

---

# FRED RECEPTOR

2.2.5

OpenEye Scientific Software, Inc.

March 17, 2009

9 Bisbee Ct, Suite D  
Santa Fe, NM 87508  
[www.eyesopen.com](http://www.eyesopen.com)  
[support@eyesopen.com](mailto:support@eyesopen.com)

Copyright © 1997-2006 OpenEye Scientific Software, Santa Fe, New Mexico. All rights reserved.

All rights reserved. This material contains proprietary information of OpenEye Scientific Software. Use of copyright notice is precautionary only and does not imply publication or disclosure.

The information supplied in this document is believed to be true but no liability is assumed for its use or the infringement of the rights of others resulting from its use. Information in this document is subject to change without notice and does not represent a commitment on the part of OpenEye Scientific Software.

This package is sold/licensed/distributed subject to the condition that it shall not, by way of trade or otherwise, be lent, re-sold, hired out or otherwise circulated without OpenEye Scientific Software's prior consent, in any form of packaging or cover other than that in which it was produced. No part of this manual or accompanying documentation, may be reproduced, stored in a retrieval system on optical or magnetic disk, tape, CD, DVD or other medium, or transmitted in any form or by any means, electronic, mechanical, photocopying recording or otherwise for any purpose other than for the purchaser's personal use without a legal agreement or other written permission granted by OpenEye.

This product should not be used in the planning, construction, maintenance, operation or use of any nuclear facility nor the flight, navigation or communication of aircraft or ground support equipment. OpenEye Scientific software, shall not be liable, in whole or in part, for any claims arising from such use, including death, bankruptcy or outbreak of war.

Windows is a registered trademark of Microsoft Corporation. Apple and Macintosh are registered trademarks of Apple Computer, Inc. AIX and IBM are registered trademarks of International Business Machines Corporation. UNIX is a registered trademark of the Open Group. RedHat is a registered trademark of RedHat, Inc. Linux is a registered trademark of Linus Torvalds. Alpha is a trademark of Digital Equipment Corporation. SPARC is a registered trademark of SPARC International Inc.

SYBYL is a registered trademark of TRIPOS, Inc. MDL is a registered trademark and ISIS is a trademark of MDL Information Systems, Inc. SMILES, SMARTS, and SMIRKS may be trademarks of Daylight Chemical Information Systems. Macromodel is a trademark of Schrödinger, Inc. Schrödinger, Inc may be a wholly owned subsidiary of the Columbia University, New York.

Python is a trademark of the Python Software Foundation.

Java is a trademark or registered trademark of Sun Microsystems, Inc. in the U.S. or other countries.

"The forefront of chemoinformatics" is a trademark of Daylight Chemical Information Systems, Inc.

Other products and software packages referenced in this document are trademarks and registered trademarks of their respective vendors or manufacturers.

# CONTENTS

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Version</b>                              | <b>1</b>  |
| 1.1      | Prototype Software . . . . .                | 1         |
| <b>2</b> | <b>Installation</b>                         | <b>2</b>  |
| 2.1      | Microsoft Windows . . . . .                 | 2         |
| 2.2      | UNIX/Linux . . . . .                        | 3         |
| 2.3      | Mac OS X . . . . .                          | 3         |
| 2.4      | Licenses . . . . .                          | 3         |
| <b>3</b> | <b>Theory</b>                               | <b>4</b>  |
| 3.1      | Input . . . . .                             | 4         |
| 3.2      | Setup Steps . . . . .                       | 4         |
| 3.3      | Output . . . . .                            | 8         |
| <b>4</b> | <b>Fred Receptor Controls</b>               | <b>10</b> |
| 4.1      | Overview . . . . .                          | 11        |
| 4.2      | Molecule Mode . . . . .                     | 13        |
| 4.3      | Box Mode . . . . .                          | 15        |
| 4.4      | Tweak Mode . . . . .                        | 16        |
| 4.5      | Shape Mode . . . . .                        | 18        |
| 4.6      | Constraint Mode . . . . .                   | 20        |
| 4.7      | Trial Docking Mode . . . . .                | 25        |
| 4.8      | Finish Mode . . . . .                       | 30        |
| <b>A</b> | <b>Release Notes</b>                        | <b>32</b> |
| A.1      | Version 2.2.5 Change Log . . . . .          | 32        |
| A.2      | Version 2.2.3 Change Log . . . . .          | 32        |
| A.3      | Version 2.2.2 Change Log . . . . .          | 32        |
| A.4      | Version 2.2.1 Change Log . . . . .          | 33        |
| <b>B</b> | <b>File formats</b>                         | <b>34</b> |
| B.1      | Valid file extensions for Reading . . . . . | 34        |
| B.2      | Valid File Extensions for Writing . . . . . | 35        |

---

# Version

This document is for version 2.2.5 of FRED RECEPTOR.

## Introduction

*fred\_receptor* is a wizard like graphical utility that prepares an active site for docking with FRED, OpenEye's docking program. *fred\_receptor* was created to make preparing an active site a more intuitive process by allowing the user to visualize the active site and how it is setup, however FRED does not require that the active site be prepared with *fred\_receptor*. Input to *fred\_receptor* is the structure of the target protein, generally from an X-ray crystallography experiment. Output is a receptor file, which is a specialized OEB (OpenEye's molecule format) file, used by FRED.

## 1.1 Prototype Software

The *fred\_receptor* program is prototype software, which means the overall design of this program may be changed in future versions. This is the first time a GUI has been created to assist Fred, or any OpenEye computational program, and as such is somewhat experimental. We expect and hope to get feedback from users regarding the *fred\_receptor* program. Based on that feedback and other considerations future versions of *fred\_receptor* may have a substantially different look and feel as well as somewhat modified functionality. For example the next version of Fred *may* have a completely graphical interface that the *fred\_receptor* programs functionality is merged into.

Prototype software does *not* mean beta software. The *fred\_receptor* is a complete stable utility to assist Fred users in preparing their active site for docking. If any bugs are found to exist users can expect that they will be addressed with periodic bugfix releases, just as any other program OpenEye releases.

# Installation

Currently *fred\_receptor* is supported on the following platforms

- Microsoft Windows 2000
- Microsoft Windows XP
- Mac OS X 10.4
- RedHat Enterprise Linux 3.0
- RedHat Enterprise Linux 3.0 64-bit
- RedHat Enterprise Linux 4.0
- RedHat Enterprise Linux 4.0 64-bit
- RedHat Linux 7.2
- RedHat Linux 8.0
- RedHat Linux 9.0
- Suse Linux 9.1
- SGI Irix 6.5

## 2.1 Microsoft Windows

A simple installation wizard is provided for Microsoft Windows. Double-clicking on the file (e.g. *fred\_receptor-2.2-microsoft-win32-i686-setup.exe*) will start the wizard which will direct the installation process. It is recommended that the installer to be allowed to install in the *C:\OpenEye\fred\_receptor* directory if possible. Once the installation process is complete, there will be a *fred\_receptor* icon on the desktop as well as in the start menu.

## 2.2 UNIX/Linux

Executables for *fred\_receptor* can be downloaded from OpenEye Scientific Software's download page at :

<http://www.eyesopen.com/download/>

Select the tar.gz file appropriate to your architecture (e.g., *fred\_receptor-2.2beta3-redhat-8.0-g++3.4-i586.tar.gz*) and download it. The tar.gz package unpacks into a directory *openeye/* with multiple subdirectories. If another OpenEye application has already been installed you should unpack the FRED tar.gz file in the same place. Once the FRED tar.gz package has been unpacked the following executables will be located in *<directory you unpacked in>/openeye/bin*.

