an integer number

query the state of the above option

10.47.39 check-chiral-volumes-from-widget

`check-chiral-volumes-from-widget` *window* [function]

Where *window* is a GtkWidget

10.47.40 fill-chiral-volume-molecule-option-menu

`fill-chiral-volume-molecule-option-menu` *w* [function]

Where *w* is a GtkWidget

10.47.41 chiral-volume-molecule-option-menu-item-select

`chiral-volume-molecule-option-menu-item-select` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.47.42 set-show-chiral-volume-errors-dialog

`set-show-chiral-volume-errors-dialog` *istate* [function]

Where *istate* is an integer number

For experienced Cooters who don't like Coot nannying about chiral volumes during refinement.

10.47.43 set-secondary-structure-restraints-type

`set-secondary-structure-restraints-type` *itype* [function]

Where *itype* is an integer number

set the type of secondary structure restraints

0 no sec str restraints

1 alpha helix restraints

2 beta strand restraints

10.47.44 secondary-structure-restraints-type

`secondary-structure-restraints-type` [function]

return the secondary structure restraints type

10.47.45 imol-refinement-map

`imol-refinement-map` [function]

the molecule number of the map used for refinement

Returns: the map number, if it has been set or there is only one map, return -1 on no map set (ambiguous) or no maps.

10.47.46 set-imol-refinement-map

`set-imol-refinement-map` *imol* [function]

Where *imol* is an integer number

set the molecule number of the map to be used for refinement/fitting.

Returns: imol on success, -1 on failure

10.47.47 does-residue-exist-p

`does-residue-exist-p` *imol chain_id resno inscode* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *inscode* is a string

Does the residue exist? (Raw function).

Returns: 0 on not-exist, 1 on does exist.

10.47.48 write-restraints-cif-dictionary

`write-restraints-cif-dictionary` *monomer_type file_name* [function]

Where:

- *monomer_type* is a string
- *file_name* is a string

10.48 Simplex Refinement Interface

10.48.1 fit-residue-range-to-map-by-simplex

`fit-residue-range-to-map-by-simplex` *res1 res2 altloc chain_id imol imol_for_map* [function]

Where:

- *res1* is an integer number
- *res2* is an integer number
- *altloc* is a string
- *chain_id* is a string
- *imol* is an integer number
- *imol_for_map* is an integer number

refine residue range using simplex optimization

10.48.2 score-residue-range-fit-to-map

`score-residue-range-fit-to-map` *res1 res2 altloc chain_id imol imol_for_map* [function]

Where:

- *res1* is an integer number
- *res2* is an integer number
- *altloc* is a string
- *chain_id* is a string
- *imol* is an integer number
- *imol_for_map* is an integer number

simply score the residue range fit to map

10.49 Nomenclature Errors

10.49.1 fix-nomenclature-errors

`fix-nomenclature-errors` *imol* [function]

Where *imol* is an integer number

fix nomenclature errors in molecule number imol

Returns: the number of resides altered.

10.50 move molecule here (wrapper to scheme function)

10.50.1 wrapped-create-move-molecule-here-dialog

`wrapped-create-move-molecule-here-dialog` [function]

10.50.2 move-molecule-here-by-widget

`move-molecule-here-by-widget` *w* [function]

Where *w* is a GtkWidget

10.50.3 move-molecule-to-screen-centre-internal

`move-molecule-to-screen-centre-internal` *imol* [function]

Where *imol* is an integer number

10.51 Atom Info Interface

10.51.1 output-atom-info-as-text

`output-atom-info-as-text` *imol chain_id resno ins_code atname altconf* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string

- *resno* is an integer number
- *ins_code* is a string
- *atname* is a string
- *altconf* is a string

output Atom Info for the give atom specs

Actually I want to return a scheme object with occ, pos, b-factor info

10.52 Residue Info

10.52.1 do-residue-info-dialog

`do-residue-info-dialog` [function]

10.52.2 output-residue-info-dialog

`output-residue-info-dialog` *atom_index imol* [function]

Where:

- *atom_index* is an integer number
- *imol* is an integer number

10.52.3 residue-info-dialog

`residue-info-dialog` *imol chain_id resno ins_code* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string

10.52.4 residue-info-dialog-is-displayed

`residue-info-dialog-is-displayed` [function]

10.52.5 output-residue-info-as-text

`output-residue-info-as-text` *atom_index imol* [function]

Where:

- *atom_index* is an integer number
- *imol* is an integer number

10.52.6 apply-residue-info-changes

`apply-residue-info-changes` *widget* [function]

Where *widget* is a GtkWidget

10.52.7 do-distance-define

`do-distance-define` [function]

10.52.8 do-angle-define

`do-angle-define` [function]

10.52.9 do-torsion-define

`do-torsion-define` [function]

10.52.10 residue-info-apply-all-checkbutton-toggled

`residue-info-apply-all-checkbutton-toggled` [function]

10.52.11 wrapped-create-residue-info-dialog

`wrapped-create-residue-info-dialog` [function]

10.52.12 clear-residue-info-edit-list

`clear-residue-info-edit-list` [function]

10.52.13 residue-info-release-memory

`residue-info-release-memory` *widget* [function]
Where *widget* is a GtkWidget

10.52.14 unset-residue-info-widget

`unset-residue-info-widget` [function]

10.52.15 clear-simple-distances

`clear-simple-distances` [function]

10.52.16 clear-last-simple-distance

`clear-last-simple-distance` [function]

10.52.17 wrapped-create-geometry-dialog

`wrapped-create-geometry-dialog` [function]

10.52.18 store-geometry-dialog

`store-geometry-dialog` *w* [function]
Where *w* is a GtkWidget

10.53 Pointer Functions

10.53.1 fill-pointer-distances-widget

`fill-pointer-distances-widget` *widget* [function]
Where *widget* is a GtkWidget

10.53.2 execute-pointer-distances-settings

`execute-pointer-distances-settings` *widget* [function]
Where *widget* is a GtkWidget

10.53.3 toggle-pointer-distances-show-distances

`toggle-pointer-distances-show-distances` *button* [function]
Where *button* is a GtkToggleButton

10.53.4 set-show-pointer-distances

`set-show-pointer-distances` *istate* [function]
Where *istate* is an integer number

10.54 Zoom Functions

10.54.1 scale-zoom

`scale-zoom` *f* [function]
Where *f* is a number

scale the view by f

external (scripting) interface (with redraw)

10.54.2 scale-zoom-internal

`scale-zoom-internal` *f* [function]
Where *f* is a number

10.54.3 zoom-factor

`zoom-factor` [function]
return the current zoom factor

10.54.4 set-smooth-scroll-do-zoom

`set-smooth-scroll-do-zoom` *i* [function]
Where *i* is an integer number

set smooth scroll with zoom

10.54.5 smooth-scroll-do-zoom

`smooth-scroll-do-zoom` [function]
return the state of the above system

10.54.6 smooth-scroll-zoom-limit

`smooth-scroll-zoom-limit` [function]

10.54.7 set-smooth-scroll-zoom-limit

`set-smooth-scroll-zoom-limit` *f* [function]
Where *f* is a number

10.54.8 set-zoom-adjustment

`set-zoom-adjustment` *w* [function]
Where *w* is a GtkWidget

10.54.9 set-zoom

`set-zoom` *f* [function]
Where *f* is a number

10.55 CNS Data Functions

10.55.1 handle-cns-data-file

`handle-cns-data-file` *filename imol* [function]
Where:

- *filename* is a string
- *imol* is an integer number

read CNS data (currently only a placeholder)

10.55.2 handle-cns-data-file-with-cell

`handle-cns-data-file-with-cell` *filename imol a b c alpha beta gamma spg_info* [function]
Where:

- *filename* is a string
- *imol* is an integer number
- *a* is a number
- *b* is a number
- *c* is a number
- *alpha* is a number
- *beta* is a number
- *gamma* is a number
- *spg_info* is a string

read CNS data (currently only a placeholder)

a, b,c are in Angstroems. alpha, beta, gamma are in degrees. spg is the space group info, either ;-delimited symmetry operators or the space group name

10.56 mmCIF Functions

10.56.1 auto-read-cif-data-with-phases

`auto-read-cif-data-with-phases` *filename* [function]
Where *filename* is a string

10.56.2 read-cif-data-with-phases-sigmaa

`read-cif-data-with-phases-sigmaa` *filename* [function]
Where *filename* is a string

10.56.3 read-cif-data-with-phases-diff-sigmaa

`read-cif-data-with-phases-diff-sigmaa` *filename* [function]
Where *filename* is a string

10.56.4 read-cif-data

`read-cif-data` *filename imol_coords* [function]
Where:

- *filename* is a string
- *imol_coords* is an integer number

10.56.5 read-cif-data-2fofc-map

`read-cif-data-2fofc-map` *filename imol_coords* [function]
Where:

- *filename* is a string
- *imol_coords* is an integer number

10.56.6 read-cif-data-fofc-map

`read-cif-data-fofc-map` *filename imol_coords* [function]
Where:

- *filename* is a string
- *imol_coords* is an integer number

10.56.7 read-cif-data-with-phases-fo-fc

`read-cif-data-with-phases-fo-fc` *filename* [function]
Where *filename* is a string

10.56.8 read-cif-data-with-phases-2fo-fc

`read-cif-data-with-phases-2fo-fc` *filename* [function]
Where *filename* is a string

10.56.9 read-cif-data-with-phases-nfo-fc

`read-cif-data-with-phases-nfo-fc` *filename map_type* [function]

Where:

- *filename* is a string
- *map_type* is an integer number

10.56.10 read-cif-data-with-phases-fo-alpha-calc

`read-cif-data-with-phases-fo-alpha-calc` *filename* [function]

Where *filename* is a string

10.56.11 handle-cif-dictionary

`handle-cif-dictionary` *filename* [function]

Where *filename* is a string

10.56.12 read-cif-dictionary

`read-cif-dictionary` *filename* [function]

Where *filename* is a string

10.56.13 write-connectivity

`write-connectivity` *monomer_name filename* [function]

Where:

- *monomer_name* is a string
- *filename* is a string

10.56.14 import-all-refmac-cifs

`import-all-refmac-cifs` [function]

10.57 SHELXL Functions

10.57.1 read-shelx-ins-file

`read-shelx-ins-file` *filename* [function]

Where *filename* is a string

read a SHELXL .ins file

10.57.2 write-shelx-ins-file

`write-shelx-ins-file` *imol filename* [function]

Where:

- *imol* is an integer number
- *filename* is a string

write a SHELXL .ins file for molecule number imol

10.57.3 handle-shelx-fcf-file-internal

`handle-shelx-fcf-file-internal` *filename* [function]
Where *filename* is a string

