

OpenEye
SCIENTIFIC

OEDocking TK – Java

Release 2026.1.0

Cadence Design Systems, Inc.

August 06, 2026

CONTENTS

1 Introduction **1**
- 1.1 Welcome to the Docking Toolkit documentation . . . 1

2 Theory **3**
- 2.1 Receptors . . . 3
  - 2.1.1 Creating a Receptor . . . 3
  - 2.1.2 Contents of a Receptor . . . 3
- 2.2 Scoring . . . 5
  - 2.2.1 Scoring Function Implementations . . . 5
- 2.3 Docking . . . 8
  - 2.3.1 Scoring Functions and Search Resolution . . . 8
  - 2.3.2 Initialization . . . 10
  - 2.3.3 Docking Molecules . . . 10
  - 2.3.4 Scoring Molecules . . . 10
  - 2.3.5 Docking Algorithm . . . 10
- 2.4 ShapeFit . . . 11
  - 2.4.1 On Clashes . . . 13
- 2.5 POSIT Theory . . . 14
  - 2.5.1 Handling of isomerisms and chirality . . . 14
  - 2.5.2 Estimated Pose Probability . . . 14
  - 2.5.3 Additional MCS Constraints . . . 16
  - 2.5.4 On Clashes . . . 16
  - 2.5.5 Using Multiple Receptors . . . 16
  - 2.5.6 Receptor Flexibility . . . 16
  - 2.5.7 TanimotoCombo . . . 17
- 2.6 Posit How to . . . 17
  - 2.6.1 Ignore Nitrogen stereo . . . 17
  - 2.6.2 Prepare molecules for Posit . . . 18
  - 2.6.3 Use multiple receptors . . . 18
  - 2.6.4 Obtain clash-free pose . . . 18

3 OEDocking Examples **19**
- 3.1 Docking and Scoring Examples . . . 19
  - 3.1.1 Docking Molecules . . . 19
  - 3.1.2 Rescoring Docked Molecules . . . 21
- 3.2 OEShapeFit Examples . . . 23
  - 3.2.1 Flexible Overlay Optimization with OEShapeFit API . . . 23
- 3.3 **POSIT** Examples . . . 23
  - 3.3.1 Generating Poses with **POSIT** . . . 23
  - 3.3.2 Generating Multiple Poses with POSIT . . . 25

3.4 Receptor Examples . . . 28
3.4.1 Making a Receptor . . . 28
3.4.2 Receptor Contour Volume . . . 29
3.4.3 Toggle Receptor Inner Contour . . . 30
3.4.4 Set Receptor Contour Volume . . . 32

4 OEModels Examples **35**
4.1 Building ROCS Query Model . . . 35

5 API **37**
5.1 OEDocking API . . . 37
5.1.1 OEDocking Classes . . . 37
5.1.2 OEDocking Constants . . . 61
5.1.3 OEDocking Functions . . . 70
5.2 Preliminary OEDocking API . . . 78
5.2.1 Preliminary OEDocking Classes . . . 78
5.3 Deprecated OEDocking API . . . 82
5.3.1 OEDocking Classes (Deprecated) . . . 82
5.3.2 OEDocking Functions (Deprecated) . . . 90
5.4 OEModels API . . . 104
5.4.1 Preliminary OEModels Classes . . . 104

6 Release History **109**
6.1 OEDocking TK 4.3.5 . . . 109
6.1.1 Minor Bug Fixes . . . 109
6.2 OEDocking TK 4.3.4 . . . 109
6.2.1 Minor Bug Fixes . . . 109
6.3 OEDocking TK 4.3.3 . . . 109
6.4 OEDocking TK 4.3.2 . . . 109
6.4.1 New features . . . 109
6.4.2 Major bug fixes . . . 110
6.4.3 Minor bug fixes . . . 110
6.5 OEDocking TK 4.3.1.3 . . . 110
6.6 OEDocking TK 4.3.1.0 . . . 110
6.7 OEDocking TK 4.3.0 . . . 110
6.7.1 New features . . . 110
6.7.2 Major bug fixes . . . 110
6.7.3 Minor bug fixes . . . 111
6.8 OEDocking TK 4.2.1 . . . 111
6.8.1 New features . . . 111
6.8.2 Major bug fixes . . . 111
6.8.3 Minor bug fixes . . . 111
6.8.4 Documentation changes . . . 111
6.9 OEDocking TK 4.2.0 . . . 111
6.9.1 New features . . . 111
6.9.2 Major bug fixes . . . 112
6.9.3 Documentation changes . . . 112
6.10 OEDocking TK 4.1.2 . . . 112
6.11 OEDocking TK 4.1.1 . . . 112
6.11.1 New features . . . 112
6.11.2 Minor bug fixes . . . 112
6.11.3 Documentation changes . . . 112
6.12 OEDocking TK 4.1.0 . . . 112
6.12.1 New features . . . 112

6.12.2 Minor bug fixes . . . 113
6.13 OEDocking TK 4.0.0.2 . . . 113
6.14 OEDocking TK 4.0.0 . . . 113
6.14.1 New features . . . 113
6.14.2 Documentation changes . . . 115
6.15 OEDocking TK 3.5.0.5 . . . 115
6.16 OEDocking TK 3.5.0 . . . 115
6.16.1 New features . . . 115
6.16.2 Major bug fixes . . . 115
6.16.3 Minor bug fixes . . . 115
6.16.4 Documentation changes . . . 116
6.17 OEDocking TK 1.4.1 . . . 116
6.17.1 New features . . . 116
6.17.2 Major bug fixes . . . 117
6.17.3 Minor bug fixes . . . 117
6.17.4 Documentation changes . . . 117
6.18 OEDocking TK 1.4.0 . . . 117
6.18.1 Major bug fixes . . . 117
6.18.2 Minor bug fixes . . . 117
6.19 OEDocking TK 1.3.7 . . . 117
6.20 OEDocking TK 1.3.6 . . . 117
6.20.1 New features . . . 117
6.21 OEDocking TK 1.3.5 . . . 117
6.22 OEDocking TK 1.3.4 . . . 118
6.23 OEDocking TK 1.3.3 . . . 118
6.24 OEDocking TK 1.3.2 . . . 118
6.25 OEDocking TK 1.3.1 . . . 118
6.25.1 New features . . . 118
6.25.2 C++-specific changes . . . 118
6.26 OEDocking TK 1.3.0 . . . 118
6.26.1 Minor bug fixes . . . 118
6.26.2 Python-specific changes . . . 118
6.27 OEDocking TK 1.2.7 . . . 118
6.27.1 Minor bug fixes . . . 118
6.28 OEDocking TK 1.2.6 . . . 118
6.28.1 New features . . . 118
6.28.2 Minor bug fixes . . . 119
6.29 OEDocking TK 1.2.5 . . . 119
6.29.1 New features . . . 119
6.29.2 Minor bug fixes . . . 119
6.30 OEDocking TK 1.2.4 . . . 119
6.31 OEDocking TK 1.2.3 . . . 119
6.31.1 New features . . . 119
6.31.2 Major bug fixes . . . 120
6.31.3 Minor bug fixes . . . 120
6.32 OEDocking TK 1.2.2 . . . 120
6.32.1 New features . . . 120
6.32.2 Major bug fixes . . . 120
6.32.3 Minor bug fixes . . . 120
6.32.4 Documentation Changes . . . 120
6.33 OEDocking TK 1.2.1 . . . 121
6.33.1 New Features . . . 121
6.33.2 Major bug fixes . . . 121
6.33.3 Minor bug fixes . . . 121

6.34 OEDocking TK 1.2.0 . . . 121
6.34.1 New features . . . 121
6.35 OEDocking TK 1.1.4 . . . 121
6.35.1 New features . . . 121
6.35.2 Minor bug fixes . . . 121
6.36 OEDocking TK 1.1.3 . . . 121
6.36.1 Bug fixes . . . 121
6.37 OEDocking TK 1.1.2 . . . 121
6.37.1 Bug fixes . . . 121
6.37.2 Improvements . . . 122
6.38 OEDocking TK 1.1.1 . . . 122
6.38.1 Bug fixes . . . 122
6.39 OEDocking TK 1.1.0 . . . 122
6.39.1 New Features . . . 122
6.39.2 Bug fixes . . . 122
6.40 OEDocking TK 1.0.0 . . . 122

7 Copyright and Trademarks **125**

8 Sample Code **127**

9 Citation **129**
9.1 Orion® . . . 129
9.2 Orion Floes . . . 129
9.3 Toolkits and Applications . . . 129
9.4 Publications for Bibliographies . . . 131
9.4.1 Orion . . . 131
9.4.2 AFITT and FLYNN Applications . . . 131
9.4.3 OEDocking Application and Toolkit . . . 131
9.4.4 OMEGA Application and Toolkit . . . 131
9.4.5 ROCS Application . . . 132
9.5 OpenEye MMDS Web Service . . . 132

10 Technology Licensing **133**
10.1 GCC . . . 205
10.1.1 GCC RUNTIME LIBRARY EXCEPTION . . . 206
10.1.2 GNU GENERAL PUBLIC LICENSE . . . 207

Index **221**

CHAPTER

ONE

INTRODUCTION

1.1 Welcome to the Docking Toolkit documentation

The *Docking Toolkit* library provides a facility for docking and scoring molecules in the context of a protein active site.

The following basic functionalities are available:

- Docking (see *Docking* chapter)
- Pose Prediction (see *POSIT Theory* chapter)
- Scoring (see *Scoring* chapter)

The aim of this manual is to familiarize the user with the *Docking Toolkit* functionalities, however, it does not provide explanations of basic *OEChem* classes and functions. Therefore reading the *OEChem* manual beforehand is highly recommended.

See also

- Java Quick Start Manual for installation instructions.
- Importing Java Toolkits section

CHAPTER

TWO

THEORY

2.1 Receptors

In receptor objects contain information about the location and characteristics of the binding pocket, and required for docking (see *Docking*), posit (see POSIT), and scoring (see *Scoring*).

A receptor is an OEReceptor object that resides within a OEDesignUnit. The essential component of a valid receptor is the negative image that describes the shape of the active site.

2.1.1 Creating a Receptor

Within the Docking toolkit the `OEMakeReceptor` function is used to create a receptor from the structure of a target protein containing in a OEDesignUnit.

The protein structure should only include molecules the ligand is expected to interact with. In general crystallographic waters, other solvents and the bound ligand should be not be included as part of the target protein structure passed to the `OEMakeReceptor` function, although in certain cases, the user may wish to retain certain key molecules as part of the target protein structure (*e.g.* a crystallographic water).

2.1.2 Contents of a Receptor

Negative Image

The negative image describes the shape of the active site. It is stored as a potential grid surrounding the active site. Potential values are always greater than or equal to zero. The negative image has high potentials where ligand atoms make many contacts with atoms of the active site without clashing and in positions some ligand atoms are likely to occupy when other atoms of the ligand make good contacts with the receptor (*e.g.* bridging positions ligand atoms will likely occupy when a ligand is stretched between two pockets).

During docking two shapes are created by contouring the negative image potential grid. The two shapes control the docking process as follows:

Outer Contour Shape

> During docking any pose examined by the exhaustive search that does not fit within this shape will be rejected. A pose is considered to fit if the center of every heavy atom is within this shape. The volume of this shape is typically between 500 and 2000 cubic Angstroms.

Inner Contour Shape

> During docking any pose examined by the exhaustive search that does not touch this shape is rejected. A pose is considered to touch if the center of at least one heavy atom falls within this shape. The volume of this shape is typically 50 to 100 cubic angstroms.

While neither the *outer contour* nor the *inner contour* shape are required, it is recommended that the *outer contour* always be used (docking speed will be dramatically slower without the *outer contour*). Using the *inner contour* improves

docking speed slightly at the expense of sampling. However, if the *inner contour* size is nearly as large as the *outer contour*, there will be little to no difference in compute time.

The negative image of the receptor is set up when the receptor is created with the `OEMakeReceptor` function (see the section on creating a receptor). The size of the *outer contour* and *inner contour* shapes are controllable by setting the contour level used to create the shape from the negative image. The `OEMakeReceptor` function will automatically set a reasonable contour level for both the *inner contour* and *outer contour*, however the *inner contour* will be disabled by setting the value to be negative (see below).

The negative image grid has only positive values, thus only contour levels that are positive create a shape. When either the *outer contour* level or *inner contour* level are negative that contour shape will be disabled (*i.e.* ignored during the docking process).

The volume of the *outer contour* shape has a significant effect on docking speed, while the volume of the *inner contour* shape has a modest effect on docking speed. In both cases the smaller volumes increase docking speed by reducing the number of poses that are scored. In general an outer and inner contour volume of between 500-2000 and 50-100 cubic Angstroms respectively is recommended.

Constraints

Constraints are key interactions ligands are required to make when docking into the active site. They are optional and user defined.

Constraints do not affect how a given pose scores, however they do affect how the docking algorithm chooses poses to score during the docking process. Any pose that does not match the docking constraints will be rejected and replaced by the next best scoring pose. If no poses of a ligand match the docking constraints the ligand will not be docked. If multiple constraints are specified every constraint must be satisfied or the pose will be rejected.

> **Note**
>
> The docking process has a resolution of approximately 1 Ångström, and thus the constraints have a similar resolution. Therefore it is possible that poses docked with a constraint may have small violations of the constraint distance, up to approximately 1 Ångström.

Receptors support two general classes of constraints; protein constraints and custom constraints. There may be any number of either class of constraint.

> **Note**
>
> Each individual custom or protein constraint has an enabled flag. A disabled constraint is ignored during the docking process.

Protein Constraints

A protein constraint specifies an interaction that must be made with a **heavy atom** of the protein structure (*i.e.* protein constraints cannot be placed on hydrogen atoms). There are five types of protein constraints.

Contact

A contact constraint is satisfied when any ligand heavy atom is within 4 angstroms of the protein heavy atom.

Lipophilic

A lipophilic constraint is satisfied when any non-polar heavy atom on the ligand is within 4 angstroms of the protein heavy atom.

Donor

A donor constraint is satisfied when a *donor on the ligand* makes a hydrogen bond interaction with the protein heavy atom.

Acceptor

An acceptor constraint is satisfied when an *acceptor on the ligand* makes a hydrogen bond interaction with the protein heavy atom. Acceptor constraints must be placed on the protein heavy atom the donor hydrogen is interacting with.

Chelator

A chelator constraint is satisfied when a *chelator on the ligand* makes a metal-chelator interaction with the protein heavy atom.

Only one protein constraint is allowed per protein heavy atom. If a protein constraint is set on a protein atom that already has a protein constraint the original protein constraint will be discarded and replaced by the new constraint.

Custom Constraints

A custom constraint consists of one or more spheres within the receptor active site, and optionally a list of SMARTS patterns. A custom constraint is satisfied when a matching atom on the ligand falls within any of the spheres associated with the custom constraint. If no SMARTS patterns are specified any heavy atom on the ligand can satisfy the constraint, otherwise only atoms that match one of the smarts patterns can satisfy the constraint.

2.2 Scoring

Scoring functions in the *Docking Toolkit* measure the fitness of ligands posed within the active site of a target protein and assign them a numerical score. Poses with better scores are more likely to be correctly docked compared to other poses of the same ligand. The score of a ligand is the best score of any pose of that ligand, and ligands with better scores are more likely to be active against the target protein compared to other ligands docked.

The following scoring functions are implemented in *Docking TK*

1. *Shapegauss* (`OEScoreType.Shapegauss`)
2. *PLP* (`OEScoreType.PLP`)
3. *Chemgauss3* (`OEScoreType.Chemgauss3`)
4. *Chemgauss4* (`OEScoreType.Chemgauss4`)
5. *Chemscore* (`OEScoreType.Chemscore`)

In the *Docking Toolkit* scoring, optimization and score annotation with any of these scoring functions is done using the *OEScore* class.

Rescoring Docked Molecules is an example program that uses the *OEScore* class to score, optimize and annotate poses.

2.2.1 Scoring Function Implementations

Shapegauss

Shapegauss [McGann-2003] is a shape based scoring function that favors poses that complement the active site well, regardless of any chemical interactions (*e.g.* hydrogen bonds). The Shapegauss scoring function represents the shape of each atom as a smooth Gaussian function.

The Shapegauss score is calculated by summing a pairwise potential between all protein atoms and all ligand heavy atoms. This potential is most favorable when the two atoms touch but do not overlap. A correction term is then applied to further penalize atoms which significantly overlap the protein.

For Shapegauss a lower score indicates a better result.

PLP

PLP [Verkhivker-2000] or Piecewise Linear Potential scoring function calculates both the shape and hydrogen bond complementarity of poses to the active site.

The PLP score is a pairwise additive scoring function. All pairs of ligand-protein heavy atoms are assigned either a hydrogen bonding potential, if the pair is an acceptor-donor pair, or otherwise a lipophilic potential. These pairwise potentials are summed to obtain the final pose score.

The PLP implementation in the *Docking Toolkit* has also been extended to include interaction between metals on the target protein and acceptors on the ligand.

For PLP a lower score indicates a better result.

Chemscore

Chemscore [Eldridge-1997] is a sum of the following interaction terms

lipophilic

Favorable interactions that occur when two non-polar heavy atoms (one ligand atom and one protein atom) are placed near each other.

hydrogen bonding

Favorable interactions that occur when acceptor-donor interactions are formed between the ligand and protein.

metal chelator

Favorable interactions that occur when acceptor atoms on the ligand are placed near metal atoms on the protein.

clash

Penalty term that occurs when heavy atoms on the ligand and protein clash.

rotatable bond

This penalty term is proportional to the number of rotatable bonds on the ligand that are no longer free to rotate when the ligand is docked.

For Chemscore a lower score indicates a better result.

Chemgauss3

The Chemgauss3 scoring function uses Gaussian smoothed potentials to measure the complementarity of ligand poses within the active site. Chemgauss3 recognizes the following types of interactions.

- Shape
- Hydrogen bonding between ligand and protein
- Hydrogen bonding interactions with implicit solvent
- Metal-chelator interactions.

All interaction potentials in Chemgauss are initially constructed using step functions to describe the interaction of atom pairs (or other chemical points) as a function of distance. These interactions are mapped onto a grid that is then convoluted with a spherical Gaussian function, which smoothes the potential making it less sensitive to small changes in the ligand position. Smoothing the score in this way serves two purposes, first docking can be run at lower resolution than would be required if the score were not smooth since small changes in position to do not cause large changes in

score. Second it reduces the error associated with the rigid protein approximation by effectively accounting for the ability of the protein to make small structural re-arrangements to accommodate the ligand.

Shape interactions in Chemgauss are based on a united atom model (*i.e.* only heavy atoms are relevant to the shape calculation). Each ligand heavy atom is assigned a fixed clash penalty score if the distance between it and a protein heavy atom is less than the sum of the VdW radii, otherwise it is assigned a score proportional to the count of the number of protein heavy atoms within 1.25 and 2.5 times the sum of the VdW radii (atoms within 2.5 count one tenth as much as those within 1.25). From this score a penalty equal to two close protein atom contacts is subtracted to represent the VdW interactions with solvent water that are lost when the ligand docks. This score is pre-computed at grid points throughout the active site and the resulting grid is then smoothed.

Hydrogen bonding groups are modeled with one or more lone-pair or polar-hydrogen positions that describe the directionality of potential hydrogen bonds (with respect to the hydrogen bonding group's heavy atom). Donor groups have lone pair positions representing the possible location of the donor hydrogen atoms relative to the donating molecule, while acceptors have lone-pair positions representing the possible locations of the donated hydrogen relative to the acceptor. A hydrogen bond is detected and assigned a constant score when a hydrogen bonding position on the ligand is within 1.0 Angstroms of a complementary hydrogen bonding position on the protein (*i.e.* when the polar-hydrogen position of a donor overlaps the lone-pair position of an acceptor). If the ligand hydrogen bonding group has multiple polar-hydrogens and/or lone-pair positions (groups can be both donors and acceptors) then this calculation is performed for each position and the result is summed. As with all Chemgauss terms the hydrogen bond potential is pre-computed at grid points throughout the site and then smoothed.

Hydrogen bonds with solvent that break when the ligand docks into the active site are penalized by the Chemgauss scoring function. Broken protein-solvent hydrogen bonds are accounted for by calculating how many hydrogen bonds water can make with the protein at the position of each heavy atom of the docked ligand, and a penalty score is assigned which is proportional to the number of hydrogen bonds. Broken ligand-solvent hydrogen bonds are accounted for by calculating desolvation positions around each hydrogen-bonding group on the ligand that represent the positions water could occupy when making a hydrogen bonding interaction with the protein. A penalty is then assessed that is proportional to the number of desolvation positions that can no longer be occupied by water because the water in these positions would clash with the protein. As before, this potential is placed on a grid and smoothed.

Chelating interactions between protein metals and ligand chelating groups are accounted for by Chemgauss (protein-chelator and ligand-metal chelating interactions are not). For each chelator on the ligand one or more chelating-positions are calculated. If a protein metal is within 1.0 Angstroms of any chelating-position of a chelating group then a fixed score is assigned, otherwise a zero score is assigned. As before this potential is placed on a grid and smoothed.

For Chemgauss3 a lower score indicates a better result.

Chemgauss4

The Chemgauss4 is a modification of the Chemgauss3 scoring function that has improved hydrogen bonding and metal chelator (The shape and implicit solvent interaction terms are identical to those in Chemgauss3). The new hydrogen bonding and metal chelator terms have better perception of the directionality of these interactions and also account for hydrogen bond networking effects.

To calculate the hydrogen bonding score for a ligand-protein hydrogen bond two distances are measured.

1. How far the donor heavy atom is from the position the acceptor atom would consider to be an ideal for a hydrogen bonding to form.
2. How far the acceptor heavy atom is from the position the donor atom would consider to be ideal for a hydrogen bonding interaction to occur.

The score for the hydrogen bond interaction is a product of two Gaussian functions of these distances scaled by the strength of the hydrogen bonding groups involved in the interaction.

HBscore = strength*g(distance1)*g(distance2)

To compute the total hydrogen bonding score for the ligand-protein complex the individual pairwise scores are calculated for all protein-ligand donor-acceptor. Individual HB interaction are then eliminated if either the donor or acceptor

exceeds the maximum number of interactions allowed (e.g., a hydroxyl with one hydrogen is not allowed to make more than one donor interaction), with the lowest scoring interactions eliminated first. The final hydrogen bond score is then calculated by summing the scores of the remained individual acceptor-donor interactions.

For Chemgauss4 a lower score indicates a better result.

Chemical Gaussian Overlay (CGO)

The Chemical Gaussian Overlay function (or CGO) is a hybrid scoring function that scores poses based on their similarity to a known bound ligand and the interactions both the docked and bound ligand make with the protein active site. **This scoring function is not implemented by the *OEScore* class.** This scoring function is used by the *OEDock* class during the exhaustive search (see *Docking Algorithm* section) when using the hybrid docking method (see *Hybrid Method* section).

CGO is primarily a ligand-based scoring functions although some information from the protein structure is used as well. The similarities computed are based on the overall shape of the molecules as well as the position of hydrogen bonding and metal chelating groups. This scoring function requires a bound ligand pose along with the structure of the target protein. Typically the ligand structure is obtained from X-ray crystallography along with the structure of the target protein, although a docked ligand could also, in principal, be used.

CGO represents molecules as a set of spherical Gaussian functions describing the shape and chemistry (acceptors, donors and chelators) of the molecule. The Gaussians representing the shape of the molecule are centered at the heavy atom positions, those for donors are centered on polar-hydrogen positions (*i.e.* positions where the donating hydrogen could be when it is involved in a hydrogen bond), those for acceptors are centered on lone-pair positions (*i.e.* positions where a donating hydrogen could be when a hydrogen bond is formed) and those for chelators are centered at chelating positions (*i.e.* locations where a metal could have a chelating interaction). The overlap of the Gaussians on the docked ligand to those on the bound ligand are computed for each type of Gaussian (*e.g.* shape, donor, acceptor and chelator) by summing the overlap of individual pairs of Gaussian. The overlap of each individual pair is calculated by integrating the product of the two. To prevent chemistry not relevant to binding from contributing to the overall score, when calculating the chemistry overlaps (*i.e.* acceptor, donor and chelator) only groups that make the interaction with the protein are accounted for (*e.g.* a chelator that does not interact with a metal on the protein is ignored in the overlap calculation). The sum of all four types of overlaps is the CGO score.

For CGO a lower score indicates a better result.

2.3 Docking

Docking is the process of determining the structure of a ligand bound in the active site of a target protein. In the *Docking Toolkit* this is done with the *OEDock* class that takes a multiconformer representation of a ligand and returns the top scoring pose (or poses if desired) within the active site. Docking is done using an exhaustive search algorithm, followed by optimization of the best poses from the exhaustive search (see *Docking Algorithm* section).

Docking Molecules is an example program that uses the *OEDock* class to dock, score and annotate multiconformer molecules.

2.3.1 Scoring Functions and Search Resolution

The resolution of the exhaustive search and scoring functions are required inputs when constructing a *OEDock* object.

dockMethod is an unsigned int constant from the `OEDockMethod` namespace, that specifies the combination of scoring functions (or method) *OEDock* uses for the exhaustive search and optimization. The available scoring methods are:

Method	Exhaustive Search Scoring	Optimization Scoring
OEDockMethod.Shapegauss	*Shapegauss*	*Shapegauss*
OEDockMethod.PLP	*PLP*	*PLP*
OEDockMethod.Chemgauss3	*Chemgauss3*	*Chemgauss3*
OEDockMethod.Chemgauss4	*Chemgauss3*	*Chemgauss4*
OEDockMethod.Chemscore	*Chemgauss3*	*Chemscore*
OEDockMethod.Hybrid1	*Chemical Gaussian Overlay (CGO)*	*Chemgauss3*
OEDockMethod.Hybrid	*Chemical Gaussian Overlay (CGO)*	*Chemgauss4*

dockResolution is an unsigned int constant from the *OESearchResolution* namespace, that specifies the docking resolution to use. The docking resolution is the rotational and translational stepsize used during the exhaustive search and optimization (the rotational stepsize is the furthest distance any heavy atom of the ligand will move in a single rotational step). The following resolutions are supported

Resolution	Exhaustive Translational	Exhaustive Rotational
OESearchResolution.High	1.0	1.0
OESearchResolution.Standard	1.0	1.5
OESearchResolution.Low	1.5	2.0

Resolution	Optimization Translational	Optimization Rotational
OESearchResolution.High	0.5	0.5
OESearchResolution.Standard	0.5	0.75
OESearchResolution.Low	0.75	1.0

All resolutions are in Angstroms.

See also

Section *Docking Algorithm* for more information docking algorithm.

Hybrid Method

The hybrid (*OEDockMethod.Hybrid2*) docking method is distinguished from other docking methods because it uses information present in the structure of a bound ligand to enhance docking performance.

The bound ligand information is used during the exhaustive search stage of the docking process (see *Docking Algorithm* section) by using the *Chemical Gaussian Overlay (CGO)* scoring function, which scores poses based on how well a docked pose overlays the shape of the bound ligand and mimics the same hydrogen bonding and interactions the bound ligand makes. After the exhaustive search, optimization is performed with *Chemgauss4* which is a standard structure based scoring function.

The final score a ligand receives using the hybrid docking method is based only on its interactions with the protein. Ligand information is used only to guide the selection of poses during the exhaustive search.

Note

In order to use the hybrid docking method the receptor OEDesignUnit must have a bound ligand (see `HasLigand`).

2.3.2 Initialization

An *OEDock* object must be initialized with a receptor object, prior to docking, scoring or annotating any molecules. This is done by passing a OEDesignUnit containing a receptor (see Receptors) to the *OEDock.Initialize* method.

2.3.3 Docking Molecules

Once the *OEDock* object has been initialized molecules are docked using the *OEDock.DockMultiConformerMolecule* method.

Docking requires a multiconformer representation of the molecule as input. Docking selects the top scoring docked pose from the provided ensemble of conformers. The score of the docked molecule can be obtained by calling the `OEMolBase.GetEnergy` method of *pose*.

OEDock can also return alternate as well as top scoring poses of the docked molecule.

2.3.4 Scoring Molecules

The final score of a molecule (using the optimization scoring function) docked with the *OEDock.DockMultiConformerMolecule* method can be obtained by calling the `OEMolBase.GetEnergy`.

OEDock can also recalculate a score of a pose (using the optimization scoring function), and calculate the contribution for each individual component of the score as in the following example.

2.3.5 Docking Algorithm

OEDock docks multiconformer molecules using an exhaustive search that systematically searches rotations and translations of each conformer of the ligand within the active site. Following the exhaustive search the top scoring poses are optimized and assigned a final score. These two steps are described in more detail below

Exhaustive Search

1. Enumerates, to given resolution, every possible rotation and translation of each conformer of the ligand being docked within a box enclosing the active site. The resolution of the exhaustive search is determined by the overall resolution setting of *OEDock* (see *Scoring Functions and Search Resolution* section).
2. Discard out poses that either clash with the protein or extend to far from the binding site using the receptor's negative image outer contour (see section_receptor_negative_image section).
3. If the negative image inner contour is enabled discard any poses that do not have at least one heavy atom that falls within the inner contour. (see section_receptor_negative_image section).
4. Discard any poses that do no match any user specified constraints. (see section_receptor_constraints section).
5. Score all remaining poses. The scoring function used here depends on what scoring method is being used (see *Scoring Functions and Search Resolution* section).
6. Sort poses by score and pass the top scoring poses to optimization. The number of posses passed to optimization depends on the search resolution (see *OESearchResolution* constant namespace).

Optimization

1. Each pose moved to 729 nearby positions, scored and the top scoring position become the new optimized pose. The positions are generated by having the initial pose take one positive and one negative step for each translational and rotational degree of freedom. The resolution of these steps is half that of the exhaustive search, and is determined by the overall resolution setting of *OEDock* (see *Scoring Functions and Search Resolution* section).
2. The best scoring pose of the 729 tests poses is selected as the final docked structure (and score) for the ligand.

2.4 ShapeFit

SHAPEFIT is a pose prediction method that is built up on the concept that similar ligands have similar binding modes in a protein active site [Kelley-2015]. Given a molecule that is known to bind, and a ligand-protein complex containing a similar ligand or the binding pose of a similar ligand, *SHAPEFIT* overlays the docked ligand onto the known bound ligand. The approach used here is an extension of the molecular- shape-based ligand-alignment algorithm used in *ROCS*. Although both are shape-based ligand alignment, unlike *ROCS*, *SHAPEFIT* allows for for intra-molecular flexibility of the docked ligand. *SHAPEFIT* also guards against the docked ligand bumping into the protein, when using a ligand-protein complex as reference system.

When multiple receptors, or bound ligand references are provided, *SHAPEFIT* searches through XRC coordinates of ligands, determines the bound-liagnd best able to predict the pose of the molecule and then generates both a pose and the probability that the pose is correct.

SHAPEFIT's basic algorithm:

1. Given a set of potential complexes or bound-ligands, *SHAPEFIT* chooses the appropriate reference system based on the 2D/3D similarity to the bound ligand. The best reference, in general, has the highest 2D/3D similarity of the input molecule to the chosen bound ligand.

2. After the complex is chosen, a flexible fitting is performed that attempts to maximize the shape and color similarities between the input molecule and the bound ligand while at the same time minimizing the intra-molecular force field on input molecule.

   SHAPEFIT seeds the flexible fit by expanding the poses generated by the original 3D similarity as described in (1) and then applying the shape constraint of the bound ligand.

   As shown in figure *SHAPEFIT Optimization*, *SHAPEFIT* works by first using the known bound ligand to position the input molecule and follows up by using the bound ligand as a shape constraint during force field optimization. While the input molecule is being forced into the shape constraint, the generated pose strain is monitored to avoid generating high strain poses.

3. The interactions between the generated poses and proteins are utilized to identify clash-free poses while simultaneously selecting the poses with the highest Pose probability.

Therefore, SHAPEFIT's optimization attempts to match the molecule being posed to the ligands' known binding mode without creating undue strain on that molecule.

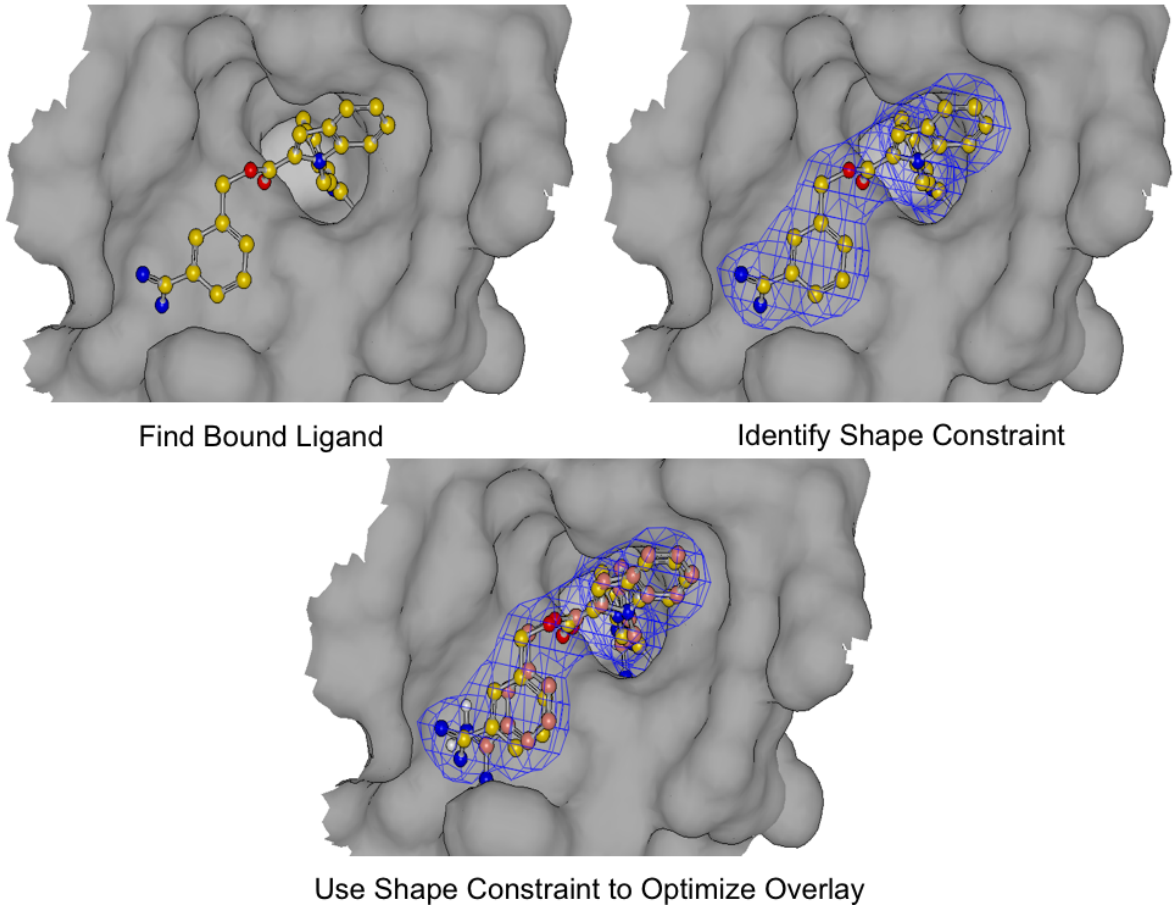


Fig. 1: **SHAPEFIT Optimization:** Starting from an initial alignment, use the shape constraint of the bound-ligand to drive a flexible fit while simultaneously minimizing intra-molecular force field of the fit molecule

As shown in figure *SHAPEFIT Cross Docking Results*, analyzing the Kinase data set used in [Tuccinardi-2010] pose-prediction using SHAPEFIT is seen to perform remarkably better for similar ligands than standard docking techniques at higher *TanimotoCombo* values:

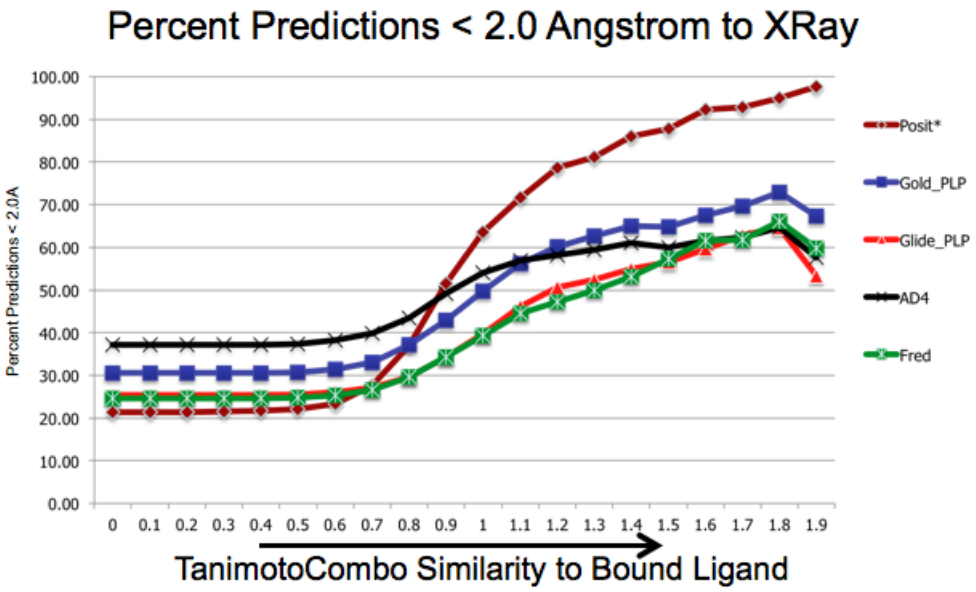


Fig. 2: **SHAPEFIT Cross Docking Results:** Probability of finding a good pose based on bound-ligand fit-ligand *TanimotoCombo* similarity. Standard docking results are essentially the same and follow the same trajectories flattening out as they hit their limit of accuracy. While *SHAPEFIT* performs worse at low similarities it continually increases as similarity increases.

While *SHAPEFIT* is not a good technique for determining the pose between known bound-ligands and fit ligands with low similarity, as the similarity increases, the probability of determining the correct pose increases rapidly. This is most likely because as the similarity increases, the active site similarity also increases.

2.4.1 On Clashes

Unlike the *FRED* and *HYBRID* methods, *SHAPEFIT* is heavily biased towards the known bound ligand. In some cases this causes the pose to clash with the protein. This is especially true if the *original bound ligand* already clashes with the protein.

2.5 POSIT Theory

POSIT is a pose-prediction tool primarily based on the assumption that similar ligands bind similarly. Pose prediction is the process of determining the structure of a ligand bound in the active site of a target protein. Pose prediction with *POSIT* assumes that the incoming ligand binds to the given protein and makes the best effort in correctly placing it in the active site. In conjunction to the predicted pose, **POSIT** also provides an estimate of confidence, in terms of an estimated probability, that the docked pose is within 2.0 Ångströms of the actual binding pose.

> **Note**
>
> Note that this probability is not the probability of binding, rather, the probability that the pose is correct given the ligand actually binds to the receptor.

POSIT consists of multiple *Docking* or *Pose Prediction* methods intentionally and automatically chooses the best method to use for any particular ligand based on the 2D (graph) and 3D (structure) similarity of the docked ligand to the bound ligand. The methods *POSIT* uses to dock are:

1. *ShapeFit* - Shape-guided ligand minimization into the receptor site
2. *Hybrid* - hybrid method that uses ligand and protein information
3. *Fred* - Standard docking method that uses no ligand information

If there is no bound ligand in the receptor, the *Fred* method is used by default since it does not rely upon the pressence of the bound ligand.

2.5.1 Handling of isomerisms and chirality

Stereo, and notably nitrogen aniline stereo centers, are currently somewhat problematic for *POSIT*. Many crystal structures have flat geometries for some stereo centers due to time-averaging during data collection. This makes stereo centers appear to have flat geometries in the 3D coordinates.

Because the POSIT algorithm internally expands conformations during the flexible fitting procedure, the full molecule must be labeled with stereo - either in the 3D coordinate sense or the 2D coordinate sense. This means that some pdb ligand structures will unfortunately be failed by the POSIT algorithm.

To get around this, posit can be told to ignore nitrogen stereo during the conformer generation phase.

> **Note**
>
> Output structures of *Posit* are not guaranteed to have the same conformations of the input molecule. This is due to the fact that force-field minimization is occasionally performed during pose prediction.

2.5.2 Estimated Pose Probability

During a drug discovery campaign, thousands of small molecule inhibitors are made in the course of optimizing molecular properties. For projects that have X-Ray crystallographic (XRC) coordinates, structure-based designs help guide the medicinal chemistry efforts. In many cases XRC provides a detailed picture of the binding of a small-molecule inhibitor into the binding site.

Many techniques exist for pose-prediction and are well documented [Erickson-2004] . However, very few provide a probability that the generated pose is correct where correct is typically considered to be less than 2.0 Ångströms RMSD (root mean square distance) from experimental crystal structure. In fact, many docking scores such as Chemscore, Chemgauss3, PLP [Tuccinardi-2010] are not very correlated with correct ligand pose, and worse are not transferable between systems. The best docking score in one system may not even be close to the best docking score in another.

POSIT overcomes these issues by comparing predicted poses to observed bound ligands in related co-crystals. As the observed ligand becomes more similar to the the predicted pose, both the binding mode and the shape of the receptor pocket itself tends to become more similar.

The similarity measures being used are 2D path-based fingerprints and the 3D *TanimotoCombo* [Hawkins-2010] that compares shape and the Mills-Dean approximation of electrostatics [Mills-Dean-1996] . These similarity measures choose the most appropriate system to dock against (when multiple receptors are available) and provide a prediction of the quality of the result. The *TanimotoCombo* measure is agnostic of how the poses in question are generated; it can be used to validate and provide a pose prediction probability regardless of how the pose was generated.

POSIT probabilities were generated using a large test set containing many pose predictions and verified against an independent set of predictions that were then validated with X-Ray crystallography. It is important to note that *POSIT* does not give a probability of binding. Rather it gives a probability that **if the ligand does actually bind**, what is the likelihood of the *POSIT* pose being the actual pose.

Figure *POSIT Probability MAP* shows how the beliefs given by the 2D and 3D measures are combined into a probability of having a good pose. Remember that this probability has been generated from ligands that actually bind, hence, it is not a probability of binding.

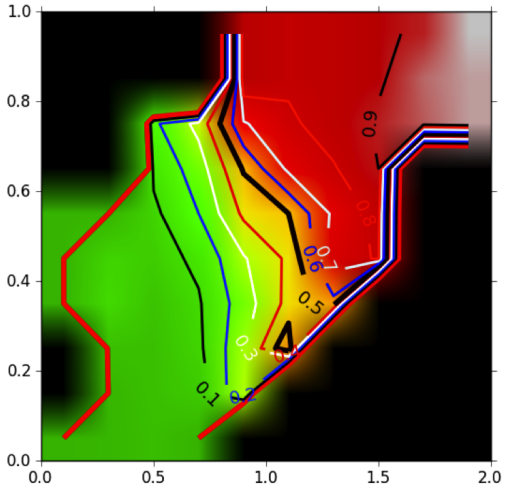


Fig. 3: **POSIT Probability Map:** Given a 2D similarity (in this case the MACCS 166 descriptor set [Durant-2002]) and a 3D similarity (*TanimotoCombo*) *posit* computes a probability of finding the correct pose based on an analysis of historical and experimental data.

This result is different from the result shown in [Tuccinardi-2010] , where they reported that having a high *TanimotoCombo* to the known bound ligand did not dramatically increase the quality of the resulting pose (even for *FRED*). The reason is subtle: Tuccinardi *et al* were computing the highest *TanimotoCombo* that the two molecules could obtain, while *Posit* computes the actual, docked, in-place *TanimotoCombo* of the fitted pose. That is, if the docking algorithm produces an alignment of fit molecule to known bound ligand that overlaps with a given *TanimotoCombo*, one can look up the probability the docking was successful. In point of fact, *Posit* is specifically designed so that the docked pose obtains the highest *TanimotoCombo* score possible while simultaneously minimizing induced strain and maintaining interactions with the protein.

Categorically the data shown in *POSIT Probability MAP*, for each pose can be binned into the following results:

Result	Meaning
GREAT	Computed pose is likely (75%-100% probability) to be within 2.0 Å of experimentally-derived pose.
GOOD	Computed pose may be (50%-75% probability) to be within 2.0 Å of experimentally-derived pose.
MEDIOCRE	Take with a grain of salt (33%-50% probability)
POOR	Take with a huge grain of salt (<33% probability)

2.5.3 Additional MCS Constraints

Walter's *et al* noted that a large portion of ligands bound to the same protein kinase share a large maximum common substructure (MCS). This was the basis for their **CORES** algorithm [Hare-2004] . *Posit* can optionally identify matching regions and use them as additional constraints during conformer generation. *Posit* performs the MCS match to the bound ligand by default.