**fred\_receptor** The *fred\_receptor* utility.

## 2.3 Mac OS X

The Mac OS X version comes pre-bundled. All that is required is to drag the *fred\_receptor* icon to the desired location on the computer such as the Applications folder. Double-clicking on the icon will launch *fred\_receptor*.

## 2.4 Licenses

In order to run *fred\_receptor*, a valid license provided by OpenEye must be present on the installed computer. OpenEye license files are typically named *oe\_license.txt* and most OpenEye software products (including *fred\_receptor*) expect to find that license file in the location specified by the *OE\_LICENSE* environment variable or in the directory specified by the *OE\_DIR* environment variable.

If neither of the above environment variables are defined on the computer, *fred\_receptor* will prompt the user at run time to locate a valid license file. If a valid license file is specified, *fred\_receptor* will remember the location of that file and will not prompt again until that license expires or is removed.

The *fred\_receptor* uses the same license as the main *fred* command line program, no additional license is required.

# Theory

## 3.1 Input

The input to the *fred\_receptor* program when creating a receptor is the structure of a protein, typically a PDB file from the Protein Data Bank or in-house crystallography. A list of all supported input file formats is located in appendix *fred\_receptor*. Once loaded the input structure is converted into a receptor in a sequence of interactive steps.

## 3.2 Setup Steps

### 3.2.1 Molecule Selection

In the molecule selection step the input structure is split into individual molecules which can then be marked a protein or ligand molecules (at least one molecule must be marked as a protein before setup can proceed to the next step).

Protein molecules are the molecules FRED will attempt to make complimentary interactions with during the docking process. In addition to macromolecules, crystallographic waters (or other small molecules) present in the input file can also be marked as part of the protein, in which case FRED will treat them like any other part of the protein. Note that FRED will not allow docked ligands to displace any molecule marked as part of the protein, thus crystallographic ligands should not be marked as part of the protein.

Marking a molecule as a ligand will enable FRED to use the CGO and CGT ligand based scoring functions. It will have no effect on any of the other scoring function in FRED. Marking a ligand is not required, however, it will help guide some of the subsequent receptor setup steps in addition to enabling the ligand based scoring functions.

Molecules not marked as protein or ligand can either be deleted or left unmarked. Unmarked molecule are completely ignored by FRED but are saved in the receptor file, and thus will be present if the receptor is saved and reloaded.

### 3.2.2 The Active Site Box

The active site box defines the general area of the protein where ligands are expected to bind. Any molecule that FRED docks must fit within the active site box, although this is only one of several criteria required for docking. FRED uses the active site box to determine the size and location of several grids, and thus a larger box size will increase FRED's memory usage, while a smaller box will reduce it. Box size also indirectly influences docking speed by allowing larger inner and outer contours (see section Defining the Active Site Shape, as well as the theory section of the FRED manual), although the contour volumes are the real controlling factor. Typical box sizes range from 1000 to 8000 cubic angstroms.

If a molecule was marked as a bound ligand in the previous step an initial active site box around that ligand will be defined, which can be manually adjusted if desired. Potential active sites can also be detected using shape based detection routines of which there are two types.

**Atomic** detection uses a Carbon atom probe to determine areas of the protein where docking is likely to occur. Qualitatively the probe favors positions where the carbon probe can make many contacts with protein atoms, without clashing. This type of site based detection is fast, but tends to detect many potential sites.

**Molecular** detection uses multiple molecular probes rather than a single carbon probe to detect likely sites. These probes are docked using a shape based potential, and regions where multiple probes dock are considered favorable. This type of site based detection is slow, it can take several minutes, but detects few sites that are in general of higher quality than atomic probe.

In general Atomic site detection is quick and dirty, while Molecular detection is slowed but more effective. Once site detection is finished the results are displayed in the 3D window and a box can be created or extended around any site simply clicking on it.

### 3.2.3 Tweaking Residues

The 'Tweaking' step allows users to make minor manual modifications of residues near the active site. This step is optional.

Not all residues can be tweakable, those that are are listed below.

**Asparagine** Tweaking flips the position of the OD1 and ND2 atoms.

**Glutamine** Tweaking flips the OE1 and NE2 atoms.

**Histidine** Tweaking cycles the protonation states of the ring nitrogens and the position of the nitrogens in the ring (i.e., ND1 swaps with CD2 and CE1 swaps with CD2). Thus there are a total of 8 states to cycle through (2 ND1 protonation x 2 NE2 protonation x nitrogen position).

**Serine** Tweaking rotates the hydroxyl.

**Threonine** Tweaking rotates the hydroxyl.

**Tyrosine** Tweaking rotates the hydroxyl.

Note that rotating hydroxyls (i.e., tweaking Serine's, Theronine's and Tyrosine's) will only effect the Screenscore and Zapbind scoring functions. The other scoring functions in Fred either ignored hydrogen position all together (Shapegauss, PLP) or calculate optimal positions for hydroxyl hydrogens on the fly (Chemgauss2, Chemgauss3, Chemscore, OEChemscore).

### 3.2.4 Defining the Active Site Shape

The active site shape is described by two contours of a shape potential, referred to as the inner and outer contour. The outer contour is the larger of the two, generally about 1000 to 1500 cubic angstroms, and FRED requires that docked poses fit within the shape of the outer contour. The outer contour volume very strongly affects the time it takes FRED to dock conformers, with larger outer contour increasing the time because there are more acceptable poses positions for FRED to score. The inner contour is smaller, generally 50-100 cubic angstroms (sized below 50 cubic angstroms are not recommended), and FRED requires that the center of at least one heavy atom of any docked pose touches the inner contour. Smaller inner contour volumes tend to increase the docking speed, however the effect is far less pronounced than adjusting the outer contour size.

The shape potential, which the inner and outer contours are created from, is created by docking multiple molecular probes, stored internally by the program, into the active site. The molecular probes are regular molecules, but have been chosen to represent common shapes of druglike molecules, and they are docked using FRED's Shapegauss function and alternate poses are retained. Once the molecules have been docked, the volume of the docked poses are averaged to create the site shape potential (with higher weight being given to poses with higher scores, and poses that appear higher up in the alternate pose list). The number of molecular probes used, and the number of alternate poses included is determined by a quality setting in the GUI as follows

**low** 25 probes, 1000 alternate poses per probe.

**medium** 100 probes, 10000 alternate poses per probe.

**high** 400 probes, 100000 alternate poses per probe.

Changing the quality tends to change the characteristics of the shape potential where there is low shape potential, while having little effect in areas with high shape potential. Specifically at lower quality settings, positions somewhat distant from the protein will have a very low shape potential, while higher quality settings will make those positions more favorable. Both high and low quality settings have zero shape potential very far from the protein, but with higher quality settings the decay is slower. Thus higher quality settings should be used if it is expected that molecules docked to the active site could be highly solvated, while low quality settings should be used for enclosed sites where docked molecule are likely to be relatively unsolvated.

### 3.2.5 Constraints

Constraints are used to restrict the poses FRED will examine while docking a molecule, to those matching user specified chemical or shape functionality. Setting up constraints is optional. There are two general types of constraints

**Custom constraints** are user defined constraints that use spheres and associated SMARTS patterns to specify regions within the active site where certain chemical functionality is required. FRED will reject any poses that do not match this functionality. Each custom constraint is referred to as a constraint feature.

Each constraint feature consists of one or more spheres, and optionally a list of SMARTS patterns. A feature without a SMARTS patterns will be satisfied if any heavy atom of the pose falls within one of the feature's spheres. If the feature has SMARTS pattern(s) only atoms which match the SMARTS pattern(s) can satisfy the constraint.

