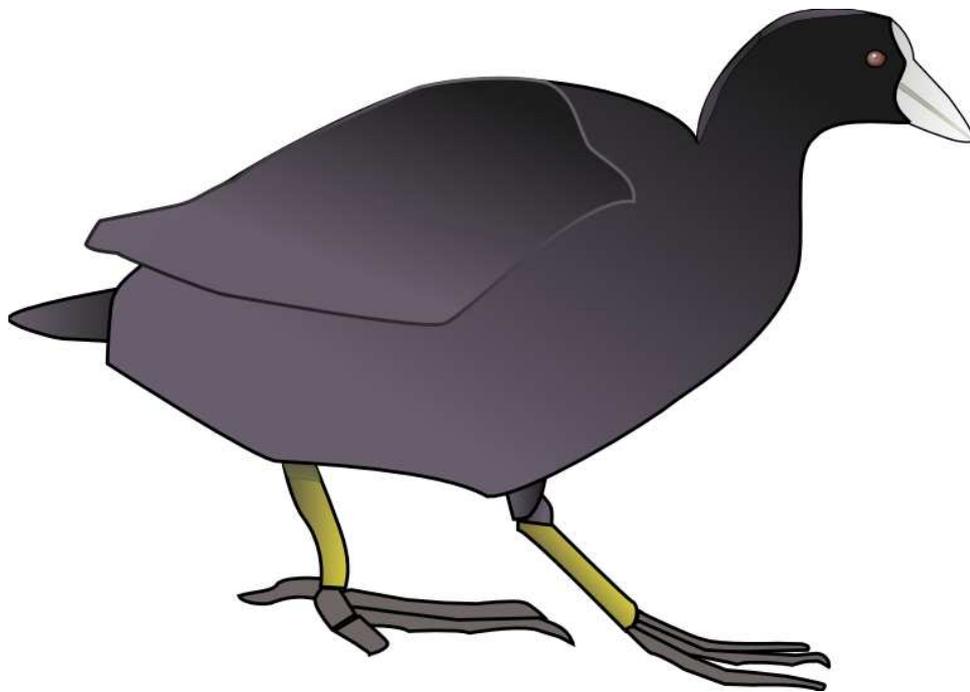


# The Coot User Manual

---

Paul Emsley  
pemsley at mrc dash lmb dot cam dot ac dot uk



This Manual is for Coot version 0.7 and beyond

Copyright © 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012 Paul Emsley

# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Citing Coot and Friends	1
1.2	What is Coot?	1
1.3	What Coot is Not	1
1.4	Hardware Requirements	2
1.4.1	Mouse	2
1.5	Environment Variables	2
1.6	Command Line Arguments	3
1.7	Web Page	4
1.8	Crash	4
<b>2</b>	<b>Mousing and Keyboarding</b>	<b>5</b>
2.1	Next Residue	5
2.2	Keyboard Contouring	5
2.3	Mouse Z Translation and Clipping	5
2.4	Keyboard Translation	6
2.5	Keyboard Zoom and Clip	6
2.6	Scrollwheel	6
2.7	Selecting Atoms	6
2.8	Virtual Trackball	6
2.9	More on Zooming	7
<b>3</b>	<b>General Features</b>	<b>8</b>
3.1	Version number	8
3.2	Antialiasing	8
3.3	Molecule Number	8
3.4	Display Issues	9
3.4.1	Stereo	9
3.4.2	Pick Cursor	9
3.4.3	Origin Marker	9
3.5	Screenshot	9
3.6	Raster3D output	10
3.7	Display Manager	10
3.8	The Modelling Toolbar	11
3.9	The file selector	11
3.9.1	File-name Filtering	11
3.9.2	Filename Sorting	11
3.9.3	Save Coordinates Directory	11
3.10	Scripting	11
3.10.1	Python	12
3.10.1.1	Python Commands	12
3.10.2	Scheme	12

3.10.3	Coot State	12
3.10.4	Key Binding	13
3.10.5	User-Defined Functions	13
3.11	Backups and Undo	13
3.11.1	Redo	14
3.11.2	Restoring from Backup	14
3.12	View Matrix	14
3.13	Space Group and Symmetry	15
3.14	Recentering View	15
3.15	Views	16
3.16	Clipping Manipulation	16
3.17	Background colour	16
3.18	Unit Cell	16
3.19	Rotation Centre Pointer	16
3.20	Orientation Axes	16
3.21	Pointer Distances	17
3.22	Crosshairs	17
3.23	3D Annotations	17
3.24	Frame Rate	17
3.25	Program Output	17
<b>4</b>	<b>Coordinate-Related Features</b>	<b>18</b>
4.1	Reading coordinates	18
4.1.1	A Note on Space Groups Names	18
4.1.2	Read multiple coordinate files	18
4.1.3	SHELX .ins/.res files	18
4.2	Atom Info	19
4.3	Atom Labeling	19
4.4	Atom Colouring	19
4.5	Bond Parameters	20
4.5.1	Bond Thickness	20
4.5.2	Display Hydrogens	20
4.5.3	NCS Ghosts Coordinates	20
4.5.4	NCS Maps	21
4.5.5	Using Strict NCS	21
4.6	Download coordinates	21
4.7	Get Coordinates and Map from EDS	21
4.8	Save Coordinates	22
4.9	Setting the Space Group	22
4.10	Anisotropic Atoms	22
4.11	Symmetry	22
4.11.1	Missing symmetry	23
4.12	Sequence View	23
4.13	Print Sequence	23
4.14	Environment Distances	23
4.15	Distances and Angles	23
4.16	Zero Occupancy Marker	23
4.17	Atomic Dots	24

4.18	Ball and Stick Representation	24
4.19	Mean, Median Temperature Factors	24
4.20	Secondary Structure Matching (SSM)	24
4.21	Least-Squares Fitting	25
4.22	Ligand Overlaying	25
4.23	Writing PDB files	26
<b>5</b>	<b>Modelling and Building</b>	<b>27</b>
5.1	Regularization and Real Space Refinement	27
5.1.1	Dictionary	28
5.1.2	Sphere Refinement	28
5.1.3	Refining Specific Residues	29
5.1.4	Refining Carbohydrates	29
5.1.5	Planar Peptide Restraints	29
5.1.6	The UNK residue type	30
5.1.7	Moving Zero Occupancy Atoms	30
5.2	Changing the Map for Building/Refinement	30
5.3	Rotate/Translate Zone	30
5.4	Rigid Body Refinement	30
5.5	Simplex Refinement	31
5.6	Post-manipulation-hook	31
5.7	Baton Building	31
5.7.1	Undo	32
5.7.2	Missing Skeleton	32
5.7.3	Building Backwards	32
5.8	Reversing Direction of Fragment	33
5.9	C\alpha -> Mainchain	33
5.10	Backbone Torsion Angles	33
5.11	Docking Sidechains	33
5.12	Rotamers	34
5.12.1	Auto Fit Rotamer	34
5.12.1.1	Backrub Rotamers	35
5.12.2	De-clashing residues	35
5.13	Editing chi Angles	35
5.14	Torsion General	36
5.14.1	Ligand Torsion angles	36
5.15	Pep-flip	36
5.16	Add Alternate Conformation	36
5.17	Mutation	37
5.17.1	Mutating DNA/RNA	37
5.17.2	Multiple mutations	37
5.17.3	Mutating to a Non-Standard Residue	38
5.17.4	Mutate and Autofit	38
5.17.5	Renumbering	38
5.18	Importing Lignds/Monomers	38
5.19	Ligand from SMILES strings	38
5.20	Find Ligands	39
5.20.1	Flexible Ligands	39

5.20.2	Adding Ligands to Model	39
5.21	Flip Ligand	40
5.22	Find Waters	40
5.22.1	Refinement Failure	40
5.22.2	Blobs	41
5.23	Add Terminal Residue	41
5.24	Add OXT Atom to Residue	41
5.25	Add Atom at Pointer	42
5.26	Place Helix	42
5.27	Building Ideal DNA and RNA	42
5.28	Merge Molecules	42
5.29	Temperature Factor for New Atoms	43
5.30	Applying NCS Edits	43
5.31	Running Refmac	43
5.32	Running SHELXL	44
5.33	Clear Pending Picks	44
5.34	Delete	44
5.35	Sequence Assignment	45
5.36	Building Links and Loops	45
5.37	Fill Partial Residues	45
5.38	Changing Chain IDs	46
5.39	Setting Occupancies	46
5.40	Fix Nomenclature Errors	46
5.41	Rotamer Fix Whole Protein	46
5.42	Refine All Waters	46
5.43	Moving Molecules/Ligands	47
5.44	Modifying the Labels on the Model/Fit/Refine dialog	47
<b>6</b>	<b>Map-Related Features</b>	<b>48</b>
6.1	Maps in General	48
6.1.1	Map Reading Bug	48
6.2	Create a Map	48
6.2.1	Auto-read MTZ file	48
6.2.2	Reading CIF data	48
6.2.3	Reading PHS data	49
6.3	Map Contouring	49
6.4	Map Extent	50
6.5	Map Contour “Scrolling” Limits	50
6.6	Map Line Width	50
6.7	“Dynamic” Map colouring	50
6.8	Difference Map Colouring	51
6.9	Make a Difference Map	51
6.10	Make an Averaged Map	51
6.11	Map Sampling	51
6.12	Dragged Map	51
6.13	Dynamic Map Sampling and Display Size	51
6.14	Skeletonization	52
6.15	Map Sharpening	52

6.16	Pattersons .....	52
6.17	Map Re-Interpolation .....	52
6.18	Masks .....	53
6.18.1	Example .....	53
6.19	Trimming .....	53
6.20	Map Transformation .....	53
6.21	Export Map .....	54
<b>7</b>	<b>Validation .....</b>	<b>55</b>
7.1	Ramachandran Plots .....	55
7.2	Geometry Analysis .....	55
7.3	Chiral Volumes .....	55
7.3.1	Fixing Chiral Volume Errors .....	56
7.4	Blobs: a.k.a. Unmodelled density .....	56
7.5	Difference Map Peaks .....	56
7.6	Check Waters by Difference Map .....	56
7.7	Molprobit Tools Interface .....	57
7.8	GLN and ASN B-factor Outliers .....	57
7.9	Validation Graphs .....	58
7.9.1	Residue Density Fit .....	58
7.9.2	Rotamer Analysis .....	58
7.9.3	Temperature Factor Variance .....	58
7.9.4	Peptide Omega Angle Distortion .....	58
<b>8</b>	<b>Representation .....</b>	<b>59</b>
8.1	Surfaces .....	59
<b>9</b>	<b>Hints and Usage Tips .....</b>	<b>60</b>
9.1	Documentation .....	60
9.2	Low Resolution .....	60
9.3	Coot Droppings .....	60
9.4	Clearing Backups .....	61
9.5	Getting out of “Translate” Mode .....	61
9.6	Getting out of “Continuous Rotation” Mode .....	61
9.7	Getting out of “Label Atom Only” Mode .....	61
9.8	Button Labels .....	61
9.9	Picking .....	61
9.10	Resizing View .....	62
9.11	Scroll-wheel .....	62
9.12	Slow Computer Configuration .....	62
<b>10</b>	<b>Other Programs .....</b>	<b>63</b>
10.1	findligand .....	63

<b>11</b>	<b>Scripting Functions</b>	<b>64</b>
11.1	Startup Functions	64
11.1.1	set-prefer-python	64
11.1.2	prefer-python	64
11.2	File System Functions	64
11.2.1	make-directory-maybe	64
11.2.2	set-show-paths-in-display-manager	64
11.2.3	show-paths-in-display-manager-state	64
11.2.4	add-coordinates-glob-extension	65
11.2.5	add-data-glob-extension	65
11.2.6	add-dictionary-glob-extension	65
11.2.7	add-map-glob-extension	65
11.2.8	remove-coordinates-glob-extension	65
11.2.9	remove-data-glob-extension	65
11.2.10	remove-dictionary-glob-extension	65
11.2.11	remove-map-glob-extension	65
11.2.12	set-sticky-sort-by-date	66
11.2.13	unset-sticky-sort-by-date	66
11.2.14	set-filter-fileselection-filenames	66
11.2.15	filter-fileselection-filenames-state	66
11.2.16	file-type-coords	66
11.2.17	open-coords-dialog	66
11.2.18	set-file-chooser-selector	66
11.3	Widget Utilities	66
11.3.1	set-main-window-title	66
11.4	MTZ and data handling utilities	67
11.4.1	manage-column-selector	67
11.5	Molecule Info Functions	67
11.5.1	chain-n-residues	67
11.5.2	seqnum-from-serial-number	67
11.5.3	insertion-code-from-serial-number	67
11.5.4	n-models	68
11.5.5	n-chains	68
11.5.6	is-solvent-chain-p	68
11.5.7	is-protein-chain-p	68
11.5.8	is-nucleotide-chain-p	69
11.5.9	n-residues	69
11.5.10	sort-chains	69
11.5.11	sort-residues	69
11.5.12	remarks-dialog	69
11.5.13	print-header-secondary-structure-info	69
11.5.14	copy-molecule	70
11.5.15	add-ligand-delete-residue-copy-molecule	70
11.5.16	exchange-chain-ids-for-seg-ids	70
11.5.17	show-remarks-browser	70
11.6	Library and Utility Functions	70
11.6.1	git-revision-count	70
11.6.2	svn-revision	71

11.6.3	molecule-name	71
11.6.4	set-molecule-name	71
11.6.5	coot-real-exit	71
11.6.6	coot-no-state-real-exit	71
11.6.7	coot-clear-backup-or-real-exit	71
11.6.8	coot-save-state-and-exit	71
11.6.9	first-coords-imol	72
11.6.10	first-small-coords-imol	72
11.6.11	first-unsaved-coords-imol	72
11.6.12	mmcif-sfs-to-mtz	72
11.7	Graphics Utility Functions	72
11.7.1	set-do-anti-aliasing	72
11.7.2	do-anti-aliasing-state	72
11.7.3	set-do-GL-lighting	72
11.7.4	do-GL-lighting-state	73
11.7.5	use-graphics-interface-state	73
11.7.6	python-at-prompt-at-startup-state	73
11.7.7	start-graphics-interface	73
11.7.8	reset-view	73
11.7.9	graphics-n-molecules	73
11.7.10	toggle-idle-spin-function	73
11.7.11	toggle-idle-rock-function	74
11.7.12	set-rocking-factors	74
11.7.13	set-idle-function-rotate-angle	74
11.7.14	idle-function-rotate-angle	74
11.7.15	handle-read-draw-molecule	74
11.7.16	allow-duplicate-sequence-numbers	74
11.7.17	set-convert-to-v2-atom-names	74
11.7.18	handle-read-draw-molecule-with-recentre	75
11.7.19	handle-read-draw-molecule-and-move-molecule-here	75
11.7.20	read-pdb	75
11.7.21	assign-hetatms	75
11.7.22	hetify-residue	75
11.7.23	residue-has-hetatms	76
11.7.24	het-group-n-atoms	76
11.7.25	replace-fragment	76
11.7.26	copy-residue-range	76
11.7.27	clear-and-update-model-molecule-from-file	77
11.7.28	screendump-image	77
11.7.29	check-for-dark-blue-density	77
11.7.30	set-draw-solid-density-surface	77
11.7.31	set-draw-map-standard-lines	77
11.7.32	set-solid-density-surface-opacity	78
11.7.33	set-flat-shading-for-solid-density-surface	78
11.8	Interface Preferences	78
11.8.1	set-scroll-by-wheel-mouse	78
11.8.2	scroll-by-wheel-mouse-state	78
11.8.3	set-default-initial-contour-level-for-map	78

11.8.4	set-default-initial-contour-level-for-difference-map	78
11.8.5	print-view-matrix	79
11.8.6	get-view-quaternion-internal	79
11.8.7	set-view-quaternion	79
11.8.8	apply-ncs-to-view-orientation	79
11.8.9	apply-ncs-to-view-orientation-and-screen-centre	79
11.8.10	set-fps-flag	80
11.8.11	get-fps-flag	80
11.8.12	set-show-origin-marker	80
11.8.13	show-origin-marker-state	80
11.8.14	hide-modelling-toolbar	80
11.8.15	show-modelling-toolbar	80
11.8.16	hide-main-toolbar	80
11.8.17	show-main-toolbar	80
11.8.18	show-model-toolbar-all-icons	80
11.8.19	show-model-toolbar-main-icons	81
11.8.20	reattach-modelling-toolbar	81
11.8.21	set-model-toolbar-docked-position	81
11.8.22	suck-model-fit-dialog	81
11.8.23	add-status-bar-text	81
11.8.24	set-model-fit-refine-dialog-stays-on-top	81
11.8.25	model-fit-refine-dialog-stays-on-top-state	81
11.8.26	accept-reject-dialog-docked-state	81
11.8.27	set-accept-reject-dialog-docked-show	81
11.8.28	accept-reject-dialog-docked-show-state	82
11.9	Mouse Buttons	82
11.9.1	set-control-key-for-rotate	82
11.9.2	control-key-for-rotate-state	82
11.9.3	blob-under-pointer-to-screen-centre	82
11.10	Cursor Function	82
11.10.1	normal-cursor	82
11.10.2	fleur-cursor	82
11.10.3	pick-cursor-maybe	82
11.10.4	rotate-cursor	82
11.10.5	set-pick-cursor-index	83
11.11	Model/Fit/Refine Functions	83
11.11.1	post-model-fit-refine-dialog	83
11.11.2	unset-model-fit-refine-dialog	83
11.11.3	unset-refine-params-dialog	83
11.11.4	show-select-map-dialog	83
11.11.5	set-model-fit-refine-rotate-translate-zone-label	83
11.11.6	set-model-fit-refine-place-atom-at-pointer-label	83
11.11.7	post-other-modelling-tools-dialog	83
11.11.8	set-refinement-move-atoms-with-zero-occupancy	83
11.11.9	refinement-move-atoms-with-zero-occupancy-state	84
11.12	Backup Functions	84
11.12.1	make-backup	84
11.12.2	turn-off-backup	84

11.12.3	turn-on-backup	84
11.12.4	backup-state	84
11.12.5	apply-undo	84
11.12.6	apply-redo	84
11.12.7	set-have-unsaved-changes	84
11.12.8	have-unsaved-changes-p	85
11.12.9	set-undo-molecule	85
11.12.10	show-set-undo-molecule-chooser	85
11.12.11	set-unpathed-backup-file-names	85
11.12.12	unpathed-backup-file-names-state	85
11.12.13	backup-compress-files-state	85
11.12.14	set-backup-compress-files	85
11.13	Recover Session Function	85
11.13.1	recover-session	86
11.14	Map Functions	86
11.14.1	calc-phases-generic	86
11.14.2	map-from-mtz-by-refmac-calc-phases	86
11.14.3	map-from-mtz-by-calc-phases	86
11.14.4	set-scroll-wheel-map	87
11.14.5	set-scrollable-map	87
11.14.6	scroll-wheel-map	87
11.14.7	save-previous-map-colour	87
11.14.8	restore-previous-map-colour	87
11.14.9	set-active-map-drag-flag	87
11.14.10	get-active-map-drag-flag	87
11.14.11	set-last-map-colour	88
11.14.12	set-map-colour	88
11.14.13	set-last-map-sigma-step	88
11.14.14	set-contour-by-sigma-step-by-mol	88
11.14.15	data-resolution	88
11.14.16	model-resolution	89
11.14.17	export-map	89
11.14.18	export-map-fragment	89
11.14.19	export-map-fragment-with-origin-shift	89
11.14.20	difference-map	90
11.14.21	reinterp-map	90
11.14.22	smooth-map	90
11.15	Density Increment	90
11.15.1	set-iso-level-increment	90
11.15.2	set-diff-map-iso-level-increment	90
11.15.3	get-diff-map-iso-level-increment	91
11.15.4	set-diff-map-iso-level-increment-from-text	91
11.15.5	set-map-sampling-rate-text	91
11.15.6	set-map-sampling-rate	91
11.15.7	get-map-sampling-rate	91
11.15.8	change-contour-level	91
11.15.9	set-last-map-contour-level	91
11.15.10	set-last-map-contour-level-by-sigma	92

11.15.11	set-stop-scroll-diff-map	92
11.15.12	set-stop-scroll-iso-map	92
11.15.13	set-stop-scroll-iso-map-level	92
11.15.14	set-stop-scroll-diff-map-level	92
11.15.15	set-residue-density-fit-scale-factor	92
11.16	Density Functions	92
11.16.1	set-map-line-width	92
11.16.2	map-line-width-state	93
11.16.3	make-and-draw-map	93
11.16.4	make-and-draw-map-with-refmac-params	93
11.16.5	make-and-draw-map-with-reso-with-refmac-params	94
11.16.6	valid-labels	94
11.16.7	mtz-file-has-phases-p	94
11.16.8	is-mtz-file-p	95
11.16.9	cns-file-has-phases-p	95
11.16.10	auto-read-do-difference-map-too-state	95
11.16.11	set-auto-read-column-labels	95
11.16.12	set-map-radius	95
11.16.13	set-density-size	95
11.16.14	set-display-intro-string	95
11.16.15	get-map-radius	96
11.16.16	set-esoteric-depth-cue	96
11.16.17	esoteric-depth-cue-state	96
11.16.18	set-swap-difference-map-colours	96
11.16.19	set-map-is-difference-map	96
11.16.20	map-is-difference-map	96
11.16.21	another-level	96
11.16.22	another-level-from-map-molecule-number	97
11.16.23	residue-density-fit-scale-factor	97
11.16.24	density-at-point	97
11.17	Parameters from map	97
11.17.1	mtz-hklin-for-map	97
11.17.2	mtz-fp-for-map	97
11.17.3	mtz-phi-for-map	98
11.17.4	mtz-weight-for-map	98
11.17.5	mtz-use-weight-for-map	98
11.18	PDB Functions	98
11.18.1	write-pdb-file	98
11.18.2	write-residue-range-to-pdb-file	98
11.18.3	quick-save	99
11.19	Info Dialog	99
11.19.1	info-dialog	99
11.19.2	info-dialog-and-text	99
11.20	Refmac Functions	99
11.20.1	set-refmac-counter	99
11.20.2	swap-map-colours	99
11.20.3	set-keep-map-colour-after-refmac	100
11.20.4	keep-map-colour-after-refmac-state	100

11.21	Symmetry Functions	100
11.21.1	set-symmetry-size	100
11.21.2	get-show-symmetry	100
11.21.3	set-show-symmetry-master	100
11.21.4	set-show-symmetry-molecule	100
11.21.5	symmetry-as-calphas	100
11.21.6	get-symmetry-as-calphas-state	101
11.21.7	set-symmetry-molecule-rotate-colour-map	101
11.21.8	symmetry-molecule-rotate-colour-map-state	101
11.21.9	set-symmetry-colour-by-symop	101
11.21.10	set-symmetry-whole-chain	101
11.21.11	set-symmetry-atom-labels-expanded	102
11.21.12	has-unit-cell-state	102
11.21.13	undo-symmetry-view	102
11.21.14	first-molecule-with-symmetry-displayed	102
11.21.15	save-symmetry-coords	102
11.21.16	new-molecule-by-symmetry	103
11.21.17	new-molecule-by-symmetry-with-atom-selection	103
11.21.18	new-molecule-by-symop	104
11.21.19	n-symops	104
11.21.20	set-space-group	105
11.21.21	set-symmetry-shift-search-size	105
11.22	History Functions	105
11.22.1	print-all-history-in-scheme	105
11.22.2	print-all-history-in-python	105
11.22.3	set-console-display-commands-state	105
11.22.4	set-console-display-commands-hilights	105
11.23	State Functions	106
11.23.1	save-state	106
11.23.2	save-state-file	106
11.23.3	save-state-file-py	106
11.23.4	set-save-state-file-name	106
11.23.5	set-run-state-file-status	106
11.23.6	run-state-file	106
11.23.7	run-state-file-maybe	106
11.24	The Virtual Trackball	107
11.24.1	vt-surface	107
11.24.2	vt-surface-status	107
11.25	Clipping Functions	107
11.25.1	set-clipping-back	107
11.25.2	set-clipping-front	107
11.26	Unit Cell interface	107
11.26.1	get-show-unit-cell	107
11.26.2	set-show-unit-cells-all	107
11.26.3	set-show-unit-cell	108
11.27	Colour	108
11.27.1	set-colour-map-rotation-on-read-pdb	108
11.27.2	set-colour-map-rotation-on-read-pdb-flag	108

11.27.3	set-colour-map-rotation-on-read-pdb-c-only-flag	108
11.27.4	set-colour-by-chain	108
11.27.5	set-colour-by-molecule	108
11.27.6	set-symmetry-colour	108
11.28	Map colour	109
11.28.1	set-colour-map-rotation-for-map	109
11.28.2	set-molecule-bonds-colour-map-rotation	109
11.28.3	get-molecule-bonds-colour-map-rotation	109
11.29	Anisotropic Atoms Interface	109
11.29.1	get-limit-aniso	109
11.29.2	get-show-limit-aniso	109
11.29.3	get-show-aniso	109
11.29.4	set-limit-aniso	109
11.29.5	set-show-aniso	110
11.29.6	set-aniso-probability	110
11.29.7	get-aniso-probability	110
11.30	Display Functions	110
11.30.1	set-graphics-window-size	110
11.30.2	set-graphics-window-position	110
11.30.3	store-graphics-window-position	110
11.30.4	graphics-draw	110
11.30.5	zalman-stereo-mode	111
11.30.6	hardware-stereo-mode	111
11.30.7	stereo-mode-state	111
11.30.8	mono-mode	111
11.30.9	side-by-side-stereo-mode	111
11.30.10	set-hardware-stereo-angle-factor	111
11.30.11	hardware-stereo-angle-factor-state	111
11.30.12	set-model-fit-refine-dialog-position	111
11.30.13	set-display-control-dialog-position	112
11.30.14	set-go-to-atom-window-position	112
11.30.15	set-delete-dialog-position	112
11.30.16	set-rotate-translate-dialog-position	112
11.30.17	set-accept-reject-dialog-position	112
11.30.18	set-ramachandran-plot-dialog-position	112
11.30.19	set-edit-chi-angles-dialog-position	113
11.30.20	set-rotamer-selection-dialog-position	113
11.31	Smooth Scrolling	113
11.31.1	set-smooth-scroll-flag	113
11.31.2	get-smooth-scroll	113
11.31.3	set-smooth-scroll-steps	113
11.31.4	set-smooth-scroll-limit	113
11.32	Font Parameters	114
11.32.1	set-font-size	114
11.32.2	get-font-size	114
11.32.3	set-font-colour	114
11.32.4	set-use-stroke-characters	114
11.33	Rotation Centre	114

11.33.1	set-rotation-centre-size	114
11.33.2	recentre-on-read-pdb	114
11.33.3	set-recentre-on-read-pdb	114
11.33.4	set-rotation-centre	115
11.33.5	go-to-ligand	115
11.33.6	set-go-to-ligand-n-atoms-limit	115
11.34	Atom Selection Utilities	115
11.34.1	median-temperature-factor	115
11.34.2	average-temperature-factor	115
11.34.3	standard-deviation-temperature-factor	115
11.34.4	clear-pending-picks	115
11.34.5	set-default-temperature-factor-for-new-atoms	116
11.34.6	default-new-atoms-b-factor	116
11.34.7	set-reset-b-factor-moved-atoms	116
11.34.8	get-reset-b-factor-moved-atoms-state	116
11.34.9	set-atom-attribute	116
11.34.10	set-atom-string-attribute	116
11.34.11	set-residue-name	117
11.35	Skeletonization Interface	117
11.35.1	skeletonize-map	117
11.35.2	unskeletonize-map	117
11.35.3	set-max-skeleton-search-depth	118
11.35.4	set-skeleton-box-size	118
11.36	Skeleton Colour	118
11.36.1	set-skeleton-colour	118
11.37	Read Maps	118
11.37.1	handle-read-ccp4-map	118
11.38	Save Coordinates	118
11.38.1	save-coordinates	118
11.38.2	set-save-coordinates-in-original-directory	119
11.39	Read Phases File Functions	119
11.39.1	read-phs-and-coords-and-make-map	119
11.39.2	read-phs-and-make-map-using-cell-symm-from-previous-mol	119
11.39.3	read-phs-and-make-map-using-cell-symm-from-mol	119
11.39.4	read-phs-and-make-map-using-cell-symm	119
11.39.5	read-phs-and-make-map-with-reso-limits	120
11.40	Graphics Move	120
11.40.1	undo-last-move	120
11.40.2	translate-molecule-by	120
11.40.3	transform-molecule-by	120
11.40.4	transform-zone	121
11.41	Go To Atom Widget Functions	122
11.41.1	post-go-to-atom-window	122
11.41.2	go-to-atom-molecule-number	122
11.41.3	go-to-atom-chain-id	122
11.41.4	go-to-atom-atom-name	122
11.41.5	go-to-atom-residue-number	122

11.41.6	go-to-atom-ins-code	122
11.41.7	go-to-atom-alt-conf	122
11.41.8	set-go-to-atom-chain-residue-atom-name	122
11.41.9	set-go-to-atom-chain-residue-atom-name-full	123
11.41.10	set-go-to-atom-chain-residue-atom-name-no-redraw	123
11.41.11	update-go-to-atom-from-current-position	123
11.41.12	atom-spec-to-atom-index	123
11.41.13	full-atom-spec-to-atom-index	124
11.41.14	update-go-to-atom-window-on-changed-mol	124
11.41.15	update-go-to-atom-window-on-new-mol	124
11.41.16	set-go-to-atom-molecule	124
11.42	Map and Molecule Control	124
11.42.1	post-display-control-window	124
11.42.2	set-map-displayed	125
11.42.3	set-mol-displayed	125
11.42.4	set-display-only-model-mol	125
11.42.5	set-mol-active	125
11.42.6	mol-is-displayed	125
11.42.7	mol-is-active	125
11.42.8	map-is-displayed	126
11.42.9	set-all-maps-displayed	126
11.42.10	set-all-models-displayed-and-active	126
11.42.11	show-spacegroup	126
11.43	Align and Mutate	126
11.43.1	align-and-mutate	126
11.43.2	set-alignment-gap-and-space-penalty	126
11.44	Renumber Residue Range	127
11.44.1	renumber-residue-range	127
11.44.2	change-residue-number	127
11.45	Scripting Interface	127
11.45.1	probe-available-p	127
11.45.2	post-scripting-window	127
11.45.3	post-scheme-scripting-window	127
11.45.4	post-python-scripting-window	128
11.46	Monomer	128
11.46.1	get-coords-for-accession-code	128
11.46.2	run-script	128
11.46.3	run-guile-script	128
11.46.4	run-python-script	128
11.46.5	import-python-module	128
11.47	Regularization and Refinement	128
11.47.1	add-planar-peptide-restraints	128
11.47.2	remove-planar-peptide-restraints	129
11.47.3	set-use-trans-peptide-restraints	129
11.47.4	add-omega-torsion-restraints	129
11.47.5	remove-omega-torsion-restraints	129
11.47.6	set-refinement-immediate-replacement	129
11.47.7	refinement-immediate-replacement-state	129

11.47.8	set-residue-selection-flash-frames-number	129
11.47.9	accept-regularizement	130
11.47.10	set-refine-with-torsion-restraints	130
11.47.11	refine-with-torsion-restraints-state	130
11.47.12	set-matrix	130
11.47.13	matrix-state	130
11.47.14	set-refine-auto-range-step	130
11.47.15	set-refine-max-residues	131
11.47.16	refine-zone-atom-index-define	131
11.47.17	refine-zone	131
11.47.18	refine-auto-range	131
11.47.19	regularize-zone	132
11.47.20	set-dragged-refinement-steps-per-frame	132
11.47.21	dragged-refinement-steps-per-frame	132
11.47.22	set-refinement-refine-per-frame	132
11.47.23	refinement-refine-per-frame-state	132
11.47.24	set-refinement-drag-elasticity	132
11.47.25	set-refine-ramachandran-angles	133
11.47.26	set-fix-chiral-volumes-before-refinement	133
11.47.27	check-chiral-volumes	133
11.47.28	set-show-chiral-volume-errors-dialog	133
11.47.29	set-secondary-structure-restraints-type	133
11.47.30	secondary-structure-restraints-type	133
11.47.31	imol-refinement-map	133
11.47.32	set-imol-refinement-map	134
11.47.33	does-residue-exist-p	134
11.47.34	add-extra-bond-restraint	134
11.47.35	delete-all-extra-restraints	135
11.47.36	delete-extra-restraints-for-residue	135
11.47.37	set-extra-restraints-prosmart-sigma-limits	135
11.47.38	write-interpolated-extra-restraints	135
11.47.39	write-interpolated-models-and-extra-restraints	135
11.47.40	set-use-only-extra-torsion-restraints-for-torsions	136
11.47.41	use-only-extra-torsion-restraints-for-torsions-state	136
11.47.42	show-restraints-editor	136
11.47.43	show-restraints-editor-by-index	136
11.47.44	write-restraints-cif-dictionary	136
11.48	Simplex Refinement Interface	136
11.48.1	fit-residue-range-to-map-by-simplex	136
11.48.2	score-residue-range-fit-to-map	137
11.49	Nomenclature Errors	137
11.49.1	fix-nomenclature-errors	137
11.49.2	set-nomenclature-errors-on-read	137
11.50	Atom Info Interface	137
11.50.1	output-atom-info-as-text	137
11.51	Residue Info	138
11.51.1	residue-info-dialog	138
11.52	Residue Environment Functions	138

11.52.1	set-show-environment-distances	138
11.52.2	set-show-environment-distances-bumps	138
11.52.3	set-show-environment-distances-h-bonds	138
11.52.4	show-environment-distances-state	138
11.52.5	set-environment-distances-distance-limits	139
11.53	Pointer Functions	139
11.53.1	set-show-pointer-distances	139
11.53.2	show-pointer-distances-state	139
11.54	Zoom Functions	139
11.54.1	scale-zoom	139
11.54.2	zoom-factor	139
11.54.3	set-smooth-scroll-do-zoom	139
11.54.4	smooth-scroll-do-zoom	139
11.55	CNS Data Functions	140
11.55.1	handle-cns-data-file	140
11.55.2	handle-cns-data-file-with-cell	140
11.56	mmCIF Functions	140
11.56.1	open-cif-dictionary-file-selector-dialog	140
11.57	SHELXL Functions	140
11.57.1	read-shelx-ins-file	140
11.57.2	write-shelx-ins-file	141
11.58	Validation Functions	141
11.58.1	difference-map-peaks	141
11.58.2	gln-asn-b-factor-outliers	141
11.59	Ramachandran Plot Functions	141
11.59.1	do-ramachandran-plot	141
11.59.2	set-kleywegt-plot-n-diffs	141
11.59.3	set-ramachandran-plot-contour-levels	142
11.59.4	set-ramachandran-plot-background-block-size	142
11.59.5	ramachandran-plot-differences	142
11.59.6	ramachandran-plot-differences-by-chain	142
11.60	Sequence View Interface	142
11.60.1	do-sequence-view	142
11.61	Atom Labelling	143
11.61.1	set-brief-atom-labels	143
11.61.2	brief-atom-labels-state	143
11.62	Screen Rotation	143
11.62.1	rotate-y-scene	143
11.62.2	rotate-x-scene	143
11.62.3	rotate-z-scene	143
11.62.4	spin-zoom-trans	143
11.63	Screen Translation	144
11.63.1	translate-scene-x	144
11.63.2	translate-scene-y	144
11.63.3	translate-scene-z	144
11.64	Views Interface	144
11.64.1	add-view-here	144
11.64.2	add-view-raw	144

11.64.3	remove-named-view	145
11.64.4	remove-view	145
11.64.5	add-view-description	145
11.64.6	add-action-view	145
11.64.7	insert-action-view-after-view	145
11.64.8	save-views	146
11.64.9	clear-all-views	146
11.65	Background Colour	146
11.65.1	set-background-colour	146
11.65.2	redraw-background	146
11.65.3	background-is-black-p	146
11.66	Ligand Fitting Functions	146
11.66.1	set-ligand-acceptable-fit-fraction	146
11.66.2	set-ligand-cluster-sigma-level	147
11.66.3	set-ligand-flexible-ligand-n-samples	147
11.66.4	set-find-ligand-n-top-ligands	147
11.66.5	set-find-ligand-multi-solutions-per-cluster	147
11.66.6	set-find-ligand-mask-waters	147
11.66.7	set-ligand-search-protein-molecule	147
11.66.8	set-ligand-search-map-molecule	148
11.66.9	add-ligand-search-ligand-molecule	148
11.66.10	add-ligand-search-wiggly-ligand-molecule	148
11.66.11	set-find-ligand-here-cluster	148
11.66.12	ligand-expert	148
11.66.13	do-find-ligands-dialog	148
11.66.14	match-ligand-atom-names	148
11.66.15	match-ligand-atom-names-to-comp-id	149
11.66.16	flip-ligand	149
11.67	Water Fitting Functions	150
11.67.1	wrapped-create-find-waters-dialog	150
11.67.2	renumber-waters	150
11.67.3	execute-find-waters-real	150
11.67.4	move-waters-to-around-protein	150
11.67.5	move-hetgroups-to-around-protein	150
11.67.6	max-water-distance	150
11.67.7	set-water-check-spherical-variance-limit	151
11.67.8	set-ligand-water-to-protein-distance-limits	151
11.67.9	set-ligand-water-n-cycles	151
11.67.10	execute-find-blobs	151
11.67.11	split-water	151
11.68	Bond Representation	152
11.68.1	set-default-bond-thickness	152
11.68.2	set-bond-thickness	152
11.68.3	set-bond-thickness-intermediate-atoms	152
11.68.4	set-bond-colour-rotation-for-molecule	152
11.68.5	get-bond-colour-rotation-for-molecule	152
11.68.6	set-default-representation-type	152
11.68.7	get-default-bond-thickness	152

11.68.8	set-draw-zero-occ-markers	153
11.68.9	set-draw-cis-peptide-markups	153
11.68.10	set-draw-hydrogens	153
11.68.11	draw-hydrogens-state	153
11.68.12	set-draw-stick-mode-atoms	153
11.68.13	graphics-to-ca-representation	153
11.68.14	graphics-to-ca-plus-ligands-representation	154
11.68.15	graphics-to-ca-plus-ligands-and-sidechains-representation	154
11.68.16	graphics-to-bonds-no-waters-representation	154
11.68.17	graphics-to-bonds-representation	154
11.68.18	graphics-to-ca-plus-ligands-sec-struct-representation	154
11.68.19	graphics-to-sec-struct-bonds-representation	154
11.68.20	graphics-to-rainbow-representation	154
11.68.21	graphics-to-b-factor-representation	154
11.68.22	graphics-to-b-factor-cas-representation	155
11.68.23	graphics-to-occupancy-representation	155
11.68.24	graphics-to-user-defined-atom-colours-representation	155
11.68.25	graphics-to-user-defined-atom-colours-all-atoms-representation	155
11.68.26	graphics-molecule-bond-type	155
11.68.27	set-b-factor-bonds-scale-factor	155
11.68.28	change-model-molecule-representation-mode	155
11.68.29	set-use-grey-carbons-for-molecule	156
11.68.30	set-grey-carbon-colour	156
11.68.31	make-ball-and-stick	156
11.68.32	clear-ball-and-stick	156
11.68.33	additional-representation-by-string	156
11.68.34	additional-representation-by-attributes	157
11.69	Dots Representation	157
11.69.1	dots	157
11.69.2	set-dots-colour	158
11.69.3	unset-dots-colour	158
11.69.4	clear-dots	158
11.69.5	clear-dots-by-name	158
11.69.6	n-dots-sets	158
11.70	Pep-flip Interface	159
11.70.1	pepflip	159
11.71	Rigid Body Refinement Interface	159
11.71.1	rigid-body-refine-zone	159
11.71.2	set-rigid-body-fit-acceptable-fit-fraction	159
11.72	Add Terminal Residue Functions	159
11.72.1	set-add-terminal-residue-immediate-addition	159
11.72.2	add-terminal-residue	160
11.72.3	add-terminal-residue-using-phi-psi	160
11.72.4	set-add-terminal-residue-default-residue-type	160
11.72.5	set-add-terminal-residue-do-post-refine	160

11.72.6	add-terminal-residue-do-post-refine-state	160
11.73	Delete Residues	161
11.73.1	delete-residue-range	161
11.73.2	delete-residue	161
11.73.3	delete-residue-with-full-spec	161
11.73.4	delete-residue-hydrogens	161
11.73.5	delete-atom	162
11.73.6	delete-residue-sidechain	162
11.73.7	delete-hydrogens	162
11.74	Mainchain Building Functions	162
11.74.1	db-mainchain	162
11.75	Rotamer Functions	163
11.75.1	set-rotamer-search-mode	163
11.75.2	set-rotamer-lowest-probability	163
11.75.3	set-rotamer-check-clashes	163
11.75.4	auto-fit-best-rotamer	163
11.75.5	set-auto-fit-best-rotamer-clash-flag	164
11.75.6	n-rotamers	164
11.75.7	set-residue-to-rotamer-number	164
11.75.8	set-residue-to-rotamer-name	164
11.75.9	fill-partial-residues	165
11.76	180 Flip Side chain	165
11.76.1	do-180-degree-side-chain-flip	165
11.77	Mutate Functions	165
11.77.1	setup-mutate-auto-fit	165
11.77.2	mutate	165
11.77.3	mutate-base	166
11.77.4	set-mutate-auto-fit-do-post-refine	166
11.77.5	mutate-auto-fit-do-post-refine-state	166
11.77.6	set-rotamer-auto-fit-do-post-refine	166
11.77.7	rotamer-auto-fit-do-post-refine-state	166
11.77.8	mutate-single-residue-by-serial-number	166
11.77.9	set-residue-type-chooser-stub-state	167
11.78	Pointer Atom Functions	167
11.78.1	create-pointer-atom-molecule-maybe	167
11.78.2	pointer-atom-molecule	167
11.79	Baton Build Interface Functions	167
11.79.1	set-baton-mode	167
11.79.2	try-set-draw-baton	167
11.79.3	accept-baton-position	167
11.79.4	baton-tip-try-another	168
11.79.5	baton-tip-previous	168
11.79.6	shorten-baton	168
11.79.7	lengthen-baton	168
11.79.8	baton-build-delete-last-residue	168
11.79.9	set-baton-build-params	168
11.80	Crosshairs Interface	168
11.80.1	set-draw-crosshairs	168

11.81	Edit Chi Angles	168
11.81.1	set-find-hydrogen-torsions	169
11.81.2	edit-chi-angles	169
11.81.3	setup-torsion-general	169
11.82	Masks	169
11.82.1	mask-map-by-molecule	169
11.82.2	set-map-mask-atom-radius	169
11.82.3	map-mask-atom-radius	170
11.83	check Waters Interface	170
11.83.1	delete-checked-waters-baddies	170
11.84	Trim	170
11.84.1	trim-molecule-by-map	170
11.85	External Ray-Tracing	170
11.85.1	raster3d	170
11.85.2	set-raster3d-bond-thickness	171
11.85.3	set-raster3d-atom-radius	171
11.85.4	set-raster3d-density-thickness	171
11.85.5	set-renderer-show-atoms	171
11.85.6	set-raster3d-bone-thickness	171
11.85.7	set-raster3d-shadows-enabled	171
11.85.8	set-raster3d-water-sphere	171
11.85.9	raster-screen-shot	171
11.86	Superposition (SSM)	172
11.86.1	superpose	172
11.86.2	superpose-with-chain-selection	172
11.86.3	superpose-with-atom-selection	172
11.87	NCS	173
11.87.1	set-draw-ncs-ghosts	173
11.87.2	draw-ncs-ghosts-state	173
11.87.3	set-ncs-ghost-bond-thickness	173
11.87.4	ncs-update-ghosts	173
11.87.5	make-dynamically-transformed-ncs-maps	173
11.87.6	add-ncs-matrix	174
11.87.7	add-strict-ncs-matrix	174
11.87.8	show-strict-ncs-state	175
11.87.9	set-show-strict-ncs	175
11.87.10	set-ncs-homology-level	175
11.87.11	copy-chain	175
11.87.12	copy-from-ncs-master-to-others	175
11.87.13	copy-residue-range-from-ncs-master-to-others	176
11.87.14	ncs-control-change-ncs-master-to-chain	176
11.87.15	ncs-control-change-ncs-master-to-chain-id	176
11.87.16	ncs-control-display-chain	176
11.88	Helices and Strands	176
11.88.1	place-helix-here	177
11.88.2	place-strand-here	177
11.88.3	place-strand-here-dialog	177
11.88.4	find-helices	177

11.88.5	find-strands	177
11.88.6	find-secondary-structure	178
11.88.7	find-secondary-structure-local	178
11.89	Nucleotides	178
11.89.1	find-nucleic-acids-local	178
11.90	New Molecule by Section Interface	179
11.90.1	new-molecule-by-residue-type-selection	179
11.90.2	new-molecule-by-atom-selection	179
11.90.3	new-molecule-by-sphere-selection	179
11.91	RNA/DNA	179
11.91.1	ideal-nucleic-acid	179
11.91.2	watson-crick-pair	180
11.91.3	watson-crick-pair-for-residue-range	180
11.92	Sequence (Assignment)	180
11.92.1	print-sequence-chain	180
11.92.2	assign-fasta-sequence	181
11.92.3	assign-pir-sequence	181
11.92.4	assign-sequence-from-file	181
11.92.5	assign-sequence-from-string	181
11.92.6	delete-all-sequences-from-molecule	181
11.92.7	delete-sequence-by-chain-id	182
11.93	Surface Interface	182
11.93.1	do-surface	182
11.93.2	set-transparent-electrostatic-surface	182
11.93.3	get-electrostatic-surface-opacity	182
11.94	FFearing	182
11.94.1	fffear-search	182
11.94.2	set-fffear-angular-resolution	183
11.94.3	fffear-angular-resolution	183
11.95	Remote Control	183
11.95.1	make-socket-listener-maybe	183
11.95.2	set-socket-string-waiting	183
11.95.3	set-socket-python-string-waiting	183
11.96	Display Lists for Maps	183
11.96.1	set-display-lists-for-maps	183
11.96.2	display-lists-for-maps-state	183
11.97	Browser Interface	184
11.97.1	browser-url	184
11.97.2	set-browser-interface	184
11.97.3	handle-online-coot-search-request	184
11.98	Molprobit Interface	184
11.98.1	handle-read-draw-probe-dots	184
11.98.2	handle-read-draw-probe-dots-unformatted	184
11.98.3	set-do-probe-dots-on-rotamers-and-chis	184
11.98.4	do-probe-dots-on-rotamers-and-chis-state	185
11.98.5	set-do-probe-dots-post-refine	185
11.98.6	do-probe-dots-post-refine-state	185
11.98.7	unmangle-hydrogen-name	185

11.98.8	set-interactive-probe-dots-molprobity-radius	185
11.98.9	interactive-probe-dots-molprobity-radius	185
11.99	Map Sharpening Interface	185
11.99.1	sharpen	185
11.99.2	set-map-sharpening-scale-limit	185
11.100	Marking Fixed Atom Interface	186
11.100.1	clear-all-fixed-atoms	186
11.101	Partial Charges	186
11.101.1	show-partial-charge-info	186
11.102	EM interface	186
11.102.1	scale-cell	186
11.103	CCP4mg Interface	186
11.103.1	write-ccp4mg-picture-description	186
11.103.2	get-atom-colour-from-mol-no	187
11.104	Aux functions	187
11.104.1	laplacian	187
11.105	SMILES	187
11.105.1	do-smiles-gui	187
11.106	PHENIX Support	187
11.106.1	set-button-label-for-external-refinement	187
11.107	Graphics Text	187
11.107.1	place-text	187
11.107.2	remove-text	188
11.107.3	text-index-near-position	188
11.108	PISA Interaction	188
11.108.1	pisa-interaction	188
11.109	Jiggle Fit	188
11.109.1	fit-to-map-by-random-jiggle	188
11.110	SBase interface	189
11.110.1	get-ccp4srs-monomer-and-dictionary	189
11.110.2	get-sbase-monomer	189
11.111	FLE-View	189
11.111.1	fle-view-set-water-dist-max	189
11.111.2	fle-view-set-h-bond-dist-max	189
11.112	LSQ-improve	189
11.112.1	lsq-improve	189
11.113	single-model view	190
11.113.1	single-model-view-model-number	190
11.113.2	single-model-view-this-model-number	190
11.113.3	single-model-view-next-model-number	190
11.113.4	single-model-view-prev-model-number	190
11.114	graphics 2D ligand view	190
11.114.1	set-show-graphics-ligand-view	190

<b>12</b>	<b>More Scripting Functions</b>	<b>191</b>
12.1	More Symmetry Functions	191
12.1.1	get-symmetry	191
12.2	Extra Map Functions	191
12.2.1	auto-read-make-and-draw-maps	191
12.2.2	auto-read-make-and-draw-maps-from-mtz	191
12.2.3	map-colour-components	191
12.3	Multi-Residue Torsion	191
12.3.1	multi-residue-torsion-fit-scm	191
12.4	Execute Refmac	192
12.4.1	execute-refmac-real	192
12.4.2	refmac-name	192
12.5	Dictionary Functions	193
12.5.1	dictionaries-read	193
12.6	Restraints Interface	193
12.6.1	set-monomer-restraints	193
12.7	Atom Information functions	193
12.7.1	rename-from-serial-number	193
12.7.2	residue-name	193
12.7.3	residue-info	194
12.7.4	chain-fragments-scm	194
12.7.5	add-molecule	194
12.7.6	clear-and-update-molecule	194
12.7.7	active-residue	195
12.7.8	closest-atom-simple-scm	195
12.7.9	closest-atom	195
12.7.10	residues-near-residue	195
12.7.11	residues-near-position-scm	195
12.7.12	hydrogenate-region	196
12.7.13	add-hydrogens-from-file	196
12.8	Refinement with specs	196
12.8.1	regularize-residues	196
12.8.2	mtz-file-name	196
12.8.3	refine-zone-with-full-residue-spec-scm	196
12.9	Water Chain Functions	197
12.9.1	water-chain-from-shelx-ins-scm	197
12.9.2	water-chain-scm	197
12.10	Glyco Tools	197
12.10.1	print-glyco-tree	197
12.11	Spin Search Functions	197
12.11.1	spin-search	197
12.12	Rotamer Scoring	198
12.12.1	score-rotamers-scm	198
12.13	protein-db	198
12.13.1	protein-db-loops	198
12.14	Coot's Hole implementation	199
12.14.1	hole	199
12.14.2	make-link	199

12.15	Drag and Drop Functions .....	199
12.15.1	handle-drag-and-drop-string .....	199
12.16	Map to Model Correlation .....	200
12.16.1	set-map-correlation-atom-radius .....	200
12.16.2	map-to-model-correlation .....	200
12.16.3	map-to-model-correlation-stats .....	200
12.16.4	map-to-model-correlation-per-residue .....	200
12.16.5	qq-plot-map-and-model-scm .....	201
12.16.6	density-score-residue .....	201
12.16.7	map-mean-scm .....	201
12.16.8	map-statistics-scm .....	201
<b>13</b>	<b>Scheme Scripting Functions .....</b>	<b>202</b>
13.1	cc-interface_8hh .....	202
13.2	More Symmetry Functions .....	202
13.2.1	get-symmetry .....	202
13.3	Extra Map Functions .....	202
13.3.1	auto-read-make-and-draw-maps .....	202
13.3.2	auto-read-make-and-draw-maps-from-mtz .....	202
13.3.3	map-colour-components .....	202
13.4	Multi-Residue Torsion .....	202
13.4.1	multi-residue-torsion-fit-scm .....	202
13.5	Execute Refmac .....	203
13.5.1	execute-refmac-real .....	203
13.5.2	refmac-name .....	203
13.6	Dictionary Functions .....	204
13.6.1	dictionaries-read .....	204
13.7	Restraints Interface .....	204
13.7.1	set-monomer-restraints .....	204
13.8	Atom Information functions .....	204
13.8.1	rename-from-serial-number .....	204
13.8.2	residue-name .....	204
13.8.3	residue-info .....	205
13.8.4	chain-fragments-scm .....	205
13.8.5	add-molecule .....	205
13.8.6	clear-and-update-molecule .....	205
13.8.7	active-residue .....	206
13.8.8	closest-atom-simple-scm .....	206
13.8.9	closest-atom .....	206
13.8.10	residues-near-residue .....	206
13.8.11	residues-near-position-scm .....	206
13.8.12	hydrogenate-region .....	207
13.8.13	add-hydrogens-from-file .....	207
13.9	Refinement with specs .....	207
13.9.1	regularize-residues .....	207
13.9.2	mtz-file-name .....	207
13.9.3	refine-zone-with-full-residue-spec-scm .....	207
13.10	Water Chain Functions .....	208

13.10.1	water-chain-from-shelx-ins-scm	208
13.10.2	water-chain-scm	208
13.11	Glyco Tools	208
13.11.1	print-glyco-tree	208
13.12	Spin Search Functions	208
13.12.1	spin-search	208
13.13	Rotamer Scoring	209
13.13.1	score-rotamers-scm	209
13.14	protein-db	209
13.14.1	protein-db-loops	209
13.15	Coot's Hole implementation	210
13.15.1	hole	210
13.15.2	make-link	210
13.16	Drag and Drop Functions	210
13.16.1	handle-drag-and-drop-string	210
13.17	Map to Model Correlation	211
13.17.1	set-map-correlation-atom-radius	211
13.17.2	map-to-model-correlation	211
13.17.3	map-to-model-correlation-stats	211
13.17.4	map-to-model-correlation-per-residue	211
13.17.5	qq-plot-map-and-model-scm	212
13.17.6	density-score-residue	212
13.17.7	map-mean-scm	212
13.17.8	map-statistics-scm	212
13.18	jligand	212
13.19	gui-ligand-sliders	213
13.20	ligand-check	213
13.21	mutate-in-scheme	213
13.22	get-ebi	214
13.23	parse-pisa-xml	214
13.24	shelx	215
13.25	hello	215
13.26	3d-generator-import	216
13.27	clear-backup	216
13.28	gui-add-linked-cho	216
13.29	ncs	217
13.30	raster3d-from-scheme	217
13.31	coot	217
13.32	extra-top-level	218
13.33	fitting	218
13.34	fascinating-things	219
13.35	run-mogul	220
13.36	background-demo	220
13.37	brute-lsqman	220
13.38	coot-crash-catcher	220
13.39	dictionary-generators	221
13.40	generic-objects	221
13.41	contact-score-isolated-ligand	222

13.42	tips-gui	222
13.43	refmac	222
13.44	redefine-functions	223
13.45	group-settings	223
13.46	tips	223
13.47	cns2coot	224
13.48	coot-utils	224
13.49	coot-lsq	235
13.50	c-interface_8h	235
13.51	Startup Functions	235
13.51.1	set-prefer-python	235
13.51.2	prefer-python	235
13.52	File System Functions	235
13.52.1	make-directory-maybe	235
13.52.2	set-show-paths-in-display-manager	236
13.52.3	show-paths-in-display-manager-state	236
13.52.4	add-coordinates-glob-extension	236
13.52.5	add-data-glob-extension	236
13.52.6	add-dictionary-glob-extension	236
13.52.7	add-map-glob-extension	236
13.52.8	remove-coordinates-glob-extension	236
13.52.9	remove-data-glob-extension	237
13.52.10	remove-dictionary-glob-extension	237
13.52.11	remove-map-glob-extension	237
13.52.12	set-sticky-sort-by-date	237
13.52.13	unset-sticky-sort-by-date	237
13.52.14	set-filter-fileselection-filenames	237
13.52.15	filter-fileselection-filenames-state	237
13.52.16	file-type-coords	237
13.52.17	open-coords-dialog	238
13.52.18	set-file-chooser-selector	238
13.53	Widget Utilities	238
13.53.1	set-main-window-title	238
13.54	MTZ and data handling utilities	238
13.54.1	manage-column-selector	238
13.55	Molecule Info Functions	238
13.55.1	chain-n-residues	238
13.55.2	seqnum-from-serial-number	238
13.55.3	insertion-code-from-serial-number	239
13.55.4	n-models	239
13.55.5	n-chains	239
13.55.6	is-solvent-chain-p	239
13.55.7	is-protein-chain-p	240
13.55.8	is-nucleotide-chain-p	240
13.55.9	n-residues	240
13.55.10	sort-chains	240
13.55.11	sort-residues	240
13.55.12	remarks-dialog	241