2.5.4 On Clashes

The definition of *clashes* is somewhat problematic for purposes of pose prediction. In general, serious clashes where interpenetration with the protein should be avoided at all costs. However, when docking into a rigid protein that does not have the appropriate conformation, rigid docking ignores that fact that the active site may adopt a conformation suitable to the posed ligand.

Posit uses the definitions from OEClashInteractionHint to identify clashes. *Posit* allows the users to specify three allowable clash levels.

Allowed Clash	Description
noclashes (none)	No clashes are allowed.
hydrogen (mild clashes)	Only hydrogen clashes are allowed.
allclashes	All clashes are allowed.

2.5.5 Using Multiple Receptors

Using multiple receptors to dock or pose molecules has been shown to greatly increase the reliability of docking.

When multiple receptors are provided *Posit* chooses the appropriate reference system based on the 2D/3D similarity to the bound ligand. The best reference, in general, has the highest 2D/3D similarity of the input molecule to the chosen bound ligand. The similarity measure used here are similar to those used in determining the best docking method in *Posit*.

2.5.6 Receptor Flexibility

Incorporating multiple receptors is the simplest way to account for receptor flexibility during pose prediction. With multiple receptors, *Posit* would pick the best suitable conformer of the receptor for a particular docked ligand.

Pose generated from *Posit* with provided receptor conformers can still sometimes contain clashes. Option is available to trigger post pose prediction relaxation, that relaxes both the posed ligand and protein residues around it.

Note that optimization of a protein-ligand complex can sometimes be time consuming, especially if the target protein and associated components are large.

2.5.7 TanimotoCombo

Posit uses the *TanimotoCombo* measure to compare (and optimize) predicted and bound ligands. The *TanimotoCombo* measure is simply two separate Tanimoto measures added together. While most uses of Tanimoto have been to compare fingerprints together, there is a direct relation between the 1D fingerprint bit vector and 3D space:

The basic equation for a field Tanimoto between two fields A and B is:

$$Tanimoto_{A,B} = \frac{\int A(\vec{r}) * B(\vec{r})}{\int A(\vec{r}) * A(\vec{r}) + \int B(\vec{r}) * B(\vec{r}) - \int A(\vec{r}) * B(\vec{r})}$$

In the case of *Posit*, the field in question can be thought of as field of voxel space. For $Tanimoto_{shape}$, where A and B are now molecules: if two objects fill the same voxels, then the Tanimoto value is 1.0. If two objects overlap by half, the Tanimoto value is 0.5 and so on. (The term voxel is used for purposes edification, in actuality the volumes estimated using a fast approximate method)

$$\text{Overlap(q,t)} = \text{Intersection of Volumes q and t}$$

$$\text{T=} \frac{\text{Overlap(q,t)}}{\text{Overlap(q,q)} + \text{Overlap(t,t)} - \text{Overlap(q,t)}}$$

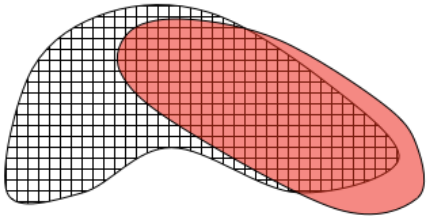


Fig. 4: **Voxel Representation of Shape:** Similar to fingerprint bits in 1D, voxels can be used to represent 3D space and compared with the Tanimoto measure. The numerator *Overlap(q,t)* is essentially the volume of the **intersection** of *q* and *t* and the denominator *Overlap(q,q) + Overlap(t,t) - Overlap(q,t)* is essentially the volume of the **union** of *q* and *t*.

The field can also contain colored representations of chemistry. For example, if two voxels are colored as hydrogen bond donors and overlap, the $Tanimoto_{color}$ increases.

Hence, *TanimotoCombo* is:

$$TanimotoCombo = Tanimoto_{shape} + Tanimoto_{color}$$

TanimotoCombo values range from 0 (no overlap) to 2.0 (full shape overlap and full color or chemistry overlap).

2.6 Posit How to

2.6.1 Ignore Nitrogen stereo

Nitrogen stereo can be ignored in *Posit* by constructing *OEPosit* with an *OEPositOptions* object that has the `OEPositOptions.SetIgnoreNitrogenStereo` method set to `True`.

2.6.2 Prepare molecules for Posit

With default settings, *Posit* internally generates the required input conformers and *Posit* can be used with 2D input molecules. However, if it is desired to pass a pre-generated set of conformers to *Posit*, following are the recommended ways to supply or prepare molecules for **POSIT**.

1. Use *OMEGA* or the *OMEGA toolkit* to prepare molecules or expand the stereo chemistry using *OEFlipper*. POSIT is guaranteed to work correctly if the input molecules have been generated by *OMEGA*.
2. Set the *OEPositOptions* fullConformationalSearch option to be false. See the section *OEPositOptions* for more details. This method of running POSIT is not recommended as a large portion of the flexible methodology will be deactivated.

2.6.3 Use multiple receptors

Multiple receptors can be passed to *Posit* by calling the `AddReceptor` with each of the receptors as shown in the example, `Listing 5`. The `OEPosit` will automatically detect the best receptor for the ligand in action and dock to it.

One of the difficulties of docking to multiple receptors, is that they can consume quite a large amount of memory so holding multiple docking objects at the same time can be problematic. This can be controlled with the `SetPoserCacheCount` method, based on the available hardware capacity and the size of the receptors in use.

2.6.4 Obtain clash-free pose

Pose generated from **Posit** with the default options can sometimes contain clashes. Users have the option to perform pose relaxation, either on the clashed poses or on all generated poses, with the appropriate option using the `SetPoseRelaxMode` method.

CHAPTER

THREE

OEDOCKING EXAMPLES

The following table lists the currently available **OEDocking TK** examples:

Program	Description
docking	docking molecules
rescoring	rescore docked molecules
posing	generating poses
posing_multi	posing with multiple receptors
make_receptor	making a receptor
contour_volume	receptor contour volume
inner_contour	toggle inner contour
outer_contour	set outer contour

3.1 Docking and Scoring Examples

3.1.1 Docking Molecules

The following code example shows how to perform docking using the *OEDock* object.

See also

- *OEDockOptions* class
- *OEDock* class
- *OEHybrid* class

Listing 1: Docking Molecules

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
```

(continues on next page)

(continued from previous page)

```
current license or subscription to the applicable Cadence offering.
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
or its use.
*/
package openeye.examples.oedocking;

import java.net.URL;
import openeye.oechem.*;
import openeye.oebio.*;
import openeye.oedocking.*;

public class DockMolecules {

    public static void main(String[] args) {

        OEDockOptions dockOpts = new OEDockOptions();

        OERefInputAppOptions opts = new OERefInputAppOptions(dockOpts, "DockMolecules",␣
↪OEFileStringType.Mol3D,
            OEFileStringType.Mol3D, OEFileStringType.DU, "-receptor");

        if (oechem.OEConfigureOpts(opts, args, false) == OEOptsConfigureStatus.Help)
            System.exit(0);

        dockOpts.UpdateValues(opts);

        oemolistream ifs = new oemolistream();
        if(!ifs.open(opts.GetInFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetInFile() + " for reading");
        oemolostream ofs = new oemolostream();
        if(!ofs.open(opts.GetOutFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetOutFile() + " for writing");
        oeifstream rfs = new oeifstream();
        if (!rfs.open(opts.GetRefFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetRefFile() + " for reading");

        OEDesignUnit du = new OEDesignUnit();
        if (! oebio.OEReadDesignUnit(opts.GetRefFile(), du))
            oechem.OEThrow.Fatal("Failed to read design unit ");
        if (!du.HasReceptor())
            oechem.OEThrow.Fatal("Design unit " + du.GetTitle() + " does not contain a␣
↪receptor");

        OEDock dock = new OEDock(dockOpts);
        dock.Initialize(du);

        OEMol mol = new OEMol();
        while (oechem.OEReadMolecule(ifs, mol)) {
```

(continues on next page)

(continued from previous page)

```
            System.err.format("docking %s\n", mol.GetTitle());
            OEGraphMol dockedMol = new OEGraphMol();
            int retCode = dock.DockMultiConformerMolecule(dockedMol, mol);
            if (retCode != OEDockingReturnCode.Success) {
                oechem.OEThrow.Error("Docking failed with error code " + oedocking.
↪OEDockingReturnCodeGetName(retCode));
            }
            String sdtag = oedocking.OEDockMethodGetName(dockOpts.GetScoreMethod());
            oedocking.OESetSDScore(dockedMol, dock, sdtag);
            dock.AnnotatePose(dockedMol);
            oechem.OEWriteMolecule(ofs, dockedMol);
        }
        ofs.close();
        ifs.close();
    }
}
```

3.1.2 Rescoring Docked Molecules

The following code example shows how to rescore previously docked molecules, using the *OEScore* object.

See also

- *OEScore* class

Listing 2: Rescoring Docked Molecules

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
current license or subscription to the applicable Cadence offering.
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
or its use.
*/
package openeye.examples.oedocking;

import java.net.URL;
import openeye.oechem.*;
```

(continues on next page)

(continued from previous page)

```
import openeye.oebio.*;
import openeye.oedocking.*;

public class RescorePoses {

    private void rescore(OEInterface itf) {
        OEDesignUnit receptor = new OEDesignUnit();
        if (!oebio.OEReadDesignUnit(itf.GetString("-receptor"), receptor)) {
            oechem.OEThrow.Fatal("Unable to read receptor");
        }

        oemolistream imstr = new oemolistream();
        if (!imstr.open(itf.GetString("-in"))) {
            oechem.OEThrow.Fatal("Unable to open input file of ligands");
        }

        oemolostream omstr = new oemolostream();
        if (!omstr.open(itf.GetString("-out"))) {
            oechem.OEThrow.Fatal("Unable to open out file for rescored ligands");
        }

        int scoreType = oedocking.OEScoreTypeGetValue(itf, "-score");
        OEScore score = new OEScore(scoreType);
        score.Initialize(receptor);

        OEMol ligand = new OEMol();
        boolean optimize = itf.GetBool("-optimize");
        while (oechem.OEReadMolecule(imstr, ligand)) {
            if (optimize) {
                score.SystematicSolidBodyOptimize(ligand);
            }
            score.AnnotatePose(ligand);
            String sdtag = score.GetName();
            oedocking.OESetSDScore(ligand, score, sdtag);
            oechem.OESortConfsBySDTag(ligand, sdtag, score.GetHighScoresAreBetter());
            oechem.OEWriteMolecule(omstr, ligand);
        }
        omstr.close();
        imstr.close();
    }

    public static void main(String argv[]) {
        RescorePoses app = new RescorePoses();
        URL fileURL = app.getClass().getResource("RescorePoses.txt");

        try {
            OEInterface itf = new OEInterface();
            oechem.OEConfigureFromURL(itf, fileURL);
            oedocking.OEScoreTypeConfigure(itf, "-score");
            if (oechem.OEParseCommandLine(itf, argv)) {
                app.rescore(itf);
            }
```

(continues on next page)

(continued from previous page)

```
        } catch (java.io.IOException e) {
            System.err.println("Unable to open interface file");
        }
    }
}
```

3.2 OEShapeFit Examples

3.2.1 Flexible Overlay Optimization with OEShapeFit API

The following code example shows how to flexibly fit input multi conformer molecules in the design unit that contains bound ligand and get the score using the *OEShapeFit* and *OEShapeFitResults* objects.

See also

- `OEFlexiOverlapFunc` class
- `OEFlexiOverlayOptions` class

3.3 POSIT Examples

3.3.1 Generating Poses with POSIT

The following code example shows how to generate a single pose for each input ligand using the *OEPosit* object.

See also

- *OEPositOptions* class
- *OEPosit* class

Listing 4: Generating Poses with POSIT

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
current license or subscription to the applicable Cadence offering.
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
```

(continues on next page)

(continued from previous page)

```
or its use.
*/
package openeye.examples.oedocking;

import java.net.URL;
import openeye.oechem.*;
import openeye.oebio.*;
import openeye.oedocking.*;

public class PoseMolecules {

    public static void main(String[] args) {

        OEPositOptions positOpts = new OEPositOptions();
        OERefInputAppOptions opts = new OERefInputAppOptions(positOpts, "PoseMolecules",␣
↪OEFileStringType.Mol3D,
                OEFileStringType.DU, OEFileStringType.DU, "-receptor");

        if (oechem.OEConfigureOpts(opts, args, false) == OEOptsConfigureStatus.Help)
            System.exit(0);

        positOpts.UpdateValues(opts);

        oemolistream ifs = new oemolistream();
        if (!ifs.open(opts.GetInFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetInFile() + " for reading");

        oeifstream rfs = new oeifstream();
        if (!rfs.open(opts.GetRefFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetRefFile() + " for reading");

        oeofstream ofs = new oeofstream();
        if (!ofs.open(opts.GetOutFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetOutFile() + " for writing");

        OEPosit poser = new OEPosit(positOpts);

        OEDesignUnit du = new OEDesignUnit();
        int count = 0;
        while (oebio.OEReadDesignUnit(rfs, du)) {
            if (!du.HasReceptor())
                oechem.OEThrow.Fatal("Design unit " + du.GetTitle() + " does not contain␣
↪a receptor");
            poser.AddReceptor(du);
            count++;
        }
        if (count == 0)
            oechem.OEThrow.Fatal("Input design unit does not contain any receptor");

        OEMol mcmol = new OEMol();
        while (oechem.OEReadMolecule(ifs, mcmol)) {
            oechem.OEThrow.Info("posing " + mcmol.GetTitle());
```

(continues on next page)

(continued from previous page)

```
            OESinglePoseResult result = new OESinglePoseResult();
            int returnCode = poser.Dock(result, mcmol);

            if (returnCode == OEDockingReturnCode.Success) {
                OEDesignUnit posedDU = new OEDesignUnit(result.GetDesignUnit());
                posedDU.SetDoubleData(poser.GetName(), result.GetProbability());
                oechem.OEThrow
                        .Info("Receptor used: " + posedDU.GetTitle() + " pose␣
↪probability: " + result.GetProbability());
                oebio.OEWriteDesignUnit(ofs, posedDU);
            } else {
                String errMsg = oedocking.OEDockingReturnCodeGetName(returnCode);
                oechem.OEThrow.Warning(mcmol.GetTitle() + " : " + errMsg);
            }
        }
        ofs.close();
        ifs.close();
        rfs.close();
    }
}
```

3.3.2 Generating Multiple Poses with POSIT

The following code example shows how to generate multiple poses for each input ligand, using the *OEPosit* object.

See also

- *OEPositOptions* class
- *OEPosit* class

Listing 5: Posit for Generating Multiple Poses for each Ligand

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
current license or subscription to the applicable Cadence offering.
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
or its use.
*/
```

(continues on next page)

(continued from previous page)

```
package openeye.examples.oedocking;

import java.net.URL;
import java.util.Iterator;
import openeye.oechem.*;
import openeye.oebio.*;
import openeye.oedocking.*;

public class GenerateMultiplePose {

    public static class MyOptions extends OEPositOptions {

        private OEParameter _param1;

        public MyOptions() {
            super();
            OEIntParameter param1 = new OEIntParameter("-numPoses", 1);
            param1.AddLegalRange("1", "20");
            param1.SetBrief("Number of poses to generate");
            _param1 = AddParameter(param1);
        }

        public int GetNumPoses() {
            if (_param1.GetHasValue())
                return (Integer.parseInt(_param1.GetStringValue()));
            return (Integer.parseInt(_param1.GetStringDefault()));
        }

    }

    public static void main(String[] args) {

        MyOptions positOpts = new MyOptions();
        OERefInputAppOptions opts = new OERefInputAppOptions(positOpts, "PoseMolecules",␣
↪OEFileStringType.Mol3D,
                OEFileStringType.DU, OEFileStringType.DU, "-receptor");

        if (oechem.OEConfigureOpts(opts, args, false) == OEOptsConfigureStatus.Help)
            System.exit(0);

        positOpts.UpdateValues(opts);

        oemolistream ifs = new oemolistream();
        if (!ifs.open(opts.GetInFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetInFile() + " for reading");

        oeifstream rfs = new oeifstream();
        if (!rfs.open(opts.GetRefFile()))
            oechem.OEThrow.Fatal("Unable to open " + opts.GetRefFile() + " for reading");

        oeofstream ofs = new oeofstream();
        if (!ofs.open(opts.GetOutFile()))
```

(continues on next page)

(continued from previous page)

```
            oechem.OEThrow.Fatal("Unable to open " + opts.GetOutFile() + " for writing");

        OEPosit poser = new OEPosit(positOpts);

        OEDesignUnit du = new OEDesignUnit();
        int count = 0;
        while (oebio.OEReadDesignUnit(rfs, du)) {
            if (!du.HasReceptor())
                oechem.OEThrow.Fatal("Design unit " + du.GetTitle() + " does not contain␣
↪a receptor");
            poser.AddReceptor(du);
            count++;
        }
        if (count == 0)
            oechem.OEThrow.Fatal("Input design unit does not contain any receptor");

        oechem.OEThrow.Info("Number of conformers: " + positOpts.GetNumPoses());
        oechem.OEThrow.Info("Best receptor pose flag: " + positOpts.
↪GetBestReceptorPoseOnly());

        OEMol mcmol = new OEMol();
        while (oechem.OEReadMolecule(ifs, mcmol)) {
            oechem.OEThrow.Info("posing " + mcmol.GetTitle());
            OEPositResults results = new OEPositResults();

            int returnCode = poser.Dock(results, mcmol, positOpts.GetNumPoses());

            if (returnCode == OEDockingReturnCode.Success) {
                for (OESinglePoseResult result : results.GetSinglePoseResults()) {
                    OEDesignUnit posedDU = new OEDesignUnit(result.GetDesignUnit());
                    posedDU.SetDoubleData(poser.GetName(), result.GetProbability());
                    oechem.OEThrow.Info(
                            "Receptor used: " + posedDU.GetTitle() + " pose probability:
↪" + result.GetProbability());
                    oebio.OEWriteDesignUnit(ofs, posedDU);
                }

            } else {
                String errMsg = oedocking.OEDockingReturnCodeGetName(returnCode);
                oechem.OEThrow.Warning(mcmol.GetTitle() + " : " + errMsg);
            }
        }

        ofs.close();
        ifs.close();
        rfs.close();
    }
}
```

3.4 Receptor Examples

3.4.1 Making a Receptor

The following code example shows how to make a receptor.

See also

- *OEMakeReceptorOptions* class
- *OEMakeReceptor* method

Listing 6: Making a Receptor

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
current license or subscription to the applicable Cadence offering.
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
or its use.
*/
package openeye.examples.oedocking;

import openeye.oechem.*;
import openeye.oedocking.*;
import openeye.oebio.*;

public class MakeReceptor {

    public static void main(String[] args) {
        OEMakeReceptorOptions recOpts = new OEMakeReceptorOptions();
        OESimpleAppOptions opts = new OESimpleAppOptions(recOpts, "MakeReceptor",␣
↪OEFileStringType.DU, OEFileStringType.DU);

        if (oechem.OEConfigureOpts(opts, args, false) == OEOptsConfigureStatus.Help)
            System.exit(0);

        recOpts.UpdateValues(opts);

        oeifstream ifs = new oeifstream(opts.GetInFile());
        oeofstream ofs = new oeofstream(opts.GetOutFile());
```

(continues on next page)

(continued from previous page)

```
        OEDesignUnit du = new OEDesignUnit();
        while (oebio.OEReadDesignUnit(ifs, du))
            if (oedocking.OEMakeReceptor(du, recOpts))
                oebio.OEWriteDesignUnit(ofs, du);
            else
                oechem.OEThrow.Fatal(du.GetTitle() + " failed to make a receptor");

        ofs.close();
        ifs.close();
    }
}
```

3.4.2 Receptor Contour Volume

The following code example shows how to estimate a receptor outer contour volume.

See also

- OEReceptor class
- OEDesignUnit class

Listing 7: Receptor Contour Volume

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
current license or subscription to the applicable Cadence offering.
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
or its use.
*/
package openeye.examples.oedocking;

import openeye.oechem.*;
import openeye.oebio.*;
import openeye.oedocking.*;

public class ReceptorOuterContourVolume {

    public static void main(String argv[]) {
```

(continues on next page)

(continued from previous page)

```
        if (argv.length != 1) {
            oechem.OEThrow.Usage("ReceptorOuterContourVolume <receptor>");
        }

        OEDesignUnit du = new OEDesignUnit();

        if (!oebio.OEReadDesignUnit(argv[0], du))
            oechem.OEThrow.Fatal(argv[0]+" is not a valid receptor.");
        if (!du.HasReceptor())
            oechem.OEThrow.Fatal("Design unit " + du.GetTitle() + " does not contain a␣
↪receptor");

        OEReceptor receptor = du.GetReceptor();
        OEFloatGrid negativeImagePotential = receptor.GetNegativeImageGrid();
        float outerContourLevel = receptor.GetOuterContourLevel();

        int outerCount = 0;
        for (int i=0; i<negativeImagePotential.GetSize();++i) {
            if (negativeImagePotential.GetValue(i) >= outerContourLevel) {
                outerCount += 1;
            }
        }
        double countToVolume = Math.pow(negativeImagePotential.GetSpacing(),3);

        System.err.format("%.2f cubic Angstroms\n", outerCount * countToVolume);
    }
}
```

3.4.3 Toggle Receptor Inner Contour

The following code example shows how to toggle receptor contour volume.

See also

- – OEReceptor class
- OEDesignUnit class

Listing 8: Toggle Receptor Inner Contour

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
current license or subscription to the applicable Cadence offering.
```

(continues on next page)

(continued from previous page)

```
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
or its use.
*/
package openeye.examples.oedocking;

import openeye.oechem.*;
import openeye.oebio.*;
import openeye.oedocking.*;

public class ToggleInnerContour {
    static {
        oechem.OEThrow.SetStrict(true);
    }

    public static void main(String argv[]) {
        if (argv.length != 1) {
            oechem.OEThrow.Usage("ToggleInnerContour <receptor>");
        }

        OEDesignUnit du = new OEDesignUnit();
        if (!oebio.OEReadDesignUnit(argv[0], du))
            oechem.OEThrow.Fatal("Unable to open receptor du file");

        if (!du.HasReceptor())
            oechem.OEThrow.Fatal("Design unit " + du.GetTitle() + " does not contain a␣
↪receptor");


        OEReceptor receptor = du.GetReceptor();

        float innerContourLevel = receptor.GetInnerContourLevel();
        receptor.SetInnerContourLevel(-innerContourLevel);

        if (innerContourLevel > 0.0) {
            oechem.OEThrow.Info("Toggling inner contour off");
        } else {
            oechem.OEThrow.Info("Toggling inner contour on");
        }

        if (!oebio.OEWriteDesignUnit(argv[0], du)) {
            oechem.OEThrow.Fatal("Unable to write updated receptor");
        }
    }
}
```

3.4.4 Set Receptor Contour Volume

The following code example shows how to change the receptor outer contour volume.

See also

- – OEReceptor class
- OEDesignUnit class

Listing 9: Set Receptor Contour Volume

```
/*
(C) 2022 Cadence Design Systems, Inc. (Cadence)
All rights reserved.
TERMS FOR USE OF SAMPLE CODE The software below ("Sample Code") is
provided to current licensees or subscribers of Cadence products or
SaaS offerings (each a "Customer").
Customer is hereby permitted to use, copy, and modify the Sample Code,
subject to these terms. Cadence claims no rights to Customer's
modifications. Modification of Sample Code is at Customer's sole and
exclusive risk. Sample Code may require Customer to have a then
current license or subscription to the applicable Cadence offering.
THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED.  OPENEYE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be
liable for any damages or liability in connection with the Sample Code
or its use.
*/
package openeye.examples.oedocking;

import java.util.ArrayList;
import java.util.Collections;
import openeye.oechem.*;
import openeye.oebio.*;
import openeye.oedocking.*;

public class SetOuterContourVolume {

    public static void main(String argv[]) {
        if (argv.length != 2) {
            oechem.OEThrow.Usage("SetOuterContourVolume <receptor> <volume>");
        }

        OEDesignUnit du = new OEDesignUnit();
        if (!oebio.OEReadDesignUnit(argv[0], du))
            oechem.OEThrow.Fatal("Unable to open receptor file");
        if (!du.HasReceptor())
            oechem.OEThrow.Fatal("Design unit " + du.GetTitle() + " does not contain a␣
↪receptor");

        OEReceptor receptor = du.GetReceptor();
```

(continues on next page)

(continued from previous page)

```
        float outerContourVolume = Float.valueOf(argv[1].trim()).floatValue();

        OEFloatGrid negativeImagePotential = receptor.GetNegativeImageGrid();

        ArrayList<Float> gridElement = new ArrayList<Float>();
        for (int i=0;i < negativeImagePotential.GetSize(); i++) {
            gridElement.add(negativeImagePotential.GetValue(i));
        }
        Collections.sort(gridElement, Collections.reverseOrder());

        float outerContourLevel = gridElement.get(gridElement.size()-1);
        float countToVolume = (float)Math.pow(negativeImagePotential.GetSpacing(),3);
        int ilevel = Math.round(outerContourVolume / countToVolume);
        if (ilevel < gridElement.size()) {
            outerContourLevel = gridElement.get(ilevel);
        }

        receptor.SetOuterContourLevel(outerContourLevel);

        if (!oebio.OEWriteDesignUnit(argv[0], du)) {
            oechem.OEThrow.Fatal("Unable to write updated receptor");
        }
    }
}
```

CHAPTER

FOUR

OEMODELS EXAMPLES

The following table lists the currently available **OEModels** examples:

4.1 Building ROCS Query Model

The following code example shows how to create a ROCS query model from a set of known ligands in their binding modes. The input ligands should be prepared by aligning the protein crystal structures.

See also

- *OEROCSQueryModelBuilder* class

CHAPTER

FIVE

API

5.1 OEDocking API

5.1.1 OEDocking Classes

OEDockOptions

```
class OEDockOptions : public OESystem::OEOptions
```

This class provides an interface to setup options required for docking with *OEDock*.

The *OEDockOptions* class defines the following public methods:

- *GetResolution* and *SetResolution*
- *GetScoreMethod* and *SetScoreMethod*

Constructor

```
OEDockOptions()
OEDockOptions(const OEDockOptions &)
```

Default and copy constructors.

GetResolution

```
unsigned GetResolution() const
```

See *SetResolution* method.

GetScoreMethod

```
unsigned GetScoreMethod() const
```

See *SetScoreMethod* method.

SetResolution

```
bool SetResolution(const unsigned)
```

Sets the **search resolution** for docking. Allowed resolutions are defined in the *OESearchResolution* namespace. Default is *OESearchResolution.Standard*.

SetScoreMethod

```
bool SetScoreMethod(const unsigned)
```

Sets the **scoring method** for docking. Allowed methods are defined in the *OEDockMethod* namespace. Default is *OEDockMethod.Chemgauss4*.

OEDock

```
class OEDock
```

OEDock is used to dock and score multiconformer molecules in an active site.

Use the following procedure to dock and score molecules with this class.

1. Initialize the active site by passing a design unit containing a receptor (see chapter_receptor chapter) to the *OEDock.Initialize* method.
2. Dock molecules using the *OEDock.DockMultiConformerMolecule* method.
3. Score docked molecules with any of the following methods
   - *OEDock.ScoreLigand*
   - *OEDock.ScoreAtom*
   - *OEDock.ScoreLigandComponent*
   - *OEDock.ScoreAtomComponent*
   - *OEDock.AnnotatePose*

Constructor

```
OEDock(unsigned int scoring = OEDockMethod::Default,
unsigned int resolution = OESearchResolution::Default)
OEDock(const OEDockOptions&)
OEDock(const OEDock&)
```

Default and copy constructors.

IsInitialized

```
bool IsInitialized() const
```

Returns true if this object has been successfully initialized and is ready to dock molecules.

Initialize

```
bool Initialize(const OEBio::OEDesignUnit&)
bool Initialize(const OEChem::OEMolBase&)
```

Sets up this object to dock molecules into a receptor. This function returns true if initialization was successful. This object does not depend upon receptor once this call is completed (*i.e.* receptor can be destroyed after calling this method). The first overload expects a design unit that contains a receptor. The second overload works on a variation of molecule that contains receptor data.

Note: This method can take several minutes to complete with larger active sites.

DockMultiConformerMolecule

Both overloads of this method dock *inputMol* into the receptor passed to the *OEDock.Initialize* method.

The return value of this method describes the result of the docking, with a value from the *OEDockingReturnCode* namespace. A result of *OEDockingReturnCode.Success* indicates docking was successful.

```
unsigned int DockMultiConformerMolecule(OEChem::OEMolBase& dockedMol,
                                        const OEChem::OEMCMolBase& inputMol)
```

This overload of *OEDock.DockMultiConformerMolecule* returns the top scoring pose.

dockedMol
: Top scoring docked pose of *inputMol*.

inputMol
: A multiconformer representation of a molecule to dock. Must have 3D coordinates assigned.

The score of the docked pose can be obtained by calling the `GetEnergy` method of *dockedMol*.

```
unsigned int DockMultiConformerMolecule(OEChem::OEMCMolBase& dockedMol,
                                        const OEChem::OEMCMolBase& inputMol,
                                        unsigned int numPoses = 1)
```

This overload of *OEDock.DockMultiConformerMolecule* can return alternate docked poses, in addition to the top scoring pose.

dockedMol
: Docked poses of *inputMol*. Poses are stored as conformers of the OEMCMolBase and are sorted by score.

inputMol
: A multiconformer representation of a molecule to dock. Must have 3D coordinates assigned.

numPoses
: Maximum number of top scoring docked poses to return in *dockedMol*. Typically this will be the number of poses returned, however, in highly restricted sites fewer than *numPoses* may be returned. The value of *numPoses* must be greater than zero.

The score of the docked poses can be obtained by calling the `GetEnergy` method on the conformers of *dockedMol*.

GetHighScoresAreBetter

```
bool GetHighScoresAreBetter() const
```

Returns true if higher scores indicate a better result.

Returns false if lower scores indicate a better result.

GetName

```
std::string GetName() const
```

Returns the name of the scoring function docked poses are scored with.

GetComponentNames

```
OESystem::OEIterBase<const std::string>* GetComponentNames() const
```

ScoreLigand

```
float ScoreLigand(const OEChem::OEMolBase& pose)
```

Rescores a pose within the active site.

pose
Structure of a pose within the active site

If an error occurs this function will return FLT_MAX if *OEDock.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

ScoreAtom

```
float ScoreAtom(const OEChem::OEAtomBase* atom,
const OEChem::OEMolBase& pose)
```

Returns the score of an atom of a given pose within the active site.

atom
Atom of *pose* to score.

pose
Structure of a pose within the active site

If an error occurs this function will return FLT_MAX if *OEDock.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

ScoreLigandComponent

```
float ScoreLigandComponent(const OEChem::OEMolBase& pose,
                           std::string compName)
```

Returns the given components contribution to the total score.

pose
Structure of a pose within the active site

compName
Name of the score component to report. Name must be one returned by *OEDock.GetComponentNames*.

If an error occurs this function will return FLT_MAX if *OEDock.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

ScoreAtomComponent

```
float ScoreAtomComponent(const OEChem::OEAtomBase* atom,
const OEChem::OEMolBase& pose, std::string compName)
```

Returns the given components contribution to the score of a given atom score.

atom
Atom of *pose* to score.

pose
Structure of a pose within the active site

compName
Name of the score component to report. Name must be one returned by *OEDock.GetComponentNames*.

If an error occurs this function will return FLT_MAX if *OEDock.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

AnnotatePose

Adds **VIDA** scoring annotation to the *pose* or *poses* passed.

The annotated poses must be written out in either *oeb* or *oeb.gz* format and are only viewable in **VIDA**.

```
bool AnnotatePose(OEChem::OEMolBase& pose)
```

This overload of *OEDock.AnnotatePose* annotates a single pose.

pose
Structure of a pose within the active site

```
bool AnnotatePose(OEChem::OEMCMolBase& poses)
```

This overload of *OEDock.AnnotatePose* annotates all poses of a given ligand.

poses
An OEMCMolBase the conformers of which are poses within the active site.

CacheScoringSetup

This function caches the current scoring setup of the *OEDock* object onto a receptor object. When another *OEDock* object is initialized with this receptor the it will read in the cached score data rather than recalculating it from scratch, thus improving the startup time of the *OEDock* object.

```
bool CacheScoringSetup(OEBio::OEDesignUnit& receptor,
                       bool clearOldData = true)
bool CacheScoringSetup(OEChem::OEMolBase& receptor,
                       bool clearOldData = true)
```

receptor
A receptor object. This must be the same receptor object the *OEDock* object was initialized with or an exact copy.

clearOldData
Flag to clear cached data from a prior call to CacheScoringSetup with this receptor. The cached data can be quite sizable (hundreds of megabytes), so leaving this flag at the default value of true is recommended.

Note

The cached score data on a receptor will be saved when the receptor is written to a file.

OEHybrid

```
class OEHybrid : public OEDock
```

OEHybrid is used to dock and score multiconformer molecules in an active site using the *Hybrid Method*.

The following methods are publicly inherited from *OEDock*:

- *IsInitialized*
- *Initialize*
- *DockMultiConformerMolecule*
- *GetHighScoresAreBetter*
- *GetName*
- *GetComponentNames*
- *ScoreLigand*
- *ScoreAtom*
- *ScoreLigandComponent*
- *ScoreAtomComponent*
- *AnnotatePose*
- *CacheScoringSetup*

Constructor

```
OEHybrid(unsigned int scoring = OEDockMethod::Hybrid2,
unsigned int resolution = OESearchResolution::Default)
```

Default and copy constructors.

OEMakeReceptorOptions

```
class OEMakeReceptorOptions : public OESystem::OEOptions
```

This class provides an interface to setup options required for *OEMakeReceptor*.

The *OEMakeReceptorOptions* class defines the following public methods:

- *GetAutoConstraints* and *SetAutoConstraints*
- *GetBoxExtension* and *SetBoxExtension*
- *GetBoxMol* and *SetBoxMol*
- *GetNegativeImageType* and *SetNegativeImageType*
- *GetTargetMask* and *SetTargetMask*
- *GetTargetPred* and *SetTargetPred*

Constructor

```
OEMakeReceptorOptions()
OEMakeReceptorOptions(const OEMakeReceptorOptions &)
```

Default and copy constructors.

GetAutoConstraints

```
bool GetAutoConstraints() const
```

See *SetAutoConstraints* method.

GetBoxExtension

```
double GetBoxExtension() const
```

See *SetBoxExtension* method.

GetBoxMol

```
const OEChem::OEMolBase& GetBoxMol() const
```

See *SetBoxMol* method.

GetNegativeImageType

```
unsigned GetNegativeImageType() const
```

See *SetNegativeImageType* method.

GetTargetMask

```
unsigned GetTargetMask() const
```

See *SetTargetMask* method.

GetTargetPred

```
std::string GetTargetPred() const
```

See *SetTargetPred* method.

SetAutoConstraints

```
bool SetAutoConstraints(const bool)
```

Sets flag if protein constraints should be automatically detected and added to the receptor. The auto generated constraints are kept as disabled. Default: True.

SetBoxExtension

```
bool SetBoxExtension(const double)
```

Sets the **amount** for each box face to be extended relative to the minimum box enclosing the *BoxMol*. Default: 0.0.

SetBoxMol

```
bool SetBoxMol(const OEChem::OEMolBase& value)
```

Sets the **molecule** representing the active site on the proteinStructure. The minimum size box that encloses the molecule plus any desired *extension* is used as the active site encloser.

SetNegativeImageType

```
bool SetNegativeImageType(const unsigned)
```

Sets the **negative image type** to be generated. Allowed methods are defined in the *OENegativeImageType* namespace. Default is *OENegativeImageType.StandardShape*.

SetTargetMask

```
bool SetTargetMask(const unsigned)
```

Sets the **target mask** specifying the components of the design unit to be used as receptor target. Allowed methods are defined in the OEDesignUnitComponents namespace. Default: targetComplexNoSolvent.

SetTargetPred

```
bool SetTargetPred(const std::string)
```

Sets the **target predicate** which subsets the components in the TargetMask from the design unit if it matches anything in a specific component.

OEPoseOptimizer

```
class OEPoseOptimizer
```

The *OEPoseOptimizer* can be used to optimize a pose generated from *OEPosit* calculation, to reduce clashes.

The *OEPoseOptimizer* class defines the following public methods:

- *Optimize*

Constructor

```
OEPoseOptimizer(const OEPoseOptimizerOptions& opts = OEPoseOptimizerOptions())
OEPoseOptimizer(const OEPoseOptimizer &)
```

Default and copy constructors.

Assignment operator.

Optimize

```
unsigned Optimize(OEBio::OEDesignUnit& du,
                  const unsigned proteinMask,
                  const unsigned ligandMask)
```

Optimizes the pair of protein and ligand in the posed design unit. Returns `true` if optimization was successful.

OEPoseOptimizerOptions

```
class OEPoseOptimizerOptions : public OESystem::OEOptions
```

This class provides an interface to setup options required for docking with *OEDock*.

The *OEPoseOptimizerOptions* class defines the following public methods:

- *GetFlexRange* and *SetFlexRange*
- *GetForceField* and *SetForceField*
- *GetNonBondCutoff* and *SetNonBondCutoff*

Constructor

```
OEPoseOptimizerOptions()
OEPoseOptimizerOptions(const OEPoseOptimizerOptions &)
```

Default and copy constructors.

GetFlexRange

```
double GetFlexRange() const
```

See *SetFlexRange* method.

GetForceField

```
const OEFF::OEComplexFF* GetForceField() const
```

See *SetForceField* method.

GetNonBondCutoff

```
double GetNonBondCutoff() const
```

See *SetNonBondCutoff* method.

SetFlexRange

```
bool SetFlexRange(const double)
```

Sets the **distance from ligand** defining residues that should be considered flexible during pose optimization. Default: 2.0

SetForceField

```
bool SetForceField(const unsigned)
bool SetForceField(const std::string&)
bool SetForceField(const OEFF::OEComplexFF&)
```

Sets the **force field** for pose optimization.

SetNonBondCutoff

```
bool SetNonBondCutoff(const double)
```

Sets the **non-bonded interactions cutoff** during pose optimization. Default: 40.0

OEPosit

```
class OEPosit
```

OEPosit is used to pose predict multiconformer molecules in an active site.

The *OEPosit* class defines the following public methods:

- *AddReceptor*
- *ClearReceptors*
- *Dock*
- *GetHighScoresAreBetter*
- *GetName*
- *GetInvalidScore*
- *MethodChoice*
- *RankDesignUnits*
- *ScoreLigand*

Deprecated methods of the class:

- *DockMultiConformerMolecule*
- *Initialize*
- *RankReceptors*

Constructor

```
OEPosit( const OEPositOptions& options = OEPositOptions() )
OEPosit(const OEPosit &)
```

Default and copy constructors.

Assignment operator.

AddReceptor

```
bool AddReceptor(const OEBio::OEDesignUnit&)
bool AddReceptor(const OEChem::OEMolBase&)
```

Adds a receptor to the list of receptors to pose against. The function returns `true` if receptor is addded successful. The first overload expects a design unit that contains a receptor. The second overload works on a variation of molecule that contains receptor data.

Note

Since **OEPosit** can generate pose using any of the methods described in *OEPositMethod*, constraints present in the receptor may or may not play any role for the specific pose generation. Constraints only effect the generated pose when *OEPositMethod.FRED* or *OEPositMethod.HYBRID* is the underlying method.

ClearReceptors

```
void ClearReceptors()
```

Removes all existing receptors. Removing receptors will un-initialize an already initialized *OEPosit* instance.

Dock

```
unsigned Dock(OESinglePoseResult& res,
              const OEChem::OEMCMolBase& mol) const
unsigned Dock(OEPositResults& res, const OEChem::OEMCMolBase& mol,
              const unsigned numPoses = 1) const
```

Dock the input molecule `mol`. The second overload can return alternate docked poses, in addition to the top scoring pose, with `numPoses` being the maximum desired number of poses.

The return value of this method describes the result of the docking, with a value from the *OEDockingReturnCode* namespace. A result of *OEDockingReturnCode.Success* indicates docking was successful.

GetHighScoresAreBetter

```
bool GetHighScoresAreBetter() const
```

This always returns `true` for `OEPosit`.

GetName

```
std::string GetName() const
```

Returns the name of the scoring function docked poses are scored with.

GetInvalidScore

```
float  GetInvalidScore() const
```

Returns the invalid score returned by OEPosit. These values are set up to sort invalid values in reverse order from the best reported values.

IsInitialized

```
bool IsInitialized() const
```

Returns true if this object has been successfully initialized and is ready to dock molecules.

MethodChoice

```
unsigned MethodChoice(const unsigned recIdx,
                      const OEChem::OEMCMolBase& mol) const
```

Returns the method that would be used for pose prediction for a given ligand against a given receptor.

RankDesignUnits

```
OESystem::OEIterBase<OEBio::OEDesignUnit>* RankDesignUnits(const␣
↪OEChem::OEMCMolBase& mol) const
```

Rank the design units, from best to worst receptors, to pose against, for the specified ligand.

DockMultiConformerMolecule

> **Warning**
>
> This is a deprecated API. Please use *Dock* instead.

Both overloads of this method dock *inputMol* into the receptor added by the *OEPosit.Initialize* method.

The return value of this method describes the result of the docking, with a value from the *OEDockingReturnCode* namespace. A result of *OEDockingReturnCode.Success* indicates docking was successful.

```
unsigned int DockMultiConformerMolecule(OEChem::OEMolBase& dockedMol,
                                        const OEChem::OEMCMolBase& inputMol)
```

This overload of *OEPosit.DockMultiConformerMolecule* returns the top scoring pose.

dockedMol
: Docked pose with the best pose prediction probability of *inputMol*.

inputMol
: A multiconformer representation of a molecule to dock.

The score of the docked pose can be obtained by calling the `GetEnergy` method of *dockedMol*.

Initialize

> **Warning**
>
> This is a deprecated API. Please use *AddReceptor* instead.

```
bool Initialize(const OEChem::OEMolBase& receptor)
```

Initializes *OEPosit* with receptor to pose ligand against. Calling initialize will remove any existing receptors in the current *OEPosit* instance. Function returns `true` if setup was successful and `false` otherwise.

RankReceptors

> **Warning**
>
> This is a deprecated API. Please use *RankDesignUnits* instead.

```
OESystem::OEIterBase<OEChem::OEMolBase>* RankReceptors(const␣
↪OEChem::OEMCMolBase&) const
```

Rank the receptors, from best to worst, to pose against, for the specified ligand.

ScoreLigand

> **Warning**
>
> This is a deprecated API. This method is only compatible with *DockMultiConformerMolecule*, for obtaining a score (pose probability) after the pose has been generated.
>
> Method provides invalid score when called on poses generated using *Dock*. The scores (pose probability) from that calculation are available in the resulting *OESinglePoseResult*.