10.58 Validation Functions

10.58.1 deviant-geometry

`deviant-geometry` *imol* [function]
Where *imol* is an integer number

10.58.2 is-valid-model-molecule

`is-valid-model-molecule` *imol* [function]
Where *imol* is an integer number

10.58.3 is-valid-map-molecule

`is-valid-map-molecule` *imol* [function]
Where *imol* is an integer number

10.58.4 free-geometry-graph

`free-geometry-graph` *dialog* [function]
Where *dialog* is a GtkWidget

10.58.5 unset-geometry-graph

`unset-geometry-graph` *dialog* [function]
Where *dialog* is a GtkWidget

10.58.6 add-on-validation-graph-mol-options

`add-on-validation-graph-mol-options` *menu type_in* [function]
Where:

- *menu* is a GtkWidget
- *type_in* is a string

10.58.7 my-delete-validaton-graph-mol-option

`my-delete-validaton-graph-mol-option` *widget* [function]
Where:

- *widget* is a GtkWidget
- is a void

10.58.8 validation-graph-b-factor-mol-selector-activate

`validation-graph-b-factor-mol-selector-activate` *menuitem* *user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.9 validation-graph-geometry-mol-selector-activate

`validation-graph-geometry-mol-selector-activate` *menuitem* *user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.10 validation-graph-omega-mol-selector-activate

`validation-graph-omega-mol-selector-activate` *menuitem* *user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.11 validation-graph-rotamer-mol-selector-activate

`validation-graph-rotamer-mol-selector-activate` *menuitem* *user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.12 validation-graph-density-fit-mol-selector-activate

`validation-graph-density-fit-mol-selector-activate` *menuitem* *user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.13 gln-and-asn-b-factor-outlier-mol-selector-activate

`gln-and-asn-b-factor-outlier-mol-selector-activate` *menuitem* *user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.14 validation-graph-ncs-diffs-mol-selector-activate

`validation-graph-ncs-diffs-mol-selector-activate` *menuitem user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.15 probe-mol-selector-activate

`probe-mol-selector-activate` *menuitem user_data* [function]

Where:

- *menuitem* is a GtkMenuItem
- *user_data* is an integer number

10.58.16 difference-map-peaks

`difference-map-peaks` *imol imol_coords level do_positive_level_flag do_negative_level_flag* [function]

Where:

- *imol* is an integer number
- *imol_coords* is an integer number
- *level* is a number
- *do_positive_level_flag* is an integer number
- *do_negative_level_flag* is an integer number

generate a list of difference map peaks

10.58.17 difference-map-peaks-by-widget

`difference-map-peaks-by-widget` *dialog* [function]

Where *dialog* is a GtkWidget

10.58.18 set-difference-map-peaks-widget

`set-difference-map-peaks-widget` *w* [function]

Where *w* is a GtkWidget

10.58.19 clear-diff-map-peaks

`clear-diff-map-peaks` [function]

10.58.20 wrapped-create-generate-diff-map-peaks-dialog

`wrapped-create-generate-diff-map-peaks-dialog` [function]

10.58.21 gln-asn-b-factor-outliers

`gln-asn-b-factor-outliers` *imol* [function]

Where *imol* is an integer number

Make a gui for GLN adn ASN B-factor outiers, compairing the O and N temperatur factors difference to the distribution of temperature factors from the other atoms.

10.59 Ramachandran Plot Functions

10.59.1 do-ramachandran-plot

`do-ramachandran-plot` *imol* [function]

Where *imol* is an integer number

Ramachandran plot for molecule number imol.

10.59.2 set-kleywegt-plot-n-diffs

`set-kleywegt-plot-n-diffs` *n_diffs* [function]

Where *n_diffs* is an integer number

set the number of biggest difference arrows on the Kleywegt plot.

10.59.3 add-on-rama-choices

`add-on-rama-choices` [function]

10.59.4 set-ramachandran-plot-contour-levels

`set-ramachandran-plot-contour-levels` *level_prefered level_allowed* [function]

Where:

- *level_prefered* is a number
- *level_allowed* is a number

set the contour levels for theremachandran plot, default values are 0.02 (prefered) 0.002 (allowed)

10.59.5 set-ramachandran-plot-background-block-size

`set-ramachandran-plot-background-block-size` *blocksize* [function]

Where *blocksize* is a number

set the ramachandran plot background block size. Smaller is smoother but slower. Should be divisible exactly into 360. Default value is 10.

10.59.6 my-delete-ramachandran-mol-option

`my-delete-ramachandran-mol-option` *widget* [function]

Where:

- *widget* is a GtkWidget
- is a void

10.59.7 set-dynarama-is-displayed

`set-dynarama-is-displayed` *dynarama_widget imol* [function]

Where:

- *dynarama_widget* is a GtkWidget
- *imol* is an integer number

10.59.8 dynarama-is-displayed-state

`dynarama-is-displayed-state` *imol* [function]

Where *imol* is an integer number

10.59.9 get-mol-from-dynarama

`get-mol-from-dynarama` *window* [function]

Where *window* is a GtkWidget

10.59.10 set-moving-atoms

`set-moving-atoms` *phi psi* [function]

Where:

- *phi* is a number
- *psi* is a number

10.59.11 accept-phi-psi-moving-atoms

`accept-phi-psi-moving-atoms` [function]

10.59.12 setup-edit-phi-psi

`setup-edit-phi-psi` *state* [function]

Where *state* is an integer number

10.59.13 setup-dynamic-distances

`setup-dynamic-distances` *state* [function]

Where *state* is an integer number

10.59.14 destroy-edit-backbone-rama-plot

`destroy-edit-backbone-rama-plot` [function]

10.59.15 ramachandran-plot-differences

`ramachandran-plot-differences` *imol1 imol2* [function]

Where:

- *imol1* is an integer number
- *imol2* is an integer number

10.59.16 ramachandran-plot-differences-by-chain

`ramachandran-plot-differences-by-chain` *imol1 imol2* a_*chain* *b_chain* [function]

Where:

- *imol1* is an integer number
- *imol2* is an integer number
- *a_chain* is a string
- *b_chain* is a string

10.59.17 wrapped-ramachandran-plot-differences-dialog

`wrapped-ramachandran-plot-differences-dialog` [function]

10.59.18 do-ramachandran-plot-differences-by-widget

`do-ramachandran-plot-differences-by-widget` *w* [function]

Where *w* is a GtkWidget

10.59.19 fill-ramachandran-plot-differences-option-menu-with-chain-options

`fill-ramachandran-plot-differences-option-menu-with-chain-options` *chain_optionmenu* *is_first_mol_flag* [function]

Where:

- *chain_optionmenu* is a GtkWidget
- *is_first_mol_flag* is an integer number

10.59.20 ramachandran-plot-differences-mol-option-menu-activate-first

`ramachandran-plot-differences-mol-option-menu-activate-first` *item* *pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.59.21 ramachandran-plot-differences-mol-option-menu-activate-second

`ramachandran-plot-differences-mol-option-menu-activate-second` *item* *pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.59.22 ramachandran-plot-differences-chain-option-menu-activate-first

`ramachandran-plot-differences-chain-option-menu-activate-first` *item* *pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.59.23 ramachandran-plot-differences-chain-option-menu-activate-second

`ramachandran-plot-differences-chain-option-menu-activate-second` *item* *pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.60 Sequence View Interface

10.60.1 do-sequence-view

`do-sequence-view` *imol* [function]

Where *imol* is an integer number

10.60.2 add-on-sequence-view-choices

`add-on-sequence-view-choices` [function]

10.60.3 set-sequence-view-is-displayed

`set-sequence-view-is-displayed` *widget imol* [function]

Where:

- *widget* is a GtkWidget
- *imol* is an integer number

10.61 Atom Labelling

10.61.1 add-atom-label

`add-atom-label` *imol chain_id iresno atom_id* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *iresno* is an integer number
- *atom_id* is a string

10.61.2 remove-atom-label

`remove-atom-label` *imol chain_id iresno atom_id* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *iresno* is an integer number
- *atom_id* is a string

10.61.3 remove-all-atom-labels

`remove-all-atom-labels` [function]

10.61.4 set-label-on-recentre-flag

`set-label-on-recentre-flag` *i* [function]

Where *i* is an integer number

10.61.5 centre-atom-label-status

`centre-atom-label-status` [function]

10.61.6 set-brief-atom-labels

`set-brief-atom-labels` *istat* [function]

Where *istat* is an integer number

use brief atom names for on-screen labels

call with istat=1 to use brief labels, istat=0 for normal labels

10.61.7 brief-atom-labels-state

`brief-atom-labels-state` [function]

the brief atom label state

10.62 Screen Rotation

10.62.1 rotate-y-scene

`rotate-y-scene` *nsteps stepsize* [function]

Where:

- *nsteps* is an integer number
- *stepsize* is a number

rotate view round y axis stepsize degrees for nstep such steps

10.62.2 rotate-x-scene

`rotate-x-scene` *nsteps stepsize* [function]

Where:

- *nsteps* is an integer number
- *stepsize* is a number

rotate view round x axis stepsize degrees for nstep such steps

10.62.3 rotate-z-scene

`rotate-z-scene` *nsteps stepsize* [function]

Where:

- *nsteps* is an integer number
- *stepsize* is a number

rotate view round z axis stepsize degrees for nstep such steps

10.62.4 spin-zoom-trans

`spin-zoom-trans` *axis nstep stepsize zoom_by x_rel y_rel z_rel* [function]

Where:

- *axis* is an integer number
- *nstep* is an integer number
- *stepsize* is a number
- *zoom_by* is a number
- *x_rel* is a number
- *y_rel* is a number
- *z_rel* is a number

Bells and whistles rotation.

spin, zoom and translate.

where axis is either x,y or z, stepsize is in degrees, zoom_by and x_rel etc are how much zoom, x,y,z should have changed by after nstep steps.

10.63 Background Colour

10.63.1 set-background-colour

`set-background-colour` *red green blue* [function]

Where:

- *red* is a number
- *green* is a number
- *blue* is a number

set the background colour

red, green and blue are numbers between 0.0 and 1.0

10.63.2 redraw-background

`redraw-background` [function]
re draw the background colour when switching between mono and stereo

10.63.3 background-is-black-p

`background-is-black-p` [function]
is the background black (or nearly black)?

Returns: 1 if the background is black (or nearly black), else return 0.

10.64 Ligand Fitting Functions

10.64.1 set-ligand-acceptable-fit-fraction

`set-ligand-acceptable-fit-fraction` *f* [function]
Where *f* is a number

set the fraction of atoms which must be in positive density after a ligand fit

10.64.2 set-ligand-cluster-sigma-level

`set-ligand-cluster-sigma-level` *f* [function]
Where *f* is a number

set the default sigma level that the map is searched to find potential ligand sites

10.64.3 set-ligand-flexible-ligand-n-samples

`set-ligand-flexible-ligand-n-samples` *i* [function]
Where *i* is an integer number

set the number of conformation samples

big ligands require more samples. Default 10.

10.64.4 set-ligand-verbose-reporting

`set-ligand-verbose-reporting` *i* [function]
Where *i* is an integer number

10.64.5 set-find-ligand-n-top-ligands

`set-find-ligand-n-top-ligands` *n* [function]
Where *n* is an integer number

search the top n sites for ligands.