13.55.13	print-header-secondary-structure-info	241
13.55.14	copy-molecule	241
13.55.15	add-ligand-delete-residue-copy-molecule	241
13.55.16	exchange-chain-ids-for-seg-ids	241
13.55.17	show-remarks-browser	242
13.56	Library and Utility Functions	242
13.56.1	git-revision-count	242
13.56.2	svn-revision	242
13.56.3	molecule-name	242
13.56.4	set-molecule-name	242
13.56.5	coot-real-exit	242
13.56.6	coot-no-state-real-exit	242
13.56.7	coot-clear-backup-or-real-exit	242
13.56.8	coot-save-state-and-exit	243
13.56.9	first-coords-imol	243
13.56.10	first-small-coords-imol	243
13.56.11	first-unsaved-coords-imol	243
13.56.12	mmcif-sfs-to-mtz	243
13.57	Graphics Utility Functions	243
13.57.1	set-do-anti-aliasing	243
13.57.2	do-anti-aliasing-state	243
13.57.3	set-do-GL-lighting	244
13.57.4	do-GL-lighting-state	244
13.57.5	use-graphics-interface-state	244
13.57.6	python-at-prompt-at-startup-state	244
13.57.7	start-graphics-interface	244
13.57.8	reset-view	244
13.57.9	graphics-n-molecules	244
13.57.10	toggle-idle-spin-function	245
13.57.11	toggle-idle-rock-function	245
13.57.12	set-rocking-factors	245
13.57.13	set-idle-function-rotate-angle	245
13.57.14	idle-function-rotate-angle	245
13.57.15	handle-read-draw-molecule	245
13.57.16	allow-duplicate-sequence-numbers	245
13.57.17	set-convert-to-v2-atom-names	245
13.57.18	handle-read-draw-molecule-with-recentre	246
13.57.19	handle-read-draw-molecule-and-move-molecule-here	246
13.57.20	read-pdb	246
13.57.21	assign-hetatms	246
13.57.22	hetify-residue	246
13.57.23	residue-has-hetatms	247
13.57.24	het-group-n-atoms	247
13.57.25	replace-fragment	247
13.57.26	copy-residue-range	247
13.57.27	clear-and-update-model-molecule-from-file	248
13.57.28	screendump-image	248
13.57.29	check-for-dark-blue-density	248

13.57.30	set-draw-solid-density-surface	248
13.57.31	set-draw-map-standard-lines	248
13.57.32	set-solid-density-surface-opacity	249
13.57.33	set-flat-shading-for-solid-density-surface	249
13.58	Interface Preferences	249
13.58.1	set-scroll-by-wheel-mouse	249
13.58.2	scroll-by-wheel-mouse-state	249
13.58.3	set-default-initial-contour-level-for-map	249
13.58.4	set-default-initial-contour-level-for-difference-map	249
13.58.5	print-view-matrix	250
13.58.6	get-view-quaternalion-internal	250
13.58.7	set-view-quaternalion	250
13.58.8	apply-ncs-to-view-orientation	250
13.58.9	apply-ncs-to-view-orientation-and-screen-centre	250
13.58.10	set-fps-flag	251
13.58.11	get-fps-flag	251
13.58.12	set-show-origin-marker	251
13.58.13	show-origin-marker-state	251
13.58.14	hide-modelling-toolbar	251
13.58.15	show-modelling-toolbar	251
13.58.16	hide-main-toolbar	251
13.58.17	show-main-toolbar	251
13.58.18	show-model-toolbar-all-icons	251
13.58.19	show-model-toolbar-main-icons	252
13.58.20	reattach-modelling-toolbar	252
13.58.21	set-model-toolbar-docked-position	252
13.58.22	suck-model-fit-dialog	252
13.58.23	add-status-bar-text	252
13.58.24	set-model-fit-refine-dialog-stays-on-top	252
13.58.25	model-fit-refine-dialog-stays-on-top-state	252
13.58.26	accept-reject-dialog-docked-state	252
13.58.27	set-accept-reject-dialog-docked-show	252
13.58.28	accept-reject-dialog-docked-show-state	253
13.59	Mouse Buttons	253
13.59.1	set-control-key-for-rotate	253
13.59.2	control-key-for-rotate-state	253
13.59.3	blob-under-pointer-to-screen-centre	253
13.60	Cursor Function	253
13.60.1	normal-cursor	253
13.60.2	fleur-cursor	253
13.60.3	pick-cursor-maybe	253
13.60.4	rotate-cursor	253
13.60.5	set-pick-cursor-index	254
13.61	Model/Fit/Refine Functions	254
13.61.1	post-model-fit-refine-dialog	254
13.61.2	unset-model-fit-refine-dialog	254
13.61.3	unset-refine-params-dialog	254
13.61.4	show-select-map-dialog	254

13.61.5	set-model-fit-refine-rotate-translate-zone-label	254
13.61.6	set-model-fit-refine-place-atom-at-pointer-label	254
13.61.7	post-other-modelling-tools-dialog	254
13.61.8	set-refinement-move-atoms-with-zero-occupancy	254
13.61.9	refinement-move-atoms-with-zero-occupancy-state	255
13.62	Backup Functions	255
13.62.1	make-backup	255
13.62.2	turn-off-backup	255
13.62.3	turn-on-backup	255
13.62.4	backup-state	255
13.62.5	apply-undo	255
13.62.6	apply-redo	255
13.62.7	set-have-unsaved-changes	255
13.62.8	have-unsaved-changes-p	256
13.62.9	set-undo-molecule	256
13.62.10	show-set-undo-molecule-chooser	256
13.62.11	set-unpathed-backup-file-names	256
13.62.12	unpathed-backup-file-names-state	256
13.62.13	backup-compress-files-state	256
13.62.14	set-backup-compress-files	256
13.63	Recover Session Function	256
13.63.1	recover-session	257
13.64	Map Functions	257
13.64.1	calc-phases-generic	257
13.64.2	map-from-mtz-by-refmac-calc-phases	257
13.64.3	map-from-mtz-by-calc-phases	257
13.64.4	set-scroll-wheel-map	258
13.64.5	set-scrollable-map	258
13.64.6	scroll-wheel-map	258
13.64.7	save-previous-map-colour	258
13.64.8	restore-previous-map-colour	258
13.64.9	set-active-map-drag-flag	258
13.64.10	get-active-map-drag-flag	258
13.64.11	set-last-map-colour	259
13.64.12	set-map-colour	259
13.64.13	set-last-map-sigma-step	259
13.64.14	set-contour-by-sigma-step-by-mol	259
13.64.15	data-resolution	259
13.64.16	model-resolution	260
13.64.17	export-map	260
13.64.18	export-map-fragment	260
13.64.19	export-map-fragment-with-origin-shift	260
13.64.20	difference-map	261
13.64.21	reinterp-map	261
13.64.22	smooth-map	261
13.65	Density Increment	261
13.65.1	set-iso-level-increment	261
13.65.2	set-diff-map-iso-level-increment	261

13.65.3	get-diff-map-iso-level-increment	262
13.65.4	set-diff-map-iso-level-increment-from-text	262
13.65.5	set-map-sampling-rate-text	262
13.65.6	set-map-sampling-rate	262
13.65.7	get-map-sampling-rate	262
13.65.8	change-contour-level	262
13.65.9	set-last-map-contour-level	262
13.65.10	set-last-map-contour-level-by-sigma	263
13.65.11	set-stop-scroll-diff-map	263
13.65.12	set-stop-scroll-iso-map	263
13.65.13	set-stop-scroll-iso-map-level	263
13.65.14	set-stop-scroll-diff-map-level	263
13.65.15	set-residue-density-fit-scale-factor	263
13.66	Density Functions	263
13.66.1	set-map-line-width	263
13.66.2	map-line-width-state	264
13.66.3	make-and-draw-map	264
13.66.4	make-and-draw-map-with-refmac-params	264
13.66.5	make-and-draw-map-with-reso-with-refmac-params	265
13.66.6	valid-labels	265
13.66.7	mtz-file-has-phases-p	265
13.66.8	is-mtz-file-p	266
13.66.9	cns-file-has-phases-p	266
13.66.10	auto-read-do-difference-map-too-state	266
13.66.11	set-auto-read-column-labels	266
13.66.12	set-map-radius	266
13.66.13	set-density-size	266
13.66.14	set-display-intro-string	266
13.66.15	get-map-radius	267
13.66.16	set-esoteric-depth-cue	267
13.66.17	esoteric-depth-cue-state	267
13.66.18	set-swap-difference-map-colours	267
13.66.19	set-map-is-difference-map	267
13.66.20	map-is-difference-map	267
13.66.21	another-level	267
13.66.22	another-level-from-map-molecule-number	268
13.66.23	residue-density-fit-scale-factor	268
13.66.24	density-at-point	268
13.67	Parameters from map	268
13.67.1	mtz-hklin-for-map	268
13.67.2	mtz-fp-for-map	268
13.67.3	mtz-phi-for-map	269
13.67.4	mtz-weight-for-map	269
13.67.5	mtz-use-weight-for-map	269
13.68	PDB Functions	269
13.68.1	write-pdb-file	269
13.68.2	write-residue-range-to-pdb-file	269
13.68.3	quick-save	270

13.69	Info Dialog	270
13.69.1	info-dialog	270
13.69.2	info-dialog-and-text	270
13.70	Refmac Functions	270
13.70.1	set-refmac-counter	270
13.70.2	swap-map-colours	270
13.70.3	set-keep-map-colour-after-refmac	271
13.70.4	keep-map-colour-after-refmac-state	271
13.71	Symmetry Functions	271
13.71.1	set-symmetry-size	271
13.71.2	get-show-symmetry	271
13.71.3	set-show-symmetry-master	271
13.71.4	set-show-symmetry-molecule	271
13.71.5	symmetry-as-calphas	271
13.71.6	get-symmetry-as-calphas-state	272
13.71.7	set-symmetry-molecule-rotate-colour-map	272
13.71.8	symmetry-molecule-rotate-colour-map-state	272
13.71.9	set-symmetry-colour-by-symop	272
13.71.10	set-symmetry-whole-chain	272
13.71.11	set-symmetry-atom-labels-expanded	273
13.71.12	has-unit-cell-state	273
13.71.13	undo-symmetry-view	273
13.71.14	first-molecule-with-symmetry-displayed	273
13.71.15	save-symmetry-coords	273
13.71.16	new-molecule-by-symmetry	274
13.71.17	new-molecule-by-symmetry-with-atom-selection	274
13.71.18	new-molecule-by-symop	275
13.71.19	n-symops	275
13.71.20	set-space-group	276
13.71.21	set-symmetry-shift-search-size	276
13.72	History Functions	276
13.72.1	print-all-history-in-scheme	276
13.72.2	print-all-history-in-python	276
13.72.3	set-console-display-commands-state	276
13.72.4	set-console-display-commands-hilights	276
13.73	State Functions	277
13.73.1	save-state	277
13.73.2	save-state-file	277
13.73.3	save-state-file-py	277
13.73.4	set-save-state-file-name	277
13.73.5	set-run-state-file-status	277
13.73.6	run-state-file	277
13.73.7	run-state-file-maybe	277
13.74	The Virtual Trackball	278
13.74.1	vt-surface	278
13.74.2	vt-surface-status	278
13.75	Clipping Functions	278
13.75.1	set-clipping-back	278

13.75.2	set-clipping-front	278
13.76	Unit Cell interface	278
13.76.1	get-show-unit-cell	278
13.76.2	set-show-unit-cells-all	278
13.76.3	set-show-unit-cell	279
13.77	Colour	279
13.77.1	set-colour-map-rotation-on-read-pdb	279
13.77.2	set-colour-map-rotation-on-read-pdb-flag	279
13.77.3	set-colour-map-rotation-on-read-pdb-c-only-flag	279
13.77.4	set-colour-by-chain	279
13.77.5	set-colour-by-molecule	279
13.77.6	set-symmetry-colour	279
13.78	Map colour	280
13.78.1	set-colour-map-rotation-for-map	280
13.78.2	set-molecule-bonds-colour-map-rotation	280
13.78.3	get-molecule-bonds-colour-map-rotation	280
13.79	Anisotropic Atoms Interface	280
13.79.1	get-limit-aniso	280
13.79.2	get-show-limit-aniso	280
13.79.3	get-show-aniso	280
13.79.4	set-limit-aniso	280
13.79.5	set-show-aniso	281
13.79.6	set-aniso-probability	281
13.79.7	get-aniso-probability	281
13.80	Display Functions	281
13.80.1	set-graphics-window-size	281
13.80.2	set-graphics-window-position	281
13.80.3	store-graphics-window-position	281
13.80.4	graphics-draw	281
13.80.5	zalman-stereo-mode	282
13.80.6	hardware-stereo-mode	282
13.80.7	stereo-mode-state	282
13.80.8	mono-mode	282
13.80.9	side-by-side-stereo-mode	282
13.80.10	set-hardware-stereo-angle-factor	282
13.80.11	hardware-stereo-angle-factor-state	282
13.80.12	set-model-fit-refine-dialog-position	282
13.80.13	set-display-control-dialog-position	283
13.80.14	set-go-to-atom-window-position	283
13.80.15	set-delete-dialog-position	283
13.80.16	set-rotate-translate-dialog-position	283
13.80.17	set-accept-reject-dialog-position	283
13.80.18	set-ramachandran-plot-dialog-position	283
13.80.19	set-edit-chi-angles-dialog-position	284
13.80.20	set-rotamer-selection-dialog-position	284
13.81	Smooth Scrolling	284
13.81.1	set-smooth-scroll-flag	284
13.81.2	get-smooth-scroll	284

13.81.3	set-smooth-scroll-steps	284
13.81.4	set-smooth-scroll-limit	284
13.82	Font Parameters	285
13.82.1	set-font-size	285
13.82.2	get-font-size	285
13.82.3	set-font-colour	285
13.82.4	set-use-stroke-characters	285
13.83	Rotation Centre	285
13.83.1	set-rotation-centre-size	285
13.83.2	recentre-on-read-pdb	285
13.83.3	set-recentre-on-read-pdb	285
13.83.4	set-rotation-centre	286
13.83.5	go-to-ligand	286
13.83.6	set-go-to-ligand-n-atoms-limit	286
13.84	Atom Selection Utilities	286
13.84.1	median-temperature-factor	286
13.84.2	average-temperature-factor	286
13.84.3	standard-deviation-temperature-factor	286
13.84.4	clear-pending-picks	286
13.84.5	set-default-temperature-factor-for-new-atoms	287
13.84.6	default-new-atoms-b-factor	287
13.84.7	set-reset-b-factor-moved-atoms	287
13.84.8	get-reset-b-factor-moved-atoms-state	287
13.84.9	set-atom-attribute	287
13.84.10	set-atom-string-attribute	287
13.84.11	set-residue-name	288
13.85	Skeletonization Interface	288
13.85.1	skeletonize-map	288
13.85.2	unskeletonize-map	288
13.85.3	set-max-skeleton-search-depth	289
13.85.4	set-skeleton-box-size	289
13.86	Skeleton Colour	289
13.86.1	set-skeleton-colour	289
13.87	Read Maps	289
13.87.1	handle-read-ccp4-map	289
13.88	Save Coordinates	289
13.88.1	save-coordinates	289
13.88.2	set-save-coordinates-in-original-directory	290
13.89	Read Phases File Functions	290
13.89.1	read-phs-and-coords-and-make-map	290
13.89.2	read-phs-and-make-map-using-cell-symm-from-previous-mol	290
13.89.3	read-phs-and-make-map-using-cell-symm-from-mol	290
13.89.4	read-phs-and-make-map-using-cell-symm	290
13.89.5	read-phs-and-make-map-with-reso-limits	291
13.90	Graphics Move	291
13.90.1	undo-last-move	291
13.90.2	translate-molecule-by	291

13.90.3	transform-molecule-by	291
13.90.4	transform-zone	292
13.91	Go To Atom Widget Functions	293
13.91.1	post-go-to-atom-window	293
13.91.2	go-to-atom-molecule-number	293
13.91.3	go-to-atom-chain-id	293
13.91.4	go-to-atom-atom-name	293
13.91.5	go-to-atom-residue-number	293
13.91.6	go-to-atom-ins-code	293
13.91.7	go-to-atom-alt-conf	293
13.91.8	set-go-to-atom-chain-residue-atom-name	293
13.91.9	set-go-to-atom-chain-residue-atom-name-full	294
13.91.10	set-go-to-atom-chain-residue-atom-name-no-redraw	294
13.91.11	update-go-to-atom-from-current-position	294
13.91.12	atom-spec-to-atom-index	294
13.91.13	full-atom-spec-to-atom-index	295
13.91.14	update-go-to-atom-window-on-changed-mol	295
13.91.15	update-go-to-atom-window-on-new-mol	295
13.91.16	set-go-to-atom-molecule	295
13.92	Map and Molecule Control	295
13.92.1	post-display-control-window	295
13.92.2	set-map-displayed	296
13.92.3	set-mol-displayed	296
13.92.4	set-display-only-model-mol	296
13.92.5	set-mol-active	296
13.92.6	mol-is-displayed	296
13.92.7	mol-is-active	296
13.92.8	map-is-displayed	297
13.92.9	set-all-maps-displayed	297
13.92.10	set-all-models-displayed-and-active	297
13.92.11	show-spacegroup	297
13.93	Align and Mutate	297
13.93.1	align-and-mutate	297
13.93.2	set-alignment-gap-and-space-penalty	297
13.94	Renumber Residue Range	298
13.94.1	renumber-residue-range	298
13.94.2	change-residue-number	298
13.95	Scripting Interface	298
13.95.1	probe-available-p	298
13.95.2	post-scripting-window	298
13.95.3	post-scheme-scripting-window	298
13.95.4	post-python-scripting-window	299
13.96	Monomer	299
13.96.1	get-coords-for-accession-code	299
13.96.2	run-script	299
13.96.3	run-guile-script	299
13.96.4	run-python-script	299
13.96.5	import-python-module	299

13.97	Regularization and Refinement	299
13.97.1	add-planar-peptide-restraints	299
13.97.2	remove-planar-peptide-restraints	300
13.97.3	set-use-trans-peptide-restraints	300
13.97.4	add-omega-torsion-restraints	300
13.97.5	remove-omega-torsion-restraints	300
13.97.6	set-refinement-immediate-replacement	300
13.97.7	refinement-immediate-replacement-state	300
13.97.8	set-residue-selection-flash-frames-number	300
13.97.9	accept-regularization	301
13.97.10	set-refine-with-torsion-restraints	301
13.97.11	refine-with-torsion-restraints-state	301
13.97.12	set-matrix	301
13.97.13	matrix-state	301
13.97.14	set-refine-auto-range-step	301
13.97.15	set-refine-max-residues	302
13.97.16	refine-zone-atom-index-define	302
13.97.17	refine-zone	302
13.97.18	refine-auto-range	302
13.97.19	regularize-zone	303
13.97.20	set-dragged-refinement-steps-per-frame	303
13.97.21	dragged-refinement-steps-per-frame	303
13.97.22	set-refinement-refine-per-frame	303
13.97.23	refinement-refine-per-frame-state	303
13.97.24	set-refinement-drag-elasticity	303
13.97.25	set-refine-ramachandran-angles	304
13.97.26	set-fix-chiral-volumes-before-refinement	304
13.97.27	check-chiral-volumes	304
13.97.28	set-show-chiral-volume-errors-dialog	304
13.97.29	set-secondary-structure-restraints-type	304
13.97.30	secondary-structure-restraints-type	304
13.97.31	imol-refinement-map	304
13.97.32	set-imol-refinement-map	305
13.97.33	does-residue-exist-p	305
13.97.34	add-extra-bond-restraint	305
13.97.35	delete-all-extra-restraints	306
13.97.36	delete-extra-restraints-for-residue	306
13.97.37	set-extra-restraints-prosmart-sigma-limits	306
13.97.38	write-interpolated-extra-restraints	306
13.97.39	write-interpolated-models-and-extra-restraints	306
13.97.40	set-use-only-extra-torsion-restraints-for-torsions	307
13.97.41	use-only-extra-torsion-restraints-for-torsions-state	307
13.97.42	show-restraints-editor	307
13.97.43	show-restraints-editor-by-index	307
13.97.44	write-restraints-cif-dictionary	307
13.98	Simplex Refinement Interface	307
13.98.1	fit-residue-range-to-map-by-simplex	307
13.98.2	score-residue-range-fit-to-map	308

13.99	Nomenclature Errors	308
13.99.1	fix-nomenclature-errors	308
13.99.2	set-nomenclature-errors-on-read	308
13.100	Atom Info Interface	308
13.100.1	output-atom-info-as-text	308
13.101	Residue Info	309
13.101.1	residue-info-dialog	309
13.102	Residue Environment Functions	309
13.102.1	set-show-environment-distances	309
13.102.2	set-show-environment-distances-bumps	309
13.102.3	set-show-environment-distances-h-bonds	309
13.102.4	show-environment-distances-state	309
13.102.5	set-environment-distances-distance-limits	310
13.103	Pointer Functions	310
13.103.1	set-show-pointer-distances	310
13.103.2	show-pointer-distances-state	310
13.104	Zoom Functions	310
13.104.1	scale-zoom	310
13.104.2	zoom-factor	310
13.104.3	set-smooth-scroll-do-zoom	310
13.104.4	smooth-scroll-do-zoom	310
13.105	CNS Data Functions	311
13.105.1	handle-cns-data-file	311
13.105.2	handle-cns-data-file-with-cell	311
13.106	mmCIF Functions	311
13.106.1	open-cif-dictionary-file-selector-dialog	311
13.107	SHELXL Functions	311
13.107.1	read-shelx-ins-file	311
13.107.2	write-shelx-ins-file	312
13.108	Validation Functions	312
13.108.1	difference-map-peaks	312
13.108.2	gln-asn-b-factor-outliers	312
13.109	Ramachandran Plot Functions	312
13.109.1	do-ramachandran-plot	312
13.109.2	set-kleywegt-plot-n-diffs	312
13.109.3	set-ramachandran-plot-contour-levels	313
13.109.4	set-ramachandran-plot-background-block-size	313
13.109.5	ramachandran-plot-differences	313
13.109.6	ramachandran-plot-differences-by-chain	313
13.110	Sequence View Interface	313
13.110.1	do-sequence-view	313
13.111	Atom Labelling	314
13.111.1	set-brief-atom-labels	314
13.111.2	brief-atom-labels-state	314
13.112	Screen Rotation	314
13.112.1	rotate-y-scene	314
13.112.2	rotate-x-scene	314
13.112.3	rotate-z-scene	314

13.112.4	spin-zoom-trans	314
13.113	Screen Translation	315
13.113.1	translate-scene-x	315
13.113.2	translate-scene-y	315
13.113.3	translate-scene-z	315
13.114	Views Interface	315
13.114.1	add-view-here	315
13.114.2	add-view-raw	315
13.114.3	remove-named-view	316
13.114.4	remove-view	316
13.114.5	add-view-description	316
13.114.6	add-action-view	316
13.114.7	insert-action-view-after-view	316
13.114.8	save-views	317
13.114.9	clear-all-views	317
13.115	Background Colour	317
13.115.1	set-background-colour	317
13.115.2	redraw-background	317
13.115.3	background-is-black-p	317
13.116	Ligand Fitting Functions	317
13.116.1	set-ligand-acceptable-fit-fraction	317
13.116.2	set-ligand-cluster-sigma-level	318
13.116.3	set-ligand-flexible-ligand-n-samples	318
13.116.4	set-find-ligand-n-top-ligands	318
13.116.5	set-find-ligand-multi-solutions-per-cluster	318
13.116.6	set-find-ligand-mask-waters	318
13.116.7	set-ligand-search-protein-molecule	318
13.116.8	set-ligand-search-map-molecule	319
13.116.9	add-ligand-search-ligand-molecule	319
13.116.10	add-ligand-search-wiggly-ligand-molecule	319
13.116.11	set-find-ligand-here-cluster	319
13.116.12	ligand-expert	319
13.116.13	do-find-ligands-dialog	319
13.116.14	match-ligand-atom-names	319
13.116.15	match-ligand-atom-names-to-comp-id	320
13.116.16	flip-ligand	320
13.117	Water Fitting Functions	321
13.117.1	wrapped-create-find-waters-dialog	321
13.117.2	renumber-waters	321
13.117.3	execute-find-waters-real	321
13.117.4	move-waters-to-around-protein	321
13.117.5	move-hetgroups-to-around-protein	321
13.117.6	max-water-distance	321
13.117.7	set-water-check-spherical-variance-limit	322
13.117.8	set-ligand-water-to-protein-distance-limits	322
13.117.9	set-ligand-water-n-cycles	322
13.117.10	execute-find-blobs	322
13.117.11	split-water	322

13.118	Bond Representation .....	323
13.118.1	set-default-bond-thickness .....	323
13.118.2	set-bond-thickness .....	323
13.118.3	set-bond-thickness-intermediate-atoms .....	323
13.118.4	set-bond-colour-rotation-for-molecule .....	323
13.118.5	get-bond-colour-rotation-for-molecule .....	323
13.118.6	set-default-representation-type .....	323
13.118.7	get-default-bond-thickness .....	323
13.118.8	set-draw-zero-occ-markers .....	324
13.118.9	set-draw-cis-peptide-markups .....	324
13.118.10	set-draw-hydrogens .....	324
13.118.11	draw-hydrogens-state .....	324
13.118.12	set-draw-stick-mode-atoms .....	324
13.118.13	graphics-to-ca-representation .....	324
13.118.14	graphics-to-ca-plus-ligands-representation .....	325
13.118.15	graphics-to-ca-plus-ligands-and-sidechains-representation .....	325
13.118.16	graphics-to-bonds-no-waters-representation .....	325
13.118.17	graphics-to-bonds-representation .....	325
13.118.18	graphics-to-ca-plus-ligands-sec-struct-representation .....	325
13.118.19	graphics-to-sec-struct-bonds-representation .....	325
13.118.20	graphics-to-rainbow-representation .....	325
13.118.21	graphics-to-b-factor-representation .....	325
13.118.22	graphics-to-b-factor-cas-representation .....	326
13.118.23	graphics-to-occupancy-representation .....	326
13.118.24	graphics-to-user-defined-atom-colours-representation .....	326
13.118.25	graphics-to-user-defined-atom-colours-all-atoms-representation .....	326
13.118.26	graphics-molecule-bond-type .....	326
13.118.27	set-b-factor-bonds-scale-factor .....	326
13.118.28	change-model-molecule-representation-mode .....	326
13.118.29	set-use-grey-carbons-for-molecule .....	327
13.118.30	set-grey-carbon-colour .....	327
13.118.31	make-ball-and-stick .....	327
13.118.32	clear-ball-and-stick .....	327
13.118.33	additional-representation-by-string .....	327
13.118.34	additional-representation-by-attributes .....	328
13.119	Dots Representation .....	328
13.119.1	dots .....	328
13.119.2	set-dots-colour .....	329
13.119.3	unset-dots-colour .....	329
13.119.4	clear-dots .....	329
13.119.5	clear-dots-by-name .....	329
13.119.6	n-dots-sets .....	329
13.120	Pep-flip Interface .....	330

13.120.1	pepflip.....	330
13.121	Rigid Body Refinement Interface.....	330
13.121.1	rigid-body-refine-zone.....	330
13.121.2	set-rigid-body-fit-acceptable-fit-fraction .....	330
13.122	Add Terminal Residue Functions.....	330
13.122.1	set-add-terminal-residue-immediate-addition.....	330
13.122.2	add-terminal-residue .....	331
13.122.3	add-terminal-residue-using-phi-psi.....	331
13.122.4	set-add-terminal-residue-default-residue-type .....	331
13.122.5	set-add-terminal-residue-do-post-refine .....	331
13.122.6	add-terminal-residue-do-post-refine-state .....	331
13.123	Delete Residues.....	332
13.123.1	delete-residue-range.....	332
13.123.2	delete-residue .....	332
13.123.3	delete-residue-with-full-spec.....	332
13.123.4	delete-residue-hydrogens .....	332
13.123.5	delete-atom.....	333
13.123.6	delete-residue-sidechain .....	333
13.123.7	delete-hydrogens.....	333
13.124	Mainchain Building Functions .....	333
13.124.1	db-mainchain .....	333
13.125	Rotamer Functions .....	334
13.125.1	set-rotamer-search-mode .....	334
13.125.2	set-rotamer-lowest-probability .....	334
13.125.3	set-rotamer-check-clashes .....	334
13.125.4	auto-fit-best-rotamer.....	334
13.125.5	set-auto-fit-best-rotamer-clash-flag .....	335
13.125.6	n-rotamers.....	335
13.125.7	set-residue-to-rotamer-number .....	335
13.125.8	set-residue-to-rotamer-name.....	335
13.125.9	fill-partial-residues .....	336
13.126	180 Flip Side chain .....	336
13.126.1	do-180-degree-side-chain-flip .....	336
13.127	Mutate Functions.....	336
13.127.1	setup-mutate-auto-fit .....	336
13.127.2	mutate .....	336
13.127.3	mutate-base .....	337
13.127.4	set-mutate-auto-fit-do-post-refine.....	337
13.127.5	mutate-auto-fit-do-post-refine-state.....	337
13.127.6	set-rotamer-auto-fit-do-post-refine.....	337
13.127.7	rotamer-auto-fit-do-post-refine-state.....	337
13.127.8	mutate-single-residue-by-serial-number .....	337
13.127.9	set-residue-type-chooser-stub-state .....	338
13.128	Pointer Atom Functions .....	338
13.128.1	create-pointer-atom-molecule-maybe .....	338
13.128.2	pointer-atom-molecule .....	338
13.129	Baton Build Interface Functions .....	338
13.129.1	set-baton-mode.....	338

13.129.2	try-set-draw-baton	338
13.129.3	accept-baton-position	338
13.129.4	baton-tip-try-another	339
13.129.5	baton-tip-previous	339
13.129.6	shorten-baton	339
13.129.7	lengthen-baton	339
13.129.8	baton-build-delete-last-residue	339
13.129.9	set-baton-build-params	339
13.130	Crosshairs Interface	339
13.130.1	set-draw-crosshairs	339
13.131	Edit Chi Angles	339
13.131.1	set-find-hydrogen-torsions	340
13.131.2	edit-chi-angles	340
13.131.3	setup-torsion-general	340
13.132	Masks	340
13.132.1	mask-map-by-molecule	340
13.132.2	set-map-mask-atom-radius	340
13.132.3	map-mask-atom-radius	341
13.133	check Waters Interface	341
13.133.1	delete-checked-waters-baddies	341
13.134	Trim	341
13.134.1	trim-molecule-by-map	341
13.135	External Ray-Tracing	341
13.135.1	raster3d	341
13.135.2	set-raster3d-bond-thickness	342
13.135.3	set-raster3d-atom-radius	342
13.135.4	set-raster3d-density-thickness	342
13.135.5	set-renderer-show-atoms	342
13.135.6	set-raster3d-bone-thickness	342
13.135.7	set-raster3d-shadows-enabled	342
13.135.8	set-raster3d-water-sphere	342
13.135.9	raster-screen-shot	342
13.136	Superposition (SSM)	343
13.136.1	superpose	343
13.136.2	superpose-with-chain-selection	343
13.136.3	superpose-with-atom-selection	343
13.137	NCS	344
13.137.1	set-draw-ncs-ghosts	344
13.137.2	draw-ncs-ghosts-state	344
13.137.3	set-ncs-ghost-bond-thickness	344
13.137.4	ncs-update-ghosts	344
13.137.5	make-dynamically-transformed-ncs-maps	344
13.137.6	add-ncs-matrix	345
13.137.7	add-strict-ncs-matrix	345
13.137.8	show-strict-ncs-state	346
13.137.9	set-show-strict-ncs	346
13.137.10	set-ncs-homology-level	346
13.137.11	copy-chain	346

13.137.12	copy-from-ncs-master-to-others	346
13.137.13	copy-residue-range-from-ncs-master-to-others	347
13.137.14	ncs-control-change-ncs-master-to-chain	347
13.137.15	ncs-control-change-ncs-master-to-chain-id	347
13.137.16	ncs-control-display-chain	347
13.138	Helices and Strands	347
13.138.1	place-helix-here	348
13.138.2	place-strand-here	348
13.138.3	place-strand-here-dialog	348
13.138.4	find-helices	348
13.138.5	find-strands	348
13.138.6	find-secondary-structure	349
13.138.7	find-secondary-structure-local	349
13.139	Nucleotides	349
13.139.1	find-nucleic-acids-local	349
13.140	New Molecule by Section Interface	350
13.140.1	new-molecule-by-residue-type-selection	350
13.140.2	new-molecule-by-atom-selection	350
13.140.3	new-molecule-by-sphere-selection	350
13.141	RNA/DNA	350
13.141.1	ideal-nucleic-acid	350
13.141.2	watson-crick-pair	351
13.141.3	watson-crick-pair-for-residue-range	351
13.142	Sequence (Assignment)	351
13.142.1	print-sequence-chain	351
13.142.2	assign-fasta-sequence	352
13.142.3	assign-pir-sequence	352
13.142.4	assign-sequence-from-file	352
13.142.5	assign-sequence-from-string	352
13.142.6	delete-all-sequences-from-molecule	352
13.142.7	delete-sequence-by-chain-id	353
13.143	Surface Interface	353
13.143.1	do-surface	353
13.143.2	set-transparent-electrostatic-surface	353
13.143.3	get-electrostatic-surface-opacity	353
13.144	FFFearing	353
13.144.1	fffear-search	353
13.144.2	set-fffear-angular-resolution	354
13.144.3	fffear-angular-resolution	354
13.145	Remote Control	354
13.145.1	make-socket-listener-maybe	354
13.145.2	set-socket-string-waiting	354
13.145.3	set-socket-python-string-waiting	354
13.146	Display Lists for Maps	354
13.146.1	set-display-lists-for-maps	354
13.146.2	display-lists-for-maps-state	354
13.147	Browser Interface	355
13.147.1	browser-url	355

13.147.2	set-browser-interface	355
13.147.3	handle-online-coot-search-request	355
13.148	Molprobity Interface	355
13.148.1	handle-read-draw-probe-dots	355
13.148.2	handle-read-draw-probe-dots-unformatted	355
13.148.3	set-do-probe-dots-on-rotamers-and-chis	355
13.148.4	do-probe-dots-on-rotamers-and-chis-state	356
13.148.5	set-do-probe-dots-post-refine	356
13.148.6	do-probe-dots-post-refine-state	356
13.148.7	unmangle-hydrogen-name	356
13.148.8	set-interactive-probe-dots-molprobity-radius	356
13.148.9	interactive-probe-dots-molprobity-radius	356
13.149	Map Sharpening Interface	356
13.149.1	sharpen	356
13.149.2	set-map-sharpening-scale-limit	356
13.150	Marking Fixed Atom Interface	357
13.150.1	clear-all-fixed-atoms	357
13.151	Partial Charges	357
13.151.1	show-partial-charge-info	357
13.152	EM interface	357
13.152.1	scale-cell	357
13.153	CCP4mg Interface	357
13.153.1	write-ccp4mg-picture-description	357
13.153.2	get-atom-colour-from-mol-no	358
13.154	Aux functions	358
13.154.1	laplacian	358
13.155	SMILES	358
13.155.1	do-smiles-gui	358
13.156	PHENIX Support	358
13.156.1	set-button-label-for-external-refinement	358
13.157	Graphics Text	358
13.157.1	place-text	358
13.157.2	remove-text	359
13.157.3	text-index-near-position	359
13.158	PISA Interaction	359
13.158.1	pisa-interaction	359
13.159	Jiggle Fit	359
13.159.1	fit-to-map-by-random-jiggle	359
13.160	SBase interface	360
13.160.1	get-ccp4srs-monomer-and-dictionary	360
13.160.2	get-sbase-monomer	360
13.161	FLE-View	360
13.161.1	fle-view-set-water-dist-max	360
13.161.2	fle-view-set-h-bond-dist-max	360
13.162	LSQ-improve	360
13.162.1	lsq-improve	360
13.163	single-model view	361
13.163.1	single-model-view-model-number	361

13.163.2	single-model-view-this-model-number	361
13.163.3	single-model-view-next-model-number	361
13.163.4	single-model-view-prev-model-number	361
13.164	graphics 2D ligand view	361
13.164.1	set-show-graphics-ligand-view	361
13.165	libcheck	362
13.166	check-for-updates	362
13.167	snarf-coot-docs	362
13.168	coot-gui	363
13.169	jligand-gui	367
13.170	remote-control	367
13.171	americanisms	368
13.172	mutate-from-scheme	369
13.173	filter	369
13.174	user-define-restraints	369
13.175	get-recent-pdbe	369
<b>Concept Index</b>		<b>371</b>
<b>Function Index</b>		<b>374</b>

# 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:

"Features and Development of Coot" P Emsley, B Lohkamp, W Scott, and K Cowtan *Acta Cryst.* (2010). D66, 486-501 *Acta Crystallographica Section D-Biological Crystallography* **66**: 486-501

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<sup>1</sup>

---

<sup>1</sup> 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<sup>2</sup>
- a protein crystallographic suite<sup>3</sup>.

## 1.4 Hardware Requirements

The code is designed to be portable to any Unix-like operating system. Coot certainly runs on RedHat Linux of various sorts, Fedora, Ubuntu, Debian, SuSe Linux and MacOS X. There is also a Window port (called WinCoot).

If you want to port to some other operating system, you are welcome<sup>4</sup>. 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)<sup>5</sup>.

## 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”<sup>6</sup>
- `COOT_SCHEME_DIR` The directory containing standard (part of the distribution) scheme files
- `COOT_SCHEME_EXTRAS_DIR` A ':'-separated list of directories containing bespoke scheme files. This variable is not set by default. If you set it, Coot will test each ':'-separated string that it points to a directory, and if it does, Coot will load all the `.scm` files in that directory.
- `COOT_PYTHON_EXTRAS_DIR` A ':'-separated list of directories containing bespoke python files. This variable is not set by default. If you set it, Coot will test each ':'-separated string that it points to a directory, and if it does, Coot will load all the `.py` 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  $C\alpha$  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_SBASE_DIR` The directory to find the SBASE dictionary (often comes with CCP4).

---

<sup>2</sup> although it does have a local refinement algorithm it is no substitute for refmac (a wrapper for refmac is available).

<sup>3</sup> that’s the job of the CCP4 Program Suite.

<sup>4</sup> it’s Free Software after all and I could give you a hand.

<sup>5</sup> I can get by with a one button Macintosh - but it’s not ideal.

<sup>6</sup> as it is known in Clipper.

- `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).

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 than 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 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
- `--update-self` command-line mode to update the coot to the latest pre-release on the server
- `--python` an argument with no parameters - used to tell Coot that the `-c` arguments should be processed as python (rather than as scheme).
- `--small-screen` start with smaller icons and font to fit on small screen displays
- `--zalman-stereo` start in Zalman stereo mode

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://www2.mrc-lmb.cam.ac.uk/personal/pemsley/coot/>

There you can read more about the CCP4 molecular graphics project in general and other projects which are important for Coot<sup>7</sup>.

## 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`<sup>8</sup>. 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<sup>9</sup> 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<sup>10</sup> - especially if you use the debugger (gdb) and send a backtrace too<sup>11</sup>. 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.

---

<sup>7</sup> coot has several influences and dependencies, but these will not be discussed here in the User Manual.

<sup>8</sup> COOT\_BACKUP\_DIR is used in preference if set

<sup>9</sup> The map-reading problem (documented in Section Section 6.1 [Maps in General], page 48) is already known.

<sup>10</sup> now there's a hostage to fortune.

<sup>11</sup> 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 Chapter 9 [Hints and Usage Tips], page 60 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 27) or Rigid Body Fit Zone (Section Section 5.4 [Rigid Body Refinement], page 30)). Select atoms with the Left-mouse. See also Picking (Section Section 9.9 [sec\_picking], page 61).

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

## 2.8 Virtual Trackball

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

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.

---

<sup>1</sup> See also Section 2.9 [more on zooming], page 7

<sup>2</sup> 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.

<sup>3</sup> (`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<sup>4</sup>.

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

---

<sup>4</sup> 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<sup>1</sup> describing the particular features and are documented in Chapter Chapter 5 [Modelling and Building], page 27.

- 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 Antialiasing

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 activated setting the environment variable `__GL_FSAA_MODE`. For me a setting of 5 works nicely and gives a better image than using Coot’s built-in antialiasing.

Also for nVidia graphics card users, there is the application `nvidia-settings`:

`Antialiasing Setting -> Override Application Settings` and slide the slider to the right. On restarting Coot, it should be in antialias mode<sup>2</sup>.

### 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.

---

<sup>1</sup> 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).

<sup>2</sup> 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 3.16 [Clipping Manipulation], page 16).

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"<sup>3</sup>.

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<sup>4</sup>. 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<sup>5</sup> (*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 "C $\alpha$ " representation There is also available "No Waters" and "C $\alpha$  + ligands" representations.

<sup>3</sup> Povray support is only semi-working, there is a problem with the orientation of the image.

<sup>4</sup> to a limited extent.

<sup>5</sup> 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<sup>6</sup> 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.

---

<sup>6</sup> 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<sup>7</sup>. Files<sup>8</sup> 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.*

---

<sup>7</sup> which causes the evaluation of the command.

<sup>8</sup> 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<sup>9</sup> 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<sup>10</sup>).

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.

### 3.10.5 User-Defined Functions

“Power users” of Coot might also like to write their own functions that occur after picking an atom (or a number of atoms)

```
(user-defined-click n_clicks udfunc)
```

define a function `func` which runs after the user has made `n_clicked` atom picks. `func` is called with a list of atom specifiers - the first member of which is the molecule number.

## 3.11 Backups and Undo

By default, each time a modification is made to a model, the old coordinates are written out<sup>11</sup>. The backups are kept in a backup directory and are tagged with the date and the history number (lower numbers are more ancient<sup>12</sup>). 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<sup>13</sup>.

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

---

<sup>9</sup> in that particular directory.

<sup>10</sup> funny that

<sup>11</sup> 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.

<sup>12</sup> The coordinates are written in pdb format - that’s OK, isn’t it?.

<sup>13</sup> 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).

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)
```

If for reasons of strange system<sup>14</sup> 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<sup>15</sup>.

### 3.11.2 Restoring from Backup

There may be certain circumstances<sup>16</sup> 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))
```

---

<sup>14</sup> or system manager.

<sup>15</sup> It works like the “Forwards” buttons in a web browser - which is not available immediately after viewing a new page.

<sup>16</sup> for example, if coot crashes.

### 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<sup>17</sup>:

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

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

```
(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.

Sometimes molecular replacement solutions (for example) create models with chains non-optimally placed relative to each other - a symmetry-related copy would be more appealing (but would be equivalent, crystallographically). For example, to move the B chain to a symmetry-related position:

Centre on an atom in the symmetry-related B chain (where you want the B chain to be)

Extensions -> Modelling -> Symm Shift Reference Chain Here.

### 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 Ca.
- 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).

---

<sup>17</sup> if no space group has been assigned it returns ‘‘No spacegroup for this molecule’’

### 3.15 Views

Coot has a views interface (you might call them "scenes") that define a particular orientation, zoom and view centre. Coot and linearly interpolate between the views. The animation play back speed can be set with the "Views Play Speed" menu item - default is a speed of 10.

The views interface can be found under the Extensions menu item.

### 3.16 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<sup>18</sup>. The clipping can also be changed using keyboard "D" and "F".

It can also be changed with `Ctrl + Right-mouse` drag up and down. Likewise the screen-Z can be changed with `Ctrl + Right-mouse` left and right<sup>19</sup>.

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).

### 3.17 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).

### 3.18 Unit Cell

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

### 3.19 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`).

### 3.20 Orientation Axes

The green axes showing the orientation of the molecule are displayed by default. To remove them use the scripting function;

```
(set-draw-axes 0)
```

---

<sup>18</sup> 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.

<sup>19</sup> Inspired by PyMol? Yep... sure was!

### 3.21 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.

### 3.22 Crosshairs

Crosshairs can be drawn at the centre of the screen, using either the C key<sup>20</sup> in graphics window or Draw -> Crosshairs.... The ticks are at 1.54Å, 2.7Å and 3.8Å.

### 3.23 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.

### 3.24 Frame Rate

Sometimes, you might ask yourself “how fast is the computer?”<sup>21</sup>. 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<sup>22</sup> gives an indication of CPU performance.

### 3.25 Program Output

Due to its “in development” nature (at the moment), Coot produces a lot of “console”<sup>23</sup> 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 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.

---

<sup>20</sup> and C again to toggle them off.

<sup>21</sup> compared to some other one.

<sup>22</sup> prompted by changing the contour level.

<sup>23</sup> *i.e.* the terminal in which you started Coot.

## 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.

By default coot does not allow reading coordinates with duplicated sequence numbers. To enable the reading of files with duplicated sequence numbers use the function:

```
(allow-duplicate-sequence-numbers)
```

Coot can read MDL mol 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 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 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<sup>1</sup> 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...`

---

<sup>1</sup> Hue Saturation Value (Intensity).

## 4.5 Bond Parameters

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

The representation style of the molecule that has the active residue (if any) can be changed using the scroll wheel with Ctrl and Shift.

### 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<sup>3</sup>.

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.

---

<sup>2</sup> they can be redisplayed using `(set-draw-hydrogens mol-no 1)`.

<sup>3</sup> 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.

#### 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 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<sup>4</sup>.

#### 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<sup>5</sup>. (e.g. EBI) server<sup>6</sup> (`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<sup>7</sup>.

### 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.* **D60**, 2240-2249.

<sup>4</sup> that also only makes sense in the region of the reference chain.

<sup>5</sup> OCA is "goose" in Spanish (and Italian)

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

<sup>7</sup> which (currently) only a fraction are.

## 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 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<sup>8</sup>. 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<sup>9</sup> for isotropic atoms.

## 4.11 Symmetry

Coordinates symmetry is “dynamic”. Symmetry atoms can be labeled<sup>10</sup>.

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. To show the symmetry operator as a string (rather than a (1-based) index into the list of symmetry operators (as is the default)) Use **Draw -> Show Symmetry> -> Expanded Symmetry Atom Labels**. Coot generates symmetry related atoms by moving the current set close to the origin by a translation, performing the symmetry expansion around the origin and moving the the symmetry coordinates back by applying the inverse of the origin translation. The origin translation is also displayed in curly braces, e.g. “{1 -1 0}”.

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)
```

---

<sup>8</sup> using thermal ellipsoids

<sup>9</sup> in the case of isotropic atoms, ellipsoids are spherical, of course.

<sup>10</sup> symmetry labels are in pale blue/purple and also provide the symmetry operator number and the translations along the a, b and c axes.

The symmetry can be represented as *Cas*. This along with representation of the molecule as *Cas* (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.

## 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<sup>11</sup>.

## 4.15 Distances and Angles

The distance between atoms can be found using `Info -> Distance`<sup>12</sup>. 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.

---

<sup>11</sup> contacts involving two hydrogens or at least one carbon atom are yellow, denoting 'bump'. Hydrogen contacts display cut-off are based on the user-defined maximum distance, but shortened by 0.5Å per hydrogen atom.

<sup>12</sup> Use `Angle` for an angle, of course.

## 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" "heavy-atom-sites" 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)
```

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<sup>13</sup>.

## 4.20 Secondary Structure Matching (SSM)

The excellent SSM algorithm<sup>14</sup> 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 )
```

<sup>13</sup> *e.g.* this molecule was a map or a closed molecule.

<sup>14</sup> the same one as in the CCP4 program SUPERPOSE

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:

- *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)
(apply-lst-matches 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 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 GUI for the function can be found under

Extensions -> Modelling -> Superpose Ligands...

### 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<sup>1</sup>. 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<sup>2</sup>) 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  $\chi^2$  values of the various geometrical restraints for the zone before and after the regularization. Usually the  $\chi^2$  values are considerably decreased - structure idealization such as this should drive the  $\chi^2$  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<sup>3</sup>. 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 side 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

---

<sup>1</sup> given a half-decent window manager

<sup>2</sup> Diamond, R. (1971). A Real-Space Refinement Procedure for Proteins. *Acta Crystallographica* **A27**, 436-452.

<sup>3</sup> but the resulting  $\chi^2$  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 allows 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<sup>4</sup>. 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.

To enable sphere refinement, Right-mouse click in the right hand side of the horizontal toolbutton menu, **Manage buttons -> [Tick] Sphere Refine -> Apply**. You will need a python-enabled Coot to do this.

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

```
(define *sphere-refine-radius* 3.5)

(add-key-binding "Refine residues in a sphere" "R"
  (lambda ()))
```

---

<sup>4</sup> And a few extras, such as phosphate

```
(using-active-atom

(let* ((rc-spec (list aa-chain-id aa-res-no aa-ins-code))
      (ls (residues-near-residue aa-imol rc-spec *sphere-refine-radius*)))
  (refine-residues aa-imol (cons rc-spec ls))))
```

### 5.1.3 Refining Specific Residues

You can specify the residues that you want to refine without using a linear or sphere selection using `refine-residues`. For example:

```
(refine-residues 0 '(("L" 501 "") ("L" 503 "")))
```

will refine residues A501 and A503 (and residue A502 (if it exists) will be an anchoring residue - used in optimizing the link geometry of the atoms in A501 and A503).

### 5.1.4 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 furanose or pyranose in the dictionary, and if they 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”.

Instead of using a sphere to make a residue selection, you can specify the residues directly using `refine-residues`, for example:

```
(refine-residues 0 '(("L" 501 "") ("L" 503 "")))
```

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

### 5.1.5 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...`

### 5.1.6 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<sup>5</sup>.

### 5.1.7 Moving Zero Occupancy Atoms

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

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

## 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<sup>6</sup>, 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<sup>7</sup>, 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<sup>8</sup>. 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)
```

---

<sup>5</sup> unless you are using Buccaneer, of course

<sup>6</sup> if you want to move only one residue, then click the same atom twice.

<sup>7</sup> like Refinement and Regularization

<sup>8</sup> like Regularization and Refinement.

## 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.

## 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 manipulated 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 52). 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)<sup>9</sup>, however it is based on a more recent and considerably larger database.

Little crosses are drawn representing directions in which 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<sup>10</sup>, 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 C $\alpha$ , this is because the next C $\alpha$  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**.

<sup>9</sup> T. J. Oldfield & R. E. Hubbard (1994). "Analysis of C $\alpha$  Geometry in Protein Structures" *Proteins-Structure Function and Genetics* **18(4)** 324 – 337.

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

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<sup>11</sup>.

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<sup>12</sup>.

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 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 C $\alpha$  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 52) 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<sup>13</sup> 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

---

<sup>11</sup> “b” again toggles the mode off.

<sup>12</sup> high-resolution map interpretation is planned.

<sup>13</sup> *i.e.* one less than the starting residue in the forward direction (defaults to 1).

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.

## 5.9 C\alpha -> Mainchain

Mainchain can be generated using a set of C $\alpha$ s as guide-points (such as those from Baton-building) along the line of Esnouf<sup>14</sup> or Jones and coworkers<sup>15</sup>. Briefly, 6-residue fragments of are generated from a list of high-quality<sup>16</sup> structures. The C $\alpha$  atoms of these fragments are matched against overlapping sets of the guide-point C $\alpha$ s. The resulting matches are merged to provide positions for the mainchain (and C $\beta$ ) atoms. This procedure works well for helices and strands, but less well<sup>17</sup> 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 C $\alpha$ s have been built with Baton Mode in Coot.

## 5.10 Backbone Torsion Angles

It is possible to edit the backbone  $\phi$  and  $\psi$  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  $\psi$  and “Rotate Carbonyl” which changes  $\phi$ . Click and drag across the button<sup>18</sup> to rotate the moving atoms in the graphics window. You should know, of course, that making these modifications alter the  $\phi/\psi$  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

<sup>14</sup> R. M. Esnouf “Polyalanine Reconstruction from C $\alpha$  Positions Using the Program CALPHA Can Aid Initial Phasing of Data by Molecular Replacement Procedures” *Acta Cryst.*, D53, 666-672 (1997).

<sup>15</sup> T.A. Jones & S. Thirup “Using known substructures in protein model building and crystallography” *EMBO J.* 5, 819-822 (1986).

<sup>16</sup> and high resolution

<sup>17</sup> *i.e.* there are severely misplaced atoms

<sup>18</sup> as for Rotate/Translate Zone (Section Section 5.3 [Rotate/Translate Zone], page 30).

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 search 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.

## 5.12 Rotamers

The rotamers are generated<sup>19</sup> from the backbone independent sidechain library of the Richardsons group<sup>20</sup>.

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

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

<sup>19</sup> since version 0.4

<sup>20</sup> 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>

*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)

*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" (*vide infra*)) or (ROTAMERSEARCHHIGHRES) (the conventional/high-resolution method using rigid-body fitting).

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

### 5.12.1.1 Backrub Rotamers

By default, Auto Fit Rotamer will switch to "Backrub Rotamer"<sup>21</sup> 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.

### 5.12.2 De-clashing residues

Sometimes you don't have a map<sup>22</sup> but nevertheless there are clashing residues<sup>23</sup> (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  $\chi$  angles (often called "torsions") "by hand" (using "Edit Chi Angles" from the "Model/Fit/Refine" dialog). To edit a residue's  $\chi_1$  press "1": to edit  $\chi_2$ , "2":  $\chi_3$  "3" and  $\chi_4$  "4". Use left-mouse click and drag to change the  $\chi$  value. Use keyboard "0"<sup>24</sup> 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  $\chi$  angles.

<sup>21</sup> "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

<sup>22</sup> for example, in preparation of a model for molecular replacement

<sup>23</sup> atoms of residues that are too close to each other

<sup>24</sup> that's "zero".

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)
```

To edit the rotatable bonds of a ligand using this tool, you will need to have read in the mmCIF dictionary beforehand.

## 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 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 27). 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 0 (*i.e.* the C, N and O atoms are rotated 180° round a line joining the C $\alpha$  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 34). 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<sup>25</sup>. 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<sup>26</sup> will not stop the mutation action occurring.

### 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<sup>27</sup>.

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<sup>28</sup>. 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 imol 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.

For mutation of nucleic acids, use:

(`mutate-nucleotide-range imol chain-id resno-start resno-end sequence`)

<sup>25</sup> Note that selecting a residue type that matches the residue in the graphics will also result in a mutation

<sup>26</sup> *i.e.* the current residue type matches the residue type to which you wish to mutate the residue

<sup>27</sup> *e.g.* after using `Ca -> Mainchain`.

<sup>28</sup> *i.e.* the residue atoms will remain untouched

### 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).

Note that if you are replacing a conventional protein residue with a modified form (*e.g.* replacing a TYR with a phospho-tyrosine or a LYS with an acetyl-lysine) you will need to make sure that the group of the resulting restraints is an **L-peptide** (use **Edit -> Restraints** to check and modify the restraints group. Likewise for modified RNA/DNA nucleotides, you need to specify the group as **RNA** or **DNA** as appropriate.

### 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 Ligands/Monomers

You can import monomers (often ligands) using **File -> Get Monomer...**<sup>29</sup> by providing the 3-letter code of your monomer/ligand. The resulting molecule will be moved so that it is 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 files so that the monomer appears quickly<sup>30</sup>.

## 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**.

---

<sup>29</sup> this is a wrapper around **LIBCHECK**, so you must have CCP4 suite installed for this function to work

<sup>30</sup> rather than running **LIBCHECK** again

## 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<sup>31</sup>. Only molecules with less than 400 atoms are suggested as potential ligands.

If you do not have any molecules with less than 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  $1.0\sigma$ . 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:

```
(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?” checkbox 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 42).

---

<sup>31</sup> you can search for many different ligand types.

## 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 choice 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). Defaults are 2.4 and 3.2Å.

`(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 7.4 [sec\_blobs], page 56).

## 5.23 Add Terminal Residue

This creates a new residue at the C or N terminal extension of the residue clicked by fitting to the map.  $\phi, \psi$  angle pairs are selected at random based on the Ramachandran plot probability (for a generic residue) and fitted to the density. 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<sup>32</sup>. Each of the trial positions are scored according to their fit to the map<sup>33</sup> 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.

Sometimes, particularly with low resolution maps, the added terminal residue will wander off to somewhere inappropriate. This can be addressed in a number of ways:

1. (`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.
2. to anneal the newly added residue back to the clicked residue (no matter where it ended up being positioned): (`set-add-terminal-residue-do-post-refine 1`)
3. (`set-add-terminal-residue-n-phi-psi-trials 200`) will change the number of trials (default is 100). This is useful if you think that Coot needs to search harder to find a good solution to the positioning of the next residue.

## 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 C $\alpha$ , 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.

<sup>32</sup> usually if this still fails after two repetitions then it never seems to work.

<sup>33</sup> The map is selected using "Refine/Regularize Control"

<sup>34</sup> *e.g.* (`add-OXT-to-residue 0 428 "" "A"`)

## 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<sup>35</sup>. 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 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.

The ideal B-form DNA is somewhat under-wound, needing 11 base-pairs to repeat (instead of the expected 10.5). There is no easy fix for this currently.

## 5.28 Merge Molecules

The dialog for this operation 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<sup>36</sup>.

The scripting interface is used like this

```
(merge-molecules molecule-list target-molecule)
```

*e.g.*

```
(merge-molecules (list 1 2) 0)
```

merges molecules 1 and 2 into molecule 0.

---

<sup>35</sup> including sulfate or phosphate ions (in such a case, it is probably useful to do a “Rigid Body Fit Zone” on that new residue).

<sup>36</sup> For example, after a ligand search has been performed.

## 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)
```

If you want to copy a residue range to a specific chain, or specific list of chains (rather than all NCS peer chains) then make a list of the chain-ids that you wish replaced:

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

in this case, just the residues in the "C" chain is replaced.