```
float ScoreLigand(const OEChem::OEMolBase& pose) const
```

Rescores a pose within the active site.

If an error occurs this function will return the limiting large double in C++.

OEPositOptions

```
class OEPositOptions : public OESystem::OEOptions
```

This class provides an interface to setup options required for performing pose prediction using POSIT.

The *OEPositOptions* class defines the following public methods:

- *GetAllowedClashType* and *SetAllowedClashType*
- *GetBestReceptorPoseOnly* and *SetBestReceptorPoseOnly*
- *GetExcludeSelf* and *SetExcludeSelf*
- *GetFlexiOverlayOptions* and *SetFlexiOverlayOptions*

- *GetFullConformationSearch* and *SetFullConformationSearch*
- *GetIgnoreNitrogenStereo* and *SetIgnoreNitrogenStereo*
- *GetMinProbability* and *SetMinProbability*
- *GetPoserCacheCount* and *SetPoserCacheCount*
- *GetPoseRelaxMode* and *SetPoseRelaxMode*
- *GetPoseRelaxOptions* and *SetPoseRelaxOptions*
- *GetPositMethods* and *SetPositMethods*
- *GetTorLib* and *SetTorLib*

Constructor

```
OEPositOptions(const bool fullConformationalSearch = true,
               const unsigned ipositMethods = OEPositMethod::ALL)
OEPositOptions(const OEPositOptions &)
```

Constructors.

fullConformationalSearch
If `true`, a full flexible conformational search is performed during pose prediction. If `false`, only a rigid overlay is used. Default: `true`.

ipositMethods
Bit-mask of POSIT pose-prediction methods to enable. See *OEPositMethod* for valid values. Default: *OEPositMethod.ALL*.

Assignment operator.

GetAllowedClashType

```
unsigned GetAllowedClashType() const
```

See *SetAllowedClashType* method.

GetBestReceptorPoseOnly

```
bool GetBestReceptorPoseOnly() const
```

See *SetBestReceptorPoseOnly* method.

GetExcludeSelf

```
bool GetExcludeSelf() const
```

See *SetExcludeSelf* method.

GetFlexiOverlayOptions

```
OEShapeFitOptions& GetFlexiOverlayOptions()
const OEShapeFitOptions& GetFlexiOverlayOptions() const
```

See *SetFlexiOverlayOptions* method.

GetFullConformationSearch

```
bool GetFullConformationSearch() const
```

See *SetFullConformationSearch* method.

GetIgnoreNitrogenStereo

```
bool GetIgnoreNitrogenStereo() const
```

See *SetIgnoreNitrogenStereo* method.

GetMinProbability

```
double GetMinProbability() const
```

See *SetMinProbability* method.

GetPoserCacheCount

```
unsigned GetPoserCacheCount()const
```

See *SetPoserCacheCount* method.

GetPoseRelaxMode

```
unsigned GetPoseRelaxMode() const
```

See *SetPoseRelaxMode* method.

GetPoseRelaxOptions

```
OEPoseOptimizerOptions& GetPoseRelaxOptions()
const OEPoseOptimizerOptions& GetPoseRelaxOptions() const
```

See *SetPoseRelaxOptions* method.

GetPositMethods

```
unsigned int GetPositMethods() const
```

See *SetPositMethods* method.

GetTorLib

```
const OETorLib* GetTorLib() const
```

See *SetTorLib* method.

SetAllowedClashType

```
bool SetAllowedClashType(const unsigned)
```

Sets the **allowed clash type** between the ligand pose and the protein, for pose generation. Allowed clash types are defined in the *OEAllowedClashType* namespace.

The allowed clash type defines if a generated pose contains a clash or not, as reported in a pose calculation result. The allowed clash type also determines if a relaxation of the pose would be carried out when *SetPoseRelaxMode* is set to *OEPoseRelaxMode.CLASHED*. Default is *OEAllowedClashType.HYDROGEN*.

SetBestReceptorPoseOnly

```
bool SetBestReceptorPoseOnly(const bool)
```

Sets flag if **only the best receptor** should be considered for pose prediction. If this flag set to `false` the pose prediction is performed using all of the receptors and the best set of poses from all of the receptors combined is provided as the results. With the default setting of `true` pose prediction is only performed on the `best receptor` for any given ligand. Note that, setting the flag to `false` could increase the runtime significantly depending on the number of receptors used for the calculation.

SetExcludeSelf

```
bool SetExcludeSelf(const bool)
```

Sets flag **to exclude receptor that contains a given ligand as its bound ligand**, when choosing the best receptor for a ligand during pose prediction. This option is meant to be used when running cross docking validation experiments, and allows the user to prevent docking a ligand to a receptor containing itself, without having to generate a number of leave one out receptor datasets. Default: **False.**

SetFlexiOverlayOptions

```
void SetFlexiOverlayOptions(const OEShapeFitOptions&)
```

Sets *options* related to flexible optimization with shape, color, and forcefield, relevant for pose generation with *OEPositMethod.SHAPEFIT*.

SetFullConformationSearch

```
bool SetFullConformationSearch(const bool)
```

Sets flag **to perform full conformation search**. Setting this option to `false` disables internal rotamer conformer searching during the flexible searching algorithm. If this option is set to `false` and a 3D input is not given, this setting is ignored and a full conformation search is automatically performed. Default: `true`.

SetIgnoreNitrogenStereo

```
bool SetIgnoreNitrogenStereo(const bool)
```

Sets flag **to ignore Nitrogen stereo specifications** in input molecule. Setting this option to `false` ignores Nitrogen stereo specifications in the input molecule, and a value to `true` ensures that any specified Nitrogen stereo information is restored. A molecule would always be processed, irrespective of this setting. When Nitrogen stereo is specified in the incoming molecule, setting flag is set to `true` would respect those stereo specifications. Default: `false`.

SetMinProbability

```
bool SetMinProbability(const double)
```

Sets the minimum pose probability, for poses to be of interest. Poses that has a probability below the minimum value are not relaxed even when the *SetPoseRelaxMode* is set to CLASHED or *ALL*`. Default is `0.33`.

SetPoserCacheCount

```
bool SetPoserCacheCount(const unsigned)
```

Sets the maximum number of posers to be kept in memory cache during **POSIT** calculation. Each poser takes ~0.75GB of memory. Ability to keep more poser in the cache speeds up the calculation when working with multiple receptors. Default is `10`.

SetPoseRelaxMode

```
bool SetPoseRelaxMode(const unsigned)
```

Sets the **trigger to perform relaxation** of the generated pose. The relaxation is performed by allowing flexibility to the ligand and parts of the receptor. Turning on relaxation can significantly slow down the calculations. Pose relaxation modes are defined in the *OEPoseRelaxMode* namespace. Default is *OEPoseRelaxMode.NONE*.

SetPoseRelaxOptions

```
void SetPoseRelaxOptions(const OEPoseOptimizerOptions&)
```

Sets *options* related to post-optimization of the generated pose.

SetPositMethods

```
bool SetPositMethods(const unsigned)
```

Sets the **posing methods** to be used. Methods are set as bits in the supplied integer, see the *OEPositMethod* namespace for more details. Default is *OEPositMethod.ALL*.

SetTorLib

```
bool SetTorLib(const unsigned)
bool SetTorLib(const std::string&)
bool SetTorLib(const OEConfGen::OETorLib&)
```

Sets the **torsion library** that is used for torsion driving during conformer generation. The first overload takes an *unsigned* from the OETorlibType namespace, and the second overload takes the corresponding string values. **Default:** Original

OEPositProbability

```
class OEPositProbability
```

The *OEPositProbability* can be used to estimate confidence, in terms of probability, that a ligand pose can be accurately predicted, with respect to the reference ligand or design unit receptor.

This function is helpful in selecting a receptor or reference ligand to pose against from a group of receptors.

The *OEPositProbability* class defines the following public methods:

- *Estimate*
- *PoseEstimate*
- *SetupRef*

Constructor

```
OEPositProbability()
OEPositProbability(const OEPositProbability &)
```

Default and copy constructors.

Estimate

```
double Estimate(const OEChem::OEMolBase& mol)
double Estimate(const OEChem::OEMCMolBase& mol)
```

Provides the estimated probability. With the OEMolBase variation of the argument, multi-conformer molecule is generated with internal settings before producing an estimate.

PoseEstimate

```
double PoseEstimate(const OEChem::OEMolBase& mol,
                    const unsigned poseMethod = OEPositMethod::UNKNOWN)
```

Provides estimated probability assuming that the specified ligand pose was generated with the reference used in *SetupRef*. The second argument `poseMethod` refers to specific methods from *OEPositMethod*.

SetupRef

```
bool SetupRef(const OEChem::OEMolBase& ligand)
bool SetupRef(const OEBio::OEDesignUnit& du)
```

Sets up the reference system against which the pose is to be generated.

OEPositResults

```
class OEPositResults
```

This class provides results of a collection of poses, obtained from *OEPosit* calculation.

The *OEPositResults* class defines the following public methods:

- *GetNumPoses*
- *GetReturnCode*
- *GetSinglePoseResults*

Constructors

```
OEPositResults()
OEPositResults(const OEPositResults&)
```

Default and copy constructors.

GetNumPoses

```
unsigned GetNumPoses() const
```

Returns the number of poses in this collection.

GetReturnCode

```
unsigned GetReturnCode() const
```

Returns the calculation return code, defined in *OEDockingReturnCode* namespace.

GetSinglePoseResults

```
OESystem::OEIterBase<OESinglePoseResult>* GetSinglePoseResults() const
```

Returns an iterator over the *OESinglePoseResult* in this collection.

OEScore

```
class OEScore
```

OEScore is a class for scoring and optimizing poses within the active site. Use of this class is as follows:

1. Initialize the active site by passing either a receptor (see chapter_receptor chapter) or a protein and box to the *OEScore.Initialize* method.
2. Score or optimize the molecules using:
   - *OEScore.ScoreLigand*
   - *OEScore.ScoreAtom*
   - *OEScore.ScoreLigandComponent*
   - *OEScore.ScoreAtomComponent*
   - *OEScore.SystematicSolidBodyOptimize*
   - *OEScore.AnnotatePose*

Constructor

```
OEScore(const unsigned int scoring = OEScoreType::Default)
```

Constructs *OEScore* to use the specified *scoring* function from the *OEScoreType* namespace.

IsInitialized

```
bool IsInitialized() const
```

Returns true if this object has been successfully initialized and is ready to score poses.

Initialize

```
bool Initialize(const OEBio::OEDesignUnit& du)
bool Initialize(const OEChem::OEMolBase& receptor)
bool Initialize(const OEChem::OEMolBase& pro,
                const OEMath::OEBox& box)
```

Sets up this object to score poses within an active site. This function returns true if initialization was successful. This object does not depend upon receptor once this call is completed (*i.e.* receptor can be destroyed after calling this method). The first overload expects a design unit that contains a receptor. The second overload works on a variation of molecule that contains receptor data. The last overload works with a protein and box enclosing the active site. Method returns `true` if setup was successful.

GetName

```
std::string GetName() const
```

Returns the name of the scoring function.

GetComponentNames

```
OESystem::OEIterBase<const std::string>* GetComponentNames() const
```

Returns the name of the scoring function components.

GetHighScoresAreBetter

```
bool GetHighScoresAreBetter() const
```

Returns true if higher scores indicate a better result.

Returns false if lower scores indicate a better result.

ScoreLigand

```
float ScoreLigand(const OEChem::OEMolBase& pose)
```

Scores a pose within the active site.

pose
: Structure of a pose within the active site

If an error occurs this function will return FLT_MAX if *OEScore.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

ScoreAtom

```
float ScoreAtom(const OEChem::OEAtomBase* atom,
const OEChem::OEMolBase& lig)
```

Returns the score of an atom of a given pose within the active site.

atom
Atom of *pose* to score.

pose
Structure of a pose within the active site

If an error occurs this function will return FLT_MAX if *OEScore.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

ScoreLigandComponent

```
float ScoreLigandComponent(const OEChem::OEMolBase& pose,
                           std::string compName)
```

Returns the given component's contribution to the total score.

pose
Structure of a pose within the active site

compName
Name of the score component to report. Name must be one returned by *OEScore.GetComponentNames*.

If an error occurs this function will return FLT_MAX if *OEScore.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

ScoreAtomComponent

```
float ScoreAtomComponent(const OEChem::OEAtomBase* atom,
const OEChem::OEMolBase& lig, std::string compName)
```

Returns the given components contribution to the score of a given atom score.

atom
Atom of *pose* to score.

pose
Structure of a pose within the active site

compName
Name of the score component to report. Name must be one returned by *OEScore.GetComponentNames*.

If an error occurs this function will return FLT_MAX if *OEScore.GetHighScoresAreBetter* returns false or -FLT_MAX otherwise.

SystematicSolidBodyOptimize

Performs a solid body optimization on the *pose* or *poses* passed. Three rotational and three translational degrees of freedom are explored, the pose (and protein) are held rigid. A plus and minus rotational step is taken for each rotational and translational degree of freedom (for a total of 3^6 or 729 positions tested for each pose).

Both overloads return true if no error occurred.

```
bool SystematicSolidBodyOptimize(OEChem::OEMCMolBase& poses,
                                 unsigned int searchResolution =␣
↪OESearchResolution::Default)
```

This overload of *OEScore.SystematicSolidBodyOptimize* optimizes all poses of an OEMCMolBase.

poses
An OEMCMolBase with conformers that are poses within the active site to be optimized.

searchResolution
Search resolution of the optimization (see *OESearchResolution* constant namespace).

The optimized poses are returned in *poses*. This poses retain their original ordering and are not ordered by the new optimized score.

```
bool SystematicSolidBodyOptimize(OEChem::OEMolBase& pose,
                                 unsigned int searchResolution =␣
↪OESearchResolution::Default)
```

This overload of *OEScore.SystematicSolidBodyOptimize* optimizes a single pose.

pose
A pose within the active site to be optimized.

searchResolution
Search resolution of the optimization (see *OESearchResolution* constant namespace).

AnnotatePose

Adds **VIDA** scoring annotation to the *pose* or *poses* passed.

The annotated poses must be written out in either *oeb* or *oeb.gz* format and are only viewable in **VIDA**.

```
bool AnnotatePose(OEChem::OEMolBase& pose)
```

This overload of *OEScore.AnnotatePose* annotates a single pose.

pose
Structure of a pose within the active site

```
bool AnnotatePose(OEChem::OEMCMolBase& poses)
```

This overload of *OEScore.AnnotatePose* annotates all poses of a given ligand.

poses
An OEMCMolBase the conformers of which are poses within the active site.

CacheScoringSetup

This function caches the current scoring setup of the *OEScore* object onto a receptor object. When another *OEScore* object is initialized with this receptor the it will read in the cached score data rather than recalculating it from scratch, thus improving the startup time of the *OEScore* object.

```
bool CacheScoringSetup(OEBio::OEDesignUnit& receptor,
                       bool clearOldData = true)
bool CacheScoringSetup(OEChem::OEMolBase& receptor,
                       bool clearOldData = true)
```

receptor
A receptor object. This must be the same receptor object the *OEScore* object was initialized with or an exact copy.

clearOldData
Flag to clear cached data from a prior call to CacheScoringSetup with this receptor. The cached data can be quite sizable (hundreds of megabytes), so leaving this flag at the default value of true is recommended.

> **Note**
>
> The cached score data on a receptor will be saved when the receptor is written to a file.

OESinglePoseResult

```
class OESinglePoseResult
```

This class provides results of a single pose, obtained from the *OEPosit* calculation.

The *OESinglePoseResult* class defines the following public methods:

- *GetDesignUnit*
- *GetTargetMask*
- *GetTargetPred*
- *GetHasClash*
- *GetNumClashes*
- *GetNumContacts*
- *GetPose*
- *GetPositMethod*
- *GetProbability*
- *GetReceptorIndex*
- *GetRelaxAttempted*
- *GetRelaxed*
- *GetRelaxStatus*
- *GetRelaxStatusStr*

Constructors

```
OESinglePoseResult()
OESinglePoseResult(const OESinglePoseResult&)
```

Default and copy constructors.

GetDesignUnit

```
const OEBio::OEDesignUnit& GetDesignUnit() const
```

Returns the resulting design unit consisting of the target and the ligand pose.

GetTargetMask

```
unsigned GetTargetMask() const
```

Returns the target mask used to generate the receptor. The mask and predicate can be used with the `OEDesignUnit.GetComponents` method to generate the molecule used in receptor generation.

GetTargetPred

```
OEBio::OEAtomMatchResidue GetTargetPred() const
```

Returns the residue based OEAtomMatchResidue predicate used with the corresponding target mask to generate the receptor. The mask and predicate can be used with the `OEDesignUnit.GetComponents` method to generate the molecule used in receptor generation.

See also

- OEAtomMatchResidue class
- `OEDesignUnit.GetComponents` method

GetHasClash

```
bool GetHasClash() const
```

Returns `true` if the pose contains clash, `false` otherwise. The *clash* is defined based on the option used in the *OEPosit* calculation.

GetNumContacts

```
unsigned GetNumContacts() const
```

Returns the *number of contacts* in the pose, based on the *clash* type defined during the *OEPosit* calculation.

GetNumClashes

```
unsigned GetNumClashes() const
```

Returns the *number of clashes* in the pose, based on the *clash* type defined during the *OEPosit* calculation.

GetPose

```
const OEChem::OEMolBase& GetPose() const
```

Returns the ligand pose conformer.

GetPositMethod

```
unsigned GetPositMethod() const
```

Returns the method used to generate this pose. The returned values correspond to *OEPositMethod*.

GetProbability

```
double GetProbability() const
```

Returns the **POSIT** probability corresponding to this pose.

GetReceptorIndex

```
unsigned GetReceptorIndex() const
```

Returns the receptor index corresponding to the order they were added to the *OEPosit* instance.

GetRelaxAttempted

```
bool GetRelaxAttempted() const
```

Returns `true` if relaxation was attempted.

GetRelaxed

```
bool GetRelaxed() const
```

Returns `true` if the pose is relaxed.

GetRelaxStatus

```
unsigned GetRelaxStatus() const
```

Returns `status code` corresponding to the attempted relaxation, with a value from the *OEDockingReturnCode* namespace.

GetRelaxStatusStr

```
std::string GetRelaxStatusStr() const
```

Returns `status` corresponding to the attempted relaxation.

5.1.2 OEDocking Constants

OEAllowedClashType

Constants in this namespace are used to define clash types allowed in a *OEPosit* calculation.

NONE

No clash is allowed

HYDROGEN

Clash involving hydrogen is allowed, but not between heavy atoms

ANY

Any clash is allowed

OEDockingReturnCode

A constant from this namespace is returned by *OEDock.DockMultiConformerMolecule* and *OEPosit.Dock* to indicate the outcome of the docking or posing.

Aborted

Docking was aborted by the user (via the `OETracerBase.Aborted` method)

ConformerGenError

Typically indicates the failure of *OEPosit.Dock* to generate any internal conformations.

CoordError

The geometry of the chemical interactions on the ligand could not be determined. This generally indicates that the ligand is broken in some way (e.g. valence errors).

EmptyLigand

Docking failed because the ligand contained no atoms.

EmptyProtein

The protein passed to the *OEDock.Initialize* or *OEPosit.Initialize* methods contained no atoms.

Failure

All other docking failures.

GridSetupError

One of the scoring grids could not be setup. This generally indicates that the protein structure is broken.

InvalidScore

Calculated score is near numeric limits and not valid.

NoConstraintMatch

The supplied ligand cannot match the docking constraints.

NotInitialized

Docking failed because the *OEDock* or *OEPosit* object has not been initialized (see *OEDock.Initialize* or *OEPosit.Initialize*)

NoValidNonClashPoses

For *OEPosit*, no poses remaining after clash removal.

NoValidPoses

No ligand poses could fit within the active site. Increasing the size of the receptor's outer contour volume (see section_receptor_negative_image section) and re-initializing the *OEDock* object with the new receptor may allow the ligand to dock.

OptimizationError

Optimization of the ligand failed.

OutsideGrid

This error code is not currently used.

ScoreError

Problem assigning scores with `OEMolBase.SetEnergy` method.

Success

Docking was successful.

TypingError

One or more of the atoms on the ligand could not be typed.

OEDockMethod

Constants in this namespace are used by the constructor of *OEDock* and *OEHybrid* (see `OEDock.Constructor` and `OEHybrid.Constructor` respectively) to specify the scoring function used during the exhaustive search and optimization and final scoring.

Shapegauss

Exhaustive Search
Shapegauss

Optimization and Scoring
Shapegauss

PLP

Exhaustive Search
PLP

Optimization and Scoring
PLP

Chemgauss3

Exhaustive Search
Chemgauss3

Optimization and Scoring
Chemgauss3

Chemgauss4

Exhaustive Search
Chemgauss3

Optimization and Scoring
Chemgauss4

Chemgauss

Same as *OEDockMethod.Chemgauss4*

Chemscore

Exhaustive Search
Chemgauss3

Optimization and Scoring
Chemscore

Hybrid1

Exhaustive Search
Chemical Gaussian Overlay (CGO)

Optimization and Scoring
Chemgauss3

Hybrid2

Exhaustive Search
Chemical Gaussian Overlay (CGO)

Optimization and Scoring
Chemgauss4

Hybrid

Same as *OEDockMethod.Hybrid2*

Default

Same as *OEDockMethod.Chemgauss4*

INVALID

Return value used to indicate an error

OENegativeImageType

These constants are used by the *OEMakeReceptor* function to specify the type of negative image it should make.

LargeShape

Create a negative image designed for larger sites. This type of negative image tends to prefer solvent exposed areas over tight cavities.

StandardShape

Standard negative image appropriate for most sites

SmallShape

Create a negative image designed for smaller sites. This type of negative image tends to prefer tight cavities over more solvent exposed areas.

Default

Same as *OENegativeImageType.StandardShape*

INVALID

Return value used to indicate an error

OEPoseRelaxMode

Constants in this namespace are used to define pose relaxation modes (i.e., when to perform a pose relaxation) in a *OEPosit* calculation.

NONE

Do not perform relaxation on any generated pose.

CLASHED

To perform relaxation only on those poses that contain a clash.

ALL

To perform relaxation on all generated poses.

OEPositMethod

Constants in this namespace are used to indicate the available posing methods used by *SetPositMethods* and *GetPositMethods*.

ALL

This is equivalent to "SHAPEFIT | HYBRID | FRED | MCS". This is the default.

FRED

The molecule was posed using the default method. See *OEDockMethod.Chemgauss4*.

HYBRID

The molecule was posed using the HYBRID method. See *OEDockMethod.Hybrid2*.

SHAPEFIT

The molecule was posed using the shapefit method. See *PLP*.

UNKNOWN

The ligand was not posed.

OEProteinConstraintType

Constants in this namespace are used by the *OEProteinConstraint* class to specify the type of ligand atom that is allowed to satisfy a protein docking constraint.

Chelator

A chelating heavy atom on the ligand must interact with the protein metal constraint atom to satisfy the constraint.

Acceptor

A hydrogen bond acceptor on the ligand must interact with the protein donor to satisfy the constraint.

Donor

A hydrogen bond donor on the ligand must interact with the protein acceptor to satisfy the constraint.

Contact

A heavy atom on the ligand must contact the protein constraint atom to satisfy the constraint.

Lipophilic

A non-polar heavy atom on the ligand must contact the protein

Unknown

Unknown constraint type. Most likely this receptor was written by a future version of this toolkit or FRED.

OEScoreType

Constants in this namespace are used by the constructor of *OEScore* (see *OEScore.Constructor*) to specify the scoring function *OEScore* will use.

Shapegauss

Shapegauss.

PLP

PLP.

Chemgauss3

Chemgauss3.

Chemgauss4

Chemgauss4.

Chemgauss

Same as *OEScoreType.Chemgauss4*.

Chemscore

Chemscore.

Default

Same as *OEScoreType.Chemgauss4*.

INVALID

Used as a return type to indicate an error.

OESearchResolution

Constants in this namespace are used by the constructor of *OEDock* (see *OEDock.Constructor*) to specify the search resolution used during the exhaustive search and optimization as well as the number of poses passed from the exhaustive search to the optimization step. It is also used by the *OEScore.SystematicSolidBodyOptimize* method to specify the resolution of the optimization (the exhaustive search values are ignored in this case).

Rotational stepsize is the furthest distance any atom will move in a single rotational step.

All distances are measured in angstroms.

High

Exhaustive Search Resolution

Translational 1.0

Rotational 1.0

Optimization Resolution

Translational 0.5

Rotational 0.5

Number exhaustive search poses optimized during docking

1000

Standard

Exhaustive Search Resolution

Translational 1.0

Rotational 1.5

Optimization Resolution

Translational 0.5

Rotational 0.75

Number exhaustive search poses optimized during docking

100

Low

Exhaustive Search Resolution

Translational 1.5

Rotational 2.0

Optimization Resolution

Translational 0.75

Rotational 1.0

Number exhaustive search poses optimized during docking

100

Default

Same as *OESearchResolution.Standard*

INVALID

Used as a return type to indicate an error

5.1.3 OEDocking Functions

OEAddDockingInteractions

```
bool OEAddDockingInteractions(OEBio::OEInteractionHintContainer& asite)
```

Perceives interactions (contact, hydrogen bonding, chelator) between the two molecules stored in the OEInteractionHintContainer object.

See also

- `OEPerceiveInteractionHints` function of **OEChem TK**
- Example in the Perceive and Print Protein-Ligand Interactions section of the **OEChem TK** manual about accessing the interactions.
- Example in the Depicting Active Site Interactions section of the **Grapheme TK** manual about visualizing the interactions.

OECreateDUFromReceptorMol

```
bool OECreateDUFromReceptorMol(OEBio::OEDesignUnit& du,
                               const OEChem::OEMolBase& recMol)
```

Create a OEDesignUnit object from receptor molecule.

See also

- `OEMakeDesignUnits` function
- *OEMakeReceptor* function

OEDockingGetArch

```
const char* OEDockingGetArch()
```

Returns the architecture of the current version of the OEDocking toolkit. Examples include:

- microsoft-win32-x86
- redhat-RHEL5-x86_64

OEDockingGetLicensee

```
bool OEDockingGetLicensee(std::string& licensee)
```

Returns the LICENSEE from the current valid Docking TK license.

OEDockingGetPlatform

```
const char* OEDockingGetPlatform()
```

Returns the platform, including compiler version, of the current version of the OEDocking toolkit. Examples include:

- microsoft-win32-msvc9-MD-x86
- redhat-RHEL5-g++4.1-x86_64

OEDockingGetRelease

```
const char* OEDockingGetRelease()
```

Returns the version of this toolkit release. For example: `1.6.1`.

OEDockingGetSite

```
bool OEDockingGetSite(std::string& site)
```

Returns the SITE from the current valid OEDocking TK license.

OEDockingGetVersion

```
unsigned int OEDockingGetVersion()
```

Returns the build date of the toolkit as an unsigned int. For example: `20090227`.

OEDockingReturnCodeGetName

```
string OEDockingReturnCodeGetName(unsigned int returnCode)
```

Returns the name of the return code. *returnCode* is a constant from the *OEDockingReturnCode* namespace.

OEDockMethodConfigure

```
bool OEDockMethodConfigure(OESystem::OEInterface& itf,
                           std::string flagName)
```

This function is used in conjunction with the *OEDockMethodGetValue*. It adds a command line flag to the *itf* to specify a dock method to use. Once the command line has been parsed (`OEParseCommandLine`) the *itf* and *flagName* are passed to *OEDockMethodGetValue* that will return the constant from the *OEDockMethod* namespace associated with the dock method the user selected (or the default scoring function if the user didn't specify one).

> **Note**
>
> The parameter added to *itf* will be of type string, name *flagName*, and have legal values associated with the dock method.

OEDockMethodGetName

```
string OEDockMethodGetName(unsigned int method)
```

Returns the name of the scoring function. *method* is a constant from the *OEDockMethod* namespace.

OEDockMethodGetValue

```
unsigned int OEDockMethodGetValue(std::string name)
```

Returns the value of a constant from the *OEDockMethod* namespace associated with the scoring function *name*, or *OEDockMethod.INVALID* if the name is not recognized.

```
unsigned int OEDockMethodGetValue(const OESystem::OEInterface& itf,
                                  std::string flagName)
```

Used in conjunction with the *OEDockMethodConfigure* function. Returns a constant from the *OEDockMethod* namespace associated with the dock method the user specified on the command line, or *OEDockMethod.Default* if the user did not specify a value. *itf* and *flagName* should be the same as those passed to the *OEDockMethodConfigure* function and `OEParseCommandLine` should have been called on *itf*.

OEGetEstimatedPositProbability

```
double OEGetEstimatedPositProbability(const OEChem::OEMCMolBase& ligand,
                                      const OEBio::OEDesignUnit& du)
double OEGetEstimatedPositProbability(const OEChem::OEMCMolBase& ligand,
                                      const OEChem::OEMolBase& receptor)
```

Given a multi conformer ligand and a receptor, return the probability that, if the ligand binds to the receptor, the binding pose of the ligand can be retrieved.

This function is helpful in selecting a receptor to pose against from a group of receptors.

OEGetPositMethod

```
unsigned int OEGetPositMethod(const OEChem::OEMolBase &)
```

Return the OEPositMethod used to pose the molecule. The molecule must be posed first. See the *OEPositMethod* namespace for more details.

OEHasClash

```
bool OEHasClash(const OEChem::OEMolBase& ligand,
                const OEChem::OEMolBase& receptor,
                const unsigned allowedType)
```

Returns `true` if there is clash between the supplied ligand and the receptor. Otherwise, it returns `false`. The third argument, `allowedType`, refers to definition of allowed clash type, according to the *OEAllowedClashType* namespace.

OEMakeBoxMolecule

```
bool OEMakeBoxMolecule(OEChem::OEMolBase& mol,
                       const OEMath::OEBox& box)
```

Creates a molecule (*mol*) out of *box*. *mol* will have 8 nitrogen atoms, one at each corner of the box and a single bond between appropriate atoms.

Note that this molecule is in no way chemically valid. The purpose of this method is to create a molecule representing *box* that can be viewed in a molecular visualizer.

OEMakeReceptor

```
bool OEMakeReceptor(OEBio::OEDesignUnit& designUnit,
                    const OEMakeReceptorOptions& opts = OEMakeReceptorOptions(),
                    OESystem::OETracerBase& tracer = OESystem::OENoTracer)
```

Create a *receptor* object in the *design unit*. This function returns true if successful.

designUnit
: The design unit to receive the receptor.

opts
: Options controlling receptor preparation.

tracer
: Optional tracer for logging messages during receptor creation.

OENegativeImageTypeGetName

```
string OENegativeImageTypeGetName(unsigned int method)
```

Returns the name of the scoring function. *method* is a constant from the *OENegativeImageType* namespace.

OENegativeImageTypeGetValue

```
unsigned int OENegativeImageTypeGetValue(std::string name)
```

Returns the value of a constant from the *OENegativeImageType* namespace associated with the scoring function *name*, or *OENegativeImageType.INVALID* if the name is not recognized.

```
unsigned int OENegativeImageTypeGetValue(const OESystem::OEInterface& itf,
                                         std::string flagName)
```

Used in conjunction with the *OENegativeImageTypeConfigure* function. Returns a constant from the *OENegativeImageType* namespace associated with the negative image type the user specified on the command line, or *OENegativeImageType.Default* if the user did not specify a value. *itf* and *flagName* should be the same as those passed to the *OENegativeImageTypeConfigure* function and `OEParseCommandLine` should have been called on *itf*.

OEPositMethodGetName

```
std::string OEPositMethodGetName(unsigned)
```

Returns method name corresponding to constants defined in *OEPositMethod* namespace.

OEScoreTypeGetName

```
string OEScoreTypeGetName(unsigned int scoring)
```

Returns the name of the scoring function. *scoring* is a constant from this namespace.

OEScoreTypeGetValue

```
unsigned int OEScoreTypeGetValue(std::string name)
```

Returns the value of a constant from this namespace associated with the scoring function *name*, or *OEScoreType.INVALID* if the name is not recognized.

This function is defined within the OEScoreType namespace.

```
unsigned int OEScoreTypeGetValue(const OESystem::OEInterface& itf,
                                 std::string flagName)
```

Used in conjunction with the *OEScoreTypeConfigure* function. Returns a constant from this namespace associated with the scoring function the user specified on the command line, or *OEScoreType.Default* if the user did not specify a value. *itf* and *flagName* should be the same as those passed to the *OEScoreTypeConfigure* function and `OEParseCommandLine` should have been called on *itf*.

OESearchResolutionGetName

```
string OESearchResolutionGetName(unsigned int resolution)
```

Returns the name of the scoring function. *resolution* is a constant from the *OESearchResolution* namespace.

OESearchResolutionGetValue

```
unsigned int OESearchResolutionGetValue(std::string name)
```

Returns the value of a constant from the *OESearchResolution* namespace associated with the scoring function *name*, or *OESearchResolution.INVALID* if the name is not recognized.

```
unsigned int OESearchResolutionGetValue(const OESystem::OEInterface& itf,
                                        std::string flagName)
```

Used in conjunction with the *OESearchResolutionConfigure* function. Returns a constant from the *OESearchResolution* namespace associated with the dock resolution the user specified on the command line, or *OESearchResolution.Default* if the user did not specify a value. *itf* and *flagName* should be the same as those passed to the *OESearchResolutionConfigure* function and should have been called on *itf*.

OESetEnergyScore

```
bool OESetEnergyScore(OEChem::OEMolBase& pose, OEDock& dock)
```

Sets the energy of *pose* (`OEMolBase.GetEnergy`) to the score of the pose calculated with the *OEDock.ScoreLigand* method of *dock*.

```
bool OESetEnergyScore(OEChem::OEMolBase& pose, OEScore& score)
```

Sets the energy of *pose* (`OEMolBase.GetEnergy`) to the score of the pose calculated with the *OEScore.ScoreLigand* method of *score*.

```
bool OESetEnergyScore(OEChem::OEMCMolBase& poses, OEDock& dock,
                      bool sortPoses = false)
```

Sets the energy of each pose of *poses* (`OEMolBase.GetEnergy`) to the score of the pose calculated with the *OEDock.ScoreLigand* method of *dock*.

If *sortPoses* is true the poses of will also be sorted by score.

```
bool OESetEnergyScore(OEChem::OEMCMolBase& poses, OEScore& score,
                      bool sortPoses = false)
```

Sets the energy of each pose of *poses* (`OEMolBase.GetEnergy`) to the score of the pose calculated with the *OEScore.ScoreLigand* method of *score*.

If *sortPoses* is true the poses will also be sorted by score.

OESetEnergyScoreComponent

```
bool OESetEnergyScoreComponent(OEChem::OEMolBase& pose, OEDock& dock,
                               std::string component)
```

Sets the energy of *pose* (`OEMolBase.GetEnergy`) to the *component* of the score calculated with *dock's* *OEDock.ScoreLigandComponent* method.

```
bool OESetEnergyScoreComponent(OEChem::OEMolBase& pose,
                               OEScore& score, std::string component)
```

Sets the energy of *pose* (`OEMolBase.GetEnergy`) to the *component* of the score calculated with *score's* *OEScore.ScoreLigandComponent* method.

```
bool OESetEnergyScoreComponent(OEChem::OEMCMolBase& pose,
                               OEDock& dock, std::string component,
                               bool sortPoses = false)
```

Sets the energy of each pose of *poses* (`OEMolBase.GetEnergy`) to the *component* of the score calculate with *dock's* *OEDock.ScoreLigandComponent* method.

If *sortPoses* is true the poses will also be sorted by score.

```
bool OESetEnergyScoreComponent(OEChem::OEMCMolBase& pose,
                               OEScore& score, std::string component,
                               bool sortPoses = false)
```

Sets the energy of each pose of *poses* (`OEMolBase.GetEnergy`) to the *component* of the score calculated with *score's* *OEScore.ScoreLigandComponent* method.

If *sortPoses* is true the poses will also be sorted by score.

OESetSDScore

```
bool OESetSDScore(OEChem::OEMolBase& pose, OEDock& dock,
                  std::string sdtag)
```

Assigns the score of *pose*, using the *OEDock.ScoreLigand* method of *dock*, to SD data of *pose* with the given *sdtag*.

```
bool OESetSDScore(OEChem::OEMolBase& pose, OEPosit& poser,
                  std::string sdtag)
```

Assigns the score of *pose*, using the *OEPosit.ScoreLigand* method of *poser*, to SD data of *pose* with the given *sdtag*.

```
bool OESetSDScore(OEChem::OEMolBase& pose, OEScore& score,
                  std::string sdtag)
```

Assigns the score of *pose*, using the *OEScore.ScoreLigand* method of *score*, to SD data of *pose* with the given *sdtag*.

```
bool OESetSDScore(OEChem::OEMCMolBase& poses, OEDock& dock,
                  std::string sdtag, bool sortPoses = false)
```

Assigns the score of each pose of *poses*, using the *OEDock.ScoreLigand* method of *dock*, to SD data of the pose with the given *sdtag*.

If *sortPoses* is true the poses will also be sorted by score.

```
bool OESetSDScore(OEChem::OEMCMolBase& poses, OEScore& score,
                  std::string sdtag, bool sortPoses = false)
```

Assigns the score of each pose of *poses*, using the *OEScore.ScoreLigand* method of *score*, to SD data of the pose with the given *sdtag*.

If *sortPoses* is true the poses will also be sorted by score.

OESetSDScoreComponent

```
bool OESetSDScoreComponent(OEChem::OEMolBase& pose, OEDock& dock,
                           std::string component, std::string sdtag)
```

Assigns the *component* of *pose's* score, calculated with *dock's* *OEDock.ScoreLigandComponent* method, to *pose's* SD data (with tag *sdtag*).

```
bool OESetSDScoreComponent(OEChem::OEMolBase& pose, OEScore& score,
                           std::string component, std::string sdtag)
```

Assigns the *component* of *pose's* score, calculated with *score's* *OEScore.ScoreLigandComponent* method, to *pose's* SD data (with tag *sdtag*).

```
bool OESetSDScoreComponent(OEChem::OEMCMolBase& poses, OEDock& dock,
                           std::string component, std::string sdtag,
                           bool sortPoses = false)
```

Assigns the *component* of the score of each pose of *poses*, calculated with *dock's* *OEDock.ScoreLigandComponent* method, to SD data (with tag *sdtag*) on the pose.

If *sortPoses* is true the poses will also be sorted by score.

```
bool OESetSDScoreComponent(OEChem::OEMCMolBase& ligand,
                           OEDock& scoreFxn, std::string component,
                           std::string tag, bool sortPoses = false)
```

Assigns the *component* of the score of each pose of *poses*, calculated with *score's* *OEScore.ScoreLigandComponent* method, to SD data (with tag *sdtag*) on the pose.

If *sortPoses* is true the poses will also be sorted by score.

OESetScore

```
bool OESetScore(OEChem::OEMolBase& pose, OEDock& dock, std::string tag)
```

Assigns the score of *pose*, using the *OEDock.ScoreLigand* method of *dock*, to float generic data of *pose* with the given *tag*.

```
bool OESetScore(OEChem::OEMolBase& pose, OEScore& score,
                std::string tag)
```

Assigns the score of *pose*, using the *OEScore.ScoreLigand* method of *score*, to float generic data of *pose* with the given *tag*.

```
bool OESetScore(OEChem::OEMCMolBase& poses, OEDock& dock,
                std::string tag, bool sortPoses = false)
```

Assigns the score of each pose of *poses*, using the *OEDock.ScoreLigand* method of *dock*, to float generic data of the pose with the given *tag*.

If *sortPoses* is true the poses will also be sorted by score.

```
bool OESetScore(OEChem::OEMCMolBase& poses, OEScore& score,
                std::string tag, bool sortPoses = false)
```

Assigns the score of each pose of *poses*, using the *OEScore.ScoreLigand* method of *score*, to float generic data of the pose with the given *tag*.

If *sortPoses* is true the poses will also be sorted by score.

OESetScoreComponent

```
bool OESetScoreComponent(OEChem::OEMolBase& pose, OEDock& dock,
                         std::string component, std::string tag)
```

Assigns the *component* of *pose's* score, calculated with *dock's* *OEDock.ScoreLigandComponent* method, to *pose's* float generic data (with tag *tag*) on *pose*.

```
bool OESetScoreComponent(OEChem::OEMolBase& pose, OEScore& score,
                         std::string component, std::string tag)
```

Assigns the *component* of *pose's* score, calculated with *score's* *OEScore.ScoreLigandComponent* method, to *pose's* float generic data (with tag *tag*) on *pose*.

```
bool OESetScoreComponent(OEChem::OEMCMolBase& pose, OEDock& dock,
                         std::string component, std::string tag,
                         bool sortPoses = false)
```

Assigns the *component* of the score of each pose of *poses*, calculated with *dock's* *OEDock.ScoreLigandComponent* method, to float generic data (with tag *tag*) on the pose.

If *sortPoses* is true the poses will also be sorted by score.

```
bool OESetScoreComponent(OEChem::OEMCMolBase& poses, OEScore& score,
                         std::string component, std::string tag,
                         bool sortPoses = false)
```

Assigns the *component* of the score of each pose of *poses*, calculated with *score's* *OEScore.ScoreLigandComponent* method, to float generic data (with tag *tag*) on the pose.

If *sortPoses* is true the poses will also be sorted by score.

OESetupBox

```
bool OESetupBox(OEMath::OEBox&, const OEChem::OEMolBase&,
                float addbox = 0.0f)
```

This function sets up the minimum box size that encloses the passed-in molecule, with an option to extend each box face by a specified value *addbox*.

```
bool OESetupBox(OEMath::OEBox&, const OEChem::OEMCMolBase&,
                float addbox = 0.0f)
```

This function sets up the minimum box size that encloses all conformers of the input molecules, with an option to extend each box face by a specified value *addbox*.

5.2 Preliminary OEDocking API

5.2.1 Preliminary OEDocking Classes

OEShapeFit

Attention

This is a preliminary API and may be improved based on user feedback. It is currently available in C++ and Python.

```
class OEShapeFit
```

Note

With the default settings, the *OEShapeFit* flexibly optimizes shape overlap and force field.

The *OEShapeFit* class provides an interface for pose generation using flexible overlay optimization between a reference bound ligand or a bound ligand present in a reference design unit and a fit molecule conformers.

The *OEShapeFit* class defines the following public methods:

- *SetupRef*
- *Fit*

Constructor

```
OEShapeFit(const OEShapeFitOptions& options = OEShapeFitOptions())
OEShapeFit(const OEShapeFit&)
```

Constructors. The parameterized constructor accepts an *OEShapeFitOptions* object; if omitted the defaults are used.

Assignment operator.

SetupRef

```
bool SetupRef(const OEBio::OEDesignUnit&)
bool SetupRef(const OEChem::OEMolBase&)
```

This method defines the interface for setting up the reference for the *OEShapeFit* ShapeFit optimization process.

Fit

```
OESystem::OEIterBase<OEShapeFitResults>* Fit(OEChem::OEMCMolBase& fitmol,
                                             const unsigned numPoses = 1)
```

This method optimizes the shape and chemical similarity between the fit molecule and a bound ligand in the design unit, along with intra-molecular forcefield energies of the fit molecule, simultaneously. The reference design unit must be set using *SetupRef* method.

OEShapeFitOptions

Attention

This is a preliminary API and may be improved based on user feedback. It is currently available in C++ and Python.

```
class OEShapeFitOptions : public OEShape::OEFlexiOverlayOptions
```

This class provides an interface to setup options to generate pose conformers using **flexible overlay optimization** between the bound ligand present in the reference design unit and a fit molecule using *OEShapeFit*.

The *OEShapeFitOptions* class defines the following public methods:

- *GetFullConformationSearch* and *SetFullConformationSearch*
- *GetMaxLocalStrain* and *SetMaxLocalStrain*
- *GetMaxOptSteps* and *SetMaxOptSteps*
- *GetTorLib* and *SetTorLib*
- *GetFlexiOverlapOptions* and *SetFlexiOverlapOptions*
- *GetRigidOverlayOptions* and *SetRigidOverlayOptions*
- *CreateCopy*

Constructor

```
OEShapeFitOptions()
OEShapeFitOptions(const OEShapeFitOptions&)
```

Default and copy constructors.

CreateCopy

```
OEShapeFitOptions* CreateCopy() const
```

Creates and returns a deep copy of this object.

GetFullConformationSearch

```
bool GetFullConformationSearch() const
```

See *SetFullConformationSearch* method.

GetMaxOptSteps

```
unsigned GetMaxOptSteps() const
```

See *SetMaxOptSteps* method.

GetTorLib

```
const OEConfGen::OETorLib* GetTorLib() const
```

See *SetTorLib* method.

GetFlexiOverlapOptions

```
OEFlexiOverlapOptions& GetFlexiOverlapOptions()
const OEFlexiOverlapOptions& GetFlexiOverlapOptions() const
```

Returns a reference to the OEFlexiOverlapOptions instance as currently set. These are options related to flexible overlap optimization using shape, color, and force field. See also *SetFlexiOverlapOptions* method.

GetMaxLocalStrain

```
double GetMaxLocalStrain() const
```

See *SetMaxLocalStrain* method.

GetRigidOverlayOptions

```
OEOverlayOptions& GetRigidOverlayOptions()
const OEOverlayOptions& GetRigidOverlayOptions() const
```

Returns a reference to the OEOverlayOptions instance as currently set. These are options related to rigid overlay optimization using shape and color. See also *SetRigidOverlayOptions* method.

SetFullConformationSearch

```
bool SetFullConformationSearch(const bool)
```

Sets flag **to perform full conformation search**. Setting this option to `true` enables internal rotamer conformer generation, prior to running *OEShapeFit*. Default: `false`.

SetMaxLocalStrain

```
bool SetMaxLocalStrain(const double)
```

Sets the **Number allowable local strain** in the generated hits. **Default: 6.5.**

SetMaxOptSteps

```
bool SetMaxOptSteps(const unsigned)
```

Sets the maximum number of optimization iteration steps in flexible overlay optimization. Default: *2000* steps.

SetTorLib

```
bool SetTorLib(const unsigned)
bool SetTorLib(const std::string&)
bool SetTorLib(const OEConfGen::OETorLib&)
```

Sets the **torsion library** that is used for torsion driving during conformer generation. The first overload takes an *unsigned* from the OETorLibType namespace, and the second overload takes the corresponding string values. **Default:** `OETorLibType.Original`

SetFlexiOverlapOptions

```
void SetFlexiOverlapOptions(const OEFlexiOverlapOptions&)
```

Sets the options related to flexible overlap calculations by passing in OEFlexiOverlapOptions instance.

SetRigidOverlayOptions

```
void SetRigidOverlayOptions(const OEOverlayOptions&)
```

Sets the options related to overlay optimization using shape and color by passing in OEFlexiOverlapOptions instance.

OEShapeFitResults

Attention

This is a preliminary API and may be improved based on user feedback. It is currently available in C++ and Python.