Default 10.

10.64.6 set-find-ligand-mask-waters

`set-find-ligand-mask-waters` *istate* [function]

Where *istate* is an integer number

how shall we treat the waters during ligand fitting?

pass with istate=1 for waters to mask the map in the same way that protein atoms do.

10.64.7 set-ligand-cluster-sigma-level-from-widget

`set-ligand-cluster-sigma-level-from-widget` *button* [function]

Where *button* is a GtkWidget

10.64.8 set-ligand-search-protein-molecule

`set-ligand-search-protein-molecule` *imol* [function]

Where *imol* is an integer number

set the protein molecule for ligand searching

10.64.9 set-ligand-search-map-molecule

`set-ligand-search-map-molecule` *imol_map* [function]

Where *imol_map* is an integer number

set the map molecule for ligand searching

10.64.10 add-ligand-search-ligand-molecule

`add-ligand-search-ligand-molecule` *imol_ligand* [function]

Where *imol_ligand* is an integer number

add a rigid ligand molecule to the list of ligands to search for in ligand searching

10.64.11 add-ligand-search-wiggly-ligand-molecule

`add-ligand-search-wiggly-ligand-molecule` *imol_ligand* [function]

Where *imol_ligand* is an integer number

add a flexible ligand molecule to the list of ligands to search for in ligand searching

10.64.12 free-ligand-search-user-data

`free-ligand-search-user-data` *button* [function]

Where *button* is a GtkWidget

10.64.13 add-ligand-clear-ligands

`add-ligand-clear-ligands` [function]

10.64.14 ligand-expert

`ligand-expert` [function]

this sets the flag to have expert option ligand entries in the Ligand Searching dialog

10.64.15 do-find-ligands-dialog

`do-find-ligands-dialog` [function]
display the find ligands dialog, if maps, coords and ligands are available

10.64.16 fill-ligands-dialog

`fill-ligands-dialog` *dialog* [function]
Where *dialog* is a GtkWidget

10.64.17 fill-ligands-dialog-map-bits

`fill-ligands-dialog-map-bits` *dialog diff_maps_only_flag* [function]
Where:

- *dialog* is a GtkWidget
- *diff_maps_only_flag* is an integer number

10.64.18 fill-ligands-dialog-protein-bits

`fill-ligands-dialog-protein-bits` *dialog* [function]
Where *dialog* is a GtkWidget

10.64.19 fill-ligands-dialog-ligands-bits

`fill-ligands-dialog-ligands-bits` *dialog* [function]
Where *dialog* is a GtkWidget

10.64.20 do-find-ligand-many-atoms-in-ligands

`do-find-ligand-many-atoms-in-ligands` *find_ligand_dialog* [function]
Where *find_ligand_dialog* is a GtkWidget

10.64.21 fill-ligands-dialog-map-bits-by-dialog-name

`fill-ligands-dialog-map-bits-by-dialog-name` *find_ligand_dialog* [function]
dialog_name diff_maps_only_flag
Where:

- *find_ligand_dialog* is a GtkWidget
- *dialog_name* is a string
- *diff_maps_only_flag* is an integer number

10.64.22 fill-ligands-dialog-protein-bits-by-dialog-name

`fill-ligands-dialog-protein-bits-by-dialog-name` [function]
find_ligand_dialog dialog_name
Where:

- *find_ligand_dialog* is a GtkWidget
- *dialog_name* is a string

10.64.23 fill-vbox-with-coords-options-by-dialog-name

`fill-vbox-with-coords-options-by-dialog-name` *find_ligand_dialog dialog_name have_ncs_flag* [function]

Where:

- *find_ligand_dialog* is a GtkWidget
- *dialog_name* is a string
- *have_ncs_flag* is an integer number

10.64.24 fill-ligands-sigma-level-entry

`fill-ligands-sigma-level-entry` *dialog* [function]

Where *dialog* is a GtkWidget

10.64.25 fill-ligands-expert-options

`fill-ligands-expert-options` *find_ligand_dialog* [function]

Where *find_ligand_dialog* is a GtkWidget

10.64.26 set-ligand-expert-options-from-widget

`set-ligand-expert-options-from-widget` *button* [function]

Where *button* is a GtkWidget

10.64.27 execute-get-mols-ligand-search

`execute-get-mols-ligand-search` *button* [function]

Where *button* is a GtkWidget

"Template"-based matching. Overlap the first residue in imol_ligand onto the residue specified by the reference parameters. Use graph matching, not atom names.

otherwise return the RT operator

Returns: success status, f = failed to find residue in either imol_ligand or imo_ref,

10.64.28 free-blob-dialog-memory

`free-blob-dialog-memory` *w* [function]

Where *w* is a GtkWidget

10.64.29 flip-ligand

`flip-ligand` *imol chain_id resno* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number

10.65 Water Fitting Functions

10.65.1 renumber-waters

`renumber-waters` *imol* [function]

Where *imol* is an integer number

Renumber the waters of molecule number imol with consecutive numbering

10.65.2 fill-find-waters-dialog

`fill-find-waters-dialog` *find_ligand_dialog* [function]

Where *find_ligand_dialog* is a GtkWidget

10.65.3 execute-find-waters

`execute-find-waters` *ok_button* [function]

Where *ok_button* is a GtkWidget

10.65.4 execute-find-waters-real

`execute-find-waters-real` *imol_for_map imol_for_protein new_waters_mol_flag sigma_cut_off* [function]

Where:

- *imol_for_map* is an integer number
- *imol_for_protein* is an integer number
- *new_waters_mol_flag* is an integer number
- *sigma_cut_off* is a number

find waters

10.65.5 find-waters

`find-waters` *imol_for_map imol_for_protein new_waters_mol_flag sigma_cut_off show_blobs_dialog* [function]

Where:

- *imol_for_map* is an integer number
- *imol_for_protein* is an integer number
- *new_waters_mol_flag* is an integer number
- *sigma_cut_off* is a number
- *show_blobs_dialog* is an integer number

10.65.6 get-text-for-find-waters-sigma-cut-off

`get-text-for-find-waters-sigma-cut-off` [function]

10.65.7 set-value-for-find-waters-sigma-cut-off

`set-value-for-find-waters-sigma-cut-off` *f* [function]

Where *f* is a number

10.65.8 on-big-blob-button-clicked

`on-big-blob-button-clicked` *button user_data* [function]

Where:

- *button* is a GtkButton
- *user_data* is an integer number

10.65.9 set-ligand-water-spherical-variance-limit

`set-ligand-water-spherical-variance-limit` *f* [function]

Where *f* is a number

10.65.10 set-ligand-water-to-protein-distance-limits

`set-ligand-water-to-protein-distance-limits` *f1 f2* [function]

Where:

- *f1* is a number
- *f2* is a number

set ligand to protein distance limits

f1 is the minimum distance, f2 is the maximum distance

10.65.11 set-ligand-water-n-cycles

`set-ligand-water-n-cycles` *i* [function]

Where *i* is an integer number

set the number of cycles of water searching

10.65.12 set-write-peaksearched-waters

`set-write-peaksearched-waters` [function]

10.65.13 execute-find-blobs

`execute-find-blobs` *imol_model imol_for_map cut_off interactive_flag* [function]

Where:

- *imol_model* is an integer number
- *imol_for_map* is an integer number
- *cut_off* is a number
- *interactive_flag* is an integer number

find blobs

10.65.14 execute-find-blobs-from-widget

`execute-find-blobs-from-widget` *dialog* [function]

Where *dialog* is a GtkWidget

10.65.15 wrapped-create-unmodelled-blobs-dialog

`wrapped-create-unmodelled-blobs-dialog` [function]

10.66 Bond Representation

10.66.1 set-default-bond-thickness

`set-default-bond-thickness` *t* [function]

Where *t* is an integer number

set the default thickness for bonds (e.g. in ~/.coot)

10.66.2 set-bond-thickness

`set-bond-thickness` *imol t* [function]

Where:

- *imol* is an integer number
- *t* is a number

set the thickness of the bonds in molecule number imol to t pixels

10.66.3 set-bond-thickness-intermediate-atoms

`set-bond-thickness-intermediate-atoms` *t* [function]

Where *t* is a number

set the thickness of the bonds of the intermediate atoms to t pixels

10.66.4 set-unbonded-atom-star-size

`set-unbonded-atom-star-size` *f* [function]

Where *f* is a number

10.66.5 get-default-bond-thickness

`get-default-bond-thickness` [function]

10.66.6 set-draw-zero-occ-markers

`set-draw-zero-occ-markers` *status* [function]

Where *status* is an integer number

set status of drawing zero occupancy markers.

default status is 1.

10.66.7 set-draw-hydrogens

`set-draw-hydrogens` *imol istat* [function]

Where:

- *imol* is an integer number
- *istat* is an integer number

set the hydrogen drawing state. istat = 0 is hydrogens off, istat = 1: show hydrogens

10.66.8 draw-hydrogens-state

`draw-hydrogens-state` *imol* [function]

Where *imol* is an integer number

the state of draw hydrogens for molecule number imol.

return -1 on bad imol.

10.66.9 graphics-to-ca-representation

`graphics-to-ca-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol as CAs

10.66.10 graphics-to-ca-plus-ligands-representation

`graphics-to-ca-plus-ligands-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol as CA + ligands

10.66.11 graphics-to-bonds-no-waters-representation

`graphics-to-bonds-no-waters-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol with no waters

10.66.12 graphics-to-bonds-representation

`graphics-to-bonds-representation` *mol* [function]

Where *mol* is an integer number

draw molecule number imol with normal bonds

10.66.13 graphics-to-ca-plus-ligands-sec-struct-representation

`graphics-to-ca-plus-ligands-sec-struct-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol with CA bonds in secondary structure representation and ligands

10.66.14 graphics-to-sec-struct-bonds-representation

`graphics-to-sec-struct-bonds-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol with bonds in secondary structure representation

10.66.15 graphics-to-rainbow-representation

`graphics-to-rainbow-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol in Jones' Rainbow

10.66.16 graphics-to-b-factor-representation

`graphics-to-b-factor-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol coloured by B-factor

10.66.17 graphics-to-b-factor-cas-representation

`graphics-to-b-factor-cas-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol coloured by B-factor, CA + ligands

10.66.18 graphics-to-occupancy-representation

`graphics-to-occupancy-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number imol coloured by occupancy

10.66.19 graphics-molecule-bond-type

`graphics-molecule-bond-type` *imol* [function]

Where *imol* is an integer number

what is the bond drawing state of molecule number imol

10.66.20 set-b-factor-bonds-scale-factor

`set-b-factor-bonds-scale-factor` *imol f* [function]

Where:

- *imol* is an integer number
- *f* is a number

scale the colours for colour by b factor representation

10.66.21 wrapped-create-bond-parameters-dialog

`wrapped-create-bond-parameters-dialog` [function]

10.66.22 apply-bond-parameters

`apply-bond-parameters` *w* [function]