**Protein constraints** Protein constraints are placed on individual protein atoms and come in three types:

**Hydrogen Bond** constraints tell FRED that a pose must make a hydrogen bond interaction with the specified protein atom to pass the constraint filter. These constraints have more geometric specificity than custom constraints (they are inherently directional), and can be specified as either acceptor or donor constraints (or both). They function by recognizing "lone pair" and "polar hydrogen" positions around acceptor and donor atoms respectively and require that one of the acceptor's "lone pair" positions is within 1.5 Angstrom of the donor's "polar hydrogen" position. Note that the actual position of a donor's polar hydrogen is not used, rather FRED generates its own set of likely polar hydrogen positions.

**Metal** constraints tell FRED that a pose must make a coordinating interaction with the metal the constraint is placed on to pass the constraint filter (metals are treated as being part of the protein by FRED). These constraints have more geometric specificity than custom constraints. Similar to hydrogen bond constraints, metal constraints work by defining "coordinating positions" around a coordinating atom, and requiring that a "coordinating position" be within 1.5 Angstrom of the metal.

**Contact** constraints tell FRED that a pose must make a contact interaction with the atom the constraint is placed upon to pass the constraint. Contact is defined as having a heavy atom within 4 Angstroms (atom center to atom center) of the protein atom the constraint is placed on. These constraints have the same geometric specificity as custom constraints (*i.e.* making a custom constraint with a 4 Angstrom radius centered on the protein atom will perform the same function).

#### Mini Constraint F.A.Q.

*Which constraint type should I use?* In general protein constraints are designed for simplicity and constrain the geometry of hydrogen bond and metal constraints more realistically than custom constraints can. Custom constraints on the other hand are more flexible in the sense that they allow users to specify their own chemistry required to satisfy the constraint, *via* SMARTS patterns.

*Is there a maximum number of constraints I can use?* There is no direct limit on the number of constraints that the user can specify, however there is a modest increase in memory requirements for each additional constraint specified. Fred is very efficient at restricting the search space of possible poses based on user defined constraints. Accordingly adding constraints will generally decrease run time.

*Why do some of the poses FRED is generating violate my constraints?* The constraint spheres are mapped onto a grid during the docking process, and the resulting interpolation error can allow atoms slightly outside a sphere (approximately 1/2 the translational stepsize or 0.5Å by default), to satisfy a constraint.

### 3.2.6 Trial Docking

Once the receptor has been setup it can be testing with a trial docking of one molecule. The molecule to be docked can be either

**A bound ligand** This option docks the bound ligand specified in the molecule selection step (if a bound ligand wasn't specified obviously it can't be docked). Note that conformers are *NOT* generated for the bound ligand, rather the crystal structure pose itself is docked. Thus docking the bound ligand is an extremely simple test.

**Loaded from a file** A molecule it loaded from a file and docked. Conformers are not generated for the molecule, although if the molecule file contains conformers for the molecule they will be used. If the file contains more than one molecule, only the first molecule in the file will be docked.

## 3.3 Output

The primary output of the *make\_receptor* program is a specialized type of OEB file (OEB is OpenEye's molecule format) referred to as a receptor file. Thus all receptor files always have either the *.oeb* or the *.oeb.gz* extension.

A receptor file always contains the following:

1. The structure of the target protein.
2. The location of the receptor site on the protein.
3. A shape potential grid that describes the shape complementarity of the active site.

Additionally a receptor file may contain:

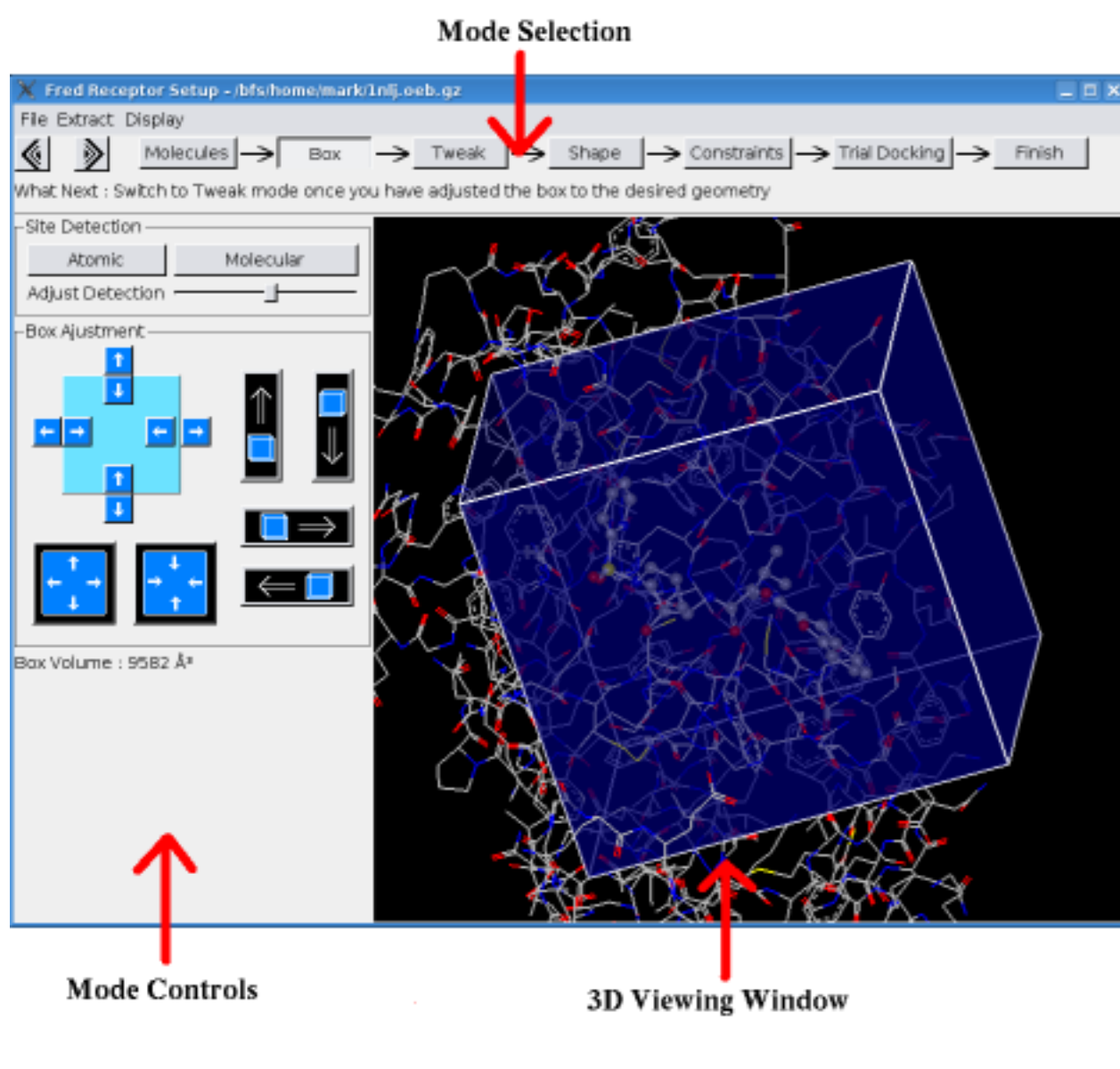
1. One or two isocontour levels of the shape potential grid which create shapes complementary to the active site.
2. The structure of a known bound ligand.
3. Docking constraints.

### 3.3.1 Extracting

The box, protein, and/or bound ligand molecules can be saved individually in any common molecule format (e.g., mol2, sdf, pdb, etc) using the extract menu. This functionality is for the convenience of user who wish to view these parts of the receptor in Non-OpenEye programs and is not required by FRED (all the data is already stored in the receptor file).