```
class OEShapeFitResults
```

See also

- OEFlexiOverlapResults class
- OEOverlapResults class

The *OEShapeFitResults* class defines the following public method:

- *GetPose*
- *GetScore*

Constructor

```
OEShapeFitResults()
OEShapeFitResults(const OEShapeFitResults&)
```

Default and copy constructors.

GetPose

```
const OEChem::OEMolBase& GetPose() const
```

Returns the flexibly optimized pose.

GetScore

```
double GetScore() const
```

Returns the **estimated probability of accuracy** of the generated pose to the bound ligand.

5.3 Deprecated OEDocking API

5.3.1 OEDocking Classes (Deprecated)

OEAlignReceptorFunction

```
class OEAlignReceptorFunction
```

OEAlignReceptorFunction is a pure virtual class that defines an API for aligning two receptors.

This pure virtual class is intended to be subclassed in order to create an aligned receptor. This is necessary because the receptor has multiple components that need to be aligned properly, such as the inner and outer contours and any bound or extra molecules attached to the receptor.

SetupTarget

```
bool SetupTarget(const OEChem::OEMolBase& target) = 0
```

Called when setting the target molecule.

operator()

```
virtual OESystem::OEIterBase<const OEChem::OETrans>* operator()(const OEChem::OEMolBase&
↪receptor_to_align) const = 0
```

Implement to return the transformations that take receptor_to_align into the targets reference frame.

CreateCopy

```
virtual OEAlignReceptorFunction * CreateCopy() const = 0
```

Must return a fully instantiated OEAlignReceptor.

OEBoxBase

```
class OEBoxBase
```

OEBoxBase class represents a box aligned to the x, y and z coordinate axis.

This pure virtual class is implemented by *OEBox*.

The following free functions take *OEBoxBase*.

OEBoxXMid	*OEBoxXDim*
OEBoxYMid	*OEBoxYDim*
OEBoxZMid	*OEBoxZDim*
OEInBox	*OEBoxTranslate*
OEBoxExtend	*OESetupBox*
OEBoxVolume	*OEBoxArea*
OEMakeBoxMolecule	

Destructor

```
virtual ~OEBoxBase() {}
```

Destructor

GetXMax

```
virtual float GetXMax() const   = 0
```

Returns the maximum X value of the box

GetYMax

```
virtual float GetYMax() const   = 0
```

Returns the maximum Y value of the box

GetZMax

```
virtual float GetZMax() const   = 0
```

Returns the maximum Z value of the box

GetXMin

```
virtual float GetXMin() const   = 0
```

Returns the minimum X value of the box

GetYMin

```
virtual float GetYMin() const   = 0
```

Returns the minimum Y value of the box

GetZMin

```
virtual float GetZMin() const   = 0
```

Returns the minimum z value of the box

Setup

```
virtual bool Setup(float x1,
                   float y1,
                   float z1,
                   float x2,
                   float y2,
                   float z2)   = 0
```

Sets up the box. The max and min X values are the max and min of *x1*, *x2* respectively. Minimum and maximum values of Y and Z are selected in an analogous fashion.

OEBox

```
class OEBox : public OEBoxBase
```

OEBox class represents a box aligned to the x, y and z coordinate axis.

This class is an implementation of the *OEBoxBase* class.

The following free functions can take *OEBox* objects by virtue of their inheritance from *OEBoxBase*.

OEBoxXMid	*OEBoxXDim*
OEBoxYMid	*OEBoxYDim*
OEBoxZMid	*OEBoxZDim*
OEInBox	*OEBoxTranslate*
OEBoxExtend	*OESetupBox*
OEBoxVolume	*OEBoxArea*
OEMakeBoxMolecule	

Constructors

```
OEBox()
```

Default constructor. All xyz min and max values are set to 0.

```
OEBox(float x1, float y1, float z1, float x2, float y2, float z2)
```

Constructor that sets up the box. The max and min X values will be set to the max and min of *x1*, *x2* respectively. Minimum and maximum values of Y and Z are selected in an analogous fashion.

```
OEBox(const OEChem::OEMolBase& mol, float addbox = 0.0f)
```

Constructs a box around the given molecule. The maximum and minimum x, y and z values are set to the maximum and minimum values x, y and z coordinates of any atom of the molecule.

If *addbox* is specified then each edge of the box is extended by *addbox*.

```
OEBox(const OEChem::OEMCMolBase& mol, float addbox = 0.0f)
```

Constructs a box around the given molecule. The maximum and minimum x, y and z values are set to the maximum and minimum values x, y and z coordinates of any atom of any pose of the molecule.

If *addbox* is specified then each edge of the box is extended by *addbox*.

```
OEBox(const OEBox& box)
```

Copy constructor.

```
OEBox(const OEBoxBase& box)
```

Constructs an *OEBox* from an *OEBoxBase*.

OECustomConstraints

```
class OECustomConstraints
```

OECustomConstraints holds a set of custom docking constraints used by *OEDock*. Each custom constraint is defined by an *OEFeature* contained by this class.

To use the constraints specified by an *OECustomConstraints* object the object must be passed to the receptor with *OEReceptorSetCustomConstraints* prior to calling the *OEDock.Initialize* method of *OEDock*.

Constructors

```
OECustomConstraints()
```

Default constructor. Object contains no features in the default constructed state.

```
OECustomConstraints(const OECustomConstraints& )
```

Copy Constructor.

GetFeatures

```
OESystem::OEIterBase<      OEFeature>* GetFeatures(bool enabledOnly = true)
OESystem::OEIterBase<const OEFeature>* GetFeatures(bool enabledOnly = true) const
```

Returns an iterator over all features contained by this class, or all enabled features if *enabledOnly* is true (see *OEFeature.GetEnabled* method).

AddFeature

```
OEFeature* AddFeature()
```

Adds a new *OEFeature* object to this class and returns a pointer to the newly created *OEFeature*. The returned *OEFeature* object is owned by this class, and will be destroyed by the destructor of this class.

DeleteFeature

```
bool DeleteFeature(const OEFeature* feature)
```

Deletes an *OEFeature* owned by this class. Note that after this method is called *feature* will no longer point to a valid *OEFeature* object.

NumFeatures

```
unsigned int NumFeatures(bool enabledOnly = true) const
```

Returns the number of *OEFeature* objects contained by this class.

If *enabledOnly* is true then only *OEFeature* objects for which the method *OEFeature.GetEnabled* returns true will be counted.

Clear

```
bool Clear()
```

Returns this object to its default constructed state, deleting any *OEFeature* objects it may currently be holding.

OEFeature

```
class OEFeature
```

OEFeature is a pure virtual class that defines a single custom docking constraint used by *OEDock*.

OEFeature objects are created by the *OECustomConstraints* class (see *OECustomConstraints.AddFeature*).

To satisfy a docking constraint a pose must either have:

1. At least one atom that matches one of the SMARTS patterns returned by *OEFeature.GetSmarts* **and** falls within one of the spheres returned by *OEFeature.GetSpheres*.
2. At least one heavy atom that falls within one of the spheres returned by *OEFeature.GetSpheres*, **if** the *OEFeature* has no SMARTS patterns.

Destructor

```
virtual ~OEFeature() {}
```

Destructor.

GetSpheres

```
OESystem::OEIterBase<const OESphereBase>* GetSpheres() const
OESystem::OEIterBase<OESphereBase>* GetSpheres()
```

Returns all spheres associated with this custom constraint feature.

AddSphere

```
OESphereBase* AddSphere()
OESphereBase* AddSphere(const OESphereBase& sph)
```

Adds an *OESphereBase* object to this class. If *sph* is passed the created sphere will be a copy of *sph* otherwise a default constructed sphere will be created.

Note that the returned memory is owned by this class and will be destroyed by the destructor of this class.

DeleteSphere

```
bool DeleteSphere(const OEMath::OESphere* sph)
```

Deletes a *OESphereBase* held by this class. Note that after this call *sph* will no longer point to a valid *OESphereBase* object.

NumSpheres

```
unsigned NumSpheres() const
```

Returns the number of spheres held by this object.

ClearSpheres

```
bool ClearSpheres()
```

Returns this object to its default constructed state. Deleting all spheres and SMARTS patterns it currently holds.

AddSmarts

```
bool AddSmarts(const std::string& smarts)
```

Adds a SMARTS pattern to this *OEFeature*.

DeleteSmarts

```
bool DeleteSmarts(const std::string& smarts)
```

Deletes the specified SMARTS pattern from this *OEFeature*.

NumSmarts

```
unsigned NumSmarts() const
```

Returns the number of SMARTS patterns held by this *OEFeature*.

ClearSmarts

```
bool ClearSmarts()
```

Deletes all SMARTS patterns associated with this *OEFeature*.

SetFeatureName

```
void SetFeatureName(const std::string&)
```

Sets the name of this feature.

GetFeatureName

```
std::string GetFeatureName() const
```

Returns the name of the feature.

SetEnabled

```
void SetEnabled(const bool enabled)
```

Sets whether this feature is enabled or not. If the feature is disabled the constraint will be ignored during the docking process.

GetEnabled

```
bool GetEnabled() const
```

Returns whether this feature is enabled. If this function returns false this constraint will be ignored during the docking process.

CreateCopy

```
OEReceptorConstraintFeature* CreateCopy() const
```

Creates a copy of this *OEFeature*. Note that the memory returned by this function is **not** owned by this class, and is the responsibility of the calling function.

OEProteinConstraint

```
class OEProteinConstraint
```

This class holds information about a single receptor protein constraint.

GetAtom

```
OEChem::OEAtomBase* GetAtom() const
```

Hold the protein atom of the receptor the constraint is associated with.

GetType

```
unsigned int GetType() const
```

Type for the constraint (see *OEProteinConstraintType*).

GetEnabled

```
bool GetEnabled() const
```

Returns true if the constraint is enabled and false otherwise.

GetName

```
std::string GetName() const
```

Returns the name of the constraint.

SetAtom

```
void SetAtom(OEChem::OEAtomBase* atom)
```

Sets the atom of the receptor the constraint will be associated with.

SetType

```
void SetType(unsigned int type)
```

Sets the constraint type (see *OEProteinConstraintType*).

SetName

```
void SetEnabled(bool enabled)
```

Sets the name of the constraint.

SetEnabled

```
void SetName(std::string name)
```

Sets the enabled flag of the constraint.

OESphereBase

OESphereBase class represents a sphere.

The following free functions take *OESphereBase*.

OESphereVolume	*OESphereArea*
OEInSphere	

Destructor

```
virtual ~OESphereBase() {}
```

Destructor.

Assignment operator.

GetX

```
virtual float GetX() const  = 0
```

Returns the X coordinate of the center of the sphere

GetY

```
virtual float GetY() const  = 0
```

Returns the Y coordinate of the center of the sphere

GetZ

```
virtual float GetZ() const  = 0
```

Return the Z coordinate of the center of the sphere

GetRad

```
virtual float GetRad() const  = 0
```

Returns the radius of the sphere

SetRad

```
virtual bool SetRad(float rad)  = 0
```

Sets the radius of the sphere to *rad*. Will fail and return false if *rad* < 0.

SetCenter

```
virtual bool SetCenter(float x,
                       float y,
                       float z)  = 0
```

Sets the center of the sphere.

5.3.2 OEDocking Functions (Deprecated)

OEAlignReceptors

```
OESystem::OEIterBase<const OEChem::OEMolBase>* OEAlignReceptors(const OEChem::OEMolBase&␣
↪target_receptor,
                                                                const OEChem::OEMolBase&␣
↪receptor_to_align)
```

Given two receptors (a target and a receptor to align) return an iterator of aligned receptors. In general, on success, the iterator will only contain one aligned receptor and will be empty on failure.

```
OESystem::OEIterBase<const OEChem::OEMolBase>* OEAlignReceptors(const OEChem::OEMolBase&␣
↪target_receptor,
                                                                const OEChem::OEMolBase&␣
↪receptor_to_align,
                                                                const␣
↪OEAlignReceptorFunction &)
```

Align two receptors using the supplied OEAlignReceptorFunction.

OEBoxArea

```
float OEBoxArea(const OEBoxBase& box)
```

Return the area of the box.

OEBoxExtend

```
bool OEBoxExtend(OEBoxBase& box, float x_ext, float y_ext, float z_ext)
```

Extends each side of the *box* by the specified amount. *x_ext* extends the two faces of the box normal to the x-axis. Thus the x dimension of the box will increase by 2 times *x_ext*. The *y_ext* and *z_ext* perform the same function for the y and z faces respectively.

```
bool OEBoxExtend(OEBoxBase& box, float   ext)
```

Extends every face of *box* by *ext*. Thus every dimension of the box will increase by 2 times *ext*.

OEBoxTranslate

```
bool OEBoxTranslate(OEBoxBase& box, float x_trans, float y_trans,
                    float z_trans)
```

Translates *box* a distance specified by (*x_trans*, *y_trans*, *z_trans*)

OEBoxVolume

```
float OEBoxVolume(const OEBoxBase& box)
```

Return the volume of the box

OEBoxXDim

```
float OEBoxXDim(const OEBoxBase& box)
```

Returns the X dimension of the box.

OEBoxXMid

```
float OEBoxXMid(const OEBoxBase& box)
```

Returns the X midpoint of the box.

OEBoxYDim

```
float OEBoxYDim(const OEBoxBase& box)
```

Returns the Y dimension of the box.

OEBoxYMid

```
float OEBoxYMid(const OEBoxBase& box)
```

Returns the Y midpoint of the box.

OEBoxZDim

```
float OEBoxZDim(const OEBoxBase& box)
```

Returns the Z dimension of the box.

OEBoxZMid

```
float OEBoxZMid(const OEBoxBase& box)
```

Returns the Z midpoint of the box.

OEInBox

```
bool OEInBox(const OEBoxBase& box, float x, float y, float z)
```

Returns true if (x , y, z) is within *box*.

Coordinates that lie on an edge of *box* are considered to be in *box*.

OEInSphere

```
bool OEInSphere(const OESphereBase& sph, float x, float y, float z)
```

Returns true if (x , y, z) is within the sphere *sph*.

Coordinates that lie on an edge of *sph* are considered to be in *sph*.

OEIsReceptor

```
bool OEIsReceptor(const OEChem::OEMolBase& receptor)
```

Returns true if *rec* is a receptor.

OEMakeReceptorDeprecated

> **Warning**
>
> This is a deprecated API. The overloads below are deprecated. Please use *OEMakeReceptor* instead.

```
bool OEMakeReceptor(OEChem::OEMolBase& receptor,
                    const OEChem::OEMolBase& proteinStructure,
                    const OEMath::OEBox& box, bool strip_water = true,
                    const unsigned int negImgType = OENegativeImageType::Default)
```

Creates a *receptor* object using a *box* and the protein structure.

***receptor*:**
The created receptor.

***proteinStructure*:**
Structure of a target protein.

***box*:**
An *OEBoxBase* enclosing the active site on the *proteinStructure*. The box should enclose all possible docked ligand positions (heavy atoms of the docked pose must fit within the box). It is not necessary to enclose the protein residues the docked ligand may interact with.

***stripWater*:**
If this flag is true water molecules present in *proteinStructure* will be stripped from the protein structure of the *receptor* and added to the extra molecules information of the *receptor*.

***negImgType*:**
Specifies the type of negative image to create (see *OENegativeImageType* namespace).

This function returns true if successful.

```
bool OEMakeReceptor(OEChem::OEMolBase& receptor,
                    const OEChem::OEMolBase& proteinStructure,
                    const OEChem::OEMolBase& boundLigand,
                    bool stripWater = true,
                    unsigned int negImgType = OENegativeImageType::Default)
```

Creates a *receptor* object using a *boundLigand* and the protein structure.

***receptor*:**
The created receptor.

***proteinStructure*:**
Structure of a target protein.

***boundLigand*:**
The structure of a ligand bound to the active site.

***stripWater*:**
If this flag is true water molecules present in *proteinStructure* will be stripped from the protein structure of the *receptor* and added to the extra molecules information of the *receptor*.

***negImgType*:**
Specifies the type of negative image to create (see *OENegativeImageType* namespace).

This function returns true if successful.

```
bool OEMakeReceptor(OEChem::OEMolBase& receptor,
                    const OEChem::OEMolBase& proteinStructure,
                    const OEChem::OEAtomBase* hint,
                    bool strip_water = true,
                    const unsigned int negImgType = OENegativeImageType::Default)
```

Creates a *receptor* object using the protein structure and the identity of one atom of the protein structure near the active site.

***receptor*:**
The created receptor.

***proteinStructure*:**
Structure of a target protein.

***hint*:**
An atom of *proteinStructure* that is near the binding site.

***stripWater*:**
If this flag is true water molecules present in *proteinStructure* will be stripped from the protein structure of the *receptor* and added to the extra molecules information of the *receptor*.

***negImgType*:**
Specifies the type of negative image to create (see *OENegativeImageType* namespace).

This function returns true if successful.

```
bool OEMakeReceptor(OEChem::OEMolBase& receptor,
                    const OEChem::OEMolBase& proteinStructure,
                    float hintX, float hintY, float hintZ,
                    bool stripWater = true,
                    unsigned int negImgType = OENegativeImageType::Default)
```

Creates a *receptor* object using the protein structure and a coordinate near the active site.

***receptor*:**
The created receptor.

***proteinStructure*:**
Structure of a target protein.

***hintX hintY hintZ*:**
Coordinate near the binding site of *proteinStructure*.

***stripWater*:**
If this flag is true water molecules present in *proteinStructure* will be stripped from the protein structure of the *receptor* and added to the extra molecules information of the *receptor*.

***negImgType*:**
Specifies the type of negative image to create (see *OENegativeImageType* namespace).

Attention

The below API using an OEDesignUnit is currently available in C++ and Python.

```
bool OEMakeReceptor(OEChem::OEMolBase& receptor,
                    const OEBio::OEDesignUnit& designUnit,
                    bool stripWater = true,
                    unsigned int negImgType = OENegativeImageType::Default)
```

Creates a *receptor* object using a *designunit*, where the binding site has already been defined

***receptor*:**
The created receptor.

***designUnit*:**
The designunit.

***stripWater*:**
If this flag is true water molecules present in *designunit* will be stripped from the structure of the *receptor* and added to the extra molecules information of the *receptor*.

***negImgType*:**
Specifies the type of negative image to create (see *OENegativeImageType* namespace).

This function returns true if successful.

OENegativeImageTypeConfigure

```
bool OENegativeImageTypeConfigure(OESystem::OEInterface& itf,
                                  std::string flagName)
```

This function is used in conjunction with the *OENegativeImageTypeGetValue*. It adds a command line flag to the *itf* to specify a negative image type to use. Once the command line has been parsed (`OEParseCommandLine`) the *itf* and *flagName* are passed to *OENegativeImageTypeGetValue* that will return the constant from the *OENegativeImageType* namespace associated with the negative image type the user selected (or the default scoring function if the user didn't specify one).

Note

The parameter added to *itf* will be of type string, name *flagName*, and have legal values associated with the negative image type.

OEReadReceptorFile

```
bool OEReadReceptorFile(OEChem::OEMolBase& receptor,
                        std::string filename)
```

Reads a receptor from file *filename* into *receptor*.

Returns true if successful.

Note

All receptor files must be in either .oeb or .oeb.gz format.

OEReadReceptorFromByteArray

```
boolean OEReadReceptorFromByteArray(OEMolBase mol, String format,
                                    byte[] data)
boolean OEReadReceptorFromByteArray(OEMolBase mol, int format,
                                    int flavor, boolean gzip,
                                    byte[] data)

boolean OEReadReceptorFromByteArray(OEMCMolBase mol, String format,
                                    byte[] data)
boolean OEReadReceptorFromByteArray(OEMCMolBase mol, int format,
```

(continues on next page)

(continued from previous page)

```
                                    int flavor, boolean gzip,
                                    byte[] data)
```

Instantiates a receptor from the contents of 'data' in terms of 'format'. 'format' must be a file extension that is readable by OpenEye, for example: ".oeb.gz". Overloads which take an unsigned 'format', unsigned 'flavor' and Boolean 'gzip' parameters expect the format and corresponding flavor to be specified by one of the constants in `OEFormat` and `OEIFlavor`.

Returns 'true' if decoding was successful and the decoded molecule is identified as a receptor.

```
byte[] bytes = oedocking.OEWriteReceptorToByteArray(".oeb", receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, ".oeb", bytes);
```

```
byte[] bytes = oedocking.OEWriteReceptorToByteArray(".oeb.gz", receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, ".oeb.gz", bytes);
```

```
boolean gzip = false;
byte[] bytes = oedocking.OEWriteReceptorToByteArray(OEFormat.OEB, oechem.
↪OEGetDefaultOFlavor(OEFormat.OEB), gzip, receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, OEFormat.OEB, oechem.
↪OEGetDefaultIFlavor(OEFormat.OEB), gzip, bytes);
```

```
boolean gzip = true;
byte[] bytes = oedocking.OEWriteReceptorToByteArray(OEFormat.OEB, oechem.
↪OEGetDefaultOFlavor(OEFormat.OEB), gzip, receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, OEFormat.OEB, oechem.
↪OEGetDefaultIFlavor(OEFormat.OEB), gzip, bytes);
```

OEReceptorAddCachedScore

```
bool OEReceptorAddCachedScore(OEChem::OEMolBase& rec, OEScore& score)
```

Equivalent to calling *OEScore.CacheScoringSetup* with *clearOldData* false.

```
bool OEReceptorAddCachedScore(OEChem::OEMolBase& rec, OEDock& dock)
```

Equivalent to calling *OEDock.CacheScoringSetup* with *clearOldData* false.

OEReceptorAddExtraMol

```
bool OEReceptorAddExtraMol(OEChem::OEMolBase& receptor,
                           const OEChem::OEMolBase& extraMol)
```

Adds a copy of *extraMol* to *receptor's* extra molecules.

Returns true if successful.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorClear

```
bool OEReceptorClear(OEChem::OEMolBase& receptor)
```

Clears all receptor information from *receptor*, except the structure of the protein.

OEReceptorClearBoundLigand

```
bool OEReceptorClearBoundLigand(OEChem::OEMolBase& receptor)
```

Clears any bound ligand from *receptor*.

Returns true if successful.

OEReceptorClearCachedScore

```
bool OEReceptorClearCachedScore(OEChem::OEMolBase& rec)
```

Clears all score setup caches, from *OEScore* or *OEDock* objects, from the receptor.

OEReceptorClearCustomConstraints

```
bool OEReceptorClearCustomConstraints(OEChem::OEMolBase& receptor)
```

Clears all custom constraints on a receptor.

Returns true if successful.

OEReceptorClearExtraMols

```
bool OEReceptorClearExtraMols(OEChem::OEMolBase& receptor)
```

Clears all extra molecules from *receptor*.

Returns true if successful.

OEReceptorClearProteinConstraint

```
bool OEReceptorClearProteinConstraint(OEChem::OEMolBase& rec,
                                      const OEProteinConstraint& pc)
```

Clears a given protein constraint from *receptor*.

OEReceptorClearProteinConstraints

```
bool OEReceptorClearProteinConstraints(OEChem::OEMolBase& receptor)
```

Clears all protein constraints from *receptor*.

OEReceptorGetBoundLigand

```
OEChem::OEGraphMol OEReceptorGetBoundLigand(const OEChem::OEMolBase& receptor)
```

Returns a copy of the *receptor's* bound ligand.

If the receptor does not have a bound ligand then an empty molecule will be returned

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorGetCustomConstraints

```
OECustomConstraints OEReceptorGetCustomConstraints(const OEChem::OEMolBase& rec)
```

Returns a copy of the *receptor's* custom constraints

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorGetExtraMols

```
OEIterBase<const OEMolBase>* OEReceptorGetExtraMols(const OEChem::OEMolBase& receptor)
```

Returns an iterator over the extra molecules held by *receptor*.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorGetInnerContourLevel

```
float OEReceptorGetInnerContourLevel(const OEChem::OEMolBase& rec)
```

Returns the inner contour *level* of the *receptor*.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorGetNegativeImageGrid

```
OESystem::OEGrid<float> OEReceptorGetNegativeImageGrid(const OEChem::OEMolBase& receptor)
```

Returns a copy of the *receptor*'s negative image grid.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorGetOuterContourLevel

```
float OEReceptorGetOuterContourLevel(const OEChem::OEMolBase& receptor)
```

Returns the outer contour *level* of the *receptor*.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorGetProteinConstraints

```
OESystem::OEIterBase<const OEProteinConstraint>*
OESystem::OEIterBase<const OEProteinConstraint>* OEReceptorGetProteinConstraints(const␣
↪OEChem::OEMolBase& rec,
                                                                                 bool␣
↪enabledOnly = true)
```

Returns an iterator over the *receptor's* protein constraints. If *enabledOnly* is true then only enabled constraints will be returned.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorHasBoundLigand

```
bool OEReceptorHasBoundLigand(const OEChem::OEMolBase& receptor)
```

Returns true if *receptor* is a valid receptor and has a bound ligand.

OEReceptorHasCachedScore

```
bool OEReceptorHasCachedScore(const OEChem::OEMolBase& rec)
```

Returns true if *rec* has a score setup cache for any *OEScore* or *OEDock* object.

OEReceptorHasCustomConstraints

```
bool OEReceptorHasCustomConstraints(const OEChem::OEMolBase& receptor,
                                    bool enabledOnly = true)
```

Returns true if *receptor* has custom constraints. If *enabledOnly* is true then at least one of the constraints must be enabled.

OEReceptorHasExtraMols

```
bool OEReceptorHasExtraMols(const OEChem::OEMolBase& receptor)
```

Returns true if the *receptor* is a valid receptor and has one or more extra molecules.

OEReceptorHasInnerContourLevel

```
bool OEReceptorHasInnerContourLevel(const OEChem::OEMolBase& receptor)
```

Returns true if the *receptor* has an inner contour level set.

> **Note**
>
> This function will return true even if the inner contour is disabled (*i.e.* even if the level is negative).

OEReceptorHasNegativeImageGrid

```
bool OEReceptorHasNegativeImageGrid(const OEChem::OEMolBase& receptor)
```

Returns true if the *receptor* has a negative image grid.

OEReceptorHasOuterContourLevel

```
bool OEReceptorHasOuterContourLevel(const OEChem::OEMolBase& receptor)
```

Returns true if the *receptor* has an outer contour level set.

> **Note**
>
> This function will return true even if the outer contour is disabled (*i.e.* even if the level is negative).

OEReceptorHasProteinConstraints

```
bool OEReceptorHasProteinConstraints(const OEChem::OEMolBase& receptor,
                                     bool enabledOnly = true)
```

Returns true if *receptor* is a valid receptor and has one or more protein constraints. If *enabledOnly* the constraint(s) must also be enabled.

OEReceptorSetBoundLigand

```
bool OEReceptorSetBoundLigand(OEChem::OEMolBase& receptor,
                              const OEChem::OEMolBase& boundLigand)
```

Sets the bound ligand on *receptor* to be a copy of *boundLigand*.

Returns true if successful.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorSetCachedScore

```
bool OEReceptorSetCachedScore(OEChem::OEMolBase& rec, OEScore& score)
```

Equivalent to calling *OEScore.CacheScoringSetup* with *clearOldData* true.

```
bool OEReceptorSetCachedScore(OEChem::OEMolBase& rec, OEDock& dock)
```

Equivalent to calling *OEDock.CacheScoringSetup* with *clearOldData* true.

OEReceptorSetCustomConstraints

```
bool OEReceptorSetCustomConstraints(OEChem::OEMolBase& rec,
                                    const OECustomConstraints& cc)
```

Sets the *receptor's* custom constraints (copied from *customConstraints*).

Returns true if successful.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorSetInnerContourLevel

```
bool OEReceptorSetInnerContourLevel(OEChem::OEMolBase& receptor,
                                    float level)
```

Sets the inner contour *level* on the *receptor*.

Returns true if successful.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorSetOuterContourLevel

```
bool OEReceptorSetOuterContourLevel(OEChem::OEMolBase& receptor,
                                    float level)
```

Sets the outer contour *level* on the *receptor*.

Returns true if successful.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEReceptorSetProteinConstraint

```
bool OEReceptorSetProteinConstraint(OEChem::OEMolBase& rec,
                                    const OEProteinConstraint& pc)
```

Sets a protein constraint. If the *receptor* already has a constraint on *OEProteinConstraint.GetAtom* the constraint will be replaced, otherwise a new one will be created.

Returns true if successful.

It is only valid to call this function if *receptor* is a valid receptor (i.e., *OEIsReceptor* returns true).

OEScoreTypeConfigure

```
bool OEScoreTypeConfigure(OESystem::OEInterface& itf,
                          std::string flagName)
```

This function is used in conjunction with the *OEScoreTypeGetValue*. It adds a command line flag to the *itf* to specify a scoring function to use. Once the command line has been parsed (`OEParseCommandLine`) the *itf* and *flagName* are passed to *OEScoreTypeGetValue* that will return the constant from this namespace associated with the scoring function the user selected (or the default scoring function if the user didn't specify one).

> **Note**
>
> The parameter added to *itf* will be of type string, name *flagName*, and have legal values associated with the scoring functions.

OESearchResolutionConfigure

```
bool OESearchResolutionConfigure(OESystem::OEInterface& itf,
                                 std::string flagName)
```

This function is used in conjunction with the *OESearchResolutionGetValue*. It adds a command line flag to the *itf* to specify a dock resolution to use. Once the command line has been parsed (`OEParseCommandLine`) the *itf* and *flagName* are passed to *OESearchResolutionGetValue* that will return the constant from the *OESearchResolution* namespace associated with the dock resolution the user selected (or the default scoring function if the user didn't specify one).

> **Note**
>
> The parameter added to *itf* will be of type string, name *flagName*, and have legal values associated with the dock resolutions.

OESetupBox

```
bool OESetupBox(OEBoxBase& box, float* xyz, unsigned int N,
                bool addbox = 0.0f)
```

Sets up *box* to contain the minimum size box that encloses all coordinates in the *xyz* array and then extends each box face by *addbox*.

box
: Box to setup

xyz
: Array of coordinates

N
: Number of coordinates in *xyz* array

addbox
: Optional parameter to extend each box face by a specified amount relative to the minimum box enclosing the *xyz* coordinates.

Returns true if setup was successful.

```
bool OESetupBox(OEMath::OEBox& box, const OEChem::OEMolBase& mol,
                float addbox = 0.0f)
```

Sets up *box* to be the minimum size box that encloses *mol* and then extends each box face by *addbox*.

box
: Box to setup

mol
: A molecule the box will be setup around

addbox
: Optional parameter to extend each box face by a specified amount relative to the minimum box enclosing the *mol*.

Returns true if setup was successful.

```
bool OESetupBox(OEMath::OEBox& box, const OEChem::OEMCMolBase& mol,
                float addbox = 0.0f)
```

Sets up *box* to be the minimum size box that encloses all conformers of *mol* and then extends each box face by *addbox*.

box
: Box to setup

mol
: A multiconformer molecule the box will be setup around

addbox
: Optional parameter to extend each box face by a specified amount relative to the minimum box enclosing the *mol*.

Returns true if setup was successful.

OESetupBoxCenterAndExtents

```
bool OESetupBoxCenterAndExtents(OEBoxBase& box, float* center,
                                float* extents)
```

Sets up *box* using a center coordinate and the dimension of the box.

box
Box to setup

center
A length 3 array holds the coordinate of the box center

extends
A length 3 array holding the x, y and z dimensions of the box.

Returns true if setup was successful.

OESphereArea

```
float OESphereArea(const OESphereBase& sph)
```

OESphereVolume

```
float OESphereVolume(const OESphereBase& sph)
```

OEWriteReceptorFile

```
bool OEWriteReceptorFile(const OEChem::OEMolBase& receptor,
                         std::string filename)
bool OEWriteReceptorFile(OEChem::OEMolBase& receptor,
                         std::string filename)
```

Writes a *receptor* to file *filename*

Returns true if successful.

Note

All receptor files must be in either .oeb or .oeb.gz format.

OEWriteReceptorToByteArray

```
byte[] OEWriteReceptorToByteArray(String format, OEMolBase mol)
byte[] OEWriteReceptorToByteArray(int format, int flavor,
                                  boolean gzip, OEMolBase mol)
```

Encodes a receptor in terms of 'format'. 'format' must be a file extension that is writeable by OpenEye, for example: ".oeb.gz". Overloads which take an unsigned 'format', unsigned 'flavor' and Boolean 'gzip' parameters expect the format and flavor to be specified by one of the constants in `OEFormat` and `OEOFlavor`.

Returns a non-empty byte[] if the molecule to encode is identified as a receptor and encoding was successful.

```
byte[] bytes = oedocking.OEWriteReceptorToByteArray(".oeb", receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, ".oeb", bytes);
```

```
byte[] bytes = oedocking.OEWriteReceptorToByteArray(".oeb.gz", receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, ".oeb.gz", bytes);
```

```
boolean gzip = false;
byte[] bytes = oedocking.OEWriteReceptorToByteArray(OEFormat.OEB, oechem.
↪OEGetDefaultOFlavor(OEFormat.OEB), gzip, receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, OEFormat.OEB, oechem.
↪OEGetDefaultIFlavor(OEFormat.OEB), gzip, bytes);
```

```
boolean gzip = true;
byte[] bytes = oedocking.OEWriteReceptorToByteArray(OEFormat.OEB, oechem.
↪OEGetDefaultOFlavor(OEFormat.OEB), gzip, receptor);
OEMolBase receptorIO = new OEGraphMol();
boolean status = oedocking.OEReadReceptorFromByteArray(receptorIO, OEFormat.OEB, oechem.
↪OEGetDefaultIFlavor(OEFormat.OEB), gzip, bytes);
```

5.4 OEModels API

5.4.1 Preliminary OEModels Classes

OEROCSQueryModelBuilder

> **Attention**
>
> This is a preliminary API and may be improved based on user feedback. It is currently available in C++ and Python.

```
class OEROCSQueryModelBuilder
```

The *OEROCSQueryModelBuilder* is used to build models of OEShapeQuery that can be used as reference systems for **ROCS**, OEROCS, OEOverlay, and other similar functions.

The *OEROCSQueryModelBuilder* can be used for building a query from known ligands in their binding modes. The input ligands should be prepared by aligning the protein crystal structures. The prepared OEDesignUnit structures can be used directly, or the ligands can be extracted and used, as input to the query model builder. The query model builder constructs a ROCS query employing a few of the ligands that are most representative of the set as a whole.

The *OEROCSQueryModelBuilder* class defines the following public methods:

- *AddLigand*
- *Build*
- *ClearLigands*
- *GetROCSQueryModelOptions*
- *SetLigands*

Constructor

```
OEROCSQueryModelBuilder(const OEROCSQueryModelOptions& options =␣
↪OEROCSQueryModelOptions())
OEROCSQueryModelBuilder(const OEROCSQueryModelBuilder&)
```

Constructors.

options
Options for the model builder. Default options are used if not provided.

AddLigand

```
bool AddLigand(const OEBio::OEDesignUnit& du)
bool AddLigand(const OEChem::OEMolBase& mol)
```

Adds the specified ligand to the set of ligands from which the query model(s) would be build. The first overload of the method adds the bound ligand from the design unit. Method returns `True` if a ligand is added successfully.

Build

```
OESystem::OEIterBase<OEShape::OEShapeQuery>* Build(OESystem::OETracerBase* tracer = &
↪OESystem::NullTracer)
```

Build the query model(s). Method returns an iterator over the generated OEShapeQuery.

ClearLigands

```
bool ClearLigands()
```

Clears set of ligands from which the query model(s) would be build. Method returns `True` if all the ligands are removed successfully.

GetROCSQueryModelOptions

```
const OEROCSQueryModelOptions& GetROCSQueryModelOptions() const
```

Get the *OEROCSQueryModelOptions* associated with this builder.

SetLigands

```
bool SetLigands(OEChem::oemolistream& ifs)
bool SetLigands(OEPlatform::oeistream& ifs)
```

Sets the ligands from which the query model(s) would be build. Calling this method would clear any existing ligand that were previously added. The first overload adds all the OEMolBase from the specified oemolistream, and the second overload adds the bound ligands from all the OEDesignUnit of the specified oeistream. Method returns `True` if ligands are added successfully.

OEROCSQueryModelOptions

> **Attention**
>
> This is a preliminary API and may be improved based on user feedback. It is currently available in C++ and Python.

```
class OEROCSQueryModelOptions : public OESystem::OEOptions
```

This class provides an interface to setup options required for building a **ROCS** query model using *OEROCSQueryModelBuilder*.

The *OEROCSQueryModelOptions* class defines the following public methods:

- *GetColorForceField* and *SetColorForceField*
- *GetMergeColorAtoms* and *SetMergeColorAtoms*
- *GetMaxMoleculesPerModel* and *SetMaxMoleculesPerModel*
- *GetNumModels* and *SetNumModels*
- *CreateCopy*

Constructor

```
OEROCSQueryModelOptions()
OEROCSQueryModelOptions(const OEROCSQueryModelOptions &)
```

Default and copy constructors.

CreateCopy

```
OEROCSQueryModelOptions* CreateCopy() const
```

Creates and returns a deep copy of this object.

GetColorForceField

```
const OEColorForceField* GetColorForceField() const
```

See *SetColorForceField* method.

GetMergeColorAtoms

```
bool GetMergeColorAtoms() const
```

See *SetMergeColorAtoms* method.

GetMaxMoleculesPerModel

```
unsigned GetMaxMoleculesPerModel() const
```

See *SetMaxMoleculesPerModel* method.

GetNumModels

```
unsigned GetNumModels() const
```

See *SetNumModels* method.

SetColorForceField

```
bool SetColorForceField(unsigned int type)
bool SetColorForceField(OEPlatform::oeistream& is)
bool SetColorForceField(const std::string& filename)
bool SetColorForceField(const OEColorForceField& cff)
```

Sets the color force field to be used. By default the ImplicitMillsDean color force field is used.

SetMergeColorAtoms

```
bool SetMergeColorAtoms(const bool)
```

Sets **flag** if color atoms should be merged if applicable. **Default: True**

SetMaxMoleculesPerModel

```
bool SetMaxMoleculesPerModel(const unsigned)
```

Sets the **maximum number of molecules** to be combined in creating a single query. **Default: 3**

SetNumModels