Where *w* is a GtkWidget

10.66.23 make-ball-and-stick

`make-ball-and-stick` *imol atom_selection_str bond_thickness sphere_size do_spheres_flag* [function]

Where:

- *imol* is an integer number
- *atom_selection_str* is a string

- *bond_thickness* is a number
- *sphere_size* is a number
- *do_spheres_flag* is an integer number

make a ball and stick representation of imol given atom selection

e.g. (make-ball-and-stick 0 "/1" 0.15 0.25 1)

10.66.24 clear-ball-and-stick

`clear-ball-and-stick` *imol* [function]

Where *imol* is an integer number

clear ball and stick representation of molecule number imol

10.66.25 set-show-additional-representation

`set-show-additional-representation` *imol representation_number on_off_flag* [function]

Where:

- *imol* is an integer number
- *representation_number* is an integer number
- *on_off_flag* is an integer number

10.66.26 delete-additional-representation

`delete-additional-representation` *imol representation_number* [function]

Where:

- *imol* is an integer number
- *representation_number* is an integer number

10.66.27 additional-representation-by-string

`additional-representation-by-string` *imol atom_selection representation_type bonds_box_type bond_width draw_hydrogens_flag* [function]

Where:

- *imol* is an integer number
- *atom_selection* is a string
- *representation_type* is an integer number
- *bonds_box_type* is an integer number
- *bond_width* is a number
- *draw_hydrogens_flag* is an integer number

10.66.28 additional-representation-by-attributes

`additional-representation-by-attributes` *imol chain_id resno_start resno_end ins_code representation_type bonds_box_type bond_width draw_hydrogens_flag* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno_start* is an integer number
- *resno_end* is an integer number
- *ins_code* is a string
- *representation_type* is an integer number
- *bonds_box_type* is an integer number
- *bond_width* is a number
- *draw_hydrogens_flag* is an integer number

10.66.29 wrapped-create-add-additional-representation-gui

`wrapped-create-add-additional-representation-gui` [function]

10.66.30 add-additional-representation-by-widget

`add-additional-representation-by-widget` *w* [function]

Where *w* is a GtkWidget

10.66.31 add-reps-molecule-option-menu-item-select

`add-reps-molecule-option-menu-item-select` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.67 Dots Representation

10.67.1 dots

`dots` *imol atom_selection_str dot_density sphere_size_scale* [function]

Where:

- *imol* is an integer number
- *atom_selection_str* is a string
- *dot_density* is a number
- *sphere_size_scale* is a number

display dotted surface

return a generic objects handle (which can be used to remove later)

10.67.2 clear-dots

`clear-dots` *imol dots_handle* [function]

Where:

- *imol* is an integer number
- *dots_handle* is an integer number

clear dots in imol with dots_handle

10.67.3 n-dots-sets

`n-dots-sets` *imol* [function]

Where *imol* is an integer number

return the number of dots sets for molecule number imol

10.68 Pep-flip Interface

10.68.1 do-pepflip

`do-pepflip` *state* [function]

Where *state* is an integer number

10.68.2 pepflip

`pepflip` *imol chain_id resno inscode* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *inscode* is a string

pepflip the given residue

10.69 Rigid Body Refinement Interface

10.69.1 do-rigid-body-refine

`do-rigid-body-refine` *state* [function]

Where *state* is an integer number

10.69.2 rigid-body-refine-zone

`rigid-body-refine-zone` *reso_start resno_end chain_id imol* [function]

Where:

- *reso_start* is an integer number
- *resno_end* is an integer number
- *chain_id* is a string
- *imol* is an integer number

setup rigid body refine zone

where we set the atom selection holders according to the arguments and then call execute_rigid_body_refine()

10.69.3 rigid-body-refine-by-atom-selection

`rigid-body-refine-by-atom-selection` *imol atom_selection_string* [function]

Where:

- *imol* is an integer number
- *atom_selection_string* is a string

10.69.4 execute-rigid-body-refine

`execute-rigid-body-refine` *auto_range_flag* [function]

Where *auto_range_flag* is an integer number

10.69.5 set-rigid-body-fit-acceptable-fit-fraction

`set-rigid-body-fit-acceptable-fit-fraction` *f* [function]

Where *f* is a number

set rigid body fraction of atoms in positive density

10.70 Dynamic Map

10.70.1 toggle-dynamic-map-display-size

`toggle-dynamic-map-display-size` [function]

10.70.2 toggle-dynamic-map-sampling

`toggle-dynamic-map-sampling` [function]

10.70.3 set-map-dynamic-map-sampling-checkbutton

`set-map-dynamic-map-sampling-checkbutton` *checkbutton* [function]

Where *checkbutton* is a GtkWidget

10.70.4 set-map-dynamic-map-display-size-checkbutton

`set-map-dynamic-map-display-size-checkbutton` *checkbutton* [function]

Where *checkbutton* is a GtkWidget

10.70.5 set-dynamic-map-size-display-on

`set-dynamic-map-size-display-on` [function]

10.70.6 set-dynamic-map-size-display-off

`set-dynamic-map-size-display-off` [function]

10.70.7 get-dynamic-map-size-display

`get-dynamic-map-size-display` [function]

10.70.8 set-dynamic-map-sampling-on

`set-dynamic-map-sampling-on` [function]

10.70.9 set-dynamic-map-sampling-off

`set-dynamic-map-sampling-off` [function]

10.70.10 get-dynamic-map-sampling

`get-dynamic-map-sampling` [function]

10.70.11 set-dynamic-map-zoom-offset

`set-dynamic-map-zoom-offset` *i* [function]
Where *i* is an integer number

10.71 Add Terminal Residue Functions

10.71.1 do-add-terminal-residue

`do-add-terminal-residue` *state* [function]
Where *state* is an integer number

10.71.2 set-add-terminal-residue-n-phi-psi-trials

`set-add-terminal-residue-n-phi-psi-trials` *n* [function]
Where *n* is an integer number

10.71.3 set-add-terminal-residue-add-other-residue-flag

`set-add-terminal-residue-add-other-residue-flag` *i* [function]
Where *i* is an integer number

10.71.4 set-terminal-residue-do-rigid-body-refine

`set-terminal-residue-do-rigid-body-refine` *v* [function]
Where *v* is an integer number

10.71.5 add-terminal-residue-immediate-addition-state

`add-terminal-residue-immediate-addition-state` [function]

10.71.6 set-add-terminal-residue-immediate-addition

`set-add-terminal-residue-immediate-addition` *i* [function]
Where *i* is an integer number

set immediate addition of terminal residue

call with i=1 for immediate addtion

10.71.7 add-terminal-residue

`add-terminal-residue` *imol chain_id residue_number residue_type immediate_add* [function]
Where:

- *imol* is an integer number
- *chain_id* is a string
- *residue_number* is an integer number
- *residue_type* is a string
- *immediate_add* is an integer number

Add a terminal residue.

return 0 on failure, 1 on success

10.71.8 set-add-terminal-residue-default-residue-type

`set-add-terminal-residue-default-residue-type` *type* [function]

Where *type* is a string

set the residue type of an added terminal residue.

10.71.9 set-add-terminal-residue-do-post-refine

`set-add-terminal-residue-do-post-refine` *istat* [function]

Where *istat* is an integer number

set a flag to run refine zone on terminal residues after an addition.

10.71.10 add-terminal-residue-do-post-refine-state

`add-terminal-residue-do-post-refine-state` [function]

what is the value of the previous flag?

10.72 Delete Residues

10.72.1 delete-atom-by-atom-index

`delete-atom-by-atom-index` *imol index do_delete_dialog* [function]

Where:

- *imol* is an integer number
- *index* is an integer number
- *do_delete_dialog* is an integer number

10.72.2 delete-residue-by-atom-index

`delete-residue-by-atom-index` *imol index do_delete_dialog* [function]

Where:

- *imol* is an integer number
- *index* is an integer number
- *do_delete_dialog* is an integer number

10.72.3 delete-residue-hydrogens-by-atom-index

`delete-residue-hydrogens-by-atom-index` *imol index do_delete_dialog* [function]

Where:

- *imol* is an integer number
- *index* is an integer number
- *do_delete_dialog* is an integer number

10.72.4 delete-residue-range

`delete-residue-range` *imol chain_id resno_start end_resno* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno_start* is an integer number
- *end_resno* is an integer number

delete residue range

10.72.5 delete-residue

`delete-residue` *imol chain_id resno inscode* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *inscode* is a string

delete residue

10.72.6 delete-residue-with-altconf

`delete-residue-with-altconf` *imol chain_id resno inscode altloc* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *inscode* is a string
- *altloc* is a string

delete residue with altconf

10.72.7 delete-residue-hydrogens

`delete-residue-hydrogens` *imol chain_id resno inscode altloc* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *inscode* is a string
- *altloc* is a string

delete hydrogen atoms in residue

10.72.8 delete-atom

`delete-atom` *imol chain_id resno ins_code at_name altloc* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string
- *at_name* is a string
- *altloc* is a string

delete atom in residue

10.72.9 delete-residue-sidechain

`delete-residue-sidechain` *imol chain_id resno ins_code do_delete_dialog* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string
- *do_delete_dialog* is an integer number

delete all atoms in residue that are not main chain or CB

10.72.10 set-delete-atom-mode

`set-delete-atom-mode` [function]

10.72.11 set-delete-residue-mode

`set-delete-residue-mode` [function]

10.72.12 set-delete-residue-zone-mode

`set-delete-residue-zone-mode` [function]

10.72.13 set-delete-residue-hydrogens-mode

`set-delete-residue-hydrogens-mode` [function]

10.72.14 set-delete-water-mode

`set-delete-water-mode` [function]

10.72.15 set-delete-sidechain-mode

`set-delete-sidechain-mode` [function]

10.72.16 delete-item-mode-is-atom-p

`delete-item-mode-is-atom-p` [function]

10.72.17 delete-item-mode-is-residue-p

`delete-item-mode-is-residue-p` [function]

10.72.18 delete-item-mode-is-water-p

`delete-item-mode-is-water-p` [function]

10.72.19 delete-item-mode-is-sidechain-p

`delete-item-mode-is-sidechain-p` [function]

10.72.20 store-delete-item-widget

`store-delete-item-widget` *widget* [function]

Where *widget* is a GtkWidget

10.72.21 clear-pending-delete-item

`clear-pending-delete-item` [function]

10.72.22 clear-delete-item-widget

`clear-delete-item-widget` [function]

10.72.23 store-delete-item-widget-position

`store-delete-item-widget-position` [function]

10.72.24 delete-item-widget-is-being-shown

`delete-item-widget-is-being-shown` [function]

10.72.25 delete-item-widget-keep-active-on

`delete-item-widget-keep-active-on` [function]

10.72.26 wrapped-create-delete-item-dialog

`wrapped-create-delete-item-dialog` [function]

10.72.27 delete-object-handle-delete-dialog

`delete-object-handle-delete-dialog` *do_delete_dialog* [function]
Where *do_delete_dialog* is an integer number

10.73 Mainchain Building Functions

10.73.1 do-db-main

`do-db-main` *state* [function]
Where *state* is an integer number

10.73.2 db-mainchain

`db-mainchain` *imol chain_id iresno_start iresno_end direction_string* [function]
Where:

- *imol* is an integer number
- *chain_id* is a string
- *iresno_start* is an integer number
- *iresno_end* is an integer number
- *direction_string* is a string

CA -> mainchain conversion.