# Fred Receptor Controls

## 4.1 Overview



In addition to the menu bar *fred\_receptor* has three major areas as follows

**Mode Selection** This area contains buttons that controls what step (or mode) the receptor setup is on, as well as a tool tip indicating the current task. The interface in the mode controls region and what is displayed in the 3D window will change depending upon the mode selected.

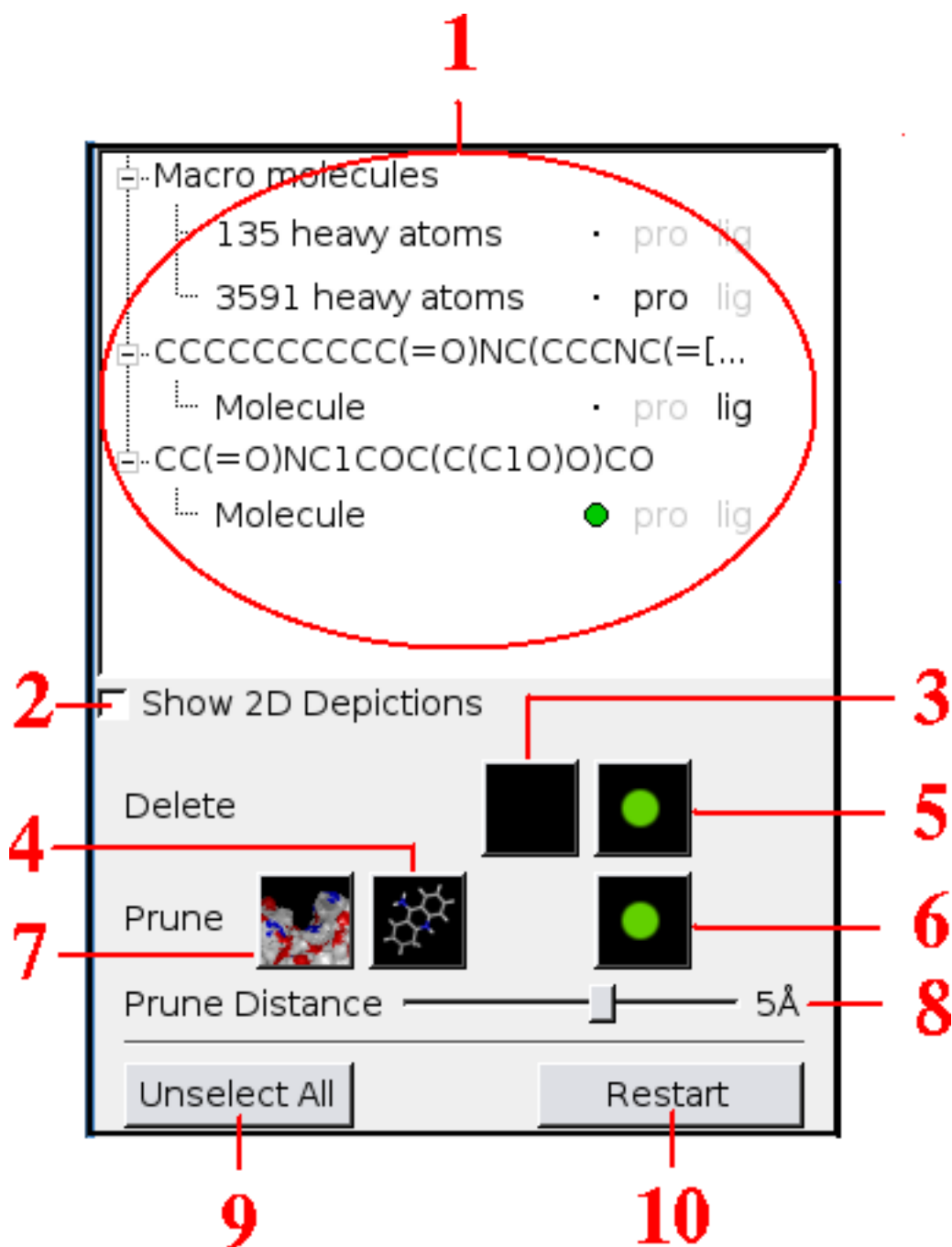
There is a general order to the steps or modes that when setting up a receptor as indicated by the arrows and placement of buttons in the mode selection region that users are expected to progress

along in a wizard like fashion. In most cases at each step/mode certain tasks must be accomplished before advancing to the next step/mode. For example after a PDB file is initially loaded the one or more molecules must be mark as a protein before progressing to the box step. In these cases steps/modes the steps further along will be selectable until the required task is complete (check the tool tip if you are uncertain what to do).

**Mode Controls** This area contains the primary controls for setting for the current mode (or step) of the receptor step. The interface in this area will change depending upon the mode. These controls are explain in more detail below.

**3D viewing window** This area contains a 3D display of the receptor relevant to the current setup step (or mode) fred\_receptor is in. What exactly is displayed varies depending upon the mode.