```
bool SetNumModels(const unsigned)
```

Sets the **maximum number query model** to be returned from modeling building. **Default: 1**

CHAPTER

SIX

RELEASE HISTORY

6.1 OEDocking TK 4.3.5

6.1.1 Minor Bug Fixes

- An issue has been fixed that was causing *OEPosit* to pick the wrong *method* in posing some ligands.

6.2 OEDocking TK 4.3.4

6.2.1 Minor Bug Fixes

- An issue has been fixed where a receptor grid could not be constructed for covalently bonded ligands.

6.3 OEDocking TK 4.3.3

Minor internal improvements have been made.

6.4 OEDocking TK 4.3.2

6.4.1 New features

- The method *Optimize* in *OEPoseOptimizer* now returns an `unsigned` code corresponding to any possible failure, instead of a `bool`. The method also takes additional arguments for protein and ligand masks in the OEDesignUnit.
- The following methods have been added to the *OESinglePoseResult*:
  - *GetRelaxAttempted*
  - *GetRelaxStatus*
  - *GetRelaxStatusStr*
- A new method, *MethodChoice*, has been added to *OEPosit* that provides information on the method that would be used for the pose prediction of a given ligand against a given receptor.
- The new methods *GetTorLib* and *SetTorLib* have been added to *OEPositOptions* to provide a choice for the torsion library that would be used for conformer generation during pose prediction.
- The following methods have been added to the *OEShapeFitOptions*:
  - *GetMaxLocalStrain*
  - *SetMaxLocalStrain*
  - *GetTorLib*

  - *SetTorLib*

6.4.2 Major bug fixes

- The method *Optimize* in *OEPoseOptimizer* is now more robust in optimizing a docked pose. Accordingly, *Docking* with relaxation in *OEPosit* is also now more robust.
- The function *OEHasClash* now properly recognizes all clashes between a ligand and a protein that were previously mistaken because of some misidentification of covalent bonds.

6.4.3 Minor bug fixes

- Selection of the best receptor has been improved in *OEPosit*, ensuring that the correct receptor will always be picked in case of a self-docking.
- The internal algorithm has been optimized to improve time requirements for *OEPosit* when working with many receptors and a few ligands.

6.5 OEDocking TK 4.3.1.3

- An issue has been fixed in *OEHasClash* such that strong atom clashes are no longer mistakenly marked as covalent bonds causing failure to recognize those as actual clashes.

6.6 OEDocking TK 4.3.1.0

- An issue has been fixed that would occasionally cause segmentation faults when *OEPosit* would internally fail to generate conformers.
- New methods, *GetPoseRelaxOptions* and *SetPoseRelaxOptions*, have been added to *OEPositOptions* to allow more flexibility in relaxing the *OEPosit* generated poses.
- Internal algorithms have been improved in *OEShapeFit* to improve strain in generated poses.

6.7 OEDocking TK 4.3.0

6.7.1 New features

- Flexible **Posit** now uses `FF14SB-SAGE` force field for the protein/ligand complex optimization.
- A new overload of *SetupRef* has been added that allows using *OEShapeFit* with a bound ligand as the reference.
- The following new methods have been added to enable internal conformation generation in *OEShapeFitOptions*:
  - *SetFullConformationSearch*
  - *GetFullConformationSearch*
- A new class, *OEShapeFitOptions*, has been added that allows estimating pose prediction confidence against different reference systems. This new class is complementary to the existing function *OEGetEstimatedPositProbability*.

6.7.2 Major bug fixes

- An issue has been fixed where relaxation failed in *OEPosit* when the pose relaxation mode was set to either *ALL* or *CLASHED*.
- An issue causing generation of bent ring structures and high strain poses when performing *OEPosit* has been fixed.

- An issue has been fixed that caused flexible *OEPosit* with *ALL* relaxation mode to fail while generating non-clashing poses when *NONE* was successful.

6.7.3 Minor bug fixes

- An issue where *Dock* passed too many conformers for flexible optimization with relaxation in *OEPosit* has been fixed. Now, *OEShapeFit* no longer passes too many conformers for flexible optimization that could occasionally cause a less accurate conformer to become the top predicted pose.
- The *Dock* method with relaxation has been modified to be more aligned with other modes of failure when ligand charge assignments fail.

6.8 OEDocking TK 4.2.1

6.8.1 New features

- A new preliminary API class, *OEShapeFitOptions*, has been added to allow better control over option parameters for *OEShapeFit*. The constructor for the preliminary class *OEShapeFit* now takes *OEShapeFitOptions* as an argument instead of OEFlexiOverlayOptions. Also, *SetFlexiOverlayOptions* and *GetFlexiOverlayOptions* now accept and return *OEShapeFitOptions*.

6.8.2 Major bug fixes

- The *Dock* method in *OEPosit* now returns poses instead of a failure, if there is a failure during flexible optimization.
- An issue that caused intermittent crash for specific calculations on specific platforms when working with *OEPosit* has been fixed.
- *OEPosit* now accurately returns *NoValidNonClashPoses* as status when the predicted pose contains clashes based on the *defined clash allowed type*. Note that the clashed poses can still be retrieved from the resulting *OESinglePoseResult* if desired.

6.8.3 Minor bug fixes

- The *RankDesignUnits* method is now properly wrapped in python.

6.8.4 Documentation changes

- Documentation has been added for *the function OEGetEstimatedPositProbability*, which works with OEDesignUnit.
- Documentation has been added for *the function OESetupBox*, which supports OEMolBase and OEMCMolBase.

6.9 OEDocking TK 4.2.0

6.9.1 New features

- Two new preliminary API classes, *OEShapeFit* and *OEShapeFitResults* have been added which provides functionality for pose generation based on flexible fitting with Shape/Color and Forcefield.
- The adiabatic optimization algorithm of `ShapeFit` method for pose generation in *OEPosit* has been replaced by a new simultaneous flexible fitting algorithm, as implemented in *OEShapeFit*. The new algorithm uses Sage as the default forcefield, instead of MMFF.
- Two new methods, *SetFlexiOverlayOptions* and *GetFlexiOverlayOptions*, have been added to *OEPositOptions* to expose parameters related to *OEShapeFit* in *OEPosit*.

6.9.2 Major bug fixes

- An issue that caused a segmentation fault when using *CacheScoringSetup* has been fixed.

6.9.3 Documentation changes

- New C++ and Python examples have been added to the *Flexible Overlay Optimization with OEShapeFit API* section that demonstrate how to flexibly fit input multi-conformer molecules into a design unit that contains a bound ligand.

6.10 OEDocking TK 4.1.2

Spring 2022

- Minor internal improvements have been made.

6.11 OEDocking TK 4.1.1

Fall 2021

6.11.1 New features

- *OEPosit* can now generate poses using all the provided receptors. This can be controlled via a new method `SetBestReceptorPoseOnly`. The default behavior remains posing with the best receptor only.
- A new preliminary API class, *OEROCSQueryModelBuilder*, has been added that builds an OEShapeQuery from structure-based data, and can be used as a query to perform an OEOverlay or **ROCS** calculation.
- A new preliminary API class, *OEROCSQueryModelOptions*, has been added that provides choices for *OEROCSQueryModelBuilder* calculations.

6.11.2 Minor bug fixes

- An issue with redundant SD data after docking a multi conformer molecule using the `OEDock.DockMultiConformerMolecule` method has been fixed.

6.11.3 Documentation changes

- Two new examples have been added to the section, *Using Multiple Receptors* that handle multiple input receptors and generate single or multiple poses for each ligand.
- Example `PoseMolsMultiReceptor` has removed.

6.12 OEDocking TK 4.1.0

July 2021

6.12.1 New features

- A predicate string can now be supplied to *OEMakeReceptorOptions* using *SetTargetPred*. This povides a way to specify a select set of water molecules in an OEDesignUnit to be part of the receptor when the docking grid is constructed and used in subsequent docking calculations.
- *OEMakeReceptor* now autogenerates protein constraints. The constraints are left disabled by default for the docking calculations. These can be enabled using the toolkits, or the Make Receptor GUI application. The autogeneration of the constraints can be turned off using the *SetAutoConstraints* method in *OEMakeReceptorOptions*.

6.12.2 Minor bug fixes

- The default value for the *SetBoxMol* has been set to `0.0` in the *OEMakeReceptorOptions* class.

6.13 OEDocking TK 4.0.0.2

March 2021

- Default value of *`Target Mask`* has been changed to `OEDesignUnitComponents.TargetComplexNoSolvent`, to reflect best practices for receptor generation.
- *`Target Mask`* in *OEMakeReceptorOptions* now provides added flexibility to create user defined mask.
- An issue causing *OEPosit* to occasionally ignore the *`Posit Methods`* has been fixed.
- Constraints (both protein constraints and custom constraints) are now properly serialized in the OEReceptor in a .OEDU file.
- All of the protein constraints, custom constraints, and extra molecules are now properly transferred in the OEReceptor and the OEDesignUnit, when OEB format receptors are used.

6.14 OEDocking TK 4.0.0

Fall 2020

6.14.1 New features

- A new class, *OEMakeReceptorOptions*, has been added that provides choices for making an OEReceptor inside an OEDesignUnit.
- The *`OEMakeReceptor`* function now takes an OEDesignUnit and an *OEMakeReceptorOptions*, and adds an OEReceptor inside the OEDesignUnit object.
- A new class, *OEDockOptions*, has been introduced that provides choices for *`OEDockMethod`* and *`OESearchResolution`* in the *OEDock* class.
- The *OEDock* class constructor now has an overload that takes an *OEDockOptions* argument.
- The following methods now have an overload that accepts as an argument an OEDesignUnit containing an OEReceptor:
  - *`OEDock.Initialize`*
  - *`OEDock.CacheScoringSetup`*
  - *`OEScore.Initialize`*
  - *`OEScore.CacheScoringSetup`*
  - *`OEPosit.AddReceptor`*
- A new method, *`OESinglePoseResult.GetDesignUnit`*, has been added added that returns the resulting design unit containing the pose ligand and the target.
- A new class, *OEPoseOptimizerOptions*, has been added that provides choices for force fields and flexibility in performing pose optimization with *OEPoseOptimizer*.
- The *OEPoseOptimizer* class constructor now has an overload that takes an *OEPoseOptimizerOptions*.
- The *`OEPoseOptimizer.Optimize`* method now takes an OEDesignUnit to optimize.
- Flexible **POSIT** with *`OEPosit.Dock`* now uses the OEFF14SBParsley for pose optimization.

- A new method, *OEPosit.RankDesignUnits*, has been added that ranks an OEDesignUnit containing an OEReceptor for posing with multiple receptors.
- With the introduction of the new OEReceptor class and the corresponding behavior of the **OEDocking TK**, several previously existing APIs have been deprecated. The following table shows the deprecated APIs and their replacements:

Deprecated API	New API
OEDocking::OEBoxBase	OEBox
OEDocking::OEBox	
OEDocking::OECustomConstraints	OEReceptorCustomConstraints
OEDocking::OEFeature	OEReceptorConstraintFeature
OEDocking::OEProteinConstraint	OEReceptorProteinConstraint
OEDocking::OESphereBase	OESphere
OEDocking::OEAlignReceptors	OEStructuralSuperposition
OEDocking::OEIsReceptor	OEDesignUnit.HasReceptor
OEDocking::OEReceptorClear	OEReceptor.Clear
OEDocking::OEReceptorClearCustomConstrain	OEReceptor.ClearCustomConstraints
OEDocking::OEReceptorClearProteinConstrai	OEReceptor.ClearProteinConstraints
OEDocking::OEReceptorGetBoundLigand	OEDesignUnit.GetLigand
OEDocking::OEReceptorGetCustomConstraints	GetCustomConstraints
OEDocking::OEReceptorGetInnerContourLevel	OEReceptor.GetInnerContourLevel
OEDocking::OEReceptorGetOuterContourLevel	OEReceptor.GetOuterContourLevel
OEDocking::OEReceptorHasBoundLigand	OEDesignUnit.HasLigand
OEDocking::OEReceptorHasCustomConstraints	OEReceptor.HasCustomConstraints
OEDocking::OEReceptorHasInnerContourLevel	OEReceptor.HasInnerContourLevel
OEDocking::OEReceptorHasNegativeImageGrid	OEReceptor.HasNegativeImageGrid
OEDocking::OEReceptorHasOuterContourLevel	OEReceptor.HasOuterContourLevel
OEDocking::OEReceptorHasProteinConstraint	OEReceptor.HasProteinConstraints
OEDocking::OEReceptorSetCustomConstraints	OEReceptor.SetCustomConstraints
OEDocking::OEReceptorSetInnerContourLevel	OEReceptor.SetInnerContourLevel
OEDocking::OEReceptorSetOuterContourLevel	OEReceptor.SetOuterContourLevel
OEDocking::OEReceptorAddCachedScore	*OEDock.CacheScoringSetup*
OEDocking::OEReceptorSetCachedScore	*OEScore.CacheScoringSetup*
OEDocking::OEReadReceptorFile	OEReadDesignUnit
OEDocking::OEReadReceptorToBytes	
OEDocking::OEWriteReceptorFile	OEWriteDesignUnit
OEDocking::OEWriteReceptorToBytes	
OEDocking::OEBoxArea	OEBoxArea
OEDocking::OEBoxExtend	OEBoxExtend
OEDocking::OEBoxTranslate	OEBoxTranslate
OEDocking::OEBoxVolume	OEBoxVolume
OEDocking::OEBoxXDim	OEBoxXDim
OEDocking::OEBoxYDim	OEBoxYDim
OEDocking::OEBoxZDim	OEBoxZDim
OEDocking::OEBoxXMid	OEBoxXMid
OEDocking::OEBoxYMid	OEBoxYMid
OEDocking::OEBoxZMid	OEBoxZMid
OEDocking::OEInBox	OEInBox
OEDocking::OESetupBox	OESetupBox
OEDocking::OESetupBoxCenterAndExtents	
OEDocking::OEInSphere	OEInSphere
OEDocking::OESphereArea	OESphereArea

continues on next page

Table 1 – continued from previous page

OEDocking::OESphereVolume	OESphereVolume

- With the introduction of the new APIs, several obsolete functionalities have been deprecated without any suitable replacements. These functionalities were added in the early development days of **OEDocking TK** and are no longer deemed useful:
  - OEDocking::OEAlignReceptorFunction
  - OEDocking::OEReceptorAddExtraMols
  - OEDocking::OEReceptorClearBoundLigand
  - OEDocking::OEReceptorClearExtraMols
  - OEDocking::OEReceptorHasExtraMols

6.14.2 Documentation changes

- The *OEHybrid* class documentation has been modified to clarify that it is derived from *OEDock*.
- Documentation for OEDocking::OEDockBase and OEDocking::OEScoreBase has been removed.
- All the **OEDocking TK** examples have been reorganized to reflect the modified API behavior.

6.15 OEDocking TK 3.5.0.5

- The OEZ file format now correctly persists receptor information.

6.16 OEDocking TK 3.5.0

As part of the integration of **OEToolkits** and **OEApplications** into a single release, the version number of **OEDocking TK** has been upgraded to 3.5.0 to make it consistent with that of the **OEDocking** applications suite.

6.16.1 New features

- Two new methods, *OEPositOptions.SetExcludeSelf* and *OEPositOptions.GetExcludeSelf*, have been added to simplify cross-docking validation experiments with *OEPosit*.
- A new method, *OEPosit.RankReceptors*, has been added that can rank receptors when posing with multiple receptors.
- The method *OEMakeReceptor* has been overloaded to take an OEDesignUnit object as input.

6.16.2 Major bug fixes

- Memory management for *OEPosit.Dock* has been improved to handle up to 100 receptors when posing with multiple receptors.

6.16.3 Minor bug fixes

- The following methods now return bool instead of a void:
  - *OEPositOptions.SetAllowedClashType*
  - *OEPositOptions.SetFullConformationSearch*
  - *OEPositOptions.SetIgnoreNitrogenStereo*

  - *OEPositOptions.SetMinProbability*
  - *OEPositOptions.SetPoserCacheCount*
  - *OEPositOptions.SetPoseRelaxMode*
- *OEPosit.Dock* now fails gracefully when a single atom ligand is passed for posing.
- *OEMakeBoxMolecule* now sets the output molecule as 3D.

6.16.4 Documentation changes

- The *PoseMolsMultiReceptor* example has been updated.
- An example has been added to the Receptors section showing how to convert an OEDesignUnit object to a receptor.

6.17 OEDocking TK 1.4.1

6.17.1 New features

- Two new result classes, *OESinglePoseResult* and *OEPositResults*, have been added that better report *OEPosit*-generated pose results.
- A new class, *OEPoseOptimizer*, has been added that optimizes an already generated pose to reduce clashes.
- A new method, *Dock*, has been introduced that provides greater flexibility in reporting the generated pose results. The `OEDocking::OEPosit::DockMultiConformerMolecule` method is now deprecated.
- Two new methods, *AddReceptor* and *ClearReceptors*, have been added to the *OEPosit* class that enable using multiple receptors in a single calculation.
- A new function, *OEHasClash*, has been added that allows clash detection.
- The following new methods have been added to the *OEPositOptions* class that enable working with multiple receptors as well as further refinement of generated poses:
  - *GetAllowedClashType*
  - *SetAllowedClashType*
  - *GetMinProbability*
  - *SetMinProbability*
  - *GetPoserCacheCount*
  - *SetPoserCacheCount*
  - *GetPoseRelaxMode*
  - *SetPoseRelaxMode*
- A new namespace, *OEAllowedClashType*, has been added that defines constants for various allowed clash types.
- A new namespace, *OEPoseRelaxMode*, has been added that defines if and when a relaxation should be performed.
- A new constant, *OEDockingReturnCode.NoValidNonClashPoses*, has been added that indicates that no non-clashing poses are found.
- A new function, *OEDockingReturnCodeGetName*, has been added that converts *OEDockingReturnCode* to a meaningful string.
- A new function, *OEPositMethodGetName*, has been added that returns the name of an *OEPositMethod* in a string format.

6.17.2 Major bug fixes

- A rare crash that occurred during docking calculations in *DockMultiConformerMolecule* has been fixed.
- *OEPosit* no longer fails to generate poses due to unspecified stereo in the input molecule.

6.17.3 Minor bug fixes

- *OEMakeReceptor* now returns an empty receptor molecule when it fails.
- *OEGetEstimatedPositProbability* no longer crashes when a failure occurs during conformer generation.

6.17.4 Documentation changes

- Two new examples have been added in the sections, *Using Multiple Receptors* and *Receptor Flexibility*, that demonstrate new POSIT features for working with multiple receptors and for further refining generated poses, respectively.
- Docking examples have been updated for better coding practices that check the return code status of docking and print out any errors.

6.18 OEDocking TK 1.4.0

6.18.1 Major bug fixes

- `OEDocking::OEPosit::DockMultiConformerMolecule` now correctly uses the *FRED* method when there is no bound ligand in the receptor.
- `OEDocking::OEPosit::DockMultiConformerMolecule` now correctly uses the *FRED* method when the similarity between the bound ligand and the posed ligand is very low.

6.18.2 Minor bug fixes

- `OEDocking::OEDock::DockMultiConformerMolecule` now returns a failure code when the best scoring pose for a molecule has an unreasonable score.

6.19 OEDocking TK 1.3.7

- Minor internal improvements have been made.

6.20 OEDocking TK 1.3.6

6.20.1 New features

- Language-specific functions have been added to export receptors to bytes and read receptors from bytes.

For Java, see *OEReadReceptorFromByteArray* and *OEWriteReceptorToByteArray*.

6.21 OEDocking TK 1.3.5

- Minor internal improvements have been made.

6.22 OEDocking TK 1.3.4

- Minor internal improvements have been made.

6.23 OEDocking TK 1.3.3

- Minor internal improvements have been made.

6.24 OEDocking TK 1.3.2

- Minor internal improvements have been made.

6.25 OEDocking TK 1.3.1

6.25.1 New features

- `OEAddDockingInteractions` function has been overloaded and can now take an OEInteractionHintContainer object. The previous *OEFragmentNetwork* version has been deprecated and will throw a warning upon calling.

6.25.2 C++-specific changes

- PoseMolsWithBestReceptor.cpp example has been rewritten to improve performance.

6.26 OEDocking TK 1.3.0

6.26.1 Minor bug fixes

- A bug that caused the docking toolkit to crash if *OEDock* or *OEScore* were given a receptor that contained an arginine with a negatively charged nitrogen has been fixed.

6.26.2 Python-specific changes

- PoseMolsWithBestReceptor.py now avoids redundantly calculating the similarity of the docking and crystallographic ligand multiple times, improving the speed of this script when using multiple receptors.
- *OEDock* and *OEScore* methods will now release the global interpreter lock while performing expensive computations.

6.27 OEDocking TK 1.2.7

6.27.1 Minor bug fixes

- `OEAddDockingInteractions` has been updated after implementation changes were made to the *OEBio::OEFragmentNetwork* class. An example has been added to **OEBio TK** that illustrates how to access the protein-ligand interactions perceived by the `OEAddDockingInteractions` function.

6.28 OEDocking TK 1.2.6

6.28.1 New features

- `OEFormat.OEB` now supports the reading and writing of receptor information on OEMCMolBase conformers. Previously, only OEMolBase receptors were supported. The receptor APIs should still be used only on the active

conformer of multi-conformer molecules; see `OEMCMolBase.GetActive`.

6.28.2 Minor bug fixes

- `OEFormat.OEB` will no longer have receptor data written to it twice. This caused the receptor OEBs to be approximately twice as large but was otherwise harmless.

6.29 OEDocking TK 1.2.5

6.29.1 New features

- A new *`OEAddDockingInteractions`* function works with Grapheme's new active-site rendering functionality to perceive all necessary interactions.

6.29.2 Minor bug fixes

- *OEPosit* will no longer leak memory.
- An issue with hydrogen bonding groups on the protein being ignored when the available volume for an interacting molecule near the hydrogen bond group was less than 100 cubic angstroms has been fixed. Previously, the scoring function was set up with a protein and box, rather than a receptor object.
- Internal refactoring has been added to improve stability.
- A bug in the hydrogen bonding geometry perception of primary planar amines with interaction directions perpendicular to the plane has been fixed. Under this bug, such an interacting group had a 50% chance of selecting the wrong perpendicular direction.

6.30 OEDocking TK 1.2.4

- Minor internal improvements.

6.31 OEDocking TK 1.2.3

6.31.1 New features

The following improvements have been made to *OEPosit*:

- Graph comparison is used to identify when default OEOmega sampling may not produce appropriate conformers. If this is detected, the ShapeFit methodology is used to further sample conformation space.
- The initial overlays for ShapeFit are now chosen with the best TanimotoCombo overlay to the bound ligand prior to optimization.
- The choice of fitting routine (ShapeFit, Hybrid, or FRED) now take into account both the TanimotoCombo overlay and the graph similarity of the molecule. The graph similarity helps during cases where OMEGA undersamples and by sampling more conformations, better overlays can be generated.
- When generating conformer ensembles, the maximum number of conformers is set to 100 times the number of rotatable bonds with a minimum number of 200 conformers. This allows better sampling of floppy conformations.
- The maximum MMFF94s induced strain is increased to 20 kCal for molecules with more than 12 rotatable bonds.
- When FRED or HYBRID can not fit a molecule into the receptor, ShapeFit is automatically chosen regardless of the expected probability.
- If the input conformations include 3D coordinates, they are retained even when generating conformations.

6.31.2 Major bug fixes

- `OEDocking::OEPosit::DockMultiConformerMolecule` would previously only generate one pose. Now up to `numPoses` poses will be generated.
- SDData is retained from the input molecule. In the case of an input conformer ensemble, the SDData is chosen from the conformer with the closest RMSD to the final optimized pose.

6.31.3 Minor bug fixes

- A GOOD pose has been set to a probability of >= 0.5 not > 0.5 to be the same as specified by the documentation.
- Removed unbounded stack allocations.
- Removed annotation of optimization steps when writing to `.oeb` format.

6.32 OEDocking TK 1.2.2

6.32.1 New features

- *`OEPositMethod`* constants added to select which *OEPosit* to use at runtime.
- OEDocking::OEPositMethod::SHAPEFIT_EXHAUSTIVE search method added for better support of matching substructures of already posed ligands in binding sites. This has been accomplished by incorporating the Subrocs algorithm for initial starting points.
- *`OEPositOptions.SetIgnoreNitrogenStereo`* added to allow *OEPosit* to ignore nitrogen stereo centers in conformer enumeration.
- *OEPosit* now annotates clashed molecules.

6.32.2 Major bug fixes

- *OEPosit* now retains SD data of the input molecule.
- Debugging information removed from *OEPosit* docked molecules.
- *`OEGetEstimatedPositProbability`* is more robust with molecules missing explicit hydrogens.

6.32.3 Minor bug fixes

- Fixed an issue where secondary sp3 hydrogen bonding groups had slightly skewed geometry. In practice this causes the relevant HB angles to be about 15 degrees off, and the net effect on the score was minimal.

6.32.4 Documentation Changes

- *OECustomConstraints* class was improperly referred to as `OECustomConstraint`, without the s, in the documentation.
- *`OEPositMethod.UNKNOWN`* capitalization in the documentation was wrong in previous versions.
- *OEPositOptions* class has been updated with new settings: - select the methods *OEPosit* will run - ignore nitrogen stereo centers when making internal conformations

6.33 OEDocking TK 1.2.1

6.33.1 New Features

- OEPosit - When full conformer search is turned off, OMEGA is automatically run when the docked ligand is not a 3D conformer.

6.33.2 Major bug fixes

- OEPosit - Fixed a crash when setting full conformer search to false and docking a ligand with missing hydrogens.

6.33.3 Minor bug fixes

- Uses of `OEThrow.Fatal` changed to `OEThrow.Error`.

6.34 OEDocking TK 1.2.0

6.34.1 New features

The POSIT posing algorithm has been integrated into the OEDocking toolkit. The algorithm uses known bound reference ligands to generate a probability that, if the query ligand actually binds, the generated pose is within 2.0 Ångströms of the experimentally derived structure. This probability also guides the selection of the appropriate receptors (if multiple are provided) for docking as well as determining which docking methodology to use: SHAPEFIT, HYBRID, or FRED. Specific details on each methodology can be found in OEDocking's pose prediction chapter.

6.35 OEDocking TK 1.1.4

6.35.1 New features

- Hydrogen bond protein constraints now have improved recognition of appropriate hydrogen bond geometry. This should result in few false positives (i.e., there should be few docked poses that make poor hydrogen bond interactions with the hydrogen bond constraint atom on the protein).

6.35.2 Minor bug fixes

- Fixed a minor bug where the OEDock object could use more memory than needed.
- Fixed a potential out of memory error when calling OEMakeReceptor on a very large protein.

6.36 OEDocking TK 1.1.3

6.36.1 Bug fixes

- Fixed a rare one time initialization race condition that could only occur when using the *Docking TK* OEInterface convenience functions.

6.37 OEDocking TK 1.1.2

6.37.1 Bug fixes

- Fixed bug where OEDocking::OEDockBase::DockMultiConformerMolecule incorrectly returned *OEDockingReturnCode.Success* in cases where no poses of the molecule that fit in the active site could be found. This method now correctly returns *OEDockingReturnCode.NoValidPoses* in this case.

6.37.2 Improvements

- Slightly increased speed of *Chemgauss4* in both docking and optimization.

6.38 OEDocking TK 1.1.1

6.38.1 Bug fixes

- Fixed bug where SD data on the input molecules was not copied to the docked output molecule.
- Fixed memory leak when scoring a molecule that does not fit within the receptor site.
- Fixed an invalid memory access bug that occurred in rare cases when docking molecules.

6.39 OEDocking TK 1.1.0

- The behavior of `OEDockMethod.Hybrid` from version 1.0 has been changed. `OEDockMethod.Hybrid` is now an alias for the most up to date docking method, currently `OEDockMethod.Hybrid2`. `OEDockMethod.Hybrid1` now provides the same functionality that `OEDockMethod.Hybrid` did in version 1.0 of *OEDocking TK*.

6.39.1 New Features

- *OEDocking TK* is now thread safe. For details on thread safety see the "Thread Safety" section of the OEChem toolkit.
- *Chemgauss4* scoring function, available both as a docking method (`OEDockMethod.Chemgauss4`) and scoring method (`OEScoreType.Chemgauss4`).
- Hybrid 2 docking method (`OEDockMethod.Hybrid2`), that uses *Chemgauss4* for the structure based portion of the docking.
- *OEHybrid* class for doing hybrid docking added for convenience. *OEHybrid* is identical to *OEDock*, except that the default method is `OEDockMethod.Hybrid` rather than `OEDockMethod.Chemgauss`.
- Scoring function setup information from *OEDock*, *OEHybrid* and *OEScore* objects can now be saved on receptor files, decreasing the setup time for future runs.
- Fixed a bug where OEB handlers were not setup correctly sometimes due to link ordering problems.

6.39.2 Bug fixes

- Fixed bug generating negative image grids, that caused the negative image to extend further from the protein and penetrate less deeply into pockets than was intended.
- Fixed a bug in the local optimization (done during docking or with a call to `OEScore.SystematicSolidBodyOptimize`) that caused the local rotations to be less evenly spaced than they could be. This bug also affected the optimization done by the *OEDock* class.

6.40 OEDocking TK 1.0.0

- Docking features
  - Exhaustive Search docking followed by pose optimization
  - Hybrid docking (uses the structure a known bound active to guide docking)
  - Docking constraints (e.g., require specific hydrogen bonding interactions)
- Scoring features

  - Score optimization (systematic solid body optimization)
  - Break down of score by atom and/or scoring function component
  - Score annotating (stores score breakdown on molecule for visualization in VIDA)
- Implementation of four scoring functions
  - Chemgauss3
  - Chemscore
  - PLP
  - Shapegauss

CHAPTER

SEVEN

COPYRIGHT AND TRADEMARKS

© 1997-2024 Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA. All rights reserved.

OpenEye, Lexichem, ROCS, Grapheme, and Orion are registered trademarks of Cadence Design Systems, Inc.

This material contains proprietary information of Cadence Design Systems, Inc. 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 Cadence Design Systems, Inc.

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 Cadence Design Systems' 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 Cadence.

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. Cadence Design Systems, Inc., 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, OS X, macOS, and Macintosh are registered trademarks of Apple Inc. UNIX is a registered trademark of the Open Group. Linux is a registered trademark of Linus Torvalds. Red Hat is a registered trademark of Red Hat, Inc. SUSE, SLED, and SLES are registered trademarks of Novell, Inc. Ubuntu is a registered trademark of Canonical Ltd.

SYBYL is a registered trademark of Certara, L.P. MDL, BIOVIA and ISIS are registered trademarks of Dassault Systemes and/or its affiliates. SMARTS and SMIRKS are registered trademarks of Daylight Chemical Information Systems, Inc. MacroModel is a registered trademark of Schrodinger, LLC.

Python is a registered trademark of the Python Software Foundation. Django is a registered trademark of the Django Software Foundation. Java is a registered trademark of Oracle and/or its affiliates.

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

CHAPTER

EIGHT

SAMPLE CODE

For all of the code examples and samples within documents published by OpenEye, Cadence Molecular Sciences, the following terms of use apply.

TERMS FOR USE OF SAMPLE CODE

The software below ("Sample Code") is provided to current licensees or subscribers of Cadence products or SaaS offerings (each a "Customer"). Customer is hereby permitted to use, copy, and modify the Sample Code, subject to these terms. Cadence claims no rights to Customer's modifications. Modification of Sample Code is at Customer's sole and exclusive risk. Sample Code may require Customer to have a then current license or subscription to the applicable Cadence offering.

THE SAMPLE CODE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. CADENCE DISCLAIMS ALL WARRANTIES, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. In no event shall Cadence be liable for any damages or liability in connection with the Sample Code or its use.

CHAPTER

NINE

CITATION

Please follow the guidelines in this document to cite use of our products in research publications and other documents.

For more information, please contact OpenEye Support.

9.1 Orion®

To cite use of Orion, please use the following:

```
Orion <version-number>. OpenEye, Cadence Molecular Sciences, Santa Fe, NM. http://www.
↪eyesopen.com.
```

For example:

```
Orion 2025.1. OpenEye, Cadence Molecular Sciences, Santa Fe, NM. http://www.eyesopen.com.
```

The version number for Orion appears in the product's release notes, for example, https://docs.eyesopen.com/orion-user-interface/releasenotes/index.html.

9.2 Orion Floes

To cite use of an Orion-based floe package, please use the following:

```
OpenEye <package-name> <version-number>. OpenEye, Cadence Molecular Sciences, Santa Fe,␣
↪NM. http://www.eyesopen.com.
```

For example:

```
OpenEye Large-Scale Floes 4.2.3. OpenEye, Cadence Molecular Sciences, Santa Fe, NM.␣
↪http://www.eyesopen.com.
```

The version number for a floe package is displayed on the first page of the package's release notes. For example: https://docs.eyesopen.com/floe/modules/large_scale_floes/docs/source/releasenotes_2025_1.html.

9.3 Toolkits and Applications

To cite OpenEye toolkits or applications used in your work, please use the following:

```
OpenEye Toolkits [or Applications] <version-number>. OpenEye, Cadence Molecular Sciences,
↪ Santa Fe, NM. http://www.eyesopen.com.
```

For example:

```
OpenEye Toolkits 2024.2.1. OpenEye, Cadence Molecular Sciences, Santa Fe, NM. http://www.
↪eyesopen.com.
```

The Applications and Toolkits are released as a package, multiple times per year. The first part of the version number designates the calendar year. The version number appears on the summary document for the package.

If you want to cite an individual application or toolkit, you can use the syntax:

```
<product name> <version-number>. OpenEye, Cadence Molecular Sciences, Santa Fe, NM.␣
↪http://www.eyesopen.com.
```

where <*product name*> is:

- Bioisostere TK
- OEChem TK
- OEDepict TK
- OEDocking TK
- OEFF TK
- Eon TK
- FastROCS TK
- Grapheme TK
- GraphSim TK
- Lexichem TK
- OEMedChem TK
- MolProp TK
- Omega TK
- Quacpac TK
- Shape TK
- Spicoli TK
- Spruce TK
- Szmap TK
- Szybki TK
- Zap TK
- AFITT Application
- BROOD Application
- OEDocking Application
- EON Application
- OMEGA Application
- PICTO Application
- pKa-Prospector Application
- QUACPAC Application

- ROCS Application
- SiteHopper Application
- SPRUCE Application
- SZMAP Application
- SZYBKI Application
- VIDA Application

You can use the version number of the individual Application or Toolkit. For example:

```
SZYBKI Application 2.8.0.1. OpenEye, Cadence Molecular Sciences, Santa Fe, NM. http://
↪www.eyesopen.com.

Szybki TK 2.8.0.1. OpenEye, Cadence Molecular Sciences, Santa Fe, NM. http://www.
↪eyesopen.com.
```

The first documentation pages give the product names and version numbers, as in https://docs.eyesopen.com/applications/szybki/index.html and https://docs.eyesopen.com/toolkits/python/szybkitk/index.html.

9.4 Publications for Bibliographies

Please also cite these publications if the corresponding product was used in your research.

9.4.1 Orion

Sørensen, J., Bannan, C.C., Calabrò, G., Jain, V., Ovanesyan, G., Smith, A., Zhang, S., Bayly, C.I., Darden, T.A., Geballe, M.T., LeBard, D.N., McGann, M., Moon, J.B., Muddana, H.S., Shewmaker, A., LaFon, J., Tolbert, R.W., Skillman, A.G., Nicholls, A. Orion® A Cloud-Native Molecular Design Platform. In *Computational Drug Discovery*. V. Poongavanam and V. Ramaswamy, Eds. Wiley-VCH, 2024; pp. 579–615. DOI: 10.1002/9783527840748.ch24

9.4.2 AFITT and FLYNN Applications

Wlodek, S.; Skillman, A.G.; Nicholls, A. Automated Ligand Placement and Refinement with a Combined Force Field and Shape Potential. *Acta Crystallogr., Sect. D: Struct. Biol.* **2006**, *62* (7), 741–749. DOI: 10.1107/s0907444906016076

9.4.3 OEDocking Application and Toolkit

Kelley, B.P.; Brown, S.P.; Warren, G.L.; Muchmore, S.W. POSIT: Flexible Shape-Guided Docking For Pose Prediction. *J. Chem. Inf. Model.* **2015**, *55* (8), 1771-1780. DOI: 10.1021/acs.jcim.5b00142

McGann, M. FRED Pose Prediction and Virtual Screening Accuracy. *J. Chem. Inf. Model.* **2011**, *51* (3), 578-596. DOI: 10.1021/ci100436p

McGann, M. FRED and HYBRID Docking Performance on Standardized Datasets. *J. Comput.-Aided Mol. Des.* **2012**, *26* (8), 897-906. DOI: 10.1007/s10822-012-9584-8

9.4.4 OMEGA Application and Toolkit

If you are using OMEGA in torsion driving mode, please cite the following two publications.

Hawkins, P.C.D.; Skillman, A.G.; Warren, G.L.; Ellingson, B.A.; Stahl, M.T. Conformer Generation with OMEGA: Algorithm and Validation Using High Quality Structures from the Protein Databank and the Cambridge Structural Database. *J. Chem. Inf. Model.* **2010**, *50* (4), 572-584. DOI: 10.1021/ci100031x

Hawkins, P.C.D. and Nicholls, A. Conformer Generation with OMEGA: Learning from the Data Set and the Analysis of Failures. *J. Chem. Inf. Model.* **2012**, *52* (11), 2919–2936. DOI: /10.1021/ci300314k

If you are using OMEGA in macrocyle mode, please cite the following.

Hawkins, P.C.D. and Wlodek, S. Decisions with Confidence: Application to the Conformation Sampling of Molecules in the Solid State. *J. Chem. Inf. Model.* **2020**, *60* (7), 3518–3533. DOI: /10.1021/acs.jcim.0c00358

9.4.5 ROCS Application

Hawkins, P.C.D.; Skillman, A.G.; Nicholls, A. Comparison of Shape-Matching and Docking as Virtual Screening Tools. *J. Med. Chem.* **2007**, *50* (1), 74-82. DOI: 10.1021/jm0603365

9.5 OpenEye MMDS Web Service

To cite use of the Macromolecular Data Service (MMDS) web service, please use the syntax:

```
Macromolecular Data Service <version-number>. OpenEye, Cadence Molecular Sciences, Santa␣
↪Fe, NM. http://www.eyesopen.com.
```

For example:

```
Macromolecular Data Service 1.1. OpenEye, Cadence Molecular Sciences, Santa Fe, NM.␣
↪http://www.eyesopen.com.
```

The MMDS version number appears on the web service's release notes.

CHAPTER

TEN

TECHNOLOGY LICENSING

OpenEye products use the following technology under license.

Some of the open source technologies we use are licensed under the GNU Public License (GPL) version 2 or 3. In all instances, these technologies are communicated with using the "at arms length" principle, using pipes and command line arguments.

While the software is in some instances assembled into a container, that assembly does not invoke the GPL copyleft scope. For each Floe package where these are used, the environment log clearly details the installed programs and their version numbers. With the links below for each technology, the source for each of those programs is available.

Table 1: Licensed Technology

Name of Project	Website	License URL
_libgcc_mutex	https://github.com/conda-forge/_libgcc_mutex-feedstock	https://github.com/conda-forge/_libgcc_mutex-feedstock/blob/master/LICENSE.txt
_openmp_mutex	https://github.com/conda-forge/_openmp_mutex-feedstock	https://github.com/conda-forge/_openmp_mutex-feedstock/blob/main/LICENSE.txt
ablang2	https://github.com/TobiasHeOl/AbLang2	https://github.com/TobiasHeOl/AbLang2/blob/main/LICENSE
abseil-cpp	https://github.com/abseil/abseil-cpp	https://github.com/abseil/abseil-cpp/blob/master/LICENSE
absl-py	https://github.com/abseil/abseil-py	https://github.com/abseil/abseil-py/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
aiohappyeyeballs	https://github.com/aio-libs/aiohappyeyeballs	https://github.com/aio-libs/aiohappyeyeballs/blob/main/LICENSE
aiohttp	https://docs.aiohttp.org/en/stable/	https://github.com/aio-libs/aiohttp/blob/master/LICENSE.txt
aiosignal	https://github.com/aio-libs/aiosignal	https://github.com/aio-libs/aiosignal/blob/master/LICENSE
Amazon Linux 2	https://aws.amazon.com/amazon-linux-2	N/A
AmberUtils	http://ambermd.org	N/A
ambit	https://github.com/khimaros/ambit	https://github.com/khimaros/ambit/blob/master/LICENSE
amqp	https://github.com/celery/py-amqp	https://github.com/celery/py-amqp/blob/master/LICENSE
anaconda-anon-usage	https://github.com/anaconda/anaconda-anon-usage	https://github.com/anaconda/anaconda-anon-usage/blob/main/LICENSE
angular	https://github.com/angular/angular.js	https://github.com/angular/angular.js/raw/master/LICENSE.md
angular-animate	https://github.com/angular/angular.js	https://github.com/angular/angular.js/raw/master/LICENSE.md
angular-cache	https://github.com/jmdobry/angular-cache	https://github.com/jmdobry/angular-cache/raw/master/LICENSE
angular-cookies	https://github.com/angular/angular.js	https://github.com/angular/angular.js/raw/master/LICENSE.md

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
angular-loggly-logger	https://github.com/ajbrown/angular-loggly-logger	https://github.com/ajbrown/angular-loggly-logger/raw/master/LICENSE.md
angular-mocks	https://github.com/angular/angular.js	https://raw.githubusercontent.com/angular/angular.js/master/LICENSE
angular-resource	https://github.com/angular/angular.js	https://raw.githubusercontent.com/angular/angular.js/master/LICENSE
angular-toggle-switch	https://github.com/cgarvis/angular-toggle-switch	https://github.com/cgarvis/angular-toggle-switch/blob/master/LICENSE
angular-ui-grid	https://github.com/angular-ui/ui-grid	https://github.com/angular-ui/ui-grid
angular-ui-router	https://github.com/angular-ui/ui-router	https://github.com/angular-ui/ui-router/raw/master/LICENSE
angular-ui-tree	https://github.com/angular-ui-tree/angular-ui-tree	https://github.com/angular-ui-tree/angular-ui-tree/raw/master/LICENSE
angular-vs-repeat	https://github.com/kamilkp/angular-vs-repeat	https://raw.githubusercontent.com/kamilkp/angular-vs-repeat/master/LICENSE
aniso8601	https://bitbucket.org/nielsenb/aniso8601	https://bitbucket.org/nielsenb/aniso8601/src/master/LICENSE
annotated-types	https://github.com/annotated-types/annotated-types	https://github.com/annotated-types/annotated-types/blob/main/LICENSE
antlr4-python3-runtime	http://www.antlr.org OR https://pypi.org/project/antlr4-python3-runtin	https://github.com/conda-forge/antlr4-python3-runtin blob/master/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
anyio	https://github.com/agronholm/anyio	https://github.com/agronholm/anyio/blob/master/LICENSE
appdirs	http://github.com/ActiveState/appdirs	http://github.com/ActiveState/appdirs/blob/master/LICENSE.txt
appengine	https://google.golang.org/appengine	https://github.com/golang/appengine/blob/master/LICENSE
arabic-reshaper	https://github.com/mpcabd/python-arabic-reshape	https://github.com/mpcabd/python-arabic-reshape/blob/master/LICENSE
archspec	https://github.com/archspec/archspec/blob/master/README.md	https://github.com/archspec/archspec/blob/master/COPYRIGHT
argon2-cffi	https://github.com/hynek/argon2-cffi	https://github.com/hynek/argon2-cffi/blob/master/LICENSE
argon2-cffi-bindings	https://github.com/hynek/argon2-cffi-bindings	https://github.com/hynek/argon2-cffi-bindings/blob/master/LICENSE
arpack	https://www.caam.rice.edu/software/ARPACK/	https://www.caam.rice.edu/software/ARPACK/
arrow	https://github.com/arrow-py/arrow	https://github.com/arrow-py/arrow?tab=Apache-2.0-1-ov-file#readme
ascii85	https://github.com/huandu/node-ascii85	https://github.com/huandu/node-ascii85/raw/master/LICENSE
ase	https://wiki.fysik.dtu.dk/ase/	https://wiki.fysik.dtu.dk/ase/development/licenseinfo.html#license-info
asgiref	https://github.com/django/asgiref/	https://github.com/django/asgiref/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
asn1crypto	https://github.com/wbond/asn1crypto	https://github.com/wbond/asn1crypto/blob/master/LICENSE
assertions go-render	https://github.com/smartystreets/assertions/internal/go-render/render	https://github.com/smartystreets/assertions/blob/master/internal/go-render/LICENSE
assertions oglmatchers	https://github.com/smartystreets/assertions/internal/oglematchers	https://github.com/smartystreets/assertions/blob/master/internal/oglematchers/LICENSE
assertions	https://github.com/smartystreets/assertions	https://github.com/smartystreets/assertions/blob/master/LICENSE.md
asttokens	https://github.com/gristlabs/asttokens	https://github.com/gristlabs/asttokens/blob/master/LICENSE
astunparse	https://github.com/simonpercivall/astunparse	https://github.com/simonpercivall/astunparse/blob/master/LICENSE
async-lru	https://github.com/aio-libs/async-lru	https://github.com/aio-libs/async-lru/blob/master/LICENSE
async-timeout	https://github.com/aio-libs/async-timeout	https://github.com/aio-libs/async-timeout/blob/master/LICENSE
atk-1.0	https://docs.gtk.org/atk/	https://gitlab.gnome.org/GNOME/atk/-/blob/master/COPYING
atomic	https://github.com/uber-go/atomic	https://github.com/uber-go/atomic/blob/master/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
atomicwrites	https://github.com/untitaker/python-atomicwrites	https://github.com/untitaker/python-atomicwrites/blob/master/LICENSE
attrs	https://www.attrs.org/	https://github.com/python-attrs/attrs/blob/main/LICENSE
aws-sdk-go	https://github.com/aws/aws-sdk-go	https://github.com/aws/aws-sdk-go/blob/master/LICENSE.txt
babel	https://babel.pocoo.org/ OR https://github.com/python-babel/babel	https://github.com/python-babel/babel/blob/master/LICENSE OR https://babel.pocoo.org/en/latest/license.html#babel-license
Babel	https://github.com/python-babel/babel	https://github.com/python-babel/babel/blob/master/LICENSE
backcall	https://github.com/takluyver/backcall	https://github.com/takluyver/backcall/blob/master/LICENSE
backports	https://github.com/brandon-rhodes/backports	https://github.com/brandon-rhodes/backports/blob/master/LICENSE.TXT
backports.functools-lru-cache	https://github.com/jaraco/backports.functools_lru_cache	https://github.com/jaraco/backports.functools_lru_cache/blob/main/LICENSE
backports.zstd	https://github.com/rogdham/backports.zstd	https://github.com/rogdham/backports.zstd?tab=License-1-ov-file#readme
base62	https://github.com/kare/base62	https://github.com/kare/base62/blob/main/LICENSE
beautifulsoup4	https://www.crummy.com/software/BeautifulSoup/	https://www.crummy.com/software/BeautifulSoup/

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
billiard	https://github.com/celery/billiard	https://github.com/celery/billiard/blob/master/LICENSE.txt
biopython	https://biopython.org	https://github.com/biopython/biopython/blob/master/LICENSE.rst
biotite	https://github.com/biotite-dev/biotite/blob/master/README.rst	https://github.com/biotite-dev/biotite/blob/master/LICENSE.rst
bitarray	https://github.com/ilanschnell/bitarray	https://github.com/ilanschnell/bitarray/blob/master/LICENSE
bitset	https://github.com/willf/bitset	https://github.com/willf/bitset/blob/master/LICENSE
blas	https://www.netlib.org/blas/	https://www.netlib.org/blas/#_licensing
bleach	https://github.com/mozilla/bleach	https://github.com/mozilla/bleach/blob/main/LICENSE
blessings	https://github.com/erikrose/blessings	https://github.com/erikrose/blessings/blob/master/LICENSE
blinker	https://pythonhosted.org/blinker/	https://github.com/pallets-eco/blinker/blob/master/LICENSE
bokeh	https://github.com/bokeh/bokeh	https://github.com/bokeh/bokeh/blob/branch-4.0/LICENSE.txt
blosc	https://github.com/Blosc/python-blosc	https://github.com/Blosc/python-blosc/blob/master/LICENSE.txt
blosc2	https://github.com/Blosc/python-blosc2	https://github.com/Blosc/c-blosc2/blob/main/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
boltons	https://github.com/mahmoud/boltons	https://github.com/mahmoud/boltons/blob/master/LICENSE
Boltz	https://boltz.bio	https://github.com/jwohlwend/boltz/blob/main/LICENSE
boost	https://www.boost.org/doc/libs/1_71_0/libs/python/doc/html/index.html	https://www.boost.org/LICENSE_1_0.txt
boost-cpp	https://www.boost.org/doc/libs/1_71_0/libs/python/doc/html/index.html	https://www.boost.org/LICENSE_1_0.txt
bootstrap-vue	https://github.com/bootstrap-vue/bootstrap-vue	https://github.com/bootstrap-vue/bootstrap-vue/raw/master/LICENSE
boto3	https://github.com/boto/boto3	https://github.com/boto/boto3/blob/develop/LICENSE
botocore	https://github.com/boto/botocore	https://github.com/boto/botocore/blob/develop/LICENSE.txt
Bottleneck	https://bottleneck.readthedocs.io/en/latest/index.html	https://bottleneck.readthedocs.io/en/latest/license.html
Brotli	https://github.com/google/brotli	https://github.com/google/brotli/blob/master/LICENSE
brotli-bin	https://github.com/google/brotli	https://github.com/google/brotli/blob/master/LICENSE
brotli-python	http://python-hyper.org/projects/brotlipy/en/latest/	https://github.com/python-hyper/brotlicffi/blob/main/LICENSE
brotlicffi	https://github.com/python-hyper/brotlicffi	https://github.com/python-hyper/brotlicffi?tab=MIT-1-ov-file#readme
brotlipy	https://github.com/python-hyper/brotlicffi	https://github.com/python-hyper/brotlicffi/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
bson	http://github.com/py-bson/bson	https://github.com/py-bson/bson/blob/master/LICENSE_APACHE
bytefmt	https://code.cloudfoundry.org/bytefmt	https://github.com/cloudfoundry/bytefmt/blob/master/LICENSE
bzip2	https://www.bzip.org	https://spdx.org/licenses/bzip2-1.0.6.html
c-ares	https://github.com/c-ares/c-ares	https://github.com/c-ares/c-ares/blob/main/LICENSE.md
ca-certificates	https://github.com/conda-forge/ca-certificates-feedsto	https://wiki.mozilla.org/CA/Included_Certificates
cached-property	https://github.com/pydanny/cached-property	https://github.com/pydanny/cached-property/blob/master/LICENSE
cachetools	https://github.com/tkem/cachetools/	https://github.com/tkem/cachetools/blob/master/LICENSE
cairo	https://pycairo.readthedocs.io/en/latest/	https://github.com/pygobject/pycairo/blob/master/COPYING
canvas	https://github.com/Automattic/node-canvas	https://github.com/Automattic/node-canvas
cctbx	https://github.com/cctbx/cctbx_project	https://github.com/cctbx/cctbx_project/blob/master/cctbx/LICENSE_2_0.txt
celery	https://github.com/celery/celery	https://github.com/celery/celery/blob/master/LICENSE
Cerberus	https://docs.python-cerberus.org/en/stable/	https://docs.python-cerberus.org/en/stable/license.html
certifi	https://certifiio.readthedocs.io/en/latest/	https://wiki.mozilla.org/index.php?title=CA/IncludedCertificates&redirect=no

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
cffi	https://github.com/python-cffi	https://github.com/cffi/cffi/blob/master/COPYRIGHT
cftime	https://pypi.org/project/cftime/	https://github.com/Unidata/cftime/blob/master/LICENSE
chardet	https://github.com/chardet/chardet	https://github.com/chardet/chardet/blob/master/LICENSE
charset-normalizer	https://github.com/ousret/charset_normalizer	https://github.com/Ousret/charset_normalizer/blob/master/LICENSE
ChEMBL Structure Pipeline	https://github.com/chembl/ChEMBL_Structure_Pipeline/tree/master	https://github.com/chembl/ChEMBL_Structure_Pipeline/blob/master/LICENSE
choreographer	https://github.com/mazpie/choreographer	https://github.com/mazpie/choreographer/blob/main/LICENSE
chunkreader	https://github.com/jackc/chunkreader/v2	https://github.com/jackc/chunkreader/blob/master/LICENSE
click	https://palletsprojects.com/p/click/	https://github.com/pallets/click/blob/main/LICENSE.rst
click-completion	https://github.com/click-contrib/click-completion	https://github.com/click-contrib/click-completion/blob/master/LICENSE
click-didyoumean	https://github.com/click-contrib/click-didyoumean	https://github.com/click-contrib/click-didyoumean/blob/master/LICENSE
click-plugins	https://github.com/click-contrib/click-plugins	https://github.com/click-contrib/click-plugins/blob/master/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
click-repl	https://github.com/untitaker/click-repl	https://github.com/click-contrib/click-repl/blob/master/LICENSE
cloudpickle	https://github.com/cloudpipe/cloudpickle	https://github.com/cloudpipe/cloudpickle/blob/master/LICENSE
cmake	https://cmake.org/	https://cmake.org/licensing/
colorama	https://github.com/tartley/colorama	https://github.com/tartley/colorama/blob/master/LICENSE.txt
comm	https://github.com/ipython/comm	https://github.com/ipython/comm/blob/master/LICENSE
compose	https://github.com/docker/compose	https://github.com/docker/compose/blob/v2/LICENSE
conda-content-trust	https://github.com/conda/conda-content-trust	https://github.com/conda/conda-content-trust/blob/main/LICENSE
conda-libmamba-solver	https://github.com/conda/conda-libmamba-solv	https://github.com/conda/conda-libmamba-solv blob/main/LICENSE
conda-pack	https://conda.github.io/conda-pack/	https://github.com/conda/conda-pack/blob/main/LICENSE.txt
conda-package-handling	https://github.com/conda/conda-package-handli	https://github.com/conda/conda-package-handli blob/main/LICENSE
conda-package-streaming	https://anaconda.org/conda-forge/conda-package-strean	https://github.com/conda/conda-package-strean blob/main/LICENSE
conda-token	https://anaconda.org/anaconda/conda-token	https://anaconda.org/anaconda/conda-token

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
conda_package_strear	https://github.com/conda/conda-package-streamtree/main/conda_package_streaming	https://github.com/conda/conda-package-streamtab=License-1-ov-file
confuse	https://github.com/beetbox/confuse	https://github.com/beetbox/confuse/blob/main/LICENSE
contourpy	https://github.com/contourpy/contourpy	https://github.com/contourpy/contourpy/blob/main/LICENSE
cpp-expected	https://github.com/TartanLlama/expected	https://github.com/TartanLlama/expected?tab=CC0-1.0-1-ov-file#readme
cpp-peglib	https://github.com/yhirose/cpp-peglib	https://github.com/yhirose/cpp-peglib/blob/master/LICENSE
cryptography	https://github.com/pyca/cryptography	https://github.com/pyca/cryptography/blob/main/LICENSE
cssselect2	https://github.com/Kozea/cssselect2/	https://github.com/Kozea/cssselect2/blob/master/LICENSE
cuda-bindings	https://nvidia.github.io/cuda-python/cuda-bindings/latest/install.html	https://github.com/NVIDIA/cuda-python/blob/main/cuda_bindings/LICENSE
cuda-pathfinder	https://nvidia.github.io/cuda-python/cuda-pathfinder/latest/	https://nvidia.github.io/cuda-python/cuda-pathfinder/latest/license.html
cudatoolkit	https://developer.nvidia.com/cuda-toolkit	https://docs.nvidia.com/cuda/eula/index.html
cuequivariance-ops-cu12	https://developer.nvidia.com/cuequivariance	https://github.com/NVIDIA/cuEquivariance/blob/main/LICENSES/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
cuequivariance-ops-torch-cu12	https://developer.nvidia.com/cuequivariance	https://github.com/NVIDIA/cuEquivariance/blob/main/LICENSES/LICENSE
cuequivariance-torch	https://docs.nvidia.com/cuda/cuequivariance/	https://github.com/NVIDIA/cuEquivariance/blob/main/LICENSES/LICENSE
cupy-cuda113	https://cupy.dev/	https://github.com/cupy/cupy/blob/main/LICENSE
cupy-cuda11x	https://cupy.dev/	https://github.com/cupy/cupy/?tab=MIT-1-ov-file#readme
curl	https://curl.se	https://curl.se/docs/copyright.html
cycler	https://github.com/matplotlib/cycler	https://github.com/matplotlib/cycler/blob/main/LICENSE
cyrus-sasl	https://github.com/cyrusimap/cyrus-sasl	https://github.com/cyrusimap/cyrus-sasl/blob/master/COPYING
Cython	https://cython.org	https://www.apache.org/licenses/LICENSE-2.0.html
d3	https://github.com/mbostock/d3	https://github.com/mbostock/d3/raw/master/LICENSE
dataclasses	https://github.com/ericvsmith/dataclasses	https://github.com/ericvsmith/dataclasses/blob/master/LICENSE.txt
ddsketch	http://github.com/datadog/sketches-py	https://github.com/DataDog/sketches-py/blob/master/LICENSE
debugpy	https://aka.ms/debugpy	https://github.com/microsoft/debugpy/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
decimal	https://github.com/shopspring/decimal	https://github.com/shopspring/decimal/blob/master/LICENSE
decorator	https://github.com/micheles/decorator	https://github.com/micheles/decorator/blob/master/LICENSE.txt
deepdiff	https://github.com/seperman/deepdiff	https://github.com/seperman/deepdiff/blob/master/LICENSE
deeptime	https://github.com/deeptime-ml/deeptime	https://github.com/deeptime-ml/deeptime/blob/main/LICENSE.txt
defusedxml	https://github.com/tiran/defusedxml	https://github.com/tiran/defusedxml/blob/main/LICENSE
dftd3	https://github.com/dftd3/tad-dftd3	https://github.com/dftd3/tad-dftd3/blob/main/LICENSE
dill	https://github.com/uqfoundation/dill	https://github.com/uqfoundation/dill/blob/master/LICENSE
distlib	https://distlib.readthedocs.io/en/latest/	https://github.com/pypa/distlib/blob/master/LICENSE.txt
distro	https://github.com/python-distro/distro	https://github.com/python-distro/distro/blob/master/LICENSE
Django	https://www.djangoproject.com/	https://github.com/django/django/blob/main/LICENSE
django-classy-tags	https://github.com/django-cms/django-classy-tags	https://github.com/django-cms/django-classy-tags/blob/master/LICENSE
django-cors-headers	https://github.com/adamchainz/django-cors-headers	https://github.com/adamchainz/django-cors-headers/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
django-csp	https://github.com/mozilla/django-csp	https://github.com/mozilla/django-csp/blob/main/LICENSE
django-extensions	https://github.com/django-extensions/django-extensions	https://github.com/django-extensions/django-extensions/blob/main/LICENSE
django-filter	https://github.com/carltongibson/django-filter/tree/master	https://github.com/carltongibson/django-filter/blob/main/LICENSE
django-redis	https://github.com/jazzband/django-redis	https://github.com/jazzband/django-redis/blob/master/LICENSE
django-taggit	https://github.com/jazzband/django-taggit	https://github.com/jazzband/django-taggit/blob/master/LICENSE
django-taggit-serializer	https://github.com/glemmaPaul/django-taggit-serializ	https://github.com/glemmaPaul/django-taggit-serializ blob/master/LICENSE
django-taggit-templatetags2	https://github.com/fizista/django-taggit-templat	https://github.com/fizista/django-taggit-templat blob/master/LICENSE.txt
djangorestframework	https://www.django-rest-framewor org/	https://github.com/encode/django-rest-framewor blob/master/LICENSE.md
dkh	https://psicode.org/psi4manual/master/dkh.html	https://github.com/psi4/dkh/blob/master/COPYING.LESSER
dm-tree	https://github.com/deepmind/tree	https://github.com/deepmind/tree/blob/master/LICENSE
docker-py	https://github.com/docker/docker-py/	https://github.com/docker/docker-py/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
docker-pycreds	https://github.com/shin-/dockerpy-creds	https://github.com/shin-/dockerpy-creds/blob/master/LICENSE
docopt	https://docopt.org	https://github.com/docopt/docopt/blob/master/LICENSE-MIT
docutils	https://docutils.sourceforge.io	https://docutils.sourceforge.io/COPYING.html
drf-dynamic-fields	https://github.com/dbrgn/drf-dynamic-fields	https://github.com/dbrgn/drf-dynamic-fields/blob/master/LICENSE
DuckDB	https://duckdb.org/	https://github.com/duckdb/duckdb?tab=MIT-1-ov-file#readme
edgembar	https://rutgerslbsr.gitlab.io/fe-toolkit/edgembar/index.html	https://gitlab.com/RutgersLBSR/fe-toolkit/-/blob/master/LICENSE
editdistance	https://github.com/roy-ht/editdistance	https://github.com/roy-ht/editdistance/blob/master/LICENSE
eigen	https://eigen.tuxfamily.org/index.php?title=Main_Page	https://eigen.tuxfamily.org/index.php?title=Main_Page#License
einops	https://github.com/arogozhnikov/einops	https://github.com/arogozhnikov/einops/blob/master/LICENSE
einx	https://github.com/fferflo/einx	https://github.com/fferflo/einx/blob/master/LICENSE
entrypoints	https://github.com/takluyver/entrypoints	https://github.com/takluyver/entrypoints/blob/master/LICENSE
errors	https://github.com/pkg/errors	https://github.com/pkg/errors/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
eslint-plugin	https://github.com/typescript-eslint/typescript-eslint	https://github.com/typescript-eslint/typescript-eslint/raw/master/LICENSE
et_xmlfile	https://foss.heptapod.net/openpyxl/et_xmlfile	https://foss.heptapod.net/openpyxl/et_xmlfile/-/blob/branch/default/LICENCE.rst
exceptiongroup	https://github.com/agronholm/exceptiongroup	https://github.com/agronholm/exceptiongroup/blob/main/LICENSE
execnet	https://execnet.readthedocs.io/en/latest/	https://github.com/pytest-dev/execnet/blob/master/LICENSE
executing	https://github.com/alexmojaki/executing	https://github.com/alexmojaki/executing/blob/master/LICENSE.txt
expat	https://libexpat.github.io	https://github.com/libexpat/libexpat/blob/master/expat/COPYING
fairscale	https://github.com/facebookresearch/fairscale	https://github.com/facebookresearch/fairscale/blob/main/LICENSE
fairseq	https://github.com/pytorch/fairseq	https://github.com/facebookresearch/fairseq/blob/main/LICENSE
fastjsonschema	https://github.com/horejsek/python-fastjsonschem	https://github.com/horejsek/python-fastjsonschem/blob/master/LICENSE
fastrlock	https://github.com/scoder/fastrlock	https://github.com/scoder/fastrlock/blob/master/LICENSE
fftw	https://www.fftw.org	commercial license

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
filebuffer	https://github.com/mattetti/filebuffer	https://github.com/mattetti/filebuffer/blob/master/LICENSE
filelock	https://py-filelock.readthedocs.io/en/latest/index.html	https://github.com/tox-dev/py-filelock/blob/main/LICENSE
finufft	https://github.com/flatironinstitute/finufft	https://github.com/flatironinstitute/finufft/blob/master/LICENSE
Flask	https://flask.palletsprojects.com/en/2.1.x/	https://flask.palletsprojects.com/en/2.1.x/license/
flatbuffers	https://google.github.io/flatbuffers/	https://github.com/google/flatbuffers/blob/master/LICENSE.txt
flexcache	https://github.com/hgrecco/flexcache	https://github.com/hgrecco/flexcache/blob/main/LICENSE
flexparser	https://github.com/hgrecco/flexparser	https://github.com/hgrecco/flexparser/blob/main/LICENSE
flit-core	https://github.com/pypa/flit	https://github.com/scientific-python/lazy_loader/blob/main/LICENSE.md
FLTK	https://www.fltk.org/index.php	https://www.fltk.org/COPYING.php
fmt	https://fmt.dev/latest/index.html	https://github.com/fmtlib/fmt/blob/master/LICENSE.rst
font-awesome	https://github.com/FortAwesome/Font-Awesome	https://github.com/FortAwesome/Font-Awesome
font-ttf-dejavu-sans-mono	https://dejavu-fonts.github.io	https://dejavu-fonts.github.io/License.html
font-ttf-inconsolata	https://fonts.google.com/specimen/Inconsolata	https://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=OFL

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
font-ttf-source-code-pro	https://fonts.google.com/specimen/Source+Code+Pro	https://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=OFL
font-ttf-ubuntu	https://fonts.google.com/specimen/Ubuntu	https://ubuntu.com/legal/font-licence
fontconfig	https://www.freedesktop.org/wiki/Software/fontconfig/	https://gitlab.freedesktop.org/fontconfig/fontconfig/-/blob/main/COPYING
fonts-conda-ecosystem	https://anaconda.org/conda-forge/fonts-conda-ecosyster	https://docs.conda.io/en/latest/license.html
fonts-conda-forge	https://anaconda.org/conda-forge/fonts-conda-forge/	https://creativecommons.org/licenses/by/4.0/
fonttools	https://github.com/fonttools/fonttools	https://github.com/fonttools/fonttools/blob/main/LICENSE
forcebalance	https://simtk.org/home/forcebalance OR https://github.com/leeping/forcebalance	https://github.com/leeping/forcebalance/blob/master/LICENSE
fqdn	https://github.com/ypcrts/fqdn	https://github.com/ypcrts/fqdn?tab=MPL-2.0-1-ov-file#readme
freetype	https://freetype.org	https://freetype.org/license.html
freetype-py	https://github.com/rougier/freetype-py	https://github.com/rougier/freetype-py?tab=License-1-ov-file#readme
fribidi	https://github.com/fribidi/fribidi	https://github.com/fribidi/fribidi/blob/master/COPYING
frozendict	https://github.com/Marco-Sulla/python-frozendict	https://github.com/Marco-Sulla/python-frozendict/blob/master/LICENSE.txt
frozenlist	https://github.com/aio-libs/frozenlist	https://github.com/aio-libs/frozenlist/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
fsmlite	https://github.com/tkem/fsmlite	https://github.com/tkem/fsmlite/blob/master/LICENSE
fsspec	https://github.com/fsspec/filesystem_spec	https://github.com/fsspec/filesystem_spec/blob/master/LICENSE
funcy	https://github.com/Suor/funcy	https://github.com/Suor/funcy/blob/master/LICENSE
gast	https://github.com/serge-sans-paille/gast/	https://github.com/serge-sans-paille/gast/blob/master/LICENSE
gau2grid	https://github.com/dgasmith/gau2grid	https://github.com/dgasmith/gau2grid/blob/master/LICENSE
gax-go	https://github.com/googleapis/gax-go/v2	https://github.com/googleapis/gax-go/blob/main/LICENSE
gdk-pixbuf	https://gitlab.gnome.org/GNOME/gdk-pixbuf	https://gitlab.gnome.org/GNOME/gdk-pixbuf/-/blob/master/COPYING
gemmi	https://gemmi.readthedocs.io/en/latest/	https://github.com/project-gemmi/gemmi/blob/master/LICENSE.txt
genproto	https://google.golang.org/genproto/googleapis	https://github.com/googleapis/go-genproto/blob/main/LICENSE
geometric	https://openbase.com/python/geometric	https://openbase.com/python/geometric
giflib	https://giflib.sourceforge.net	https://github.com/mirrorer/giflib/blob/master/COPYING
gitdb	https://github.com/gitpython-developers/gitdb	https://github.com/gitpython-developers/gitdb/blob/master/LICENSE
GitPython	https://github.com/gitpython-developers/GitPython	https://github.com/gitpython-developers/GitPython/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
glib	https://docs.gtk.org/glib/	https://gitlab.gnome.org/GNOME/glib/-/blob/main/LICENSES/LGPL-2.1-or-later.txt
GLM Library	https://github.com/g-truc/glm	https://github.com/g-truc/glm/blob/master/manual.md#section0
gls	https://github.com/jtolds/gls	https://github.com/jtolds/gls/blob/master/LICENSE
gmpy2	https://github.com/aleaxit/gmpy	https://github.com/gmpy2/gmpy2/blob/master/COPYING.LESSER
go-cleanhttp	https://github.com/hashicorp/go-cleanhttp	https://github.com/hashicorp/go-cleanhttp/blob/master/LICENSE
go-connections	https://github.com/docker/go-connections	https://github.com/docker/go-connections/blob/master/LICENSE
go-cpio	https://github.com/cavaliercoder/go-cpio	https://github.com/cavaliercoder/go-cpio/blob/master/LICENSE
go-getter	https://github.com/hashicorp/go-getter	https://github.com/hashicorp/go-getter/blob/master/LICENSE
go-homedir	https://github.com/mitchellh/go-homedir	https://github.com/mitchellh/go-homedir/blob/master/LICENSE
go-ini	https://github.com/go-ini/ini	https://github.com/go-ini/ini/blob/master/LICENSE
go-jmespath	https://github.com/jmespath/go-jmespath	https://github.com/jmespath/go-jmespath/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
go-junit-report	https://github.com/jstemmer/go-junit-report	https://github.com/jstemmer/go-junit-report/blob/master/LICENSE
go-netrc	https://github.com/bgentry/go-netrc/netrc	https://github.com/bgentry/go-netrc/blob/master/LICENSE
go-ole	https://github.com/go-ole/go-ole	https://github.com/go-ole/go-ole/blob/master/LICENSE
go-pg	https://github.com/go-pg/pg	https://github.com/go-pg/pg/blob/master/LICENSE
go-redis	https://github.com/go-redis/redis/v8	https://github.com/go-redis/redis/blob/master/LICENSE
go-rendezvous	https://github.com/dgryski/go-rendezvous	https://github.com/dgryski/go-rendezvous/blob/master/LICENSE
go-safetemp	https://github.com/hashicorp/go-safetemp	https://github.com/hashicorp/go-safetemp/blob/master/LICENSE
go-sysconf	https://github.com/tklauser/go-sysconf	https://github.com/tklauser/go-sysconf/blob/master/LICENSE
go-testing-interface	https://github.com/mitchellh/go-testing-interface	https://github.com/mitchellh/go-testing-interface/blob/master/LICENSE
go-units	https://github.com/docker/go-units	https://github.com/docker/go-units/blob/master/LICENSE
go-version	https://github.com/hashicorp/go-version	https://github.com/hashicorp/go-version/blob/master/LICENSE
go-zglob	https://github.com/mattn/go-zglob	https://github.com/mattn/go-zglob/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
go.opencensus	https://go.opencensus.io	https://github.com/census-instrumentatio opencensus-go/blob/master/LICENSE
gobrake.v2	https://gopkg.in/airbrake/gobrake.v2	https://github.com/airbrake/gobrake/blob/v2.0.9/LICENSE
goconvey	https://github.com/smartystreets/goconvey	https://github.com/smartystreets/goconvey/blob/master/LICENSE.md
golden-layout	https://github.com/deepstreamIO/golden-layout	https://github.com/deepstreamIO/golden-layout/raw/master/LICENSE
google-auth	https://google-auth.readthedocs.io/en/master/	https://github.com/GoogleCloudPlatform google-auth-library-p blob/master/LICENSE
google-auth-oauthlib	https://github.com/googleapis/google-auth-library-p	https://github.com/googleapis/google-auth-library-p blob/main/LICENSE
google-cloud-go	https://cloud.google.com/go	https://github.com/googleapis/google-cloud-go/blob/main/LICENSE
google-cloud-go/storage	https://cloud.google.com/go/storage	https://github.com/googleapis/google-cloud-go/blob/main/LICENSE
google-pasta	https://github.com/google/pasta	https://github.com/google/pasta/blob/master/LICENSE
google.golang.org/api	https://google.golang.org/api	https://github.com/googleapis/google-api-go-client/blob/main/LICENSE
gopsutil	https://github.com/shirou/gopsutil	https://github.com/shirou/gopsutil/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
gorilla websocket	https://github.com/gorilla/websocket	https://github.com/gorilla/websocket/blob/master/LICENSE
gpytorch	https://gpytorch.ai	https://github.com/cornellius-gp/gpytorch?tab=MIT-1-ov-file#readme
graphite2	https://github.com/silnrsi/graphite	https://github.com/silnrsi/graphite/blob/master/LICENSE
graphviz	https://graphviz.org/	https://graphviz.org/license/
greenlet	https://greenlet.readthedocs.io/en/latest/	https://github.com/python-greenlet/greenlet/blob/master/LICENSE
groupcache	https://github.com/golang/groupcache	https://github.com/golang/groupcache/blob/master/LICENSE
grpc	https://google.golang.org/grpc	https://github.com/grpc/grpc-go/blob/master/LICENSE
grpc-cpp	https://github.com/grpc/grpc	https://github.com/grpc/grpc/blob/master/LICENSE
grpcio	https://github.com/grpc/grpc.io/blob/main/LICENSE	https://github.com/grpc/grpc.io/blob/main/LICENSE
gtk2	https://gitlab.gnome.org/GNOME/gtk	https://gitlab.gnome.org/GNOME/gtk/-/blob/main/COPYING
gts	https://sourceforge.net/projects/gts/	https://www.gnu.org/licenses/lgpl-3.0.html
h11	https://github.com/python-hyper/h11	https://github.com/python-hyper/h11/blob/master/LICENSE.txt
h2	https://github.com/python-hyper/h2	https://github.com/python-hyper/h2?tab=MIT-1-ov-file#readme

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
h5py	https://www.h5py.org	https://docs.h5py.org/en/stable/licenses.html
harfbuzz	https://github.com/harfbuzz/harfbuzz	https://github.com/harfbuzz/harfbuzz/blob/main/COPYING
hdbscan	https://hdbscan.readthedocs.io/en/latest/	https://github.com/scikit-learn-contrib/hdbscan/blob/master/LICENSE
hdf4	https://www.hdfgroup.org/solutions/hdf4/	https://github.com/HDFGroup/hdf4/blob/master/COPYING
hdf5	https://www.hdfgroup.org/solutions/hdf5/	https://github.com/HDFGroup/hdf5/blob/master/COPYING
he	https://github.com/mathiasbynens/he	https://github.com/mathiasbynens/he/raw/master/LICENSE-MIT.txt
hpack	https://github.com/python-hyper/hpack	https://github.com/python-hyper/hpack?tab=MIT-1-ov-file#readme
html-loader	https://github.com/webpack-contrib/html-loader	https://github.com/webpack-contrib/html-loader/raw/master/LICENSE
html5lib	https://github.com/html5lib/html5lib-python	https://github.com/html5lib/html5lib-python/blob/master/LICENSE
htslib	https://www.htslib.org	https://github.com/samtools/htslib/blob/develop/LICENSE
httpcore	https://www.encode.io/httpcore/	https://github.com/encode/httpcore/blob/master/LICENSE.md
httpx	https://github.com/encode/httpx	https://github.com/encode/httpx/blob/master/LICENSE.md

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
humanize	https://github.com/jmoiron/humanize	https://github.com/jmoiron/humanize/blob/master/LICENCE
hydra-core	https://github.com/facebookresearch/hydra	https://github.com/facebookresearch/hydra/blob/main/LICENSE
hyperframe	https://github.com/python-hyper/hyperframe	https://github.com/python-hyper/hyperframe?tab=MIT-1-ov-file#readme
icu	https://github.com/unicode-org/icu	https://github.com/unicode-org/icu/blob/main/icu4c/LICENSE
idna	https://github.com/kjd/idna	https://github.com/kjd/idna/blob/master/LICENSE.md
ihm	https://github.com/ihmwg/python-ihm	https://github.com/ihmwg/python-ihm/blob/main/LICENSE
imageio	https://github.com/imageio/imageio	https://github.com/imageio/imageio/blob/master/LICENSE
imagesize	https://github.com/shibukawa/imagesize_py	https://github.com/shibukawa/imagesize_py/blob/master/LICENSE.rst
ImmuneBuilder	https://github.com/oxpig/ImmuneBuilder/tree/main	https://github.com/oxpig/ImmuneBuilder/blob/main/LICENSE
importlib-metadata	https://github.com/python/importlib_metadata	https://github.com/python/importlib_metadata/blob/main/LICENSE
importlib_metadata	https://github.com/python/importlib_metadata	https://github.com/python/importlib_metadata/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
importlib_resources	https://github.com/python/importlib_resources	https://github.com/python/importlib_resources/blob/main/LICENSE
InChI	https://iupac.org/who-we-are/divisions/division-details/inchi/	https://iupac.org/wp-content/uploads/2021/06/iupac-inchi-license_2020.pdf
inflection	https://github.com/jinzhu/inflection	https://github.com/jinzhu/inflection/blob/master/LICENSE
ini.v1	https://gopkg.in/ini.v1	https://github.com/go-ini/ini/blob/v1.66.2/LICENSE
iniconfig	https://github.com/pytest-dev/iniconfig	https://github.com/pytest-dev/iniconfig/blob/main/LICENSE
innersvg-polyfill	https://github.com/dnozay/innersvg-polyfill	https://github.com/dnozay/innersvg-polyfill/raw/master/LICENSE
intel-openmp	https://github.com/hermitcore/libomp_oss	https://github.com/hermitcore/libomp_oss/blob/hermit/LICENSE.txt
ipykernel	https://ipython.org	https://ipython.org
ipython	https://ipython.org	https://github.com/ipython/ipython/blob/master/LICENSE
ipython-genutils	http://ipython.org	https://github.com/ipython/ipython_genutils/blob/master/COPYING.md
ipython_pygments_le	https://github.com/ipython/ipython-pygments-lex	https://github.com/ipython/ipython-pygments-lex blob/main/LICENSE
ipywidgets	http://jupyter.org	https://github.com/jupyter-widgets/ipywidgets/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
isodate	https://github.com/gweis/isodate/	https://github.com/gweis/isodate/blob/master/LICENSE
isoduration	https://github.com/bolsote/isoduration	https://github.com/bolsote/isoduration?tab=ISC-1-ov-file#readme
itsdangerous	https://palletsprojects.com/p/itsdangerous/	https://itsdangerous.palletsprojects.com/en/2.1.x/license/
jax	https://github.com/google/jax	https://github.com/google/jax/blob/main/LICENSE
jaxlib	https://github.com/google/jax	https://github.com/google/jax/blob/main/LICENSE
jaxtyping	https://github.com/patrick-kidger/jaxtyping	https://github.com/patrick-kidger/jaxtyping?tab=License-1-ov-file#readme
jedi	https://jedi.readthedocs.io/en/latest/	https://github.com/davidhalter/jedi/blob/master/LICENSE.txt
Jinja2	https://palletsprojects.com/p/jinja/	https://github.com/pallets/jinja/blob/main/LICENSE.rst
jmespath	https://github.com/jmespath/jmespath.py	https://github.com/jmespath/jmespath.py/blob/develop/LICENSE.txt
joblib	https://joblib.readthedocs.io/en/latest/	https://github.com/joblib/joblib/blob/master/LICENSE.txt
jpeg	https://www.ijg.org	https://www.ijg.org/files/README
json5	https://github.com/dpranke/pyjson5	https://github.com/dpranke/pyjson5/blob/main/LICENSE
jsonfield	https://github.com/dmkoch/django-jsonfield/	https://github.com/rpkilby/jsonfield/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
jsonpatch	https://github.com/stefankoegl/python-json-patch	https://github.com/stefankoegl/python-json-patch/blob/master/LICENSE
jsonpickle	https://github.com/jsonpickle/jsonpickle	https://github.com/jsonpickle/jsonpickle/blob/main/LICENSE
jsonpointer	https://github.com/stefankoegl/python-json-pointer	https://github.com/stefankoegl/python-json-pointer/blob/master/LICENSE.txt
jsonschema	https://github.com/python-jsonschema/jsonschema	https://github.com/python-jsonschema/jsonschema/blob/master/COPYING
jsonschema-specifications	https://github.com/python-jsonschema/jsonschema-specificat	https://github.com/python-jsonschema/jsonschema-specificat blob/main/COPYING
jstat	https://github.com/jstat/jstat	https://www.opensource.org/licenses/mit-license.php
jupyter-client	https://jupyter.org	https://github.com/jupyter/jupyter_client/blob/main/COPYING.md
jupyter-core	https://jupyter.org	https://github.com/jupyter/jupyter_core/blob/master/COPYING.md
jupyter-events	https://github.com/jupyter/jupyter_events	https://github.com/jupyter/jupyter_events/blob/main/LICENSE
jupyter-lsp	https://github.com/jupyter-lsp/jupyterlab-lsp	https://github.com/jupyter-lsp/jupyterlab-lsp
jupyter-serverhttp://jupyter.or	https://github.com/jupyter-server/jupyter_server/blob/main/LICENSE	
jupyter_client	http://jupyter.org	https://github.com/jupyter/jupyter_client/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
jupyter_core	http://jupyter.org	https://github.com/jupyter/jupyter_core/blob/master/LICENSE
jupyter_server	https://github.com/jupyter-server/jupyter_server	https://github.com/jupyter-server/jupyter_server/blob/main/LICENSE
jupyter_server_termin	https://github.com/jupyter-server/jupyter_server_terminals	https://github.com/jupyter-server/jupyter_server_terminals/blob/main/LICENSE
jupyterlab	https://github.com/jupyterlab/jupyterlab	https://github.com/jupyterlab/jupyterlab/blob/main/LICENSE
jupyterlab-pygments	http://jupyter.org	https://github.com/jupyterlab/jupyterlab_pygments/blob/main/LICENSE
jupyterlab-widgets	http://jupyter.org	https://github.com/jupyter-widgets/ipywidgets/blob/master/LICENSE
jupyter-lab_pygments	https://github.com/jupyterlab/jupyterlab_pygments	https://github.com/jupyterlab/jupyterlab_pygments?tab=BSD-3-Clause-1-ov-file#readme
jupyterlab_server	https://github.com/jupyterlab/jupyterlab_server	https://github.com/jupyterlab/jupyterlab_server/blob/main/LICENSE
jupyterlab_widgets	https://github.com/jupyter-widgets/ipywidgets	https://github.com/jupyter-widgets/ipywidgets?tab=BSD-3-Clause-1-ov-file#readme
kaleido	https://github.com/plotly/Kaleido	https://github.com/plotly/Kaleido/blob/master/LICENSE.txt
keras	https://github.com/keras-team/keras	https://github.com/keras-team/keras/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
Keras-Preprocessing	https://github.com/keras-team/keras-preprocessing	https://github.com/keras-team/keras-preprocessing/blob/master/LICENSE
keras-tuner	https://github.com/keras-team/keras-tuner	https://github.com/keras-team/keras-tuner/blob/master/LICENSE
keyring	https://github.com/jaraco/keyring	https://github.com/jaraco/keyring/blob/main/LICENSE
keyutils	https://github.com/sassoftware/python-keyutils	https://github.com/sassoftware/python-keyutils/blob/master/LICENSE
kiwisolver	https://kiwisolver.readthedocs.io/en/latest/	https://github.com/nucleic/kiwi/blob/main/LICENSE
kombu	https://kombu.readthedocs.io	https://github.com/celery/kombu/blob/master/LICENSE
krb5	https://web.mit.edu/kerberos/	https://github.com/krb5/krb5/blob/master/src/lib/gssapi/LICENSE
kt-legacy	https://github.com/haifeng-jin/kt-legacy	https://github.com/haifeng-jin/kt-legacy/blob/main/LICENSE
lark	https://github.com/lark-parser/lark	https://github.com/lark-parser/lark?tab=MIT-1-ov-file#readme
lazy_loader	https://github.com/scientific-python/lazy_loader	https://github.com/scientific-python/lazy_loader/blob/main/LICENSE.md
lcms2	https://www.littlecms.com	https://github.com/sk1project/python-lcms2/blob/master/LICENSE
legacy-cgi	https://github.com/jackrosenthal/legacy-cgi	https://github.com/jackrosenthal/legacy-cgi?tab=License-1-ov-file#readme

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
lerc	https://github.com/Esri/lerc	https://github.com/Esri/lerc/blob/master/LICENSE
libarchive	http://www.libarchive.org	https://raw.githubusercontent.com/libarchive/libarchive/master/COPYING
libblas	https://www.netlib.org/blas/	https://www.netlib.org/blas/#_licensing
libbrotlicommon	https://github.com/google/brotli	https://github.com/google/brotli/blob/master/LICENSE
libbrotlidec	https://github.com/google/brotli	https://github.com/google/brotli/blob/master/LICENSE
libbrotlienc	https://github.com/google/brotli	https://github.com/google/brotli/blob/master/LICENSE
libcblas	https://anaconda.org/conda-forge/libcblas	N/A
libclang	https://github.com/sighingnow/libclang	https://github.com/sighingnow/libclang/blob/master/LICENSE.TXT
libcurl	https://curl.se/libcurl/	https://curl.se/docs/copyright.html
libcxx	https://github.com/llvm-mirror/libcxx	https://github.com/llvm-mirror/libcxx/blob/master/LICENSE.TXT
libdb	https://www.oracle.com/technology/software/products/berkeley-db/index.html	https://github.com/berkeleydb/libdb/blob/master/LEGAL
libdeflate	https://github.com/ebiggers/libdeflate	https://github.com/ebiggers/libdeflate/blob/master/COPYING
libedit	https://thrysoee.dk/editline/	http://www.netbsd.org/Goals/redistribution.html
libev	https://software.schmorp.de/pkg/libev.html	https://software.schmorp.de/pkg/libev.html

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
libexpat	https://libexpat.github.io/	https://github.com/libexpat/libexpat?tab=MIT-1-ov-file#readme
libffi	https://github.com/libffi/libffi	https://github.com/libffi/libffi/blob/master/LICENSE
libgcc	https://gcc.gnu.org/onlinedocs/gccint/Libgcc.html	https://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html
libgcrypt	https://gnupg.org/software/index.html	https://git.gnupg.org/cgi-bin/gitweb.cgi?p=libgcrypt.git;a=blob;f=COPYING.LIB;hb=HEAD
libgd	https://libgd.github.io	https://libgd.github.io/manuals/2.2.1/files/license-txt.html
libglib	https://github.com/PolMine/libglib	https://github.com/PolMine/libglib/blob/master/include/glib-2.0/glib.h
libiconv	https://www.gnu.org/software/libiconv/	https://www.gnu.org/software/libiconv/#downloading
libint	https://tinyurl.com/yvw97wbw	https://tinyurl.com/yvw97wbw
liblapack	http://www.netlib.org/lapack/	https://www.netlib.org/lapack/LICENSE.txt
liblapacke	https://anaconda.org/conda-forge/liblapacke	https://www.netlib.org/lapack/LICENSE.txt
liblzma	https://tukaani.org/xz/	https://git.tukaani.org/?p=xz.git;a=blob;f=COPYING.0BSD
libmamba	https://medium.com/@QuantStack/open-software-packag	https://github.com/mamba-org/mamba/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
libmambapy	https://www.anaconda.com/blog/a-faster-conda-for-a-g	https://github.com/conda/conda-libmamba-solv/blob/main/LICENSE
libnetcdf	https://www.unidata.ucar.edu/software/netcdf/	https://www.unidata.ucar.edu/software/netcdf/copyright.html
libnghttp2	https://github.com/nghttp2/nghttp2	https://github.com/nghttp2/nghttp2/blob/master/COPYING
libnsl	https://github.com/thkukuk/libnsl	https://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html
libopenblas	https://www.openblas.net/	https://github.com/xianyi/OpenBLAS/blob/develop/LICENSE
libpng	https://www.libpng.org/pub/png/libpng.html	https://www.libpng.org/pub/png/src/libpng-LICENSE.txt
libpq	https://www.postgresql.org/	https://www.postgresql.org/about/licence/
librsvg	https://gitlab.gnome.org/GNOME/librsvg	https://gitlab.gnome.org/GNOME/librsvg/-/blob/main/COPYING.LIB
libsolv	https://github.com/openSUSE/libsolv	https://github.com/openSUSE/libsolv/blob/master/LICENSE.BSD
libsqlite	https://github.com/mackyle/sqlite	https://github.com/mackyle/sqlite?tab=License-1-ov-file#readme
libssh2	https://github.com/libssh2/libssh2	https://github.com/libssh2/libssh2/blob/master/COPYING
libstdcxx	https://gcc.gnu.org/	https://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
libtiff	https://www.libtiff.org/	https://www.libtiff.org/misc.html
libtrust	https://github.com/docker/libtrust	https://github.com/docker/libtrust/blob/master/LICENSE
libuuid	https://sourceforge.net/projects/libuuid/	https://sourceforge.net/p/libuuid/code/ci/master/tree/COPYING
libuv	https://github.com/libuv/libuv	https://github.com/libuv/libuv/blob/v1.x/LICENSE
libwebp	https://chromium.googlesource.com/?format=HTML	https://chromium.googlesource.com/chromium/src/third_party/+/refs/heads/main/libwebp/LICENSE
libwebp-base	https://chromium.googlesource.com/?format=HTML	https://chromium.googlesource.com/chromium/src/third_party/+/refs/heads/main/libwebp/LICENSE
libxc	https://www.tddft.org/programs/libxc/	https://gitlab.com/libxc/libxc/-/blob/master/COPYING
libxcb	https://xcb.freedesktop.org	https://github.com/freedesktop/xcb-libxcb/blob/master/COPYING
libxcrypt	https://github.com/besser82/libxcrypt	https://github.com/besser82/libxcrypt?tab=LGPL-2.1-1-ov-file#readme
libxml2	https://git.gnome.org/browse/libxml2/	https://gitlab.gnome.org/GNOME/libxml2/-/blob/master/Copyright
libxmlsec1	https://github.com/lsh123/xmlsec	https://github.com/lsh123/xmlsec/blob/master/README
libxslt	https://gitlab.gnome.org/GNOME/libxslt/-/wikis/home	https://gitlab.gnome.org/GNOME/libxslt/-/blob/master/Copyright

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
libzlib	https://zlib.net	https://zlib.net/zlib_license.html
lightning-utilities	https://github.com/Lightning-AI/utilities	https://github.com/Lightning-AI/utilities?tab=Apache-2.0-1-ov-file#readme
lime	https://github.com/marcotcr/lime	https://github.com/marcotcr/lime/blob/master/LICENSE
linear-operator	https://github.com/cornellius-gp/linear_operator	https://github.com/cornellius-gp/linear_operator?tab=MIT-1-ov-file#readme
lit	http://llvm.org	https://releases.llvm.org/14.0.0/LICENSE.TXT
llvm-openmp	https://github.com/llvm-mirror/openmp	https://github.com/llvm-mirror/openmp/blob/master/LICENSE.txt
llvmlite	http://llvmlite.readthedocs.io	https://github.com/numba/llvmlite/blob/main/LICENSE
loader-utils	https://github.com/webpack/loader-utils	https://github.com/webpack/loader-utils/raw/master/LICENSE
logistro	https://github.com/geopozo/logistro	https://github.com/geopozo/logistro?tab=MIT-1-ov-file#readme
logomaker	https://logomaker.readthedocs.io/en/latest/	https://github.com/jbkinney/logomaker/blob/master/LICENSE
logrus	https://github.com/sirupsen/logrus	https://github.com/sirupsen/logrus/blob/master/LICENSE
logrus-airbrake-hook.v2	https://gopkg.in/gemnasium/logrus-airbrake-hook.v2	https://github.com/gemnasium/logrus-airbrake-hook/blob/v2.1.2/LICENSE
lxml	https://lxml.de	https://lxml.de/index.html#license

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
lz4-c	https://www.lz4.org/	https://github.com/lz4/lz4/blob/dev/LICENSE
lzo	https://www.oberhumer.com/opensource/lzo/	https://www.oberhumer.com/opensource/gpl.html
mamba	https://github.com/mamba-org/mamba	https://github.com/mamba-org/mamba?tab=BSD-3-Clause-1-ov-f readme
markdown	https://github.com/evilstreak/markdown-js	https://www.opensource.org/licenses/mit-license.php
markdown-it-py	https://github.com/executablebooks/markdown-it-py	https://github.com/executablebooks/markdown-it-py/blob/master/LICENSE
markupsafe	https://github.com/pallets/markupsafe/tree/main/src/markupsafe	https://github.com/pallets/markupsafe/tree/main?tab=BSD-3-Clause-1-ov-f