10.74 Close Molecule FUnctions

10.74.1 close-molecule

`close-molecule` *imol* [function]
Where *imol* is an integer number

10.74.2 close-molecule-by-widget

`close-molecule-by-widget` *optionmenu* [function]
Where *optionmenu* is a GtkWidget

10.74.3 fill-close-option-menu-with-all-molecule-options

`fill-close-option-menu-with-all-molecule-options` *optionmenu* [function]
Where *optionmenu* is a GtkWidget

10.74.4 close-molecule-item-select

`close-molecule-item-select` *item pos* [function]
Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.74.5 new-close-molecules

`new-close-molecules` *window* [function]
Where *window* is a GtkWidget

10.74.6 wrapped-create-new-close-molecules-dialog

`wrapped-create-new-close-molecules-dialog` [function]

10.75 Rotatmer Functions

10.75.1 setup-rotamers

`setup-rotamers` *state* [function]
Where *state* is an integer number

10.75.2 do-rotamers

`do-rotamers` *atom_index imol* [function]
Where:

- *atom_index* is an integer number
- *imol* is an integer number

10.75.3 show-rotamers-dialog

`show-rotamers-dialog` *imol chain_id resno ins_code altconf* [function]
Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string
- *altconf* is a string

10.75.4 set-rotamer-lowest-probability

`set-rotamer-lowest-probability` *f* [function]
Where *f* is a number

10.75.5 set-rotamer-check-clashes

`set-rotamer-check-clashes` *i* [function]
Where *i* is an integer number

set a flag: 0 is off, 1 is on

10.75.6 auto-fit-best-rotamer

`auto-fit-best-rotamer` *resno altloc insertion_code chain_id imol_coords imol_map clash_flag lowest_probability* [function]

Where:

- *resno* is an integer number
- *altloc* is a string
- *insertion_code* is a string
- *chain_id* is a string
- *imol_coords* is an integer number
- *imol_map* is an integer number
- *clash_flag* is an integer number
- *lowest_probability* is a number

auto fit by rotamer search.

return the score, for some not very good reason. clash_flag determines if we use clashes with other residues in the score for this rotamer (or not). It would be cool to call this from a script that went residue by residue along a (newly-built) chain (now available).

10.75.7 set-auto-fit-best-rotamer-clash-flag

`set-auto-fit-best-rotamer-clash-flag` *i* [function]

Where *i* is an integer number

set the clash flag for rotamer search

And this functions for [pre-setting] the variables for auto_fit_best_rotamer called interactively (using a graphics_info_t function). 0 off, 1 on.

10.75.8 rotamer-score

`rotamer-score` *imol chain_id res_no insertion_code* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *res_no* is an integer number
- *insertion_code* is a string

10.75.9 setup-auto-fit-rotamer

`setup-auto-fit-rotamer` *state* [function]

Where *state* is an integer number

10.75.10 n-rotamers

`n-rotamers` *imol chain_id resno ins_code* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string

return the number of rotamers for this residue - return -1 on no residue found.

10.75.11 set-residue-to-rotamer-number

`set-residue-to-rotamer-number` *imol chain_id resno ins_code rotamer_number* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string
- *rotamer_number* is an integer number

set the residue specified to the rotamer number specifed.

10.75.12 fill-partial-residues

`fill-partial-residues` *imol* [function]

Where *imol* is an integer number

fill all the residues of molecule number imol that have missing atoms.

To be used to remove the effects of chainsaw.

10.75.13 fill-partial-residue

`fill-partial-residue` *imol chain_id resno inscode* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *inscode* is a string

10.75.14 set-graphics-rotamer-dialog

`set-graphics-rotamer-dialog` *w* [function]

Where *w* is a GtkWidget

10.76 180 Flip Side chain

10.76.1 do-180-degree-side-chain-flip

`do-180-degree-side-chain-flip` *imol chain_id resno inscode altconf* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *inscode* is a string
- *altconf* is a string

rotate 180 degrees round the last chi angle

10.76.2 setup-180-degree-flip

`setup-180-degree-flip` *state* [function]

Where *state* is an integer number

10.77 Mutate Functions

10.77.1 setup-mutate

`setup-mutate` *state* [function]

Where *state* is an integer number

10.77.2 setup-mutate-auto-fit

`setup-mutate-auto-fit` *state* [function]

Where *state* is an integer number

Mutate then fit to map.

that we have a map define is checked first

10.77.3 do-mutation

`do-mutation` *type is_stub_flag* [function]

Where:

- *type* is a string
- *is_stub_flag* is an integer number

10.77.4 progressive-residues-in-chain-check

`progressive-residues-in-chain-check` *chain_id imol* [function]

Where:

- *chain_id* is a string
- *imol* is an integer number

10.77.5 mutate

`mutate` *imol chain_id ires inscode target_res_type* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *ires* is an integer number
- *inscode* is a string
- *target_res_type* is a string

mutate a given residue

target_res_type is a three-letter-code.

Return 1 on a good mutate.

10.77.6 mutate-base

`mutate-base` *imol chain_id res_no ins_code res_type* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *res_no* is an integer number
- *ins_code* is a string
- *res_type* is a string

10.77.7 set-mutate-auto-fit-do-post-refine

`set-mutate-auto-fit-do-post-refine` *istate* [function]

Where *istate* is an integer number

Do you want Coot to automatically run a refinement after every mutate and autofit?

1 for yes, 0 for no.

10.77.8 mutate-auto-fit-do-post-refine-state

`mutate-auto-fit-do-post-refine-state` [function]

what is the value of the previous flag?

10.77.9 set-rotamer-auto-fit-do-post-refine

`set-rotamer-auto-fit-do-post-refine` *istate* [function]

Where *istate* is an integer number

Do you want Coot to automatically run a refinement after every rotamer autofit?

1 for yes, 0 for no.

10.77.10 rotamer-auto-fit-do-post-refine-state

`rotamer-auto-fit-do-post-refine-state` [function]

what is the value of the previous flag?

10.77.11 mutate-single-residue-by-serial-number

`mutate-single-residue-by-serial-number` *ires_ser chain_id imol target_res_type* [function]

Where:

- *ires_ser* is an integer number
- *chain_id* is a string
- *imol* is an integer number
- *target_res_type* is a character

an alternate interface to mutation of a singe residue.

ires-ser is the serial number of the residue, not the seqnum There 2 functions don't make backups, but

does - CHECKME Hence

is for use as a "one-by-one" type and the following 2 by wrappers that muate either a residue range or a whole chain

Note that the target_res_type is a char, not a string (or a char *). So from the scheme interface you'd use (for example) hash backslash A for ALA.

Returns: 1 on success, 0 on failure

10.77.12 mutate-single-residue-by-seqno

`mutate-single-residue-by-seqno` *ires inscode chain_id imol target_res_type* [function]

Where:

- *ires* is an integer number
- *inscode* is a string
- *chain_id* is a string
- *imol* is an integer number
- *target_res_type* is a character

10.77.13 do-base-mutation

`do-base-mutation` *type* [function]

Where *type* is a string

10.77.14 set-residue-type-chooser-stub-state

`set-residue-type-chooser-stub-state` *istat* [function]

Where *istat* is an integer number

set a flag saying that the residue chosen by mutate or auto-fit mutate should only be added as a stub (mainchain + CB)

10.78 Alternative Conformation

10.78.1 alt-conf-split-type-number

`alt-conf-split-type-number` [function]

10.78.2 set-add-alt-conf-split-type-number

`set-add-alt-conf-split-type-number` *i* [function]
Where *i* is an integer number

10.78.3 setup-alt-conf-with-dialog

`setup-alt-conf-with-dialog` *dialog* [function]
Where *dialog* is a GtkWidget

10.78.4 unset-add-alt-conf-dialog

`unset-add-alt-conf-dialog` [function]

10.78.5 unset-add-alt-conf-define

`unset-add-alt-conf-define` [function]

10.78.6 altconf

`altconf` [function]

10.78.7 set-add-alt-conf-new-atoms-occupancy

`set-add-alt-conf-new-atoms-occupancy` *f* [function]
Where *f* is a number

10.78.8 set-show-alt-conf-intermediate-atoms

`set-show-alt-conf-intermediate-atoms` *i* [function]
Where *i* is an integer number

10.78.9 show-alt-conf-intermediate-atoms-state

`show-alt-conf-intermediate-atoms-state` [function]

10.78.10 zero-occupancy-residue-range

`zero-occupancy-residue-range` *imol chain_id ires1 ires2* [function]
Where:

- *imol* is an integer number
- *chain_id* is a string
- *ires1* is an integer number
- *ires2* is an integer number

10.78.11 fill-occupancy-residue-range

`fill-occupancy-residue-range` *imol chain_id ires1 ires2* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *ires1* is an integer number
- *ires2* is an integer number

10.78.12 set-b-factor-residue-range

`set-b-factor-residue-range` *imol chain_id ires1 ires2 bval* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *ires1* is an integer number
- *ires2* is an integer number
- *bval* is a number

10.78.13 reset-b-factor-residue-range

`reset-b-factor-residue-range` *imol chain_id ires1 ires2* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *ires1* is an integer number
- *ires2* is an integer number

10.79 Pointer Atom Functions

10.79.1 place-atom-at-pointer

`place-atom-at-pointer` [function]

10.79.2 place-atom-at-pointer-by-window

`place-atom-at-pointer-by-window` [function]

10.79.3 place-typed-atom-at-pointer

`place-typed-atom-at-pointer` *type* [function]

Where *type* is a string

10.79.4 set-pointer-atom-is-dummy

`set-pointer-atom-is-dummy` *i* [function]

Where *i* is an integer number

10.79.5 fill-place-atom-molecule-option-menu

`fill-place-atom-molecule-option-menu` *optionmenu* [function]
Where *optionmenu* is a GtkWidget

10.79.6 display-where-is-pointer

`display-where-is-pointer` [function]

10.79.7 pointer-atom-molecule

`pointer-atom-molecule` [function]

10.79.8 set-pointer-atom-molecule

`set-pointer-atom-molecule` *imol* [function]
Where *imol* is an integer number

10.80 Terminal OXT Atom

10.80.1 add-OXT-to-residue

`add-OXT-to-residue` *imol reso insertion_code chain_id* [function]
Where:

- *imol* is an integer number
- *reso* is an integer number
- *insertion_code* is a string
- *chain_id* is a string

10.80.2 wrapped-create-add-OXT-dialog

`wrapped-create-add-OXT-dialog` [function]

10.80.3 apply-add-OXT-from-widget

`apply-add-OXT-from-widget` *w* [function]
Where *w* is a GtkWidget

10.81 Crosshairs Interface

10.81.1 set-draw-crosshairs

`set-draw-crosshairs` *i* [function]
Where *i* is an integer number

10.81.2 draw-crosshairs-state

`draw-crosshairs-state` [function]

10.82 Edit Chi Angles

10.82.1 setup-edit-chi-angles

`setup-edit-chi-angles` *state* [function]
Where *state* is an integer number

10.82.2 rotate-chi

`rotate-chi` *am* [function]
Where *am* is a number

10.82.3 set-find-hydrogen-torsions

`set-find-hydrogen-torsions` *state* [function]
Where *state* is an integer number

show torsions that rotate hydrogens in the torsion angle manipulation dialog. Note that this may be needed if, in the dictionary cif file torsion which have as a 4th atom both a hydrogen and a heavier atom bonding to the 3rd atom, but list the 4th atom as a hydrogen (not a heavier atom).