## 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"<sup>37</sup> until Refmac has terminated<sup>38</sup>.

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")
```

<sup>37</sup> i.e. Coot is idle and ignores all input.

<sup>38</sup> 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. . . .

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  $\sigma_A$  map and a difference map.

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<sup>39</sup> 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

---

<sup>39</sup> such that Coot would subsequently expect an atom selection “pick” in the graphics window.

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<sup>40</sup>. 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<sup>41</sup>. 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)
```

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.

If you want to fill the side chain of just one residue

```
(fill-partial-residue imol chain-id res-no ins-code)
```

this does a auto-fit-best-rotamer and a refinement on the resulting side-chain position.

---

<sup>40</sup> the current single function doesn’t always perform very well in tests

<sup>41</sup> I suspect that there is not one tool that fits for all.

### 5.38 Changing Chain IDs

You can change the chain ids of chains using `Calculate -> Change Chain IDs...`. Coot will block an attempt to change the whole of a chain and the target chain id already exists in the molecule.

If you use the "Residue Range" option then you can insert residues with non-conflicting residue number into pre-existing chains.

### 5.39 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)`

### 5.40 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 `WATCHCHECK`<sup>42</sup> *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.

### 5.41 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.

### 5.42 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).

---

<sup>42</sup> R.W.W. Hooft, G. Vriend, C. Sander, E.E. Abola, Errors in protein structures. *Nature* (1996) **381**, 272-272.

### 5.43 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.

### 5.44 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 your 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 -> Draggd 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<sup>1</sup>, 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 (a -90 degree phase shift will be applied). 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  $F_o$ s and  $\alpha_{calc}$

---

<sup>1</sup> 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  $F_o$ ,  $F_c$  and  $\alpha_c$
- `(read-cif-data-with-phases-fo-fc cif-file-name)` Calculate an difference map using  $F_o$ ,  $F_c$  and  $\alpha_c$ .

### 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 is 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<sup>2</sup>.

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 checkbox 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)

---

<sup>2</sup> like I don't on my Mac.

By default the map radius<sup>3</sup> is 10Å. The default increment to the electron density depends on whether or not this is a difference map (0.05  $e^-/\text{Å}^3$  for a “2Fo-Fc” style map and 0.005  $e^-/\text{Å}^3$  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<sup>4</sup>, 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  $e^-/\text{Å}^3$ .

## 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<sup>5</sup>. 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)
```

---

<sup>3</sup> actually, it’s a box.

<sup>4</sup> 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.

<sup>5</sup> takes you right back to the good old Frodo days, no?

## 6.8 Difference Map Colouring

For some strange reason, some crystallographers<sup>6</sup> 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 gridding, or in the same space group even. The resulting map will be on the same gridding 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<sup>7</sup>.

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.

---

<sup>6</sup> Jan Dohnalek, for instance.

<sup>7</sup> 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”<sup>8</sup>) 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 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...`<sup>9</sup>. 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 +/- 200Å<sup>2</sup>, this can be changed (in this case to 300) using:

```
(set-map-sharpening-scale-limit 300)
```

This currently only works on maps created by reading an MTZ (or other) reflection data file.

## 6.16 Pattersons

Pattersons can be generated using the `make-and-draw-patterson` function. Example usage:

```
(make-and-draw-patterson mtz-file-name f-col sig-f-col)
```

where `use-weights-flag` is either 0 or 1.

*e.g.*

```
(make-and-draw-patterson "native.mtz" "FP_nat" "SIGFP_nat")
```

## 6.17 Map Re-Interpolation

Maps can be re-interpolated to match a reference map.

```
(reinterp-map map-no reference-map-no)
```

will create a copy of `map-no` in the same cell, spacegroup and grid spacing as the `reference-map-no` map.

---

<sup>8</sup> If you're living in Sweden... or Captain Kirk, that is.

<sup>9</sup> you may think it strange that a box has a radius, this is an idiosyncrasy of Coot.

## 6.18 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.

### 6.18.1 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 re mac). Let's say this map is molecule number 2.
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).

## 6.19 Trimming

If you want to remove all the atoms<sup>10</sup> 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.

## 6.20 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 Ångströms).

*point* is a 3-membered list of numbers (centre point in Ångströms).

*radius* is a single number (also in Ångströms).

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*.

---

<sup>10</sup> or set their occupancy to zero

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 transform 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)
```

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.

Here's 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.

## 6.21 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  $\phi/\psi$  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<sup>1</sup>.

When you mouse over a representation of a residue (a little square or triangle<sup>2</sup>) the residue label pops up. The residue is “active” *i.e.* it can be clicked. The “graphics” view changes so that the C $\alpha$  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 Geometry Analysis

A restraints-based geometry analysis of the molecule. The distortion is weighted by atom occupancy. The distortion of the geometry due to links is shared between the contributing residues.

Note that only the first model of a multi-model molecule is analysed.

### 7.3 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<sup>3</sup>. You can inhibit this dialog like this:

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

---

<sup>1</sup> the not-GLY-or-PRO is the most familiar Ramachandran plot.

<sup>2</sup> prolines have a grey outline rather than a black one, triangles are glycines.

<sup>3</sup> but that’s partly the idea, I suppose.

### 7.3.1 Fixing Chiral Volume Errors

There are two obvious ways:

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

## 7.4 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 electron 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<sup>4</sup> at the top.

## 7.5 Difference Map Peaks

This is one of the fastest ways to validate a model and its data (presuming that the difference map comes from a post-refinement mFo-DFc map). It highlights regions where the model and the data do not agree.

Lesser peaks within a certain distance (by default, 2.0Å) of a large peak are not shown. This cuts down on the number of times one is navigated to a particular region because of ripple or other noise peaks around a central peak.

This value can be queried:

```
(difference-map-peaks-max-closeness)
```

and adjusted:

```
(set-difference-map-peaks-max-closeness 0.1)
```

## 7.6 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<sup>5</sup>. 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 something else, perhaps a cation, anion, sulphate or a ligand. If such an atom diffracts anomalously this can be identified and corrected.

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

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

The scripting interface is:

---

<sup>4</sup> and therefore most interesting

<sup>5</sup> or the water should be more properly modelled as anisotropic or a split partial site

```
(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 or some other refinement first first<sup>6</sup>.

## 7.7 Molprobit Tools Interface

The molprobit 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 your ‘~/ .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)
```

## 7.8 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.

---

<sup>6</sup> and remember to check the difference map button in the “Run Refmac” dialog

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.

## 7.9 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.

### 7.9.1 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  $1/(4 * rmd_{map})$  (as a rule of thumb).

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

There is also a GUI to this:

```
Extensions -> Refine... -> Set Density Fit Graph Weight...
```

### 7.9.2 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)
```

### 7.9.3 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.

### 7.9.4 Peptide Omega Angle Distortion

Some variability of the  $\omega$  is to be expected in the peptide bond. But not too much. Anything more than 13 degrees is suspicious. Unexpected peptide bonds show up red by default. If cis peptides *are* to be expected, and should not be marked as bad, then you can tell this to Coot using:

```
Edit -> Preferences -> Geometry -> Cis-Peptides -> No
```

## 8 Representation

### 8.1 Surfaces

Coot uses the surface code from Gruber and Noble (2004).

Coot uses the partial charges of the atoms (the *partial\_charge* field in the *\_chem\_comp\_atom* block) from the charge dictionary item in the refmac (or other) cif dictionary. However, partial charges are only used under certain conditions

- 1) the molecule consists of less than 100 atoms

or

- 2) the number of atoms in the molecule that are hydrogens is at least 15% of the total number of atoms in the molecule

If partial charges are not used, then the fall-back is to use charges from side-chains charged at physiological pH (Arg, Lys, Asp, Glu).

## 9 Hints and Usage Tips

### 9.1 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](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’

### 9.2 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]

### 9.3 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.

## 9.4 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 than 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))
```

## 9.5 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.

## 9.6 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.

## 9.7 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.

## 9.8 Button Labels

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

## 9.9 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.

## 9.10 Resizing View

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

## 9.11 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?`

## 9.12 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-use-stroke-characters 1)` ; default is to use bitmap characters
- `(set-smooth-scroll-steps 8)` ; default 40
- `(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

## 10 Other Programs

### 10.1 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.

## 11 Scripting Functions

### 11.1 Startup Functions

#### 11.1.1 set-prefer-python

`set-prefer-python` [function]  
tell coot that you prefer to run python scripts if/when there is an option to do so.

#### 11.1.2 prefer-python

`prefer-python` [function]  
the python-prefered mode.  
This is available so that the scripting functions know whether on not to put themselves onto in as menu items.  
If you consider using this, consider in preference `use_gui_qm == 2`, which is used elsewhere to stop python functions adding to the gui, when guile-gtk functions have already done so. We should clean up this (rather obscure) interface at some stage.  
return 1 for python is prefered, 0 for not.

### 11.2 File System Functions

#### 11.2.1 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.

#### 11.2.2 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.

#### 11.2.3 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

### 11.2.4 add-coordinates-glob-extension

`add-coordinates-glob-extension ext` [function]

Where *ext* is a string

add an extension to be treated as coordinate files

### 11.2.5 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

### 11.2.6 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

### 11.2.7 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

### 11.2.8 remove-coordinates-glob-extension

`remove-coordinates-glob-extension ext` [function]

Where *ext* is a string

remove an extension to be treated as coordinate files

### 11.2.9 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

### 11.2.10 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

### 11.2.11 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

**11.2.12 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

**11.2.13 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

**11.2.14 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

**11.2.15 filter-fileselection-filenames-state**

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

**11.2.16 file-type-coords**

`file-type-coords` *file\_name* [function]  
 Where *file\_name* is a string  
 is the given file name suitable to be read as coordinates?

**11.2.17 open-coords-dialog**

`open-coords-dialog` [function]  
 display the open coordinates dialog

**11.2.18 set-file-chooser-selector**

`set-file-chooser-selector` *istate* [function]  
 Where *istate* is an integer number  
 this flag set chooser as default for windows, otherwise use selector 0 is selector 1 is  
 chooser

**11.3 Widget Utilities****11.3.1 set-main-window-title**

`set-main-window-title` *s* [function]  
 Where *s* is a string  
 set the main window title.  
 function added for Lothar Esser

## 11.4 MTZ and data handling utilities

### 11.4.1 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

## 11.5 Molecule Info Functions

### 11.5.1 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

### 11.5.2 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

### 11.5.3 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 (scheme False) on failure.

### 11.5.4 n-models

`n-models imol` [function]

Where *imol* is an integer number

the *chain\_id* (string) of the *ichain*-th chain molecule number *imol*

return the number of models in molecule number *imol*

useful for NMR or other such multi-model molecules.

return the number of models or -1 if there was a problem with the given molecule.

Returns: the chain-id

### 11.5.5 n-chains

`n-chains imol` [function]

Where *imol* is an integer number

number of chains in molecule number *imol*

Returns: the number of chains

### 11.5.6 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`)

This wraps the `mmdb` function `isSolventChain()`.

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.

### 11.5.7 is-protein-chain-p

`is-protein-chain-p imol chain_id` [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string

is this a protein chain? [Raw function]

This is a raw interface function, you should generally not use this, but instead use (`is-protein-chain? imol chain-id`)

This wraps the `mmdb` function `isAminoacidChain()`.

Returns: -1 on error, 0 for no, 1 for is "a protein chain". We wouldn't want to be doing rotamer searches and the like on such a chain.

### 11.5.8 is-nucleotide-chain-p

`is-nucleotide-chain-p` *imol chain\_id* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string

is this a nucleic acid chain? [Raw function]

This is a raw interface function, you should generally not use this, but instead use (`is-nucleicacid-chain? imol chain-id`)

This wraps the mmdb function `isNucleotideChain()`. For completeness.

Returns: -1 on error, 0 for no, 1 for is "a nucleicacid chain". We wouldn't want to be doing rotamer searches and the like on such a chain.

### 11.5.9 n-residues

`n-residues` *imol* [function]

Where *imol* is an integer number

return the number of residues in the molecule,

return -1 if this is a map or closed.

### 11.5.10 sort-chains

`sort-chains` *imol* [function]

Where *imol* is an integer number

return a list of the remarks of hte molecule number *imol*

sort the chain ids of the *imol*-th molecule in lexicographical order

### 11.5.11 sort-residues

`sort-residues` *imol* [function]

Where *imol* is an integer number

sort the residues of the *imol*-th molecule

### 11.5.12 remarks-dialog

`remarks-dialog` *imol* [function]

Where *imol* is an integer number

a gui dialog showing remarks header info (for a model molecule).

### 11.5.13 print-header-secondary-structure-info

`print-header-secondary-structure-info` *imol* [function]

Where *imol* is an integer number

simply print secondary structure info to the terminal/console. In future, this could/should return the info.

### 11.5.14 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)

### 11.5.15 add-ligand-delete-residue-copy-molecule

`add-ligand-delete-residue-copy-molecule imol_ligand_new` [function]

`chain_id_ligand_new resno_ligand_new imol_current chain_id_ligand_current`  
`resno_ligand_current`

Where:

- *imol\_ligand\_new* is an integer number
- *chain\_id\_ligand\_new* is a string
- *resno\_ligand\_new* is an integer number
- *imol\_current* is an integer number
- *chain\_id\_ligand\_current* is a string
- *resno\_ligand\_current* is an integer number

Copy a molecule with addition of a ligand and a deletion of current ligand.

This function is used when adding a new (modified) ligand to a structure. It creates a new molecule that is a copy of the current molecule except that the new ligand is added and the current ligand/residue is deleted.

### 11.5.16 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?

### 11.5.17 show-remarks-browser

`show-remarks-browser` [function]

`show the remarks browser`

## 11.6 Library and Utility Functions

### 11.6.1 git-revision-count

`git-revision-count` [function]

return the git revision count for for this build.

### 11.6.2 svn-revision

`svn-revision` [function]  
an alias to `git_revision_count()` for backwards compatibility

### 11.6.3 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 ( -> False in scheme) e.g. "/a/b/c.pdb" for "d/e/f.mtz  
FWT PHWT"

### 11.6.4 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

### 11.6.5 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.

### 11.6.6 coot-no-state-real-exit

`coot-no-state-real-exit retval` [function]  
Where *retval* is an integer number  
exit without writing a state file

### 11.6.7 coot-clear-backup-or-real-exit

`coot-clear-backup-or-real-exit retval` [function]  
Where *retval* is an integer number  
exit coot doing clear-backup maybe

### 11.6.8 coot-save-state-and-exit

`coot-save-state-and-exit retval save_state_flag` [function]  
Where:  

- *retval* is an integer number
- *save\_state\_flag* is an integer number

exit coot, write a state file

### 11.6.9 first-coords-imol

`first-coords-imol` [function]  
What is the molecule number of first coordinates molecule?  
return -1 when there is none.

### 11.6.10 first-small-coords-imol

`first-small-coords-imol` [function]  
molecule number of first small (<400 atoms) molecule.  
return -1 on no such molecule

### 11.6.11 first-unsaved-coords-imol

`first-unsaved-coords-imol` [function]  
What is the molecule number of first unsaved coordinates molecule?  
return -1 when there is none.

### 11.6.12 mmCIF-sfs-to-mtz

`mmCIF-sfs-to-mtz` *cif\_file\_name* *mtz\_file\_name* [function]  
Where:  

- *cif\_file\_name* is a string
- *mtz\_file\_name* is a string

convert the structure factors in *cif\_file\_name* to an mtz file.  
Return 1 on success. Return 0 on a file without Rfree, return -1 on complete failure to write a file.

## 11.7 Graphics Utility Functions

### 11.7.1 set-do-anti-aliasing

`set-do-anti-aliasing` *state* [function]  
Where *state* is an integer number  
set the bond lines to be antialiased

### 11.7.2 do-anti-aliasing-state

`do-anti-aliasing-state` [function]  
return the flag for antialiasing the bond lines

### 11.7.3 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

### 11.7.4 do-GL-lighting-state

**do-GL-lighting-state** [function]  
return the flag for GL lighting

### 11.7.5 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.

### 11.7.6 python-at-prompt-at-startup-state

**python-at-prompt-at-startup-state** [function]  
is the python interpreter at the prompt?  
Returns: 1 for yes, 0 for no.

### 11.7.7 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

### 11.7.8 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.

### 11.7.9 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)

### 11.7.10 toggle-idle-spin-function

**toggle-idle-spin-function** [function]  
Spin spin spin (or not)

### 11.7.11 toggle-idle-rock-function

`toggle-idle-rock-function` [function]  
Rock (not roll) (self-timed)

### 11.7.12 set-rocking-factors

`set-rocking-factors` *width\_scale frequency\_scale* [function]

Where:

- *width\_scale* is a number
- *frequency\_scale* is a number

Settings for the inevitable discontents who dislike the default rocking rates (defaults 1 and 1)

### 11.7.13 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

### 11.7.14 idle-function-rotate-angle

`idle-function-rotate-angle` [function]

what is the idle function rotation angle?

### 11.7.15 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)

### 11.7.16 allow-duplicate-sequence-numbers

`allow-duplicate-sequence-numbers` [function]

enable reading PDB/pdbx files with duplicate sequence numbers

### 11.7.17 set-convert-to-v2-atom-names

`set-convert-to-v2-atom-names` *state* [function]

Where *state* is an integer number

shall we convert nucleotides to match the old dictionary names?

Usually (after 2006 or so) we do not want to do this (given current Coot architecture). Coot should handle the residue synonyms transparently.

default off (0).

### 11.7.18 handle-read-draw-molecule-with-recentre

`handle-read-draw-molecule-with-recentre` *filename* [function]  
*recentre\_on\_read\_pdb\_flag*

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.

### 11.7.19 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.

### 11.7.20 read-pdb

`read-pdb` *filename* [function]  
Where *filename* is a string

read coordinates from *filename*

### 11.7.21 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.

### 11.7.22 hetify-residue

`hetify-residue` *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

if this is not a standard group, then turn the atoms to HETATMs.

Return 1 on atoms changes, 0 on not. Return -1 if residue not found.

### 11.7.23 residue-has-hetatms

`residue-has-hetatms` *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

residue has HETATMs?

return 1 if all atoms of the specified residue are HETATMs, else, return 0. If residue not found, return -1.

### 11.7.24 het-group-n-atoms

`het-group-n-atoms` *comp\_id* [function]

Where *comp\_id* is a string

return the number of non-hydrogen atoms in the given het-group (comp-id).

Return -1 on comp-id not found in dictionary.

### 11.7.25 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*

### 11.7.26 copy-residue-range

`copy-residue-range` *imol\_target chain\_id\_target imol\_reference chain\_id\_reference resno\_range\_start resno\_range\_end* [function]

Where:

- *imol\_target* is an integer number
- *chain\_id\_target* is a string
- *imol\_reference* is an integer number
- *chain\_id\_reference* is a string
- *resno\_range\_start* is an integer number
- *resno\_range\_end* is an integer number

copy the given residue range from the reference chain to the target chain  
*resno\_range\_start* and *resno\_range\_end* are inclusive.

### 11.7.27 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 the given residues from the reference molecule to the target molecule

replace pdb. Fail if *molecule\_number* is not a valid model molecule. Return -1 on failure. Else return *molecule\_number*

### 11.7.28 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

### 11.7.29 check-for-dark-blue-density

`check-for-dark-blue-density` [function]

give a warning dialog if density it too dark (blue)

### 11.7.30 set-draw-solid-density-surface

`set-draw-solid-density-surface` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

sets the density map of the given molecule to be drawn as a (transparent) solid surface.

### 11.7.31 set-draw-map-standard-lines

`set-draw-map-standard-lines` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

toggle for standard lines representation of map.

This turns off/on standard lines representation of map. transparent surface is another representation type.

If you want to just turn off a map, don't use this, use

.

### 11.7.32 set-solid-density-surface-opacity

`set-solid-density-surface-opacity` *imol opacity* [function]

Where:

- *imol* is an integer number
- *opacity* is a number

set the opacity of density surface representation of the given map.

0.0 is totally transparent, 1.0 is completely opaque and (because the objects are no longer depth sorted) considerably faster to render. 0.3 is a reasonable number.

### 11.7.33 set-flat-shading-for-solid-density-surface

`set-flat-shading-for-solid-density-surface` *state* [function]

Where *state* is an integer number

set the flag to do flat shading rather than smooth shading for solid density surface.

Default is 1 (on).

## 11.8 Interface Preferences

### 11.8.1 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

### 11.8.2 scroll-by-wheel-mouse-state

`scroll-by-wheel-mouse-state` [function]

return the internal state of the scroll-wheel map contouring

### 11.8.3 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 initial contour for 2FoFc-style map

in sigma

### 11.8.4 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 initial contour for FoFc-style map

in sigma

### 11.8.5 print-view-matrix

`print-view-matrix` [function]  
print the view matrix to the console, useful for molscript, perhaps

### 11.8.6 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`

### 11.8.7 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.

### 11.8.8 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.

### 11.8.9 apply-ncs-to-view-orientation-and-screen-centre

`apply-ncs-to-view-orientation-and-screen-centre` *imol current\_chain next\_ncs\_chain forward\_flag* [function]  
Where:

- *imol* is an integer number
- *current\_chain* is a string
- *next\_ncs\_chain* is a string
- *forward\_flag* is an integer number

as above, but shift the screen centre also.

### 11.8.10 set-fps-flag

`set-fps-flag` *t* [function]  
Where *t* is an integer number  
set show frame-per-second flag

### 11.8.11 get-fps-flag

`get-fps-flag` [function]  
set the state of show frames-per-second flag

### 11.8.12 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.

### 11.8.13 show-origin-marker-state

`show-origin-marker-state` [function]  
return the origin marker shown? state

### 11.8.14 hide-modelling-toolbar

`hide-modelling-toolbar` [function]  
hide the vertical modelling toolbar in the GTK2 version

### 11.8.15 show-modelling-toolbar

`show-modelling-toolbar` [function]  
show the vertical modelling toolbar in the GTK2 version (the toolbar is shown by default)

### 11.8.16 hide-main-toolbar

`hide-main-toolbar` [function]  
hide the horizontal main toolbar in the GTK2 version

### 11.8.17 show-main-toolbar

`show-main-toolbar` [function]  
show the horizontal main toolbar in the GTK2 version (the toolbar is shown by default)

### 11.8.18 show-model-toolbar-all-icons

`show-model-toolbar-all-icons` [function]  
show all available icons in the modelling toolbar (same as MFR dialog)

### 11.8.19 show-model-toolbar-main-icons

`show-model-toolbar-main-icons` [function]  
show only a selection of icons in the modelling toolbar

### 11.8.20 reattach-modelling-toolbar

`reattach-modelling-toolbar` [function]  
reattach the modelling toolbar to the last attached position

### 11.8.21 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

### 11.8.22 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

### 11.8.23 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).

### 11.8.24 set-model-fit-refine-dialog-stays-on-top

`set-model-fit-refine-dialog-stays-on-top` *istate* [function]  
Where *istate* is an integer number  
model-fit-refine dialog stays on top

### 11.8.25 model-fit-refine-dialog-stays-on-top-state

`model-fit-refine-dialog-stays-on-top-state` [function]  
return the state model-fit-refine dialog stays on top

### 11.8.26 accept-reject-dialog-docked-state

`accept-reject-dialog-docked-state` [function]  
the accept/reject dialog docked state

### 11.8.27 set-accept-reject-dialog-docked-show

`set-accept-reject-dialog-docked-show` *state* [function]  
Where *state* is an integer number  
set the accept/reject dialog docked show state

**11.8.28 accept-reject-dialog-docked-show-state**

`accept-reject-dialog-docked-show-state` [function]  
 what is the accept/reject dialog docked show state?

**11.9 Mouse Buttons****11.9.1 set-control-key-for-rotate**

`set-control-key-for-rotate` *state* [function]  
 Where *state* is an integer number  
 Alternate mode for rotation.  
 Preferred by some, including Dirk Kostrewa. I don't think this mode works properly yet

**11.9.2 control-key-for-rotate-state**

`control-key-for-rotate-state` [function]  
 return the control key rotate state

**11.9.3 blob-under-pointer-to-screen-centre**

`blob-under-pointer-to-screen-centre` [function]  
 Put the blob under the cursor to the screen centre. Check only positive blobs. Useful function if bound to a key.  
 The refinement map must be set. (We can't check all maps because they are not (or may not be) on the same scale).  
 Returns: 1 if successfully found a blob and moved there. return 0 if no move.

**11.10 Cursor Function****11.10.1 normal-cursor**

`normal-cursor` [function]  
 normal cursor

**11.10.2 fleur-cursor**

`fleur-cursor` [function]  
 fleur cursor

**11.10.3 pick-cursor-maybe**

`pick-cursor-maybe` [function]  
 pick cursor maybe

**11.10.4 rotate-cursor**

`rotate-cursor` [function]  
 rotate cursor

### 11.10.5 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

## 11.11 Model/Fit/Refine Functions

### 11.11.1 post-model-fit-refine-dialog

`post-model-fit-refine-dialog` [function]

display the Model/Fit/Refine dialog

### 11.11.2 unset-model-fit-refine-dialog

`unset-model-fit-refine-dialog` [function]

unset model/fit/refine dialog

### 11.11.3 unset-refine-params-dialog

`unset-refine-params-dialog` [function]

unset refine params dialog

### 11.11.4 show-select-map-dialog

`show-select-map-dialog` [function]

display the Display Manager dialog

### 11.11.5 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".

### 11.11.6 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".

### 11.11.7 post-other-modelling-tools-dialog

`post-other-modelling-tools-dialog` [function]

display the Other Modelling Tools dialog

### 11.11.8 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)

### 11.11.9 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?"

## 11.12 Backup Functions

### 11.12.1 make-backup

`make-backup` *imol* [function]  
Where *imol* is an integer number  
make backup for molecule number *imol*

### 11.12.2 turn-off-backup

`turn-off-backup` *imol* [function]  
Where *imol* is an integer number  
turn off backups for molecule number *imol*

### 11.12.3 turn-on-backup

`turn-on-backup` *imol* [function]  
Where *imol* is an integer number  
turn on backups for molecule number *imol*

### 11.12.4 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

### 11.12.5 apply-undo

`apply-undo` [function]  
apply undo - the "Undo" button callback

### 11.12.6 apply-redo

`apply-redo` [function]  
apply redo - the "Redo" button callback

### 11.12.7 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

### 11.12.8 have-unsaved-changes-p

`have-unsaved-changes-p imol` [function]

Where *imol* is an integer number

does molecule number *imol* have unsaved changes?

Returns: -1 on bad *imol*, 0 on no unsaved changes, 1 on has unsaved changes

### 11.12.9 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*

### 11.12.10 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.

### 11.12.11 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

### 11.12.12 unpathed-backup-file-names-state

`unpathed-backup-file-names-state` [function]

return the state for adding paths to backup file names

### 11.12.13 backup-compress-files-state

`backup-compress-files-state` [function]

return the state for compression of backup files

### 11.12.14 set-backup-compress-files

`set-backup-compress-files state` [function]

Where *state* is an integer number

set if backup files will be compressed or not using gzip

## 11.13 Recover Session Function

### 11.13.1 recover-session

**recover-session** [function]  
recover session

After a crash, 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 than the read file. (Not very good, because you need to remember which files you read in before the crash - should be improved.)

## 11.14 Map Functions

### 11.14.1 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()`

### 11.14.2 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.

### 11.14.3 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.

#### 11.14.4 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()`.

#### 11.14.5 set-scrollable-map

`set-scrollable-map` *imol* [function]

Where *imol* is an integer number

return the molecule number to which the mouse scroll wheel is attached

set the map that has its contour level changed by the scrolling the mouse wheel to molecule number *imol* (same as  
).

#### 11.14.6 scroll-wheel-map

`scroll-wheel-map` [function]

the contouring of which map is altered when the scroll wheel changes?

#### 11.14.7 save-previous-map-colour

`save-previous-map-colour` *imol* [function]

Where *imol* is an integer number

save previous colour map for molecule number *imol*

#### 11.14.8 restore-previous-map-colour

`restore-previous-map-colour` *imol* [function]

Where *imol* is an integer number

restore previous colour map for molecule number *imol*

#### 11.14.9 set-active-map-drag-flag

`set-active-map-drag-flag` *t* [function]

Where *t* is an integer number

set the state of immediate map update on map drag.

By default, it is on (*t*=1). On slower computers it might be better to set *t*=0.

#### 11.14.10 get-active-map-drag-flag

`get-active-map-drag-flag` [function]

return the state of the dragged map flag

### 11.14.11 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

### 11.14.12 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 *imol*th map

### 11.14.13 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

### 11.14.14 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)

### 11.14.15 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 Å (eg. 2.0)

### 11.14.16 model-resolution

`model-resolution` *imol* [function]

Where *imol* is an integer number

return the resolution set in the header of the model/coordinates file. If this number is not available, return a number less than 0.

### 11.14.17 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.

### 11.14.18 export-map-fragment

`export-map-fragment` *imol x y z radius filename* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number
- *radius* is a number
- *filename* is a string

export a fragment of the map about (x,y,z)

### 11.14.19 export-map-fragment-with-origin-shift

`export-map-fragment-with-origin-shift` *imol x y z radius filename* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number
- *radius* is a number
- *filename* is a string

export a fragment of the map about (x,y,z)

### 11.14.20 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  $\text{map\_scale} * \text{imap2}$  from *imap1*, on the grid of *imap1*. Return the new molecule number. Return -1 on failure.

### 11.14.21 reinterp-map

`reinterp-map` *map\_no reference\_map\_no* [function]

Where:

- *map\_no* is an integer number
- *reference\_map\_no* is an integer number

make a new map (a copy of *map\_no*) that is in the cell, spacegroup and gridding of the map in *reference\_map\_no*.

Return the new map molecule number - return -1 on failure

### 11.14.22 smooth-map

`smooth-map` *map\_no sampling\_multiplier* [function]

Where:

- *map\_no* is an integer number
- *sampling\_multiplier* is a number

make a new map (a copy of *map\_no*) that is in the cell, spacegroup and a multiple of the sampling of the input map (a sampling factor of more than 1 makes the output maps smoother)

## 11.15 Density Increment

### 11.15.1 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

### 11.15.2 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

### 11.15.3 get-diff-map-iso-level-increment

`get-diff-map-iso-level-increment` [function]  
return difference maps iso-map level increment

### 11.15.4 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

set the difference maps iso-map level increment

### 11.15.5 set-map-sampling-rate-text

`set-map-sampling-rate-text` *text* [function]

Where *text* is a string

sampling rate

find the molecule for which the single map dialog applies and set the contour level and redraw

### 11.15.6 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.

### 11.15.7 get-map-sampling-rate

`get-map-sampling-rate` [function]

return the map sampling rate

### 11.15.8 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.

### 11.15.9 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

### 11.15.10 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

### 11.15.11 set-stop-scroll-diff-map

`set-stop-scroll-diff-map` *i* [function]

Where *i* is an integer number

create a lower limit to the "Fo-Fc-style" map contour level changing

(default 1 on)

### 11.15.12 set-stop-scroll-iso-map

`set-stop-scroll-iso-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)

### 11.15.13 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)

### 11.15.14 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)

### 11.15.15 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

## 11.16 Density Functions

### 11.16.1 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*

### 11.16.2 map-line-width-state

**map-line-width-state** [function]  
 return the width in which density contours are drawn

### 11.16.3 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

### 11.16.4 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, except 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

### 11.16.5 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.

### 11.16.6 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 columns that we want it to have?

### 11.16.7 mtz-file-has-phases-p

`mtz-file-has-phases-p` *mtz\_file\_name* [function]

Where *mtz\_file\_name* is a string

does the mtz file have phases?

### 11.16.8 is-mtz-file-p

`is-mtz-file-p filename` [function]

Where *filename* is a string  
is the given filename an mtz file?

### 11.16.9 cns-file-has-phases-p

`cns-file-has-phases-p cns_file_name` [function]

Where *cns\_file\_name* is a string  
does the given file have cns phases?

### 11.16.10 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.

### 11.16.11 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 expected MTZ columns for Auto-reading MTZ file.

Not every program uses the default reftmac labels ("FWT"/"PHWT") for its MTZ file. Here we can tell coot to expect other labels so that coot can "Auto-open" such MTZ files.

e.g. (`set-auto-read-column-labels "2FOFCWT" "PH2FOFCWT" 0`)

### 11.16.12 set-map-radius

`set-map-radius f` [function]

Where *f* is a number  
set the extent of the box/radius of electron density contours

### 11.16.13 set-density-size

`set-density-size f` [function]

Where *f* is a number  
another (old) way of setting the radius of the map

### 11.16.14 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.

### 11.16.15 get-map-radius

`get-map-radius` [function]  
return the extent of the box/radius of electron density contours

### 11.16.16 set-esoteric-depth-cue

`set-esoteric-depth-cue` *istate* [function]  
Where *istate* is an integer number  
not everyone likes coot's esoteric depth cueing system  
Pass an argument *istate*=1 to turn it off  
(this function is currently disabled).

### 11.16.17 esoteric-depth-cue-state

`esoteric-depth-cue-state` [function]  
native depth cueing system  
return the state of the esoteric depth cueing flag

### 11.16.18 set-swap-difference-map-colours

`set-swap-difference-map-colours` *i* [function]  
Where *i* is an integer number  
not everyone likes coot's default difference map colouring.  
Pass an argument *i*=1 to swap the difference map colouring so that red is positive and green is negative.

### 11.16.19 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)

### 11.16.20 map-is-difference-map

`map-is-difference-map` *imol* [function]  
Where *imol* is an integer number  
map is difference map?

### 11.16.21 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

### 11.16.22 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

### 11.16.23 residue-density-fit-scale-factor

`residue-density-fit-scale-factor` [function]

return the scale factor for the Residue Density fit analysis

### 11.16.24 density-at-point

`density-at-point` *imol\_map* *x* *y* *z* [function]

Where:

- *imol\_map* 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

## 11.17 Parameters from map

### 11.17.1 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).

### 11.17.2 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).

Caller should dispose of returned pointer.

### 11.17.3 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). Caller should dispose of returned pointer.

### 11.17.4 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. Caller should dispose of returned pointer.

### 11.17.5 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.

## 11.18 PDB Functions

### 11.18.1 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*

### 11.18.2 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*

### 11.18.3 quick-save

`quick-save` [function]  
save all modified coordinates molecules to the default names and save the state too.

## 11.19 Info Dialog

### 11.19.1 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).

### 11.19.2 info-dialog-and-text

`info-dialog-and-text txt` [function]  
Where *txt* is a string  
create a dialog with information and print to console  
as `info_dialog` but print to console as well.

## 11.20 Refmac Functions

### 11.20.1 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

### 11.20.2 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

### 11.20.3 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

### 11.20.4 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"

## 11.21 Symmetry Functions

### 11.21.1 set-symmetry-size

`set-symmetry-size` *f* [function]  
Where *f* is a number  
set the size of the displayed symmetry

### 11.21.2 get-show-symmetry

`get-show-symmetry` [function]  
is symmetry master display control on?

### 11.21.3 set-show-symmetry-master

`set-show-symmetry-master` *state* [function]  
Where *state* is an integer number  
set display symmetry, master controller

### 11.21.4 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

### 11.21.5 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

### 11.21.6 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

### 11.21.7 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*

### 11.21.8 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

### 11.21.9 set-symmetry-colour-by-symop

`set-symmetry-colour-by-symop imol state` [function]  
Where:  

- *imol* is an integer number
- *state* is an integer number

set symmetry colour by symop mode

### 11.21.10 set-symmetry-whole-chain

`set-symmetry-whole-chain imol state` [function]  
Where:  

- *imol* is an integer number
- *state* is an integer number

set symmetry colour for the chain

### 11.21.11 set-symmetry-atom-labels-expanded

`set-symmetry-atom-labels-expanded` *state* [function]

Where *state* is an integer number

set use expanded symmetry atom labels

### 11.21.12 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"

### 11.21.13 undo-symmetry-view

`undo-symmetry-view` [function]

Undo symmetry view. Translate back to main molecule from this symmetry position.

### 11.21.14 first-molecule-with-symmetry-displayed

`first-molecule-with-symmetry-displayed` [function]

return the molecule number.

Returns: -1 if there is no molecule with symmetry displayed.

### 11.21.15 save-symmetry-coords

`save-symmetry-coords` *imol filename symop\_no shift\_a shift\_b shift\_c* [function]

*pre\_shift\_to\_origin\_na pre\_shift\_to\_origin\_nb pre\_shift\_to\_origin\_nc*

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 applied (this is how symmetry works in Coot internals).

### 11.21.16 new-molecule-by-symmetry

```
new-molecule-by-symmetry imol name m11 m12 m13 m21 m22 m23 [function]
    m31 m32 m33 tx ty tz pre_shift_to_origin_na pre_shift_to_origin_nb
    pre_shift_to_origin_nc
```

Where:

- *imol* is an integer number
- *name* 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
- *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 coordinates.

Allow a shift of the coordinates to the origin before symmetry expansion is applied.

Pass "" as the name-in and a name will be constructed for you.

Return -1 on failure.

### 11.21.17 new-molecule-by-symmetry-with-atom-selection

```
new-molecule-by-symmetry-with-atom-selection imol name [function]
    mmdb_atom_selection_string 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
```

Where:

- *imol* is an integer number
- *name* is a string
- *mmdb\_atom\_selection\_string* 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
- *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*, but only for atom that match the *mmdb\_atom\_selection\_string*.

The rotation/translation matrix components are given in coordinates.

Allow a shift of the coordinates to the origin before symmetry expansion is applied.

Pass "" as the name-in and a name will be constructed for you.

Return -1 on failure.

### 11.21.18 new-molecule-by-sympop

**new-molecule-by-sympop** *imol sympop\_string pre\_shift\_to\_origin\_na* [function]  
*pre\_shift\_to\_origin\_nb pre\_shift\_to\_origin\_nc*

Where:

- *imol* is an integer number
- *sympop\_string* is a string
- *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*.

### 11.21.19 n-sympops

**n-sympops** *imol* [function]

Where *imol* is an integer number

return the number of symmetry operators for the given molecule

return -1 on no-symmetry for molecule or inappropriate *imol* number

### 11.21.20 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 initiated with a HM symbol or a symm list for clipper. Return the success status of the setting.

### 11.21.21 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.

## 11.22 History Functions

### 11.22.1 print-all-history-in-scheme

`print-all-history-in-scheme` [function]

print the history in scheme format

### 11.22.2 print-all-history-in-python

`print-all-history-in-python` [function]

print the history in python format

### 11.22.3 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.

### 11.22.4 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

set a flag to show the text command equivalent of gui commands in the console as they happen in bold and colours.

*colour\_flag*: pass 1 for on, 0 for off.

*colour\_index* 0 to 7 inclusive for various different colourings.

## 11.23 State Functions

### 11.23.1 save-state

**save-state** [function]  
save the current state to the default filename

### 11.23.2 save-state-file

**save-state-file** *filename* [function]  
Where *filename* is a string  
save the current state to file filename

### 11.23.3 save-state-file-py

**save-state-file-py** *filename* [function]  
Where *filename* is a string  
save the current state to file filename

### 11.23.4 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)

### 11.23.5 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

### 11.23.6 run-state-file

**run-state-file** [function]  
run the state file (reading from default filename)

### 11.23.7 run-state-file-maybe

**run-state-file-maybe** [function]  
run the state file depending on the state variables

## 11.24 The Virtual Trackball

### 11.24.1 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"

### 11.24.2 vt-surface-status

`vt-surface-status` [function]

return the mouse view status mode

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

## 11.25 Clipping Functions

### 11.25.1 set-clipping-back

`set-clipping-back v` [function]

Where *v* is a number

set clipping plane back

### 11.25.2 set-clipping-front

`set-clipping-front v` [function]

Where *v* is a number

set clipping plane front

## 11.26 Unit Cell interface

### 11.26.1 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

### 11.26.2 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

### 11.26.3 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

## 11.27 Colour

### 11.27.1 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

### 11.27.2 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?

### 11.27.3 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?

### 11.27.4 set-colour-by-chain

`set-colour-by-chain imol` [function]

Where *imol* is an integer number

colour molecule number *imol* by chain type

### 11.27.5 set-colour-by-molecule

`set-colour-by-molecule imol` [function]

Where *imol* is an integer number

colour molecule number *imol* by molecule

### 11.27.6 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

## 11.28 Map colour

### 11.28.1 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.

### 11.28.2 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

### 11.28.3 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*.

## 11.29 Anisotropic Atoms Interface

### 11.29.1 get-limit-aniso

`get-limit-aniso` [function]  
get the aniso radius limit

### 11.29.2 get-show-limit-aniso

`get-show-limit-aniso` [function]  
get show the aniso limit

### 11.29.3 get-show-aniso

`get-show-aniso` [function]  
return show-aniso-atoms state

### 11.29.4 set-limit-aniso

`set-limit-aniso state` [function]  
Where *state* is an integer number  
set the aniso atom limit

### 11.29.5 set-show-aniso

`set-show-aniso` *state* [function]  
Where *state* is an integer number  
set show aniso atoms

### 11.29.6 set-aniso-probability

`set-aniso-probability` *f* [function]  
Where *f* is a number  
set aniso probability

### 11.29.7 get-aniso-probability

`get-aniso-probability` [function]  
get aniso probability

## 11.30 Display Functions

### 11.30.1 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

### 11.30.2 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 graphics window position

### 11.30.3 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

store the graphics window position

### 11.30.4 graphics-draw

`graphics-draw` [function]  
draw a frame

**11.30.5 zalman-stereo-mode**

`zalman-stereo-mode` [function]  
 try to turn on Zalman stereo mode

**11.30.6 hardware-stereo-mode**

`hardware-stereo-mode` [function]  
 try to turn on stereo mode

**11.30.7 stereo-mode-state**

`stereo-mode-state` [function]  
 what is the stereo state?  
 Returns: 1 for in hardware stereo, 2 for side by side stereo, else return 0.

**11.30.8 mono-mode**

`mono-mode` [function]  
 try to turn on mono mode

**11.30.9 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 by side stereo mode

**11.30.10 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?

**11.30.11 hardware-stereo-angle-factor-state**

`hardware-stereo-angle-factor-state` [function]  
 return the hardware stereo angle factor

**11.30.12 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

### 11.30.13 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

### 11.30.14 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

### 11.30.15 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

### 11.30.16 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

### 11.30.17 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

### 11.30.18 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

### 11.30.19 set-edit-chi-angles-dialog-position

`set-edit-chi-angles-dialog-position` *x\_pos* *y\_pos* [function]

Where:

- *x\_pos* is an integer number
- *y\_pos* is an integer number

set edit chi angles dialog position

### 11.30.20 set-rotamer-selection-dialog-position

`set-rotamer-selection-dialog-position` *x\_pos* *y\_pos* [function]

Where:

- *x\_pos* is an integer number
- *y\_pos* is an integer number

set rotamer selection dialog position

## 11.31 Smooth Scrolling

### 11.31.1 set-smooth-scroll-flag

`set-smooth-scroll-flag` *v* [function]

Where *v* is an integer number

set smooth scrolling

### 11.31.2 get-smooth-scroll

`get-smooth-scroll` [function]

return the smooth scrolling state

### 11.31.3 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).

### 11.31.4 set-smooth-scroll-limit

`set-smooth-scroll-limit` *lim* [function]

Where *lim* is a number

do not scroll for distances greater this limit

## 11.32 Font Parameters

### 11.32.1 set-font-size

`set-font-size` *i* [function]  
 Where *i* is an integer number  
 set the font size

### 11.32.2 get-font-size

`get-font-size` [function]  
 return the font size  
 Returns: 1 (small) 2 (medium, default) 3 (large)

### 11.32.3 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

### 11.32.4 set-use-stroke-characters

`set-use-stroke-characters` *state* [function]  
 Where *state* is an integer number  
 set use stroke characters

## 11.33 Rotation Centre

### 11.33.1 set-rotation-centre-size

`set-rotation-centre-size` *f* [function]  
 Where *f* is a number  
 set rotation centre marker size

### 11.33.2 recentre-on-read-pdb

`recentre-on-read-pdb` [function]  
 return the recentre-on-pdb state

### 11.33.3 set-recentre-on-read-pdb

`set-recentre-on-read-pdb` *int* [function]  
 Where *int* is a short  
 set the recentre-on-pdb state

### 11.33.4 set-rotation-centre

`set-rotation-centre x y z` [function]

Where:

- *x* is a number
- *y* is a number
- *z* is a number

set the rotation centre

### 11.33.5 go-to-ligand

`go-to-ligand` [function]

centre on the ligand of the "active molecule", if we are already there, centre on the next hetgroup (etc)

### 11.33.6 set-go-to-ligand-n-atoms-limit

`set-go-to-ligand-n-atoms-limit n_atom_min` [function]

Where *n\_atom\_min* is an integer number

go to the ligand that has more than *n\_atom\_min* atoms

## 11.34 Atom Selection Utilities

### 11.34.1 median-temperature-factor

`median-temperature-factor imol` [function]

Where *imol* is an integer number

return the median temperature factor for *imol*

### 11.34.2 average-temperature-factor

`average-temperature-factor imol` [function]

Where *imol* is an integer number

return the average temperature factor for the atoms in *imol*

### 11.34.3 standard-deviation-temperature-factor

`standard-deviation-temperature-factor imol` [function]

Where *imol* is an integer number

return the standard deviation of the atom temperature factors for *imol*

### 11.34.4 clear-pending-picks

`clear-pending-picks` [function]

clear pending picks (stop coot thinking that the user is about to pick an atom).

**11.34.5 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)

**11.34.6 default-new-atoms-b-factor**

`default-new-atoms-b-factor` [function]

return the default temperature factor for newly created atoms

**11.34.7 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)

**11.34.8 get-reset-b-factor-moved-atoms-state**

`get-reset-b-factor-moved-atoms-state` [function]

return the state if temperature factors should be reset for moved atoms

**11.34.9 set-atom-attribute**

`set-atom-attribute` *imol chain\_id resno ins\_code atom\_name alt\_conf* [function]  
*attribute\_name val*

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 numerical attribute to the atom with the given specifier.

Attributes can be "x", "y", "z", "B", "occ" and the attribute *val* is a floating point number

**11.34.10 set-atom-string-attribute**

`set-atom-string-attribute` *imol chain\_id resno ins\_code atom\_name* [function]  
*alt\_conf attribute\_name attribute\_value*

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 attribute to the atom with the given specifier.

Attributes can be "atom-name", "alt-conf", "element" or "segid".

### 11.34.11 set-residue-name

**set-residue-name** *imol chain\_id res\_no ins\_code new\_residue\_name* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string
- *new\_residue\_name* is a string

set lots of atom attributes at once by-passing the rebonding and redrawing of the above 2 functions

set the residue name of the specified residue

## 11.35 Skeletonization Interface

### 11.35.1 skeletonize-map

**skeletonize-map** *imol prune\_flag* [function]

Where:

- *imol* is an integer number
- *prune\_flag* is an integer number

skeletonize molecule number *imol*

the *prune\_flag* should almost always be 0.

NOTE:: The arguments to have been reversed for coot 0.8.3 and later (now the molecule number comes first).

### 11.35.2 unskeletonize-map

**unskeletonize-map** *imol* [function]

Where *imol* is an integer number

undisplay the skeleton on molecule number *imol*

### 11.35.3 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)

### 11.35.4 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

## 11.36 Skeleton Colour

### 11.36.1 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

set the skeleton colour

## 11.37 Read Maps

### 11.37.1 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).

## 11.38 Save Coordinates

### 11.38.1 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.

### 11.38.2 set-save-coordinates-in-original-directory

`set-save-coordinates-in-original-directory` *i* [function]  
Where *i* is an integer number  
set save coordinates in the starting directory

## 11.39 Read Phases File Functions

### 11.39.1 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.

### 11.39.2 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

### 11.39.3 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).

### 11.39.4 read-phs-and-make-map-using-cell-symm

`read-phs-and-make-map-using-cell-symm` *phs\_file\_name* [function]  
*hm\_spacegroup a b c alpha beta gamma*  
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

### 11.39.5 read-phs-and-make-map-with-reso-limits

`read-phs-and-make-map-with-reso-limits` *imol phs\_file\_name* [function]  
*reso\_lim\_low reso\_lim\_high*

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 Angstroms).

## 11.40 Graphics Move

### 11.40.1 undo-last-move

`undo-last-move` [function]  
undo last move

### 11.40.2 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

### 11.40.3 transform-molecule-by

`transform-molecule-by` *imol m11 m12 m13 m21 m22 m23 m31 m32 m33* [function]  
*x y z*

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

#### 11.40.4 transform-zone

`transform-zone` *imol chain\_id resno\_start resno\_end ins\_code m11 m12* [function]  
*m13 m21 m22 m23 m31 m32 m33 x y z*

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
- *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 fragment of molecule number *imol* by the given rotation matrix, then translate by (x,y,z) in Angstroms

## 11.41 Go To Atom Widget Functions

### 11.41.1 post-go-to-atom-window

`post-go-to-atom-window` [function]  
Post the Go To Atom Window.

### 11.41.2 go-to-atom-molecule-number

`go-to-atom-molecule-number` [function]  
the go-to-atom molecule number

### 11.41.3 go-to-atom-chain-id

`go-to-atom-chain-id` [function]  
the go-to-atom chain-id

### 11.41.4 go-to-atom-atom-name

`go-to-atom-atom-name` [function]  
the go-to-atom atom name

### 11.41.5 go-to-atom-residue-number

`go-to-atom-residue-number` [function]  
the go-to-atom residue number

### 11.41.6 go-to-atom-ins-code

`go-to-atom-ins-code` [function]  
the go-to-atom insertion code

### 11.41.7 go-to-atom-alt-conf

`go-to-atom-alt-conf` [function]  
the go-to-atom alt conf

### 11.41.8 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")

Returns: the success status of the go to. 0 for fail, 1 for success.

### 11.41.9 set-go-to-atom-chain-residue-atom-name-full

`set-go-to-atom-chain-residue-atom-name-full` *chain\_id resno* [function]  
*ins\_code atom\_name alt\_conf*

Where:

- *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

set the go to (full) 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")

Returns: the success status of the go to. 0 for fail, 1 for success.

### 11.41.10 set-go-to-atom-chain-residue-atom-name-no-redraw

`set-go-to-atom-chain-residue-atom-name-no-redraw` *t1 iresno t3* [function]  
*make\_the\_move\_flag*

Where:

- *t1* is a string
- *iresno* is an integer number
- *t3* is a string
- *make\_the\_move\_flag* is an integer number

set go to atom but don't redraw

### 11.41.11 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.

### 11.41.12 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?

### 11.41.13 full-atom-spec-to-atom-index

`full-atom-spec-to-atom-index` *imol chain resno inscode atom\_name* [function]  
*altloc*

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?

### 11.41.14 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

### 11.41.15 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.

### 11.41.16 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 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.

20090914-PE `set-go-to-atom-molecule` can be used in a script and it should change the `go-to-atom-molecule` in the Go To Atom dialog (if it is being displayed). This does mean, of course that using the ramachandran plot to centre on atoms will change the Go To Atom dialog. Maybe that is surprising (maybe not).

## 11.42 Map and Molecule Control

### 11.42.1 post-display-control-window

`post-display-control-window` [function]  
 display the Display Control window

### 11.42.2 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

### 11.42.3 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

### 11.42.4 set-display-only-model-mol

`set-display-only-model-mol` *imol* [function]

Where *imol* is an integer number

from all the model molecules, display only *imol*

This stops flashing/delayed animations with many molecules

### 11.42.5 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/inactive (clickable), 0 for off, 1 for on

### 11.42.6 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

### 11.42.7 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

### 11.42.8 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

### 11.42.9 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

### 11.42.10 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.

### 11.42.11 show-spacegroup

`show-spacegroup` *imol* [function]

Where *imol* is an integer number

return the spacegroup of molecule number *imol* . Deprecated.

Returns: "No Spacegroup" when the spacegroup of a molecule has not been set.

## 11.43 Align and Mutate

### 11.43.1 align-and-mutate

`align-and-mutate` *imol chain\_id fasta\_maybe renumber\_residues\_flag* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *fasta\_maybe* is a string
- *renumber\_residues\_flag* is an integer number

align and mutate the given chain to the given sequence

### 11.43.2 set-alignment-gap-and-space-penalty

`set-alignment-gap-and-space-penalty` *wgap wspace* [function]

Where:

- *wgap* is a number
- *wspace* is a number

set the penalty for affine gap and space when aligning, defaults -3.0 and -0.4

## 11.44 Renumber Residue Range

### 11.44.1 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

### 11.44.2 change-residue-number

`change-residue-number` *imol chain\_id current\_resno current\_inscore* [function]  
*new\_resno new\_inscore*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *current\_resno* is an integer number
- *current\_inscore* is a string
- *new\_resno* is an integer number
- *new\_inscore* is a string

change chain id, residue number or insertion code for given residue

## 11.45 Scripting Interface

### 11.45.1 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

### 11.45.2 post-scripting-window

`post-scripting-window` [function]

do nothing - compatibility function

### 11.45.3 post-scheme-scripting-window

`post-scheme-scripting-window` [function]

pop-up a scripting window for scheming

### 11.45.4 post-python-scripting-window

`post-python-scripting-window` [function]  
pop-up a scripting window for pythoning

## 11.46 Monomer

### 11.46.1 get-coords-for-accession-code

`get-coords-for-accession-code` *code* [function]  
Where *code* is a string  
if possible, read in the new coords getting coords via web.  
(no return value because get-url-str does not return one).

### 11.46.2 run-script

`run-script` *filename* [function]  
Where *filename* is a string  
run script file

### 11.46.3 run-guile-script

`run-guile-script` *filename* [function]  
Where *filename* is a string  
guile run script file

### 11.46.4 run-python-script

`run-python-script` *filename* [function]  
Where *filename* is a string  
run python script file

### 11.46.5 import-python-module

`import-python-module` *module\_name use\_namespace* [function]  
Where:  

- *module\_name* is a string
- *use\_namespace* is an integer number

import python module

## 11.47 Regularization and Refinement

### 11.47.1 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 editing the `mon_lib_list.cif` file.

### 11.47.2 remove-planar-peptide-restraints

`remove-planar-peptide-restraints` [function]  
remove restraints on peptides to make them planar.

### 11.47.3 set-use-trans-peptide-restraints

`set-use-trans-peptide-restraints` *on\_off\_state* [function]  
Where *on\_off\_state* is an integer number  
add a restraint on peptides to keep trans peptides trans  
i.e. omega in trans-peptides is restraints to 180 degrees.

### 11.47.4 add-omega-torsion-restraints

`add-omega-torsion-restraints` [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).

### 11.47.5 remove-omega-torsion-restraints

`remove-omega-torsion-restraints` [function]  
remove omega restraints on CIS and TRANS linked residues.

### 11.47.6 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

### 11.47.7 refinement-immediate-replacement-state

`refinement-immediate-replacement-state` [function]  
query the state of the immediate replacement mode

### 11.47.8 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.

### 11.47.9 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`.

### 11.47.10 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.

### 11.47.11 refine-with-torsion-restraints-state

`refine-with-torsion-restraints-state` [function]

return the state of above

### 11.47.12 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).

### 11.47.13 matrix-state

`matrix-state` [function]

return the relative weight of the geometric terms to the map terms.

### 11.47.14 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 allow 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

### 11.47.15 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

### 11.47.16 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

### 11.47.17 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

### 11.47.18 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

### 11.47.19 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

Returns: a status, whether the regularisation was done or not. 0 for no, 1 for yes.

### 11.47.20 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: 80.

### 11.47.21 dragged-refinement-steps-per-frame

`dragged-refinement-steps-per-frame` [function]

return the number of steps per frame in dragged refinement

### 11.47.22 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.

### 11.47.23 refinement-refine-per-frame-state

`refinement-refine-per-frame-state` [function]

query the state of the above option

### 11.47.24 set-refinement-drag-elasticity

`set-refinement-drag-elasticity` *e* [function]

Where *e* is a number

the elasticity of the dragged atom in refinement mode.