## 4.2 Molecule Mode

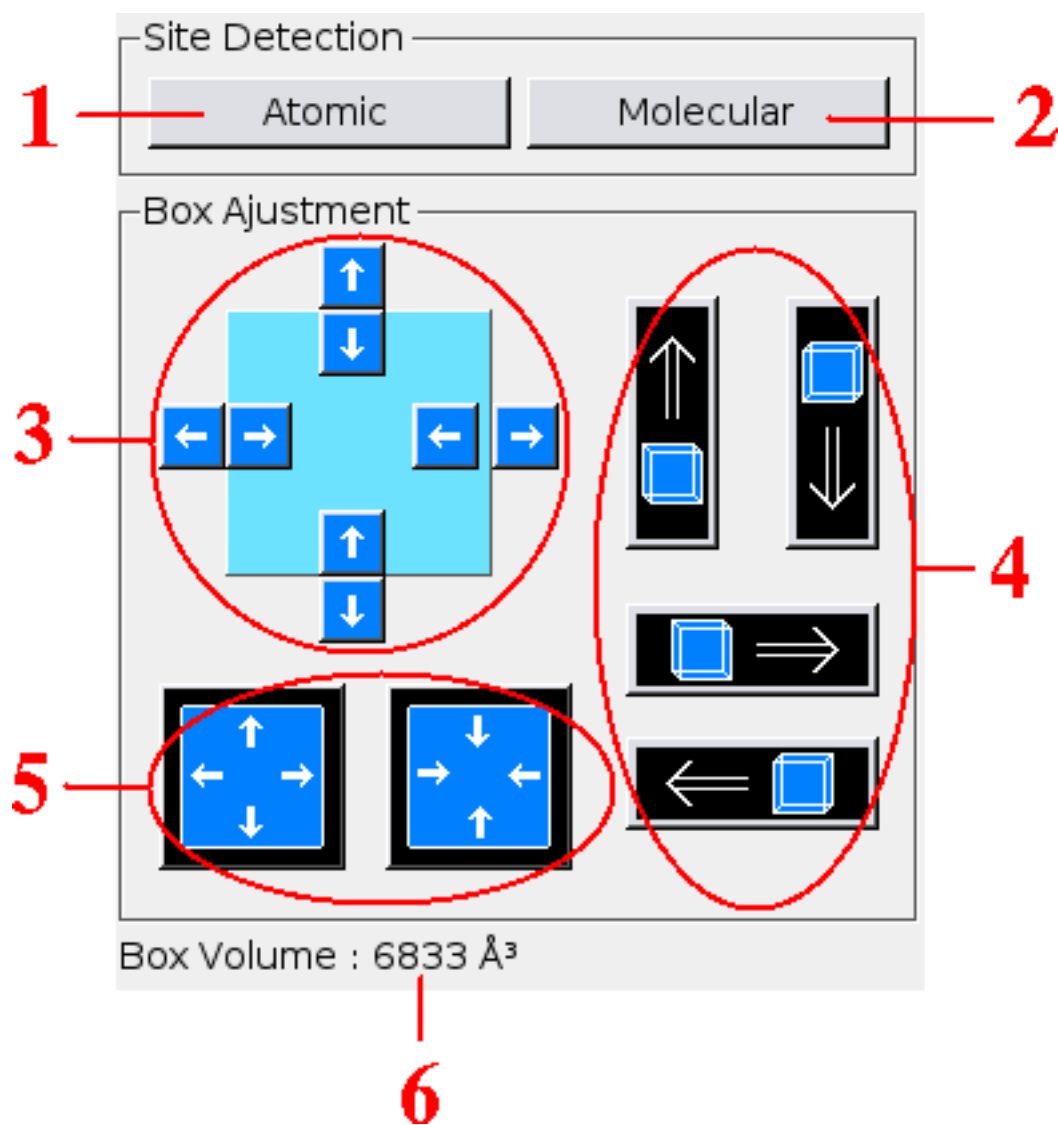


1. This area list all the different molecules present. Each molecule can be marked or unmarked as protein or ligand by clicking on the "pro" or "lig" text after the molecule name respectively. Molecule cannot be marked as both protein and ligand. Additionally individual molecules can

be selected or unselected by clicking on the molecule name. Selected molecules are displayed as CPK in the 3D window and a green dot is placed next to their name. Note that double clicking a molecule will select or deselect all molecules of that type.

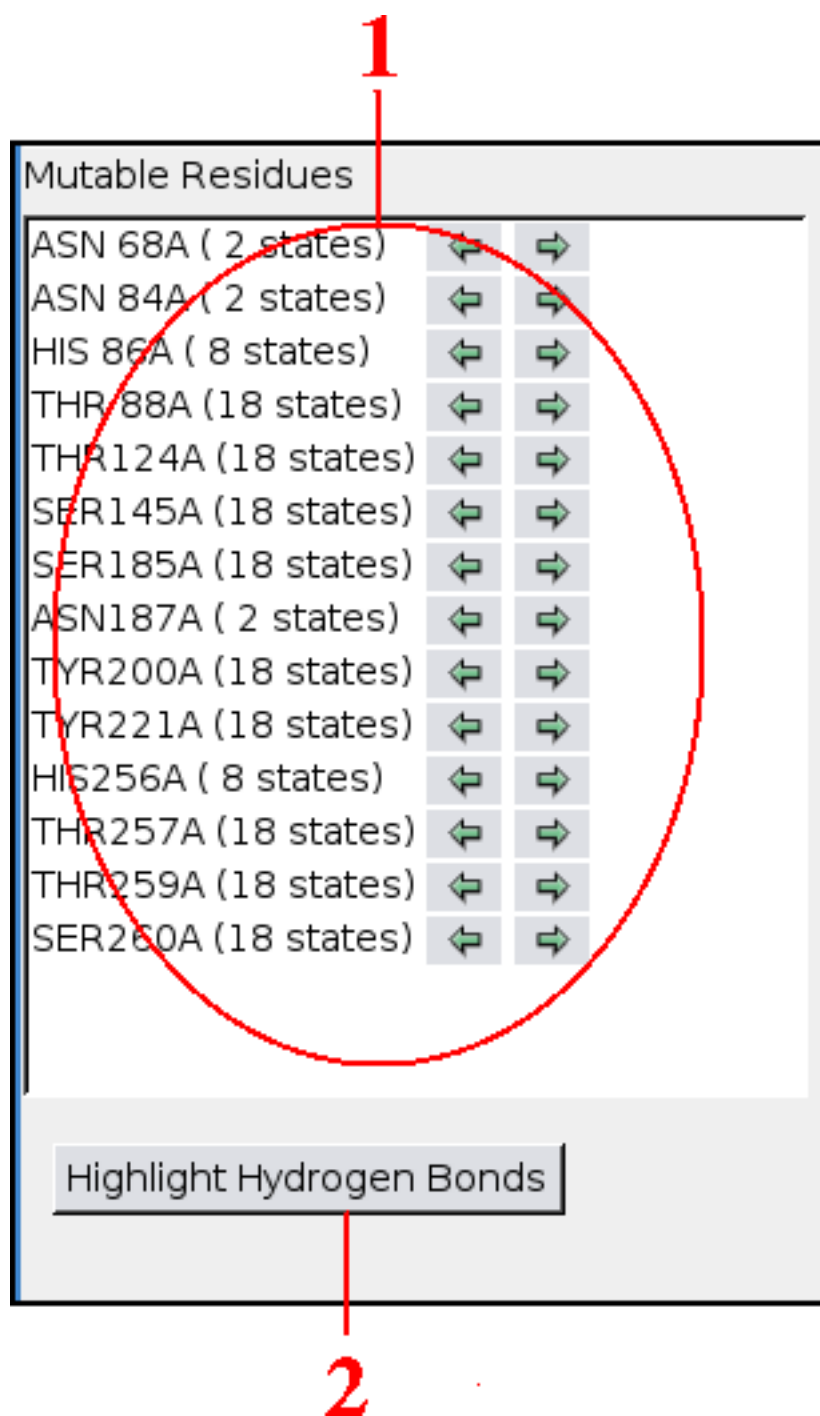
2. The show depictions checkbox controls whether a 2d image of each molecule type is displayed in area 1.
3. Delete all unselected molecules (i.e., ones that don't have green dots next to their name).
4. Delete all molecules that are more than the prune distance (see 8) from the bound ligand.
5. Delete all selected molecules (i.e., ones that have green dots next to their name).
6. Delete all molecules that are more than the prune distance (see 8) from any selected molecule.
7. Delete all molecules that are more than the prune distance (see 8) from any molecule marked as protein.
8. The pruning distance is controlled by this slider. One of the prune buttons is pressed all molecules more than the prune distance will be deleted. The prune distance between any two molecules is the smallest atom center to atom center distance of any atom on each molecule.
9. Deselect all currently selected molecules.
10. Restore all deleted molecules and unmark all molecules.

## 4.3 Box Mode



1. Detect active sites using an atomic site detection method.
2. Detect active sites using a molecular site detection method.
3. Expand or contract the top, bottom, left or right of the box.
4. Translate the box up, down, left or right.
5. Expand or contract the box.
6. Display of current box volume.