10.82.4 set-graphics-edit-current-chi

`set-graphics-edit-current-chi` *ichi* [function]
Where *ichi* is an integer number

10.82.5 unset-moving-atom-move-chis

`unset-moving-atom-move-chis` [function]

10.82.6 edit-chi-angles

`edit-chi-angles` *imol chain_id resno ins_code altconf* [function]
Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string
- *altconf* is a string

display the edit chi angles gui for the given residue

return a status of 0 if it failed to fined the residue, return a value of 1 if it worked.

10.82.7 set-show-chi-angle-bond

`set-show-chi-angle-bond` *imode* [function]
Where *imode* is an integer number

10.82.8 fill-chi-angles-vbox

`fill-chi-angles-vbox` *vbox* [function]

Where *vbox* is a GtkWidget

10.82.9 setup-torsion-general

`setup-torsion-general` *state* [function]

Where *state* is an integer number

beloved torsion general at last makes an entrance onto the Coot scene...

10.82.10 toggle-torsion-general-reverse

`toggle-torsion-general-reverse` [function]

10.83 Masks

10.83.1 mask-map-by-molecule

`mask-map-by-molecule` *map_mol_no coord_mol_no invert_flag* [function]

Where:

- *map_mol_no* is an integer number
- *coord_mol_no* is an integer number
- *invert_flag* is an integer number

generate a new map that has been masked by some coordinates

(mask-map-by-molecule map-no mol-no invert?) creates and displays a masked map, cuts down density where the coordinates are (invert is 0). If invert? is 1, cut the density down where there are no atoms atoms.

10.83.2 mask-map-by-atom-selection

`mask-map-by-atom-selection` *map_mol_no coords_mol_no mmdb_atom_selection invert_flag* [function]

Where:

- *map_mol_no* is an integer number
- *coords_mol_no* is an integer number
- *mmdb_atom_selection* is a string
- *invert_flag* is an integer number

10.83.3 set-map-mask-atom-radius

`set-map-mask-atom-radius` *rad* [function]

Where *rad* is a number

set the atom radius for map masking

10.83.4 map-mask-atom-radius

`map-mask-atom-radius` [function]

get the atom radius for map masking

10.84 check Waters Interface

10.84.1 wrapped-create-check-waters-dialog

`wrapped-create-check-waters-dialog` [function]

10.84.2 set-check-waters-b-factor-limit

`set-check-waters-b-factor-limit` *f* [function]

Where *f* is a number

10.84.3 set-check-waters-map-sigma-limit

`set-check-waters-map-sigma-limit` *f* [function]

Where *f* is a number

10.84.4 set-check-waters-min-dist-limit

`set-check-waters-min-dist-limit` *f* [function]

Where *f* is a number

10.84.5 set-check-waters-max-dist-limit

`set-check-waters-max-dist-limit` *f* [function]

Where *f* is a number

10.84.6 check-waters-molecule-menu-item-activate

`check-waters-molecule-menu-item-activate` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.84.7 do-check-waters-by-widget

`do-check-waters-by-widget` *dialog* [function]

Where *dialog* is a GtkWidget

10.84.8 store-checked-waters-baddies-dialog

`store-checked-waters-baddies-dialog` *dialog* [function]

Where *dialog* is a GtkWidget

10.84.9 wrapped-checked-waters-baddies-dialog

`wrapped-checked-waters-baddies-dialog` *imol b_factor_lim map_sigma_lim min_dist max_dist part_occ_contact_flag zero_occ_flag logical_operator_and_or_flag* [function]

Where:

- *imol* is an integer number

- *b_factor_lim* is a number
- *map_sigma_lim* is a number
- *min_dist* is a number
- *max_dist* is a number
- *part_occ_contact_flag* is an integer number
- *zero_occ_flag* is an integer number
- *logical_operator_and_or_flag* is an integer number

10.84.10 delete-checked-waters-baddies

`delete-checked-waters-baddies` *imol b_factor_lim map_sigma_lim min_dist max_dist part_occ_contact_flag zero_occ_flag logical_operator_and_or_flag* [function]

Where:

- *imol* is an integer number
- *b_factor_lim* is a number
- *map_sigma_lim* is a number
- *min_dist* is a number
- *max_dist* is a number
- *part_occ_contact_flag* is an integer number
- *zero_occ_flag* is an integer number
- *logical_operator_and_or_flag* is an integer number

Delete waters that are fail to meet the given criteria.

10.84.11 check-waters-by-difference-map

`check-waters-by-difference-map` *imol_waters imol_diff_map interactive_flag* [function]

Where:

- *imol_waters* is an integer number
- *imol_diff_map* is an integer number
- *interactive_flag* is an integer number

10.84.12 check-waters-by-difference-map-by-widget

`check-waters-by-difference-map-by-widget` *window* [function]

Where *window* is a GtkWidget

10.84.13 wrapped-create-check-waters-diff-map-dialog

`wrapped-create-check-waters-diff-map-dialog` [function]

10.84.14 check-waters-by-difference-map-sigma-level-state

`check-waters-by-difference-map-sigma-level-state` [function]

10.84.15 set-check-waters-by-difference-map-sigma-level

`set-check-waters-by-difference-map-sigma-level` *f* [function]
Where *f* is a number

10.85 Least-Squares matching

10.85.1 clear-lsq-matches

`clear-lsq-matches` [function]

10.85.2 add-lsq-match

`add-lsq-match` *reference_resno_start reference_resno_end chain_id_reference moving_resno_start moving_resno_end chain_id_moving match_type* [function]
Where:

- *reference_resno_start* is an integer number
- *reference_resno_end* is an integer number
- *chain_id_reference* is a string
- *moving_resno_start* is an integer number
- *moving_resno_end* is an integer number
- *chain_id_moving* is a string
- *match_type* is an integer number

10.85.3 apply-lsq-matches-simple

`apply-lsq-matches-simple` *imol_reference imol_moving* [function]
Where:

- *imol_reference* is an integer number
- *imol_moving* is an integer number

10.85.4 setup-lsq-deviation

`setup-lsq-deviation` *state* [function]
Where *state* is an integer number

10.85.5 setup-lsq-plane-define

`setup-lsq-plane-define` *state* [function]
Where *state* is an integer number

10.85.6 wrapped-create-lsq-plane-dialog

`wrapped-create-lsq-plane-dialog` [function]

10.85.7 unset-lsq-plane-dialog

`unset-lsq-plane-dialog` [function]

10.85.8 remove-last-lsq-plane-atom

`remove-last-lsq-plane-atom` [function]

10.85.9 wrapped-create-least-squares-dialog

`wrapped-create-least-squares-dialog` [function]

10.85.10 apply-lsq-matches-by-widget

`apply-lsq-matches-by-widget` *lsq_dialog* [function]

Where *lsq_dialog* is a GtkWidget

10.85.11 lsq-ref-mol-option-menu-changed

`lsq-ref-mol-option-menu-changed` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.85.12 lsq-mov-mol-option-menu-changed

`lsq-mov-mol-option-menu-changed` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.85.13 lsq-reference-chain-option-menu-item-activate

`lsq-reference-chain-option-menu-item-activate` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.85.14 lsq-moving-chain-option-menu-item-activate

`lsq-moving-chain-option-menu-item-activate` *item pos* [function]

Where:

- *item* is a GtkWidget
- *pos* is a GtkPositionType

10.85.15 fill-lsq-option-menu-with-chain-options

`fill-lsq-option-menu-with-chain-options` *chain_optionmenu* [function]
is_reference_structure_flag

Where:

- *chain_optionmenu* is a GtkWidget
- *is_reference_structure_flag* is an integer number

10.86 Trim

10.86.1 trim-molecule-by-map

`trim-molecule-by-map` *imol_coords imol_map map_level delete_or_zero_occ_flag* [function]

Where:

- *imol_coords* is an integer number
- *imol_map* is an integer number
- *map_level* is a number
- *delete_or_zero_occ_flag* is an integer number

10.87 External Ray-Tracing

10.87.1 raster3d

`raster3d` *rd3_filename* [function]

Where *rd3_filename* is a string

create a r3d file for the current view

10.87.2 povray

`povray` *filename* [function]

Where *filename* is a string

10.87.3 renderman

`renderman` *rib_filename* [function]

Where *rib_filename* is a string

10.87.4 make-image-raster3d

`make-image-raster3d` *filename* [function]

Where *filename* is a string

10.87.5 make-image-povray

`make-image-povray` *filename* [function]

Where *filename* is a string

10.87.6 set-raster3d-bond-thickness

`set-raster3d-bond-thickness` *f* [function]

Where *f* is a number

set the bond thickness for the Raster3D representation

10.87.7 set-raster3d-atom-radius

`set-raster3d-atom-radius` *f* [function]

Where *f* is a number

10.87.8 set-raster3d-density-thickness

`set-raster3d-density-thickness` *f* [function]
Where *f* is a number

set the density line thickness for the Raster3D representation

10.87.9 set-renderer-show-atoms

`set-renderer-show-atoms` *istate* [function]
Where *istate* is an integer number

set the flag to show atoms for the Raster3D representation

10.87.10 set-raster3d-bone-thickness

`set-raster3d-bone-thickness` *f* [function]
Where *f* is a number

set the bone (skeleton) thickness for the Raster3D representation

10.87.11 set-raster3d-shadows-enabled

`set-raster3d-shadows-enabled` *state* [function]
Where *state* is an integer number

turn off shadows for raster3d output - give argument 0 to turn off

10.87.12 raster-screen-shot

`raster-screen-shot` [function]
run raster3d and display the resulting image.

10.88 Superposition (SSM)

10.88.1 superpose

`superpose` *imol1 imol2 move_imol2_flag* [function]
Where:

- *imol1* is an integer number
- *imol2* is an integer number
- *move_imol2_flag* is an integer number

simple interface to superposition.

Superpose all residues of imol2 onto imol1. imol1 is reference, we can either move imol2 or copy it to generate a new molecule depending on the vaule of move_imol2_flag (1 for move 0 for copy).