Default 0.33

Bigger numbers mean bigger movement of the other atoms.

### 11.47.25 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  
name consistent with  
!?

### 11.47.26 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.

### 11.47.27 check-chiral-volumes

`check-chiral-volumes` *imol* [function]  
Where *imol* is an integer number  
query the state of the above option

### 11.47.28 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.

### 11.47.29 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

### 11.47.30 secondary-structure-restraints-type

`secondary-structure-restraints-type` [function]  
return the secondary structure restraints type

### 11.47.31 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.

### 11.47.32 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

### 11.47.33 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.

### 11.47.34 add-extra-bond-restraint

`add-extra-bond-restraint` *imol chain\_id\_1 res\_no\_1 ins\_code\_1* [function]  
*atom\_name\_1 alt\_conf\_1 chain\_id\_2 res\_no\_2 ins\_code\_2 atom\_name\_2*  
*alt\_conf\_2 bond\_dist esd*

Where:

- *imol* is an integer number
- *chain\_id\_1* is a string
- *res\_no\_1* is an integer number
- *ins\_code\_1* is a string
- *atom\_name\_1* is a string
- *alt\_conf\_1* is a string
- *chain\_id\_2* is a string
- *res\_no\_2* is an integer number
- *ins\_code\_2* is a string
- *atom\_name\_2* is a string
- *alt\_conf\_2* is a string
- *bond\_dist* is a number
- *esd* is a number

add a user-define bond restraint

this extra restraint is used when the given atoms are selected in refinement or regularization.

Returns: the index of the new restraint.

Returns: -1 when the atoms were not found and no extra bond restraint was stored.

### 11.47.35 delete-all-extra-restraints

`delete-all-extra-restraints` *imol* [function]

Where *imol* is an integer number

clear out all the extra/user-defined restraints for molecule number *imol*

### 11.47.36 delete-extra-restraints-for-residue

`delete-extra-restraints-for-residue` *imol chain\_id res\_no ins\_code* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string

clear out all the extra/user-defined restraints for this residue in molecule number *imol*

### 11.47.37 set-extra-restraints-prosmart-sigma-limits

`set-extra-restraints-prosmart-sigma-limits` *imol limit\_high limit\_low* [function]

Where:

- *imol* is an integer number
- *limit\_high* is a number
- *limit\_low* is a number

often we don't want to see all prosmart restraints, just the (big) violations

### 11.47.38 write-interpolated-extra-restraints

`write-interpolated-extra-restraints` *imol\_1 imol\_2 n\_steps file\_name\_stub* [function]

Where:

- *imol\_1* is an integer number
- *imol\_2* is an integer number
- *n\_steps* is an integer number
- *file\_name\_stub* is a string

proSMART interpolated restraints for model morphing

### 11.47.39 write-interpolated-models-and-extra-restraints

`write-interpolated-models-and-extra-restraints` *imol\_1 imol\_2 n\_steps file\_name\_stub interpolation\_mode* [function]

Where:

- *imol\_1* is an integer number
- *imol\_2* is an integer number

- *n\_steps* is an integer number
- *file\_name\_stub* is a string
- *interpolation\_mode* is an integer number

proSMART interpolated restraints for model morphing and write interpolated model interpolation\_mode is currently dummy - in due course I will add torion angle interpolation.

#### 11.47.40 set-use-only-extra-torsion-restraints-for-torsions

`set-use-only-extra-torsion-restraints-for-torsions state` [function]  
 Where *state* is an integer number  
 set use only extra torsion restraints for torsions

#### 11.47.41 use-only-extra-torsion-restraints-for-torsions-state

`use-only-extra-torsion-restraints-for-torsions-state` [function]  
 return only-use-extra-torsion-restraints-for-torsions state

#### 11.47.42 show-restraints-editor

`show-restraints-editor monomer_type` [function]  
 Where *monomer\_type* is a string  
 show restraints editor

#### 11.47.43 show-restraints-editor-by-index

`show-restraints-editor-by-index menu_item_index` [function]  
 Where *menu\_item\_index* is an integer number  
 show restraints editor using residue type index

#### 11.47.44 write-restraints-cif-dictionary

`write-restraints-cif-dictionary monomer_type file_name` [function]  
 Where:  

- *monomer\_type* is a string
- *file\_name* is a string

 write cif restraints for monomer

### 11.48 Simplex Refinement Interface

#### 11.48.1 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

### 11.48.2 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

## 11.49 Nomenclature Errors

### 11.49.1 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 residues altered.

### 11.49.2 set-nomenclature-errors-on-read

`set-nomenclature-errors-on-read` *mode* [function]

Where *mode* is a string

set way nomenclature errors should be handled on reading coordinates.

mode should be "auto-correct", "ignore", "prompt". The default is "prompt"

## 11.50 Atom Info Interface

### 11.50.1 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 to the terminal the Atom Info for the give atom specs

## 11.51 Residue Info

### 11.51.1 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

show residue info dialog for given residue

## 11.52 Residue Environment Functions

### 11.52.1 set-show-environment-distances

`set-show-environment-distances state` [function]

Where *state* is an integer number

show environment distances. If state is 0, distances are turned off, otherwise distances are turned on.

### 11.52.2 set-show-environment-distances-bumps

`set-show-environment-distances-bumps state` [function]

Where *state* is an integer number

show bumps environment distances. If state is 0, bump distances are turned off, otherwise bump distances are turned on.

### 11.52.3 set-show-environment-distances-h-bonds

`set-show-environment-distances-h-bonds state` [function]

Where *state* is an integer number

show H-bond environment distances. If state is 0, bump distances are turned off, otherwise H-bond distances are turned on.

### 11.52.4 show-environment-distances-state

`show-environment-distances-state` [function]

show the state of display of the environment distances

### 11.52.5 set-environment-distances-distance-limits

`set-environment-distances-distance-limits` *min\_dist* *max\_dist* [function]

Where:

- *min\_dist* is a number
- *max\_dist* is a number

min and max distances for the environment distances

## 11.53 Pointer Functions

### 11.53.1 set-show-pointer-distances

`set-show-pointer-distances` *istate* [function]

Where *istate* is an integer number

turn on (or off) the pointer distance by passing 1 (or 0).

### 11.53.2 show-pointer-distances-state

`show-pointer-distances-state` [function]

show the state of display of the pointer distances

## 11.54 Zoom Functions

### 11.54.1 scale-zoom

`scale-zoom` *f* [function]

Where *f* is a number

scale the view by *f*

external (scripting) interface (with redraw)

### 11.54.2 zoom-factor

`zoom-factor` [function]

return the current zoom factor

### 11.54.3 set-smooth-scroll-do-zoom

`set-smooth-scroll-do-zoom` *i* [function]

Where *i* is an integer number

set smooth scroll with zoom

### 11.54.4 smooth-scroll-do-zoom

`smooth-scroll-do-zoom` [function]

return the state of the above system

## 11.55 CNS Data Functions

### 11.55.1 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)

### 11.55.2 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

## 11.56 mmCIF Functions

### 11.56.1 open-cif-dictionary-file-selector-dialog

`open-cif-dictionary-file-selector-dialog` [function]  
open the cif dictionary file selector dialog

## 11.57 SHELXL Functions

### 11.57.1 read-shelx-ins-file

`read-shelx-ins-file filename recentre_flag` [function]

Where:

- *filename* is a string
- *recentre\_flag* is an integer number

read a SHELXL .ins file

### 11.57.2 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*

## 11.58 Validation Functions

### 11.58.1 difference-map-peaks

`difference-map-peaks` *imol imol\_coords level max\_closeness 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
- *max\_closeness* 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

peaks within *max\_closeness* (2.0 Å typically) of a larger peak are not listed.

### 11.58.2 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 outliers, comparing the O and N temperature factors difference to the distribution of temperature factors from the other atoms.

## 11.59 Ramachandran Plot Functions

### 11.59.1 do-ramachandran-plot

`do-ramachandran-plot` *imol* [function]

Where *imol* is an integer number

Ramachandran plot for molecule number *imol*.

### 11.59.2 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.

### 11.59.3 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)

### 11.59.4 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

### 11.59.5 ramachandran-plot-differences

`ramachandran-plot-differences` *imol1* *imol2* [function]

Where:

- *imol1* is an integer number
- *imol2* is an integer number

2 molecule ramachandran plot (NCS differences) a.k.a. A Kleywegt Plot.

### 11.59.6 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

A chain-specific Kleywegt Plot.

## 11.60 Sequence View Interface

### 11.60.1 do-sequence-view

`do-sequence-view` *imol* [function]

Where *imol* is an integer number

display the sequence view dialog for molecule number imol

## 11.61 Atom Labelling

### 11.61.1 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

### 11.61.2 brief-atom-labels-state

`brief-atom-labels-state` [function]

the brief atom label state

## 11.62 Screen Rotation

### 11.62.1 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

### 11.62.2 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

### 11.62.3 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

### 11.62.4 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.

## 11.63 Screen Translation

### 11.63.1 translate-scene-x

`translate-scene-x` *nsteps* [function]

Where *nsteps* is an integer number

translate rotation centre relative to screen axes for nsteps

### 11.63.2 translate-scene-y

`translate-scene-y` *nsteps* [function]

Where *nsteps* is an integer number

translate rotation centre relative to screen axes for nsteps

### 11.63.3 translate-scene-z

`translate-scene-z` *nsteps* [function]

Where *nsteps* is an integer number

translate rotation centre relative to screen axes for nsteps

## 11.64 Views Interface

### 11.64.1 add-view-here

`add-view-here` *view\_name* [function]

Where *view\_name* is a string

return the view number

### 11.64.2 add-view-raw

`add-view-raw` *rcx rcy rcz quat1 quat2 quat3 quat4 zoom view\_name* [function]

Where:

- *rcx* is a number
- *rcy* is a number

- *rcz* is a number
- *quat1* is a number
- *quat2* is a number
- *quat3* is a number
- *quat4* is a number
- *zoom* is a number
- *view\_name* is a string

return the view number

### 11.64.3 remove-named-view

`remove-named-view` *view\_name* [function]  
Where *view\_name* is a string  
the view with the given name

### 11.64.4 remove-view

`remove-view` *view\_number* [function]  
Where *view\_number* is an integer number  
the given view number

### 11.64.5 add-view-description

`add-view-description` *view\_number* *description* [function]  
Where:

- *view\_number* is an integer number
- *description* is a string

Add a view description/annotation to the give view number.

### 11.64.6 add-action-view

`add-action-view` *view\_name* *action\_function* [function]  
Where:

- *view\_name* is a string
- *action\_function* is a string

add a view (not add to an existing view) that does something (e.g. displays or undisplays a molecule) rather than move the graphics.

Returns: the view number for this (new) view.

### 11.64.7 insert-action-view-after-view

`insert-action-view-after-view` *view\_number* *view\_name* *action\_function* [function]

Where:

- *view\_number* is an integer number

- *view\_name* is a string
- *action\_function* is a string

add an action view after the view of the given view number

Returns: the view number for this (new) view.

### 11.64.8 save-views

`save-views view_file_name` [function]

Where *view\_file\_name* is a string

save views to *view\_file\_name*

### 11.64.9 clear-all-views

`clear-all-views` [function]

Clear the view list.

## 11.65 Background Colour

### 11.65.1 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

### 11.65.2 redraw-background

`redraw-background` [function]

re draw the background colour when switching between mono and stereo

### 11.65.3 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.

## 11.66 Ligand Fitting Functions

### 11.66.1 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

### 11.66.2 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

### 11.66.3 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.

### 11.66.4 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.

### 11.66.5 set-find-ligand-multi-solutions-per-cluster

`set-find-ligand-multi-solutions-per-cluster lim_1 lim_2` [function]

Where:

- *lim\_1* is a number
- *lim\_2* is a number

allow multiple ligand solutions per cluster.

The first limit is the fraction of the top scored positions that go on to correlation scoring (closer to 1 means less and faster - default 0.7).

The second limit is the fraction of the top correlation score that is considered interesting. Limits the number of solutions displayed to user. Default 0.9.

There is currently no chi-angle set redundancy filtering - I suspect that there should be.

Nino-mode.

### 11.66.6 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.

### 11.66.7 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

### 11.66.8 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

### 11.66.9 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

### 11.66.10 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

### 11.66.11 set-find-ligand-here-cluster

`set-find-ligand-here-cluster state` [function]  
Where *state* is an integer number  
Allow the user a scripting means to find ligand at the rotation centre.

### 11.66.12 ligand-expert

`ligand-expert` [function]  
this sets the flag to have expert option ligand entries in the Ligand Searching dialog

### 11.66.13 do-find-ligands-dialog

`do-find-ligands-dialog` [function]  
display the find ligands dialog  
if maps, coords and ligands are available, that is.

### 11.66.14 match-ligand-atom-names

`match-ligand-atom-names imol_ligand chain_id_ligand resno_ligand` [function]  
`ins_code_ligand imol_reference chain_id_reference resno_reference`  
`ins_code_reference`

Where:

- *imol\_ligand* is an integer number
- *chain\_id\_ligand* is a string
- *resno\_ligand* is an integer number
- *ins\_code\_ligand* is a string
- *imol\_reference* is an integer number
- *chain\_id\_reference* is a string

- *resno\_reference* is an integer number
- *ins\_code\_reference* is a string

Overlap residue with "template"-based matching.

Overlap the first residue in *imol\_ligand* onto the residue specified by the reference parameters. Use graph matching, not atom names.

Match ligand atom names

By using graph matching, make the names of the atoms of the given ligand/residue match those of the reference residue/ligand as closely as possible - where there would be an atom name clash, invent a new atom name.

Returns: success status, False = failed to find residue in either *imol\_ligand* or *imo\_ref*. If success, return the RT operator.

### 11.66.15 match-ligand-atom-names-to-comp-id

`match-ligand-atom-names-to-comp-id` *imol\_ligand chain\_id\_ligand* [function]  
*resno\_ligand ins\_code\_ligand comp\_id\_ref*

Where:

- *imol\_ligand* is an integer number
- *chain\_id\_ligand* is a string
- *resno\_ligand* is an integer number
- *ins\_code\_ligand* is a string
- *comp\_id\_ref* is a string

Match ligand atom names to a reference ligand type (*comp-id*)

By using graph matching, make the names of the atoms of the given ligand/residue match those of the reference ligand from the geometry store as closely as possible. Where there would be an atom name clash, invent a new atom name.

This doesn't create a new dictionary for the selected ligand - and that's a big problem (see `match_residue_and_dictionary`).

### 11.66.16 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

flip the ligand (usually active residue) around its eigen vectors to the next flip number. Immediate replacement (like flip peptide).

## 11.67 Water Fitting Functions

### 11.67.1 wrapped-create-find-waters-dialog

`wrapped-create-find-waters-dialog` [function]  
create a dialog for water fitting

### 11.67.2 renumber-waters

`renumber-waters` *imol* [function]  
Where *imol* is an integer number  
Renumber the waters of molecule number *imol* with consecutive numbering.

### 11.67.3 execute-find-waters-real

`execute-find-waters-real` *imol\_for\_map imol\_for\_protein* [function]  
*new\_waters\_mol\_flag rmsd\_cut\_off*

Where:

- *imol\_for\_map* is an integer number
- *imol\_for\_protein* is an integer number
- *new\_waters\_mol\_flag* is an integer number
- *rmsd\_cut\_off* is a number

find waters

### 11.67.4 move-waters-to-around-protein

`move-waters-to-around-protein` *imol* [function]  
Where *imol* is an integer number  
move waters of molecule number *imol* so that they are around the protein.  
Returns: the number of moved waters.

### 11.67.5 move-hetgroups-to-around-protein

`move-hetgroups-to-around-protein` *imol* [function]  
Where *imol* is an integer number  
move all hetgroups (including waters) of molecule number *imol* so that they are around the protein.

### 11.67.6 max-water-distance

`max-water-distance` *imol* [function]  
Where *imol* is an integer number  
return the maximum minimum distance of any water atom to any protein atom - used in validation of `move_waters_to_around_protein()` function.

### 11.67.7 set-water-check-spherical-variance-limit

`set-water-check-spherical-variance-limit` *f* [function]

Where *f* is a number

set the limit of interesting variance, above which waters are listed (otherwise ignored)

default 0.12.

### 11.67.8 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

### 11.67.9 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

### 11.67.10 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

### 11.67.11 split-water

`split-water` *imol chain\_id res\_no ins\_code* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string

split the given water and fit to map.

If refinement map is not defined, don't do anything.

If there is more than one atom in the specified residue, don't do anything.

If the given atom does not have an alt conf of "", don't do anything.

## 11.68 Bond Representation

### 11.68.1 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`)

### 11.68.2 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

### 11.68.3 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

### 11.68.4 set-bond-colour-rotation-for-molecule

`set-bond-colour-rotation-for-molecule imol f` [function]  
Where:

- *imol* is an integer number
- *f* is a number

set bond colour for molecule

### 11.68.5 get-bond-colour-rotation-for-molecule

`get-bond-colour-rotation-for-molecule imol` [function]  
Where *imol* is an integer number  
get the bond colour for molecule.  
Return -1 on err (bad molecule number)

### 11.68.6 set-default-representation-type

`set-default-representation-type type` [function]  
Where *type* is an integer number  
set the default representation type (default 1).

### 11.68.7 get-default-bond-thickness

`get-default-bond-thickness` [function]  
get the default thickness for bonds

### 11.68.8 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.

### 11.68.9 set-draw-cis-peptide-markups

`set-draw-cis-peptide-markups` *status* [function]

Where *status* is an integer number

set status of drawing cis-peptide markups

default status is 1.

### 11.68.10 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

### 11.68.11 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*.

### 11.68.12 set-draw-stick-mode-atoms

`set-draw-stick-mode-atoms` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

draw little coloured balls on atoms

turn off with *state* = 0

turn on with *state* = 1

### 11.68.13 graphics-to-ca-representation

`graphics-to-ca-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* as CAs

**11.68.14 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

**11.68.15 graphics-to-ca-plus-ligands-and-sidechains-representation**

`graphics-to-ca-plus-ligands-and-sidechains-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* as CA + ligands + sidechains

**11.68.16 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

**11.68.17 graphics-to-bonds-representation**

`graphics-to-bonds-representation` *mol* [function]

Where *mol* is an integer number

draw molecule number *imol* with normal bonds

**11.68.18 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

**11.68.19 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

**11.68.20 graphics-to-rainbow-representation**

`graphics-to-rainbow-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* in Jones' Rainbow

**11.68.21 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

### 11.68.22 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

### 11.68.23 graphics-to-occupancy-representation

`graphics-to-occupancy-representation imol` [function]

Where *imol* is an integer number

draw molecule number *imol* coloured by occupancy

### 11.68.24 graphics-to-user-defined-atom-colours-representation

`graphics-to-user-defined-atom-colours-representation imol` [function]

Where *imol* is an integer number

draw molecule number *imol* in CA+Ligands mode coloured by user-defined atom colours

### 11.68.25 graphics-to-user-defined-atom-colours-all-atoms-representation

`graphics-to-user-defined-atom-colours-all-atoms-representation` [function]

*imol*

Where *imol* is an integer number

draw molecule number *imol* all atoms coloured by user-defined atom colours

### 11.68.26 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*

### 11.68.27 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

### 11.68.28 change-model-molecule-representation-mode

`change-model-molecule-representation-mode up_or_down` [function]

Where *up\_or\_down* is an integer number

change the representation of the model molecule closest to the centre of the screen

### 11.68.29 set-use-grey-carbons-for-molecule

`set-use-grey-carbons-for-molecule` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

make the carbon atoms for molecule *imol* be grey

### 11.68.30 set-grey-carbon-colour

`set-grey-carbon-colour` *imol r g b* [function]

Where:

- *imol* is an integer number
- *r* is a number
- *g* is a number
- *b* is a number

set the colour for the carbon atoms

can be not grey if you desire, *r*, *g*, *b* in the range 0 to 1.

### 11.68.31 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`)

### 11.68.32 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*

### 11.68.33 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

return the index of the additional representation. Return -1 on error

### 11.68.34 additional-representation-by-attributes

**additional-representation-by-attributes** *imol chain\_id* [function]  
*resno\_start resno\_end ins\_code representation\_type bonds\_box\_type bond\_width*  
*draw\_hydrogens\_flag*

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

return the index of the additional representation.

Returns: -1 on error.

## 11.69 Dots Representation

### 11.69.1 dots

**dots** *imol atom\_selection\_str dots\_object\_name dot\_density* [function]  
*sphere\_size\_scale*

Where:

- *imol* is an integer number
- *atom\_selection\_str* is a string
- *dots\_object\_name* 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)

### 11.69.2 set-dots-colour

`set-dots-colour` *imol r g b* [function]

Where:

- *imol* is an integer number
- *r* is a number
- *g* is a number
- *b* is a number

set the colour of the surface dots of the *imol*-th molecule to be the given single colour

*r,g,b* are values between 0.0 and 1.0

### 11.69.3 unset-dots-colour

`unset-dots-colour` *imol* [function]

Where *imol* is an integer number

no longer set the dots of molecule *imol* to a single colour

i.e. go back to element-based colours.

### 11.69.4 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*

### 11.69.5 clear-dots-by-name

`clear-dots-by-name` *imol dots\_object\_name* [function]

Where:

- *imol* is an integer number
- *dots\_object\_name* is a string

clear the first dots object for *imol* with given name

### 11.69.6 n-dots-sets

`n-dots-sets` *imol* [function]

Where *imol* is an integer number

return the number of dots sets for molecule number *imol*

## 11.70 Pep-flip Interface

### 11.70.1 pepflip

`pepflip` *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

pepflip the given residue

## 11.71 Rigid Body Refinement Interface

### 11.71.1 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()`

### 11.71.2 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

## 11.72 Add Terminal Residue Functions

### 11.72.1 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

### 11.72.2 add-terminal-residue

`add-terminal-residue` *imol chain\_id residue\_number residue\_type* [function]  
*immediate\_add*

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.

residue type can be "auto" and *immediate\_add* is recommended to be 1.

return 0 on failure, 1 on success

### 11.72.3 add-terminal-residue-using-phi-psi

`add-terminal-residue-using-phi-psi` *imol chain\_id res\_no* [function]  
*residue\_type phi psi*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *residue\_type* is a string
- *phi* is a number
- *psi* is a number

Add a terminal residue using given phi and psi angles.

### 11.72.4 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.

### 11.72.5 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.

### 11.72.6 add-terminal-residue-do-post-refine-state

`add-terminal-residue-do-post-refine-state` [function]

what is the value of the previous flag?

## 11.73 Delete Residues

### 11.73.1 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

### 11.73.2 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

### 11.73.3 delete-residue-with-full-spec

`delete-residue-with-full-spec` *imol imodel chain\_id resno inscode altloc* [function]

Where:

- *imol* is an integer number
- *imodel* 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

### 11.73.4 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

### 11.73.5 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

### 11.73.6 delete-residue-sidechain

`delete-residue-sidechain` *imol chain\_id resno ins\_code* [function]

*do\_delete\_dialog*

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

### 11.73.7 delete-hydrogens

`delete-hydrogens` *imol* [function]

Where *imol* is an integer number

delete all hydrogens in molecule,

Returns: number of hydrogens deleted.

## 11.74 Mainchain Building Functions

### 11.74.1 db-mainchain

`db-mainchain` *imol chain\_id iresno\_start iresno\_end direction* [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* is a string

CA -> mainchain conversion.

direction is either "forwards" or "backwards"

return the new molecule number

## 11.75 Rotamer Functions

### 11.75.1 set-rotamer-search-mode

`set-rotamer-search-mode mode` [function]

Where *mode* is an integer number

set the mode of rotamer search, options are (ROTAMERSEARCHAUTOMATIC), (ROTAMERSEARCHLOWRES) (aka. "backrub rotamers"), (ROTAMERSEARCHHIGHRES) (with rigid body fitting)

### 11.75.2 set-rotamer-lowest-probability

`set-rotamer-lowest-probability f` [function]

Where *f* is a number

For Dunbrack rotamers, set the lowest probability to be considered. Set as a percentage i.e. 1.00 is quite low. For Richardson Rotamers, this has no effect.

### 11.75.3 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

### 11.75.4 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).

### 11.75.5 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.

### 11.75.6 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.

### 11.75.7 set-residue-to-rotamer-number

`set-residue-to-rotamer-number` *imol chain\_id resno ins\_code alt\_conf* [function]  
*rotamer\_number*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno* is an integer number
- *ins\_code* is a string
- *alt\_conf* is a string
- *rotamer\_number* is an integer number

set the residue specified to the rotamer number specified.

### 11.75.8 set-residue-to-rotamer-name

`set-residue-to-rotamer-name` *imol chain\_id resno ins\_code alt\_conf* [function]  
*rotamer\_name*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno* is an integer number
- *ins\_code* is a string
- *alt\_conf* is a string
- *rotamer\_name* is a string

set the residue specified to the rotamer name specified.

(rotamer names are the Richardson rotamer names.)

return value is 0 if atoms were not moved (e.g. because rotamer-name was not know)

### 11.75.9 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.

## 11.76 180 Flip Side chain

### 11.76.1 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

## 11.77 Mutate Functions

### 11.77.1 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

### 11.77.2 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.

### 11.77.3 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

mutate a base. return success status, 1 for a good mutate.

### 11.77.4 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.

### 11.77.5 mutate-auto-fit-do-post-refine-state

`mutate-auto-fit-do-post-refine-state` [function]

what is the value of the previous flag?

### 11.77.6 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.

### 11.77.7 rotamer-auto-fit-do-post-refine-state

`rotamer-auto-fit-do-post-refine-state` [function]

what is the value of the previous flag?

### 11.77.8 mutate-single-residue-by-serial-number

`mutate-single-residue-by-serial-number` *ires\_ser chain\_id imol* [function]

*target\_res\_type*

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 single 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 mutate 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

### 11.77.9 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)

## 11.78 Pointer Atom Functions

### 11.78.1 create-pointer-atom-molecule-maybe

`create-pointer-atom-molecule-maybe` [function]

Return the current pointer atom molecule, create a pointer atom molecule if necessary (i.e. when the user has not set it).

### 11.78.2 pointer-atom-molecule

`pointer-atom-molecule` [function]

Return the current pointer atom molecule.

## 11.79 Baton Build Interface Functions

### 11.79.1 set-baton-mode

`set-baton-mode` *i* [function]

Where *i* is an integer number

toggle so that mouse movement moves the baton not rotates the view.

### 11.79.2 try-set-draw-baton

`try-set-draw-baton` *i* [function]

Where *i* is an integer number

draw the baton or not

### 11.79.3 accept-baton-position

`accept-baton-position` [function]

accept the baton tip position - a prime candidate for a key binding

### 11.79.4 baton-tip-try-another

`baton-tip-try-another` [function]  
move the baton tip position - another prime candidate for a key binding

### 11.79.5 baton-tip-previous

`baton-tip-previous` [function]  
move the baton tip to the previous position

### 11.79.6 shorten-baton

`shorten-baton` [function]  
shorten the baton length

### 11.79.7 lengthen-baton

`lengthen-baton` [function]  
lengthen the baton

### 11.79.8 baton-build-delete-last-residue

`baton-build-delete-last-residue` [function]  
delete the most recently build CA position

### 11.79.9 set-baton-build-params

`set-baton-build-params` *istart\_resno chain\_id direction* [function]  
Where:

- *istart\_resno* is an integer number
- *chain\_id* is a string
- *direction* is a string

set the parameters for the start of a new baton-built fragment. *direction* can either be "forwards" or "backwards"

## 11.80 Crosshairs Interface

### 11.80.1 set-draw-crosshairs

`set-draw-crosshairs` *i* [function]  
Where *i* is an integer number  
draw the distance crosshairs, 0 for off, 1 for on.

## 11.81 Edit Chi Angles

### 11.81.1 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).

### 11.81.2 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 find the residue, return a value of 1 if it worked.

### 11.81.3 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...

## 11.82 Masks

### 11.82.1 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.

### 11.82.2 set-map-mask-atom-radius

`set-map-mask-atom-radius` *rad* [function]

Where *rad* is a number

set the atom radius for map masking

### 11.82.3 map-mask-atom-radius

`map-mask-atom-radius` [function]  
get the atom radius for map masking

## 11.83 check Waters Interface

### 11.83.1 delete-checked-waters-baddies

`delete-checked-waters-baddies` *imol b\_factor\_lim map\_sigma\_lim* [function]  
*min\_dist max\_dist part\_occ\_contact\_flag zero\_occ\_flag*  
*logical\_operator\_and\_or\_flag*

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.

## 11.84 Trim

### 11.84.1 trim-molecule-by-map

`trim-molecule-by-map` *imol\_coords imol\_map map\_level* [function]  
*delete\_or\_zero\_occ\_flag*

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

cut off (delete or give zero occupancy) atoms in the given molecule if they are below the given map (absolute) level.

## 11.85 External Ray-Tracing

### 11.85.1 raster3d

`raster3d` *rd3\_filename* [function]

Where *rd3\_filename* is a string

create a r3d file for the current view

### 11.85.2 set-raster3d-bond-thickness

`set-raster3d-bond-thickness f` [function]  
Where *f* is a number  
set the bond thickness for the Raster3D representation

### 11.85.3 set-raster3d-atom-radius

`set-raster3d-atom-radius f` [function]  
Where *f* is a number  
set the atom radius for the Raster3D representation

### 11.85.4 set-raster3d-density-thickness

`set-raster3d-density-thickness f` [function]  
Where *f* is a number  
set the density line thickness for the Raster3D representation

### 11.85.5 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

### 11.85.6 set-raster3d-bone-thickness

`set-raster3d-bone-thickness f` [function]  
Where *f* is a number  
set the bone (skeleton) thickness for the Raster3D representation

### 11.85.7 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

### 11.85.8 set-raster3d-water-sphere

`set-raster3d-water-sphere istate` [function]  
Where *istate* is an integer number  
set the flag to show waters as spheres for the Raster3D representation. 1 show as spheres, 0 the usual stars.

### 11.85.9 raster-screen-shot

`raster-screen-shot` [function]  
run raster3d and display the resulting image.

## 11.86 Superposition (SSM)

### 11.86.1 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 value of *move\_imol2\_flag* (1 for move 0 for copy).

### 11.86.2 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 value of *move\_imol2\_flag* (1 for move 0 for copy).

### 11.86.3 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 value of *move\_imol2\_flag* (1 for move 0 for copy).

Returns: the index of the superposed molecule - which could either be a new molecule (if *move\_imol2\_flag* was 1) or the *imol2* or -1 (signifying failure to do the SMM superposition).

## 11.87 NCS

### 11.87.1 set-draw-ncs-ghosts

`set-draw-ncs-ghosts imol istrate` [function]

Where:

- *imol* is an integer number
- *istrate* is an integer number

set drawing state of NCS ghosts for molecule number *imol*

### 11.87.2 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.

### 11.87.3 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*

### 11.87.4 ncs-update-ghosts

`ncs-update-ghosts imol` [function]

Where *imol* is an integer number

update ghosts for molecule number *imol*

### 11.87.5 make-dynamically-transformed-ncs-maps

`make-dynamically-transformed-ncs-maps imol_model imol_map  
overwrite_maps_of_same_name_flag` [function]

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

### 11.87.6 add-ncs-matrix

`add-ncs-matrix imol this_chain_id target_chain_id m11 m12 m13 m21` [function]  
`m22 m23 m31 m32 m33 t1 t2 t3`

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.

### 11.87.7 add-strict-ncs-matrix

`add-strict-ncs-matrix imol this_chain_id target_chain_id m11 m12 m13` [function]  
`m21 m22 m23 m31 m32 m33 t1 t2 t3`

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

### 11.87.8 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*

### 11.87.9 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*

### 11.87.10 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)

### 11.87.11 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.

### 11.87.12 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.

### 11.87.13 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.

### 11.87.14 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)

### 11.87.15 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)

### 11.87.16 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

## 11.88 Helices and Strands

### 11.88.1 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 molecule called "Reverse Helix" if the helix orientation isn't completely unequivocal.

Returns: the index of the new molecule.

### 11.88.2 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 rigorous search, but it will be slower.

Returns: the index of the new molecule.

### 11.88.3 place-strand-here-dialog

`place-strand-here-dialog` [function]  
show the strand placement gui.

Choose the python version in there, if needed. Call scripting function, display it in place, don't return a widget.

### 11.88.4 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.

### 11.88.5 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.

### 11.88.6 find-secondary-structure

`find-secondary-structure` *use\_helix helix\_length helix\_target* [function]  
*use\_strand strand\_length strand\_target*

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.

### 11.88.7 find-secondary-structure-local

`find-secondary-structure-local` *use\_helix helix\_length helix\_target* [function]  
*use\_strand strand\_length strand\_target radius*

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.

## 11.89 Nucleotides

### 11.89.1 find-nucleic-acids-local

`find-nucleic-acids-local` *radius* [function]

Where *radius* is a number

autobuild nucleic acid chains

Find secondary structure local to current view in the current map. Add to a molecule called "NuclAcid", create it if needed.

Returns: the index of the new molecule.

## 11.90 New Molecule by Section Interface

### 11.90.1 new-molecule-by-residue-type-selection

`new-molecule-by-residue-type-selection` *imol residue\_type* [function]

Where:

- *imol* is an integer number
- *residue\_type* is a string

create a new molecule that consists of only the residue of type *residue\_type* in molecule number *imol*

Returns: the new molecule number, -1 means an error.

### 11.90.2 new-molecule-by-atom-selection

`new-molecule-by-atom-selection` *imol atom\_selection* [function]

Where:

- *imol* is an integer number
- *atom\_selection* is a string

create a new molecule that consists of only the atoms specified by the mmdb atoms selection string in molecule number *imol*

Returns: the new molecule number, -1 means an error.

### 11.90.3 new-molecule-by-sphere-selection

`new-molecule-by-sphere-selection` *imol x y z r allow\_partial\_residues* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number
- *r* is a number
- *allow\_partial\_residues* is an integer number

create a new molecule that consists of only the atoms within the given radius (*r*) of the given position.

Returns: the new molecule number, -1 means an error.

## 11.91 RNA/DNA

### 11.91.1 ideal-nucleic-acid

`ideal-nucleic-acid` *RNA\_or\_DNA form single\_stranded\_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

### 11.91.2 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

Return a molecule that contains a residue that is the WC pair partner of the clicked/picked/selected residue.

### 11.91.3 watson-crick-pair-for-residue-range

`watson-crick-pair-for-residue-range` *imol chain\_id resno\_start resno\_end* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno\_start* is an integer number
- *resno\_end* is an integer number

add base pairs for the given residue range, modify molecule *imol* by creating a new chain

## 11.92 Sequence (Assignment)

### 11.92.1 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.

### 11.92.2 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.

### 11.92.3 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. If the chain of the molecule already had a chain assigned to it, then this will overwrite that old assignment with the new one.

### 11.92.4 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.

### 11.92.5 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.

### 11.92.6 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.

### 11.92.7 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.

## 11.93 Surface Interface

### 11.93.1 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 *istate* = 1 draw the surface (normal representation goes away)

if *istate* = 0 don't draw surface

### 11.93.2 set-transparent-electrostatic-surface

`set-transparent-electrostatic-surface` *imol opacity* [function]

Where:

- *imol* is an integer number
- *opacity* is a number

simple on/off screendoor transparency at the moment, an *opacity* > 0.0 will turn on screendoor transparency (stippling).

### 11.93.3 get-electrostatic-surface-opacity

`get-electrostatic-surface-opacity` *imol* [function]

Where *imol* is an integer number

return 1.0 for non transparent and 0.5 if screendoor transparency has been turned on.

## 11.94 FFFearing

### 11.94.1 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*

### 11.94.2 set-ffear-angular-resolution

`set-ffear-angular-resolution` *f* [function]

Where *f* is a number

set and return the ffear angular resolution in degrees

### 11.94.3 fffear-angular-resolution

`fffear-angular-resolution` [function]

return the ffear angular resolution in degrees

## 11.95 Remote Control

### 11.95.1 make-socket-listener-maybe

`make-socket-listener-maybe` [function]

try to make socket listener

### 11.95.2 set-socket-string-waiting

`set-socket-string-waiting` *s* [function]

Where *s* is a string

feed the main thread a scheme script to evaluate

### 11.95.3 set-socket-python-string-waiting

`set-socket-python-string-waiting` *s* [function]

Where *s* is a string

feed the main thread a python script to evaluate

## 11.96 Display Lists for Maps

### 11.96.1 set-display-lists-for-maps

`set-display-lists-for-maps` *i* [function]

Where *i* is an integer number

Should display lists be used for maps? It may speed things up if these are turned on (or off) - depends on graphics card and drivers. Pass 1 for on, 0 for off.

### 11.96.2 display-lists-for-maps-state

`display-lists-for-maps-state` [function]

return the state of `display_lists_for_maps`.

## 11.97 Browser Interface

### 11.97.1 browser-url

`browser-url` *url* [function]  
Where *url* is a string  
try to open given url in Web browser

### 11.97.2 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"

### 11.97.3 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.

## 11.98 Molprobit Interface

### 11.98.1 handle-read-draw-probe-dots

`handle-read-draw-probe-dots` *dots\_file* [function]  
Where *dots\_file* is a string  
pass a filename that contains molprobit's probe output in XtalView format

### 11.98.2 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 molprobit's probe output in unformatted format

### 11.98.3 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 molprobit for on edit chi angles intermediate atoms?

#### 11.98.4 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?

#### 11.98.5 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?

#### 11.98.6 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?

#### 11.98.7 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).

#### 11.98.8 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

#### 11.98.9 interactive-probe-dots-molprobity-radius

`interactive-probe-dots-molprobity-radius` [function]  
return the radius over which we can run interactive probe.

### 11.99 Map Sharpening Interface

#### 11.99.1 sharpen

`sharpen` *imol* *b\_factor* [function]  
Where:

- *imol* is an integer number
- *b\_factor* is a number

Sharpen map *imol* by *b\_factor* (note (of course) that positive numbers blur the map).

#### 11.99.2 set-map-sharpening-scale-limit

`set-map-sharpening-scale-limit` *f* [function]  
Where *f* is a number  
set the limit of the b-factor map sharpening slider (default 30)

## 11.100 Marking Fixed Atom Interface

### 11.100.1 clear-all-fixed-atoms

`clear-all-fixed-atoms imol` [function]

Where *imol* is an integer number

clear all fixed atoms

## 11.101 Partial Charges

### 11.101.1 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

show the partial charges for the residue of the given specs (charges are read from the dictionary)

## 11.102 EM interface

### 11.102.1 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

Scale the cell, for use with EM maps, where the cell needs to be adjusted. Use like: (scale-cell 2 1.012 1.012 1.012). Return error status, 1 means it worked, 0 means it did not work.

## 11.103 CCP4mg Interface

### 11.103.1 write-ccp4mg-picture-description

`write-ccp4mg-picture-description filename` [function]

Where *filename* is a string

write a ccp4mg picture description file

### 11.103.2 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

get element colour for imol as Python formatted list char

## 11.104 Aux functions

### 11.104.1 laplacian

`laplacian` *imol* [function]

Where *imol* is an integer number

Create the "Laplacian" (-ve second derivative) of the given map.

## 11.105 SMILES

### 11.105.1 do-smiles-gui

`do-smiles-gui` [function]

display the SMILES string dialog

## 11.106 PHENIX Support

### 11.106.1 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

## 11.107 Graphics Text

### 11.107.1 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

### 11.107.2 remove-text

`remove-text` *text\_handle* [function]

Where *text\_handle* is an integer number

Remove "3d" text item.

### 11.107.3 text-index-near-position

`text-index-near-position` *x y z r* [function]

Where:

- *x* is a number
- *y* is a number
- *z* is a number
- *r* is a number

return the closest text that is with r Å of the given position. If no text item is close, then return -1

## 11.108 PISA Interaction

### 11.108.1 pisa-interaction

`pisa-interaction` *imol\_1 imol\_2* [function]

Where:

- *imol\_1* is an integer number
- *imol\_2* is an integer number

return the molecule number of the interacting residues. Return -1 if no new model was created. Old, not very useful.

## 11.109 Jiggle Fit

### 11.109.1 fit-to-map-by-random-jiggle

`fit-to-map-by-random-jiggle` *imol chain\_id resno ins\_code n\_trials jiggle\_scale\_factor* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno* is an integer number
- *ins\_code* is a string
- *n\_trials* is an integer number
- *jiggle\_scale\_factor* is a number

jiggle fit to the current refinement map. return < -100 if not possible, else return the new best fit for this residue.

## 11.110 SBase interface

### 11.110.1 get-ccp4srs-monomer-and-dictionary

`get-ccp4srs-monomer-and-dictionary` *comp\_id* [function]

Where *comp\_id* is a string

return the new molecule number of the monomer.

The monomer will have chainid "A" and residue number 1.

Return -1 on failure to get monomer.

### 11.110.2 get-sbase-monomer

`get-sbase-monomer` *comp\_id* [function]

Where *comp\_id* is a string

same as above but using old name for back-compatibility

## 11.111 FLE-View

### 11.111.1 fle-view-set-water-dist-max

`fle-view-set-water-dist-max` *dist\_max* [function]

Where *dist\_max* is a number

set the maximum considered distance to water

default 3.25 Å.

### 11.111.2 fle-view-set-h-bond-dist-max

`fle-view-set-h-bond-dist-max` *h\_bond\_dist\_max* [function]

Where *h\_bond\_dist\_max* is a number

set the maximum considered hydrogen bond distance

default 3.9 Å.

## 11.112 LSQ-improve

### 11.112.1 lsq-improve

`lsq-improve` *imol\_ref* *ref\_selection* *imol\_moving* *moving\_selection* *n\_res* *dist\_crit* [function]

Where:

- *imol\_ref* is an integer number
- *ref\_selection* is a string
- *imol\_moving* is an integer number
- *moving\_selection* is a string
- *n\_res* is an integer number
- *dist\_crit* is a number

an slightly-modified implementation of the "lsq\_improve" algorithm of Kleywegt and Jones (1997).

Note that if a residue selection is specified in the residue selection(s), then the first residue of the given range must exist in the molecule (if not, then mmdb will not select any atoms from that molecule).

Kleywegt and Jones set `n_res` to 4 and `dist_crit` to 6.0.

## 11.113 single-model view

### 11.113.1 single-model-view-model-number

`single-model-view-model-number` *imol imodel* [function]

Where:

- *imol* is an integer number
- *imodel* is an integer number

put molecule number *imol* to display only model number *imodel*

### 11.113.2 single-model-view-this-model-number

`single-model-view-this-model-number` *imol* [function]

Where *imol* is an integer number

the current model number being displayed

return 0 on non-multimodel-molecule.

### 11.113.3 single-model-view-next-model-number

`single-model-view-next-model-number` *imol* [function]

Where *imol* is an integer number

change the representation to the next model number to be displayed

return 0 on non-multimodel-molecule.

### 11.113.4 single-model-view-prev-model-number

`single-model-view-prev-model-number` *imol* [function]

Where *imol* is an integer number

change the representation to the previous model number to be displayed

return 0 on non-multimodel-molecule.

## 11.114 graphics 2D ligand view

### 11.114.1 set-show-graphics-ligand-view

`set-show-graphics-ligand-view` *state* [function]

Where *state* is an integer number

set the graphics ligand view state

(default is 1 (on)).

## 12 More Scripting Functions

### 12.1 More Symmetry Functions

#### 12.1.1 get-symmetry

`get-symmetry imol` [function]

Where *imol* is an integer number

return the symmetry of the *imol*th molecule

Return as a list of strings the symmetry operators of the given molecule. If *imol* is a not a valid molecule, return an empty list.

### 12.2 Extra Map Functions

#### 12.2.1 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: a list of molecule numbers for the new maps

#### 12.2.2 auto-read-make-and-draw-maps-from-mtz

`auto-read-make-and-draw-maps-from-mtz filename` [function]

Where *filename* is a string

set the flag to do a difference map (too) on auto-read MTZ

#### 12.2.3 map-colour-components

`map-colour-components imol` [function]

Where *imol* is an integer number

return the colour triple of the *imol*th map

(e.g.: (list 0.4 0.6 0.8)). If invalid *imol* return scheme false.

### 12.3 Multi-Residue Torsion

#### 12.3.1 multi-residue-torsion-fit-scm

`multi-residue-torsion-fit-scm imol residues_specs_scm n_trials` [function]

Where:

- *imol* is an integer number
- *residues\_specs\_scm* is a SCM
- *n\_trials* is an integer number

fit residues

(note: fit to the current-refinement map)

## 12.4 Execute Refmac

### 12.4.1 execute-refmac-real

```
execute-refmac-real pdb_in_filename pdb_out_filename mtz_in_filename [function]
  mtz_out_filename cif_lib_filename fobs_col_name sigfobs_col_name
  r_free_col_name have_sensible_free_r_flag make_molecules_flag
  refmac_count_string swap_map_colours_post_refmac_flag imol_refmac_map
  diff_map_flag phase_combine_flag phib_string fom_string ccp4i_project_dir
```

Where:

- *pdb\_in\_filename* is a std::string
- *pdb\_out\_filename* is a std::string
- *mtz\_in\_filename* is a std::string
- *mtz\_out\_filename* is a std::string
- *cif\_lib\_filename* is a std::string
- *fobs\_col\_name* is a std::string
- *sigfobs\_col\_name* is a std::string
- *r\_free\_col\_name* is a std::string
- *have\_sensible\_free\_r\_flag* is an integer number
- *make\_molecules\_flag* is an integer number
- *refmac\_count\_string* is a std::string
- *swap\_map\_colours\_post\_refmac\_flag* is an integer number
- *imol\_refmac\_map* is an integer number
- *diff\_map\_flag* is an integer number
- *phase\_combine\_flag* is an integer number
- *phib\_string* is a std::string
- *fom\_string* is a std::string
- *ccp4i\_project\_dir* is a std::string

if *swap\_map\_colours\_post\_refmac\_flag* is not 1 then *imol\_refmac\_map* is ignored.

### 12.4.2 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

## 12.5 Dictionary Functions

### 12.5.1 dictionaries-read

`dictionaries-read` [function]  
return a list of all the dictionaries read

## 12.6 Restraints Interface

### 12.6.1 set-monomer-restraints

`set-monomer-restraints` *monomer\_type* *restraints* [function]  
Where:

- *monomer\_type* is a string
- *restraints* is a SCM

set the monomer restraints of the given *monomer\_type*  
Returns: scheme false or true for success or failure to set the restrains for *monomer\_type*

## 12.7 Atom Information functions

### 12.7.1 rename-from-serial-number

`rename-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: blank ("" ) on failure.

### 12.7.2 residue-name

`residue-name` *imol* *chain\_id* *resno* *ins\_code* [function]  
Where:

- *imol* is an integer number
- *chain\_id* is a const std::string &
- *resno* is an integer number
- *ins\_code* is a const std::string &

return the residue name of the specified residue

### 12.7.3 residue-info

`residue-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

Return a list of atom info for each atom in the specified residue.

output is like this: (list (list (list atom-name alt-conf) (list occ temp-fact element) (list x y z)))

occ can be a single number or a list of seven numbers of which the first is the isotropic B.

### 12.7.4 chain-fragments-scm

`chain-fragments-scm` *imol screen\_output\_also* [function]

Where:

- *imol* is an integer number
- *screen\_output\_also* is an integer number

chain fragments

### 12.7.5 add-molecule

`add-molecule` *molecule\_expression name* [function]

Where:

- *molecule\_expression* is a SCM
- *name* is a string

generate a molecule from an s-expression

return a molecule number, -1 on error

### 12.7.6 clear-and-update-molecule

`clear-and-update-molecule` *molecule\_number molecule\_expression* [function]

Where:

- *molecule\_number* is an integer number
- *molecule\_expression* is a SCM

update a molecule from a s-expression

And going the other way, given an s-expression, update *molecule\_number* by the given molecule. Clear what's currently there first though.

### 12.7.7 active-residue

`active-residue` [function]

return specs of the atom close to screen centre

Return a list of (list imol chain-id resno ins-code atom-name alt-conf) for atom that is closest to the screen centre in any displayed molecule. If there are multiple models with the same coordinates at the screen centre, return the attributes of the atom in the highest number molecule number.

return scheme false if no active residue

### 12.7.8 closest-atom-simple-scm

`closest-atom-simple-scm` [function]

return the specs of the closest displayed atom

Return a list of (list imol chain-id resno ins-code atom-name alt-conf (list x y z)) for atom that is closest to the screen centre in the given molecule (unlike active-residue, potential CA substitution is not performed). If there is no atom, or if imol is not a valid model molecule, return scheme false.

### 12.7.9 closest-atom

`closest-atom imol` [function]

Where *imol* is an integer number

return the specs of the closest atom in imolth molecule

Return a list of (list imol chain-id resno ins-code atom-name alt-conf (list x y z)) for atom that is closest to the screen centre in the given molecule (unlike active-residue, no account is taken of the displayed state of the molecule). If there is no atom, or if imol is not a valid model molecule, return scheme false.

### 12.7.10 residues-near-residue

`residues-near-residue imol residue_in radius` [function]

Where:

- *imol* is an integer number
- *residue\_in* is a SCM
- *radius* is a number

return residues near residue

Return residue specs for residues that have atoms that are closer than radius Angstroms to any atom in the residue specified by *res\_in*.

### 12.7.11 residues-near-position-scm

`residues-near-position-scm imol pos radius` [function]

Where:

- *imol* is an integer number
- *pos* is a SCM

- *radius* is a number

residues near residue

Return a list of pairs of (imol, residue\_spec). pos is a list of 3 numbers. (get imol from active-atom)

Returns: residues within radius of pos (x,y,z) position

### 12.7.12 hydrogenate-region

`hydrogenate-region` *radius* [function]

Where *radius* is a number

find the active residue, find the near residues (within radius) create a new molecule, run reduce on that, import hydrogens from the result and apply them to the molecule of the active residue.

### 12.7.13 add-hydrogens-from-file

`add-hydrogens-from-file` *imol pdb\_with\_Hs\_file\_name* [function]

Where:

- *imol* is an integer number
- *pdb\_with\_Hs\_file\_name* is a std::string

Add hydrogens to imol from the given pdb file.

## 12.8 Refinement with specs

### 12.8.1 regularize-residues

`regularize-residues` *imol residues* [function]

Where:

- *imol* is an integer number
- *residues* is a const std::vector< coot::residue\_spec\_t > &

regularize the given residues

### 12.8.2 mtz-file-name

`mtz-file-name` *imol* [function]

Where *imol* is an integer number

presumes that imol\_Refinement\_Map has been set

### 12.8.3 refine-zone-with-full-residue-spec-scm

`refine-zone-with-full-residue-spec-scm` *imol chain\_id resno1 incode\_1 resno2 incode\_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 the given residue range.

## 12.9 Water Chain Functions

### 12.9.1 water-chain-from-shelx-ins-scm

`water-chain-from-shelx-ins-scm imol` [function]  
 Where *imol* is an integer number  
 return the chain id of the water chain from a shelx molecule. Raw interface  
 Returns: scheme false if no chain or bad imol

### 12.9.2 water-chain-scm

`water-chain-scm imol` [function]  
 Where *imol* is an integer number  
 return the chain id of the water chain. Raw interface

## 12.10 Glyco Tools

### 12.10.1 print-glyco-tree

`print-glyco-tree imol chain_id resno ins_code` [function]  
 Where:

- *imol* is an integer number
- *chain\_id* is a const std::string &
- *resno* is an integer number
- *ins\_code* is a const std::string &

print the glycosylation tree that contains the specified residue

## 12.11 Spin Search Functions

### 12.11.1 spin-search

`spin-search imol_map imol chain_id resno ins_code direction_atoms_list  
 moving_atoms_list` [function]  
 Where:

- *imol\_map* is an integer number
- *imol* is an integer number
- *chain\_id* is a string

- *resno* is an integer number
- *ins\_code* is a string
- *direction\_atoms\_list* is a SCM
- *moving\_atoms\_list* is a SCM

for the given residue, spin the atoms in *moving\_atom\_list* around the bond defined by *direction\_atoms\_list* looking for the best fit to density of *imol\_map* map of the first atom in *moving\_atom\_list*. Works (only) with atoms in altconf ""

## 12.12 Rotamer Scoring

### 12.12.1 score-rotamers-scm

`score-rotamers-scm imol chain_id res_no ins_code alt_conf imol_map clash_flag lowest_probability` [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string
- *alt\_conf* is a string
- *imol\_map* is an integer number
- *clash\_flag* is an integer number
- *lowest\_probability* is a number

return the scores of the rotamers for this residue.

## 12.13 protein-db

### 12.13.1 protein-db-loops

`protein-db-loops imol_coords residue_specs imol_map nfrags preserve_residue_names` [function]

Where:

- *imol\_coords* is an integer number
- *residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *imol\_map* is an integer number
- *nfrags* is an integer number
- *preserve\_residue\_names* is a bool

Cowtan's protein\_db loops.

## 12.14 Coot's Hole implementation

### 12.14.1 hole

`hole imol start_x start_y start_z end_x end_y end_z colour_map_multiplier` [function]  
`colour_map_offset n_runs show_probe_radius_graph_flag`  
`export_surface_dots_file_name`

Where:

- *imol* is an integer number
- *start\_x* is a number
- *start\_y* is a number
- *start\_z* is a number
- *end\_x* is a number
- *end\_y* is a number
- *end\_z* is a number
- *colour\_map\_multiplier* is a number
- *colour\_map\_offset* is a number
- *n\_runs* is an integer number
- *show\_probe\_radius\_graph\_flag* is a bool
- *export\_surface\_dots\_file\_name* is a std::string

starting point and end point, colour map multiplier and shall the probe radius graph be shown (dummy value currently).

if `export_dots_file_name` string length is zero, then don't try to export the surface dots.

### 12.14.2 make-link

`make-link imol spec_1 spec_2 link_name length` [function]

Where:

- *imol* is an integer number
- *spec\_1* is a `coot::atom_spec_t &`
- *spec\_2* is a `coot::atom_spec_t &`
- *link\_name* is a `const std::string &`
- *length* is a number

make a link between the specified atoms

## 12.15 Drag and Drop Functions

### 12.15.1 handle-drag-and-drop-string

`handle-drag-and-drop-string uri` [function]

Where *uri* is a `const std::string &`

handle the string that get when a file or URL is dropped.

## 12.16 Map to Model Correlation

### 12.16.1 set-map-correlation-atom-radius

`set-map-correlation-atom-radius` *r* [function]

Where *r* is a number

The atom radius is not passed as a parameter to correlation.

### 12.16.2 map-to-model-correlation

`map-to-model-correlation` *imol* *residue\_specs* *neigh\_residue\_specs* *atom\_mask\_mode* *imol\_map* [function]

Where:

- *imol* is an integer number
- *residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *neigh\_residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

*atom\_mask\_mode* is as follows:

### 12.16.3 map-to-model-correlation-stats

`map-to-model-correlation-stats` *imol* *residue\_specs* *neigh\_residue\_specs* *atom\_mask\_mode* *imol\_map* [function]

Where:

- *imol* is an integer number
- *residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *neigh\_residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

map to model density correlation stats

*atom\_mask\_mode* is as follows:

### 12.16.4 map-to-model-correlation-per-residue

`map-to-model-correlation-per-residue` *imol* *specs* *atom\_mask\_mode* *imol\_map* [function]

Where:

- *imol* is an integer number
- *specs* is a const std::vector< coot::residue\_spec\_t > &
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

map to model density correlation, reported per residue

*atom\_mask\_mode* is as follows:

### 12.16.5 qq-plot-map-and-model-scm

`qq-plot-map-and-model-scm` *imol* *residue\_specs\_scm* [function]  
*neigh\_residue\_specs\_scm* *atom\_mask\_mode* *imol\_map*

Where:

- *imol* is an integer number
- *residue\_specs\_scm* is a SCM
- *neigh\_residue\_specs\_scm* is a SCM
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

QQ plot of the model density correlation, reported per residue.  
*atom-mask-mode* is as follows:

### 12.16.6 density-score-residue

`density-score-residue` *imol* *chain\_id* *res\_no* *ins\_code* *imol\_map* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string
- *imol\_map* is an integer number

simple density score for given residue (over-ridden by scripting function)

### 12.16.7 map-mean-scm

`map-mean-scm` *imol* [function]

Where *imol* is an integer number

return sigma for the given map. Return scheme False if not a valid map molecule number.

### 12.16.8 map-statistics-scm

`map-statistics-scm` *imol* [function]

Where *imol* is an integer number

return either scheme false on non-a-map or list (mean, standard-deviation, skew, kurtosis)

## 13 Scheme Scripting Functions

### 13.1 cc-interface\_8hh

### 13.2 More Symmetry Functions

#### 13.2.1 get-symmetry

`get-symmetry` *imol* [function]

Where *imol* is an integer number

return the symmetry of the *imol*th molecule

Return as a list of strings the symmetry operators of the given molecule. If *imol* is a not a valid molecule, return an empty list.