MarkupSafe	https://palletsprojects.com/p/markupsafe/	https://github.com/pallets/markupsafe/blob/main/LICENSE.rst
mashumaro	https://github.com/Fatal1ty/mashumaro	https://github.com/Fatal1ty/mashumaro
matplotlib	https://matplotlib.org	https://matplotlib.org/stable/users/project/license.html
matplotlib-base	https://matplotlib.org	https://matplotlib.org/stable/users/project/license.html
matplotlib-inline	https://github.com/ipython/matplotlib-inline	https://github.com/ipython/matplotlib-inline/blob/master/LICENSE
mda-xdrlib	https://github.com/MDAnalysis/mda-xdrlib	https://github.com/MDAnalysis/mda-xdrlib/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
mda_xdrlib	https://github.com/MDAnalysis/mda-xdrlib	https://github.com/MDAnalysis/mda-xdrlib?tab=License-1-ov-file#readme
mdtraj	https://www.mdtraj.org/	https://www.mdtraj.org/1.9.8.dev0/index.html#license
mdurl	https://github.com/markdown-it/mdurl	https://github.com/markdown-it/mdurl/blob/master/LICENSE
menuinst	https://github.com/conda/menuinst	https://github.com/conda/menuinst/blob/main/LICENSE.txt
mergo	https://github.com/imdario/mergo	https://github.com/imdario/mergo/blob/master/LICENSE
mistune	https://github.com/lepture/mistune	https://github.com/lepture/mistune/blob/master/LICENSE
mkl	https://github.com/rust-math/intel-mkl-src	https://github.com/rust-math/intel-mkl-src/blob/master/License.txt
mkl-fft	https://github.com/IntelPython/mkl_fft	https://github.com/IntelPython/mkl_fft/blob/master/LICENSE.txt
mkl-random	https://github.com/IntelPython/mkl_random	https://github.com/IntelPython/mkl_random/blob/master/LICENSE.txt
mkl-service	https://github.com/IntelPython/mkl-service	https://github.com/IntelPython/mkl-service/blob/master/LICENSE.txt
mkl_fft	http://github.com/IntelPython/mkl_fft	https://github.com/IntelPython/mkl_fft?tab=BSD-3-Clause-1-ov-f readme

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
mkl_random	http://github.com/IntelPython/mkl_random	https://github.com/IntelPython/mkl_random?tab=BSD-3-Clause-1-ov-f readme
ml-dtypes	https://github.com/jax-ml/ml_dtypes	https://github.com/jax-ml/ml_dtypes/blob/main/LICENSE
ml_dtypes	https://github.com/jax-ml/ml_dtypes	https://github.com/jax-ml/ml_dtypes/blob/main/LICENSE
modelcif	https://github.com/ihmwg/python-modelcif	https://github.com/ihmwg/python-modelcif/blob/main/LICENSE
molecupy	https://molecupy.readthedocs.io/en/latest/	https://github.com/samirelanduk/atomium/blob/master/LICENSE
moment	https://github.com/moment/moment	https://github.com/moment/moment/raw/master/LICENSE
moment-precise-range-plugin	https://github.com/codebox/moment-precise-range	https://github.com/codebox/moment-precise-range raw/master/LICENSE.md
more-itertools	https://github.com/more-itertools/more-itertools	https://github.com/more-itertools/more-itertools/blob/master/LICENSE
mpiplus	https://github.com/choderalab/mpiplus	https://github.com/choderalab/mpiplus/blob/main/LICENSE
mpmath	http://mpmath.org/	https://github.com/mpmath/mpmath/blob/master/LICENSE
mrcfile	https://github.com/ccpem/mrcfile	https://github.com/ccpem/mrcfile/blob/master/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
msgpack	https://msgpack.org/	https://github.com/msgpack/msgpack-python/blob/main/COPYING
multidict	https://github.com/aio-libs/multidict	https://github.com/aio-libs/multidict/blob/master/LICENSE
multierr	https://go.uber.org/multierr	https://github.com/uber-go/multierr/blob/master/LICENSE.txt
multiprocess	https://github.com/uqfoundation/multiprocess	https://github.com/uqfoundation/multiprocess/blob/master/LICENSE
munkres	https://software.clapper.org/munkres/	https://github.com/bmc/munkres/blob/master/LICENSE.md
myesui uuid	https://github.com/myesui/uuid	https://pkg.go.dev/github.com/myesui/uuid?tab=licenses
namex	https://github.com/bcgov/namex	https://github.com/bcgov/namex/blob/main/LICENSE
narwhals	https://github.com/narwhals-dev/narwhals	https://github.com/narwhals-dev/narwhals?tab=MIT-1-ov-file#readme
nbclassic	http://jupyter.org	https://github.com/jupyter/nbclassic/blob/main/LICENSE
nbclient	https://jupyter.org	https://github.com/jupyter/nbclient/blob/main/LICENSE
nbconvert	https://jupyter.org	https://github.com/jupyter/nbconvert/blob/main/LICENSE
nbformat	http://jupyter.org	https://github.com/jupyter/nbformat/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
ncurses	https://invisible-island.net/ncurses/	https://invisible-island.net/ncurses/ncurses-license.html
ndfes	https://rutgerslbsr.gitlab.io/fe-toolkit/	https://gitlab.com/RutgersLBSR/fe-toolkit/-/blob/master/LICENSE
ndindex	https://quansight-labs.github.io/ndindex/	https://github.com/Quansight-Labs/ndindex?tab=MIT-1-ov-file#readme
nest-asyncio	https://github.com/erdewit/nest_asyncio	https://github.com/erdewit/nest_asyncio/blob/master/LICENSE
nest_asyncio	https://github.com/erdewit/nest_asyncio	https://github.com/erdewit/nest_asyncio?tab=BSD-2-Clause-1-ov-f readme
netcdf-fortran	https://www.unidata.ucar.edu/software/netcdf/	https://www.unidata.ucar.edu/software/netcdf/copyright.html
netCDF4	http://github.com/Unidata/netcdf4-python	https://github.com/Unidata/netcdf4-python/blob/master/LICENSE
nettle	https://git.lysator.liu.se/nettle/nettle	https://git.lysator.liu.se/nettle/nettle/-/blob/master/AUTHORS?ref_type=heads
networkx	https://networkx.org	https://github.com/networkx/networkx/blob/main/LICENSE.txt
nfpm	https://github.com/goreleaser/nfpm	https://github.com/goreleaser/nfpm/blob/master/LICENSE.md
ng-tags-input	https://github.com/mbenford/ngTagsInput	https://github.com/mbenford/ngTagsInput
ng-toast	https://github.com/tameraydin/ngToast	https://github.com/tameraydin/ngToast

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
ngdraggable	https://github.com/fatlinesofcode/ngDraggable	https://github.com/fatlinesofcode/ngDraggable/raw/master/LICENSE.txt
ngVue	https://github.com/ngVue/ngVue	https://github.com/ngVue/ngVue/raw/master/LICENSE
nh3	https://nh3.readthedocs.io/en/latest/	https://github.com/messense/nh3/blob/main/LICENSE
nlohmann_json	https://github.com/nlohmann/json	https://github.com/nlohmann/json?tab=MIT-1-ov-file#readme
nlopt	https://nlopt.readthedocs.io/en/latest/NLopt_Python_Reference/	https://nlopt.readthedocs.io/en/latest/NLopt_License_and_Copyright/
nodejs	https://nodejs.org/en/	https://raw.githubusercontent.com/nodejs/node/master/LICENSE
nomkl	https://github.com/conda-forge/nomkl-feedstock	https://github.com/conda-forge/nomkl-feedstock/blob/main/LICENSE.txt
notebook	http://jupyter.org	https://github.com/jupyter/notebook/blob/main/LICENSE
notebook-shim	https://github.com/jupyter/notebook_shim	https://github.com/jupyter/notebook_shim
notebook_shim	http://jupyter.org	https://github.com/jupyter/notebook_shim/blob/main/LICENSE
numba	https://numba.pydata.org	https://github.com/numba/numba/blob/main/LICENSE
numcpus	https://github.com/tklauser/numcpus	https://github.com/tklauser/numcpus/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
numexpr	https://github.com/pydata/numexpr	https://github.com/pydata/numexpr/blob/master/LICENSE.txt
numpy	https://numpy.org	https://github.com/numpy/numpy/blob/main/LICENSE.txt
numpy-base	https://numpy.org	https://github.com/numpy/numpy/blob/main/LICENSE.txt
numpydoc	https://numpydoc.readthedocs.io/en/latest/index.html	https://github.com/numpy/numpydoc/blob/main/LICENSE.txt
nvidia-cublas-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cublas-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cuda-cupti-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cuda-cupti-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cuda-nvrtc-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cuda-nvrtc-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cuda-runtime-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cuda-runtime-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cudnn-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/deeplearning/cudnn/sla/index.html
nvidia-cudnn-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/deeplearning/cudnn/sla/index.html

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
nvidia-cuEquivariance	https://github.com/NVIDIA/cuEquivariance	https://github.com/NVIDIA/cuEquivariance/blob/main/LICENSES/LICENSE
nvidia-cufft-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cufft-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cufile-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html#
nvidia-curand-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-curand-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cusolver-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cusolver-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cusparse-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cusparse-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-cusparselt-cu12	https://docs.nvidia.com/cuda/cusparselt/index.html	https://docs.nvidia.com/cuda/cusparselt/license.html
nvidia-ml-py	https://pypi.org/project/nvidia-ml-py/	https://pypi.org/project/nvidia-ml-py/
nvidia-nccl-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/hpc-sdk/eula/index.html
nvidia-nccl-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/hpc-sdk/eula/index.html
nvidia-nvjitlink-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
nvidia-nvshmem-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-nvtx-cu11	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
nvidia-nvtx-cu12	https://developer.nvidia.com/cuda-zone	https://docs.nvidia.com/cuda/eula/index.html
Oat++	https://oatpp.io/	https://github.com/oatpp/oatpp/blob/master/LICENSE
oauthlib	https://github.com/oauthlib/oauthlib	https://github.com/oauthlib/oauthlib/blob/master/LICENSE
ocl-icd	https://github.com/OCL-dev/ocl-icd	https://github.com/OCL-dev/ocl-icd/blob/master/COPYING
ocl-icd-system	https://github.com/conda-forge/ocl-icd-system-feedst	https://github.com/conda-forge/ocl-icd-system-feedst blob/main/LICENSE.txt
olefile	https://www.decalage.info/python/olefileio	https://olefile.readthedocs.io/en/latest/License.html
omegaconf	https://github.com/omry/omegaconf	https://github.com/omry/omegaconf/blob/master/LICENSE
OmegaFold	https://github.com/HeliXonProtein/OmegaFold/tree/main	https://github.com/HeliXonProtein/OmegaFold/blob/main/LICENSE
omnicanvas	https://omnicanvas.readthedocs.io/en/latest/	https://github.com/samirelanduk/omnicanvas/blob/master/LICENSE
OpenFF	https://openforcefield.org/	https://github.com/openforcefield/openff-forcefields
openff-amber-ff-ports	https://github.com/openforcefield/openff-amber-ff-ports	https://github.com/openforcefield/openff-amber-ff-ports blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
openff-forcefields	https://openforcefield.org	https://github.com/openforcefield/openff-forcefields/blob/master/LICENSE
openff-fragmenter	https://github.com/openforcefield/openff-fragmenter	https://github.com/openforcefield/openff-fragmenter/blob/main/LICENSE
openff-interchange	https://github.com/openforcefield/openff-interchange	https://github.com/openforcefield/openff-interchange/blob/main/LICENSE
openff-models	https://github.com/openforcefield/openff-models	https://github.com/openforcefield/openff-models/blob/main/LICENSE
openff-nagl	https://docs.openforcefield.org/projects/nagl/en/latest/?badge=latest	https://github.com/openforcefield/openff-nagl/blob/main/LICENSE
openff-nagl-models	https://github.com/openforcefield/openff-nagl-models	https://github.com/openforcefield/openff-nagl-models/blob/main/LICENSE
openff-toolkit	https://openforcefield.org	https://github.com/openforcefield/openff-toolkit/blob/master/LICENSE
openff-toolkit-base	https://openforcefield.org	https://github.com/openforcefield/openff-toolkit/blob/master/LICENSE
openff-units	https://github.com/openforcefield/openff-units	https://github.com/openforcefield/openff-units/blob/main/LICENSE
openff-utilities	https://github.com/openforcefield/openff-utilities	https://github.com/openforcefield/openff-utilities/blob/main/LICENSE
openjpeg	https://github.com/uclouvain/openjpeg	https://github.com/uclouvain/openjpeg/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
openldap	https://www.openldap.org/software/repo.html	https://git.openldap.org/openldap/openldap/-/blob/master/LICENSE
OpenMM	https://openmm.org	https://docs.openmm.org/7.1.0/userguide/library.html#license
openmmtools	https://github.com/choderalab/openmmtools	https://github.com/choderalab/openmmtools/blob/main/LICENSE
openmoltools	https://github.com/choderalab/openmoltools	https://github.com/choderalab/openmoltools/blob/master/LICENSE
openpyxl	https://openpyxl.readthedocs.io/en/stable/	https://foss.heptapod.net/openpyxl/openpyxl/-/blob/branch/3.0/LICENCE.rst
openssl	https://www.openssl.org	https://github.com/openssl/openssl/blob/master/LICENSE.txt
opt-einsum	https://github.com/dgasmith/opt_einsum	https://github.com/dgasmith/opt_einsum/blob/master/LICENSE
opt_einsum	https://github.com/dgasmith/opt_einsum	https://github.com/dgasmith/opt_einsum/blob/main/LICENSE
OptKing	https://github.com/psi-rking/optking	https://github.com/psi-rking/optking/blob/master/LICENSE
optree	https://optree.readthedocs.io/en/latest/	https://github.com/metaopt/optree/blob/main/LICENSE
oscrypto	https://github.com/wbond/oscrypto	https://github.com/wbond/oscrypto/blob/master/LICENSE
overrides	https://github.com/mkorpela/overrides	https://github.com/mkorpela/overrides/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
packaging	https://github.com/pypa/packaging	https://github.com/pypa/packaging/blob/main/LICENSE
packmol	https://leandro.iqm.unicamp.br/m3g/packmol/home.shtml	https://github.com/m3g/packmol/blob/master/LICENSE
packmol_memgen	http://cpclab.uni-duesseldorf.de/	https://gitlab.com/cbjh/memgen/py-memgen/-/blob/main/LICENSE
pandas	https://pandas.pydata.org	https://github.com/pandas-dev/pandas/blob/main/LICENSE
pandocfilters	http://github.com/jgm/pandocfilters	https://github.com/jgm/pandocfilters/blob/master/LICENSE
panedr	https://github.com/MDAnalysis/panedr	https://github.com/MDAnalysis/panedr/blob/master/LICENSE.txt
pango	https://pango.gnome.org	https://gitlab.gnome.org/GNOME/pango/-/blob/main/COPYING
ParmEd	https://parmed.github.io/ParmEd/html/index.html	https://github.com/ParmEd/ParmEd#license
parser	https://github.com/typescript-eslint/typescript-eslint	https://github.com/typescript-eslint/typescript-eslint/raw/master/LICENSE
parso	https://parso.readthedocs.io/en/latest/	https://github.com/davidhalter/parso/blob/master/LICENSE.txt
pathos	https://github.com/uqfoundation/pathos	https://github.com/uqfoundation/pathos/blob/master/LICENSE
patsy	https://patsy.readthedocs.io/en/latest/	https://github.com/pydata/patsy/blob/master/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
pbkdf2	https://golang.org/x/crypto/pbkdf2	https://cs.opensource.google/go/x/crypto/+/master:LICENSE
pbr	https://docs.openstack.org/pbr/latest/	https://opendev.org/openstack/pbr/src/branch/master/LICENSE
pcmsolver	https://pcmsolver.readthedocs.io/en/stable/	https://github.com/PCMSolver/pcmsolver/blob/master/LICENSE
pcre	https://www.pcre.org	https://www.pcre.org/licence.txt
pcre2	https://www.pcre.org	https://github.com/PCRE2Project/pcre2/blob/master/LICENCE
pdb4amber	https://github.com/Amber-MD/pdb4amber	https://github.com/Amber-MD/pdb4amber/blob/master/LICENSE
pdbfixer	https://github.com/openmm/pdbfixer	https://github.com/openmm/pdbfixer/blob/master/LICENSE
pexpect	https://pexpect.readthedocs.io/	https://github.com/pexpect/pexpect/blob/master/LICENSE
pgconn	https://github.com/jackc/pgconn	https://github.com/jackc/pgconn/blob/master/LICENSE
pgio	https://github.com/jackc/pgio	https://github.com/jackc/pgio/blob/master/LICENSE
pgpassfile	https://github.com/jackc/pgpassfile	https://github.com/jackc/pgpassfile/blob/master/LICENSE
pgproto3	https://github.com/jackc/pgproto3/v2	https://github.com/jackc/pgproto3/blob/master/LICENSE
pgtype	https://github.com/jackc/pgtype	https://github.com/jackc/pgtype/blob/master/LICENSE
pgx	https://github.com/jackc/pgx/v4	https://github.com/jackc/pgx/blob/v4.0.1/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
phonopy	https://github.com/phonopy/phono3py	https://github.com/phonopy/phono3py/blob/develop/LICENSE
pickleshare	https://github.com/pickleshare/pickleshare	https://github.com/pickleshare/pickleshare/blob/master/LICENSE
Pillow	https://python-pillow.org	https://github.com/python-pillow/Pillow/blob/main/LICENSE
Pint	https://pint.readthedocs.io/en/stable/	https://github.com/hgrecco/pint/blob/master/LICENSE
pip	https://pip.pypa.io/	https://github.com/pypa/pip/blob/main/LICENSE.txt
pip-licenses	https://github.com/raimon49/pip-licenses	https://github.com/raimon49/pip-licenses/blob/master/LICENSE
pixman	https://pixman.org	https://cgit.freedesktop.org/pixman/tree/COPYING
pkginfo	https://launchpad.net/pkginfo	https://bazaar.launchpad.net/~tseaver/pkginfo/trunk/view/head:/LICENSE.txt
pkgutil-resolve-name	https://github.com/graingert/pkgutil-resolve-name	https://github.com/graingert/pkgutil-resolve-name?tab=License-1-ov-file#readme
pkgutil_resolve_name	https://github.com/graingert/pkgutil-resolve-name	https://github.com/graingert/pkgutil-resolve-name?tab=License-1-ov-file#readme
platformdirs	https://github.com/platformdirs/platformdirs	https://github.com/platformdirs/platformdirs/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
plotly	https://plotly.com/python/	https://github.com/plotly/plotly.py/blob/master/LICENSE.txt
plotly-orca	https://github.com/plotly/orca	https://github.com/plotly/orca/blob/master/LICENSE
plotly.js	https://github.com/plotly/plotly.js	https://github.com/plotly/plotly.js/raw/master/LICENSE
pluggy	https://pluggy.readthedocs.io/en/stable/index.html	https://github.com/pytest-dev/pluggy/blob/main/LICENSE
pooch	https://github.com/fatiando/pooch	https://github.com/fatiando/pooch/blob/main/LICENSE.txt
portalocker	https://github.com/WoLpH/portalocker	https://github.com/wolph/portalocker/blob/develop/LICENSE
pox	https://github.com/uqfoundation/pox	https://github.com/uqfoundation/pox/blob/master/LICENSE
ppft	https://github.com/uqfoundation/ppft	https://github.com/uqfoundation/ppft/blob/master/LICENSE
pq	https://github.com/lib/pq	https://github.com/lib/pq/blob/master/LICENSE.md
ProDy	http://www.csb.pitt.edu/ProDy	https://github.com/prody/ProDy/blob/master/LICENSE.rst
prometheus-client	https://github.com/prometheus/client_python	https://github.com/prometheus/client_python/blob/master/LICENSE
prometheus_client	https://github.com/prometheus/client_python	https://github.com/prometheus/client_python/blob/master/LICENSE
prompt-toolkit	https://python-prompt-toolkit.readthedocs.io/en/stable/	https://github.com/prompt-toolkit/python-prompt-toolkit/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
prompt_toolkit	https://github.com/prompt-toolkit/python-prompt-toolki	https://github.com/prompt-toolkit/python-prompt-toolki tab=BSD-3-Clause-1-ov-f readme
propcache	https://github.com/aio-libs/propcache	https://github.com/aio-libs/propcache?tab=Apache-2.0-1-ov-file#readme
protobuf	https://google.golang.org/protobuf	https://github.com/protocolbuffers/protobuf-go/blob/master/LICENSE
psi4	https://psicode.org	https://github.com/psi4/psi4/blob/master/COPYING.LESSER
psutil	https://psutil.readthedocs.io/en/latest/	https://github.com/giampaolo/psutil/blob/master/LICENSE
psycopg2	https://psycopg.org/	https://github.com/psycopg/psycopg2/blob/master/LICENSE
PTable	https://github.com/kxxoling/PTable	https://github.com/kxxoling/PTable/blob/master/COPYING
pthread-stubs	https://xcb.freedesktop.org	https://github.com/freedesktop/xcb-pthread-stubs/blob/master/COPYING
ptyprocess	https://ptyprocess.readthedocs.io/en/latest/	https://github.com/pexpect/ptyprocess/blob/master/LICENSE
pure-eval	http://github.com/alexmojaki/pure_eval	http://github.com/alexmojaki/pure_eval/blob/master/LICENSE.txt
pure_eval	http://github.com/alexmojaki/pure_eval	https://github.com/alexmojaki/pure_eval?tab=MIT-1-ov-file#readme

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
py	https://py.readthedocs.io/en/latest/	https://github.com/pytest-dev/py/blob/master/LICENSE
py-cpuinfo	https://github.com/workhorsy/py-cpuinfo	https://github.com/workhorsy/py-cpuinfo/blob/master/LICENSE
pyasn1	https://github.com/etingof/pyasn1	https://github.com/etingof/pyasn1/blob/master/LICENSE.rst
pyasn1-modules	https://github.com/etingof/pyasn1-modules	https://github.com/etingof/pyasn1-modules/blob/master/LICENSE.txt
pyasn1_modules	https://pyasn1.readthedocs.io/en/latest/contents.html	https://github.com/pyasn1/pyasn1-modules?tab=BSD-2-Clause-1-ov-file#readme
pybind11	http://pybind11.readthedocs.io/en/stable/	https://github.com/pybind/pybind11/blob/master/LICENSE
pybind11-abi	https://github.com/pybind/pybind11	https://github.com/pybind/pybind11/blob/master/LICENSE
pybind11_global	http://pybind11.readthedocs.io/en/stable/	https://github.com/pybind/pybind11/blob/master/LICENSE
pycairo	https://pycairo.readthedocs.io/en/latest/index.html	https://github.com/pygobject/pycairo/blob/master/COPYING
pycosat	https://github.com/conda/pycosat	https://github.com/conda/pycosat/blob/main/LICENSE
pycparser	https://github.com/eliben/pycparser	https://github.com/eliben/pycparser/blob/master/LICENSE
pydantic	https://pydantic-docs.helpmanual.io	https://github.com/samuelcolvin/pydantic/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
pydantic-core	https://github.com/pydantic/pydantic-core	https://github.com/pydantic/pydantic-core
pyedr	https://github.com/MDAnalysis/panedr	https://github.com/MDAnalysis/panedr/blob/master/LICENSE.txt
pyEMMA	http://github.com/markovmodel/PyEMMA	https://github.com/markovmodel/PyEMMA/blob/devel/LICENSE.txt
Pygments	https://pygments.org	https://github.com/pygments/pygments/blob/master/LICENSE
pygments	https://pygments.org/	https://github.com/pygments/pygments?tab=BSD-2-Clause-1-ov-file#readme
pygraphviz	https://pygraphviz.github.io	https://github.com/pygraphviz/pygraphviz/blob/main/LICENSE
pygtop	https://pygtop.readthedocs.io/en/latest/	https://github.com/samirelanduk/pygtop/blob/master/LICENSE
pyHanko	https://github.com/MatthiasValvekens/pyHanko	https://github.com/MatthiasValvekens/pyHanko/blob/master/LICENSE
pyhanko-certvalidator	https://github.com/MatthiasValvekens/certvalidator	https://github.com/MatthiasValvekens/certvalidator/blob/master/LICENSE
PyJWT	https://github.com/jpadilla/pyjwt	https://github.com/jpadilla/pyjwt/blob/master/LICENSE
pymbar	https://pymbar.org	https://github.com/choderalab/pymbar/blob/master/LICENSE
pynvml	https://github.com/gpuopenanalytics/pynvml	https://github.com/gpuopenanalytics/pynvml/blob/master/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
pyOpenSSL	https://pyopenssl.org/	https://github.com/pyca/pyopenssl/blob/main/LICENSE
pyparsing	https://pyparsing-docs.readthedocs.io/en/latest/	https://github.com/pyparsing/pyparsing/blob/master/LICENSE
PyPDF3	https://github.com/sfneal/PyPDF3	https://github.com/sfneal/PyPDF3/blob/master/LICENSE
pyrsistent	http://github.com/tobgu/pyrsistent/	https://github.com/tobgu/pyrsistent/blob/master/LICENSE.mit
pysam	https://github.com/pysam-developers/pysam	https://github.com/pysam-developers/pysam/blob/master/COPYING
PySocks	https://github.com/Anorov/PySocks	https://github.com/Anorov/PySocks/blob/master/LICENSE
pytables	https://www.pytables.org	https://github.com/PyTables/PyTables/blob/master/LICENSE.txt
python	https://www.python.org/	https://docs.python.org/3/license.html
python-bidi	https://github.com/MeirKriheli/python-bidi	https://github.com/MeirKriheli/python-bidi/blob/master/COPYING
python-constraint	https://github.com/python-constraint/python-constraint	https://github.com/python-constraint/python-constraint/blob/master/LICENSE
python-dateutil	https://dateutil.readthedocs.io	https://github.com/dateutil/dateutil/blob/master/LICENSE
python-json-logger	http://github.com/madzak/python-json-logger	https://github.com/madzak/python-json-logger/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
python-pptx	https://github.com/scanny/python-pptx	https://github.com/scanny/python-pptx/blob/master/LICENSE
python-ldap	https://www.python-ldap.org/	https://github.com/python-ldap/python-ldap/blob/python-ldap-3.4.0/LICENCE
python3-saml	https://github.com/onelogin/python3-saml	https://github.com/onelogin/python3-saml/blob/master/LICENSE
python_abi	https://github.com/conda-forge/python_abi-feedstock	https://github.com/conda-forge/python_abi-feedstock/blob/main/LICENSE.txt
pytorch-lightning	https://github.com/Lightning-AI/pytorch-lightning	https://github.com/Lightning-AI/pytorch-lightning/blob/master/LICENSE
pytz	https://pythonhosted.org/pytz	https://git.launchpad.net/pytz/tree/LICENSE.txt
pytz-deprecation-shim	https://github.com/pganssle/pytz-deprecation-shim	https://github.com/pganssle/pytz-deprecation-shim/blob/master/LICENSE
PyWavelets	https://github.com/PyWavelets/pywt	https://github.com/PyWavelets/pywt/blob/master/LICENSE
PyYAML	https://pyyaml.org/	https://github.com/yaml/pyyaml/blob/master/LICENSE
pyyml	No longer available	No longer available
pyzmq	https://pyzmq.readthedocs.io/en/latest/	https://github.com/zeromq/pyzmq/blob/main/COPYING.BSD
qcelemental	https://github.com/MolSSI/QCElemental	https://github.com/MolSSI/QCElemental/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
qcengine	https://github.com/MolSSI/QCEngine	https://github.com/MolSSI/QCEngine/blob/master/LICENSE
qrcode	https://github.com/lincolnloop/python-qrcode	https://github.com/lincolnloop/python-qrcode/blob/main/LICENSE
ramda	https://github.com/ramda/ramda	https://github.com/ramda/ramda/raw/master/LICENSE.txt
rapidjson	https://rapidjson.org/	https://github.com/Tencent/rapidjson/blob/master/license.txt
rdkit	https://www.rdkit.org	https://github.com/rdkit/rdkit/blob/master/license.txt
re2	https://github.com/google/re2	https://github.com/google/re2/blob/main/LICENSE
readme-renderer	https://github.com/pypa/readme_renderer	https://github.com/pypa/readme_renderer/blob/main/LICENSE
redis-py	https://github.com/andymccurdy/redis-py	https://github.com/redis/redis-py/blob/master/LICENSE
referencing	https://github.com/python-jsonschema/referencing	https://github.com/python-jsonschema/referencing
regex	https://github.com/mrabarnett/mrab-regex	https://github.com/mrabarnett/mrab-regex/blob/hg/LICENSE.txt
reportlab	https://www.reportlab.com	https://www.reportlab.com/dev/docs/faq/#1.3
reproc	https://github.com/DaanDeMeyer/reproc	https://github.com/DaanDeMeyer/reproc/blob/main/LICENSE
reproc-cpp	https://github.com/DaanDeMeyer/reproc	https://github.com/DaanDeMeyer/reproc/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
requests	https://requests.readthedocs.io	https://github.com/psf/requests/blob/main/LICENSE
requests-oauthlib	https://github.com/requests/requests-oauthlib	https://github.com/requests/requests-oauthlib/blob/master/LICENSE
requests-toolbelt	https://toolbelt.readthedocs.org	https://github.com/requests/toolbelt/blob/master/LICENSE
resumable	https://github.com/stevvooe/resumable	https://github.com/stevvooe/resumable/blob/master/LICENSE
retrying	https://github.com/rholder/retrying	https://github.com/rholder/retrying/blob/master/LICENSE
rfc3339-validator	https://github.com/naimetti/rfc3339-validator	https://github.com/naimetti/rfc3339-validator/blob/master/LICENSE
rfc3339_validator	https://github.com/naimetti/rfc3339-validator	https://github.com/naimetti/rfc3339-validator?tab=MIT-1-ov-file#readme
rfc3986	https://rfc3986.readthedocs.io/en/latest/	https://github.com/python-hyper/rfc3986/blob/main/LICENSE
rfc3986-validator	https://github.com/naimetti/rfc3986-validator	https://github.com/naimetti/rfc3986-validator/blob/master/LICENSE
rfc3987-syntax	https://github.com/willynilly/rfc3987-syntax	https://github.com/willynilly/rfc3987-syntax?tab=MIT-1-ov-file#readme
rich	https://github.com/Textualize/rich	https://github.com/Textualize/rich/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
rich-argparse	https://github.com/hamdanal/rich-argparse	https://github.com/hamdanal/rich-argparse/blob/main/LICENSE
rlPyCairo	https://pypi.org/project/rlPyCairo/	https://hg.reportlab.com/hg-public/rlPyCairo/file/tip/rlPyCairo/LICENSE.txt
RocksDict	https://github.com/rocksdict/RocksDict	https://github.com/rocksdict/RocksDict?tab=MIT-1-ov-file#readme
roman-numerals	https://github.com/AA-Turner/roman-numerals	https://github.com/AA-Turner/roman-numerals?tab=License-1-ov-file#readme
roman-numerals-py	https://github.com/AA-Turner/roman-numerals	https://github.com/AA-Turner/roman-numerals?tab=License-1-ov-file#readme
rotary-embedding-torch	https://github.com/lucidrains/rotary-embedding-tor	https://github.com/lucidrains/rotary-embedding-tor blob/main/LICENSE
rpds-py	https://github.com/crate-py/rpds	https://github.com/crate-py/rpds
rpmpack	https://github.com/google/rpmpack	https://github.com/google/rpmpack/blob/master/LICENSE
rsa	https://stuvel.eu/rsa	https://github.com/sybrenstuvel/python-rsa/blob/main/LICENSE
ruamel-yaml	https://sourceforge.net/p/ruamel-yaml/code/ci/default/tree/	https://sourceforge.net/p/ruamel-yaml/code/ci/default/tree/LICENSE
ruamel.yaml.clib	https://sourceforge.net/p/ruamel-yaml-clib/code/ci/default/tree/	https://sourceforge.net/p/ruamel-yaml-clib/code/ci/default/tree/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
s3transfer	https://github.com/boto/s3transfer	https://github.com/boto/s3transfer/blob/develop/LICENSE.txt
sacrebleu	https://github.com/mjpost/sacrebleu	https://github.com/mjpost/sacrebleu/blob/master/LICENSE.txt
sapiens	https://github.com/Merck/Sapiens	https://github.com/Merck/Sapiens/blob/main/LICENSE
sasl	https://mellium.im/sasl	https://github.com/mellium/sasl/blob/main/LICENSE
scikit-gstat	https://mmaelicke.github.io/scikit-gstat/	https://github.com/mmaelicke/scikit-gstat/blob/main/LICENSE
scikit-image	https://scikit-image.org	https://scikit-image.org/docs/dev/license.html
scikit-learn	https://scikit-learn.org/stable/	https://github.com/scikit-learn/scikit-learn/blob/main/COPYING
scikit-learn-extra	https://github.com/scikit-learn-contrib/scikit-learn-extra	https://github.com/scikit-learn-contrib/scikit-learn-extra/blob/main/LICENSE
scikit-network	https://github.com/sknetwork-team/scikit-network	https://github.com/sknetwork-team/scikit-network?tab=License-1-ov-file#readme
scipy	https://scipy.org	https://github.com/scipy/scipy/blob/main/LICENSE.txt
seaborn	https://seaborn.pydata.org	https://github.com/mwaskom/seaborn/blob/master/LICENSE
seaborn-base	https://seaborn.pydata.org	https://github.com/mwaskom/seaborn/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
semver	https://github.com/Masterminds/semver/v3	https://github.com/Masterminds/semver/blob/v3.0.3/LICENSE.txt
Send2Trash	https://github.com/arsenetar/send2trash	https://github.com/arsenetar/send2trash/blob/master/LICENSE
sentry-sdk	https://github.com/getsentry/sentry-python	https://github.com/getsentry/sentry-python/blob/master/LICENSE
setproctitle	https://github.com/dvarrazzo/py-setproctitle	https://github.com/dvarrazzo/py-setproctitle/blob/master/LICENSE
setuptools	https://github.com/pypa/setuptools	https://github.com/pypa/setuptools/blob/main/LICENSE
setuptools-scm	https://github.com/pypa/setuptools_scm/	https://github.com/pypa/setuptools_scm/blob/main/LICENSE
sh	https://github.com/amoffat/sh	https://github.com/amoffat/sh/blob/develop/LICENSE.txt
shellingham	https://github.com/sarugaku/shellingham	https://github.com/sarugaku/shellingham/blob/master/LICENSE
simdjson	https://github.com/simdjson/simdjson/tree/master	https://github.com/simdjson/simdjson/blob/master/LICENSE
simint	https://www.bennyp.org/research/simint/	https://github.com/simint-chem/simint-generator/blob/master/LICENSE
simplejson	https://github.com/simplejson/simplejson	https://github.com/simplejson/simplejson/blob/master/LICENSE.txt
six	https://github.com/benjaminp/six	https://github.com/benjaminp/six/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
smirnoff99frosst	https://github.com/openforcefield/smirnoff99frosst	https://github.com/openforcefield/smirnoff99Frosst/blob/master/LICENSE
smmap	https://github.com/gitpython-developers/smmap	https://github.com/gitpython-developers/smmap/blob/master/LICENSE
snappy	https://github.com/google/snappy	https://github.com/google/snappy/blob/main/COPYING
sniffio	https://github.com/python-trio/sniffio	https://github.com/python-trio/sniffio/blob/master/LICENSE.MIT
snowballstemmer	https://github.com/snowballstem/snowball	https://github.com/snowballstem/snowball/blob/master/COPYING
soupsieve	https://github.com/facelessuser/soupsieve	https://github.com/facelessuser/soupsieve/blob/main/LICENSE.md
spdlog	https://github.com/gabime/spdlog	https://github.com/gabime/spdlog?tab=License-1-ov-file#readme
spglib	https://spglib.readthedocs.io/	https://github.com/spglib/spglib/blob/develop/COPYING
sphinx	https://github.com/sphinx-doc/sphinx	https://github.com/sphinx-doc/sphinx/blob/5.x/LICENSE
sphinxcontrib-applehelp	https://github.com/sphinx-doc/sphinxcontrib-applehe	https://github.com/sphinx-doc/sphinxcontrib-applehe blob/master/LICENSE
sphinxcontrib-devhelp	https://github.com/sphinx-doc/sphinxcontrib-devhelp	https://github.com/sphinx-doc/sphinxcontrib-devhelp blob/master/LICENSE
sphinxcontrib-htmlhelp	https://github.com/sphinx-doc/sphinxcontrib-htmlhel	https://github.com/sphinx-doc/sphinxcontrib-htmlhel blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
sphinxcontrib-jsmath	https://github.com/sphinx-doc/sphinxcontrib-jsmath	https://github.com/sphinx-doc/sphinxcontrib-jsmath/blob/master/LICENSE
sphinxcontrib-qthelp	https://github.com/sphinx-doc/sphinxcontrib-qthelp	https://github.com/sphinx-doc/sphinxcontrib-qthelp/blob/master/LICENSE
sphinxcontrib-serializinghtml	https://github.com/sphinx-doc/sphinxcontrib-serializ	https://github.com/sphinx-doc/sphinxcontrib-serializ blob/master/LICENSE
SQLAlchemy	https://www.sqlalchemy.org	https://docs.sqlalchemy.org/en/14/copyright.html
SQLAlchemy	https://www.sqlalchemy.org/	https://github.com/sqlalchemy/sqlalchemy?tab=MIT-1-ov-file#readme
sqlite	https://sqlite.org/index.html	https://sqlite.org/copyright.html
sqlparse	https://github.com/andialbrecht/sqlparse	https://github.com/andialbrecht/sqlparse/blob/master/LICENSE
stack-data	http://github.com/alexmojaki/stack_data	https://github.com/alexmojaki/stack_data/blob/master/LICENSE.txt
stack_data	http://github.com/alexmojaki/stack_data	https://github.com/alexmojaki/stack_data?tab=MIT-1-ov-file#readme
starfile	https://github.com/alisterburt/starfile	https://github.com/teamtomo/starfile/blob/main/LICENSE
statsmodels	https://github.com/statsmodels/statsmodels	https://github.com/statsmodels/statsmodels/blob/main/LICENSE.txt
structlog	https://www.structlog.org/	https://github.com/hynek/structlog/blob/main/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
streamlit	https://github.com/streamlit/streamlit	https://github.com/streamlit/streamlit/blob/develop/LICENSE
svglib	https://github.com/deeplook/svglib	https://github.com/deeplook/svglib/blob/master/LICENSE.txt
sympy	https://sympy.org	https://github.com/sympy/sympy/blob/master/LICENSE
tables	https://www.pytables.org/	https://github.com/PyTables/PyTables/blob/master/LICENSE.txt
tabulate	https://github.com/astanin/python-tabulate	https://github.com/astanin/python-tabulate/blob/master/LICENSE
tbb	https://github.com/oneapi-src/oneTBB	https://github.com/oneapi-src/oneTBB/blob/master/LICENSE.txt
tenacity	https://github.com/jd/tenacity	https://github.com/jd/tenacity/blob/master/LICENSE
tensorboard	https://github.com/tensorflow/tensorboard	https://github.com/tensorflow/tensorboard/blob/master/LICENSE
tensorboard-data-server	https://github.com/tensorflow/tensorboard	https://github.com/tensorflow/tensorboard/blob/master/LICENSE
tensorboard-plugin-wit	https://github.com/pair-code/what-if-tool	https://github.com/PAIR-code/what-if-tool/blob/master/LICENSE
tensorboardX	https://github.com/lanpa/tensorboardX	https://github.com/lanpa/tensorboardX/blob/master/LICENSE
tensorflow	https://github.com/tensorflow/tensorflow	https://github.com/tensorflow/tensorflow/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
tensorflow-estimator	https://github.com/tensorflow/estimator	https://github.com/tensorflow/estimator/blob/master/LICENSE
tensorflow-io-gcs-filesystem	https://github.com/tensorflow/io	https://github.com/tensorflow/io/blob/master/LICENSE
tensorflow-probability	https://github.com/tensorflow/probability	https://github.com/tensorflow/probability/blob/main/LICENSE
termcolor	https://github.com/hugovk/termcolor	https://github.com/hugovk/termcolor/blob/main/COPYING.txt
terminado	https://github.com/jupyter/terminado	https://github.com/jupyter/terminado/blob/main/LICENSE
testpath	https://github.com/jupyter/testpath	https://github.com/jupyter/testpath/blob/master/LICENSE
textangular	https://github.com/fraywing/textAngular	https://github.com/fraywing/textAngular
texttable	https://github.com/foutaise/texttable/	https://github.com/foutaise/texttable/?tab=MIT-1-ov-file#readme
tf_keras	https://github.com/keras-team/keras-io	https://github.com/keras-team/keras-io/blob/master/LICENSE
tf_keras	https://github.com/keras-team/keras-io	https://github.com/keras-team/keras-io/blob/master/LICENSE
threadpoolctl	https://github.com/joblib/threadpoolctl	https://github.com/joblib/threadpoolctl/blob/master/LICENSE
three	https://github.com/mrdoob/three.js	https://github.com/mrdoob/three.js/raw/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
tiffile	https://github.com/cgohlke/tiffile/	https://github.com/cgohlke/tiffile/blob/master/LICENSE
tinycss2	https://github.com/Kozea/tinycss2/	https://github.com/Kozea/tinycss2/blob/master/LICENSE
tinyxml2	https://github.com/leethomason/tinyxml2	https://github.com/leethomason/tinyxml2/blob/master/LICENSE.txt
tk	https://www.tcl.tk/	https://www.tcl.tk/software/tcltk/license.html
toml	https://github.com/toml-lang/toml	https://github.com/toml-lang/toml/blob/main/LICENSE
tomli	https://github.com/hukkin/tomli	https://github.com/hukkin/tomli/blob/master/LICENSE
toolz	https://github.com/pytoolz/toolz	https://github.com/pytoolz/toolz/blob/master/LICENSE.txt
torch	https://pytorch.org/	https://pypi.org/project/torch/#license
torch-geometric	https://pyg.org/	https://github.com/pyg-team/pytorch_geometric?tab=MIT-1-ov-file#readme
torcheval	https://github.com/pytorch/torcheval	https://github.com/pytorch/torcheval?tab=License-1-ov-file#readme
torchmetrics	https://github.com/Lightning-AI/torchmetrics	https://github.com/Lightning-AI/torchmetrics?tab=Apache-2.0-1-ov-file#readme
tornado	https://www.tornadoweb.org	https://github.com/tornadoweb/tornado/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL		
tqdm	https://github.com/tqdm/tqdm	https://github.com/tqdm/tqdm/blob/master/LICENCE		
tracy	https://github.com/gear-genomics/tracy	https://github.com/gear-genomics/tracy/blob/main/LICENSE		
traitlets	https://github.com/ipython/traitlets	https://github.com/ipython/traitlets/blob/main/COPYING.md		
triton	https://github.com/openai/triton/	https://github.com/openai/triton/blob/main/LICENSE		
truststore	https://github.com/sethmlarson/trus	https://github.com/sethmlarson/truststore/blob/main/LICENSE		
truststore	https://github.com/sethmlarson/truststore	https://github.com/sethmlarson/truststore/blob/main/LICENSEtruststore	https://github.com/sethmlarson/truststore	https://github.com/sethmlarson/truststore/blob/main/LICENSE
ts-jest	https://github.com/kulshekhar/ts-jest	https://github.com/kulshekhar/ts-jest/raw/master/LICENSE.md		
ts-loader	https://github.com/TypeStrong/ts-loader	https://github.com/TypeStrong/ts-loader/raw/master/LICENSE		
twine	https://github.com/pypa/twine	https://github.com/pypa/twine/blob/main/LICENSE		
twinj uuid	https://github.com/twinj/uuid	https://github.com/twinj/uuid/blob/master/LICENSE		
types	https://github.com/babel/babel	https://github.com/babel/babel/raw/master/LICENSE		
types-python-dateutil	https://github.com/python/typeshed	https://github.com/python/typeshed?tab=License-1-ov-file#readme		
types-requests	https://pypi.org/project/types-requests/	https://github.com/python/typeshed/blob/main/LICENSE		

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
typescript	https://github.com/Microsoft/TypeScript	https://github.com/Microsoft/TypeScript/raw/master/LICENSE.txt
typing-inspection	https://github.com/pydantic/typing-inspection	https://github.com/pydantic/typing-inspection?tab=MIT-1-ov-file#readme
typing-utils	https://github.com/bojiang/typing_utils	https://github.com/bojiang/typing_utils?tab=Apache-2.0-1-ov-file#readme
typing_extensions	https://github.com/python/typing	https://github.com/python/typing/blob/master/LICENSE
typing_utils	https://github.com/bojiang/typing_utils	https://github.com/bojiang/typing_utils?tab=Apache-2.0-1-ov-file#readme
tzdata	https://github.com/python/tzdata	https://github.com/python/tzdata/blob/master/LICENSE
tzlocal	https://github.com/regebro/tzlocal	https://github.com/regebro/tzlocal/blob/master/LICENSE.txt
umf	https://github.com/oneapi-src/unified-memory-fram	https://github.com/oneapi-src/unified-memory-fram tab=License-1-ov-file#readme
umi-tools	https://github.com/CGATOxford/UMI-tools	https://github.com/CGATOxford/UMI-tools/blob/master/LICENSE
unicodedata2	https://github.com/fonttools/unicodedata2	https://github.com/fonttools/unicodedata2/blob/master/LICENSE
uri-template	https://gitlab.linss.com/open-source/python/uri-template	https://gitlab.linss.com/open-source/python/uri-template/-/blob/main/LICENSE
uritools	https://github.com/tkem/uritools/	https://github.com/tkem/uritools/blob/master/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL		
urllib3	https://urllib3.readthedocs.io/	https://github.com/urllib3/urllib3/blob/main/LICENSE.txt		
vine	https://github.com/celery/vine	https://github.com/celery/vine/blob/master/LICENSE		
vue	https://github.com/vuejs/vue	https://github.com/vuejs/vue/raw/master/LICENSE		
wadler_lindig	https://github.com/patrick-kidger/wadler_lindig	https://github.com/patrick-kidger/wadler_lindig?tab=Apache-2.0-1-ov-file#readme		
wadler_lindig	https://github.com/patrick-kidger/wadler_lindig	https://github.com/patrick-kidger/wadler_lindig?tab=Apache-2.0-1-ov-file#readmewcwidth	https://github.com/jquast/wcwidth	https://github.com/jquast/wcwidth/blob/master/LICENSE
wandb	https://github.com/wandb/wandb	https://github.com/wandb/wandb/blob/main/LICENSE		
webcolors	https://github.com/ubernostrum/webcolors	https://github.com/ubernostrum/webcolors?tab=BSD-3-Clause-1-ov-file#readme		
webencodings	https://github.com/gsnedders/python-webencodings	https://github.com/gsnedders/python-webencodings/blob/master/LICENSE		
websocket-client	https://github.com/websocket-client/websocket-client.git	https://github.com/websocket-client/websocket-client/blob/master/LICENSE		
Werkzeug	https://palletsprojects.com/p/werkzeug/	https://werkzeug.palletsprojects.com/en/2.1.x/license/		
westpa	http://github.com/westpa/westpa	https://github.com/westpa/westpa/blob/westpa2/LICENSE		
wheel	https://github.com/pypa/wheel	https://github.com/pypa/wheel/blob/main/LICENSE.txt		

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
widgetsnbextension	https://github.com/jupyter-widgets/ipywidgets#readme	https://github.com/jupyter-widgets/ipywidgets/blob/main/LICENSE
wrapt	https://github.com/GrahamDumpleton/wrapt	https://github.com/GrahamDumpleton/wrapt/blob/develop/LICENSE
wsutil	https://github.com/yhat/wsutil	https://github.com/yhat/wsutil/blob/master/LICENSE
x/lint	https://golang.org/x/lint	https://github.com/golang/lint/blob/master/LICENSE
x/mod	https://golang.org/x/mod/semver	https://cs.opensource.google/go/x/mod/+/master:LICENSE
x/net	https://golang.org/x/net	https://cs.opensource.google/go/x/net/+/master:LICENSE
x/oauth2	https://golang.org/x/oauth2	https://cs.opensource.google/go/x/oauth2/+/master:LICENSE
x/sys	https://golang.org/x/sys	https://cs.opensource.google/go/x/sys/+/master:LICENSE
x/text	https://golang.org/x/text	https://cs.opensource.google/go/x/text/+/master:LICENSE
x/tools	https://golang.org/x/tools	https://cs.opensource.google/go/x/tools/+/master:LICENSE
x/xerrors	https://golang.org/x/xerrors	https://cs.opensource.google/go/x/xerrors/+/master:LICENSE
xgboost	https://github.com/dmlc/xgboost	https://github.com/dmlc/xgboost/blob/master/LICENSE
xhtml2pdf	http://github.com/xhtml2pdf/xhtml2pdf	https://github.com/xhtml2pdf/xhtml2pdf/blob/master/LICENSE.txt

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
xlrd	https://github.com/python-excel/xlrd	https://github.com/python-excel/xlrd/blob/master/LICENSE
xlsxwriter	https://github.com/jmcnamara/XlsxWriter	https://github.com/jmcnamara/XlsxWriter/blob/main/LICENSE.txt
xmlsec	https://github.com/mehcode/python-xmlsec	https://github.com/mehcode/python-xmlsec/blob/master/LICENSE
xmltodict	https://github.com/martinblech/xmltodict	https://github.com/martinblech/xmltodict/blob/master/LICENSE
xorg-kbproto	https://gitlab.freedesktop.org/xorg/proto/kbproto	https://gitlab.freedesktop.org/xorg/proto/kbproto/-/blob/master/COPYING
xorg-libice	https://gitlab.freedesktop.org/xorg/lib/libice	https://gitlab.freedesktop.org/xorg/lib/libice/-/blob/master/COPYING
xorg-libsm	https://gitlab.freedesktop.org/xorg/lib/libsm	https://gitlab.freedesktop.org/xorg/lib/libsm/-/blob/master/COPYING
xorg-libx11	https://gitlab.freedesktop.org/xorg/lib/libx11	https://gitlab.freedesktop.org/xorg/lib/libx11/-/blob/master/COPYING
xorg-libxau	https://gitlab.freedesktop.org/xorg/lib/libxau	https://gitlab.freedesktop.org/xorg/lib/libxau/-/blob/master/COPYING
xorg-libxdmcp	https://gitlab.freedesktop.org/xorg/lib/libxdmcp	https://gitlab.freedesktop.org/xorg/lib/libxdmcp/-/blob/master/COPYING

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
xorg-libxext	https://gitlab.freedesktop.org/xorg/lib/libxext	https://gitlab.freedesktop.org/xorg/lib/libxext/-/blob/master/COPYING
xorg-libxrender	https://gitlab.freedesktop.org/xorg/lib/libxrender	https://gitlab.freedesktop.org/xorg/lib/libxrender/-/blob/master/COPYING
xorg-libxt	https://gitlab.freedesktop.org/xorg/lib/libxt	https://gitlab.freedesktop.org/xorg/lib/libxt/-/blob/master/COPYING
xorg-renderproto	https://gitlab.freedesktop.org/xorg/proto/renderproto	https://gitlab.freedesktop.org/xorg/proto/renderproto/-/blob/master/COPYING
xorg-xextproto	https://gitlab.freedesktop.org/xorg/proto/xextproto	https://gitlab.freedesktop.org/xorg/proto/xextproto/-/blob/master/COPYING
xorg-xproto	https://gitlab.freedesktop.org/xorg/proto/xproto	https://gitlab.freedesktop.org/xorg/proto/xproto/-/blob/master/COPYING
xxhash	https://github.com/cespare/xxhash/v2	https://github.com/cespare/xxhash/blob/main/LICENSE.txt
xz	https://github.com/ulikunitz/xz	https://github.com/ulikunitz/xz/blob/master/LICENSE
yaml	https://pyyaml.org/	https://github.com/yaml/pyyaml/blob/master/LICENSE
yaml-cpp	https://github.com/jbeder/yaml-cpp	https://github.com/jbeder/yaml-cpp/blob/master/LICENSE
yaml.v2	https://gopkg.in/yaml.v2	https://github.com/go-yaml/yaml/blob/v2.4.0/LICENSE
yaml.v3	https://gopkg.in/yaml.v3	https://github.com/go-yaml/yaml/blob/v3/LICENSE

continues on next page

Table 1 – continued from previous page

Name of Project	Website	License URL
yarl	https://github.com/aio-libs/yarl/	https://github.com/aio-libs/yarl/blob/master/LICENSE
yaspin	https://github.com/pavdmyt/yaspin	https://github.com/pavdmyt/yaspin/blob/master/LICENSE
yfiles	https://www.yworks.com/products/yfiles	commercial license
yml	https://pypi.org/project/yml/	N/A
zap	https://go.uber.org/zap	https://github.com/uber-go/zap/blob/master/LICENSE.txt
zipp	https://github.com/jaraco/zipp	https://github.com/jaraco/zipp/blob/main/LICENSE
zlib	https://zlib.net/	https://zlib.net/zlib_license.html
zstandard	https://github.com/indygreg/python-zstandard	https://github.com/indygreg/python-zstandard/blob/main/LICENSE
zstd	https://facebook.github.io/zstd/	https://github.com/facebook/zstd/blob/dev/LICENSE
click-option-group	https://github.com/click-contrib/click-option-group	https://github.com/click-contrib/click-option-group/blob/master/LICENSE

10.1 GCC

On the Linux platform, The OpenEye toolkits are linked with the GCC libraries using an eligible compilation process under the terms of the GCC RUNTIME LIBRARY EXCEPTION (see below).

The OpenEye toolkits are NOT open source software and are NOT governed by GPL or other open source licenses. Your right to use the toolkits are governed by the OpenEye license which is binding to You.

For certain Linux distributions, GCC library components are included as object files. Your rights to use GCC library components (libstdc++, libgcc, and libgomp) included with these distributions are governed by the GPLv3 license as follows:

The components libstdc++, libgcc, and libgomp are part of GCC.

- GCC is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.
- GCC is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the

implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with GCC (see below). If not, see Licenses GNU Project .

Per GPLv3, section 6d, the source for GCC is available for download from the following location: GNU. The GCC library components included here are from version gcc-4.9.3 and are built using the standard instructions at Installing GCC.

10.1.1 GCC RUNTIME LIBRARY EXCEPTION