10.88.2 superpose-with-chain-selection

`superpose-with-chain-selection` *imol1 imol2 chain_imol1 chain_imol2 chain_used_flag_imol1 chain_used_flag_imol2 move_imol2_copy_flag* [function]

Where:

- *imol1* is an integer number
- *imol2* is an integer number
- *chain_imol1* is a string
- *chain_imol2* is a string
- *chain_used_flag_imol1* is an integer number
- *chain_used_flag_imol2* is an integer number
- *move_imol2_copy_flag* is an integer number

chain-based interface to superposition.

Superpose the given chains of imol2 onto imol1. imol1 is reference, we can either move imol2 or copy it to generate a new molecule depending on the vaule of move_imol2_flag (1 for move 0 for copy).

10.88.3 superpose-with-atom-selection

`superpose-with-atom-selection` *imol1 imol2 mmdb_atom_sel_str_1 mmdb_atom_sel_str_2 move_imol2_copy_flag* [function]

Where:

- *imol1* is an integer number
- *imol2* is an integer number
- *mmdb_atom_sel_str_1* is a string
- *mmdb_atom_sel_str_2* is a string
- *move_imol2_copy_flag* is an integer number

detailed interface to superposition.

Superpose the given atom selection (specified by the mmdb atom selection strings) of imol2 onto imol1. imol1 is reference, we can either move imol2 or copy it to generate a new molecule depending on the vaule of move_imol2_flag (1 for move 0 for copy).

10.88.4 execute-superpose

`execute-superpose` *w* [function]

Where *w* is a GtkWidget

10.88.5 wrapped-create-superpose-dialog

`wrapped-create-superpose-dialog` [function]

10.88.6 fill-superpose-option-menu-with-chain-options

`fill-superpose-option-menu-with-chain-options` [function]
chain_optionmenu is_reference_structure_flag

Where:

- *chain_optionmenu* is a GtkWidget
- *is_reference_structure_flag* is an integer number

10.89 NCS

10.89.1 set-draw-ncs-ghosts

`set-draw-ncs-ghosts` *imol istate* [function]

Where:

- *imol* is an integer number
- *istate* is an integer number

set drawing state of NCS ghosts for molecule number imol

10.89.2 draw-ncs-ghosts-state

`draw-ncs-ghosts-state` *imol* [function]

Where *imol* is an integer number

return the drawing state of NCS ghosts for molecule number imol. Return -1 on imol is a bad molecule or no ghosts.

10.89.3 set-ncs-ghost-bond-thickness

`set-ncs-ghost-bond-thickness` *imol f* [function]

Where:

- *imol* is an integer number
- *f* is a number

set bond thickness of NCS ghosts for molecule number imol

10.89.4 ncs-update-ghosts

`ncs-update-ghosts` *imol* [function]

Where *imol* is an integer number

update ghosts for molecule number imol

10.89.5 make-dynamically-transformed-ncs-maps

`make-dynamically-transformed-ncs-maps` *imol_model imol_map* [function]
overwrite_maps_of_same_name_flag

Where:

- *imol_model* is an integer number
- *imol_map* is an integer number

- *overwrite_maps_of_same_name_flag* is an integer number

make NCS map

10.89.6 make-dynamically-transformed-ncs-maps-by-widget

`make-dynamically-transformed-ncs-maps-by-widget` *dialog* [function]
Where *dialog* is a GtkWidget

10.89.7 wrapped-create-ncs-maps-dialog

`wrapped-create-ncs-maps-dialog` [function]

10.89.8 make-ncs-ghosts-maybe

`make-ncs-ghosts-maybe` *imol* [function]
Where *imol* is an integer number

10.89.9 add-ncs-matrix

`add-ncs-matrix` *imol this_chain_id target_chain_id m11 m12 m13 m21 m22 m23 m31 m32 m33 t1 t2 t3* [function]
Where:

- *imol* is an integer number
- *this_chain_id* is a string
- *target_chain_id* is a string
- *m11* is a number
- *m12* is a number
- *m13* is a number
- *m21* is a number
- *m22* is a number
- *m23* is a number
- *m31* is a number
- *m32* is a number
- *m33* is a number
- *t1* is a number
- *t2* is a number
- *t3* is a number

Add NCS matrix.

10.89.10 clear-ncs-ghost-matrices

`clear-ncs-ghost-matrices` *imol* [function]
Where *imol* is an integer number

10.89.11 add-strict-ncs-matrix

`add-strict-ncs-matrix` *imol this_chain_id target_chain_id m11 m12 m13 m21 m22 m23 m31 m32 m33 t1 t2 t3* [function]

Where:

- *imol* is an integer number
- *this_chain_id* is a string
- *target_chain_id* is a string
- *m11* is a number
- *m12* is a number
- *m13* is a number
- *m21* is a number
- *m22* is a number
- *m23* is a number
- *m31* is a number
- *m32* is a number
- *m33* is a number
- *t1* is a number
- *t2* is a number
- *t3* is a number

add an NCS matrix for strict NCS molecule representation

for CNS strict NCS usage: expand like normal symmetry does

10.89.12 show-strict-ncs-state

`show-strict-ncs-state` *imol* [function]

Where *imol* is an integer number

return the state of NCS ghost molecules for molecule number imol

10.89.13 set-show-strict-ncs

`set-show-strict-ncs` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

set display state of NCS ghost molecules for molecule number imol

10.89.14 set-ncs-homology-level

`set-ncs-homology-level` *flev* [function]

Where *flev* is a number

At what level of homology should we say that we can't see homology for NCS calculation? (default 0.8).

10.89.15 copy-chain

`copy-chain` *imol from_chain to_chain* [function]

Where:

- *imol* is an integer number
- *from_chain* is a string
- *to_chain* is a string

Copy single NCS chain.

10.89.16 copy-from-ncs-master-to-others

`copy-from-ncs-master-to-others` *imol chain_id* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string

Copy chain from master to all related NCS chains.

10.89.17 copy-residue-range-from-ncs-master-to-others

`copy-residue-range-from-ncs-master-to-others` *imol master_chain_id residue_range_start residue_range_end* [function]

Where:

- *imol* is an integer number
- *master_chain_id* is a string
- *residue_range_start* is an integer number
- *residue_range_end* is an integer number

Copy residue range to all related NCS chains. If the target residues do not exist in the peer chains, then create them.

10.89.18 wrapped-create-ncs-control-dialog

`wrapped-create-ncs-control-dialog` [function]

10.89.19 ncs-control-change-ncs-master-to-chain

`ncs-control-change-ncs-master-to-chain` *imol ichain* [function]

Where:

- *imol* is an integer number
- *ichain* is an integer number

change the NCS master chain (by number)

10.89.20 ncs-control-change-ncs-master-to-chain-id

`ncs-control-change-ncs-master-to-chain-id` *imol chain_id* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string

change the NCS master chain (by chain_id)

10.89.21 ncs-control-change-ncs-master-to-chain-update-widget

`ncs-control-change-ncs-master-to-chain-update-widget` *w imol ichain* [function]

Where:

- *w* is a GtkWidget
- *imol* is an integer number
- *ichain* is an integer number

10.89.22 ncs-control-display-chain

`ncs-control-display-chain` *imol ichain state* [function]

Where:

- *imol* is an integer number
- *ichain* is an integer number
- *state* is an integer number

display the NCS master chain

10.89.23 set-ncs-matrix-type

`set-ncs-matrix-type` *flag* [function]

Where *flag* is an integer number

10.89.24 get-ncs-matrix-state

`get-ncs-matrix-state` [function]

10.90 Helices and Strands

10.90.1 place-helix-here

`place-helix-here` [function]

add a helix

Add a helix somewhere close to this point in the map, try to fit the orientation. Add to a molecule called "Helix", create it if needed. Create another moecule called "Reverse Helix" if the helix orientation isn't completely unequivocal.

Returns: the index of the new molecule.

10.90.2 place-strand-here

`place-strand-here` *n_residues n_sample_strands* [function]

Where:

- *n_residues* is an integer number
- *n_sample_strands* is an integer number

add a strands

Add a strand close to this point in the map, try to fit the orientation. Add to a molecule called "Strand", create it if needed. n_residues is the estimated number of residues in the strand.

n_sample_strands is the number of strands from the database tested to fit into this strand density. 8 is a suggested number. 20 for a more rigourous search, but it will be slower.

Returns: the index of the new molecule.

10.90.3 find-helices

`find-helices` [function]

autobuild helices

Find secondary structure in the current map. Add to a molecule called "Helices", create it if needed.

Returns: the index of the new molecule.

10.90.4 find-strands

`find-strands` [function]

autobuild strands

Find secondary structure in the current map. Add to a molecule called "Strands", create it if needed.

Returns: the index of the new molecule.

10.90.5 find-secondary-structure

`find-secondary-structure` *use_helix helix_length helix_target use_strand strand_length strand_target* [function]

Where:

- *use_helix* is an integer number
- *helix_length* is an integer number
- *helix_target* is an integer number
- *use_strand* is an integer number
- *strand_length* is an integer number
- *strand_target* is an integer number

autobuild secondary structure

Find secondary structure in the current map. Add to a molecule called "SecStruc", create it if needed.

Returns: the index of the new molecule.

10.90.6 find-secondary-structure-local

`find-secondary-structure-local` *use_helix helix_length helix_target use_strand strand_length strand_target radius* [function]

Where:

- *use_helix* is an integer number
- *helix_length* is an integer number
- *helix_target* is an integer number
- *use_strand* is an integer number
- *strand_length* is an integer number
- *strand_target* is an integer number
- *radius* is a number

autobuild secondary structure

Find secondary structure local to current view in the current map. Add to a molecule called "SecStruc", create it if needed.

Returns: the index of the new molecule.

10.91 RNA/DNA

10.91.1 ideal-nucleic-acid

`ideal-nucleic-acid` *RNA_or_DNA form single_stranged_flag sequence* [function]

Where:

- *RNA_or_DNA* is a string
- *form* is a string
- *single_stranged_flag* is an integer number
- *sequence* is a string

create a molecule of idea nucleotides

use the given sequence (single letter code)

RNA_or_DNA is either "RNA" or "DNA"

form is either "A" or "B"

Returns: the new molecule number or -1 if a problem

10.91.2 wrapped-nucleotide-builder-dialog

`wrapped-nucleotide-builder-dialog` [function]

10.91.3 ideal-nucleic-acid-by-widget

`ideal-nucleic-acid-by-widget` *builder_dialog* [function]

Where *builder_dialog* is a GtkWidget

10.91.4 watson-crick-pair

`watson-crick-pair` *imol chain_id resno* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number

10.91.5 setup-base-pairing

`setup-base-pairing` *state* [function]

Where *state* is an integer number

10.92 Sequence (Assignment)

10.92.1 print-sequence-chain

`print-sequence-chain` *imol chain_id* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string

Print the sequence to the console of the given molecule.