### 13.3 Extra Map Functions

#### 13.3.1 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: a list of molecule numbers for the new maps

#### 13.3.2 auto-read-make-and-draw-maps-from-mtz

`auto-read-make-and-draw-maps-from-mtz` *filename* [function]

Where *filename* is a string

set the flag to do a difference map (too) on auto-read MTZ

#### 13.3.3 map-colour-components

`map-colour-components` *imol* [function]

Where *imol* is an integer number

return the colour triple of the *imol*th map

(e.g.: (list 0.4 0.6 0.8). If invalid *imol* return scheme false.

### 13.4 Multi-Residue Torsion

#### 13.4.1 multi-residue-torsion-fit-scm

`multi-residue-torsion-fit-scm` *imol residues\_specs\_scm n\_trials* [function]

Where:

- *imol* is an integer number

- *residues\_specs\_scm* is a SCM
- *n\_trials* is an integer number

fit residues

(note: fit to the current-refinement map)

## 13.5 Execute Refmac

### 13.5.1 execute-refmac-real

```
execute-refmac-real pdb_in_filename pdb_out_filename mtz_in_filename [function]
                   mtz_out_filename cif_lib_filename fobs_col_name sigfobs_col_name
                   r_free_col_name have_sensible_free_r_flag make_molecules_flag
                   refmac_count_string swap_map_colours_post_refmac_flag imol_refmac_map
                   diff_map_flag phase_combine_flag phib_string fom_string ccp4i_project_dir
```

Where:

- *pdb\_in\_filename* is a std::string
- *pdb\_out\_filename* is a std::string
- *mtz\_in\_filename* is a std::string
- *mtz\_out\_filename* is a std::string
- *cif\_lib\_filename* is a std::string
- *fobs\_col\_name* is a std::string
- *sigfobs\_col\_name* is a std::string
- *r\_free\_col\_name* is a std::string
- *have\_sensible\_free\_r\_flag* is an integer number
- *make\_molecules\_flag* is an integer number
- *refmac\_count\_string* is a std::string
- *swap\_map\_colours\_post\_refmac\_flag* is an integer number
- *imol\_refmac\_map* is an integer number
- *diff\_map\_flag* is an integer number
- *phase\_combine\_flag* is an integer number
- *phib\_string* is a std::string
- *fom\_string* is a std::string
- *ccp4i\_project\_dir* is a std::string

if *swap\_map\_colours\_post\_refmac\_flag* is not 1 then *imol\_refmac\_map* is ignored.

### 13.5.2 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

## 13.6 Dictionary Functions

### 13.6.1 dictionaries-read

`dictionaries-read` [function]  
return a list of all the dictionaries read

## 13.7 Restraints Interface

### 13.7.1 set-monomer-restraints

`set-monomer-restraints` *monomer\_type* *restraints* [function]  
Where:

- *monomer\_type* is a string
- *restraints* is a SCM

set the monomer restraints of the given *monomer\_type*  
Returns: scheme false or true for success or failure to set the restrains for *monomer\_type*

## 13.8 Atom Information functions

### 13.8.1 rename-from-serial-number

`rename-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: blank ("") on failure.

### 13.8.2 residue-name

`residue-name` *imol* *chain\_id* *resno* *ins\_code* [function]  
Where:

- *imol* is an integer number
- *chain\_id* is a const std::string &
- *resno* is an integer number
- *ins\_code* is a const std::string &

return the residue name of the specified residue

### 13.8.3 residue-info

`residue-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

Return a list of atom info for each atom in the specified residue.

output is like this: (list (list (list atom-name alt-conf) (list occ temp-fact element) (list x y z)))

occ can be a single number or a list of seven numbers of which the first is the isotropic B.

### 13.8.4 chain-fragments-scm

`chain-fragments-scm` *imol screen\_output\_also* [function]

Where:

- *imol* is an integer number
- *screen\_output\_also* is an integer number

chain fragments

### 13.8.5 add-molecule

`add-molecule` *molecule\_expression name* [function]

Where:

- *molecule\_expression* is a SCM
- *name* is a string

generate a molecule from an s-expression

return a molecule number, -1 on error

### 13.8.6 clear-and-update-molecule

`clear-and-update-molecule` *molecule\_number molecule\_expression* [function]

Where:

- *molecule\_number* is an integer number
- *molecule\_expression* is a SCM

update a molecule from a s-expression

And going the other way, given an s-expression, update *molecule\_number* by the given molecule. Clear what's currently there first though.

### 13.8.7 active-residue

`active-residue` [function]

return specs of the atom close to screen centre

Return a list of (list imol chain-id resno ins-code atom-name alt-conf) for atom that is closest to the screen centre in any displayed molecule. If there are multiple models with the same coordinates at the screen centre, return the attributes of the atom in the highest number molecule number.

return scheme false if no active residue

### 13.8.8 closest-atom-simple-scm

`closest-atom-simple-scm` [function]

return the specs of the closest displayed atom

Return a list of (list imol chain-id resno ins-code atom-name alt-conf (list x y z)) for atom that is closest to the screen centre in the given molecule (unlike active-residue, potential CA substitution is not performed). If there is no atom, or if imol is not a valid model molecule, return scheme false.

### 13.8.9 closest-atom

`closest-atom imol` [function]

Where *imol* is an integer number

return the specs of the closest atom in imolth molecule

Return a list of (list imol chain-id resno ins-code atom-name alt-conf (list x y z)) for atom that is closest to the screen centre in the given molecule (unlike active-residue, no account is taken of the displayed state of the molecule). If there is no atom, or if imol is not a valid model molecule, return scheme false.

### 13.8.10 residues-near-residue

`residues-near-residue imol residue_in radius` [function]

Where:

- *imol* is an integer number
- *residue\_in* is a SCM
- *radius* is a number

return residues near residue

Return residue specs for residues that have atoms that are closer than radius Angstroms to any atom in the residue specified by *res\_in*.

### 13.8.11 residues-near-position-scm

`residues-near-position-scm imol pos radius` [function]

Where:

- *imol* is an integer number
- *pos* is a SCM

- *radius* is a number

residues near residue

Return a list of pairs of (imol, residue\_spec). pos is a list of 3 numbers. (get imol from active-atom)

Returns: residues within radius of pos (x,y,z) position

### 13.8.12 hydrogenate-region

`hydrogenate-region` *radius* [function]

Where *radius* is a number

find the active residue, find the near residues (within radius) create a new molecule, run reduce on that, import hydrogens from the result and apply them to the molecule of the active residue.

### 13.8.13 add-hydrogens-from-file

`add-hydrogens-from-file` *imol pdb\_with\_Hs\_file\_name* [function]

Where:

- *imol* is an integer number
- *pdb\_with\_Hs\_file\_name* is a std::string

Add hydrogens to imol from the given pdb file.

## 13.9 Refinement with specs

### 13.9.1 regularize-residues

`regularize-residues` *imol residues* [function]

Where:

- *imol* is an integer number
- *residues* is a const std::vector< coot::residue\_spec\_t > &

regularize the given residues

### 13.9.2 mtz-file-name

`mtz-file-name` *imol* [function]

Where *imol* is an integer number

presumes that imol\_Refinement\_Map has been set

### 13.9.3 refine-zone-with-full-residue-spec-scm

`refine-zone-with-full-residue-spec-scm` *imol chain\_id resno1* [function]

*inscode\_1 resno2 inscode\_2 altconf*

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 the given residue range.

## 13.10 Water Chain Functions

### 13.10.1 water-chain-from-shelx-ins-scm

`water-chain-from-shelx-ins-scm imol` [function]  
Where *imol* is an integer number  
return the chain id of the water chain from a shelx molecule. Raw interface  
Returns: scheme false if no chain or bad imol

### 13.10.2 water-chain-scm

`water-chain-scm imol` [function]  
Where *imol* is an integer number  
return the chain id of the water chain. Raw interface

## 13.11 Glyco Tools

### 13.11.1 print-glyco-tree

`print-glyco-tree imol chain_id resno ins_code` [function]  
Where:

- *imol* is an integer number
- *chain\_id* is a const std::string &
- *resno* is an integer number
- *ins\_code* is a const std::string &

print the glycosylation tree that contains the specified residue

## 13.12 Spin Search Functions

### 13.12.1 spin-search

`spin-search imol_map imol chain_id resno ins_code direction_atoms_list  
moving_atoms_list` [function]  
Where:

- *imol\_map* is an integer number
- *imol* is an integer number
- *chain\_id* is a string

- *resno* is an integer number
- *ins\_code* is a string
- *direction\_atoms\_list* is a SCM
- *moving\_atoms\_list* is a SCM

for the given residue, spin the atoms in *moving\_atom\_list* around the bond defined by *direction\_atoms\_list* looking for the best fit to density of *imol\_map* map of the first atom in *moving\_atom\_list*. Works (only) with atoms in altconf ""

## 13.13 Rotamer Scoring

### 13.13.1 score-rotamers-scm

`score-rotamers-scm` *imol chain\_id res\_no ins\_code alt\_conf imol\_map* [function]  
*clash\_flag lowest\_probability*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string
- *alt\_conf* is a string
- *imol\_map* is an integer number
- *clash\_flag* is an integer number
- *lowest\_probability* is a number

return the scores of the rotamers for this residue.

## 13.14 protein-db

### 13.14.1 protein-db-loops

`protein-db-loops` *imol\_coords residue\_specs imol\_map nfrags* [function]  
*preserve\_residue\_names*

Where:

- *imol\_coords* is an integer number
- *residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *imol\_map* is an integer number
- *nfrags* is an integer number
- *preserve\_residue\_names* is a bool

Cowtan's protein\_db loops.

## 13.15 Coot's Hole implementation

### 13.15.1 hole

`hole` *imol start\_x start\_y start\_z end\_x end\_y end\_z colour\_map\_multiplier* [function]  
*colour\_map\_offset n\_runs show\_probe\_radius\_graph\_flag*  
*export\_surface\_dots\_file\_name*

Where:

- *imol* is an integer number
- *start\_x* is a number
- *start\_y* is a number
- *start\_z* is a number
- *end\_x* is a number
- *end\_y* is a number
- *end\_z* is a number
- *colour\_map\_multiplier* is a number
- *colour\_map\_offset* is a number
- *n\_runs* is an integer number
- *show\_probe\_radius\_graph\_flag* is a bool
- *export\_surface\_dots\_file\_name* is a std::string

starting point and end point, colour map multiplier and shall the probe radius graph be shown (dummy value currently).

if `export_dots_file_name` string length is zero, then don't try to export the surface dots.

### 13.15.2 make-link

`make-link` *imol spec\_1 spec\_2 link\_name length* [function]

Where:

- *imol* is an integer number
- *spec\_1* is a `coot::atom_spec_t` &
- *spec\_2* is a `coot::atom_spec_t` &
- *link\_name* is a const std::string &
- *length* is a number

make a link between the specified atoms

## 13.16 Drag and Drop Functions

### 13.16.1 handle-drag-and-drop-string

`handle-drag-and-drop-string` *uri* [function]

Where *uri* is a const std::string &

handle the string that get when a file or URL is dropped.

## 13.17 Map to Model Correlation

### 13.17.1 set-map-correlation-atom-radius

`set-map-correlation-atom-radius` *r* [function]

Where *r* is a number

The atom radius is not passed as a parameter to correlation.

### 13.17.2 map-to-model-correlation

`map-to-model-correlation` *imol residue\_specs neigh\_residue\_specs* [function]  
*atom\_mask\_mode imol\_map*

Where:

- *imol* is an integer number
- *residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *neigh\_residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

atom-mask-mode is as follows:

### 13.17.3 map-to-model-correlation-stats

`map-to-model-correlation-stats` *imol residue\_specs* [function]  
*neigh\_residue\_specs atom\_mask\_mode imol\_map*

Where:

- *imol* is an integer number
- *residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *neigh\_residue\_specs* is a const std::vector< coot::residue\_spec\_t > &
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

map to model density correlation stats

atom-mask-mode is as follows:

### 13.17.4 map-to-model-correlation-per-residue

`map-to-model-correlation-per-residue` *imol specs atom\_mask\_mode* [function]  
*imol\_map*

Where:

- *imol* is an integer number
- *specs* is a const std::vector< coot::residue\_spec\_t > &
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

map to model density correlation, reported per residue

atom-mask-mode is as follows:

### 13.17.5 qq-plot-map-and-model-scm

`qq-plot-map-and-model-scm` *imol* *residue\_specs\_scm* [function]  
*neigh\_residue\_specs\_scm* *atom\_mask\_mode* *imol\_map*

Where:

- *imol* is an integer number
- *residue\_specs\_scm* is a SCM
- *neigh\_residue\_specs\_scm* is a SCM
- *atom\_mask\_mode* is an integer number
- *imol\_map* is an integer number

QQ pplot of the model density correlation, reported per residue.  
*atom-mask-mode* is as follows:

### 13.17.6 density-score-residue

`density-score-residue` *imol* *chain\_id* *res\_no* *ins\_code* *imol\_map* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string
- *imol\_map* is an integer number

simple density score for given residue (over-ridden by scripting function)

### 13.17.7 map-mean-scm

`map-mean-scm` *imol* [function]

Where *imol* is an integer number

return sigma for the given map. Return scheme False if not a valid map molecule number.

### 13.17.8 map-statistics-scm

`map-statistics-scm` *imol* [function]

Where *imol* is an integer number

return either scheme false on non-a-map or list (mean, standard-deviation, skew, kurtosis)

## 13.18 jligand

`*to-jligand-secret-file-name*` [procedure]

JLigand sends Coot a file that contains filenames for the link and new dictionary/restraints.

`*imol-jligand-link*` [procedure]

jligand internal parameter

`get-file-mtime` *file-name* [procedure]  
 this could be in utils

`start-jligand-listener` [procedure]  
 every fraction of a second look to see if `*from-jligand-secret-link-file-name*` has been updated. If so, then run the `handle-read-from-jligand-file` function.

### 13.19 gui-ligand-sliders

`mtz-file-name->refinement-column-labels` *file-name* [procedure]  
 return 3 values

### 13.20 ligand-check

`get-metrics-for-ligand` *imol chain-id res-no ins-code* [procedure]  
*refmac-input-mtz-file-name fobs-col sig-fobs-col rfree-col refmac-dir*  
 test: (using-active-atom (get-metrics-for-ligand aa-imol aa-chain-id aa-res-no aa-ins-code "rnasa-1.8-all\_refmac1.mtz" "FGMP18" "SIGFGMP18" "FreeR\_flag" "coot-refmac"))

`refmac-dir`: the dir where input/output refmac files should be written.

return (list correlation mogul bumps difference-map-stats b-factor-info) where: mogul is an improper pair (mogul-z-worst . mogul-out-file-name) or an error status difference-map-stats: the output of `map-to-model-correlation-stats-scm` (list correlation var\_x var\_y n sum\_x sum\_y D D2 (based on mean/sd of the map at the ligand) map\_mean map\_mean\_at\_ligand map\_sd map\_sd\_at\_ligand SCM\_UNDEFINED) and b-factor-info list of (median-ratio median-ligand median-env ks-test-result)

`*ligand-check-resolution-limit*` [procedure]  
 only look at ligands in maps with resolution worse than this: (to look at everything, set it to 0.1)

`filter-out-waters` *imol env-residues* [procedure]  
 remove residues that are waters from env-residues

### 13.21 mutate-in-scheme

`mutate-chain` *imol chain-id sequence* [procedure]  
 Mutate chain-id of molecule number imol given sequence. This presumes a protein sequence.

The number of residues in chain-id must match the length of sequence.

`multi-mutate` [procedure]  
 An internal function of mutate. This presumes a protein sequence.

`mutate-residue-range` *imol chain-id start-res-no stop-res-no sequence* [procedure]  
 The stop-res-no is inclusive, so usage e.g. (mutate-residue-range 0 "A" 1 2 "AC")  
 This presumes a protein sequence (not nucleic acid).

- mutate-and-autofit-residue-range** *imol chain-id start-res-no stop-res-no sequence* [procedure]  
 mutate and auto fit a residue range.  
 This presumes a protein sequence (not nucleic acid).  
 The sequence is a string of one letter codes
- mutate-and-auto-fit** *residue-number chain-id mol mol-for-map residue-type* [procedure]  
 mutate and autofit whole chain  
 This presumes a protein sequence (not nucleic acid).
- maf** [procedure]  
 a short-hand for mutate-and-auto-fit
- 3-letter-code->single-letter** *residue-type* [procedure]  
 return a char, return #\A for unknown residue-type
- mutate-residue-redundant** [procedure]  
 a wrapper for mutate-single-residue-by-seqno (which uses slightly inconvenient single letter code)  
 Here residue-type is the 3-letter code
- delete-sidechain-range** [procedure]  
 Delete (back to the CB stub) the side change in the range resno-start to resno-end

## 13.22 get-ebi

- check-dir-and-get-url** [procedure]  
 check the directory and get url url-string.
- get-url-str** [procedure]  
 get url-string for data type 'pdb or 'sfs
- get-ebi-pdb** *id* [procedure]  
 Return a molecule number on success or not a number (#f) or -1 on error.
- get-eds-pdb-and-mtz** *id* [procedure]  
 Return a list of molecules (i.e. the model molecule and the 2 maps).  
 or, if it didn't work then return #f

## 13.23 parse-pisa-xml

- \*pisa-command\*** [procedure]  
 (define \*pisa-command\* "/home/paule/ccp4/ccp4-6.1.2/bin/pisa")
- pisa-assemblies-xml** *imol file-name* [procedure]  
 it calls parse-pisa-assemblies which does the work
- pisa-handle-sxml-molecule** *imol molecule pisa-results-type* [procedure]  
 a interface-molecule record contains information about pvalue and residues.

**parse-pisa-assemblies** [procedure]

----- pisa assemblies: -----

pisa\_results name status total\_asm asm\_set ser\_no assembly id size mmsize  
diss\_energy asa bas entropy diss\_area int\_energy n\_uc n\_diss symNumber molecule  
chain\_id rxx rxy rxz tx ryx ryy ryz ty rzx rzy rzz tz rxx-f rxy-f rxz-f tx-f ryx-f ryy-f  
ryz-f ty-f rzx-f rzy-f rzz-f tz-f

**prep-for-pisa** *mode imol* [procedure]

20100213 prep-for-pisa needs to make directory, config file, write the pdb file and thre  
return value should be #f if there was a problem or some value where we can check  
that pisa -analyse ran (probably a directory). It is not clear right now where the  
output is going. config files has PISA\_WORK\_ROOT coot-pisa but things seems to  
be written to DATA\_ROOT /home/emsley/ccp4/ccp4-6.1.3/share/pisa which seems  
like a bug (or something like it) in pisa. Needs rechecking.

**cached-pisa-analysis** *dir* [procedure]

needs fleshing out (see notes for prep-for-pisa).

**parse-pisa-interfaces** *imol sxml-entity* [procedure]

pdb\_entry pdb\_code status n\_interfaces interface id type n\_occ int\_area int\_solv\_en  
pvalue stab\_en css overlap x-rel fixed h-bonds n\_bonds bond chain-1 res-1 seqnum-  
1 inscode-1 atname-1 chain-2 res-2 seqnum-2 inscode-2 atname-2 dist salt-bridges  
n\_bonds bond chain-1 res-1 seqnum-1 inscode-1 atname-1 chain-2 res-2 seqnum-2  
inscode-2 atname-2 dist ss-bonds n\_bonds bond chain-1 res-1 seqnum-1 inscode-1  
atname-1 chain-2 res-2 seqnum-2 inscode-2 atname-2 dist cov-bonds n\_bonds bond  
chain-1 res-1 seqnum-1 inscode-1 atname-1 chain-2 res-2 seqnum-2 inscode-2 atname-  
2 dist molecule id chain\_id class symop symop\_no cell\_i cell\_j cell\_k rxx rxy rxz tx  
ryx ryy ryz ty rzx rzy rzz tz int\_natoms int\_nres int\_area int\_solv\_en pvalue residues  
residue ser\_no name seq\_num ins\_code bonds asa bsa solv\_en

## 13.24 shelx

**shelxl-refine** [procedure]

**shelxl-refine-primitive** [procedure]

hkl-file-in can be null '() or (list astring).

**read-shelx-lst-file** [procedure]

ie. create a interesting-things GUI for split (and other things?) in a shelx .lst file.

## 13.25 hello

**first-non-trivial-name** [procedure]

Primarily for Indian Names.

Say we are given str: (list "M." "D." "Albert" "Dorkins"). We want to return  
""Albert" not "M.") We reject names that are one character long and names that  
are 2 characters long that end in ".". So, "M." is rejected, but "Ma" is not.

An exercise for the committed is to also reject run-together dotted initials such as "D.K.". I was sufficiently committed.

### 13.26 3d-generator-import

`prodr-g-xyzin` [procedure]  
if there is a `prodr-g-xyzin` set the current-time to its mtime, else #f

`import-from-3d-generator-from-mdl` *mdl-file-name comp-id* [procedure]  
This function can be overwritten by your favourite 3d conformer and restraints generator.

`new-molecule-by-smiles-string` *tlc-text smiles-text* [procedure]  
Run libcheck to convert from SMILES string

`get-mdl-latest-time` [procedure]  
not needed?

`prodr-g-flat` *imol-in chain-id-in res-no-in* [procedure]  
return #f (if fail) or a list of: the molecule number of the selected residue, the prodr-g output mol file-name, the prodr-g output pdb file-name

`get-ccp4srs-monomer-and-overlay` *comp-id* [procedure]  
import from SBASE, callback using `sbase_import_function`

### 13.27 clear-backup

`delete-coot-backup-files` *action-type* [procedure]

`clear-backup-gui` [procedure]  
Return #t or #f depending on if the GUI dialog was shown (it isn't show if there are no files to delete).

`clear-backups-maybe` [procedure]  
Return a status, #f or #t, did the GUI run?

Note that `clear-backup-gui` returns either #t or #f too.

If this function returns #f, then `coot_real_exit()` just exits with `coot_real_exit()`. Otherwise we wait for the GUI.

### 13.28 gui-add-linked-cho

`*add-linked-residue-tree-correlation-cut-off*` [procedure]  
now users can set this

### 13.29 ncs

**skip-to-next-ncs-chain** *direction* [procedure]

Skip the residue in the next chain (typically of a molecule with NCS) with the same residue number. If on the last chain, then wrap to beginning. If it can't find anything then don't move (and put a message in the status bar)

**single-manual-ncs-ghost** *imol resno-start resno-end ref-chain peer-chain* [procedure]

We can only see one peer at a time with this (each time we do a clear-ncs-ghost-matrices).

**manual-ncs-ghosts** *imol resno-start resno-end chain-id-list* [procedure]

chain-id-list is (list "A" "B" "C" "D"), i.e. the reference/target/master chain-id first and then the peers. This allows us to add many peers at the same time (unlike above function).

**update-ncs-ghosts-by-local-sphere** [procedure]

Update NCS ghosts based on local environment (residues within 6Å of (and including) the active residue).

Typically one would bind this function to a key.

**ncs-master-chain-id** *imol* [procedure]

Return the first master chain id (usually there is only one of course) or #f.

**ncs-ligand** *imol-protein ncs-master-chain-id imol-ligand chain-id-ligand resno-ligand-start resno-ligand-stop* [procedure]

This was designed to create an NCS copy of a ligand (or range of residues) in the active site of one chain to the as yet unoccupied active site of another, i.e. it makes a NCS ligand "D"1 that is a NCS copy of ligand "C"1 using an NCS operator that maps protein chain "A" onto chain "B".

### 13.30 raster3d-from-scheme

**render-image** [procedure]

run raster3d

**raytrace** [procedure]

Run either raster3d or povray

### 13.31 coot

**\*probe-command\*** [procedure]

This is full pathname of molprobity's probe program

**\*reduce-command\*** [procedure]

This is full pathname of molprobity's reduce program

**\*do-coot-tips-flag\*** [procedure]

This has been here (in a general place) because tips-gui (where it used to be is conditionally loaded). (default to tips-gui displayed is true).

**load-all-scheme** [procedure]

Note the position of `coot-gui` is important. It seem that if there are too many files in front of it (even blank ones!) `coot` barfs when it gets to `coot-gui.scm`.

20060203 I have now enabled `coot` in scripting mode (no graphics (`-no-graphics` command line option)). In that case, we need to not load up scheme files which load up `gtk` (currently `coot-gui` and `tips-gui`).

### 13.32 extra-top-level

**my-button-callback** [procedure]

This is what to do when the button is pressed. It can be any guile or `coot` function.

**my-top-level** [procedure]

define a simple window and put a button in it

### 13.33 fitting

**fit-protein *imol*** [procedure]

Note that residue with alt confs do not undergo auto-fit-rotamer. This is because that `autofit-rotamer` then `refine` will tend to put both rotamers into the same place. Not good. It seems a reasonable expectation that residues with an alternate conformation are already resonably well-fitted. So residues with alternate conformations undergo only real space refinement.

This is simple-minded and outdated now we have the interruptible version (below).

**\*continue-multi-refine\*** [procedure]

These 2 variables are used by `multi-refine` function(s), called by idle functions to refine just one residue.

**fit-protein-make-specs *imol chain-specifier*** [procedure]

`chain-specifier` can be a string, where it is the chain of interest. or `'all-chains`, where all chains are chosen.

**interruptible-fit-protein *imol func*** [procedure]

`func` is a refinement function that takes 2 args, one a residue spec, the other the `imol-refinement-map`. e.g. `fit-protein-fit-function`

**fit-chain *imol chain-id*** [procedure]

For each residue in chain `chain-id` of molecule number `imol`, do a rotamer fit and real space refinement of each residue. Don't update the graphics while this is happening (which makes it faster than `fit-protein`, but much less interesting to look at).

**fit-waters *imol . animate?*** [procedure]

For each residue in the solvent chains of molecule number `imol`, do a rigid body fit of the water to the density.

**stepped-refine-protein *imol . res-step*** [procedure]

Step through the residues of molecule number `imol` and at each step do a residue range refinement (unlike `fit-protein` for example, which does real-space refinement for every residue).

The step is set internally to 2.

- stepped-refine-protein-for-rama** *imol* [procedure]  
refine each residue with ramachandran restraints
- stepped-refine-protein-with-refine-func** *imol refine-func . res-step* [procedure]
- post-ligand-fit-gui** [procedure]  
The GUI that you see after ligand finding.
- molecules-matching-criteria** *test-func* [procedure]  
*test-func* is a function given one argument (a molecule number) that returns either #f if the condition is not satisfied or something else if it is. And that "something else" can be a list like (list label x y z) or (list "Bad Chiral" 0 "A" 23 "" "CA" "A")  
It is used in the create a button label and "what to do when the button is pressed".
- refine-active-residue-generic** *side-residue-offset* [procedure]  
This totally ignores insertion codes. A clever algorithm would need a re-write, I think. Well, we'd have at this end a function that took a chain-id res-no-1 ins-code-1 res-no-2 ins-code-2  
And refine-zone would need to be re-written too, of course. So let's save that for a rainy day (days... (weeks)).
- refine-active-residue** [procedure]  
refine active residue
- refine-active-residue-triple** [procedure]  
refine active residue triple
- manual-refine-residues** *side-residue-offset* [procedure]  
For just one (this) residue, side-residue-offset is 0.
- pepflip-active-residue** [procedure]  
Pepflip the active residue - needs a key binding.
- auto-fit-rotamer-active-residue** [procedure]  
Auto-fit rotamer on active residues
- add-extra-restraints-to-other-molecule** *imol chain-id resno-range-start resno-range-end atom-sel-type imol-ref* [procedure]  
Restrain the atoms in imol (in give range selection) to corresponding atoms in imol-ref. atom-sel-type is either 'all 'main-chain or 'ca

### 13.34 fascinating-things

- fascinating-clusters-gui** *window-name sorting-options cluster-list* [procedure]  
(list cluster-name-string cluster-center-go-button-label-string ccgb-x ccgb-y ccgb-z ; now a list of specific items (list (list specific-button-label-string button-red button-green button-blue specific-x specific-y specific-z) (list specific-button-label-string button-red button-green button-blue specific-x specific-y specific-z))))))

### 13.35 run-mogul

`run-mogul` *mode imol chain-id res-no ins-code prefix-str use-cache?* [procedure]  
 prefix-str can be blank  
 return: a string that is the mogul-out file-name a symbol: something went wrong

`run-mogul-show-results` *imol code mode chain-id res-no ins-code* [procedure]  
 return a string that is the mogul results file name that gets parsed to make results  
 (return #f on failure) mode is 'all or 'bonds-and-angles

### 13.36 background-demo

`background-demo` [procedure]  
 flash the background different colours in some uninteresting way.

### 13.37 brute-lsqman

`lsqman-count` [procedure]  
 brute-lsqman - run lsqman on chains A of two pdb-files and read in the result to coot.  
 Charlie Bond 2003. Can keep a count of the number of successful runs if necessary

### 13.38 coot-crash-catcher

`filter` *fn ls* [procedure]  
 Basic scheme function, filter the objects in list ls by function fn. e.g. (filter even?  
 (list 0 1 2 3) -> '(0 2))

`string-concatenate` *ls* [procedure]  
 simple scheme functions to concat the strings in ls (ls must contain only strings)

`directory-files` *dir* [procedure]  
 The following functions from PLEAC (guile version thereof of course).  
 or define a utility function for this

`glob` *pat dir* [procedure]  
 return a list of file names that match pattern pat in directory dir.

`run-command/strings` *cmd args data-list* [procedure]  
 Return the strings screen output of cmd (reversed) or #f if command was not found

`command-in-path?` *cmd* [procedure]  
 Return #t or #f:

`string->list-of-strings` *str* [procedure]  
 Return a list if str is a string, else return '()

`shell-command-to-string` *cmd* [procedure]  
 code from thi <ttn at mingle.glug.org>  
 run command and put the output into a string and return it. (c.f. `run-command/strings`)

|                                                              |             |
|--------------------------------------------------------------|-------------|
| <code>make-gdb-script</code>                                 | [procedure] |
| <code>get-gdb-strings</code><br>return #f or list of strings | [procedure] |
| <code>make-gui</code><br>gui                                 | [procedure] |

### 13.39 dictionary-generators

|                                                                                                                                                |             |
|------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| <code>generate-molecule-from-mmCIF-by-dict-gen</code> <i>comp-id</i><br><i>mmCIF-file-name</i><br>return a molecule number. Return -1 on fail. | [procedure] |
|------------------------------------------------------------------------------------------------------------------------------------------------|-------------|

### 13.40 generic-objects

|                                                                                                                                                                                                                                                                                                   |             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| <code>generic-object-is-displayed?</code><br>map to scheme names:                                                                                                                                                                                                                                 | [procedure] |
| <code>is-closed-generic-object?</code><br>map to scheme names:                                                                                                                                                                                                                                    | [procedure] |
| <code>generic-object-with-name</code><br>return a new generic object number for the given object obj-name. If there is no such object with name obj-name, then create a new one. (Maybe you want the generic object to be cleaned if it exists before returning, this function does not do that). | [procedure] |
| <code>generic-objects-gui</code><br>display a GUI for generic objects                                                                                                                                                                                                                             | [procedure] |
| <code>reduce-on-pdb-file</code> <i>imol pdb-in pdb-out</i><br>return status.                                                                                                                                                                                                                      | [procedure] |
| <code>probe</code><br>run molprobit (well reduce and probe) to make generic objects (and display the generic objects gui)                                                                                                                                                                         | [procedure] |
| <code>write-reduce-het-dict</code> <i>imol reduce-het-dict-file-name</i><br>Write the connectivity for the non-standard (non-water) residues in the given molecule for which we have the dictionary.<br>Don't return anything interesting.                                                        | [procedure] |
| <code>*interactive-probe-is-OK?*</code><br>gets set the first time we run interactive-probe. Takes values unset (initial value) 'yes and 'no)                                                                                                                                                     | [procedure] |
| <code>interactive-probe</code><br>run "probe" interactively, which in the current implementation, means that this function can run during a edit-chi angles manipulation, or after a real space refine zone.                                                                                      | [procedure] |

Thus function presumes that there are 2 pdb files in the current directory, one of which is the reference pdb file and the other is a pdb file containing the tmp/moving atom set.

The function takes arguments for the centre of the probe dots and the radius of the probe dots sphere. The chain id and residue number are also needed to pass as arguments to probe.

**probe-local-sphere** *imol radius* [procedure]  
 Update the generic objects probe dots from residues within radius of the screen centre.  
 Return nothing interesting.

### 13.41 contact-score-isolated-ligand

**contact-score-ligand** *imol res-spec* [procedure]  
 This will hydrogenate the active residue, not imol

### 13.42 tips-gui

**show-coot-tip-from-list** *n text* [procedure]  
 given a number and a gtk text widget *text*, put tip number *n* into the widget.

**increment-coot-tip-number** [procedure]  
 increment the tip number when the user sees a tip

**decrease-coot-tip-number** [procedure]  
 decrement the tip number when the user sees a tip

**tips-gui** [procedure]  
 run the tips gui.

### 13.43 refmac

**refmac-exe** [procedure]  
 This is the default refmac version, it is presumed to be in the path. It can be overridden using a re-definition either at the scripting interface or in one's ~/.coot file. E.g.: (define refmac-exec "/y/programs/xtal/refmac-latest/bin/refmac5-3-dec-2004")

**refmac-extra-params** [procedure]  
 Set this to a list of parameter strings:  
 If refmac-extra-params is a list of strings, it is used in preference to the "refmac-extra-params" file (should it exist). e.g. (set! refmac-extra-params (list "WEIGHT 0.2" "NCYC 10" "REFI BREF ISO" "REFI METH CGMAT" "REFI TYPE REST RESO 20 1.64"))

**run-refmac-by-filename** *pdb-in-filename pdb-out-filename* [procedure]  
*mtz-in-filename mtz-out-filename extra-cif-lib-filename imol-refmac-count*  
*swap-map-colours-post-refmac? imol-mtz-molecule show-diff-map-flag*  
*phase-combine-flag phib-fom-pair force-n-cycles make-molecules-flag*  
*ccp4i-project-dir f-col sig-f-col . r-free-col*

**extra-params-include-weight?** *params-list* [procedure]  
 Return #t if the list of strings *params-list* contains a string beginning with "WEIGHT". If not return #f

**get-refmac-extra-params** [procedure]  
 If *refmac-extra-params* is defined (as a list of strings), then return that, else read the file "refmac-extra-params".  
 Return a list a list of strings.

**run-refmac-for-phases** [procedure]  
 this is not run as a sub-thread, no useful return value.

### 13.44 redefine-functions

**enhanced-ligand-coot?** [procedure]  
 scm aliases.

**set-find-hydrogen-torsion** [procedure]  
 fix typo of *set-find-hydrogen-torsions* (backward compatibility in case anyone was using that)

**cif-file-for-comp-id** [procedure]  
 (define *residue->sdf-file residue-to-sdf-file*)

**toggle-idle-ligand-interactions** [procedure]  
 I changed the function name - save those (just a few) with scripts that I've handed out

### 13.45 group-settings

**rapper-dir** [procedure]  
 The rapper installation dir. This is just a guess, that it is installed in the users home directory.

### 13.46 tips

**tip-list** [procedure]  
 a list of tips for Coot

**no-coot-tips** [procedure]  
 Function to turn off coot tips at start

### 13.47 cns2coot

`cns->coot 2fofc-coeffs fofc-coeffs model-pdb` [procedure]  
 Read in cns coeff-data (given filenames) and a pdb molecule filename to make maps.

### 13.48 coot-utils

`*annotations*` [procedure]  
 3D annotations - a bit of a hack currently

`*default-ball-and-stick-selection*` [procedure]  
 used in Extensions -> Representation -> Ball & Stick

`rigid-body-refine-by-residue-ranges` [procedure]  
 rigid body refine using residue ranges. Takes 2 arguments, the first is the molecule number, the second is a list of residue-ranges. A residue range is (list *chain-id resno-start resno-end*).

`find-aligned-residue-type` [procedure]  
 add terminal residue is the normal thing we do with an aligned sequence, but also we can try to find the residue type of a residue in the middle of the chain that is modelled as an ALA, say.

`all-true? ls` [procedure]  
 not tail-recursive

`get-directory dir-name` [procedure]  
 make the directory and return the directory name. If you cant make it directly in this directory try to make it in \$HOME. Return #f if complete failure. e.g. coot-ccp4 or coot-backup

`molecule-has-hydrogens? imol` [procedure]  
 schemify function

`add-hydrogens-using-refmac-inner imol in-file-name out-file-name` [procedure]  
 out-file-name has been written, in-file-name needs to be read.

`post-manipulation-hook` [procedure]

`pre-release?` [procedure]  
 Return a boolean

`molecule-number-list` [procedure]  
 Return a list of molecule numbers (closed and open) The elements of the returned list need to be tested against is-valid-model-molecule?

`first-n n ls` [procedure]  
 first n fields of ls. if length ls is less than n, return ls. if ls is not a list, return ls. If n is negative, return ls.

- directory-is-modifiable?** *prefix-dir* [procedure]  
Test for *prefix-dir* (1) being a string (2) existing (3) being a directory (4) modifiable by user (ie. u+rwx). *prefix-dir* must be a string.  
Return #t or #f.
- absolutify** *file-name* [procedure]  
return an absolute *file-name* for *file-name* or #f
- file-name-directory** *file-name* [procedure]  
return the directory component of *file-name*, leave "/" on, if it's there. Note "x", "",  
"/" -> ""
- file-name-file** *file-name* [procedure]  
return the file component of *file-name* "x/y" -> "y" "x" -> "x"
- most-recently-created-file** [procedure]  
return #f on no-such-file
- residue-spec->atom-selection-string** *centre-residue-spec* [procedure]  
Convert a residue-spec to an mmdb atom selection string.
- residue-atom->atom-name** *ra* [procedure]  
residue-info atom
- residue-atom->alt-conf** *ra* [procedure]  
residue-info atom
- residue-atom->occupancy** *ra* [procedure]  
residue-info atom
- residue-spec->chain-id** *rs* [procedure]  
residue spec (e.g. from residue-near-residue)
- map-molecule-list** [procedure]  
Return a list of molecules that are maps
- model-molecule-list** [procedure]  
Return a list of molecules that are maps
- shelx-molecule?** *imol* [procedure]  
Return #t (#f) if *imol* is (isn't) a shelx molecule.
- set-virtual-trackball-type** *type* [procedure]  
Set the virtual trackball behaviour.  
trackball *type* is a symbol: either 'flat or 'spherical-surface.
- list-of-strings?** *ls* [procedure]  
Is *ls* a list of strings? Return #t or #f
- coot-replace-string** *string-in target replacing-str* [procedure]  
(coot-replace-string "one two three" " " "-") -> "one\_two\_three"

- string-append-with-spaces** *ls* [procedure]  
 string concat with spaces, *ls* must be a list of strings.
- rotation-centre** [procedure]  
 The screen centre.  
 return the rotation centre as a 3 membered list of numbers
- number-list** *a b* [procedure]  
 Make list of integers, *a* to *b*: eg (number-list 2 5) -> (2 3 4 5)
- string-member?** *atom ls* [procedure]  
*ls* must be a list of strings, *atom* must be a string.  
 return either #t or #f.
- member?** *atom ls* [procedure]
- range** *first . second* [procedure]  
 range: works like the eponymous python function e.g. (range 3) -> '(0, 1, 2) e.g.  
 (range 1 3) -> '(1, 2)
- shell-command-to-string** *cmd* [procedure]  
 code from thi <ttn at mingle.glug.org>  
 run command and put the output into a string and return it. (c.f. `run-command/strings`)
- shell-command-to-file-with-data** *cmd file-name data-list* [procedure]  
 run *cmd* putting output to *file-name* and reading commands data from the list of strings *data-list*.
- command-in-path?** *f* [procedure]  
 Is the command *f* in the path? return #t or #f.  
 Thank you for this, rixed at happyleptic.org
- command-in-path-or-absolute?** *cmd* [procedure]  
 Return #t or #f
- goosh-command** *cmd args data-list log-file-name screen-output-also?* [procedure]  
 Where *cmd* is e.g. "refmac" *args* is (list "HKLIN" "thing.mtz") *log-file-name* is "refmac.log" *data-list* is (list "HEAD" "END")  
 Return the exist status e.g. 0 or 1.
- goosh-command-with-file-input** *cmd args input-file log-file-name* [procedure]  
 run commands from an input file.
- run-command/strings** *cmd args data-list* [procedure]  
 Return the strings screen output of *cmd* or #f if command was not found
- close-float?** *x1 x2* [procedure]  
 Crude test to see if 2 floats are the same (more or less). Used in a greg test after setting the atom position.

|                                                                                                                                                         |             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| <b>strip-spaces</b> <i>str</i>                                                                                                                          | [procedure] |
| if passed a string, return a string with no spaces, else return #f.                                                                                     |             |
| <b>strip-leading-spaces</b> <i>str</i>                                                                                                                  | [procedure] |
| " 53" -> "53", " " -> ""                                                                                                                                |             |
| <b>string-append-with-string</b> <i>str-ls tag-str</i>                                                                                                  | [procedure] |
| Append strings with tag-str between them                                                                                                                |             |
| <b>strip-extension</b> <i>s</i>                                                                                                                         | [procedure] |
| "a.b.res" -> "a.b" file-name-sans-extension                                                                                                             |             |
| <b>file-name-extension</b>                                                                                                                              | [procedure] |
| What is the extension of file-name?                                                                                                                     |             |
| "a.pdb" -> "pdb" "" -> ""                                                                                                                               |             |
| <b>add-tmp-extension-to</b>                                                                                                                             | [procedure] |
| e.g. "a.pdb" -> "a-tmp.pdb"                                                                                                                             |             |
| <b>file-name-sans-extension</b>                                                                                                                         | [procedure] |
| Same function as strip-extension, different name, as per scsh, in fact.                                                                                 |             |
| <b>strip-path</b> <i>s</i>                                                                                                                              | [procedure] |
| /a/b.t -> b.t d/e.ext -> e.ext file-name-sans-path                                                                                                      |             |
| <b>slash-start?</b> <i>s</i>                                                                                                                            | [procedure] |
| does s start with a "/" ? return #t or #f                                                                                                               |             |
| <b>string-concatenate</b> <i>ls</i>                                                                                                                     | [procedure] |
| simple scheme functions to concat the strings in ls (ls must contain only strings)                                                                      |             |
| <b>unique-date/time-str</b>                                                                                                                             | [procedure] |
| return a string that contains the date/time e.g. "2006-01-02_2216.03"                                                                                   |             |
| <b>every-nth</b> <i>ls n</i>                                                                                                                            | [procedure] |
| return a list that has only every-nth members; e.g. (every-nth '(0 1 2 3 4 5 6 7 8) 2) -> '(0 2 3 6 8) (every-nth '(0 1 2 3 4 5 6 7 8) 3) -> '(0 3 6)   |             |
| <i>n</i> must be positive                                                                                                                               |             |
| <b>get-atom-from-residue</b> <i>atom-name residue-atoms alt-conf</i>                                                                                    | [procedure] |
| now useful in testing                                                                                                                                   |             |
| residue-atoms must be a list                                                                                                                            |             |
| <b>get-atom</b> <i>imol chain-id resno ins-code atom-name . alt-conf</i>                                                                                | [procedure] |
| Return an atom info or #f (if atom not found).                                                                                                          |             |
| <b>multi-read-pdb</b> <i>glob-pattern dir</i>                                                                                                           | [procedure] |
| multi-read-pdb reads all the files matching <i>glob-pattern</i> in directory <i>dir</i> . Typical usage of this might be: (multi-read-pdb "a*.pdb" ".") |             |
| <b>read-pdb-all</b>                                                                                                                                     | [procedure] |
| read-pdb-all reads all the "*.pdb" files in the current directory.                                                                                      |             |

- string->list-of-strings** *str* [procedure]  
Return a list if *str* is a string, else return '()
- append-dir-file** *dir-name file-name* [procedure]  
In a laughable attempt to minimise system dependence.
- append-dir-dir** *dir-name sub-dir-name* [procedure]  
Similarly attempting to minimise system dependence.
- directory-as-file-name** *dir* [procedure]  
remove any trailing /s
- is-directory?** *file-name* [procedure]  
return #t or #f depending on if *file-name* (which must be a string) is a directory.
- coot-mkdir** *dir-name* [procedure]  
return the *dir-name* on success.  
return #f if *dir-name* is a file or we can't do the mkdir.
- directory-files** *dir* [procedure]  
The following functions from PLEAC (guile version thereof of course).  
Return: a list of files in the given directory
- glob** *pat dir* [procedure]  
return a list of file names that match pattern *pat* in directory *dir*.
- useful-glob** *dir pat* [procedure]  
prepend the files with the directory *dir*  
(note that the args are reversed to "natural" order (which is the reverse of glob args))
- view-matrix** [procedure]  
return the view matrix (useful for molscript, perhaps).
- view-quaternion** [procedure]  
return the view quaternion
- add-view** *position quaternion zoom view-name* [procedure]  
Return the view number
- matrix->quaternion** [procedure]  
Convert a view matrix to a view quaternion to set Coot view internals.
- set-view-matrix** *m00 m10 m20 m01 m11 m21 m02 m12 m22* [procedure]  
Set the view matrix using *matrix->quaternion*.  
Useful for using a view matrix from another program, perhaps.
- molecule-centre** *imol* [procedure]  
Return the molecule centre as a list of 3 numbers.  
Note: *mol-cen* could contain values less than -9999.

- move-molecule-to-screen-centre** *imol* [procedure]  
Move the centre of molecule number *imol* to the current screen centre
- move-molecule-here** [procedure]  
This is a short name for the above.
- move-molecule-to-screen-center** [procedure]  
this is an americanism
- identity-matrix** [procedure]  
Return a nine-membered list of numbers.
- translation** *axis length* [procedure]  
e.g. (translation 'x 2) -> '(2 0 0) Return: "scheme false" on error
- rotate-about-screen-axis** *axis degrees* [procedure]  
Rotate degrees about screen axis, where axis is either 'x, 'y or 'z.
- toggle-display-map** *imol idummy* [procedure]  
Support for old toggle functions. (consider instead the raw functions use the direct set\_displayed functions).
- toggle-display-mol** *imol* [procedure]  
toggle the display of *imol*
- toggle-active-mol** *imol* [procedure]  
toggle the active state (clickability) of *imol*
- scheme-representation** *imol* [procedure]  
return a scheme representation of molecule *imol*, or #f if we can't do it (*imol* is a map, say).
- reorder-chains** *imol* [procedure]  
reorder chains
- transform-coords-molecule** *imol rtop* [procedure]  
transform a coordinates molecule by a coot-rtop (which is a SCM expression of a clipper::RTop), i.e. a list of a 9-element list and a 3 element list. e.g. (list (list 1 0 0 0 1 0 0 0 1) (list 4.5 0.4 1.2)).
- transform-map** [procedure]  
(transform-map *imol mat trans about-pt radius space-group cell*)  
where space-group is a HM-symbol and cell is a list of 6 parameters, where the cell angles are in degrees.  
or (transform-map *imol trans about-pt radius*) for a simple translation  
or (transform-map *imol trans radius*) when using the default rotation-centre as the about-pt
- get-first-ncs-master-chain** [procedure]  
return then NCS master of the first molecule that has ncs.  
return "" on fail to find an ncs chain

- transform-map-using-lsq-matrix** *imol-ref ref-chain ref-resno-start* [procedure]  
*ref-resno-end imol-mov mov-chain mov-resno-start mov-resno-end imol-map*  
*about-pt radius*  
Remember, that now the about-pt is the "to" point, i.e. the maps are brought from somewhere else and generated about the about-pt.
- brighten-map** *imol scale-factor* [procedure]  
Make the imol-th map brighter.
- brighten-maps** [procedure]  
Make all maps brighter
- darken-maps** [procedure]  
Make all maps darker.
- chain-ids** *imol* [procedure]  
Return a list of chain ids for given molecule number *imol*. return empty list on error
- is-solvent-chain?** *imol chain-id* [procedure]  
convert from interface name to schemish name  
Return #t or #f.
- valid-model-molecule?** *imol* [procedure]  
schemey interface to eponymous scripting interface function. Return scheme true or false
- valid-map-molecule?** *imol* [procedure]  
schemey interface to eponymous scripting interface function. Return scheme true or false.
- valid-refinement-map?** [procedure]  
Return #t or #f
- shelx-molecule?** *imol* [procedure]  
schemey interface to shelx molecule test  
Return #t or #f.
- is-difference-map?** *imol-map* [procedure]  
Return #t or #f.
- residue-exists?** *imol chain-id resno ins-code* [procedure]  
Does residue resno with insertion code ins-code of chain chain-id and in molecule number imol exist?  
Return #t or #f.
- residue-has-hetatms?** *imol chain-id res-no ins-code* [procedure]  
Does the residue contain hetatoms? Return #t or #f.
- centre-of-mass** *imol* [procedure]  
Return a list of 3 float for the centre of mas of molecule number imol.  
on failure return #f.

- atom-specs** *imol chain-id resno ins-code atom-name alt-conf* [procedure]  
Return as a list the occupancy temperature-factor element x y z coordinates of the given atom. (the x,y,z are in Cartesian Angstroms).  
on error (e.g. atom not found) return #f
- guess-refinement-map** [procedure]  
return a guess at the map to be refined (usually called after imol-refinement-map returns -1)
- target-auto-weighting-value** [procedure]  
Ian Tickle says (as far as I can understand) that the target rmsd should be 0.25 or thereabouts. You can over-ride it now.
- auto-weight-for-refinement** [procedure]  
Set the refinement weight (matrix) by iterating the refinement and varying the weight until the chi squares (not including the non-bonded terms) reach 1.0 +/- 10%. It uses sphere refinement. The refinement map must be set. At the end show the new weight in the status bar. Seems to take about 5 rounds.
- print-sequence** *imol* [procedure]  
Print the sequence of molecule number *imol*  
This is not really a util, perhaps it should be somewhere else?
- pir-file-name->pir-sequence** *pir-file-name* [procedure]  
simple utility function to return the contents of a file as a string.
- associate-pir-file** *imol chain-id pir-file-name* [procedure]  
Associate the contents of a PIR file with a molecule.
- graphics-comma-key-pressed-hook** [procedure]  
comma key hook
- graphics-dot-key-pressed-hook** [procedure]  
dot key hook
- \*key-bindings\*** [procedure]  
a list of (code key name thunk) e.g. '(103 "g" "Goto Blob" (blob-under-pointer-to-screen-centre))
- add-key-binding** *name key thunk* [procedure]  
Add a key binding  
with a given name, key (e.g. "x" or "S") and the function to run (a thunk) when that key is pressed.
- graphics-general-key-press-hook** *key* [procedure]  
general key press hook, not for public use.
- read-vu-file** [procedure]  
Function requested by Mark White.  
read XtalView (and maybe other) .vu files and convert them into generic objects.  
Pass the filename and an object name e.g. (read-vu-file "axes.vu" "axes")  
Returns: nothing interesting.

- residues-matching-criteria** *imol residue-test-func* [procedure]  
Return a list of residues, each of which has a return value at the start, ie. (list return-value chain-id res-no ins-code)
- all-residues** *imol* [procedure]  
Return residue specs for all residues in imol (each spec is preceded by #t)
- residues-with-alt-confs** *imol* [procedure]  
Return a list of all residues that have alt confs: where a residue is specified thusly: (list chain-id resno ins-code)
- residue-alt-confs** *imol chain-id res-no ins-code* [procedure]  
Return a list of all the altconfs in the residue. Typically this will return (list "") or (list "A" "B")
- res-spec->chain-id** *res-spec* [procedure]  
simple extraction function
- res-spec->res-no** *res-spec* [procedure]  
simple extraction function
- res-spec->ins-code** *res-spec* [procedure]  
simple extraction function
- residue-spec->atom-for-centre** *imol chain-id res-no ins-code* [procedure]  
Return #f if no atom can be found given the spec else return a list consisting of the atom name and alt-conf specifier.  
Choose an atom that is called " CA ". Failing that choose the first atom.
- update-go-to-atom-from-current-atom** [procedure]
- flip-active-ligand** [procedure]
- delete-atom-by-active-residue** [procedure]  
Typically one might want to use this on a water, but it deletes the nearest CA currently... Needs a re-think. Should active-atom just return the nearest atom and not be clever about returning a CA.
- merge-solvent-chains** *imol* [procedure]
- mutate-by-overlap** *imol chain-id-in resno tlc* [procedure]  
change chain ids with residue range for the PTY
- phosphorylate-active-residue** [procedure]  
A bit of fun
- overlay-my-ligands** *imol-mov chain-id-mov resno-mov imol-ref chain-id-ref resno-ref* [procedure]  
A function for Overlaying ligands. The transformation is applied to all the atoms of the molecule that contains the moving ligand.

|                                                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |             |
|-------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| <code>label-all-CAs</code>                      | <i>imol</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | [procedure] |
| <code>label-all-atoms-in-residue</code>         | <i>imol chain-id resno inscode</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | [procedure] |
| <code>label-all-active-residue-atoms</code>     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | [procedure] |
| <code>sanitise-alt-confs</code>                 | <i>atom-info atom-ls</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | [procedure] |
|                                                 | Resets alt confs and occupancies of atoms in residue that have orphan alt-loc attributes.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |             |
| <code>sanitise-alt-confs-in-residue</code>      | <i>imol chain-id resno inscode</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | [procedure] |
| <code>sanitise-alt-confs-active-residue</code>  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | [procedure] |
|                                                 | Resets alt confs and occupancies of atoms in residue that have orphan alt-loc attributes. Use the active-residue.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |             |
| <code>print-molecule-names</code>               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | [procedure] |
| <code>save-dialog-positions-to-init-file</code> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | [procedure] |
| <code>multi-chicken</code>                      | <i>imol . n-colours</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | [procedure] |
|                                                 | multiple maps of varying colour from a given map.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |             |
| <code>BALL_AND_STICK</code>                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | [procedure] |
|                                                 | simple enumeration                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |             |
| <code>hilight-binding-site</code>               | <i>imol centre-residue-spec hilight-colour radius</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | [procedure] |
|                                                 | hilight-colour is specified in degrees (round the colour wheel - starting at yellow (e.g. 230 is purple))                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |             |
| <code>pukka-puckers?</code>                     | <i>imol</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | [procedure] |
|                                                 | To paraphrase: The distance of the plane of the base to the following phosphate is highly correlated to the pucker of the ribose.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |             |
|                                                 | An analysis of the structures in RNADB2005 shows that a critical distance of 3.3A provides a partition function to separate C2' from C3' endo pucker. Not all ribose follow this rule. There may be some errors in the models comprising RNADB2005. So we check the distance of the following phosphate to the plane of the ribose and record the riboses that are inconsitent. We also report puckers that are not C2' or C3'. The puckers are determined by the most out-of-plane atom of the ribose (the rms deviation of the 4 atoms in the plane is calculated, but not used to determine the puckering atom). |             |
| <code>*use-mogul*</code>                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | [procedure] |
|                                                 | if you don't have mogul, set this to #f                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |             |
| <code>prodrgr-ify</code>                        | <i>imol chain-id res-no ins-code</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | [procedure] |
|                                                 | Generate restraints from the residue at the centre of the screen using PRODRG. Delete hydrogens from the residue because PRODRG has anomalous hydrogens.                                                                                                                                                                                                                                                                                                                                                                                                                                                            |             |
| <code>add-annotation-here</code>                | <i>text</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | [procedure] |
| <code>save-annotations</code>                   | <i>file-name</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | [procedure] |
| <code>load-annotations</code>                   | <i>file-name</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | [procedure] |

**make-latest-version-url** [procedure]

Here we construct the url that contains the latest (pre) release info adding in "pre-release" if this binary is a pre-release. args ends up as something like: ("s" "xxx/phone-home.scm" "pre-release" "binary" "Linux-1386-fedora-10-python-gtk2" "command-line" "/home/xx/coot/bin/coot")

**run-download-binary-curl** *revision version-string* [procedure]  
*pending-install-in-place-func set-file-name-func*

Get the binary (i.e. the action that happens when the download button is pressed). This is run in a thread, so it can't do any graphics stuff.

return #t if tar file was successfully downloaded and untared and #f if not.