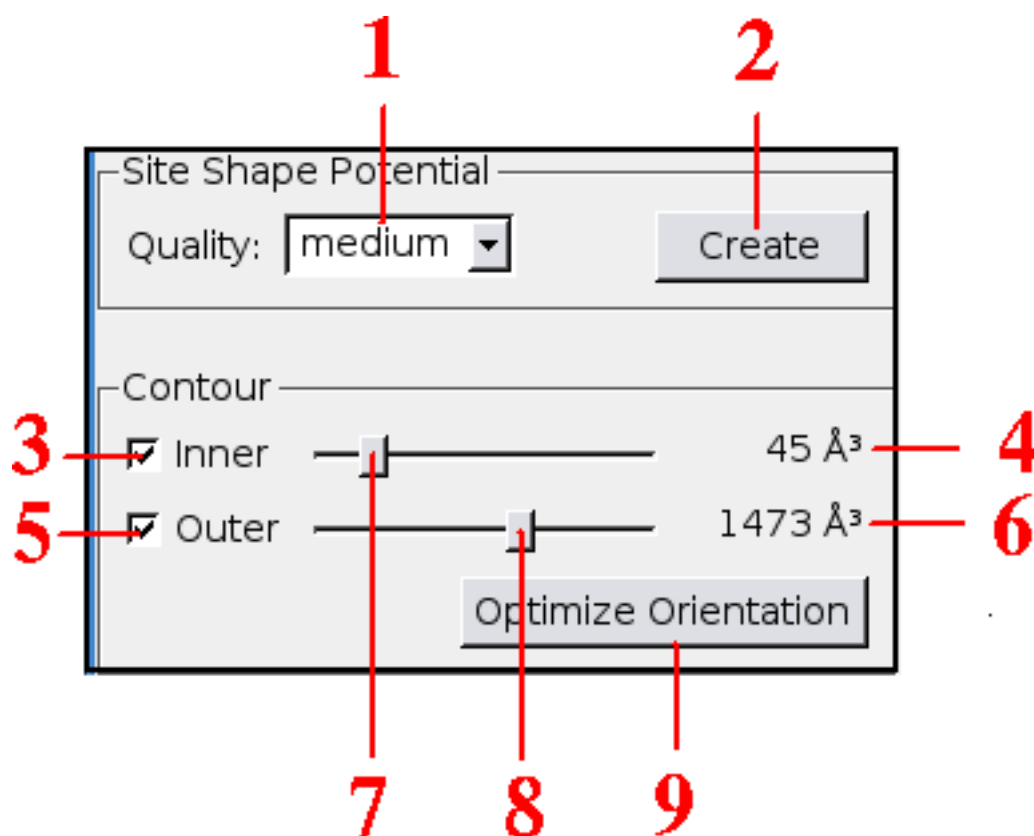
## 4.4 Tweak Mode



1. List of tweakable residues. Residues can be modified by clicking on either arrow key.

2. Highlight areas of likely hydrogen bonding using the Chemgauss3 scoring function.

## 4.5 Shape Mode

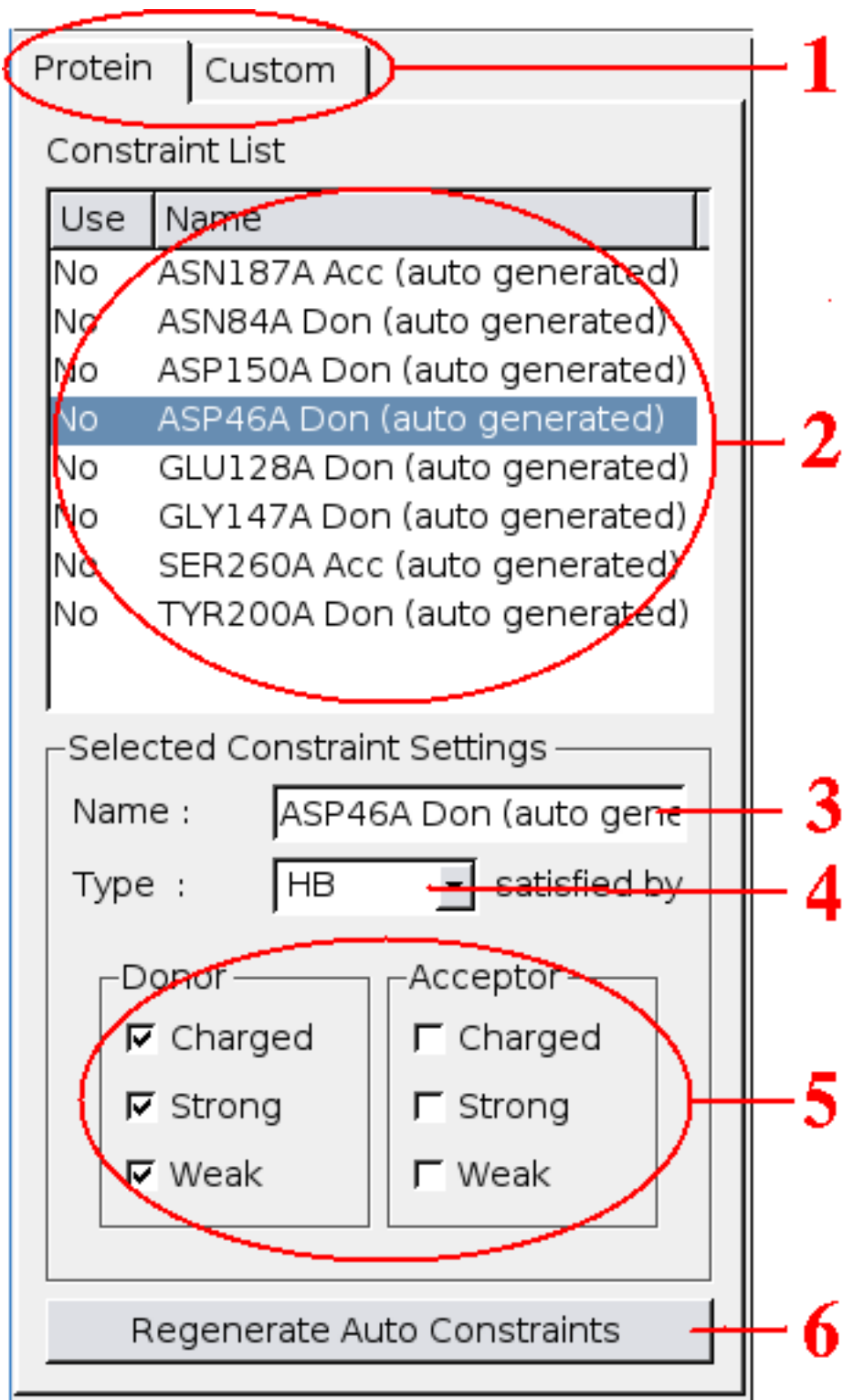


1. Sets the quality of the shape potential that will be generate when the create button is pressed, or optimize orientation is selected.
2. Create the site shape potential (or re-generate it if one already exists).
3. Check box for enabling the inner contour.
4. Volume of the inner contour.
5. Check box for enabling the outer contour. (Disabling the outer contour is not recommended unless you really know what you're doing).
6. Volume of the outer contour.
7. Slider for adjusting the inner contour volume (50 to 100 cubic angstroms is the recommended range).
8. Slide for adjusting the outer contour volume (500 to 2000 cubic angstroms is the recommended range).

9. This button alters rotates the coordinate system in such a way to minimize the amount of memory Fred will use when docking the active site. All molecules docked to the receptor will use this new coordinate system. Optimizing the orientation of the receptor is optional, and is recommended only for those users highly concerned with Fred's speed and memory usage (quality of results will not be affected one way or the other).