10.92.2 assign-fasta-sequence

`assign-fasta-sequence` *imol chain_id_in seq* [function]

Where:

- *imol* is an integer number
- *chain_id_in* is a string
- *seq* is a string

Assign a FASTA sequence to a given chain in the molecule.

10.92.3 assign-pir-sequence

`assign-pir-sequence` *imol chain_id_in seq* [function]

Where:

- *imol* is an integer number
- *chain_id_in* is a string
- *seq* is a string

Assign a PIR sequence to a given chain in the molecule.

10.92.4 assign-sequence

`assign-sequence` *imol_model imol_map chain_id* [function]

Where:

- *imol_model* is an integer number
- *imol_map* is an integer number
- *chain_id* is a string

10.92.5 assign-sequence-from-file

`assign-sequence-from-file` *imol file* [function]

Where:

- *imol* is an integer number
- *file* is a string

Assign a sequence to a given molecule from (whatever) sequence file.

10.92.6 assign-sequence-from-string

`assign-sequence-from-string` *imol chain_id_in seq* [function]

Where:

- *imol* is an integer number
- *chain_id_in* is a string
- *seq* is a string

Assign a sequence to a given molecule from a simple string.

10.92.7 delete-all-sequences-from-molecule

`delete-all-sequences-from-molecule` *imol* [function]

Where *imol* is an integer number

Delete all the sequences from a given molecule.

10.92.8 delete-sequence-by-chain-id

`delete-sequence-by-chain-id` *imol chain_id_in* [function]

Where:

- *imol* is an integer number
- *chain_id_in* is a string

Delete the sequence for a given chain_id from a given molecule.

10.93 Surfaces

10.93.1 do-surface

`do-surface` *imol istate* [function]

Where:

- *imol* is an integer number
- *istate* is an integer number

draw surface of molecule number imol

if state = 1 draw the surface (normal representation goes away)

if state = 0 don't draw surface

10.94 FFFearing

10.94.1 fffear-search

`fffear-search` *imol_model imol_map* [function]

Where:

- *imol_model* is an integer number
- *imol_map* is an integer number

fffear search model in molecule number imol_model in map number imol_map

10.94.2 set-fffear-angular-resolution

`set-fffear-angular-resolution` *f* [function]

Where *f* is a number

set and return the fffear angular resolution in degrees

10.94.3 fffear-angular-resolution

`fffear-angular-resolution` [function]

return the fffear angular resolution in degrees

10.95 Remote Control

10.95.1 make-socket-listener-maybe

`make-socket-listener-maybe` [function]

try to make socket listener

10.95.2 coot-socket-listener-idle-func

`coot-socket-listener-idle-func` *w* [function]

Where *w* is a GtkWidget

10.95.3 set-coot-listener-socket-state-internal

`set-coot-listener-socket-state-internal` *sock_state* [function]

Where *sock_state* is an integer number

10.95.4 set-socket-string-waiting

`set-socket-string-waiting` *s* [function]
Where *s* is a string

10.95.5 set-remote-control-port

`set-remote-control-port` *port_number* [function]
Where *port_number* is an integer number

10.95.6 get-remote-control-port-number

`get-remote-control-port-number` [function]

10.95.7 set-tip-of-the-day-flag

`set-tip-of-the-day-flag` *state* [function]
Where *state* is an integer number

10.96 Browser Interface

10.96.1 browser-url

`browser-url` *url* [function]
Where *url* is a string
try to open given url in Web browser

10.96.2 set-browser-interface

`set-browser-interface` *browser* [function]
Where *browser* is a string
set command to open the web browser,
examples are "open" or "mozilla"

10.96.3 handle-online-coot-search-request

`handle-online-coot-search-request` *entry_text* [function]
Where *entry_text* is a string
the search interface
find words, construct a url and open it.

10.97 Generic Objects

10.97.1 new-generic-object-number

`new-generic-object-number` *objname* [function]
Where *objname* is a string
create a new generic object with name objname and return the index of the object

10.97.2 to-generic-object-add-line

`to-generic-object-add-line` *object_number colour line_width from_x1 from_y1 from_z1 to_x2 to_y2 to_z2* [function]

Where:

- *object_number* is an integer number
- *colour* is a string
- *line_width* is an integer number
- *from_x1* is a number
- *from_y1* is a number
- *from_z1* is a number
- *to_x2* is a number
- *to_y2* is a number
- *to_z2* is a number

add line to generic object object_number

10.97.3 to-generic-object-add-point

`to-generic-object-add-point` *object_number colour point_width from_x1 from_y1 from_z1* [function]

Where:

- *object_number* is an integer number
- *colour* is a string
- *point_width* is an integer number
- *from_x1* is a number
- *from_y1* is a number
- *from_z1* is a number

add point to generic object object_number

10.97.4 to-generic-object-add-display-list-handle

`to-generic-object-add-display-list-handle` *object_number display_list_id* [function]

Where:

- *object_number* is an integer number
- *display_list_id* is an integer number

add a display list handle generic object

10.97.5 set-display-generic-object

`set-display-generic-object` *object_number istate* [function]

Where:

- *object_number* is an integer number

- *istate* is an integer number

set the display status of object number object_number,

when they are created, by default objects are not displayed, so we generally need this function.

10.97.6 generic-object-is-displayed-p

`generic-object-is-displayed-p` *object_number* [function]

Where *object_number* is an integer number

is generic display object displayed?

Returns: 1 for yes, otherwise 0

10.97.7 generic-object-index

`generic-object-index` *name* [function]

Where *name* is a string

return the index of the object with name name, if not, return -1;

10.97.8 number-of-generic-objects

`number-of-generic-objects` [function]

what is the name of generic object number obj_number?

return the number of generic display objects

Returns: 0 (NULL) f on obj_number not available

10.97.9 generic-object-info

`generic-object-info` [function]

print to the console the name and display status of the generic display objects

10.97.10 generic-object-has-objects-p

`generic-object-has-objects-p` *obj_no* [function]

Where *obj_no* is an integer number

does generic display object number obj_no have things to display? (predicate name)

Returns: 0 for no things, 1 for things.

10.97.11 close-generic-object

`close-generic-object` *object_number* [function]

Where *object_number* is an integer number

close generic object, clear the lines/points etc, not available for buttons/displaying etc

10.97.12 is-closed-generic-object-p

`is-closed-generic-object-p` *object_number* [function]

Where *object_number* is an integer number

has the generic object been closed?

Returns: 1 for yes, 0 othersize

10.97.13 generic-object-clear

`generic-object-clear` *object_number* [function]

Where *object_number* is an integer number

clear out the lines and points from object_number, but keep it displayable (not closed).

10.97.14 generic-objects-gui-wrapper

`generic-objects-gui-wrapper` [function]

a kludgey thing, so that the generic objects gui can be called from a callback.

10.98 Molprobity Interface

10.98.1 handle-read-draw-probe-dots

`handle-read-draw-probe-dots` *dots_file* [function]

Where *dots_file* is a string

pass a filename that contains molprobity's probe output in XtalView format

10.98.2 handle-read-draw-probe-dots-unformatted

`handle-read-draw-probe-dots-unformatted` *dots_file imol show_clash_gui_flag* [function]

Where:

- *dots_file* is a string
- *imol* is an integer number
- *show_clash_gui_flag* is an integer number

pass a filename that contains molprobity's probe output in unformatted format

10.98.3 set-do-probe-dots-on-rotamers-and-chis

`set-do-probe-dots-on-rotamers-and-chis` *state* [function]

Where *state* is an integer number

shall we run molprobity for on edit chi angles intermediate atoms?

10.98.4 do-probe-dots-on-rotamers-and-chis-state

`do-probe-dots-on-rotamers-and-chis-state` [function]

return the state of if run molprobity for on edit chi angles intermediate atoms?

10.98.5 set-do-probe-dots-post-refine

`set-do-probe-dots-post-refine` *state* [function]

Where *state* is an integer number

shall we run molprobity after a refinement has happened?

10.98.6 do-probe-dots-post-refine-state

`do-probe-dots-post-refine-state` [function]

show the state of shall we run molprobity after a refinement has happened?

10.98.7 unmangle-hydrogen-name

`unmangle-hydrogen-name` *pdb_hydrogen_name* [function]

Where *pdb_hydrogen_name* is a string

make an attempt to convert pdb hydrogen name to the name used in Coot (and the refmac dictionary, perhaps).

10.98.8 set-interactive-probe-dots-molprobity-radius

`set-interactive-probe-dots-molprobity-radius` *r* [function]

Where *r* is a number

set the radius over which we can run interactive probe, bigger is better but slower.

default is 6.0

10.98.9 interactive-probe-dots-molprobity-radius

`interactive-probe-dots-molprobity-radius` [function]

return the radius over which we can run interactive probe.

10.98.10 probe-available-p

`probe-available-p` [function]

Can we run probe (was the executable variable set properly?) (predicate).

Returns: 1 for yes, 2 for no

10.99 Partial Charges

10.99.1 show-partial-charge-info

`show-partial-charge-info` *imol chain_id resno ins_code* [function]

Where:

- *imol* is an integer number
- *chain_id* is a string
- *resno* is an integer number
- *ins_code* is a string

10.100 EM interface

10.100.1 scale-cell

`scale-cell` *imol_map fac_u fac_v fac_w* [function]

Where:

- *imol_map* is an integer number
- *fac_u* is a number
- *fac_v* is a number
- *fac_w* is a number

10.101 CCP4mg Interface

10.101.1 write-ccp4mg-picture-description

`write-ccp4mg-picture-description` *filename* [function]

Where *filename* is a string

10.101.2 get-atom-colour-from-mol-no

`get-atom-colour-from-mol-no` *imol element* [function]

Where:

- *imol* is an integer number
- *element* is a string

10.102 Dipoles

10.102.1 delete-dipole

`delete-dipole` *imol dipole_number* [function]

Where:

- *imol* is an integer number
- *dipole_number* is an integer number

10.103 Aux functions

10.103.1 laplacian

`laplacian` *imol* [function]

Where *imol* is an integer number

10.104 SMILES

10.104.1 do-smiles-gui

`do-smiles-gui` [function]

display the SMILES string dialog

10.105 PHENIX Support

10.105.1 set-button-label-for-external-refinement

`set-button-label-for-external-refinement` *button_label* [function]

Where *button_label* is a string

set the button label of the external Refinement program

10.106 Graphics Text

10.106.1 place-text

`place-text` *text x y z size* [function]

Where:

- *text* is a string
- *x* is a number
- *y* is a number
- *z* is a number
- *size* is an integer number

Put text at x,y,z.

size variable is currently ignored.

Returns: a text handle

10.106.2 remove-text

`remove-text` *text_handle* [function]

Where *text_handle* is an integer number

Remove text.

10.107 PISA Interaction

10.107.1 pisa-interaction

`pisa-interaction` *imol_1 imol_2* [function]

Where:

- *imol_1* is an integer number
- *imol_2* is an integer number

11 More Scripting Functions

12 Scheme Scripting Functions

Index

I

L

M

N

O

P

R

S

T

U

V

W

Y

Z