**get-revision-from-string** *str* [procedure]  
(used in downloading new version)

**coot-split-version-string** *str* [procedure]  
e.g. input: "coot-0.6.2-pre-1-revision-2765\n" output: "coot-0.6.2-pre-1-revision-2765"

**load-default-sequence** [procedure]

In the first case the sequence is assigned to the closest match (model sequence to target sequence), subsequently only chains without a sequence associated with them are candidates for matching. The protein sequence has to have at least 95% sequence identity with the target sequence in "default.seq"

**coot-updates-error-handler** *key . args* [procedure]  
not really for public manipulation.

**update-self** [procedure]  
keep a copy of the old directories around in a directory named after expiration time.

**chiral-centre-inverter-scm-version** [procedure]  
This should almost all be c++ code so that Bernie doesn't have to redo it. This is temporary then.

**get-drug-via-wikipedia** *drug-name-in* [procedure]  
return a string "DB00xxxx.mol" or some such - this is the file name of the mdl mol file from drugbank. Or #f/undefined on fail. Test result with string?.

If you are debugging this function, you probably want to be looking at handle-rev-string or drugbox->drugitem.

**coot-has-pygtk?** [procedure]  
to determine if we have pygtk

## 13.49 coot-lsq

`lsq-match-type-symbol->number` *match-type-in* [procedure]  
 Internal type conversion for LSQ fitting. Return a number according to the symbol *match-type-in*

`set-match-element` [procedure]  
 Create matchers, 7 elements: (list *ref-start-resno* *ref-end-resno* *ref-chain-id* *imol-ref* *mov-start-resno* *mov-end-resno* *mov-chain-id* *imol-mov* *match-type*)

`lsq-match` *imol-ref* *imol-moving* *match-list* [procedure]  
 The scripting interface to LSQ matching. Pass molecule numbers for the reference (*imol-ref*) and moving (*imol-moving*) molecules and a match list. The match list format is described in the manual.

`simple-lsq-match` *ref-start-resno* *ref-end-resno* *ref-chain-id* *imol-ref* [procedure]  
*mov-start-resno* *mov-end-resno* *mov-chain-id* *imol-mov* *match-type*  
 Simple interface to LSQ fitting. More often than not this is what you will want, I imagine, e.g. (simple-lsq-match 940 950 "A" 0 940 950 "A" 1 'main)

## 13.50 c-interface\_8h

### 13.51 Startup Functions

#### 13.51.1 set-prefer-python

`set-prefer-python` [function]  
 tell coot that you prefer to run python scripts if/when there is an option to do so.

#### 13.51.2 prefer-python

`prefer-python` [function]  
 the python-preferred mode.  
 This is available so that the scripting functions know whether or not to put themselves onto in as menu items.  
 If you consider using this, consider in preference `use_gui_qm == 2`, which is used elsewhere to stop python functions adding to the gui, when `guile-gtk` functions have already done so. We should clean up this (rather obscure) interface at some stage.  
 return 1 for python is preferred, 0 for not.

### 13.52 File System Functions

#### 13.52.1 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.

### 13.52.2 `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.

### 13.52.3 `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

### 13.52.4 `add-coordinates-glob-extension`

`add-coordinates-glob-extension` *ext* [function]

Where *ext* is a string

add an extension to be treated as coordinate files

### 13.52.5 `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

### 13.52.6 `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

### 13.52.7 `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

### 13.52.8 `remove-coordinates-glob-extension`

`remove-coordinates-glob-extension` *ext* [function]

Where *ext* is a string

remove an extension to be treated as coordinate files

**13.52.9 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

**13.52.10 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

**13.52.11 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

**13.52.12 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

**13.52.13 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

**13.52.14 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

**13.52.15 filter-fileselection-filenames-state**

`filter-fileselection-filenames-state` [function]

, return the state of the above variable

**13.52.16 file-type-coords**

`file-type-coords` *file\_name* [function]

Where *file\_name* is a string

is the given file name suitable to be read as coordinates?

### 13.52.17 open-coords-dialog

`open-coords-dialog` [function]  
display the open coordinates dialog

### 13.52.18 set-file-chooser-selector

`set-file-chooser-selector` *istate* [function]  
Where *istate* is an integer number  
this flag set chooser as default for windows, otherwise use selector 0 is selector 1 is  
chooser

## 13.53 Widget Utilities

### 13.53.1 set-main-window-title

`set-main-window-title` *s* [function]  
Where *s* is a string  
set the main window title.  
function added for Lothar Esser

## 13.54 MTZ and data handling utilities

### 13.54.1 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

## 13.55 Molecule Info Functions

### 13.55.1 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

### 13.55.2 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

### 13.55.3 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 (scheme False) on failure.

### 13.55.4 n-models

`n-models` *imol* [function]

Where *imol* is an integer number

the *chain\_id* (string) of the *ichain*-th chain molecule number *imol*

return the number of models in molecule number *imol*

useful for NMR or other such multi-model molecules.

return the number of models or -1 if there was a problem with the given molecule.

Returns: the chain-id

### 13.55.5 n-chains

`n-chains` *imol* [function]

Where *imol* is an integer number

number of chains in molecule number *imol*

Returns: the number of chains

### 13.55.6 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*)

This wraps the mmdb function isSolventChain().

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.

### 13.55.7 is-protein-chain-p

`is-protein-chain-p` *imol chain\_id* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string

is this a protein chain? [Raw function]

This is a raw interface function, you should generally not use this, but instead use (is-protein-chain? imol chain-id)

This wraps the mmdb function isAminoacidChain().

Returns: -1 on error, 0 for no, 1 for is "a protein chain". We wouldn't want to be doing rotamer searches and the like on such a chain.

### 13.55.8 is-nucleotide-chain-p

`is-nucleotide-chain-p` *imol chain\_id* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string

is this a nucleic acid chain? [Raw function]

This is a raw interface function, you should generally not use this, but instead use (is-nucleicacid-chain? imol chain-id)

This wraps the mmdb function isNucleotideChain(). For completeness.

Returns: -1 on error, 0 for no, 1 for is "a nucleicacid chain". We wouldn't want to be doing rotamer searches and the like on such a chain.

### 13.55.9 n-residues

`n-residues` *imol* [function]

Where *imol* is an integer number

return the number of residues in the molecule,

return -1 if this is a map or closed.

### 13.55.10 sort-chains

`sort-chains` *imol* [function]

Where *imol* is an integer number

return a list of the remarks of hte molecule number imol

sort the chain ids of the imol-th molecule in lexographical order

### 13.55.11 sort-residues

`sort-residues` *imol* [function]

Where *imol* is an integer number

sort the residues of the imol-th molecule

### 13.55.12 remarks-dialog

`remarks-dialog` *imol* [function]

Where *imol* is an integer number

a gui dialog showing remarks header info (for a model molecule).

### 13.55.13 print-header-secondary-structure-info

`print-header-secondary-structure-info` *imol* [function]

Where *imol* is an integer number

simply print secondary structure info to the terminal/console. In future, this could/should return the info.

### 13.55.14 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)

### 13.55.15 add-ligand-delete-residue-copy-molecule

`add-ligand-delete-residue-copy-molecule` *imol\_ligand\_new* [function]

*chain\_id\_ligand\_new resno\_ligand\_new imol\_current chain\_id\_ligand\_current*

*resno\_ligand\_current*

Where:

- *imol\_ligand\_new* is an integer number
- *chain\_id\_ligand\_new* is a string
- *resno\_ligand\_new* is an integer number
- *imol\_current* is an integer number
- *chain\_id\_ligand\_current* is a string
- *resno\_ligand\_current* is an integer number

Copy a molecule with addition of a ligand and a deletion of current ligand.

This function is used when adding a new (modified) ligand to a structure. It creates a new molecule that is a copy of the current molecule except that the new ligand is added and the current ligand/residue is deleted.

### 13.55.16 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?

### 13.55.17 show-remarks-browser

`show-remarks-browser` [function]  
show the remarks browser

## 13.56 Library and Utility Functions

### 13.56.1 git-revision-count

`git-revision-count` [function]  
return the git revision count for for this build.

### 13.56.2 svn-revision

`svn-revision` [function]  
an alias to `git_revision_count()` for backwards compatibility

### 13.56.3 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 ( -> False in scheme) e.g. "/a/b/c.pdb" for "d/e/f.mtz  
FWT PHWT"

### 13.56.4 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

### 13.56.5 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.

### 13.56.6 coot-no-state-real-exit

`coot-no-state-real-exit` *retval* [function]  
Where *retval* is an integer number  
exit without writing a state file

### 13.56.7 coot-clear-backup-or-real-exit

`coot-clear-backup-or-real-exit` *retval* [function]  
Where *retval* is an integer number  
exit coot doing clear-backup maybe

### 13.56.8 coot-save-state-and-exit

`coot-save-state-and-exit` *retval save\_state\_flag* [function]

Where:

- *retval* is an integer number
- *save\_state\_flag* is an integer number

exit coot, write a state file

### 13.56.9 first-coords-imol

`first-coords-imol` [function]

What is the molecule number of first coordinates molecule?

return -1 when there is none.

### 13.56.10 first-small-coords-imol

`first-small-coords-imol` [function]

molecule number of first small (<400 atoms) molecule.

return -1 on no such molecule

### 13.56.11 first-unsaved-coords-imol

`first-unsaved-coords-imol` [function]

What is the molecule number of first unsaved coordinates molecule?

return -1 when there is none.

### 13.56.12 mmCIF-sfs-to-mtz

`mmCIF-sfs-to-mtz` *cif\_file\_name mtz\_file\_name* [function]

Where:

- *cif\_file\_name* is a string
- *mtz\_file\_name* is a string

convert the structure factors in *cif\_file\_name* to an mtz file.

Return 1 on success. Return 0 on a file without Rfree, return -1 on complete failure to write a file.

## 13.57 Graphics Utility Functions

### 13.57.1 set-do-anti-aliasing

`set-do-anti-aliasing` *state* [function]

Where *state* is an integer number

set the bond lines to be antialiased

### 13.57.2 do-anti-aliasing-state

`do-anti-aliasing-state` [function]

return the flag for antialiasing the bond lines

### 13.57.3 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

### 13.57.4 do-GL-lighting-state

**do-GL-lighting-state** [function]  
return the flag for GL lighting

### 13.57.5 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.

### 13.57.6 python-at-prompt-at-startup-state

**python-at-prompt-at-startup-state** [function]  
is the python interpreter at the prompt?  
Returns: 1 for yes, 0 for no.

### 13.57.7 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

### 13.57.8 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.

### 13.57.9 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)

### 13.57.10 toggle-idle-spin-function

`toggle-idle-spin-function` [function]  
Spin spin spin (or not)

### 13.57.11 toggle-idle-rock-function

`toggle-idle-rock-function` [function]  
Rock (not roll) (self-timed)

### 13.57.12 set-rocking-factors

`set-rocking-factors` *width\_scale frequency\_scale* [function]  
Where:

- *width\_scale* is a number
- *frequency\_scale* is a number

Settings for the inevitable discontents who dislike the default rocking rates (defaults 1 and 1)

### 13.57.13 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

### 13.57.14 idle-function-rotate-angle

`idle-function-rotate-angle` [function]  
what is the idle function rotation angle?

### 13.57.15 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)

### 13.57.16 allow-duplicate-sequence-numbers

`allow-duplicate-sequence-numbers` [function]  
enable reading PDB/pdbx files with duplicate sequence numbers

### 13.57.17 set-convert-to-v2-atom-names

`set-convert-to-v2-atom-names` *state* [function]  
Where *state* is an integer number  
shall we convert nucleotides to match the old dictionary names?  
Usually (after 2006 or so) we do not want to do this (given current Coot architecture).  
Coot should handle the residue synonyms transparently.  
default off (0).

### 13.57.18 handle-read-draw-molecule-with-recentre

`handle-read-draw-molecule-with-recentre` *filename* [function]  
*recentre\_on\_read\_pdb\_flag*

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.

### 13.57.19 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.

### 13.57.20 read-pdb

`read-pdb` *filename* [function]  
Where *filename* is a string

read coordinates from *filename*

### 13.57.21 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.

### 13.57.22 hetify-residue

`hetify-residue` *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

if this is not a standard group, then turn the atoms to HETATMs.

Return 1 on atoms changes, 0 on not. Return -1 if residue not found.

### 13.57.23 residue-has-hetatms

`residue-has-hetatms` *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

residue has HETATMs?

return 1 if all atoms of the specified residue are HETATMs, else, return 0. If residue not found, return -1.

### 13.57.24 het-group-n-atoms

`het-group-n-atoms` *comp\_id* [function]

Where *comp\_id* is a string

return the number of non-hydrogen atoms in the given het-group (comp-id).

Return -1 on comp-id not found in dictionary.

### 13.57.25 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*

### 13.57.26 copy-residue-range

`copy-residue-range` *imol\_target chain\_id\_target imol\_reference* [function]

*chain\_id\_reference resno\_range\_start resno\_range\_end*

Where:

- *imol\_target* is an integer number
- *chain\_id\_target* is a string
- *imol\_reference* is an integer number
- *chain\_id\_reference* is a string
- *resno\_range\_start* is an integer number
- *resno\_range\_end* is an integer number

copy the given residue range from the reference chain to the target chain

*resno\_range\_start* and *resno\_range\_end* are inclusive.

### 13.57.27 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 the given residues from the reference molecule to the target molecule

replace pdb. Fail if *molecule\_number* is not a valid model molecule. Return -1 on failure. Else return *molecule\_number*

### 13.57.28 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

### 13.57.29 check-for-dark-blue-density

`check-for-dark-blue-density` [function]

give a warning dialog if density it too dark (blue)

### 13.57.30 set-draw-solid-density-surface

`set-draw-solid-density-surface` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

sets the density map of the given molecule to be drawn as a (transparent) solid surface.

### 13.57.31 set-draw-map-standard-lines

`set-draw-map-standard-lines` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

toggle for standard lines representation of map.

This turns off/on standard lines representation of map. transparent surface is another representation type.

If you want to just turn off a map, don't use this, use

.

**13.57.32 set-solid-density-surface-opacity**

`set-solid-density-surface-opacity` *imol opacity* [function]

Where:

- *imol* is an integer number
- *opacity* is a number

set the opacity of density surface representation of the given map.

0.0 is totally transparent, 1.0 is completely opaque and (because the objects are no longer depth sorted) considerably faster to render. 0.3 is a reasonable number.

**13.57.33 set-flat-shading-for-solid-density-surface**

`set-flat-shading-for-solid-density-surface` *state* [function]

Where *state* is an integer number

set the flag to do flat shading rather than smooth shading for solid density surface.

Default is 1 (on).

**13.58 Interface Preferences****13.58.1 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

**13.58.2 scroll-by-wheel-mouse-state**

`scroll-by-wheel-mouse-state` [function]

return the internal state of the scroll-wheel map contouring

**13.58.3 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 initial contour for 2FoFc-style map

in sigma

**13.58.4 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 initial contour for FoFc-style map

in sigma

### 13.58.5 print-view-matrix

`print-view-matrix` [function]  
print the view matrix to the console, useful for molscript, perhaps

### 13.58.6 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`

### 13.58.7 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.

### 13.58.8 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.

### 13.58.9 apply-ncs-to-view-orientation-and-screen-centre

`apply-ncs-to-view-orientation-and-screen-centre` *imol current\_chain next\_ncs\_chain forward\_flag* [function]  
Where:

- *imol* is an integer number
- *current\_chain* is a string
- *next\_ncs\_chain* is a string
- *forward\_flag* is an integer number

as above, but shift the screen centre also.

**13.58.10 set-fps-flag**

`set-fps-flag` *t* [function]

Where *t* is an integer number  
set show frame-per-second flag

**13.58.11 get-fps-flag**

`get-fps-flag` [function]

set the state of show frames-per-second flag

**13.58.12 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.

**13.58.13 show-origin-marker-state**

`show-origin-marker-state` [function]

return the origin marker shown? state

**13.58.14 hide-modelling-toolbar**

`hide-modelling-toolbar` [function]

hide the vertical modelling toolbar in the GTK2 version

**13.58.15 show-modelling-toolbar**

`show-modelling-toolbar` [function]

show the vertical modelling toolbar in the GTK2 version (the toolbar is shown by default)

**13.58.16 hide-main-toolbar**

`hide-main-toolbar` [function]

hide the horizontal main toolbar in the GTK2 version

**13.58.17 show-main-toolbar**

`show-main-toolbar` [function]

show the horizontal main toolbar in the GTK2 version (the toolbar is shown by default)

**13.58.18 show-model-toolbar-all-icons**

`show-model-toolbar-all-icons` [function]

show all available icons in the modelling toolbar (same as MFR dialog)

**13.58.19 show-model-toolbar-main-icons**

`show-model-toolbar-main-icons` [function]  
show only a selection of icons in the modelling toolbar

**13.58.20 reattach-modelling-toolbar**

`reattach-modelling-toolbar` [function]  
reattach the modelling toolbar to the last attached position

**13.58.21 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

**13.58.22 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

**13.58.23 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).

**13.58.24 set-model-fit-refine-dialog-stays-on-top**

`set-model-fit-refine-dialog-stays-on-top` *istate* [function]  
Where *istate* is an integer number  
model-fit-refine dialog stays on top

**13.58.25 model-fit-refine-dialog-stays-on-top-state**

`model-fit-refine-dialog-stays-on-top-state` [function]  
return the state model-fit-refine dialog stays on top

**13.58.26 accept-reject-dialog-docked-state**

`accept-reject-dialog-docked-state` [function]  
the accept/reject dialog docked state

**13.58.27 set-accept-reject-dialog-docked-show**

`set-accept-reject-dialog-docked-show` *state* [function]  
Where *state* is an integer number  
set the accept/reject dialog docked show state

**13.58.28 accept-reject-dialog-docked-show-state**

`accept-reject-dialog-docked-show-state` [function]  
 what is the accept/reject dialog docked show state?

**13.59 Mouse Buttons****13.59.1 set-control-key-for-rotate**

`set-control-key-for-rotate` *state* [function]  
 Where *state* is an integer number  
 Alternate mode for rotation.  
 Preferred by some, including Dirk Kostrewa. I don't think this mode works properly yet

**13.59.2 control-key-for-rotate-state**

`control-key-for-rotate-state` [function]  
 return the control key rotate state

**13.59.3 blob-under-pointer-to-screen-centre**

`blob-under-pointer-to-screen-centre` [function]  
 Put the blob under the cursor to the screen centre. Check only positive blobs. Useful function if bound to a key.  
 The refinement map must be set. (We can't check all maps because they are not (or may not be) on the same scale).  
 Returns: 1 if successfully found a blob and moved there. return 0 if no move.

**13.60 Cursor Function****13.60.1 normal-cursor**

`normal-cursor` [function]  
 normal cursor

**13.60.2 fleur-cursor**

`fleur-cursor` [function]  
 fleur cursor

**13.60.3 pick-cursor-maybe**

`pick-cursor-maybe` [function]  
 pick cursor maybe

**13.60.4 rotate-cursor**

`rotate-cursor` [function]  
 rotate cursor

### 13.60.5 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

## 13.61 Model/Fit/Refine Functions

### 13.61.1 post-model-fit-refine-dialog

`post-model-fit-refine-dialog` [function]

display the Model/Fit/Refine dialog

### 13.61.2 unset-model-fit-refine-dialog

`unset-model-fit-refine-dialog` [function]

unset model/fit/refine dialog

### 13.61.3 unset-refine-params-dialog

`unset-refine-params-dialog` [function]

unset refine params dialog

### 13.61.4 show-select-map-dialog

`show-select-map-dialog` [function]

display the Display Manager dialog

### 13.61.5 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".

### 13.61.6 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".

### 13.61.7 post-other-modelling-tools-dialog

`post-other-modelling-tools-dialog` [function]

display the Other Modelling Tools dialog

### 13.61.8 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)

### 13.61.9 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?"

## 13.62 Backup Functions

### 13.62.1 make-backup

`make-backup` *imol* [function]  
Where *imol* is an integer number  
make backup for molecule number *imol*

### 13.62.2 turn-off-backup

`turn-off-backup` *imol* [function]  
Where *imol* is an integer number  
turn off backups for molecule number *imol*

### 13.62.3 turn-on-backup

`turn-on-backup` *imol* [function]  
Where *imol* is an integer number  
turn on backups for molecule number *imol*

### 13.62.4 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

### 13.62.5 apply-undo

`apply-undo` [function]  
apply undo - the "Undo" button callback

### 13.62.6 apply-redo

`apply-redo` [function]  
apply redo - the "Redo" button callback

### 13.62.7 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

### 13.62.8 have-unsaved-changes-p

`have-unsaved-changes-p` *imol* [function]

Where *imol* is an integer number

does molecule number *imol* have unsaved changes?

Returns: -1 on bad *imol*, 0 on no unsaved changes, 1 on has unsaved changes

### 13.62.9 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*

### 13.62.10 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.

### 13.62.11 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

### 13.62.12 unpathed-backup-file-names-state

`unpathed-backup-file-names-state` [function]

return the state for adding paths to backup file names

### 13.62.13 backup-compress-files-state

`backup-compress-files-state` [function]

return the state for compression of backup files

### 13.62.14 set-backup-compress-files

`set-backup-compress-files` *state* [function]

Where *state* is an integer number

set if backup files will be compressed or not using gzip

## 13.63 Recover Session Function

### 13.63.1 recover-session

`recover-session` [function]  
 recover session

After a crash, 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 than the read file. (Not very good, because you need to remember which files you read in before the crash - should be improved.)

## 13.64 Map Functions

### 13.64.1 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()`

### 13.64.2 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.

### 13.64.3 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.

### 13.64.4 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()`.

### 13.64.5 set-scrollable-map

`set-scrollable-map` *imol* [function]

Where *imol* is an integer number

return the molecule number to which the mouse scroll wheel is attached

set the map that has its contour level changed by the scrolling the mouse wheel to molecule number *imol* (same as  
).

### 13.64.6 scroll-wheel-map

`scroll-wheel-map` [function]

the contouring of which map is altered when the scroll wheel changes?

### 13.64.7 save-previous-map-colour

`save-previous-map-colour` *imol* [function]

Where *imol* is an integer number

save previous colour map for molecule number *imol*

### 13.64.8 restore-previous-map-colour

`restore-previous-map-colour` *imol* [function]

Where *imol* is an integer number

restore previous colour map for molecule number *imol*

### 13.64.9 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.

### 13.64.10 get-active-map-drag-flag

`get-active-map-drag-flag` [function]

return the state of the dragged map flag

### 13.64.11 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

### 13.64.12 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 *imol*th map

### 13.64.13 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

### 13.64.14 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)

### 13.64.15 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 Å (eg. 2.0)

### 13.64.16 model-resolution

`model-resolution` *imol* [function]

Where *imol* is an integer number

return the resolution set in the header of the model/coordinates file. If this number is not available, return a number less than 0.

### 13.64.17 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.

### 13.64.18 export-map-fragment

`export-map-fragment` *imol x y z radius filename* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number
- *radius* is a number
- *filename* is a string

export a fragment of the map about (x,y,z)

### 13.64.19 export-map-fragment-with-origin-shift

`export-map-fragment-with-origin-shift` *imol x y z radius filename* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number
- *radius* is a number
- *filename* is a string

export a fragment of the map about (x,y,z)

### 13.64.20 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  $\text{map\_scale} * \text{imap2}$  from *imap1*, on the grid of *imap1*. Return the new molecule number. Return -1 on failure.

### 13.64.21 reinterp-map

`reinterp-map` *map\_no reference\_map\_no* [function]

Where:

- *map\_no* is an integer number
- *reference\_map\_no* is an integer number

make a new map (a copy of *map\_no*) that is in the cell, spacegroup and gridding of the map in *reference\_map\_no*.

Return the new map molecule number - return -1 on failure

### 13.64.22 smooth-map

`smooth-map` *map\_no sampling\_multiplier* [function]

Where:

- *map\_no* is an integer number
- *sampling\_multiplier* is a number

make a new map (a copy of *map\_no*) that is in the cell, spacegroup and a multiple of the sampling of the input map (a sampling factor of more than 1 makes the output maps smoother)

## 13.65 Density Increment

### 13.65.1 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

### 13.65.2 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

### 13.65.3 get-diff-map-iso-level-increment

`get-diff-map-iso-level-increment` [function]  
return difference maps iso-map level increment

### 13.65.4 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

set the difference maps iso-map level increment

### 13.65.5 set-map-sampling-rate-text

`set-map-sampling-rate-text` *text* [function]  
Where *text* is a string  
sampling rate  
find the molecule for which the single map dialog applies and set the contour level and redraw

### 13.65.6 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.

### 13.65.7 get-map-sampling-rate

`get-map-sampling-rate` [function]  
return the map sampling rate

### 13.65.8 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.

### 13.65.9 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

### 13.65.10 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

### 13.65.11 set-stop-scroll-diff-map

`set-stop-scroll-diff-map` *i* [function]

Where *i* is an integer number

create a lower limit to the "Fo-Fc-style" map contour level changing

(default 1 on)

### 13.65.12 set-stop-scroll-iso-map

`set-stop-scroll-iso-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)

### 13.65.13 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)

### 13.65.14 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)

### 13.65.15 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

## 13.66 Density Functions

### 13.66.1 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*

### 13.66.2 map-line-width-state

`map-line-width-state` [function]  
 return the width in which density contours are drawn

### 13.66.3 make-and-draw-map

`make-and-draw-map` *mtz\_file\_name f\_col phi\_col weight use\_weights* [function]  
*is\_diff\_map*

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

### 13.66.4 make-and-draw-map-with-refmac-params

`make-and-draw-map-with-refmac-params` *mtz\_file\_name a b weight* [function]  
*use\_weights is\_diff\_map have\_refmac\_params fobs\_col sigfobs\_col r\_free\_col*  
*sensible\_f\_free\_col*

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, except 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

### 13.66.5 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.

### 13.66.6 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 columns that we want it to have?

### 13.66.7 mtz-file-has-phases-p

`mtz-file-has-phases-p` *mtz\_file\_name* [function]

Where *mtz\_file\_name* is a string

does the mtz file have phases?

### 13.66.8 is-mtz-file-p

`is-mtz-file-p filename` [function]  
Where *filename* is a string  
is the given filename an mtz file?

### 13.66.9 cns-file-has-phases-p

`cns-file-has-phases-p cns_file_name` [function]  
Where *cns\_file\_name* is a string  
does the given file have cns phases?

### 13.66.10 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.

### 13.66.11 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 expected MTZ columns for Auto-reading MTZ file.  
Not every program uses the default reftac labels ("FWT"/"PHWT") for its MTZ file. Here we can tell coot to expect other labels so that coot can "Auto-open" such MTZ files.  
e.g. (set-auto-read-column-labels "2FOFCWT" "PH2FOFCWT" 0)

### 13.66.12 set-map-radius

`set-map-radius f` [function]  
Where *f* is a number  
set the extent of the box/radius of electron density contours

### 13.66.13 set-density-size

`set-density-size f` [function]  
Where *f* is a number  
another (old) way of setting the radius of the map

### 13.66.14 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.

### 13.66.15 get-map-radius

`get-map-radius` [function]  
return the extent of the box/radius of electron density contours

### 13.66.16 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).

### 13.66.17 esoteric-depth-cue-state

`esoteric-depth-cue-state` [function]  
native depth cueing system  
return the state of the esoteric depth cueing flag

### 13.66.18 set-swap-difference-map-colours

`set-swap-difference-map-colours` *i* [function]  
Where *i* is an integer number  
not everone likes coot's default difference map colouring.  
Pass an argument *i*=1 to swap the difference map colouring so that red is positive and green is negative.

### 13.66.19 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)

### 13.66.20 map-is-difference-map

`map-is-difference-map` *imol* [function]  
Where *imol* is an integer number  
map is difference map?

### 13.66.21 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

### 13.66.22 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

### 13.66.23 residue-density-fit-scale-factor

`residue-density-fit-scale-factor` [function]

return the scale factor for the Residue Density fit analysis

### 13.66.24 density-at-point

`density-at-point` *imol\_map* *x* *y* *z* [function]

Where:

- *imol\_map* 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

## 13.67 Parameters from map

### 13.67.1 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).

### 13.67.2 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).

Caller should dispose of returned pointer.

### 13.67.3 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). Caller should dispose of returned pointer.

### 13.67.4 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. Caller should dispose of returned pointer.

### 13.67.5 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.

## 13.68 PDB Functions

### 13.68.1 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*

### 13.68.2 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*

### 13.68.3 quick-save

`quick-save` [function]  
save all modified coordinates molecules to the default names and save the state too.

## 13.69 Info Dialog

### 13.69.1 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).

### 13.69.2 info-dialog-and-text

`info-dialog-and-text txt` [function]  
Where *txt* is a string  
create a dialog with information and print to console  
as `info_dialog` but print to console as well.

## 13.70 Refmac Functions

### 13.70.1 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

### 13.70.2 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

### 13.70.3 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

### 13.70.4 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"

## 13.71 Symmetry Functions

### 13.71.1 set-symmetry-size

`set-symmetry-size` *f* [function]  
Where *f* is a number  
set the size of the displayed symmetry

### 13.71.2 get-show-symmetry

`get-show-symmetry` [function]  
is symmetry master display control on?

### 13.71.3 set-show-symmetry-master

`set-show-symmetry-master` *state* [function]  
Where *state* is an integer number  
set display symmetry, master controller

### 13.71.4 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

### 13.71.5 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

### 13.71.6 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

### 13.71.7 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*

### 13.71.8 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

### 13.71.9 set-symmetry-colour-by-symop

`set-symmetry-colour-by-symop imol state` [function]  
Where:

- *imol* is an integer number
- *state* is an integer number

set symmetry colour by symop mode

### 13.71.10 set-symmetry-whole-chain

`set-symmetry-whole-chain imol state` [function]  
Where:

- *imol* is an integer number
- *state* is an integer number

set symmetry colour for the chain

### 13.71.11 set-symmetry-atom-labels-expanded

`set-symmetry-atom-labels-expanded` *state* [function]

Where *state* is an integer number

set use expanded symmetry atom labels

### 13.71.12 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"

### 13.71.13 undo-symmetry-view

`undo-symmetry-view` [function]

Undo symmetry view. Translate back to main molecule from this symmetry position.

### 13.71.14 first-molecule-with-symmetry-displayed

`first-molecule-with-symmetry-displayed` [function]

return the molecule number.

Returns: -1 if there is no molecule with symmetry displayed.

### 13.71.15 save-symmetry-coords

`save-symmetry-coords` *imol filename symop\_no shift\_a shift\_b shift\_c* [function]

*pre\_shift\_to\_origin\_na pre\_shift\_to\_origin\_nb pre\_shift\_to\_origin\_nc*

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 applied (this is how symmetry works in Coot internals).

**13.71.16 new-molecule-by-symmetry**

`new-molecule-by-symmetry` *imol name m11 m12 m13 m21 m22 m23* [function]  
*m31 m32 m33 tx ty tz pre\_shift\_to\_origin\_na pre\_shift\_to\_origin\_nb*  
*pre\_shift\_to\_origin\_nc*

Where:

- *imol* is an integer number
- *name* 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
- *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 coordinates.

Allow a shift of the coordinates to the origin before symmetry expansion is applied.

Pass "" as the name-in and a name will be constructed for you.

Return -1 on failure.

**13.71.17 new-molecule-by-symmetry-with-atom-selection**

`new-molecule-by-symmetry-with-atom-selection` *imol name* [function]  
*mmdb\_atom\_selection\_string 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*

Where:

- *imol* is an integer number
- *name* is a string
- *mmdb\_atom\_selection\_string* 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
- *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*, but only for atom that match the *mmdb\_atom\_selection\_string*.

The rotation/translation matrix components are given in coordinates.

Allow a shift of the coordinates to the origin before symmetry expansion is applied.

Pass "" as the name-in and a name will be constructed for you.

Return -1 on failure.

### 13.71.18 new-molecule-by-sympop

`new-molecule-by-sympop imol sympop_string pre_shift_to_origin_na` [function]  
`pre_shift_to_origin_nb pre_shift_to_origin_nc`

Where:

- *imol* is an integer number
- *sympop\_string* is a string
- *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*.

### 13.71.19 n-sympops

`n-sympops imol` [function]

Where *imol* is an integer number

return the number of symmetry operators for the given molecule

return -1 on no-symmetry for molecule or inappropriate *imol* number

### 13.71.20 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 initiated with a HM symbol or a symm list for clipper. Return the success status of the setting.

### 13.71.21 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.

## 13.72 History Functions

### 13.72.1 print-all-history-in-scheme

`print-all-history-in-scheme` [function]

print the history in scheme format

### 13.72.2 print-all-history-in-python

`print-all-history-in-python` [function]

print the history in python format

### 13.72.3 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.

### 13.72.4 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

set a flag to show the text command equivalent of gui commands in the console as they happen in bold and colours.

*colour\_flag*: pass 1 for on, 0 for off.

*colour\_index* 0 to 7 inclusive for various different colourings.

## 13.73 State Functions

### 13.73.1 save-state

**save-state** [function]  
save the current state to the default filename

### 13.73.2 save-state-file

**save-state-file** *filename* [function]  
Where *filename* is a string  
save the current state to file filename

### 13.73.3 save-state-file-py

**save-state-file-py** *filename* [function]  
Where *filename* is a string  
save the current state to file filename

### 13.73.4 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)

### 13.73.5 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

### 13.73.6 run-state-file

**run-state-file** [function]  
run the state file (reading from default filename)

### 13.73.7 run-state-file-maybe

**run-state-file-maybe** [function]  
run the state file depending on the state variables

## 13.74 The Virtual Trackball

### 13.74.1 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"

### 13.74.2 vt-surface-status

`vt-surface-status` [function]

return the mouse view status mode

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

## 13.75 Clipping Functions

### 13.75.1 set-clipping-back

`set-clipping-back` *v* [function]

Where *v* is a number

set clipping plane back

### 13.75.2 set-clipping-front

`set-clipping-front` *v* [function]

Where *v* is a number

set clipping plane front

## 13.76 Unit Cell interface

### 13.76.1 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

### 13.76.2 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

### 13.76.3 set-show-unit-cell

`set-show-unit-cell` *imol istrate* [function]

Where:

- *imol* is an integer number
- *istrate* is an integer number

set the state of show unit cell for the particular molecule number *imol*  
1 for displayed 0 for undisplayed

## 13.77 Colour

### 13.77.1 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

### 13.77.2 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?

### 13.77.3 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?

### 13.77.4 set-colour-by-chain

`set-colour-by-chain` *imol* [function]

Where *imol* is an integer number

colour molecule number *imol* by chain type

### 13.77.5 set-colour-by-molecule

`set-colour-by-molecule` *imol* [function]

Where *imol* is an integer number

colour molecule number *imol* by molecule

### 13.77.6 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

## 13.78 Map colour

### 13.78.1 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.

### 13.78.2 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

### 13.78.3 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*.

## 13.79 Anisotropic Atoms Interface

### 13.79.1 get-limit-aniso

`get-limit-aniso` [function]

get the aniso radius limit

### 13.79.2 get-show-limit-aniso

`get-show-limit-aniso` [function]

get show the aniso limit

### 13.79.3 get-show-aniso

`get-show-aniso` [function]

return show-aniso-atoms state

### 13.79.4 set-limit-aniso

`set-limit-aniso` *state* [function]

Where *state* is an integer number

set the aniso atom limit

### 13.79.5 set-show-aniso

`set-show-aniso` *state* [function]  
Where *state* is an integer number  
set show aniso atoms

### 13.79.6 set-aniso-probability

`set-aniso-probability` *f* [function]  
Where *f* is a number  
set aniso probability

### 13.79.7 get-aniso-probability

`get-aniso-probability` [function]  
get aniso probability

## 13.80 Display Functions

### 13.80.1 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

### 13.80.2 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 graphics window position

### 13.80.3 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

store the graphics window position

### 13.80.4 graphics-draw

`graphics-draw` [function]  
draw a frame

**13.80.5 zalman-stereo-mode**

`zalman-stereo-mode` [function]  
 try to turn on Zalman stereo mode

**13.80.6 hardware-stereo-mode**

`hardware-stereo-mode` [function]  
 try to turn on stereo mode

**13.80.7 stereo-mode-state**

`stereo-mode-state` [function]  
 what is the stereo state?  
 Returns: 1 for in hardware stereo, 2 for side by side stereo, else return 0.

**13.80.8 mono-mode**

`mono-mode` [function]  
 try to turn on mono mode

**13.80.9 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 by side stereo mode

**13.80.10 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?

**13.80.11 hardware-stereo-angle-factor-state**

`hardware-stereo-angle-factor-state` [function]  
 return the hardware stereo angle factor

**13.80.12 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

### 13.80.13 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

### 13.80.14 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

### 13.80.15 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

### 13.80.16 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

### 13.80.17 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

### 13.80.18 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

### 13.80.19 set-edit-chi-angles-dialog-position

`set-edit-chi-angles-dialog-position` *x\_pos* *y\_pos* [function]

Where:

- *x\_pos* is an integer number
- *y\_pos* is an integer number

set edit chi angles dialog position

### 13.80.20 set-rotamer-selection-dialog-position

`set-rotamer-selection-dialog-position` *x\_pos* *y\_pos* [function]

Where:

- *x\_pos* is an integer number
- *y\_pos* is an integer number

set rotamer selection dialog position

## 13.81 Smooth Scrolling

### 13.81.1 set-smooth-scroll-flag

`set-smooth-scroll-flag` *v* [function]

Where *v* is an integer number

set smooth scrolling

### 13.81.2 get-smooth-scroll

`get-smooth-scroll` [function]

return the smooth scrolling state

### 13.81.3 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).

### 13.81.4 set-smooth-scroll-limit

`set-smooth-scroll-limit` *lim* [function]

Where *lim* is a number

do not scroll for distances greater this limit

## 13.82 Font Parameters

### 13.82.1 set-font-size

`set-font-size` *i* [function]  
Where *i* is an integer number  
set the font size

### 13.82.2 get-font-size

`get-font-size` [function]  
return the font size  
Returns: 1 (small) 2 (medium, default) 3 (large)

### 13.82.3 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

### 13.82.4 set-use-stroke-characters

`set-use-stroke-characters` *state* [function]  
Where *state* is an integer number  
set use stroke characters

## 13.83 Rotation Centre

### 13.83.1 set-rotation-centre-size

`set-rotation-centre-size` *f* [function]  
Where *f* is a number  
set rotation centre marker size

### 13.83.2 recentre-on-read-pdb

`recentre-on-read-pdb` [function]  
return the recentre-on-pdb state

### 13.83.3 set-recentre-on-read-pdb

`set-recentre-on-read-pdb` *int* [function]  
Where *int* is a short  
set the recentre-on-pdb state

### 13.83.4 set-rotation-centre

`set-rotation-centre` *x y z* [function]

Where:

- *x* is a number
- *y* is a number
- *z* is a number

set the rotation centre

### 13.83.5 go-to-ligand

`go-to-ligand` [function]

centre on the ligand of the "active molecule", if we are already there, centre on the next hetgroup (etc)

### 13.83.6 set-go-to-ligand-n-atoms-limit

`set-go-to-ligand-n-atoms-limit` *n\_atom\_min* [function]

Where *n\_atom\_min* is an integer number

go to the ligand that has more than *n\_atom\_min* atoms

## 13.84 Atom Selection Utilities

### 13.84.1 median-temperature-factor

`median-temperature-factor` *imol* [function]

Where *imol* is an integer number

return the median temperature factor for *imol*

### 13.84.2 average-temperature-factor

`average-temperature-factor` *imol* [function]

Where *imol* is an integer number

return the average temperature factor for the atoms in *imol*

### 13.84.3 standard-deviation-temperature-factor

`standard-deviation-temperature-factor` *imol* [function]

Where *imol* is an integer number

return the standard deviation of the atom temperature factors for *imol*

### 13.84.4 clear-pending-picks

`clear-pending-picks` [function]

clear pending picks (stop coot thinking that the user is about to pick an atom).

**13.84.5 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)

**13.84.6 default-new-atoms-b-factor**

`default-new-atoms-b-factor` [function]

return the default temperature factor for newly created atoms

**13.84.7 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)

**13.84.8 get-reset-b-factor-moved-atoms-state**

`get-reset-b-factor-moved-atoms-state` [function]

return the state if temperature factors should be reset for moved atoms

**13.84.9 set-atom-attribute**

`set-atom-attribute` *imol chain\_id resno ins\_code atom\_name alt\_conf* [function]  
*attribute\_name val*

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 numerical attribute to the atom with the given specifier.

Attributes can be "x", "y", "z", "B", "occ" and the attribute val is a floating point number

**13.84.10 set-atom-string-attribute**

`set-atom-string-attribute` *imol chain\_id resno ins\_code atom\_name* [function]  
*alt\_conf attribute\_name attribute\_value*

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 attribute to the atom with the given specifier.

Attributes can be "atom-name", "alt-conf", "element" or "segid".

### 13.84.11 set-residue-name

**set-residue-name** *imol chain\_id res\_no ins\_code new\_residue\_name* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string
- *new\_residue\_name* is a string

set lots of atom attributes at once by-passing the rebonding and redrawing of the above 2 functions

set the residue name of the specified residue

## 13.85 Skeletonization Interface

### 13.85.1 skeletonize-map

**skeletonize-map** *imol prune\_flag* [function]

Where:

- *imol* is an integer number
- *prune\_flag* is an integer number

skeletonize molecule number *imol*

the *prune\_flag* should almost always be 0.

NOTE:: The arguments to have been reversed for coot 0.8.3 and later (now the molecule number comes first).

### 13.85.2 unskeletonize-map

**unskeletonize-map** *imol* [function]

Where *imol* is an integer number

undisplay the skeleton on molecule number *imol*

### 13.85.3 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)

### 13.85.4 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

## 13.86 Skeleton Colour

### 13.86.1 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

set the skeleton colour

## 13.87 Read Maps

### 13.87.1 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).

## 13.88 Save Coordinates

### 13.88.1 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.

### 13.88.2 set-save-coordinates-in-original-directory

`set-save-coordinates-in-original-directory` *i* [function]  
Where *i* is an integer number  
set save coordinates in the starting directory

## 13.89 Read Phases File Functions

### 13.89.1 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.

### 13.89.2 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

### 13.89.3 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).

### 13.89.4 read-phs-and-make-map-using-cell-symm

`read-phs-and-make-map-using-cell-symm` *phs\_file\_name* [function]  
*hm\_spacegroup a b c alpha beta gamma*  
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

### 13.89.5 read-phs-and-make-map-with-reso-limits

`read-phs-and-make-map-with-reso-limits` *imol phs\_file\_name* [function]  
*reso\_lim\_low reso\_lim\_high*

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 Angstroms).

## 13.90 Graphics Move

### 13.90.1 undo-last-move

`undo-last-move` [function]  
undo last move

### 13.90.2 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

### 13.90.3 transform-molecule-by

`transform-molecule-by` *imol m11 m12 m13 m21 m22 m23 m31 m32 m33* [function]  
*x y z*

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

#### 13.90.4 transform-zone

`transform-zone` *imol chain\_id resno\_start resno\_end ins\_code m11 m12* [function]  
*m13 m21 m22 m23 m31 m32 m33 x y z*

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
- *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 fragment of molecule number *imol* by the given rotation matrix, then translate by (x,y,z) in Angstroms

## 13.91 Go To Atom Widget Functions

### 13.91.1 post-go-to-atom-window

`post-go-to-atom-window` [function]  
Post the Go To Atom Window.

### 13.91.2 go-to-atom-molecule-number

`go-to-atom-molecule-number` [function]  
the go-to-atom molecule number

### 13.91.3 go-to-atom-chain-id

`go-to-atom-chain-id` [function]  
the go-to-atom chain-id

### 13.91.4 go-to-atom-atom-name

`go-to-atom-atom-name` [function]  
the go-to-atom atom name

### 13.91.5 go-to-atom-residue-number

`go-to-atom-residue-number` [function]  
the go-to-atom residue number

### 13.91.6 go-to-atom-ins-code

`go-to-atom-ins-code` [function]  
the go-to-atom insertion code

### 13.91.7 go-to-atom-alt-conf

`go-to-atom-alt-conf` [function]  
the go-to-atom alt conf

### 13.91.8 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")

Returns: the success status of the go to. 0 for fail, 1 for success.

### 13.91.9 set-go-to-atom-chain-residue-atom-name-full

`set-go-to-atom-chain-residue-atom-name-full` *chain\_id resno* [function]  
*ins\_code atom\_name alt\_conf*

Where:

- *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

set the go to (full) 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")

Returns: the success status of the go to. 0 for fail, 1 for success.

### 13.91.10 set-go-to-atom-chain-residue-atom-name-no-redraw

`set-go-to-atom-chain-residue-atom-name-no-redraw` *t1 iresno t3* [function]  
*make\_the\_move\_flag*

Where:

- *t1* is a string
- *iresno* is an integer number
- *t3* is a string
- *make\_the\_move\_flag* is an integer number

set go to atom but don't redraw

### 13.91.11 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.

### 13.91.12 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?

**13.91.13 full-atom-spec-to-atom-index**

`full-atom-spec-to-atom-index` *imol chain resno inscode atom\_name* [function]  
*altloc*

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?

**13.91.14 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

**13.91.15 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.

**13.91.16 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 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.

20090914-PE `set-go-to-atom-molecule` can be used in a script and it should change the `go-to-atom-molecule` in the Go To Atom dialog (if it is being displayed). This does mean, of course that using the ramachandran plot to centre on atoms will change the Go To Atom dialog. Maybe that is surprising (maybe not).

**13.92 Map and Molecule Control****13.92.1 post-display-control-window**

`post-display-control-window` [function]  
 display the Display Control window

### 13.92.2 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

### 13.92.3 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

### 13.92.4 set-display-only-model-mol

`set-display-only-model-mol` *imol* [function]

Where *imol* is an integer number

from all the model molecules, display only *imol*

This stops flashing/delayed animations with many molecules

### 13.92.5 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/inactive (clickable), 0 for off, 1 for on

### 13.92.6 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

### 13.92.7 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

### 13.92.8 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

### 13.92.9 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

### 13.92.10 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.

### 13.92.11 show-spacegroup

`show-spacegroup` *imol* [function]

Where *imol* is an integer number

return the spacegroup of molecule number *imol* . Deprecated.

Returns: "No Spacegroup" when the spacegroup of a molecule has not been set.

## 13.93 Align and Mutate

### 13.93.1 align-and-mutate

`align-and-mutate` *imol chain\_id fasta\_maybe renumber\_residues\_flag* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *fasta\_maybe* is a string
- *renumber\_residues\_flag* is an integer number

align and mutate the given chain to the given sequence

### 13.93.2 set-alignment-gap-and-space-penalty

`set-alignment-gap-and-space-penalty` *wgap wspace* [function]

Where:

- *wgap* is a number
- *wspace* is a number

set the penalty for affine gap and space when aligning, defaults -3.0 and -0.4

## 13.94 Renumber Residue Range

### 13.94.1 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

### 13.94.2 change-residue-number

`change-residue-number` *imol chain\_id current\_resno current\_inscore* [function]  
*new\_resno new\_inscore*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *current\_resno* is an integer number
- *current\_inscore* is a string
- *new\_resno* is an integer number
- *new\_inscore* is a string

change chain id, residue number or insertion code for given residue

## 13.95 Scripting Interface

### 13.95.1 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

### 13.95.2 post-scripting-window

`post-scripting-window` [function]

do nothing - compatibility function

### 13.95.3 post-scheme-scripting-window

`post-scheme-scripting-window` [function]

pop-up a scripting window for scheming

### 13.95.4 post-python-scripting-window

`post-python-scripting-window` [function]  
pop-up a scripting window for pythoning

## 13.96 Monomer

### 13.96.1 get-coords-for-accession-code

`get-coords-for-accession-code` *code* [function]  
Where *code* is a string  
if possible, read in the new coords getting coords via web.  
(no return value because get-url-str does not return one).

### 13.96.2 run-script

`run-script` *filename* [function]  
Where *filename* is a string  
run script file

### 13.96.3 run-guile-script

`run-guile-script` *filename* [function]  
Where *filename* is a string  
guile run script file

### 13.96.4 run-python-script

`run-python-script` *filename* [function]  
Where *filename* is a string  
run python script file

### 13.96.5 import-python-module

`import-python-module` *module\_name use\_namespace* [function]  
Where:  

- *module\_name* is a string
- *use\_namespace* is an integer number

import python module

## 13.97 Regularization and Refinement

### 13.97.1 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 editing the `mon_lib_list.cif` file.

### 13.97.2 remove-planar-peptide-restraints

`remove-planar-peptide-restraints` [function]  
remove restraints on peptides to make them planar.

### 13.97.3 set-use-trans-peptide-restraints

`set-use-trans-peptide-restraints` *on\_off\_state* [function]  
Where *on\_off\_state* is an integer number  
add a restraint on peptides to keep trans peptides trans  
i.e. omega in trans-peptides is restraints to 180 degrees.

### 13.97.4 add-omega-torsion-restraints

`add-omega-torsion-restraints` [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).

### 13.97.5 remove-omega-torsion-restraints

`remove-omega-torsion-restraints` [function]  
remove omega restraints on CIS and TRANS linked residues.

### 13.97.6 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

### 13.97.7 refinement-immediate-replacement-state

`refinement-immediate-replacement-state` [function]  
query the state of the immediate replacement mode

### 13.97.8 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.

### 13.97.9 accept-regularization

`accept-regularization` [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`.

### 13.97.10 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.

### 13.97.11 refine-with-torsion-restraints-state

`refine-with-torsion-restraints-state` [function]

return the state of above

### 13.97.12 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).

### 13.97.13 matrix-state

`matrix-state` [function]

return the relative weight of the geometric terms to the map terms.

### 13.97.14 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 allow 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

### 13.97.15 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

### 13.97.16 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

### 13.97.17 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

### 13.97.18 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

### 13.97.19 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

Returns: a status, whether the regularisation was done or not. 0 for no, 1 for yes.

### 13.97.20 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: 80.

### 13.97.21 dragged-refinement-steps-per-frame

`dragged-refinement-steps-per-frame` [function]

return the number of steps per frame in dragged refinement

### 13.97.22 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.

### 13.97.23 refinement-refine-per-frame-state

`refinement-refine-per-frame-state` [function]

query the state of the above option

### 13.97.24 set-refinement-drag-elasticity

`set-refinement-drag-elasticity` *e* [function]

Where *e* is a number

the elasticity of the dragged atom in refinement mode.

Default 0.33

Bigger numbers mean bigger movement of the other atoms.

### 13.97.25 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  
name consistent with  
!?

### 13.97.26 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.

### 13.97.27 check-chiral-volumes

`check-chiral-volumes` *imol* [function]  
Where *imol* is an integer number  
query the state of the above option

### 13.97.28 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.

### 13.97.29 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

### 13.97.30 secondary-structure-restraints-type

`secondary-structure-restraints-type` [function]  
return the secondary structure restraints type

### 13.97.31 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.

### 13.97.32 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

### 13.97.33 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.

### 13.97.34 add-extra-bond-restraint

`add-extra-bond-restraint` *imol chain\_id\_1 res\_no\_1 ins\_code\_1* [function]  
*atom\_name\_1 alt\_conf\_1 chain\_id\_2 res\_no\_2 ins\_code\_2 atom\_name\_2*  
*alt\_conf\_2 bond\_dist esd*

Where:

- *imol* is an integer number
- *chain\_id\_1* is a string
- *res\_no\_1* is an integer number
- *ins\_code\_1* is a string
- *atom\_name\_1* is a string
- *alt\_conf\_1* is a string
- *chain\_id\_2* is a string
- *res\_no\_2* is an integer number
- *ins\_code\_2* is a string
- *atom\_name\_2* is a string
- *alt\_conf\_2* is a string
- *bond\_dist* is a number
- *esd* is a number

add a user-define bond restraint

this extra restraint is used when the given atoms are selected in refinement or regularization.

Returns: the index of the new restraint.

Returns: -1 when the atoms were not found and no extra bond restraint was stored.

### 13.97.35 delete-all-extra-restraints

`delete-all-extra-restraints` *imol* [function]

Where *imol* is an integer number

clear out all the extra/user-defined restraints for molecule number *imol*

### 13.97.36 delete-extra-restraints-for-residue

`delete-extra-restraints-for-residue` *imol chain\_id res\_no ins\_code* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string

clear out all the extra/user-defined restraints for this residue in molecule number *imol*

### 13.97.37 set-extra-restraints-prosmart-sigma-limits

`set-extra-restraints-prosmart-sigma-limits` *imol limit\_high limit\_low* [function]

Where:

- *imol* is an integer number
- *limit\_high* is a number
- *limit\_low* is a number

often we don't want to see all prosmart restraints, just the (big) violations

### 13.97.38 write-interpolated-extra-restraints

`write-interpolated-extra-restraints` *imol\_1 imol\_2 n\_steps file\_name\_stub* [function]

Where:

- *imol\_1* is an integer number
- *imol\_2* is an integer number
- *n\_steps* is an integer number
- *file\_name\_stub* is a string

proSMART interpolated restraints for model morphing

### 13.97.39 write-interpolated-models-and-extra-restraints

`write-interpolated-models-and-extra-restraints` *imol\_1 imol\_2 n\_steps file\_name\_stub interpolation\_mode* [function]

Where:

- *imol\_1* is an integer number
- *imol\_2* is an integer number

- *n\_steps* is an integer number
- *file\_name\_stub* is a string
- *interpolation\_mode* is an integer number

proSMART interpolated restraints for model morphing and write interpolated model interpolation\_mode is currently dummy - in due course I will add torion angle interpolation.

### 13.97.40 set-use-only-extra-torsion-restraints-for-torsions

`set-use-only-extra-torsion-restraints-for-torsions state` [function]

Where *state* is an integer number

set use only extra torsion restraints for torsions

### 13.97.41 use-only-extra-torsion-restraints-for-torsions-state

`use-only-extra-torsion-restraints-for-torsions-state` [function]

return only-use-extra-torsion-restraints-for-torsions state

### 13.97.42 show-restraints-editor

`show-restraints-editor monomer_type` [function]

Where *monomer\_type* is a string

show restraints editor

### 13.97.43 show-restraints-editor-by-index

`show-restraints-editor-by-index menu_item_index` [function]

Where *menu\_item\_index* is an integer number

show restraints editor using residue type index

### 13.97.44 write-restraints-cif-dictionary

`write-restraints-cif-dictionary monomer_type file_name` [function]

Where:

- *monomer\_type* is a string
- *file\_name* is a string

write cif restraints for monomer

## 13.98 Simplex Refinement Interface

### 13.98.1 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

### 13.98.2 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

## 13.99 Nomenclature Errors

### 13.99.1 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 residues altered.

### 13.99.2 set-nomenclature-errors-on-read

`set-nomenclature-errors-on-read` *mode* [function]

Where *mode* is a string

set way nomenclature errors should be handled on reading coordinates.

mode should be "auto-correct", "ignore", "prompt". The default is "prompt"

## 13.100 Atom Info Interface

### 13.100.1 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 to the terminal the Atom Info for the give atom specs

## 13.101 Residue Info

### 13.101.1 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

show residue info dialog for given residue

## 13.102 Residue Environment Functions

### 13.102.1 set-show-environment-distances

`set-show-environment-distances state` [function]

Where *state* is an integer number

show environment distances. If state is 0, distances are turned off, otherwise distances are turned on.

### 13.102.2 set-show-environment-distances-bumps

`set-show-environment-distances-bumps state` [function]

Where *state* is an integer number

show bumps environment distances. If state is 0, bump distances are turned off, otherwise bump distances are turned on.

### 13.102.3 set-show-environment-distances-h-bonds

`set-show-environment-distances-h-bonds state` [function]

Where *state* is an integer number

show H-bond environment distances. If state is 0, bump distances are turned off, otherwise H-bond distances are turned on.

### 13.102.4 show-environment-distances-state

`show-environment-distances-state` [function]

show the state of display of the environment distances

### 13.102.5 set-environment-distances-distance-limits

`set-environment-distances-distance-limits` *min\_dist max\_dist* [function]

Where:

- *min\_dist* is a number
- *max\_dist* is a number

min and max distances for the environment distances

## 13.103 Pointer Functions

### 13.103.1 set-show-pointer-distances

`set-show-pointer-distances` *istate* [function]

Where *istate* is an integer number

turn on (or off) the pointer distance by passing 1 (or 0).