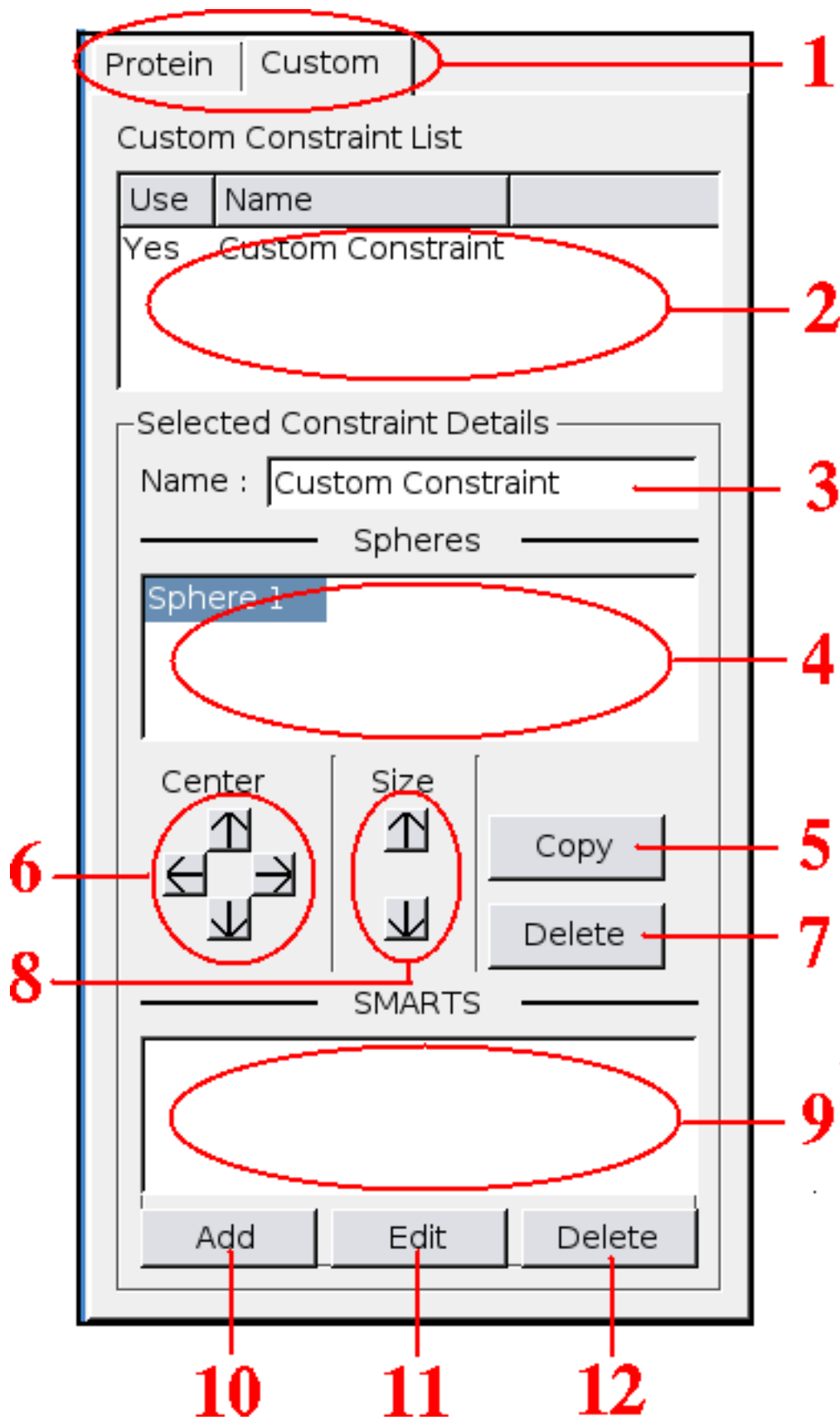
## 4.6 Constraint Mode

## 4.6.1 Standard Constraints



1. Tab for switching between protein and custom constraints.
2. List of existing protein constraints. Individual constraints can be enabled or disabled by clicking No or Yes in the "Use" column.
3. Name of the currently selected constraint.
4. Type of the currently selected constraint.
5. Type of atoms that will satisfy the constraint (the available types will depend on the type of constraint).
6. Button to regenerate auto constraints. This button, and auto constraints in general, are only available if a known bound ligand was specified in molecule mode.

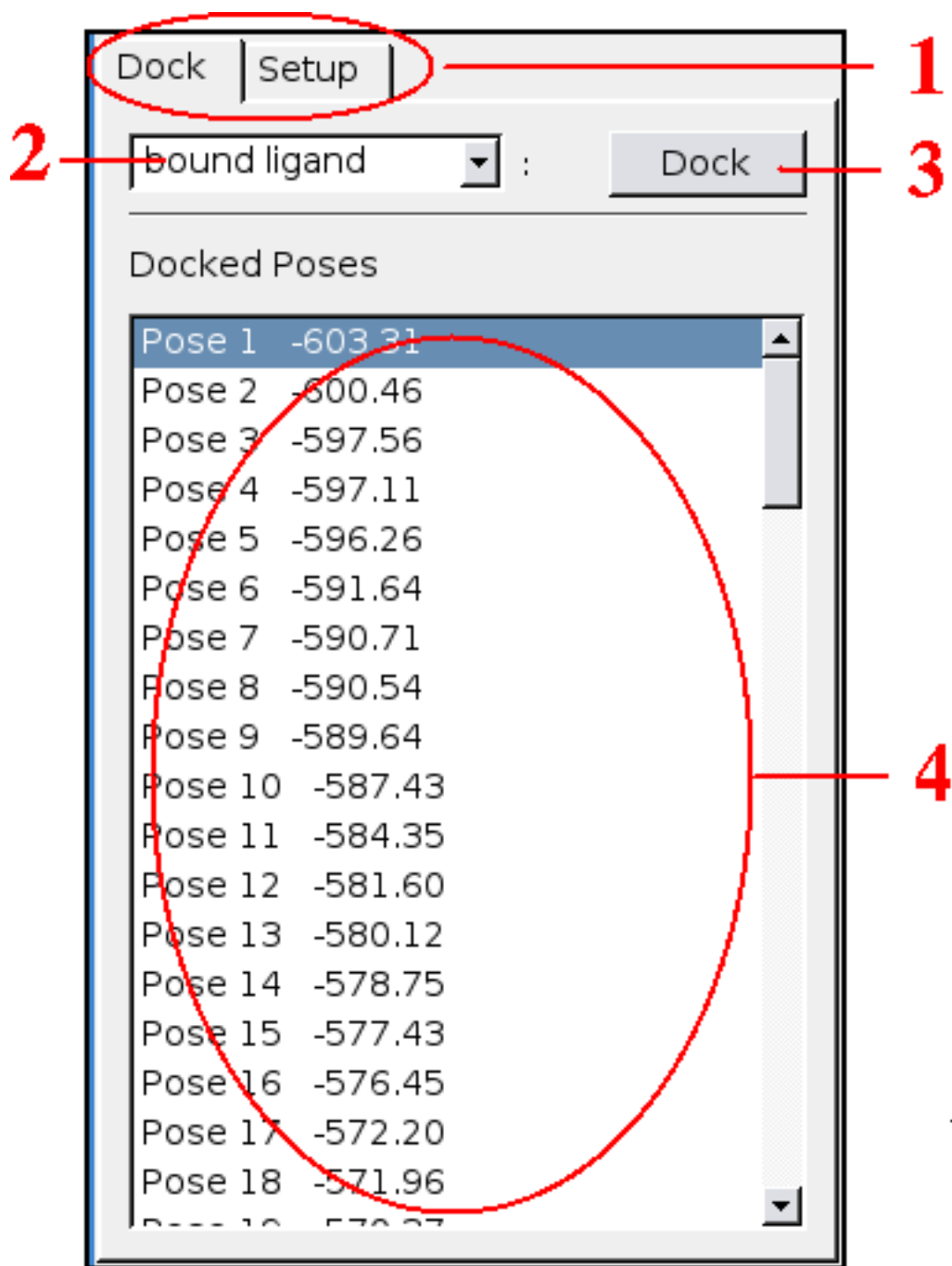
#### 4.6.2 Custom Constraints



1. Tab for switching between protein and custom constraints.
2. List of custom constraints.
3. Name of currently selected custom constraint.
4. List of sphere associated with currently selected custom constraint.
5. Makes of copy of the currently selected sphere.
6. Moves the currently selected sphere up, down, left or right.
7. Deletes the currently selected sphere (a custom constraint must always have at least one sphere).
8. Increases or decreases the size of the currently selected sphere.
9. List of smarts patterns associated with the currently selected custom constraint.
10. Add a new smarts pattern to the currently selected custom constraint.
11. Edit the currently selected smarts pattern.
12. Delete the currently selected smarts pattern.