### 13.103.2 show-pointer-distances-state

`show-pointer-distances-state` [function]

show the state of display of the pointer distances

## 13.104 Zoom Functions

### 13.104.1 scale-zoom

`scale-zoom` *f* [function]

Where *f* is a number

scale the view by *f*

external (scripting) interface (with redraw)

### 13.104.2 zoom-factor

`zoom-factor` [function]

return the current zoom factor

### 13.104.3 set-smooth-scroll-do-zoom

`set-smooth-scroll-do-zoom` *i* [function]

Where *i* is an integer number

set smooth scroll with zoom

### 13.104.4 smooth-scroll-do-zoom

`smooth-scroll-do-zoom` [function]

return the state of the above system

## 13.105 CNS Data Functions

### 13.105.1 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)

### 13.105.2 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 Angstroms. *alpha*, *beta*, *gamma* are in degrees. *spg* is the space group info, either ;-delimited symmetry operators or the space group name

## 13.106 mmCIF Functions

### 13.106.1 open-cif-dictionary-file-selector-dialog

`open-cif-dictionary-file-selector-dialog` [function]  
open the cif dictionary file selector dialog

## 13.107 SHELXL Functions

### 13.107.1 read-shelx-ins-file

`read-shelx-ins-file filename recentre_flag` [function]

Where:

- *filename* is a string
- *recentre\_flag* is an integer number

read a SHELXL .ins file

### 13.107.2 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*

## 13.108 Validation Functions

### 13.108.1 difference-map-peaks

`difference-map-peaks` *imol imol\_coords level max\_closeness*  
*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
- *max\_closeness* 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

peaks within *max\_closeness* (2.0 Å typically) of a larger peak are not listed.

### 13.108.2 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 outliers, comparing the O and N temperature factors difference to the distribution of temperature factors from the other atoms.

## 13.109 Ramachandran Plot Functions

### 13.109.1 do-ramachandran-plot

`do-ramachandran-plot` *imol* [function]

Where *imol* is an integer number

Ramachandran plot for molecule number *imol*.

### 13.109.2 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.

### 13.109.3 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)

### 13.109.4 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

### 13.109.5 ramachandran-plot-differences

`ramachandran-plot-differences` *imol1* *imol2* [function]

Where:

- *imol1* is an integer number
- *imol2* is an integer number

2 molecule ramachandran plot (NCS differences) a.k.a. A Kleywegt Plot.

### 13.109.6 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

A chain-specific Kleywegt Plot.

## 13.110 Sequence View Interface

### 13.110.1 do-sequence-view

`do-sequence-view` *imol* [function]

Where *imol* is an integer number

display the sequence view dialog for molecule number *imol*

## 13.111 Atom Labelling

### 13.111.1 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

### 13.111.2 brief-atom-labels-state

`brief-atom-labels-state` [function]

the brief atom label state

## 13.112 Screen Rotation

### 13.112.1 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

### 13.112.2 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

### 13.112.3 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

### 13.112.4 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.

## 13.113 Screen Translation

### 13.113.1 translate-scene-x

`translate-scene-x` *nsteps* [function]

Where *nsteps* is an integer number

translate rotation centre relative to screen axes for nsteps

### 13.113.2 translate-scene-y

`translate-scene-y` *nsteps* [function]

Where *nsteps* is an integer number

translate rotation centre relative to screen axes for nsteps

### 13.113.3 translate-scene-z

`translate-scene-z` *nsteps* [function]

Where *nsteps* is an integer number

translate rotation centre relative to screen axes for nsteps

## 13.114 Views Interface

### 13.114.1 add-view-here

`add-view-here` *view\_name* [function]

Where *view\_name* is a string

return the view number

### 13.114.2 add-view-raw

`add-view-raw` *rcx rcy rcz quat1 quat2 quat3 quat4 zoom view\_name* [function]

Where:

- *rcx* is a number
- *rcy* is a number

- *rcz* is a number
- *quat1* is a number
- *quat2* is a number
- *quat3* is a number
- *quat4* is a number
- *zoom* is a number
- *view\_name* is a string

return the view number

### 13.114.3 remove-named-view

`remove-named-view` *view\_name* [function]  
 Where *view\_name* is a string  
 the view with the given name

### 13.114.4 remove-view

`remove-view` *view\_number* [function]  
 Where *view\_number* is an integer number  
 the given view number

### 13.114.5 add-view-description

`add-view-description` *view\_number* *description* [function]  
 Where:

- *view\_number* is an integer number
- *description* is a string

Add a view description/annotation to the give view number.

### 13.114.6 add-action-view

`add-action-view` *view\_name* *action\_function* [function]  
 Where:

- *view\_name* is a string
- *action\_function* is a string

add a view (not add to an existing view) that does something (e.g. displays or undisplay a molecule) rather than move the graphics.

Returns: the view number for this (new) view.

### 13.114.7 insert-action-view-after-view

`insert-action-view-after-view` *view\_number* *view\_name* *action\_function* [function]

Where:

- *view\_number* is an integer number

- *view\_name* is a string
- *action\_function* is a string

add an action view after the view of the given view number

Returns: the view number for this (new) view.

### 13.114.8 save-views

`save-views view_file_name` [function]

Where *view\_file\_name* is a string

save views to *view\_file\_name*

### 13.114.9 clear-all-views

`clear-all-views` [function]

Clear the view list.

## 13.115 Background Colour

### 13.115.1 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

### 13.115.2 redraw-background

`redraw-background` [function]

re draw the background colour when switching between mono and stereo

### 13.115.3 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.

## 13.116 Ligand Fitting Functions

### 13.116.1 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

### 13.116.2 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

### 13.116.3 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.

### 13.116.4 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.

### 13.116.5 set-find-ligand-multi-solutions-per-cluster

`set-find-ligand-multi-solutions-per-cluster` *lim\_1* *lim\_2* [function]

Where:

- *lim\_1* is a number
- *lim\_2* is a number

allow multiple ligand solutions per cluster.

The first limit is the fraction of the top scored positions that go on to correlation scoring (closer to 1 means less and faster - default 0.7).

The second limit is the fraction of the top correlation score that is considered interesting. Limits the number of solutions displayed to user. Default 0.9.

There is currently no chi-angle set redundancy filtering - I suspect that there should be.

Nino-mode.

### 13.116.6 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.

### 13.116.7 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

### 13.116.8 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

### 13.116.9 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

### 13.116.10 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

### 13.116.11 set-find-ligand-here-cluster

`set-find-ligand-here-cluster` *state* [function]  
Where *state* is an integer number  
Allow the user a scripting means to find ligand at the rotation centre.

### 13.116.12 ligand-expert

`ligand-expert` [function]  
this sets the flag to have expert option ligand entries in the Ligand Searching dialog

### 13.116.13 do-find-ligands-dialog

`do-find-ligands-dialog` [function]  
display the find ligands dialog  
if maps, coords and ligands are available, that is.

### 13.116.14 match-ligand-atom-names

`match-ligand-atom-names` *imol\_ligand chain\_id\_ligand resno\_ligand* [function]  
*ins\_code\_ligand imol\_reference chain\_id\_reference resno\_reference*  
*ins\_code\_reference*

Where:

- *imol\_ligand* is an integer number
- *chain\_id\_ligand* is a string
- *resno\_ligand* is an integer number
- *ins\_code\_ligand* is a string
- *imol\_reference* is an integer number
- *chain\_id\_reference* is a string

- *resno\_reference* is an integer number
- *ins\_code\_reference* is a string

Overlap residue with "template"-based matching.

Overlap the first residue in *imol\_ligand* onto the residue specified by the reference parameters. Use graph matching, not atom names.

Match ligand atom names

By using graph matching, make the names of the atoms of the given ligand/residue match those of the reference residue/ligand as closely as possible - where there would be an atom name clash, invent a new atom name.

Returns: success status, False = failed to find residue in either *imol\_ligand* or *imo\_ref*. If success, return the RT operator.

### 13.116.15 match-ligand-atom-names-to-comp-id

`match-ligand-atom-names-to-comp-id` *imol\_ligand chain\_id\_ligand* [function]  
*resno\_ligand ins\_code\_ligand comp\_id\_ref*

Where:

- *imol\_ligand* is an integer number
- *chain\_id\_ligand* is a string
- *resno\_ligand* is an integer number
- *ins\_code\_ligand* is a string
- *comp\_id\_ref* is a string

Match ligand atom names to a reference ligand type (*comp-id*)

By using graph matching, make the names of the atoms of the given ligand/residue match those of the reference ligand from the geometry store as closely as possible. Where there would be an atom name clash, invent a new atom name.

This doesn't create a new dictionary for the selected ligand - and that's a big problem (see `match_residue_and_dictionary`).

### 13.116.16 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

flip the ligand (usually active residue) around its eigen vectors to the next flip number. Immediate replacement (like flip peptide).

## 13.117 Water Fitting Functions

### 13.117.1 wrapped-create-find-waters-dialog

`wrapped-create-find-waters-dialog` [function]  
create a dialog for water fitting

### 13.117.2 renumber-waters

`renumber-waters` *imol* [function]  
Where *imol* is an integer number  
Renumber the waters of molecule number *imol* with consecutive numbering.

### 13.117.3 execute-find-waters-real

`execute-find-waters-real` *imol\_for\_map imol\_for\_protein* [function]  
*new\_waters\_mol\_flag rmsd\_cut\_off*

Where:

- *imol\_for\_map* is an integer number
- *imol\_for\_protein* is an integer number
- *new\_waters\_mol\_flag* is an integer number
- *rmsd\_cut\_off* is a number

find waters

### 13.117.4 move-waters-to-around-protein

`move-waters-to-around-protein` *imol* [function]  
Where *imol* is an integer number  
move waters of molecule number *imol* so that they are around the protein.  
Returns: the number of moved waters.

### 13.117.5 move-hetgroups-to-around-protein

`move-hetgroups-to-around-protein` *imol* [function]  
Where *imol* is an integer number  
move all hetgroups (including waters) of molecule number *imol* so that they are around the protein.

### 13.117.6 max-water-distance

`max-water-distance` *imol* [function]  
Where *imol* is an integer number  
return the maximum minimum distance of any water atom to any protein atom - used in validation of `move_waters_to_around_protein()` function.

### 13.117.7 set-water-check-spherical-variance-limit

`set-water-check-spherical-variance-limit` *f* [function]

Where *f* is a number

set the limit of interesting variance, above which waters are listed (otherwise ignored)

default 0.12.

### 13.117.8 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

### 13.117.9 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

### 13.117.10 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

### 13.117.11 split-water

`split-water` *imol chain\_id res\_no ins\_code* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *ins\_code* is a string

split the given water and fit to map.

If refinement map is not defined, don't do anything.

If there is more than one atom in the specified residue, don't do anything.

If the given atom does not have an alt conf of "", don't do anything.

## 13.118 Bond Representation

### 13.118.1 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`)

### 13.118.2 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

### 13.118.3 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

### 13.118.4 set-bond-colour-rotation-for-molecule

`set-bond-colour-rotation-for-molecule` *imol f* [function]  
Where:

- *imol* is an integer number
- *f* is a number

set bond colour for molecule

### 13.118.5 get-bond-colour-rotation-for-molecule

`get-bond-colour-rotation-for-molecule` *imol* [function]  
Where *imol* is an integer number  
get the bond colour for molecule.  
Return -1 on err (bad molecule number)

### 13.118.6 set-default-representation-type

`set-default-representation-type` *type* [function]  
Where *type* is an integer number  
set the default representation type (default 1).

### 13.118.7 get-default-bond-thickness

`get-default-bond-thickness` [function]  
get the default thickness for bonds

### 13.118.8 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.

### 13.118.9 set-draw-cis-peptide-markups

`set-draw-cis-peptide-markups` *status* [function]

Where *status* is an integer number

set status of drawing cis-peptide markups

default status is 1.

### 13.118.10 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

### 13.118.11 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*.

### 13.118.12 set-draw-stick-mode-atoms

`set-draw-stick-mode-atoms` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

draw little coloured balls on atoms

turn off with *state* = 0

turn on with *state* = 1

### 13.118.13 graphics-to-ca-representation

`graphics-to-ca-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* as CAs

**13.118.14 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

**13.118.15 graphics-to-ca-plus-ligands-and-sidechains-representation**

`graphics-to-ca-plus-ligands-and-sidechains-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* as CA + ligands + sidechains

**13.118.16 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

**13.118.17 graphics-to-bonds-representation**

`graphics-to-bonds-representation` *mol* [function]

Where *mol* is an integer number

draw molecule number *imol* with normal bonds

**13.118.18 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

**13.118.19 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

**13.118.20 graphics-to-rainbow-representation**

`graphics-to-rainbow-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* in Jones' Rainbow

**13.118.21 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

**13.118.22 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

**13.118.23 graphics-to-occupancy-representation**

`graphics-to-occupancy-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* coloured by occupancy

**13.118.24 graphics-to-user-defined-atom-colours-representation**

`graphics-to-user-defined-atom-colours-representation` *imol* [function]

Where *imol* is an integer number

draw molecule number *imol* in CA+Ligands mode coloured by user-defined atom colours

**13.118.25 graphics-to-user-defined-atom-colours-all-atoms-representation**

`graphics-to-user-defined-atom-colours-all-atoms-representation` [function]

*imol*

Where *imol* is an integer number

draw molecule number *imol* all atoms coloured by user-defined atom colours

**13.118.26 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*

**13.118.27 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

**13.118.28 change-model-molecule-representation-mode**

`change-model-molecule-representation-mode` *up\_or\_down* [function]

Where *up\_or\_down* is an integer number

change the representation of the model molecule closest to the centre of the screen

### 13.118.29 set-use-grey-carbons-for-molecule

`set-use-grey-carbons-for-molecule` *imol state* [function]

Where:

- *imol* is an integer number
- *state* is an integer number

make the carbon atoms for molecule *imol* be grey

### 13.118.30 set-grey-carbon-colour

`set-grey-carbon-colour` *imol r g b* [function]

Where:

- *imol* is an integer number
- *r* is a number
- *g* is a number
- *b* is a number

set the colour for the carbon atoms

can be not grey if you desire, r, g, b in the range 0 to 1.

### 13.118.31 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`)

### 13.118.32 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*

### 13.118.33 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

return the index of the additional representation. Return -1 on error

### 13.118.34 additional-representation-by-attributes

**additional-representation-by-attributes** *imol chain\_id* [function]  
*resno\_start resno\_end ins\_code representation\_type bonds\_box\_type bond\_width*  
*draw\_hydrogens\_flag*

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

return the index of the additional representation.

Returns: -1 on error.

## 13.119 Dots Representation

### 13.119.1 dots

**dots** *imol atom\_selection\_str dots\_object\_name dot\_density* [function]  
*sphere\_size\_scale*

Where:

- *imol* is an integer number
- *atom\_selection\_str* is a string
- *dots\_object\_name* 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)

### 13.119.2 set-dots-colour

`set-dots-colour` *imol r g b* [function]

Where:

- *imol* is an integer number
- *r* is a number
- *g* is a number
- *b* is a number

set the colour of the surface dots of the *imol*-th molecule to be the given single colour  
*r,g,b* are values between 0.0 and 1.0

### 13.119.3 unset-dots-colour

`unset-dots-colour` *imol* [function]

Where *imol* is an integer number

no longer set the dots of molecule *imol* to a single colour  
i.e. go back to element-based colours.

### 13.119.4 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*

### 13.119.5 clear-dots-by-name

`clear-dots-by-name` *imol dots\_object\_name* [function]

Where:

- *imol* is an integer number
- *dots\_object\_name* is a string

clear the first dots object for *imol* with given name

### 13.119.6 n-dots-sets

`n-dots-sets` *imol* [function]

Where *imol* is an integer number

return the number of dots sets for molecule number *imol*

## 13.120 Pep-flip Interface

### 13.120.1 pepflip

`pepflip` *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

`pepflip` the given residue

## 13.121 Rigid Body Refinement Interface

### 13.121.1 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()`

### 13.121.2 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

## 13.122 Add Terminal Residue Functions

### 13.122.1 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

### 13.122.2 add-terminal-residue

`add-terminal-residue` *imol chain\_id residue\_number residue\_type* [function]  
*immediate\_add*

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.

residue type can be "auto" and *immediate\_add* is recommended to be 1.

return 0 on failure, 1 on success

### 13.122.3 add-terminal-residue-using-phi-psi

`add-terminal-residue-using-phi-psi` *imol chain\_id res\_no* [function]  
*residue\_type phi psi*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *res\_no* is an integer number
- *residue\_type* is a string
- *phi* is a number
- *psi* is a number

Add a terminal residue using given phi and psi angles.

### 13.122.4 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.

### 13.122.5 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.

### 13.122.6 add-terminal-residue-do-post-refine-state

`add-terminal-residue-do-post-refine-state` [function]

what is the value of the previous flag?

## 13.123 Delete Residues

### 13.123.1 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

### 13.123.2 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

### 13.123.3 delete-residue-with-full-spec

`delete-residue-with-full-spec` *imol imodel chain\_id resno inscode altloc* [function]

Where:

- *imol* is an integer number
- *imodel* 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

### 13.123.4 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

### 13.123.5 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

### 13.123.6 delete-residue-sidechain

`delete-residue-sidechain` *imol chain\_id resno ins\_code* [function]

*do\_delete\_dialog*

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

### 13.123.7 delete-hydrogens

`delete-hydrogens` *imol* [function]

Where *imol* is an integer number

delete all hydrogens in molecule,

Returns: number of hydrogens deleted.

## 13.124 Mainchain Building Functions

### 13.124.1 db-mainchain

`db-mainchain` *imol chain\_id iresno\_start iresno\_end direction* [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* is a string

CA -> mainchain conversion.

direction is either "forwards" or "backwards"

return the new molecule number

## 13.125 Rotamer Functions

### 13.125.1 set-rotamer-search-mode

`set-rotamer-search-mode` *mode* [function]

Where *mode* is an integer number

set the mode of rotamer search, options are (ROTAMERSEARCHAUTOMATIC), (ROTAMERSEARCHLOWRES) (aka. "backrub rotamers"), (ROTAMERSEARCHHIGHRES) (with rigid body fitting)

### 13.125.2 set-rotamer-lowest-probability

`set-rotamer-lowest-probability` *f* [function]

Where *f* is a number

For Dunbrack rotamers, set the lowest probability to be considered. Set as a percentage i.e. 1.00 is quite low. For Richardson Rotamers, this has no effect.

### 13.125.3 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

### 13.125.4 auto-fit-best-rotamer

`auto-fit-best-rotamer` *resno altloc insertion\_code chain\_id imol\_coords* [function]  
*imol\_map clash\_flag lowest\_probability*

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).

### 13.125.5 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.

### 13.125.6 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.

### 13.125.7 set-residue-to-rotamer-number

`set-residue-to-rotamer-number` *imol chain\_id resno ins\_code alt\_conf* [function]  
*rotamer\_number*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno* is an integer number
- *ins\_code* is a string
- *alt\_conf* is a string
- *rotamer\_number* is an integer number

set the residue specified to the rotamer number specified.

### 13.125.8 set-residue-to-rotamer-name

`set-residue-to-rotamer-name` *imol chain\_id resno ins\_code alt\_conf* [function]  
*rotamer\_name*

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno* is an integer number
- *ins\_code* is a string
- *alt\_conf* is a string
- *rotamer\_name* is a string

set the residue specified to the rotamer name specified.

(rotamer names are the Richardson rotamer names.)

return value is 0 if atoms were not moved (e.g. because rotamer-name was not know)

### 13.125.9 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.

### 13.126 180 Flip Side chain

#### 13.126.1 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

### 13.127 Mutate Functions

#### 13.127.1 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

#### 13.127.2 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.

### 13.127.3 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

mutate a base. return success status, 1 for a good mutate.

### 13.127.4 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.

### 13.127.5 mutate-auto-fit-do-post-refine-state

`mutate-auto-fit-do-post-refine-state` [function]

what is the value of the previous flag?

### 13.127.6 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.

### 13.127.7 rotamer-auto-fit-do-post-refine-state

`rotamer-auto-fit-do-post-refine-state` [function]

what is the value of the previous flag?

### 13.127.8 mutate-single-residue-by-serial-number

`mutate-single-residue-by-serial-number` *ires\_ser chain\_id imol* [function]

*target\_res\_type*

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 single 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 mutate 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

### 13.127.9 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)

## 13.128 Pointer Atom Functions

### 13.128.1 create-pointer-atom-molecule-maybe

`create-pointer-atom-molecule-maybe` [function]

Return the current pointer atom molecule, create a pointer atom molecule if necessary (i.e. when the user has not set it).

### 13.128.2 pointer-atom-molecule

`pointer-atom-molecule` [function]

Return the current pointer atom molecule.

## 13.129 Baton Build Interface Functions

### 13.129.1 set-baton-mode

`set-baton-mode` *i* [function]

Where *i* is an integer number

toggle so that mouse movement moves the baton not rotates the view.

### 13.129.2 try-set-draw-baton

`try-set-draw-baton` *i* [function]

Where *i* is an integer number

draw the baton or not

### 13.129.3 accept-baton-position

`accept-baton-position` [function]

accept the baton tip position - a prime candidate for a key binding

**13.129.4 baton-tip-try-another**

`baton-tip-try-another` [function]  
 move the baton tip position - another prime candidate for a key binding

**13.129.5 baton-tip-previous**

`baton-tip-previous` [function]  
 move the baton tip to the previous position

**13.129.6 shorten-baton**

`shorten-baton` [function]  
 shorten the baton length

**13.129.7 lengthen-baton**

`lengthen-baton` [function]  
 lengthen the baton

**13.129.8 baton-build-delete-last-residue**

`baton-build-delete-last-residue` [function]  
 delete the most recently build CA position

**13.129.9 set-baton-build-params**

`set-baton-build-params` *istart\_resno chain\_id direction* [function]  
 Where:

- *istart\_resno* is an integer number
- *chain\_id* is a string
- *direction* is a string

set the parameters for the start of a new baton-built fragment. *direction* can either be "forwards" or "backwards"

**13.130 Crosshairs Interface****13.130.1 set-draw-crosshairs**

`set-draw-crosshairs` *i* [function]  
 Where *i* is an integer number

draw the distance crosshairs, 0 for off, 1 for on.

**13.131 Edit Chi Angles**

### 13.131.1 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).

### 13.131.2 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 find the residue, return a value of 1 if it worked.

### 13.131.3 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...

## 13.132 Masks

### 13.132.1 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.

### 13.132.2 set-map-mask-atom-radius

`set-map-mask-atom-radius` *rad* [function]

Where *rad* is a number

set the atom radius for map masking

### 13.132.3 map-mask-atom-radius

`map-mask-atom-radius` [function]  
get the atom radius for map masking

## 13.133 check Waters Interface

### 13.133.1 delete-checked-waters-baddies

`delete-checked-waters-baddies` *imol b\_factor\_lim map\_sigma\_lim* [function]  
*min\_dist max\_dist part\_occ\_contact\_flag zero\_occ\_flag*  
*logical\_operator\_and\_or\_flag*

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.

## 13.134 Trim

### 13.134.1 trim-molecule-by-map

`trim-molecule-by-map` *imol\_coords imol\_map map\_level* [function]  
*delete\_or\_zero\_occ\_flag*

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

cut off (delete or give zero occupancy) atoms in the given molecule if they are below the given map (absolute) level.

## 13.135 External Ray-Tracing

### 13.135.1 raster3d

`raster3d` *rd3\_filename* [function]

Where *rd3\_filename* is a string

create a r3d file for the current view

### 13.135.2 set-raster3d-bond-thickness

`set-raster3d-bond-thickness` *f* [function]  
Where *f* is a number  
set the bond thickness for the Raster3D representation

### 13.135.3 set-raster3d-atom-radius

`set-raster3d-atom-radius` *f* [function]  
Where *f* is a number  
set the atom radius for the Raster3D representation

### 13.135.4 set-raster3d-density-thickness

`set-raster3d-density-thickness` *f* [function]  
Where *f* is a number  
set the density line thickness for the Raster3D representation

### 13.135.5 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

### 13.135.6 set-raster3d-bone-thickness

`set-raster3d-bone-thickness` *f* [function]  
Where *f* is a number  
set the bone (skeleton) thickness for the Raster3D representation

### 13.135.7 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

### 13.135.8 set-raster3d-water-sphere

`set-raster3d-water-sphere` *istate* [function]  
Where *istate* is an integer number  
set the flag to show waters as spheres for the Raster3D representation. 1 show as spheres, 0 the usual stars.

### 13.135.9 raster-screen-shot

`raster-screen-shot` [function]  
run raster3d and display the resulting image.

## 13.136 Superposition (SSM)

### 13.136.1 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 value of *move\_imol2\_flag* (1 for move 0 for copy).

### 13.136.2 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 value of *move\_imol2\_flag* (1 for move 0 for copy).

### 13.136.3 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 value of *move\_imol2\_flag* (1 for move 0 for copy).

Returns: the index of the superposed molecule - which could either be a new molecule (if *move\_imol2\_flag* was 1) or the *imol2* or -1 (signifying failure to do the SMM superposition).

## 13.137 NCS

### 13.137.1 set-draw-ncs-ghosts

`set-draw-ncs-ghosts imol istrate` [function]

Where:

- *imol* is an integer number
- *istrate* is an integer number

set drawing state of NCS ghosts for molecule number *imol*

### 13.137.2 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.

### 13.137.3 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*

### 13.137.4 ncs-update-ghosts

`ncs-update-ghosts imol` [function]

Where *imol* is an integer number

update ghosts for molecule number *imol*

### 13.137.5 make-dynamically-transformed-ncs-maps

`make-dynamically-transformed-ncs-maps imol_model imol_map  
overwrite_maps_of_same_name_flag` [function]

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

### 13.137.6 add-ncs-matrix

`add-ncs-matrix` *imol this\_chain\_id target\_chain\_id m11 m12 m13 m21* [function]  
*m22 m23 m31 m32 m33 t1 t2 t3*

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.

### 13.137.7 add-strict-ncs-matrix

`add-strict-ncs-matrix` *imol this\_chain\_id target\_chain\_id m11 m12 m13* [function]  
*m21 m22 m23 m31 m32 m33 t1 t2 t3*

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

### 13.137.8 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*

### 13.137.9 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*

### 13.137.10 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)

### 13.137.11 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.

### 13.137.12 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.

### 13.137.13 copy-residue-range-from-ncs-master-to-others

`copy-residue-range-from-ncs-master-to-others` *imol* [function]  
*master\_chain\_id residue\_range\_start residue\_range\_end*

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.

### 13.137.14 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)

### 13.137.15 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)

### 13.137.16 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

## 13.138 Helices and Strands

### 13.138.1 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 molecule called "Reverse Helix" if the helix orientation isn't completely unequivocal.

Returns: the index of the new molecule.

### 13.138.2 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 rigorous search, but it will be slower.

Returns: the index of the new molecule.

### 13.138.3 place-strand-here-dialog

`place-strand-here-dialog` [function]  
show the strand placement gui.

Choose the python version in there, if needed. Call scripting function, display it in place, don't return a widget.

### 13.138.4 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.

### 13.138.5 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.

### 13.138.6 find-secondary-structure

`find-secondary-structure` *use\_helix helix\_length helix\_target* [function]  
*use\_strand strand\_length strand\_target*

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.

### 13.138.7 find-secondary-structure-local

`find-secondary-structure-local` *use\_helix helix\_length helix\_target* [function]  
*use\_strand strand\_length strand\_target radius*

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.

## 13.139 Nucleotides

### 13.139.1 find-nucleic-acids-local

`find-nucleic-acids-local` *radius* [function]

Where *radius* is a number

autobuild nucleic acid chains

Find secondary structure local to current view in the current map. Add to a molecule called "NuclAcid", create it if needed.

Returns: the index of the new molecule.

## 13.140 New Molecule by Section Interface

### 13.140.1 new-molecule-by-residue-type-selection

`new-molecule-by-residue-type-selection` *imol residue\_type* [function]

Where:

- *imol* is an integer number
- *residue\_type* is a string

create a new molecule that consists of only the residue of type *residue\_type* in molecule number *imol*

Returns: the new molecule number, -1 means an error.

### 13.140.2 new-molecule-by-atom-selection

`new-molecule-by-atom-selection` *imol atom\_selection* [function]

Where:

- *imol* is an integer number
- *atom\_selection* is a string

create a new molecule that consists of only the atoms specified by the mmdb atoms selection string in molecule number *imol*

Returns: the new molecule number, -1 means an error.

### 13.140.3 new-molecule-by-sphere-selection

`new-molecule-by-sphere-selection` *imol x y z r allow\_partial\_residues* [function]

Where:

- *imol* is an integer number
- *x* is a number
- *y* is a number
- *z* is a number
- *r* is a number
- *allow\_partial\_residues* is an integer number

create a new molecule that consists of only the atoms within the given radius (*r*) of the given position.

Returns: the new molecule number, -1 means an error.

## 13.141 RNA/DNA

### 13.141.1 ideal-nucleic-acid

`ideal-nucleic-acid` *RNA\_or\_DNA form single\_stranded\_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

### 13.141.2 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

Return a molecule that contains a residue that is the WC pair partner of the clicked/picked/selected residue.

### 13.141.3 watson-crick-pair-for-residue-range

`watson-crick-pair-for-residue-range` *imol chain\_id resno\_start resno\_end* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno\_start* is an integer number
- *resno\_end* is an integer number

add base pairs for the given residue range, modify molecule *imol* by creating a new chain

## 13.142 Sequence (Assignment)

### 13.142.1 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.

### 13.142.2 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.

### 13.142.3 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. If the chain of the molecule already had a chain assigned to it, then this will overwrite that old assignment with the new one.

### 13.142.4 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.

### 13.142.5 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.

### 13.142.6 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.

### 13.142.7 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.

## 13.143 Surface Interface

### 13.143.1 do-surface

`do-surface` *imol istrate* [function]

Where:

- *imol* is an integer number
- *istrate* 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

### 13.143.2 set-transparent-electrostatic-surface

`set-transparent-electrostatic-surface` *imol opacity* [function]

Where:

- *imol* is an integer number
- *opacity* is a number

simple on/off screendoor transparency at the moment, an *opacity* > 0.0 will turn on screendoor transparency (stippling).

### 13.143.3 get-electrostatic-surface-opacity

`get-electrostatic-surface-opacity` *imol* [function]

Where *imol* is an integer number

return 1.0 for non transparent and 0.5 if screendoor transparency has been turned on.

## 13.144 FFFearing

### 13.144.1 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*

### 13.144.2 set-ffear-angular-resolution

`set-ffear-angular-resolution` *f* [function]

Where *f* is a number

set and return the ffear angular resolution in degrees

### 13.144.3 ffear-angular-resolution

`ffear-angular-resolution` [function]

return the ffear angular resolution in degrees

## 13.145 Remote Control

### 13.145.1 make-socket-listener-maybe

`make-socket-listener-maybe` [function]

try to make socket listener

### 13.145.2 set-socket-string-waiting

`set-socket-string-waiting` *s* [function]

Where *s* is a string

feed the main thread a scheme script to evaluate

### 13.145.3 set-socket-python-string-waiting

`set-socket-python-string-waiting` *s* [function]

Where *s* is a string

feed the main thread a python script to evaluate

## 13.146 Display Lists for Maps

### 13.146.1 set-display-lists-for-maps

`set-display-lists-for-maps` *i* [function]

Where *i* is an integer number

Should display lists be used for maps? It may speed things up if these are turned on (or off) - depends on graphics card and drivers. Pass 1 for on, 0 for off.

### 13.146.2 display-lists-for-maps-state

`display-lists-for-maps-state` [function]

return the state of `display_lists_for_maps`.

## 13.147 Browser Interface

### 13.147.1 browser-url

`browser-url` *url* [function]  
Where *url* is a string  
try to open given url in Web browser

### 13.147.2 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"

### 13.147.3 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.

## 13.148 Molprobit Interface

### 13.148.1 handle-read-draw-probe-dots

`handle-read-draw-probe-dots` *dots\_file* [function]  
Where *dots\_file* is a string  
pass a filename that contains molprobit's probe output in XtalView format

### 13.148.2 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 molprobit's probe output in unformatted format

### 13.148.3 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 molprobit for on edit chi angles intermediate atoms?

#### 13.148.4 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?

#### 13.148.5 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?

#### 13.148.6 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?

#### 13.148.7 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).

#### 13.148.8 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

#### 13.148.9 interactive-probe-dots-molprobity-radius

`interactive-probe-dots-molprobity-radius` [function]  
return the radius over which we can run interactive probe.

### 13.149 Map Sharpening Interface

#### 13.149.1 sharpen

`sharpen` *imol b\_factor* [function]  
Where:

- *imol* is an integer number
- *b\_factor* is a number

Sharpen map *imol* by *b\_factor* (note (of course) that positive numbers blur the map).

#### 13.149.2 set-map-sharpening-scale-limit

`set-map-sharpening-scale-limit` *f* [function]  
Where *f* is a number  
set the limit of the b-factor map sharpening slider (default 30)

## 13.150 Marking Fixed Atom Interface

### 13.150.1 clear-all-fixed-atoms

`clear-all-fixed-atoms` *imol* [function]

Where *imol* is an integer number

clear all fixed atoms

## 13.151 Partial Charges

### 13.151.1 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

show the partial charges for the residue of the given specs (charges are read from the dictionary)

## 13.152 EM interface

### 13.152.1 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

Scale the cell, for use with EM maps, where the cell needs to be adjusted. Use like: (scale-cell 2 1.012 1.012 1.012). Return error status, 1 means it worked, 0 means it did not work.

## 13.153 CCP4mg Interface

### 13.153.1 write-ccp4mg-picture-description

`write-ccp4mg-picture-description` *filename* [function]

Where *filename* is a string

write a ccp4mg picture description file

### 13.153.2 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

get element colour for imol as Python formatted list char

## 13.154 Aux functions

### 13.154.1 laplacian

`laplacian` *imol* [function]

Where *imol* is an integer number

Create the "Laplacian" (-ve second derivative) of the given map.

## 13.155 SMILES

### 13.155.1 do-smiles-gui

`do-smiles-gui` [function]

display the SMILES string dialog

## 13.156 PHENIX Support

### 13.156.1 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

## 13.157 Graphics Text

### 13.157.1 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

### 13.157.2 remove-text

`remove-text` *text\_handle* [function]

Where *text\_handle* is an integer number

Remove "3d" text item.

### 13.157.3 text-index-near-position

`text-index-near-position` *x y z r* [function]

Where:

- *x* is a number
- *y* is a number
- *z* is a number
- *r* is a number

return the closest text that is with r Å of the given position. If no text item is close, then return -1

## 13.158 PISA Interaction

### 13.158.1 pisa-interaction

`pisa-interaction` *imol\_1 imol\_2* [function]

Where:

- *imol\_1* is an integer number
- *imol\_2* is an integer number

return the molecule number of the interacting residues. Return -1 if no new model was created. Old, not very useful.

## 13.159 Jiggle Fit

### 13.159.1 fit-to-map-by-random-jiggle

`fit-to-map-by-random-jiggle` *imol chain\_id resno ins\_code n\_trials jiggle\_scale\_factor* [function]

Where:

- *imol* is an integer number
- *chain\_id* is a string
- *resno* is an integer number
- *ins\_code* is a string
- *n\_trials* is an integer number
- *jiggle\_scale\_factor* is a number

jiggle fit to the current refinement map. return < -100 if not possible, else return the new best fit for this residue.

## 13.160 SBase interface

### 13.160.1 get-ccp4srs-monomer-and-dictionary

`get-ccp4srs-monomer-and-dictionary` *comp\_id* [function]

Where *comp\_id* is a string

return the new molecule number of the monomer.

The monomer will have chainid "A" and residue number 1.

Return -1 on failure to get monomer.

### 13.160.2 get-sbase-monomer

`get-sbase-monomer` *comp\_id* [function]

Where *comp\_id* is a string

same as above but using old name for back-compatibility

## 13.161 FLE-View

### 13.161.1 fle-view-set-water-dist-max

`fle-view-set-water-dist-max` *dist\_max* [function]

Where *dist\_max* is a number

set the maximum considered distance to water

default 3.25 Å.

### 13.161.2 fle-view-set-h-bond-dist-max

`fle-view-set-h-bond-dist-max` *h\_bond\_dist\_max* [function]

Where *h\_bond\_dist\_max* is a number

set the maximum considered hydrogen bond distance

default 3.9 Å.

## 13.162 LSQ-improve

### 13.162.1 lsq-improve

`lsq-improve` *imol\_ref* *ref\_selection* *imol\_moving* *moving\_selection* *n\_res* *dist\_crit* [function]

Where:

- *imol\_ref* is an integer number
- *ref\_selection* is a string
- *imol\_moving* is an integer number
- *moving\_selection* is a string
- *n\_res* is an integer number
- *dist\_crit* is a number

an slightly-modified implementation of the "lsq\_improve" algorithm of Kleywegt and Jones (1997).

Note that if a residue selection is specified in the residue selection(s), then the first residue of the given range must exist in the molecule (if not, then mmdb will not select any atoms from that molecule).

Kleywegt and Jones set `n_res` to 4 and `dist_crit` to 6.0.

## 13.163 single-model view

### 13.163.1 single-model-view-model-number

`single-model-view-model-number` *imol imodel* [function]

Where:

- *imol* is an integer number
- *imodel* is an integer number

put molecule number *imol* to display only model number *imodel*

### 13.163.2 single-model-view-this-model-number

`single-model-view-this-model-number` *imol* [function]

Where *imol* is an integer number

the current model number being displayed

return 0 on non-multimodel-molecule.

### 13.163.3 single-model-view-next-model-number

`single-model-view-next-model-number` *imol* [function]

Where *imol* is an integer number

change the representation to the next model number to be displayed

return 0 on non-multimodel-molecule.

### 13.163.4 single-model-view-prev-model-number

`single-model-view-prev-model-number` *imol* [function]

Where *imol* is an integer number

change the representation to the previous model number to be displayed

return 0 on non-multimodel-molecule.

## 13.164 graphics 2D ligand view

### 13.164.1 set-show-graphics-ligand-view

`set-show-graphics-ligand-view` *state* [function]

Where *state* is an integer number

set the graphics ligand view state

(default is 1 (on)).

### 13.165 libcheck

`libcheck-exe` [procedure]  
 this is override-able by the user in their `.coot` file (for example).

`monomer-molecule-from-3-let-code` *code dict-cif-libin .* [procedure]  
*ccp4i-project-dir*

Return -2 on *code* is not a string Return -3 on libcheck failure Return *imol* on success  
 Return `handle-read-draw-molecule` error code on failure to read resultant pdb file  
 Actually, it would be nice to know if this code represented a full description - or a minimal one... perhaps we can parse the log file and call a pop-up that will tell us.  
*dict-cif-libin* should be a string. If it is "" then it is ignored. If it is not "" then it is used to create input to libcheck (not a command line argument) so that bespoke dictionary libraries can produce coords using "Get Monomer".

### 13.166 check-for-updates

`get-stable-release-from-server-string` *str* [procedure]  
 Is this true?

`get-stable-release-from-coot-version` [procedure]  
 Needs testing.

`notify-of-new-version` *str* [procedure]  
 show the dialog

`download-binary-dialog` [procedure]  
 version-string is something like: "coot-0.6-pre-1-revision-2060"

`directory-is-modifiable?` *prefix-dir* [procedure]  
 Test for prefix-dir 1) existing 2) being a directory 3) modifiable by user (ie. `u+rwx`)  
 Return #t or #f.

### 13.167 snarf-coot-docs

`directory-as-file-name` *dir* [procedure]  
 remove any trailing /s

`file-name-file` *file-name* [procedure]  
 return the file component of file-name "x/y" -> "y" "x" -> "x"

`useful-glob` *dir pat* [procedure]  
 prepend the files with the directory dir  
 (note that the args are reversed to "natural" order (which is the reverse of glob args))

`glob` *pat dir* [procedure]  
 return a list of file names that match pattern pat in directory dir.

`strip-extension` *s* [procedure]  
 "a.b.res" -> "a.b" file-name-sans-extension

- command-in-path?** *cmd* [procedure]  
Return #t or #f:
- goosh-command** *cmd args data-list log-file-name screen-output-also?* [procedure]  
Where *cmd* is e.g. "refmac" *args* is (list "HKLIN" "thing.mtz") *log-file-name* is "refmac.log" *data-list* is (list "HEAD" "END")  
Return the exist status e.g. 0 or 1.
- run-command/strings** *cmd args data-list* [procedure]  
Return the strings screen output of *cmd* or #f if command was not found
- string->list-of-strings** *str* [procedure]  
Return a list if *str* is a string, else return '()
- directory-files** *dir* [procedure]  
The following functions from PLEAC (guile version thereof of course).  
or define a utility function for this
- add-to-list-section-texis** [procedure]
- delete-section-texi-files** [procedure]

### 13.168 coot-gui

- run-gtk-pending-events** [procedure]
- coot-gui** [procedure]  
Fire up the coot scripting gui. This function is called from the main C++ code of coot. Not much use if you don't have a gui to type functions in to start with.
- handle-smiles-go** *tlc-entry smiles-entry* [procedure]  
The callback from pressing the Go button in the smiles widget, an interface to run libcheck.
- smiles-gui** [procedure]  
smiles GUI
- generic-single-entry** *function-label entry-1-default-text go-button-label handle-go-function* [procedure]  
Generic single entry widget  
Pass the hint labels of the entries and a function that gets called when user hits "Go". The handle-go-function accepts one argument that is the entry text when the go button is pressed.
- generic-double-entry** *label-1 label-2 entry-1-default-text entry-2-default-text check-button-label handle-check-button-function go-button-label handle-go-function* [procedure]  
handle-go-function takes 3 arguments, the third of which is the state of the check button.  
if check-button-label not a string, then we don't display (or create, even) the check-button. If it *\*is\** a string, create a check button and add the callback handle-check-button-function which takes as an argument the active-state of the the checkbutton.

- generic-multiple-entries-with-check-button** *entry-info-list* [procedure]  
*check-button-info go-button-label handle-go-function*  
 generic double entry widget, now with a check button  
 OLD:  
 pass a the hint labels of the entries and a function (handle-go-function) that gets called when user hits "Go" (which takes two string arguments and the active-state of the check button (either #t or #f)).  
 if check-button-label not a string, then we don't display (or create, even) the check-button. If it \*is\* a string, create a check button and add the callback handle-check-button-function which takes as an argument the active-state of the the checkbutton.
- molecule-centres-gui** [procedure]  
 A demo gui to move about to molecules.
- old-coot?** [procedure]  
 old coot test
- interesting-things-gui** *title baddie-list* [procedure]  
 We can either go to a place (in which case the element is a list of button label (string) and 3 numbers that represent the x y z coordinates) or an atom (in which case the element is a list of a button label (string) followed by the molecule-number chain-id residue-number ins-code atom-name altconf)  
 e.g. (interesting-things-gui "Bad things by Analysis X" (list (list "Bad Chiral" 0 "A" 23 "" "CA" "A") (list "Bad Density Fit" 0 "B" 65 "" "CA" "") (list "Interesting blob" 45.6 46.7 87.5)))
- interesting-things-with-fix-maybe** *title baddie-list* [procedure]  
 In this case, each baddie can have a function at the end which is called when the fix button is clicked.
- fill-option-menu-with-mol-options** *menu filter-function* [procedure]  
 Fill an option menu with the "right type" of molecules. If filter-function returns #t then add it. Typical value of filter-function is valid-model-molecule?
- fill-option-menu-with-coordinates-mol-options** *menu* [procedure]  
 Helper function for molecule chooser. Not really for users.  
 Return a list of models, corresponding to the menu items of the option menu.  
 The returned list will not contain references to map or closed molecules.
- fill-option-menu-with-number-options** *menu number-list* [procedure]  
*default-option-value*
- get-option-menu-active-molecule** *option-menu model-mol-list* [procedure]  
 Helper function for molecule chooser. Not really for users.  
 return the molecule number of the active item in the option menu, or return #f if there was a problem (e.g. closed molecule)
- get-option-menu-active-item** *option-menu item-list* [procedure]  
 Here we return the active item in an option menu of generic items

**molecule-chooser-gui-generic** *chooser-label callback-function* [procedure]  
*option-menu-fill-function*

Typically option-menu-fill-function is fill-option-menu-with-coordinates-mol-options

**molecule-chooser-gui** [procedure]

Fire up a coordinates/model molecule chooser dialog, with a given label and on OK we call the call-back-fuction with an argument of the chosen molecule number.

chooser-label is a directive to the user such as "Choose a Molecule"

callback-function is a function that takes a molecule number as an argument.

**map-molecule-chooser-gui** [procedure]

Fire up a map molecule chooser dialog, with a given label and on OK we call the call-back-fuction with an argument of the chosen molecule number.

chooser-label is a directive to the user such as "Choose a Molecule"

callback-function is a function that takes a molecule number as an argument.

**generic-chooser-and-entry** *chooser-label entry-hint-text* [procedure]  
*default-entry-text callback-function*

A pair of widgets, a molecule chooser and an entry. The callback-function is a function that takes a molecule number and a text string.

**generic-chooser-entry-and-file-selector** *chooser-label* [procedure]  
*chooser-filter entry-hint-text default-entry-text file-selector-hint*  
*callback-function*

Create a window

Return a pair of widgets, a molecule chooser and an entry. The callback-function is a function that takes a molecule number and 2 text strings (e.g chain-id and file-name)

chooser-filter is typically valid-map-molecule? or valid-model-molecule?

**generic-chooser-and-file-selector** *chooser-label chooser-filter* [procedure]  
*file-selector-hint default-file-name callback-function*

Create a window.

Return a pair of widgets, a molecule chooser and an entry. callback-function is a function that takes a molecule number and a file-name

chooser-filter is typically valid-map-molecule? or valid-model-molecule?

**coot-menubar-menu** *menu-label* [procedure]

If a menu with label menu-label is not found in the coot main menubar, then create it and return it. If it does exist, simply return it.

**add-simple-coot-menu-menuitem** *menu menu-item-label* [procedure]  
*activate-function*

Given that we have a menu (e.g. one called "Extensions") provide a cleaner interface to adding something to it:

activate-function is a thunk.

**missing-atoms-gui** *imol* [procedure]

Make an interesting things GUI for residues with missing atoms

- generic-buttons-dialog** *dialog-name button-list* [procedure]  
 button-list is a list of pairs (improper list) the first item of which is the button label text the second item is a lambda function, what to do when the button is pressed.
- generic-interesting-things** *imol gui-title-string residue-test-func* [procedure]  
 Generic interesting things gui: user passes a function that takes 4 args: the chain-id, resno, inscode and residue-serial-number (should it be needed) and returns either #f or something interesting (e.g. a label/value). It is the residue-test-func of the residue-matching-criteria function.
- generic-number-chooser** *number-list default-option-value hint-text* [procedure]  
*go-button-label go-function*  
 A gui that makes a generic number chooser the go function is a lambda function that takes the value of the active menu item - as a number.
- generic-molecule-chooser** *hbox hint-text* [procedure]  
 pack a hint text and a molecule chooser option menu into the given vbox.  
 return the option-menu and model molecule list:
- file-selector-entry** *hbox hint-text* [procedure]  
 Return an entry, the widget is inserted into the hbox passed to this function.
- cootaneer-gui** *imol* [procedure]  
 Cootaneer gui
- view-saver-gui** [procedure]  
 The gui for saving views
- add-view-to-views-panel** *view-name view-number* [procedure]
- dialog-box-of-buttons-with-check-button** *window-name* [procedure]  
*geometry buttons close-button-label check-button-label check-button-func*  
*check-button-is-initially-on-flag*  
 If check-button-label is #f, don't make one, otherwise create with with the given label and "on" state.
- dialog-box-of-pairs-of-buttons** *imol window-name geometry* [procedure]  
*buttons close-button-label*  
 geometry is an improper list of ints buttons is a list of: (list (list button-1-label button-1-action button-2-label button-2-action)) The button-1-action function takes as an argument the imol The button-2-action function takes as an argument the imol
- views-panel-gui** [procedure]  
 A gui showing views:
- nudge-screen-centre-gui** [procedure]  
 nudge screen centre box. Useful when Ctrl left-mouse has been taken over by another function.
- make-difference-map-gui** [procedure]  
 A gui to make a difference map (from arbitrarily gridded maps (that's it's advantage))

- `cis-peptides-gui` *imol* [procedure]  
A GUI to display all the CIS peptides and navigate to them.
- `associate-pir-with-molecule-gui` *do-alignment?* [procedure]  
Associate the contents of a PIR file with a molecule. Select file from a GUI.
- `alignment-mismatches-gui` *imol* [procedure]  
Make a box-of-buttons GUI for the various modifications that need to be made to match the model sequence to the assigned sequence(s).  
Call this when the associated sequence(s) have been read in already.
- `residue-range-gui` [procedure]  
(Notice that we are not dealing with insertion codes).
- `user-mods-gui` *imol pdb-file-name* [procedure]  
USER MODS gui
- `rename-residue-gui` [procedure]  
simple rename residue GUI
- `water-coordination-gui` [procedure]  
close-button
- `min-max-residues-from-atom-specs` *specs* [procedure]  
return a list, or *#f* (e.g. if not in same chain and molecule)
- `*db-loop-preserve-residue-names*` [procedure]  
by default, rename loop residues to UNK. If scheme true, then leave them as the residue names found in the database.

### 13.169 jligand-gui

- `launch-jligand-function` [procedure]  
This happens when user clicks on the "Launch JLigand" button. It starts a jligand and puts it in the background.
- `click-select-residues-for-jligand` [procedure]  
This happens when user clicks on the "Select Residues for JLigand" (or some such) button. It expects the user to click on atoms of the two residues involved in the link.

### 13.170 remote-control

- `%coot-listener-socket` [procedure]  
a place-holder for the coot listener socket
- `open-coot-listener-socket` *port-number host-name* [procedure]  
Open a coot listener socket, return nothing much, but do set! `%coot-listener-socket`.  
Hmmm... why make a side-effect like that? Why not return `%coot-listener-socket` so that the caller can set it? There may be a reason...  
And the reason is that I can then call `coot-listener-idle-function-proc` without having to use a `c++` variable.

`open-coot-listener-socket-with-timeout` *port-number host-name* [procedure]  
yet another go to make a coot port reader work. This time, we use a gtk-timer to read stuff from the socket.

The gtk-timer function must return 1 to be called again. When we want to close the socket reader, simply make the function return 0.

`coot-socket-timeout-func` [procedure]  
based on `coot-listener-idle-function-proc`

`coot-listener-idle-function-proc` [procedure]  
Do this thing when idle  
currently set to listen to the `%coot-listener-socket`

`eval-socket-string` *s* [procedure]  
the function to run from the main thread to evaluate a string:

### 13.171 americanisms

`set-rotation-center` [procedure]  
an americanism

`handle-read-draw-molecule-with-recenter` [procedure]  
an americanism

`set-rotation-center-size` [procedure]  
an americanism

`center-atom-label-status` [procedure]  
an americanism

`set-last-map-color` [procedure]  
an americanism

`center-of-mass` [procedure]  
an americanism

`set-swap-difference-map-colors` [procedure]  
an americanism

`set-font-color` [procedure]  
an americanism

`set-dots-color` [procedure]  
an americanism

`invert-chiral-center` [procedure]  
an americanism

### 13.172 mutate-from-scheme

`povray-version` [procedure]  
 this should negate the need for Bill's patch here.

`povray-args` [procedure]  
 args not including the output filename

`povray-image` [procedure]  
 Run povray using current displayed image and write .pov file to default filename

### 13.173 filter

`filter fn ls` [procedure]  
 Basic scheme function, filter the objects in list *ls* by function *fn*. e.g. (filter even? (list 0 1 2 3) -> '(0 2))

### 13.174 user-define-restraints

`add-base-restraint imol spec-1 spec-2 atom-name-1 atom-name-2 dist` [procedure]  
 spec-1 and spec-2 are 7-element atom-specs

`run-prosmart imol-target imol-ref include-side-chains?` [procedure]  
 target is my molecule, ref is the homologous (high-res) model

`write-refmac-parallel-plane-restraint file-name res-spec-0` [procedure]  
*res-spec-1 atom-list-0 atom-list-1*  
 exte stac plan 1 firs resi 99 chai A atoms CB CG CD1 CD2 CE1 CE2 CZ OH plan  
 2 firs resi 61 chai B atoms CB CG CD1 CD2 CE1 CE2 CZ dist 3.4 sddi 0.2 sdan  
 6.0 type 1

### 13.175 get-recent-pdbe

`coot-thread-dispatcher` [procedure]  
 (coot-thread-dispatcher image-name thunk)

`dialog-box-of-buttons-with-async-ligands window-name` [procedure]  
*geometry buttons close-button-label*  
 If check-button-label is #f, don't make one, otherwise create with with the given label and "on" state.

`cache-or-net-get-image image-url image-name func` [procedure]  
 Get image-name (caller doesn't care how) and when it is in place run thunk func.  
 This is a generally useful function, so it has been promoted outside of dig-table.

`refmac-calc-sfs-make-mtz-with-columns pdb-in-file-name` [procedure]  
*mtz-file-name mtz-refmaced-file-name f-col sigf-col r-free-col*  
 return refmac-result or #f

`pdbe-get-pdb-and-sfs-cif` *include-get-sfs-flag* *entry-id* [procedure]

Use progress bars

*include-get-sfs-flag* is either 'no-sfs or 'include-sfs

`pdbe-latest-releases-gui` [procedure]

Sameer Velankar says to get this file for the latest releases "<http://www.ebi.ac.uk/pdbe-apps/jsonizer/latest/released/>" (note the end "/").