## 4.7 Trial Docking Mode

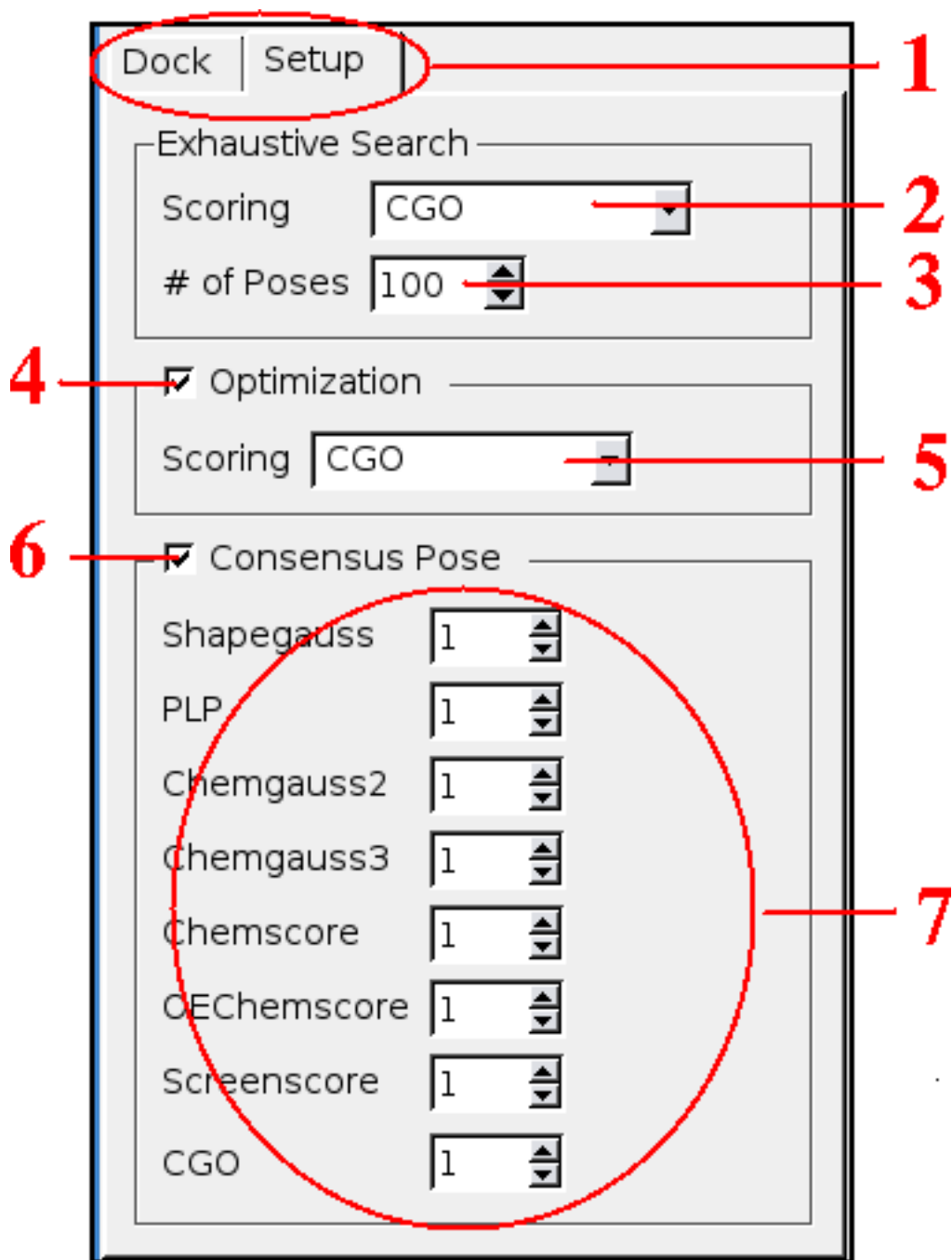
## 4.7.1 Docking



1. Tab to switch between docking and docking setup controls.
2. Specifies the ligand to perform a trial docking with.

3. Dock the trial ligand.
4. Poses and scores of the docked trial ligand. Clicking on a pose will display it in the 3D window.

## 4.7.2 Settings



1. Tab to switch between docking and docking setup controls.
2. Specify the scoring function to use during the exhaustive search. This setting is equivalent to specifying the flag `-exhaustive_scoring` on Fred's command line.

3. Number of poses the exhaustive search will generate. This setting is equivalent to specifying *-num\_poses* on the command line.
4. Check box to enable optimization of the poses returned by the exhaustive search. Unchecking this box is equivalent to setting *-optimization none* on Fred's command line.
5. Specifies the scoring function to optimize poses from the exhaustive search with. The setting is equivalent to specifying a *-optimization* on Fred's command line.
6. Check box to enable consensus structure (in which poses from the exhaustive search are ranked based upon the consensus of several scoring functions).
7. Weights of individual scoring functions for consensus structure.

## 4.8 Finish Mode

Molecule Titles

Protein  1

Bound Ligand  2

Contours 3

Inner 4  Outer 5

62 Å<sup>3</sup> 1892 Å<sup>3</sup>

Protein Constraints 6

| Enabled | Name                        |
|---------|-----------------------------|
| No      | CYS25A Acc (auto generated) |
| No      | GLY66A Acc (auto generated) |

7

Custom Constraints

| Enabled | Name |
|---------|------|
|---------|------|

8

9 Save 10 Exit

1. Name of the target protein
2. Name of the bound ligand, if one is present.

3. Slider to adjust the inner contour volume.
4. Checkbox to enable or disable the inner contour.
5. Slider to adjust the outer contour volume.
6. Checkbox to enable or disable the outer contour.
7. List of protein constraints.
8. List of custom constraints.
9. Save the receptor.
10. Exit fred\_receptor.

Note that items 3 - 8 are functionally equivalent to some of the controls in either shape or constraint mode. They are provided here as a convenience.

---

# Release Notes

## A.1 Version 2.2.5 Change Log

1. FRED RECEPTOR no longer modifies existing residue information, it only updates residue information if it doesn't exist.
2. FRED RECEPTOR licenser has been updated to work with licenses that expire past the year 2009.
3. The FRED RECEPTOR licenser supports oe\_license.txt located in the users home directory.
4. On Microsoft Windows platforms, the installer adds the ability to open command prompts that setup the user environment to run specific versions of FRED or the latest version of FRED.
5. FRED and FRED RECEPTOR are now versioned and shipped together.

## A.2 Version 2.2.3 Change Log

1. Fixed a bug that caused trial docked molecules to always fail to dock with NoConstraintMatch code when two or more custom constraints were enabled.

## A.3 Version 2.2.2 Change Log

1. Corrected a deficiency in Chemgauss3 metal term and metal constraints. The metal chelator interaction function were picking up on some but not all of the allowable geometries for metal-chelator interactions.

## A.4 Version 2.2.1 Change Log

1. Fixed a bug that could cause a potential crash when a protein with a metal is loaded and constraint mode is selected.
2. Program version is now displayed in the title bar until a molecule or receptor is loaded.

# File formats

The *fred\_receptor* program supports a wide range of input file formats for molecules. *fred\_receptor* determines the file format by parsing the extension of input files. The following is a list of recognized extensions:

## B.1 Valid file extensions for Reading

**dat** Macromodel

**ent** PDB

**mdl** MDL Mol

**mmd** Macromodel

**mmod** Macromodel

**mol2h** MOL2 with H

**mol2** Tripos MOL2

**mol** MDL Mol

**mopac** MOPAC

**oeb** OEBinary v2

**pac** MOPAC

**pdb** PDB

**rxn** MDL Mol

**sd** MDL SDF

**sdf** MDL SDF

**syb** Tripos MOL2

**xyz** XYZ

## B.2 Valid File Extensions for Writing

**oeb** OEBinary v2

*Note* : Reading and writing of gzipped files is also supported (e.g., ent.gz, mol2.gz, oeb.gz).