# Concept Index

- - .coot ..... 12
  - .coot.py ..... 12
- ## A
- alternate conformation ..... 36
  - anisotropic atoms ..... 22
  - annotations ..... 17
  - atom colouring ..... 10, 19
  - atom info ..... 19
  - atom label, brief ..... 19
  - atom labeling ..... 19
  - atom picking ..... 44
  - atomic dots ..... 24
  - auto-fit rotamer ..... 34
  - average map ..... 51
- ## B
- B-Factor, New Atoms ..... 43
  - background colour ..... 16
  - backrub rotamers ..... 35
  - backups ..... 13
  - backups, clearing ..... 61
  - ball and stick ..... 24
  - baton build ..... 31
  - baton mode ..... 31
  - big maps ..... 51
  - blobs ..... 56
  - bond thickness ..... 20
  - bones ..... 52
- ## C
- C-terminus ..... 41
  - C\alpha representation ..... 10
  - C\alpha symmetry ..... 23
  - carbohydrates ..... 29
  - chain id ..... 46
  - change contour level ..... 48
  - changing chain ids ..... 46
  - changing the Refinement Map ..... 30
  - chi angles ..... 35
  - cif dictionary ..... 28
  - citing coot ..... 1
  - clashing residues ..... 35
  - Clear Pending Picks ..... 44
  - clearing backups ..... 61
  - clipping ..... 16
  - colour by chain ..... 10
  - colouring, atoms ..... 19
  - colouring, map ..... 50
  - command line arguments ..... 3
  - contouring, map ..... 6, 49
  - coordinates format ..... 18
  - coot droppings ..... 60
  - crash ..... 4
  - crash recovery ..... 4
  - crosshairs ..... 17
- ## D
- debugger ..... 4
  - default reftmac executable ..... 43
  - delete ..... 44
  - density line thickness ..... 50
  - depth-cueing ..... 9
  - dictionary, cif ..... 2, 39
  - dictionary, ligands ..... 2
  - difference map ..... 49, 51
  - difference map colours ..... 49
  - directory for save coordinates ..... 11
  - Display Manager ..... 10
  - DNA, ideal ..... 42
  - DNA, mutating ..... 37
  - docking sidechains ..... 33
  - droppings ..... 60
  - dual conformations ..... 36
- ## E
- edit \chi angles ..... 35
  - edit B-factors ..... 19
  - edit occupancy ..... 19
  - Electron Density Server ..... 21
  - executing commands ..... 12
  - export map ..... 54
  - EYC-TIED ..... 48
- ## F
- file-name filtering ..... 11
  - fill partial residues ..... 45
  - findligand ..... 63
  - flip peptide ..... 36
  - fragment direction change ..... 33
  - frame rate ..... 17
- ## G
- gdb ..... 4
  - GLN and ASN B-factor Outliers ..... 57
  - goose ..... 21
  - guano ..... 60
  - guile ..... 12

**H**

|                       |    |
|-----------------------|----|
| helix placement ..... | 42 |
| hydrogens .....       | 20 |

**I**

|                           |    |
|---------------------------|----|
| initialization file ..... | 12 |
|---------------------------|----|

**L**

|                             |        |
|-----------------------------|--------|
| Least Squares Fitting ..... | 25     |
| ligand orientation .....    | 40     |
| ligand torsion angles ..... | 36     |
| ligand, overlay .....       | 25     |
| ligands .....               | 38, 39 |
| ligands, flexible .....     | 39     |

**M**

|                                              |    |
|----------------------------------------------|----|
| mainchain .....                              | 33 |
| mainchain torsions .....                     | 33 |
| map box radius .....                         | 50 |
| map changing (for refinement/building) ..... | 30 |
| map extent .....                             | 50 |
| map line width .....                         | 50 |
| map re-Interpolation .....                   | 52 |
| map scrolling .....                          | 6  |
| map transformation .....                     | 53 |
| masks .....                                  | 53 |
| mean B-factor .....                          | 24 |
| median B-factor .....                        | 24 |
| merge molecules .....                        | 42 |
| missing symmetry .....                       | 23 |
| mmCIF dictionary .....                       | 28 |
| modelling toolbar .....                      | 11 |
| modified labels .....                        | 47 |
| molecule centre .....                        | 47 |
| molecule number .....                        | 8  |
| Molprobit Tools .....                        | 57 |
| monomers .....                               | 38 |
| mouse .....                                  | 2  |
| mouse buttons .....                          | 5  |
| moving ligands .....                         | 47 |
| moving molecules .....                       | 47 |
| Moving Zero Occupancy Atoms .....            | 30 |
| multi-mutate .....                           | 37 |
| multiple coordinates files .....             | 18 |
| mutate .....                                 | 37 |
| mutating DNA .....                           | 37 |
| mutating RNA .....                           | 37 |

**N**

|                               |    |
|-------------------------------|----|
| NCS .....                     | 20 |
| NCS averaging .....           | 21 |
| NCS edits .....               | 43 |
| negative contour levels ..... | 50 |

**O**

|                         |    |
|-------------------------|----|
| OCA .....               | 21 |
| OpenGL .....            | 9  |
| Orientation Axes .....  | 16 |
| origin marker .....     | 9  |
| output .....            | 17 |
| overlying ligands ..... | 25 |
| OXT atom .....          | 41 |

**P**

|                                  |    |
|----------------------------------|----|
| packing diagram .....            | 23 |
| Patterson .....                  | 52 |
| pepflip .....                    | 36 |
| peptide restraints, planar ..... | 29 |
| PHS data .....                   | 49 |
| PHS data format .....            | 48 |
| picking .....                    | 61 |
| pink pointer .....               | 16 |
| planar peptide restraints .....  | 29 |
| planes .....                     | 27 |
| polynucleotides .....            | 42 |
| post-manipulation-hook .....     | 31 |
| Print Sequence .....             | 23 |
| probe .....                      | 57 |
| python .....                     | 12 |

**R**

|                                  |    |
|----------------------------------|----|
| Ramachandran plot .....          | 55 |
| Raster3D .....                   | 10 |
| reading multiple pdb files ..... | 18 |
| recentring view .....            | 15 |
| recover session .....            | 4  |
| redo .....                       | 14 |
| reduce .....                     | 57 |
| reference manual .....           | 1  |
| reference structures .....       | 2  |
| refine single click .....        | 27 |
| refinement .....                 | 27 |
| refinement weight .....          | 27 |
| refinement, rigid body .....     | 30 |
| refinement, simplex .....        | 31 |
| refining residues .....          | 29 |
| refmac .....                     | 43 |
| refmac map colour .....          | 44 |
| refmac parameters .....          | 43 |
| refmac, default .....            | 43 |
| regularization .....             | 27 |
| render .....                     | 10 |
| renumbering residues .....       | 38 |
| residue info .....               | 19 |
| resizing view .....              | 62 |
| restore after crash .....        | 14 |
| reverse direction .....          | 33 |
| rigid body fit .....             | 30 |
| RNA, ideal .....                 | 42 |
| RNA, mutating .....              | 37 |

- rotamers ..... 34  
rotate/translate, manual ..... 30  
rotation centre ..... 15  
rotation centre pointer ..... 16  
running refmac ..... 43  
running SHELXL ..... 44
- S**
- save coordinates directory ..... 11  
scheme ..... 12  
screenshot ..... 9  
scripting ..... 11  
Scroll ..... 10  
scroll wheel, map for ..... 62  
scroll, map contour change by ..... 6  
scroll-wheel ..... 6  
Secondary Structure Matching (SSM) ..... 24  
sequence view ..... 23  
**set-matrix** ..... 27  
set-rotation-centre ..... 15  
setting space group ..... 22  
sharpening, map ..... 52  
SHELX .ins ..... 18  
SHELXL ..... 44  
sidechains ..... 33  
simplex refinement ..... 31  
single click refine ..... 27  
skeleton updating ..... 52  
skeleton, missing ..... 32  
skeletonization ..... 52  
slab ..... 16  
sliding ..... 15  
slow computer ..... 62  
SMILES strings ..... 38  
space group names ..... 18  
space group operators ..... 15  
sphere refinement ..... 28  
startup dialog (state) ..... 13  
startup settings (python) ..... 12  
startup settings (scheme) ..... 12  
state ..... 12  
superposition ..... 24  
symmetry ..... 22  
symmetry operators ..... 15
- T**
- terminal oxygen ..... 41
- terminal residue ..... 41  
thickness of density lines ..... 50  
tooltips ..... 8  
torsion angles, ligand ..... 36  
torsion general ..... 36  
torsion restraints ..... 27  
torsions ..... 35  
trackball, virtual ..... 6  
traffic lights ..... 28  
translate molecule ..... 47  
translation, keyboard ..... 6  
translation, mouse ..... 5  
trimming atoms ..... 53
- U**
- undo ..... 13  
unexplained density ..... 56  
unit cell ..... 16  
UNK residue ..... 30  
unmodelled density ..... 56
- V**
- Validation Graphs ..... 58  
version number ..... 8  
view matrix ..... 14
- W**
- waters, finding ..... 40  
web page ..... 4  
weight, real space refinement ..... 27  
width, bonds ..... 20  
write map ..... 54  
writing PDBs ..... 26
- Y**
- yellow box ..... 9
- Z**
- z-rotation ..... 6  
zero occupancy ..... 23  
zoom ..... 62  
zoom, slider ..... 7

# Function Index

## %

%coot-listener-socket ..... 367

## \*

\*add-linked-residue-tree-correlation-cut-off\* ..... 216  
 \*annotations\* ..... 224  
 \*continue-multi-refine\* ..... 218  
 \*db-loop-preserve-residue-names\* ..... 367  
 \*default-ball-and-stick-selection\* ..... 224  
 \*do-coot-tips-flag\* ..... 217  
 \*imol-jligand-link\* ..... 212  
 \*interactive-probe-is-OK?\* ..... 221  
 \*key-bindings\* ..... 231  
 \*ligand-check-resolution-limit\* ..... 213  
 \*pisa-command\* ..... 214  
 \*probe-command\* ..... 217  
 \*reduce-command\* ..... 217  
 \*to-jligand-secret-file-name\* ..... 212  
 \*use-mogul\* ..... 233

## 3

3-letter-code->single-letter ..... 214

## A

absolutify ..... 225  
 accept-baton-position ..... 167, 338  
 accept-regularizement ..... 130, 301  
 accept-reject-dialog-docked-show-state ... 82, 253  
 accept-reject-dialog-docked-state .... 81, 252  
 active-residue ..... 195, 206  
 add-action-view ..... 145, 316  
 add-annotation-here ..... 233  
 add-base-restraint ..... 369  
 add-coordinates-glob-extension ..... 65, 236  
 add-data-glob-extension ..... 65, 236  
 add-dictionary-glob-extension ..... 65, 236  
 add-extra-bond-restraint ..... 134, 305  
 add-extra-restraints-to-other-molecule .. 219  
 add-hydrogens-from-file ..... 196, 207  
 add-hydrogens-using-refmac-inner ..... 224  
 add-key-binding ..... 231  
 add-ligand-delete-residue-copy-molecule ..... 70, 241  
 add-ligand-search-ligand-molecule ... 148, 319  
 add-ligand-search-wiggly-ligand-molecule ..... 148, 319  
 add-map-glob-extension ..... 65, 236  
 add-molecule ..... 194, 205  
 add-ncs-matrix ..... 174, 345

add-omega-torsion-restraints ..... 129, 300  
 add-planar-peptide-restraints ..... 128, 299  
 add-simple-coot-menu-menuitem ..... 365  
 add-status-bar-text ..... 81, 252  
 add-strict-ncs-matrix ..... 174, 345  
 add-terminal-residue ..... 160, 331  
 add-terminal-residue-do-post-refine-state ..... 160, 331  
 add-terminal-residue-using-phi-psi .. 160, 331  
 add-tmp-extension-to ..... 227  
 add-to-list-section-texis ..... 363  
 add-view ..... 228  
 add-view-description ..... 145, 316  
 add-view-here ..... 144, 315  
 add-view-raw ..... 144, 315  
 add-view-to-views-panel ..... 366  
 additional-representation-by-attributes ..... 157, 328  
 additional-representation-by-string ..... 156, 327  
 align-and-mutate ..... 126, 297  
 alignment-mismatches-gui ..... 367  
 all-residues ..... 232  
 all-true? ..... 224  
 allow-duplicate-sequence-numbers ..... 74, 245  
 another-level ..... 96, 267  
 another-level-from-map-molecule-number ... 97, 268  
 append-dir-dir ..... 228  
 append-dir-file ..... 228  
 apply-ncs-to-view-orientation ..... 79, 250  
 apply-ncs-to-view-orientation-and-screen-centre ..... 79, 250  
 apply-redo ..... 84, 255  
 apply-undo ..... 84, 255  
 assign-fasta-sequence ..... 181, 352  
 assign-hetatms ..... 75, 246  
 assign-pir-sequence ..... 181, 352  
 assign-sequence-from-file ..... 181, 352  
 assign-sequence-from-string ..... 181, 352  
 associate-pir-file ..... 231  
 associate-pir-with-molecule-gui ..... 367  
 atom-spec-to-atom-index ..... 123, 294  
 atom-specs ..... 231  
 auto-fit-best-rotamer ..... 163, 334  
 auto-fit-rotamer-active-residue ..... 219  
 auto-read-do-difference-map-too-state ... 95, 266  
 auto-read-make-and-draw-maps ..... 191, 202  
 auto-read-make-and-draw-maps-from-mtz ... 191, 202  
 auto-weight-for-refinement ..... 231  
 average-temperature-factor ..... 115, 286

**B**

background-demo ..... 220  
background-is-black-p ..... 146, 317  
backup-compress-files-state ..... 85, 256  
backup-state ..... 84, 255  
BALL\_AND\_STICK ..... 233  
baton-build-delete-last-residue ..... 168, 339  
baton-tip-previous ..... 168, 339  
baton-tip-try-another ..... 168, 339  
blob-under-pointer-to-screen-centre .. 82, 253  
brief-atom-labels-state ..... 143, 314  
brighten-map ..... 230  
brighten-maps ..... 230  
browser-url ..... 184, 355

**C**

cache-or-net-get-image ..... 369  
cached-pisa-analysis ..... 215  
calc-phases-generic ..... 86, 257  
center-atom-label-status ..... 368  
center-of-mass ..... 368  
centre-of-mass ..... 230  
chain-fragments-scm ..... 194, 205  
chain-ids ..... 230  
chain-n-residues ..... 67, 238  
change-contour-level ..... 91, 262  
change-model-molecule-representation-mode  
..... 155, 326  
change-residue-number ..... 127, 298  
check-chiral-volumes ..... 133, 304  
check-dir-and-get-url ..... 214  
check-for-dark-blue-density ..... 77, 248  
chiral-centre-inverter-scm-version ..... 234  
cif-file-for-comp-id ..... 223  
cis-peptides-gui ..... 367  
clear-all-fixed-atoms ..... 186, 357  
clear-all-views ..... 146, 317  
clear-and-update-model-molecule-from-file  
..... 77, 248  
clear-and-update-molecule ..... 194, 205  
clear-backup-gui ..... 216  
clear-backups-maybe ..... 216  
clear-ball-and-stick ..... 156, 327  
clear-dots ..... 158, 329  
clear-dots-by-name ..... 158, 329  
clear-pending-picks ..... 115, 286  
click-select-residues-for-jligand ..... 367  
close-float? ..... 226  
closest-atom ..... 195, 206  
closest-atom-simple-scm ..... 195, 206  
cns->coot ..... 224  
cns-file-has-phases-p ..... 95, 266  
command-in-path-or-absolute? ..... 226  
command-in-path? ..... 220, 226, 363  
contact-score-ligand ..... 222  
control-key-for-rotate-state ..... 82, 253  
coot-clear-backup-or-real-exit ..... 71, 242

coot-gui ..... 363  
coot-has-pygtk? ..... 234  
coot-listener-idle-function-proc ..... 368  
coot-menubar-menu ..... 365  
coot-mkdir ..... 228  
coot-no-state-real-exit ..... 71, 242  
coot-real-exit ..... 71, 242  
coot-replace-string ..... 225  
coot-save-state-and-exit ..... 71, 243  
coot-socket-timeout-func ..... 368  
coot-split-version-string ..... 234  
coot-thread-dispatcher ..... 369  
coot-updates-error-handler ..... 234  
cootaneer-gui ..... 366  
copy-chain ..... 175, 346  
copy-from-ncs-master-to-others ..... 175, 346  
copy-molecule ..... 70, 241  
copy-residue-range ..... 76, 247  
copy-residue-range-from-ncs-master-to-  
others ..... 176, 347  
create-pointer-atom-molecule-maybe .. 167, 338

**D**

darken-maps ..... 230  
data-resolution ..... 88, 259  
db-mainchain ..... 162, 333  
decrease-coot-tip-number ..... 222  
default-new-atoms-b-factor ..... 116, 287  
delete-all-extra-restraints ..... 135, 306  
delete-all-sequences-from-molecule .. 181, 352  
delete-atom ..... 162, 333  
delete-atom-by-active-residue ..... 232  
delete-checked-waters-baddies ..... 170, 341  
delete-coot-backup-files ..... 216  
delete-extra-restraints-for-residue .... 135,  
306  
delete-hydrogens ..... 162, 333  
delete-residue ..... 161, 332  
delete-residue-hydrogens ..... 161, 332  
delete-residue-range ..... 161, 332  
delete-residue-sidechain ..... 162, 333  
delete-residue-with-full-spec ..... 161, 332  
delete-section-teni-files ..... 363  
delete-sequence-by-chain-id ..... 182, 353  
delete-sidechain-range ..... 214  
density-at-point ..... 97, 268  
density-score-residue ..... 201, 212  
dialog-box-of-buttons-with-async-ligands  
..... 369  
dialog-box-of-buttons-with-check-button  
..... 366  
dialog-box-of-pairs-of-buttons ..... 366  
dictionaries-read ..... 193, 204  
difference-map ..... 90, 261  
difference-map-peaks ..... 141, 312  
directory-as-file-name ..... 228, 362  
directory-files ..... 220, 228, 363

directory-is-modifiable?..... 225, 362  
display-lists-for-maps-state..... 183, 354  
do-180-degree-side-chain-flip..... 165, 336  
do-anti-aliasing-state..... 72, 243  
do-find-ligands-dialog..... 148, 319  
do-GL-lighting-state..... 73, 244  
do-probe-dots-on-rotamers-and-chis-state  
..... 185, 356  
do-probe-dots-post-refine-state..... 185, 356  
do-ramachandran-plot..... 141, 312  
do-sequence-view..... 142, 313  
do-smiles-gui..... 187, 358  
do-surface..... 182, 353  
does-residue-exist-p..... 134, 305  
dots..... 157, 328  
download-binary-dialog..... 362  
dragged-refinement-steps-per-frame.. 132, 303  
draw-hydrogens-state..... 153, 324  
draw-ncs-ghosts-state..... 173, 344

## E

edit-chi-angles..... 169, 340  
enhanced-ligand-coot?..... 223  
esoteric-depth-cue-state..... 96, 267  
eval-socket-string..... 368  
every-nth..... 227  
exchange-chain-ids-for-seg-ids..... 70, 241  
execute-find-blobs..... 151, 322  
execute-find-waters-real..... 150, 321  
execute-refmac-real..... 192, 203  
export-map..... 89, 260  
export-map-fragment..... 89, 260  
export-map-fragment-with-origin-shift.... 89,  
260  
extra-params-include-weight?..... 223

## F

fascinating-clusters-gui..... 219  
ffear-angular-resolution..... 183, 354  
ffear-search..... 182, 353  
file-name-directory..... 225  
file-name-extension..... 227  
file-name-file..... 225, 362  
file-name-sans-extension..... 227  
file-selector-entry..... 366  
file-type-coords..... 66, 237  
fill-option-menu-with-coordinates-mol-  
options..... 364  
fill-option-menu-with-mol-options..... 364  
fill-option-menu-with-number-options.... 364  
fill-partial-residues..... 165, 336  
filter..... 220, 369  
filter-fileselection-filenames-state..... 66,  
237  
filter-out-waters..... 213  
find-aligned-residue-type..... 224

find-helices..... 177, 348  
find-nucleic-acids-local..... 178, 349  
find-secondary-structure..... 178, 349  
find-secondary-structure-local..... 178, 349  
find-strands..... 177, 348  
first-coords-imol..... 72, 243  
first-molecule-with-symmetry-displayed  
..... 102, 273  
first-n..... 224  
first-non-trivial-name..... 215  
first-small-coords-imol..... 72, 243  
first-unsaved-coords-imol..... 72, 243  
fit-chain..... 218  
fit-protein..... 218  
fit-protein-make-specs..... 218  
fit-residue-range-to-map-by-simplex.... 136,  
307  
fit-to-map-by-random-jiggle..... 188, 359  
fit-waters..... 218  
fix-nomenclature-errors..... 137, 308  
file-view-set-h-bond-dist-max..... 189, 360  
file-view-set-water-dist-max..... 189, 360  
fleur-cursor..... 82, 253  
flip-active-ligand..... 232  
flip-ligand..... 149, 320  
full-atom-spec-to-atom-index..... 124, 295

## G

generate-molecule-from-mmcif-by-dict-gen  
..... 221  
generic-buttons-dialog..... 366  
generic-chooser-and-entry..... 365  
generic-chooser-and-file-selector..... 365  
generic-chooser-entry-and-file-selector  
..... 365  
generic-double-entry..... 363  
generic-interesting-things..... 366  
generic-molecule-chooser..... 366  
generic-multiple-entries-with-check-button  
..... 364  
generic-number-chooser..... 366  
generic-object-is-displayed?..... 221  
generic-object-with-name..... 221  
generic-objects-gui..... 221  
generic-single-entry..... 363  
get-active-map-drag-flag..... 87, 258  
get-aniso-probability..... 110, 281  
get-atom..... 227  
get-atom-colour-from-mol-no..... 187, 358  
get-atom-from-residue..... 227  
get-bond-colour-rotation-for-molecule... 152,  
323  
get-ccp4srs-monomer-and-dictionary.. 189, 360  
get-ccp4srs-monomer-and-overlay..... 216  
get-coords-for-accession-code..... 128, 299  
get-default-bond-thickness..... 152, 323  
get-diff-map-iso-level-increment..... 91, 262

get-directory ..... 224  
 get-drug-via-wikipedia ..... 234  
 get-ebi-pdb ..... 214  
 get-eds-pdb-and-mtz ..... 214  
 get-electrostatic-surface-opacity ... 182, 353  
 get-file-mtime ..... 213  
 get-first-ncs-master-chain ..... 229  
 get-font-size ..... 114, 285  
 get-fps-flag ..... 80, 251  
 get-gdb-strings ..... 221  
 get-limit-aniso ..... 109, 280  
 get-map-radius ..... 96, 267  
 get-map-sampling-rate ..... 91, 262  
 get-mdl-latest-time ..... 216  
 get-metrics-for-ligand ..... 213  
 get-molecule-bonds-colour-map-rotation  
 ..... 109, 280  
 get-option-menu-active-item ..... 364  
 get-option-menu-active-molecule ..... 364  
 get-refmac-extra-params ..... 223  
 get-reset-b-factor-moved-atoms-state ... 116,  
 287  
 get-revision-from-string ..... 234  
 get-sbase-monomer ..... 189, 360  
 get-show-aniso ..... 109, 280  
 get-show-limit-aniso ..... 109, 280  
 get-show-symmetry ..... 100, 271  
 get-show-unit-cell ..... 107, 278  
 get-smooth-scroll ..... 113, 284  
 get-stable-release-from-coot-version ... 362  
 get-stable-release-from-server-string ... 362  
 get-symmetry ..... 191, 202  
 get-symmetry-as-calphas-state ..... 101, 272  
 get-url-str ..... 214  
 get-view-quaternion-internal ..... 79, 250  
 git-revision-count ..... 70, 242  
 gln-asn-b-factor-outliers ..... 141, 312  
 glob ..... 220, 228, 362  
 go-to-atom-alt-conf ..... 122, 293  
 go-to-atom-atom-name ..... 122, 293  
 go-to-atom-chain-id ..... 122, 293  
 go-to-atom-ins-code ..... 122, 293  
 go-to-atom-molecule-number ..... 122, 293  
 go-to-atom-residue-number ..... 122, 293  
 go-to-ligand ..... 115, 286  
 goosh-command ..... 226, 363  
 goosh-command-with-file-input ..... 226  
 graphics-comma-key-pressed-hook ..... 231  
 graphics-dot-key-pressed-hook ..... 231  
 graphics-draw ..... 110, 281  
 graphics-general-key-press-hook ..... 231  
 graphics-molecule-bond-type ..... 155, 326  
 graphics-n-molecules ..... 73, 244  
 graphics-to-b-factor-cas-representation  
 ..... 155, 326  
 graphics-to-b-factor-representation .... 154,  
 325

graphics-to-bonds-no-waters-representation  
 ..... 154, 325  
 graphics-to-bonds-representation .... 154, 325  
 graphics-to-ca-plus-ligands-and-sidechains-  
 representation ..... 154, 325  
 graphics-to-ca-plus-ligands-representation  
 ..... 154, 325  
 graphics-to-ca-plus-ligands-sec-struct-  
 representation ..... 154, 325  
 graphics-to-ca-representation ..... 153, 324  
 graphics-to-occupancy-representation .... 155,  
 326  
 graphics-to-rainbow-representation .. 154, 325  
 graphics-to-sec-struct-bonds-representation  
 ..... 154, 325  
 graphics-to-user-defined-atom-colours-all-  
 atoms-representation ..... 155, 326  
 graphics-to-user-defined-atom-colours-  
 representation ..... 155, 326  
 guess-refinement-map ..... 231

## H

handle-cns-data-file ..... 140, 311  
 handle-cns-data-file-with-cell ..... 140, 311  
 handle-drag-and-drop-string ..... 199, 210  
 handle-online-coot-search-request ... 184, 355  
 handle-read-ccp4-map ..... 118, 289  
 handle-read-draw-molecule ..... 74, 245  
 handle-read-draw-molecule-and-move-  
 molecule-here ..... 75, 246  
 handle-read-draw-molecule-with-recenter  
 ..... 368  
 handle-read-draw-molecule-with-recentre  
 ..... 75, 246  
 handle-read-draw-probe-dots ..... 184, 355  
 handle-read-draw-probe-dots-unformatted  
 ..... 184, 355  
 handle-smiles-go ..... 363  
 hardware-stereo-angle-factor-state .. 111, 282  
 hardware-stereo-mode ..... 111, 282  
 has-unit-cell-state ..... 102, 273  
 have-unsaved-changes-p ..... 85, 256  
 het-group-n-atoms ..... 76, 247  
 hetify-residue ..... 75, 246  
 hide-main-toolbar ..... 80, 251  
 hide-modelling-toolbar ..... 80, 251  
 highlight-binding-site ..... 233  
 hole ..... 199, 210  
 hydrogenate-region ..... 196, 207

## I

ideal-nucleic-acid ..... 179, 350  
 identity-matrix ..... 229  
 idle-function-rotate-angle ..... 74, 245  
 imol-refinement-map ..... 133, 304  
 import-from-3d-generator-from-mdl ..... 216

- import-python-module ..... 128, 299
  - increment-coot-tip-number ..... 222
  - info-dialog ..... 99, 270
  - info-dialog-and-text ..... 99, 270
  - insert-action-view-after-view ..... 145, 316
  - insertion-code-from-serial-number ... 67, 239
  - interactive-probe ..... 221
  - interactive-probe-dots-molprobity-radius  
..... 185, 356
  - interesting-things-gui ..... 364
  - interesting-things-with-fix-maybe ..... 364
  - interruptible-fit-protein ..... 218
  - invert-chiral-center ..... 368
  - is-closed-generic-object? ..... 221
  - is-difference-map? ..... 230
  - is-directory? ..... 228
  - is-mtz-file-p ..... 95, 266
  - is-nucleotide-chain-p ..... 69, 240
  - is-protein-chain-p ..... 68, 240
  - is-solvent-chain-p ..... 68, 239
  - is-solvent-chain? ..... 230
- K**
- keep-map-colour-after-refmac-state.. 100, 271
- L**
- label-all-active-residue-atoms ..... 233
  - label-all-atoms-in-residue ..... 233
  - label-all-CAs ..... 233
  - laplacian ..... 187, 358
  - launch-jligand-function ..... 367
  - lengthen-baton ..... 168, 339
  - libcheck-exe ..... 362
  - ligand-expert ..... 148, 319
  - list-of-strings? ..... 225
  - load-all-scheme ..... 218
  - load-annotations ..... 233
  - load-default-sequence ..... 234
  - lsq-improve ..... 189, 360
  - lsq-match ..... 235
  - lsq-match-type-symbol->number ..... 235
  - lsqman-count ..... 220
- M**
- maf ..... 214
  - make-and-draw-map ..... 93, 264
  - make-and-draw-map-with-refmac-params ..... 93,  
264
  - make-and-draw-map-with-reso-with-refmac-  
params ..... 94, 265
  - make-backup ..... 84, 255
  - make-ball-and-stick ..... 156, 327
  - make-difference-map-gui ..... 366
  - make-directory-maybe ..... 64, 235
  - make-dynamically-transformed-ncs-maps... 173,  
344
  - make-gdb-script ..... 221
  - make-gui ..... 221
  - make-latest-version-url ..... 234
  - make-link ..... 199, 210
  - make-socket-listener-maybe ..... 183, 354
  - manage-column-selector ..... 67, 238
  - manual-ncs-ghosts ..... 217
  - manual-refine-residues ..... 219
  - map-colour-components ..... 191, 202
  - map-from-mtz-by-calc-phases ..... 86, 257
  - map-from-mtz-by-refmac-calc-phases ... 86, 257
  - map-is-difference-map ..... 96, 267
  - map-is-displayed ..... 126, 297
  - map-line-width-state ..... 93, 264
  - map-mask-atom-radius ..... 170, 341
  - map-mean-scm ..... 201, 212
  - map-molecule-chooser-gui ..... 365
  - map-molecule-list ..... 225
  - map-statistics-scm ..... 201, 212
  - map-to-model-correlation ..... 200, 211
  - map-to-model-correlation-per-residue... 200,  
211
  - map-to-model-correlation-stats ..... 200, 211
  - mask-map-by-molecule ..... 169, 340
  - match-ligand-atom-names ..... 148, 319
  - match-ligand-atom-names-to-comp-id.. 149, 320
  - matrix->quaternion ..... 228
  - matrix-state ..... 130, 301
  - max-water-distance ..... 150, 321
  - median-temperature-factor ..... 115, 286
  - member? ..... 226
  - merge-solvent-chains ..... 232
  - min-max-residues-from-atom-specs ..... 367
  - missing-atoms-gui ..... 365
  - mmcif-sfs-to-mtz ..... 72, 243
  - model-fit-refine-dialog-stays-on-top-state  
..... 81, 252
  - model-molecule-list ..... 225
  - model-resolution ..... 89, 260
  - mol-is-active ..... 125, 296
  - mol-is-displayed ..... 125, 296
  - molecule-centre ..... 228
  - molecule-centres-gui ..... 364
  - molecule-chooser-gui ..... 365
  - molecule-chooser-gui-generic ..... 365
  - molecule-has-hydrogens? ..... 224
  - molecule-name ..... 71, 242
  - molecule-number-list ..... 224
  - molecules-matching-criteria ..... 219
  - mono-mode ..... 111, 282
  - monomer-molecule-from-3-let-code ..... 362
  - most-recently-created-file ..... 225
  - move-hetgroups-to-around-protein ... 150, 321
  - move-molecule-here ..... 229
  - move-molecule-to-screen-center ..... 229
  - move-molecule-to-screen-centre ..... 229

move-waters-to-around-protein..... 150, 321  
mtz-file-has-phases-p..... 94, 265  
mtz-file-name..... 196, 207  
mtz-file-name->refinement-column-labels  
..... 213  
mtz-fp-for-map..... 97, 268  
mtz-hklin-for-map..... 97, 268  
mtz-phi-for-map..... 98, 269  
mtz-use-weight-for-map..... 98, 269  
mtz-weight-for-map..... 98, 269  
multi-chicken..... 233  
multi-mutate..... 213  
multi-read-pdb..... 227  
multi-residue-torsion-fit-scm..... 191, 202  
mutate..... 165, 336  
mutate-and-auto-fit..... 214  
mutate-and-autofit-residue-range..... 214  
mutate-auto-fit-do-post-refine-state.... 166,  
337  
mutate-base..... 166, 337  
mutate-by-overlap..... 232  
mutate-chain..... 213  
mutate-residue-range..... 213  
mutate-residue-redundant..... 214  
mutate-single-residue-by-serial-number  
..... 166, 337  
my-button-callback..... 218  
my-top-level..... 218

## N

n-chains..... 68, 239  
n-dots-sets..... 158, 329  
n-models..... 68, 239  
n-residues..... 69, 240  
n-rotamers..... 164, 335  
n-symops..... 104, 275  
ncs-control-change-ncs-master-to-chain  
..... 176, 347  
ncs-control-change-ncs-master-to-chain-id  
..... 176, 347  
ncs-control-display-chain..... 176, 347  
ncs-ligand..... 217  
ncs-master-chain-id..... 217  
ncs-update-ghosts..... 173, 344  
new-molecule-by-atom-selection..... 179, 350  
new-molecule-by-residue-type-selection  
..... 179, 350  
new-molecule-by-smiles-string..... 216  
new-molecule-by-sphere-selection.... 179, 350  
new-molecule-by-symmetry..... 103, 274  
new-molecule-by-symmetry-with-atom-  
selection..... 103, 274  
new-molecule-by-symop..... 104, 275  
no-coot-tips..... 223  
normal-cursor..... 82, 253  
notify-of-new-version..... 362  
nudge-screen-centre-gui..... 366

number-list..... 226

## O

old-coot?..... 364  
open-cif-dictionary-file-selector-dialog  
..... 140, 311  
open-coords-dialog..... 66, 238  
open-coot-listener-socket..... 367  
open-coot-listener-socket-with-timeout.. 368  
output-atom-info-as-text..... 137, 308  
overlay-my-ligands..... 232

## P

parse-pisa-assemblies..... 215  
parse-pisa-interfaces..... 215  
pdbe-get-pdb-and-sfs-cif..... 370  
pdbe-latest-releases-gui..... 370  
pepflip..... 159, 330  
pepflip-active-residue..... 219  
phosphorylate-active-residue..... 232  
pick-cursor-maybe..... 82, 253  
pir-file-name->pir-sequence..... 231  
pisa-assemblies-xml..... 214  
pisa-handle-sxml-molecule..... 214  
pisa-interaction..... 188, 359  
place-helix-here..... 177, 348  
place-strand-here..... 177, 348  
place-strand-here-dialog..... 177, 348  
place-text..... 187, 358  
pointer-atom-molecule..... 167, 338  
post-display-control-window..... 124, 295  
post-go-to-atom-window..... 122, 293  
post-ligand-fit-gui..... 219  
post-manipulation-hook..... 224  
post-model-fit-refine-dialog..... 83, 254  
post-other-modelling-tools-dialog.... 83, 254  
post-python-scripting-window..... 128, 299  
post-scheme-scripting-window..... 127, 298  
post-scripting-window..... 127, 298  
povray-args..... 369  
povray-image..... 369  
povray-version..... 369  
pre-release?..... 224  
prefer-python..... 64, 235  
prep-for-pisa..... 215  
print-all-history-in-python..... 105, 276  
print-all-history-in-scheme..... 105, 276  
print-glyco-tree..... 197, 208  
print-header-secondary-structure-info.... 69,  
241  
print-molecule-names..... 233  
print-sequence..... 231  
print-sequence-chain..... 180, 351  
print-view-matrix..... 79, 250  
probe..... 221  
probe-available-p..... 127, 298

probe-local-sphere ..... 222  
prodrgr-flat ..... 216  
prodrgr-ify ..... 233  
prodrgr-xyzin ..... 216  
protein-db-loops ..... 198, 209  
pukka-puckers? ..... 233  
python-at-prompt-at-startup-state .... 73, 244

## Q

qq-plot-map-and-model-scm ..... 201, 212  
quick-save ..... 99, 270

## R

ramachandran-plot-differences ..... 142, 313  
ramachandran-plot-differences-by-chain  
..... 142, 313  
range ..... 226  
rapper-dir ..... 223  
raster-screen-shot ..... 171, 342  
raster3d ..... 170, 341  
raytrace ..... 217  
read-pdb ..... 75, 246  
read-pdb-all ..... 227  
read-phs-and-coords-and-make-map .... 119, 290  
read-phs-and-make-map-using-cell-symm... 119,  
290  
read-phs-and-make-map-using-cell-symm-from-  
mol ..... 119, 290  
read-phs-and-make-map-using-cell-symm-from-  
previous-mol ..... 119, 290  
read-phs-and-make-map-with-reso-limits  
..... 120, 291  
read-shelx-ins-file ..... 140, 311  
read-shelx-lst-file ..... 215  
read-vu-file ..... 231  
reattach-modelling-toolbar ..... 81, 252  
recentre-on-read-pdb ..... 114, 285  
recover-session ..... 86, 257  
redraw-background ..... 146, 317  
reduce-on-pdb-file ..... 221  
refine-active-residue ..... 219  
refine-active-residue-generic ..... 219  
refine-active-residue-triple ..... 219  
refine-auto-range ..... 131, 302  
refine-with-torsion-restraints-state ... 130,  
301  
refine-zone ..... 131, 302  
refine-zone-atom-index-define ..... 131, 302  
refine-zone-with-full-residue-spec-scm  
..... 196, 207  
refinement-immediate-replacement-state  
..... 129, 300  
refinement-move-atoms-with-zero-occupancy-  
state ..... 84, 255  
refinement-refine-per-frame-state ... 132, 303  
refmac-calc-sfs-make-mtz-with-columns ... 369

refmac-exe ..... 222  
refmac-extra-params ..... 222  
refmac-name ..... 192, 203  
regularize-residues ..... 196, 207  
regularize-zone ..... 132, 303  
reinterp-map ..... 90, 261  
remarks-dialog ..... 69, 241  
remove-coordinates-glob-extension .... 65, 236  
remove-data-glob-extension ..... 65, 237  
remove-dictionary-glob-extension ..... 65, 237  
remove-map-glob-extension ..... 65, 237  
remove-named-view ..... 145, 316  
remove-omega-torsion-restraints ..... 129, 300  
remove-planar-peptide-restraints .... 129, 300  
remove-text ..... 188, 359  
remove-view ..... 145, 316  
rename-residue-gui ..... 367  
render-image ..... 217  
renumber-residue-range ..... 127, 298  
renumber-waters ..... 150, 321  
reorder-chains ..... 229  
replace-fragment ..... 76, 247  
res-spec->chain-id ..... 232  
res-spec->ins-code ..... 232  
res-spec->res-no ..... 232  
reset-view ..... 73, 244  
residue-alt-confs ..... 232  
residue-atom->alt-conf ..... 225  
residue-atom->atom-name ..... 225  
residue-atom->occupancy ..... 225  
residue-density-fit-scale-factor ..... 97, 268  
residue-exists? ..... 230  
residue-has-hetatms ..... 76, 247  
residue-has-hetatms? ..... 230  
residue-info ..... 194, 205  
residue-info-dialog ..... 138, 309  
residue-name ..... 193, 204  
residue-range-gui ..... 367  
residue-spec->atom-for-centre ..... 232  
residue-spec->atom-selection-string ..... 225  
residue-spec->chain-id ..... 225  
residues-matching-criteria ..... 232  
residues-near-position-scm ..... 195, 206  
residues-near-residue ..... 195, 206  
residues-with-alt-confs ..... 232  
rename-from-serial-number ..... 193, 204  
restore-previous-map-colour ..... 87, 258  
rigid-body-refine-by-residue-ranges ..... 224  
rigid-body-refine-zone ..... 159, 330  
rotamer-auto-fit-do-post-refine-state... 166,  
337  
rotate-about-screen-axis ..... 229  
rotate-cursor ..... 82, 253  
rotate-x-scene ..... 143, 314  
rotate-y-scene ..... 143, 314  
rotate-z-scene ..... 143, 314  
rotation-centre ..... 226  
run-command/strings ..... 220, 226, 363

- run-download-binary-curl ..... 234  
run-gtk-pending-events ..... 363  
run-guile-script ..... 128, 299  
run-mogul ..... 220  
run-mogul-show-results ..... 220  
run-prosmart ..... 369  
run-python-script ..... 128, 299  
run-refmac-by-filename ..... 223  
run-refmac-for-phases ..... 223  
run-script ..... 128, 299  
run-state-file ..... 106, 277  
run-state-file-maybe ..... 106, 277
- S**
- sanitise-alt-confs ..... 233  
sanitise-alt-confs-active-residue ..... 233  
sanitise-alt-confs-in-residue ..... 233  
save-annotations ..... 233  
save-coordinates ..... 118, 289  
save-dialog-positions-to-init-file ..... 233  
save-previous-map-colour ..... 87, 258  
save-state ..... 106, 277  
save-state-file ..... 106, 277  
save-state-file-py ..... 106, 277  
save-symmetry-coords ..... 102, 273  
save-views ..... 146, 317  
scale-cell ..... 186, 357  
scale-zoom ..... 139, 310  
scheme-representation ..... 229  
score-residue-range-fit-to-map ..... 137, 308  
score-rotamers-scm ..... 198, 209  
screendump-image ..... 77, 248  
scroll-by-wheel-mouse-state ..... 78, 249  
scroll-wheel-map ..... 87, 258  
secondary-structure-restraints-type ..... 133, 304  
seqnum-from-serial-number ..... 67, 238  
set-accept-reject-dialog-docked-show ..... 81, 252  
set-accept-reject-dialog-position ... 112, 283  
set-active-map-drag-flag ..... 87, 258  
set-add-terminal-residue-default-residue-type ..... 160, 331  
set-add-terminal-residue-do-post-refine ..... 160, 331  
set-add-terminal-residue-immediate-addition ..... 159, 330  
set-alignment-gap-and-space-penalty ..... 126, 297  
set-all-maps-displayed ..... 126, 297  
set-all-models-displayed-and-active ..... 126, 297  
set-aniso-probability ..... 110, 281  
set-atom-attribute ..... 116, 287  
set-atom-string-attribute ..... 116, 287  
set-auto-fit-best-rotamer-clash-flag .... 164, 335  
set-auto-read-column-labels ..... 95, 266  
set-b-factor-bonds-scale-factor ..... 155, 326  
set-background-colour ..... 146, 317  
set-backup-compress-files ..... 85, 256  
set-baton-build-params ..... 168, 339  
set-baton-mode ..... 167, 338  
set-bond-colour-rotation-for-molecule... 152, 323  
set-bond-thickness ..... 152, 323  
set-bond-thickness-intermediate-atoms... 152, 323  
set-brief-atom-labels ..... 143, 314  
set-browser-interface ..... 184, 355  
set-button-label-for-external-refinement ..... 187, 358  
set-clipping-back ..... 107, 278  
set-clipping-front ..... 107, 278  
set-colour-by-chain ..... 108, 279  
set-colour-by-molecule ..... 108, 279  
set-colour-map-rotation-for-map ..... 109, 280  
set-colour-map-rotation-on-read-pdb ..... 108, 279  
set-colour-map-rotation-on-read-pdb-c-only-flag ..... 108, 279  
set-colour-map-rotation-on-read-pdb-flag ..... 108, 279  
set-console-display-commands-hilights... 105, 276  
set-console-display-commands-state .. 105, 276  
set-contour-by-sigma-step-by-mol ..... 88, 259  
set-control-key-for-rotate ..... 82, 253  
set-convert-to-v2-atom-names ..... 74, 245  
set-default-bond-thickness ..... 152, 323  
set-default-initial-contour-level-for-difference-map ..... 78, 249  
set-default-initial-contour-level-for-map ..... 78, 249  
set-default-representation-type ..... 152, 323  
set-default-temperature-factor-for-new-atoms ..... 116, 287  
set-delete-dialog-position ..... 112, 283  
set-density-size ..... 95, 266  
set-diff-map-iso-level-increment ..... 90, 261  
set-diff-map-iso-level-increment-from-text ..... 91, 262  
set-display-control-dialog-position .... 112, 283  
set-display-intro-string ..... 95, 266  
set-display-lists-for-maps ..... 183, 354  
set-display-only-model-mol ..... 125, 296  
set-do-anti-aliasing ..... 72, 243  
set-do-GL-lighting ..... 72, 244  
set-do-probe-dots-on-rotamers-and-chis ..... 184, 355  
set-do-probe-dots-post-refine ..... 185, 356  
set-dots-color ..... 368  
set-dots-colour ..... 158, 329

- set-dragged-refinement-steps-per-frame  
..... 132, 303
- set-draw-cis-peptide-markups ..... 153, 324
- set-draw-crosshairs ..... 168, 339
- set-draw-hydrogens ..... 153, 324
- set-draw-map-standard-lines ..... 77, 248
- set-draw-ncs-ghosts ..... 173, 344
- set-draw-solid-density-surface ..... 77, 248
- set-draw-stick-mode-atoms ..... 153, 324
- set-draw-zero-occ-markers ..... 153, 324
- set-edit-chi-angles-dialog-position .... 113, 284
- set-environment-distances-distance-limits  
..... 139, 310
- set-esoteric-depth-cue ..... 96, 267
- set-extra-restraints-prosmart-sigma-limits  
..... 135, 306
- set-ffear-angular-resolution ..... 183, 354
- set-file-chooser-selector ..... 66, 238
- set-filter-fileselection-filenames ... 66, 237
- set-find-hydrogen-torsion ..... 223
- set-find-hydrogen-torsions ..... 169, 340
- set-find-ligand-here-cluster ..... 148, 319
- set-find-ligand-mask-waters ..... 147, 318
- set-find-ligand-multi-solutions-per-cluster  
..... 147, 318
- set-find-ligand-n-top-ligands ..... 147, 318
- set-fix-chiral-volumes-before-refinement  
..... 133, 304
- set-flat-shading-for-solid-density-surface  
..... 78, 249
- set-font-color ..... 368
- set-font-colour ..... 114, 285
- set-font-size ..... 114, 285
- set-fps-flag ..... 80, 251
- set-go-to-atom-chain-residue-atom-name  
..... 122, 293
- set-go-to-atom-chain-residue-atom-name-full  
..... 123, 294
- set-go-to-atom-chain-residue-atom-name-no-  
redraw ..... 123, 294
- set-go-to-atom-molecule ..... 124, 295
- set-go-to-atom-window-position ..... 112, 283
- set-go-to-ligand-n-atoms-limit ..... 115, 286
- set-graphics-window-position ..... 110, 281
- set-graphics-window-size ..... 110, 281
- set-grey-carbon-colour ..... 156, 327
- set-hardware-stereo-angle-factor .... 111, 282
- set-have-unsaved-changes ..... 84, 255
- set-idle-function-rotate-angle ..... 74, 245
- set-imol-refinement-map ..... 134, 305
- set-interactive-probe-dots-molprobity-  
radius ..... 185, 356
- set-iso-level-increment ..... 90, 261
- set-keep-map-colour-after-refmac .... 100, 271
- set-kleywegt-plot-n-diffs ..... 141, 312
- set-last-map-color ..... 368
- set-last-map-colour ..... 88, 259
- set-last-map-contour-level ..... 91, 262
- set-last-map-contour-level-by-sigma .. 92, 263
- set-last-map-sigma-step ..... 88, 259
- set-ligand-acceptable-fit-fraction .. 146, 317
- set-ligand-cluster-sigma-level ..... 147, 318
- set-ligand-flexible-ligand-n-samples ... 147, 318
- set-ligand-search-map-molecule ..... 148, 319
- set-ligand-search-protein-molecule .. 147, 318
- set-ligand-water-n-cycles ..... 151, 322
- set-ligand-water-to-protein-distance-limits  
..... 151, 322
- set-limit-aniso ..... 109, 280
- set-main-window-title ..... 66, 238
- set-map-colour ..... 88, 259
- set-map-correlation-atom-radius ..... 200, 211
- set-map-displayed ..... 125, 296
- set-map-is-difference-map ..... 96, 267
- set-map-line-width ..... 92, 263
- set-map-mask-atom-radius ..... 169, 340
- set-map-radius ..... 95, 266
- set-map-sampling-rate ..... 91, 262
- set-map-sampling-rate-text ..... 91, 262
- set-map-sharpening-scale-limit ..... 185, 356
- set-match-element ..... 235
- set-matrix ..... 130, 301
- set-max-skeleton-search-depth ..... 118, 289
- set-model-fit-refine-dialog-position ... 111, 282
- set-model-fit-refine-dialog-stays-on-top  
..... 81, 252
- set-model-fit-refine-place-atom-at-pointer-  
label ..... 83, 254
- set-model-fit-refine-rotate-translate-zone-  
label ..... 83, 254
- set-model-toolbar-docked-position .... 81, 252
- set-mol-active ..... 125, 296
- set-mol-displayed ..... 125, 296
- set-molecule-bonds-colour-map-rotation  
..... 109, 280
- set-molecule-name ..... 71, 242
- set-monomer-restraints ..... 193, 204
- set-mutate-auto-fit-do-post-refine .. 166, 337
- set-ncs-ghost-bond-thickness ..... 173, 344
- set-ncs-homology-level ..... 175, 346
- set-nomenclature-errors-on-read ..... 137, 308
- set-pick-cursor-index ..... 83, 254
- set-prefer-python ..... 64, 235
- set-ramachandran-plot-background-block-size  
..... 142, 313
- set-ramachandran-plot-contour-levels ... 142, 313
- set-ramachandran-plot-dialog-position... 112, 283
- set-raster3d-atom-radius ..... 171, 342
- set-raster3d-bond-thickness ..... 171, 342
- set-raster3d-bone-thickness ..... 171, 342
- set-raster3d-density-thickness ..... 171, 342

- set-raster3d-shadows-enabled ..... 171, 342
- set-raster3d-water-sphere ..... 171, 342
- set-recentre-on-read-pdb ..... 114, 285
- set-refine-auto-range-step ..... 130, 301
- set-refine-max-residues ..... 131, 302
- set-refine-ramachandran-angles ..... 133, 304
- set-refine-with-torsion-restraints.. 130, 301
- set-refinement-drag-elasticity ..... 132, 303
- set-refinement-immediate-replacement... 129, 300
- set-refinement-move-atoms-with-zero-occupancy ..... 83, 254
- set-refinement-refine-per-frame ..... 132, 303
- set-refmac-counter ..... 99, 270
- set-renderer-show-atoms ..... 171, 342
- set-reset-b-factor-moved-atoms ..... 116, 287
- set-residue-density-fit-scale-factor .... 92, 263
- set-residue-name ..... 117, 288
- set-residue-selection-flash-frames-number ..... 129, 300
- set-residue-to-rotamer-name ..... 164, 335
- set-residue-to-rotamer-number ..... 164, 335
- set-residue-type-chooser-stub-state .... 167, 338
- set-rigid-body-fit-acceptable-fit-fraction ..... 159, 330
- set-rocking-factors ..... 74, 245
- set-rotamer-auto-fit-do-post-refine .... 166, 337
- set-rotamer-check-clashes ..... 163, 334
- set-rotamer-lowest-probability ..... 163, 334
- set-rotamer-search-mode ..... 163, 334
- set-rotamer-selection-dialog-position... 113, 284
- set-rotate-translate-dialog-position... 112, 283
- set-rotation-center ..... 368
- set-rotation-center-size ..... 368
- set-rotation-centre ..... 115, 286
- set-rotation-centre-size ..... 114, 285
- set-run-state-file-status ..... 106, 277
- set-save-coordinates-in-original-directory ..... 119, 290
- set-save-state-file-name ..... 106, 277
- set-scroll-by-wheel-mouse ..... 78, 249
- set-scroll-wheel-map ..... 87, 258
- set-scrollable-map ..... 87, 258
- set-secondary-structure-restraints-type ..... 133, 304
- set-show-aniso ..... 110, 281
- set-show-chiral-volume-errors-dialog... 133, 304
- set-show-environment-distances ..... 138, 309
- set-show-environment-distances-bumps ... 138, 309
- set-show-environment-distances-h-bonds ..... 138, 309
- set-show-graphics-ligand-view ..... 190, 361
- set-show-origin-marker ..... 80, 251
- set-show-paths-in-display-manager .... 64, 236
- set-show-pointer-distances ..... 139, 310
- set-show-strict-ncs ..... 175, 346
- set-show-symmetry-master ..... 100, 271
- set-show-symmetry-molecule ..... 100, 271
- set-show-unit-cell ..... 108, 279
- set-show-unit-cells-all ..... 107, 278
- set-skeleton-box-size ..... 118, 289
- set-skeleton-colour ..... 118, 289
- set-smooth-scroll-do-zoom ..... 139, 310
- set-smooth-scroll-flag ..... 113, 284
- set-smooth-scroll-limit ..... 113, 284
- set-smooth-scroll-steps ..... 113, 284
- set-socket-python-string-waiting .... 183, 354
- set-socket-string-waiting ..... 183, 354
- set-solid-density-surface-opacity .... 78, 249
- set-space-group ..... 105, 276
- set-sticky-sort-by-date ..... 66, 237
- set-stop-scroll-diff-map ..... 92, 263
- set-stop-scroll-diff-map-level ..... 92, 263
- set-stop-scroll-iso-map ..... 92, 263
- set-stop-scroll-iso-map-level ..... 92, 263
- set-swap-difference-map-colors ..... 368
- set-swap-difference-map-colours ..... 96, 267
- set-symmetry-atom-labels-expanded ... 102, 273
- set-symmetry-colour ..... 108, 279
- set-symmetry-colour-by-sympop ..... 101, 272
- set-symmetry-molecule-rotate-colour-map ..... 101, 272
- set-symmetry-shift-search-size ..... 105, 276
- set-symmetry-size ..... 100, 271
- set-symmetry-whole-chain ..... 101, 272
- set-transparent-electrostatic-surface... 182, 353
- set-undo-molecule ..... 85, 256
- set-unpathed-backup-file-names ..... 85, 256
- set-use-grey-carbons-for-molecule ... 156, 327
- set-use-only-extra-torsion-restraints-for-torsions ..... 136, 307
- set-use-stroke-characters ..... 114, 285
- set-use-trans-peptide-restraints .... 129, 300
- set-view-matrix ..... 228
- set-view-quaternion ..... 79, 250
- set-virtual-trackball-type ..... 225
- set-water-check-spherical-variance-limit ..... 151, 322
- setup-mutate-auto-fit ..... 165, 336
- setup-torsion-general ..... 169, 340
- sharpen ..... 185, 356
- shell-command-to-file-with-data ..... 226
- shell-command-to-string ..... 220, 226
- shelx-molecule? ..... 225, 230
- shelxl-refine ..... 215
- shelxl-refine-primitive ..... 215
- shorten-baton ..... 168, 339
- show-coot-tip-from-list ..... 222

show-environment-distances-state .... 138, 309  
show-main-toolbar..... 80, 251  
show-model-toolbar-all-icons ..... 80, 251  
show-model-toolbar-main-icons ..... 81, 252  
show-modelling-toolbar ..... 80, 251  
show-origin-marker-state..... 80, 251  
show-partial-charge-info..... 186, 357  
show-paths-in-display-manager-state.. 64, 236  
show-pointer-distances-state..... 139, 310  
show-remarks-browser ..... 70, 242  
show-restraints-editor..... 136, 307  
show-restraints-editor-by-index..... 136, 307  
show-select-map-dialog..... 83, 254  
show-set-undo-molecule-chooser..... 85, 256  
show-spacegroup..... 126, 297  
show-strict-ncs-state..... 175, 346  
side-by-side-stereo-mode..... 111, 282  
simple-lsq-match..... 235  
single-manual-ncs-ghost ..... 217  
single-model-view-model-number..... 190, 361  
single-model-view-next-model-number.... 190, 361  
single-model-view-prev-model-number.... 190, 361  
single-model-view-this-model-number.... 190, 361  
skeletonize-map..... 117, 288  
skip-to-next-ncs-chain..... 217  
slash-start?..... 227  
smiles-gui ..... 363  
smooth-map..... 90, 261  
smooth-scroll-do-zoom..... 139, 310  
sort-chains..... 69, 240  
sort-residues ..... 69, 240  
spin-search..... 197, 208  
spin-zoom-trans..... 143, 314  
split-water..... 151, 322  
standard-deviation-temperature-factor... 115, 286  
start-graphics-interface..... 73, 244  
start-jligand-listener..... 213  
stepped-refine-protein..... 218  
stepped-refine-protein-for-rama ..... 219  
stepped-refine-protein-with-refine-func  
..... 219  
stereo-mode-state..... 111, 282  
store-graphics-window-position..... 110, 281  
string->list-of-strings..... 220, 228, 363  
string-append-with-spaces..... 226  
string-append-with-string..... 227  
string-concatenate ..... 220, 227  
string-member?..... 226  
strip-extension..... 227, 362  
strip-leading-spaces..... 227  
strip-path..... 227  
strip-spaces ..... 227  
suck-model-fit-dialog..... 81, 252  
superpose..... 172, 343

superpose-with-atom-selection..... 172, 343  
superpose-with-chain-selection..... 172, 343  
svn-revision..... 71, 242  
swap-map-colours..... 99, 270  
symmetry-as-calphas ..... 100, 271  
symmetry-molecule-rotate-colour-map-state  
..... 101, 272

## T

target-auto-weighting-value..... 231  
text-index-near-position..... 188, 359  
tip-list..... 223  
tips-gui..... 222  
toggle-active-mol ..... 229  
toggle-display-map..... 229  
toggle-display-mol ..... 229  
toggle-idle-ligand-interactions ..... 223  
toggle-idle-rock-function..... 74, 245  
toggle-idle-spin-function..... 73, 245  
transform-coords-molecule..... 229  
transform-map..... 229  
transform-map-using-lsq-matrix ..... 230  
transform-molecule-by..... 120, 291  
transform-zone..... 121, 292  
translate-molecule-by..... 120, 291  
translate-scene-x..... 144, 315  
translate-scene-y..... 144, 315  
translate-scene-z..... 144, 315  
translation..... 229  
trim-molecule-by-map..... 170, 341  
try-set-draw-baton ..... 167, 338  
turn-off-backup..... 84, 255  
turn-on-backup..... 84, 255

## U

undo-last-move..... 120, 291  
undo-symmetry-view ..... 102, 273  
unique-date/time-str ..... 227  
unmangle-hydrogen-name..... 185, 356  
unpathed-backup-file-names-state..... 85, 256  
unset-dots-colour..... 158, 329  
unset-model-fit-refine-dialog..... 83, 254  
unset-refine-params-dialog ..... 83, 254  
unset-sticky-sort-by-date..... 66, 237  
unskeletonize-map..... 117, 288  
update-go-to-atom-from-current-atom..... 232  
update-go-to-atom-from-current-position  
..... 123, 294  
update-go-to-atom-window-on-changed-mol  
..... 124, 295  
update-go-to-atom-window-on-new-mol.... 124, 295  
update-ncs-ghosts-by-local-sphere..... 217  
update-self..... 234  
use-graphics-interface-state..... 73, 244

use-only-extra-torsion-restraints-for-torsions-state..... 136, 307  
useful-glob..... 228, 362  
user-mods-gui..... 367

## V

valid-labels..... 94, 265  
valid-map-molecule?..... 230  
valid-model-molecule?..... 230  
valid-refinement-map?..... 230  
view-matrix..... 228  
view-quaternion..... 228  
view-saver-gui..... 366  
views-panel-gui..... 366  
vt-surface..... 107, 278  
vt-surface-status..... 107, 278

## W

water-chain-from-shelx-ins-scm..... 197, 208

water-chain-scm..... 197, 208  
water-coordination-gui..... 367  
watson-crick-pair..... 180, 351  
watson-crick-pair-for-residue-range..... 180, 351  
wrapped-create-find-waters-dialog... 150, 321  
write-ccp4mg-picture-description.... 186, 357  
write-interpolated-extra-restraints.... 135, 306  
write-interpolated-models-and-extra-restraints..... 135, 306  
write-pdb-file..... 98, 269  
write-reduce-het-dict..... 221  
write-refmac-parallel-plane-restraint... 369  
write-residue-range-to-pdb-file..... 98, 269  
write-restraints-cif-dictionary..... 136, 307  
write-shelx-ins-file..... 141, 312

## Z

zalman-stereo-mode..... 111, 282  
zoom-factor..... 139, 310