```
Version 3.1, 31 March 2009

Copyright © 2009 Free Software Foundation, Inc. http://fsf.org/

Everyone is permitted to copy and distribute verbatim copies of this license document,␣
↪but changing it is not allowed.

This GCC Runtime Library Exception ("Exception") is an additional permission under␣
↪section 7 of the GNU General Public License,
version 3 ("GPLv3"). It applies to a given file (the "Runtime Library") that bears a␣
↪notice placed by the copyright holder
of the file stating that the file is governed by GPLv3 along with this Exception.

When you use GCC to compile a program, GCC may combine portions of certain GCC header␣
↪files and runtime libraries with
the compiled program. The purpose of this Exception is to allow compilation of non-GPL␣
↪(including proprietary) programs
to use, in this way, the header files and runtime libraries covered by this Exception.

0. Definitions.

A file is an "Independent Module" if it either requires the Runtime Library for␣
↪execution after a Compilation Process,
or makes use of an interface provided by the Runtime Library, but is not otherwise based␣
↪on the Runtime Library.

"GCC" means a version of the GNU Compiler Collection, with or without modifications,␣
↪governed by version 3 (or a specified
later version) of the GNU General Public License (GPL) with the option of using any␣
↪subsequent versions published by the FSF.

"GPL-compatible Software" is software whose conditions of propagation, modification and␣
↪use would permit combination with
GCC in accord with the license of GCC.

"Target Code" refers to output from any compiler for a real or virtual target processor␣
↪architecture, in executable
form or suitable for input to an assembler, loader, linker and/or execution phase.␣
↪Notwithstanding that, Target Code
does not include data in any format that is used as a compiler intermediate␣
↪representation, or used for producing a
compiler intermediate representation.
```

(continues on next page)

(continued from previous page)

```
The "Compilation Process" transforms code entirely represented in non-intermediate␣
↪languages designed for human-written
code, and/or in Java Virtual Machine byte code, into Target Code. Thus, for example, use␣
↪of source code generators and
preprocessors need not be considered part of the Compilation Process, since the␣
↪Compilation Process can be understood
as starting with the output of the generators or preprocessors.

A Compilation Process is "Eligible" if it is done using GCC, alone or with other GPL-
↪compatible software, or if it is
done without using any work based on GCC. For example, using non-GPL-compatible Software␣
↪to optimize any GCC intermediate
representations would not qualify as an Eligible Compilation Process.

1. Grant of Additional Permission.

You have permission to propagate a work of Target Code formed by combining the Runtime␣
↪Library with Independent Modules,
even if such propagation would otherwise violate the terms of GPLv3, provided that all␣
↪Target Code was generated by
Eligible Compilation Processes. You may then convey such a combination under terms of␣
↪your choice, consistent with the
licensing of the Independent Modules.

2. No Weakening of GCC Copyleft.

The availability of this Exception does not imply any general presumption that third-
↪party software is unaffected by
the copyleft requirements of the license of GCC.
```

See also

- http://www.gnu.org/licenses/gcc-exception.html

10.1.2 GNU GENERAL PUBLIC LICENSE

```
                    GNU GENERAL PUBLIC LICENSE
                       Version 3, 29 June 2007

 Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.

                            Preamble

 The GNU General Public License is a free, copyleft license for
 software and other kinds of works.

   The licenses for most software and other practical works are designed
```

(continues on next page)

(continued from previous page)

```
to take away your freedom to share and change the works.  By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users.  We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors.  You can apply it to
your programs, too.

  When we speak of free software, we are referring to freedom, not
price.  Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

  To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights.  Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

  For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received.  You must make sure that they, too, receive
or can get the source code.  And you must show them these terms so they
know their rights.

  Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

  For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software.  For both users' and
authors' sake, the GPL requires that modified versions be marked as
changed, so that their problems will not be attributed erroneously to
authors of previous versions.

  Some devices are designed to deny users access to install or run
modified versions of the software inside them, although the manufacturer
can do so.  This is fundamentally incompatible with the aim of
protecting users' freedom to change the software.  The systematic
pattern of such abuse occurs in the area of products for individuals to
use, which is precisely where it is most unacceptable.  Therefore, we
have designed this version of the GPL to prohibit the practice for those
products.  If such problems arise substantially in other domains, we
stand ready to extend this provision to those domains in future versions
of the GPL, as needed to protect the freedom of users.

  Finally, every program is threatened constantly by software patents.
States should not allow patents to restrict development and use of
software on general-purpose computers, but in those that do, we wish to
avoid the special danger that patents applied to a free program could
make it effectively proprietary.  To prevent this, the GPL assures that
```

(continues on next page)

(continued from previous page)

```
patents cannot be used to render the program non-free.

  The precise terms and conditions for copying, distribution and
modification follow.

                    TERMS AND CONDITIONS

  0. Definitions.

  "This License" refers to version 3 of the GNU General Public License.

  "Copyright" also means copyright-like laws that apply to other kinds of
works, such as semiconductor masks.

  "The Program" refers to any copyrightable work licensed under this
License.  Each licensee is addressed as "you".  "Licensees" and
"recipients" may be individuals or organizations.

  To "modify" a work means to copy from or adapt all or part of the work
in a fashion requiring copyright permission, other than the making of an
exact copy.  The resulting work is called a "modified version" of the
earlier work or a work "based on" the earlier work.

  A "covered work" means either the unmodified Program or a work based
on the Program.

  To "propagate" a work means to do anything with it that, without
permission, would make you directly or secondarily liable for
infringement under applicable copyright law, except executing it on a
computer or modifying a private copy.  Propagation includes copying,
distribution (with or without modification), making available to the
public, and in some countries other activities as well.

  To "convey" a work means any kind of propagation that enables other
parties to make or receive copies.  Mere interaction with a user through
a computer network, with no transfer of a copy, is not conveying.

  An interactive user interface displays "Appropriate Legal Notices"
to the extent that it includes a convenient and prominently visible
feature that (1) displays an appropriate copyright notice, and (2)
tells the user that there is no warranty for the work (except to the
extent that warranties are provided), that licensees may convey the
work under this License, and how to view a copy of this License.  If
the interface presents a list of user commands or options, such as a
menu, a prominent item in the list meets this criterion.

  1. Source Code.

  The "source code" for a work means the preferred form of the work
for making modifications to it.  "Object code" means any non-source
form of a work.
```

(continues on next page)

(continued from previous page)

```
  A "Standard Interface" means an interface that either is an official
standard defined by a recognized standards body, or, in the case of
interfaces specified for a particular programming language, one that
is widely used among developers working in that language.

  The "System Libraries" of an executable work include anything, other
than the work as a whole, that (a) is included in the normal form of
packaging a Major Component, but which is not part of that Major
Component, and (b) serves only to enable use of the work with that
Major Component, or to implement a Standard Interface for which an
implementation is available to the public in source code form.  A
"Major Component", in this context, means a major essential component
(kernel, window system, and so on) of the specific operating system
(if any) on which the executable work runs, or a compiler used to
produce the work, or an object code interpreter used to run it.

  The "Corresponding Source" for a work in object code form means all
the source code needed to generate, install, and (for an executable
work) run the object code and to modify the work, including scripts to
control those activities.  However, it does not include the work's
System Libraries, or general-purpose tools or generally available free
programs which are used unmodified in performing those activities but
which are not part of the work.  For example, Corresponding Source
includes interface definition files associated with source files for
the work, and the source code for shared libraries and dynamically
linked subprograms that the work is specifically designed to require,
such as by intimate data communication or control flow between those
subprograms and other parts of the work.

  The Corresponding Source need not include anything that users
can regenerate automatically from other parts of the Corresponding
Source.

  The Corresponding Source for a work in source code form is that
same work.

  2. Basic Permissions.

  All rights granted under this License are granted for the term of
copyright on the Program, and are irrevocable provided the stated
conditions are met.  This License explicitly affirms your unlimited
permission to run the unmodified Program.  The output from running a
covered work is covered by this License only if the output, given its
content, constitutes a covered work.  This License acknowledges your
rights of fair use or other equivalent, as provided by copyright law.

  You may make, run and propagate covered works that you do not
convey, without conditions so long as your license otherwise remains
in force.  You may convey covered works to others for the sole purpose
of having them make modifications exclusively for you, or provide you
with facilities for running those works, provided that you comply with
the terms of this License in conveying all material for which you do
```

(continues on next page)

(continued from previous page)

```
not control copyright.  Those thus making or running the covered works
for you must do so exclusively on your behalf, under your direction
and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

  Conveying under any other circumstances is permitted solely under
the conditions stated below.  Sublicensing is not allowed; section 10
makes it unnecessary.

  3. Protecting Users' Legal Rights From Anti-Circumvention Law.

  No covered work shall be deemed part of an effective technological
measure under any applicable law fulfilling obligations under article
11 of the WIPO copyright treaty adopted on 20 December 1996, or
similar laws prohibiting or restricting circumvention of such
measures.

  When you convey a covered work, you waive any legal power to forbid
circumvention of technological measures to the extent such circumvention
is effected by exercising rights under this License with respect to
the covered work, and you disclaim any intention to limit operation or
modification of the work as a means of enforcing, against the work's
users, your or third parties' legal rights to forbid circumvention of
technological measures.

  4. Conveying Verbatim Copies.

  You may convey verbatim copies of the Program's source code as you
receive it, in any medium, provided that you conspicuously and
appropriately publish on each copy an appropriate copyright notice;
keep intact all notices stating that this License and any
non-permissive terms added in accord with section 7 apply to the code;
keep intact all notices of the absence of any warranty; and give all
recipients a copy of this License along with the Program.


  You may charge any price or no price for each copy that you convey,
and you may offer support or warranty protection for a fee.

  5. Conveying Modified Source Versions.

  You may convey a work based on the Program, or the modifications to
produce it from the Program, in the form of source code under the
terms of section 4, provided that you also meet all of these conditions:

   a) The work must carry prominent notices stating that you modified
   it, and giving a relevant date.

   b) The work must carry prominent notices stating that it is
   released under this License and any conditions added under section
   7.  This requirement modifies the requirement in section 4 to
   "keep intact all notices".
```

(continues on next page)

(continued from previous page)

```
    c) You must license the entire work, as a whole, under this
    License to anyone who comes into possession of a copy.  This
    License will therefore apply, along with any applicable section 7
    additional terms, to the whole of the work, and all its parts,
    regardless of how they are packaged.  This License gives no
    permission to license the work in any other way, but it does not
    invalidate such permission if you have separately received it.

    d) If the work has interactive user interfaces, each must display
    Appropriate Legal Notices; however, if the Program has interactive
    interfaces that do not display Appropriate Legal Notices, your
    work need not make them do so.

  A compilation of a covered work with other separate and independent
works, which are not by their nature extensions of the covered work,
and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an
"aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit.  Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

  6. Conveying Non-Source Forms.

  You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License,
in one of these ways:

  a) Convey the object code in, or embodied in, a physical product
  (including a physical distribution medium), accompanied by the
  Corresponding Source fixed on a durable physical medium
  customarily used for software interchange.

  b) Convey the object code in, or embodied in, a physical product
  (including a physical distribution medium), accompanied by a
  written offer, valid for at least three years and valid for as
  long as you offer spare parts or customer support for that product
  model, to give anyone who possesses the object code either (1) a
  copy of the Corresponding Source for all the software in the
  product that is covered by this License, on a durable physical
  medium customarily used for software interchange, for a price no
  more than your reasonable cost of physically performing this
  conveying of source, or (2) access to copy the
  Corresponding Source from a network server at no charge.

  c) Convey individual copies of the object code with a copy of the
  written offer to provide the Corresponding Source.  This
  alternative is allowed only occasionally and noncommercially, and
  only if you received the object code with such an offer, in accord
```

(continues on next page)

(continued from previous page)

```
   with subsection 6b.

   d) Convey the object code by offering access from a designated
   place (gratis or for a charge), and offer equivalent access to the
   Corresponding Source in the same way through the same place at no
   further charge.  You need not require recipients to copy the
   Corresponding Source along with the object code.  If the place to
   copy the object code is a network server, the Corresponding Source
   may be on a different server (operated by you or a third party)
   that supports equivalent copying facilities, provided you maintain
   clear directions next to the object code saying where to find the
   Corresponding Source.  Regardless of what server hosts the
   Corresponding Source, you remain obligated to ensure that it is
   available for as long as needed to satisfy these requirements.

   e) Convey the object code using peer-to-peer transmission, provided
   you inform other peers where the object code and Corresponding
   Source of the work are being offered to the general public at no
   charge under subsection 6d.

  A separable portion of the object code, whose source code is excluded
from the Corresponding Source as a System Library, need not be
included in conveying the object code work.

  A "User Product" is either (1) a "consumer product", which means any
tangible personal property which is normally used for personal, family,
or household purposes, or (2) anything designed or sold for incorporation
into a dwelling.  In determining whether a product is a consumer product,
doubtful cases shall be resolved in favor of coverage.  For a particular
product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user
actually uses, or expects or is expected to use, the product.  A product
is a consumer product regardless of whether the product has substantial
commercial, industrial or non-consumer uses, unless such uses represent
the only significant mode of use of the product.

  "Installation Information" for a User Product means any methods,
procedures, authorization keys, or other information required to install
and execute modified versions of a covered work in that User Product from
a modified version of its Corresponding Source.  The information must
suffice to ensure that the continued functioning of the modified object
code is in no case prevented or interfered with solely because
modification has been made.

  If you convey an object code work under this section in, or with, or
specifically for use in, a User Product, and the conveying occurs as
part of a transaction in which the right of possession and use of the
User Product is transferred to the recipient in perpetuity or for a
fixed term (regardless of how the transaction is characterized), the
Corresponding Source conveyed under this section must be accompanied
by the Installation Information.  But this requirement does not apply
```

(continues on next page)

(continued from previous page)

```
if neither you nor any third party retains the ability to install
modified object code on the User Product (for example, the work has
been installed in ROM).

  The requirement to provide Installation Information does not include a
requirement to continue to provide support service, warranty, or updates
for a work that has been modified or installed by the recipient, or for
the User Product in which it has been modified or installed.  Access to a
network may be denied when the modification itself materially and
adversely affects the operation of the network or violates the rules and
protocols for communication across the network.

  Corresponding Source conveyed, and Installation Information provided,
in accord with this section must be in a format that is publicly
documented (and with an implementation available to the public in
source code form), and must require no special password or key for
unpacking, reading or copying.

  7. Additional Terms.

  "Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions.
Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent
that they are valid under applicable law.  If additional permissions
apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

  When you convey a copy of a covered work, you may at your option
remove any additional permissions from that copy, or from any part of
it.  (Additional permissions may be written to require their own
removal in certain cases when you modify the work.)  You may place
additional permissions on material, added by you to a covered work,
for which you have or can give appropriate copyright permission.

  Notwithstanding any other provision of this License, for material you
add to a covered work, you may (if authorized by the copyright holders of
that material) supplement the terms of this License with terms:

    a) Disclaiming warranty or limiting liability differently from the
    terms of sections 15 and 16 of this License; or

    b) Requiring preservation of specified reasonable legal notices or
    author attributions in that material or in the Appropriate Legal
    Notices displayed by works containing it; or

    c) Prohibiting misrepresentation of the origin of that material, or
    requiring that modified versions of such material be marked in
    reasonable ways as different from the original version; or

    d) Limiting the use for publicity purposes of names of licensors or
```

(continues on next page)

(continued from previous page)

```
   authors of the material; or

   e) Declining to grant rights under trademark law for use of some
   trade names, trademarks, or service marks; or

   f) Requiring indemnification of licensors and authors of that
   material by anyone who conveys the material (or modified versions of
   it) with contractual assumptions of liability to the recipient, for
   any liability that these contractual assumptions directly impose on
   those licensors and authors.

  All other non-permissive additional terms are considered "further
restrictions" within the meaning of section 10.  If the Program as you
received it, or any part of it, contains a notice stating that it is
governed by this License along with a term that is a further
restriction, you may remove that term.  If a license document contains
a further restriction but permits relicensing or conveying under this
License, you may add to a covered work material governed by the terms
of that license document, provided that the further restriction does
not survive such relicensing or conveying.

  If you add terms to a covered work in accord with this section, you
must place, in the relevant source files, a statement of the
additional terms that apply to those files, or a notice indicating
where to find the applicable terms.

  Additional terms, permissive or non-permissive, may be stated in the
form of a separately written license, or stated as exceptions;
the above requirements apply either way.

  8. Termination.

  You may not propagate or modify a covered work except as expressly
provided under this License.  Any attempt otherwise to propagate or
modify it is void, and will automatically terminate your rights under
this License (including any patent licenses granted under the third
paragraph of section 11).

  However, if you cease all violation of this License, then your
license from a particular copyright holder is reinstated (a)
provisionally, unless and until the copyright holder explicitly and
finally terminates your license, and (b) permanently, if the copyright
holder fails to notify you of the violation by some reasonable means
prior to 60 days after the cessation.

  Moreover, your license from a particular copyright holder is
reinstated permanently if the copyright holder notifies you of the
violation by some reasonable means, this is the first time you have
received notice of violation of this License (for any work) from that
copyright holder, and you cure the violation prior to 30 days after
your receipt of the notice.
```

(continues on next page)

(continued from previous page)

```
  Termination of your rights under this section does not terminate the
licenses of parties who have received copies or rights from you under
this License.  If your rights have been terminated and not permanently
reinstated, you do not qualify to receive new licenses for the same
material under section 10.

  9. Acceptance Not Required for Having Copies.

  You are not required to accept this License in order to receive or
run a copy of the Program.  Ancillary propagation of a covered work
occurring solely as a consequence of using peer-to-peer transmission
to receive a copy likewise does not require acceptance.  However,
nothing other than this License grants you permission to propagate or
modify any covered work.  These actions infringe copyright if you do
not accept this License.  Therefore, by modifying or propagating a
covered work, you indicate your acceptance of this License to do so.

  10. Automatic Licensing of Downstream Recipients.

  Each time you convey a covered work, the recipient automatically
receives a license from the original licensors, to run, modify and
propagate that work, subject to this License.  You are not responsible
for enforcing compliance by third parties with this License.

  An "entity transaction" is a transaction transferring control of an
organization, or substantially all assets of one, or subdividing an
organization, or merging organizations.  If propagation of a covered
work results from an entity transaction, each party to that
transaction who receives a copy of the work also receives whatever
licenses to the work the party's predecessor in interest had or could
give under the previous paragraph, plus a right to possession of the
Corresponding Source of the work from the predecessor in interest, if
the predecessor has it or can get it with reasonable efforts.

  You may not impose any further restrictions on the exercise of the
rights granted or affirmed under this License.  For example, you may
not impose a license fee, royalty, or other charge for exercise of
rights granted under this License, and you may not initiate litigation
(including a cross-claim or counterclaim in a lawsuit) alleging that
any patent claim is infringed by making, using, selling, offering for
sale, or importing the Program or any portion of it.

  11. Patents.

  A "contributor" is a copyright holder who authorizes use under this
License of the Program or a work on which the Program is based.  The
work thus licensed is called the contributor's "contributor version".

  A contributor's "essential patent claims" are all patent claims
owned or controlled by the contributor, whether already acquired or
hereafter acquired, that would be infringed by some manner, permitted
by this License, of making, using, or selling its contributor version,
```

(continues on next page)

```
but do not include claims that would be infringed only as a
consequence of further modification of the contributor version.  For
purposes of this definition, "control" includes the right to grant
patent sublicenses in a manner consistent with the requirements of
this License.

  Each contributor grants you a non-exclusive, worldwide, royalty-free
patent license under the contributor's essential patent claims, to
make, use, sell, offer for sale, import and otherwise run, modify and
propagate the contents of its contributor version.

  In the following three paragraphs, a "patent license" is any express
agreement or commitment, however denominated, not to enforce a patent
(such as an express permission to practice a patent or covenant not to
sue for patent infringement).  To "grant" such a patent license to a
party means to make such an agreement or commitment not to enforce a
patent against the party.

  If you convey a covered work, knowingly relying on a patent license,
and the Corresponding Source of the work is not available for anyone
to copy, free of charge and under the terms of this License, through a
publicly available network server or other readily accessible means,
then you must either (1) cause the Corresponding Source to be so
available, or (2) arrange to deprive yourself of the benefit of the
patent license for this particular work, or (3) arrange, in a manner
consistent with the requirements of this License, to extend the patent
license to downstream recipients.  "Knowingly relying" means you have
actual knowledge that, but for the patent license, your conveying the
covered work in a country, or your recipient's use of the covered work
in a country, would infringe one or more identifiable patents in that
country that you have reason to believe are valid.

  If, pursuant to or in connection with a single transaction or
arrangement, you convey, or propagate by procuring conveyance of, a
covered work, and grant a patent license to some of the parties
receiving the covered work authorizing them to use, propagate, modify
or convey a specific copy of the covered work, then the patent license
you grant is automatically extended to all recipients of the covered
work and works based on it.

  A patent license is "discriminatory" if it does not include within
the scope of its coverage, prohibits the exercise of, or is
conditioned on the non-exercise of one or more of the rights that are
specifically granted under this License.  You may not convey a covered
work if you are a party to an arrangement with a third party that is
in the business of distributing software, under which you make payment
to the third party based on the extent of your activity of conveying
the work, and under which the third party grants, to any of the
parties who would receive the covered work from you, a discriminatory
patent license (a) in connection with copies of the covered work
conveyed by you (or copies made from those copies), or (b) primarily
for and in connection with specific products or compilations that
contain the covered work, unless you entered into that arrangement,
```

(continues on next page)

(continued from previous page)

```
or that patent license was granted, prior to 28 March 2007.

  Nothing in this License shall be construed as excluding or limiting
any implied license or other defenses to infringement that may
otherwise be available to you under applicable patent law.

  12. No Surrender of Others' Freedom.

  If conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License.  If you cannot convey a
covered work so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you may
not convey it at all.  For example, if you agree to terms that obligate you
to collect a royalty for further conveying from those to whom you convey
the Program, the only way you could satisfy both those terms and this
License would be to refrain entirely from conveying the Program.

  13. Use with the GNU Affero General Public License.

  Notwithstanding any other provision of this License, you have
permission to link or combine any covered work with a work licensed
under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work.  The terms of this
License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License,
section 13, concerning interaction through a network will apply to the
combination as such.

  14. Revised Versions of this License.

  The Free Software Foundation may publish revised and/or new versions of
the GNU General Public License from time to time.  Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

  Each version is given a distinguishing version number.  If the
Program specifies that a certain numbered version of the GNU General
Public License "or any later version" applies to it, you have the
option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software
Foundation.  If the Program does not specify a version number of the
GNU General Public License, you may choose any version ever published
by the Free Software Foundation.

  If the Program specifies that a proxy can decide which future
versions of the GNU General Public License can be used, that proxy's
public statement of acceptance of a version permanently authorizes you
to choose that version for the Program.

  Later license versions may give you additional or different
permissions.  However, no additional obligations are imposed on any
```

(continues on next page)

(continued from previous page)

```
author or copyright holder as a result of your choosing to follow a
later version.

  15. Disclaimer of Warranty.

  THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW.  EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE.  THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
IS WITH YOU.  SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

  16. Limitation of Liability.

  IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

  17. Interpretation of Sections 15 and 16.

  If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

                     END OF TERMS AND CONDITIONS

            How to Apply These Terms to Your New Programs

  If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

  To do so, attach the following notices to the program.  It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

    <one line to give the program's name and a brief idea of what it does.>
    Copyright (C) <year>  <name of author>

    This program is free software: you can redistribute it and/or modify
```

(continues on next page)

(continued from previous page)

```
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation, either version 3 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program.  If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

  If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

    <program>  Copyright (C) <year>  <name of author>
    This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

  You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

  The GNU General Public License does not permit incorporating your program
into proprietary programs.  If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.  But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.
```

See also

- http://www.gnu.org/licenses/gpl.txt

INDEX

A

AddFeature
    OEDocking::OECustomConstraints, 85
AddLigand
    OEModels::OEROCSQueryModelBuilder, 105
AddReceptor
    OEDocking::OEPosit, 46
AddSmarts
    OEDocking::OEFeature, 87
AddSphere
    OEDocking::OEFeature, 86
AnnotatePose
    OEDocking::OEDock, 41
    OEDocking::OEScore, 58

B

Build
    OEModels::OEROCSQueryModelBuilder, 105

C

CacheScoringSetup
    OEDocking::OEDock, 41
    OEDocking::OEScore, 58
Clear
    OEDocking::OECustomConstraints, 86
ClearLigands
    OEModels::OEROCSQueryModelBuilder, 105
ClearReceptors
    OEDocking::OEPosit, 47
ClearSmarts
    OEDocking::OEFeature, 87
ClearSpheres
    OEDocking::OEFeature, 87
Constructor
    OEDocking::OEDock, 38
    OEDocking::OEDockOptions, 37
    OEDocking::OEHybrid, 42
    OEDocking::OEMakeReceptorOptions, 42
    OEDocking::OEPoseOptimizer, 44
    OEDocking::OEPoseOptimizerOptions, 45
    OEDocking::OEPosit, 46
    OEDocking::OEPositOptions, 50
    OEDocking::OEPositProbability, 54
    OEDocking::OEScore, 55
    OEDocking::OEShapeFit, 78
    OEDocking::OEShapeFitOptions, 79
    OEDocking::OEShapeFitResults, 81
    OEModels::OEROCSQueryModelBuilder, 104
    OEModels::OEROCSQueryModelOptions, 106
Constructors
    OEDocking::OEBox, 84
    OEDocking::OECustomConstraints, 85
    OEDocking::OEPositResults, 54
    OEDocking::OESinglePoseResult, 59
CreateCopy
    OEDocking::OEAlignReceptorFunction, 82
    OEDocking::OEFeature, 88
    OEDocking::OEShapeFitOptions, 79
    OEModels::OEROCSQueryModelOptions, 106

D

DeleteFeature
    OEDocking::OECustomConstraints, 85
DeleteSmarts
    OEDocking::OEFeature, 87
DeleteSphere
    OEDocking::OEFeature, 87
Destructor
    OEDocking::OEBoxBase, 83
    OEDocking::OEFeature, 86
    OEDocking::OESphereBase, 89
Dock
    OEDocking::OEPosit, 47
dockmolecules.java
    Example Code, 19
DockMultiConformerMolecule
    OEDocking::OEDock, 38
    OEDocking::OEPosit, 48

E

Estimate
    OEDocking::OEPositProbability, 54
Example Code
    dockmolecules.java, 19

generatemultiplepose.java, 25
makereceptor.java, 28
posemolecules.java, 23
receptoroutercontourvolume.java, 29
rescoreposes.java, 21
setoutercontourvolume.java, 32
toggleinnercontour.java, 30

F

Fit
OEDocking::OEShapeFit, 79

G

generatemultiplepose.java
Example Code, 25
GetAllowedClashType
OEDocking::OEPositOptions, 50
GetAtom
OEDocking::OEProteinConstraint, 88
GetAutoConstraints
OEDocking::OEMakeReceptorOptions, 43
GetBestReceptorPoseOnly
OEDocking::OEPositOptions, 50
GetBoxExtension
OEDocking::OEMakeReceptorOptions, 43
GetBoxMol
OEDocking::OEMakeReceptorOptions, 43
GetColorForceField
OEModels::OEROCSQueryModelOptions, 106
GetComponentNames
OEDocking::OEDock, 39
OEDocking::OEScore, 56
GetDesignUnit
OEDocking::OESinglePoseResult, 59
GetEnabled
OEDocking::OEFeature, 88
OEDocking::OEProteinConstraint, 88
GetExcludeSelf
OEDocking::OEPositOptions, 50
GetFeatureName
OEDocking::OEFeature, 88
GetFeatures
OEDocking::OECustomConstraints, 85
GetFlexiOverlapOptions
OEDocking::OEShapeFitOptions, 80
GetFlexiOverlayOptions
OEDocking::OEPositOptions, 50
GetFlexRange
OEDocking::OEPoseOptimizerOptions, 45
GetForceField
OEDocking::OEPoseOptimizerOptions, 45
GetFullConformationSearch
OEDocking::OEPositOptions, 50
OEDocking::OEShapeFitOptions, 79
GetHasClash
OEDocking::OESinglePoseResult, 60
GetHighScoresAreBetter
OEDocking::OEDock, 39
OEDocking::OEPosit, 47
OEDocking::OEScore, 56
GetIgnoreNitrogenStereo
OEDocking::OEPositOptions, 51
GetInvalidScore
OEDocking::OEPosit, 47
GetMaxLocalStrain
OEDocking::OEShapeFitOptions, 80
GetMaxMoleculesPerModel
OEModels::OEROCSQueryModelOptions, 106
GetMaxOptSteps
OEDocking::OEShapeFitOptions, 80
GetMergeColorAtoms
OEModels::OEROCSQueryModelOptions, 106
GetMinProbability
OEDocking::OEPositOptions, 51
GetName
OEDocking::OEDock, 39
OEDocking::OEPosit, 47
OEDocking::OEProteinConstraint, 89
OEDocking::OEScore, 56
GetNegativeImageType
OEDocking::OEMakeReceptorOptions, 43
GetNonBondCutoff
OEDocking::OEPoseOptimizerOptions, 45
GetNumClashes
OEDocking::OESinglePoseResult, 60
GetNumContacts
OEDocking::OESinglePoseResult, 60
GetNumModels
OEModels::OEROCSQueryModelOptions, 106
GetNumPoses
OEDocking::OEPositResults, 55
GetPose
OEDocking::OEShapeFitResults, 82
OEDocking::OESinglePoseResult, 60
GetPoserCacheCount
OEDocking::OEPositOptions, 51
GetPoseRelaxMode
OEDocking::OEPositOptions, 51
GetPoseRelaxOptions
OEDocking::OEPositOptions, 51
GetPositMethod
OEDocking::OESinglePoseResult, 60
GetPositMethods
OEDocking::OEPositOptions, 51
GetProbability
OEDocking::OESinglePoseResult, 61
GetRad
OEDocking::OESphereBase, 90

GetReceptorIndex
    OEDocking::OESinglePoseResult, 61
GetRelaxAttempted
    OEDocking::OESinglePoseResult, 61
GetRelaxed
    OEDocking::OESinglePoseResult, 61
GetRelaxStatus
    OEDocking::OESinglePoseResult, 61
GetRelaxStatusStr
    OEDocking::OESinglePoseResult, 61
GetResolution
    OEDocking::OEDockOptions, 37
GetReturnCode
    OEDocking::OEPositResults, 55
GetRigidOverlayOptions
    OEDocking::OEShapeFitOptions, 80
GetROCSQueryModelOptions
    OEModels::OEROCSQueryModelBuilder, 105
GetScore
    OEDocking::OEShapeFitResults, 82
GetScoreMethod
    OEDocking::OEDockOptions, 37
GetSinglePoseResults
    OEDocking::OEPositResults, 55
GetSpheres
    OEDocking::OEFeature, 86
GetTargetMask
    OEDocking::OEMakeReceptorOptions, 43
    OEDocking::OESinglePoseResult, 60
GetTargetPred
    OEDocking::OEMakeReceptorOptions, 43
    OEDocking::OESinglePoseResult, 60
GetTorLib
    OEDocking::OEPositOptions, 51
    OEDocking::OEShapeFitOptions, 80
GetType
    OEDocking::OEProteinConstraint, 88
GetX
    OEDocking::OESphereBase, 89
GetXMax
    OEDocking::OEBoxBase, 83
GetXMin
    OEDocking::OEBoxBase, 83
GetY
    OEDocking::OESphereBase, 90
GetYMax
    OEDocking::OEBoxBase, 83
GetYMin
    OEDocking::OEBoxBase, 83
GetZ
    OEDocking::OESphereBase, 90
GetZMax
    OEDocking::OEBoxBase, 83
GetZMin
    OEDocking::OEBoxBase, 84

I

Initialize
    OEDocking::OEDock, 38
    OEDocking::OEPosit, 48
    OEDocking::OEScore, 56
IsInitialized
    OEDocking::OEDock, 38
    OEDocking::OEPosit, 48
    OEDocking::OEScore, 55

M

makereceptor.java
    Example Code, 28
MethodChoice
    OEDocking::OEPosit, 48

N

NumFeatures
    OEDocking::OECustomConstraints, 86
NumSmarts
    OEDocking::OEFeature, 87
NumSpheres
    OEDocking::OEFeature, 87

O

OEDocking::OEAddDockingInteractions, 70
OEDocking::OEAlignReceptorFunction, 82
    CreateCopy, 82
    operator(), 82
    SetupTarget, 82
OEDocking::OEAlignReceptors, 90
OEDocking::OEAllowedClashType, 61
OEDocking::OEAllowedClashType::ANY, 62
OEDocking::OEAllowedClashType::HYDROGEN, 62
OEDocking::OEAllowedClashType::NONE, 61
OEDocking::OEBox, 84
    Constructors, 84
OEDocking::OEBoxArea, 91
OEDocking::OEBoxBase, 82
    Destructor, 83
    GetXMax, 83
    GetXMin, 83
    GetYMax, 83
    GetYMin, 83
    GetZMax, 83
    GetZMin, 84
    Setup, 84
OEDocking::OEBoxExtend, 91
OEDocking::OEBoxTranslate, 91
OEDocking::OEBoxVolume, 91
OEDocking::OEBoxXDim, 91

OEDocking::OEBoxXMid, 91
OEDocking::OEBoxYDim, 91
OEDocking::OEBoxYMid, 92
OEDocking::OEBoxZDim, 92
OEDocking::OEBoxZMid, 92
OEDocking::OECreateDUFromReceptorMol, 70
OEDocking::OECustomConstraints, 85
    AddFeature, 85
    Clear, 86
    Constructors, 85
    DeleteFeature, 85
    GetFeatures, 85
    NumFeatures, 86
OEDocking::OEDock, 38
    AnnotatePose, 41
    CacheScoringSetup, 41
    Constructor, 38
    DockMultiConformerMolecule, 38
    GetComponentNames, 39
    GetHighScoresAreBetter, 39
    GetName, 39
    Initialize, 38
    IsInitialized, 38
    ScoreAtom, 40
    ScoreAtomComponent, 40
    ScoreLigand, 40
    ScoreLigandComponent, 40
OEDocking::OEDockingGetArch, 70
OEDocking::OEDockingGetLicensee, 70
OEDocking::OEDockingGetPlatform, 71
OEDocking::OEDockingGetRelease, 71
OEDocking::OEDockingGetSite, 71
OEDocking::OEDockingGetVersion, 71
OEDocking::OEDockingReturnCode, 62
OEDocking::OEDockingReturnCode::Aborted, 62
OEDocking::OEDockingReturnCode::ConformerGenError, 62
OEDocking::OEDockingReturnCode::CoordError, 62
OEDocking::OEDockingReturnCode::EmptyLigand, 62
OEDocking::OEDockingReturnCode::EmptyProtein, 62
OEDocking::OEDockingReturnCode::Failure, 62
OEDocking::OEDockingReturnCode::GridSetupError, 62
OEDocking::OEDockingReturnCode::InvalidScore, 63
OEDocking::OEDockingReturnCode::NoConstraintMatch, 63
OEDocking::OEDockingReturnCode::NotInitialized, 63
OEDocking::OEDockingReturnCode::NoValidNonClashPoses, 63
OEDocking::OEDockingReturnCode::NoValidPoses, 63
OEDocking::OEDockingReturnCode::OptimizationError, 63
OEDocking::OEDockingReturnCode::OutsideGrid, 63
OEDocking::OEDockingReturnCode::ScoreError, 63
OEDocking::OEDockingReturnCode::Success, 63
OEDocking::OEDockingReturnCode::TypingError, 64
OEDocking::OEDockingReturnCodeGetName, 71
OEDocking::OEDockMethod, 64
OEDocking::OEDockMethod::Chemgauss, 64
OEDocking::OEDockMethod::Chemgauss3, 64
OEDocking::OEDockMethod::Chemgauss4, 64
OEDocking::OEDockMethod::Chemscore, 65
OEDocking::OEDockMethod::Default, 65
OEDocking::OEDockMethod::Hybrid, 65
OEDocking::OEDockMethod::Hybrid1, 65
OEDocking::OEDockMethod::Hybrid2, 65
OEDocking::OEDockMethod::INVALID, 65
OEDocking::OEDockMethod::PLP, 64
OEDocking::OEDockMethod::Shapegauss, 64
OEDocking::OEDockMethodConfigure, 71
OEDocking::OEDockMethodGetName, 71
OEDocking::OEDockMethodGetValue, 72
OEDocking::OEDockOptions, 37
    Constructor, 37
    GetResolution, 37
    GetScoreMethod, 37
    SetResolution, 37
    SetScoreMethod, 37
OEDocking::OEFeature, 86
    AddSmarts, 87
    AddSphere, 86
    ClearSmarts, 87
    ClearSpheres, 87
    CreateCopy, 88
    DeleteSmarts, 87
    DeleteSphere, 87
    Destructor, 86
    GetEnabled, 88
    GetFeatureName, 88
    GetSpheres, 86
    NumSmarts, 87
    NumSpheres, 87
    SetEnabled, 88
    SetFeatureName, 87
OEDocking::OEGetEstimatedPositProbability, 72
OEDocking::OEGetPositMethod, 72
OEDocking::OEHasClash, 72
OEDocking::OEHybrid, 41
    Constructor, 42

OEDocking::OEInBox, 92
OEDocking::OEInSphere, 92
OEDocking::OEIsReceptor, 92
OEDocking::OEMakeBoxMolecule, 72
OEDocking::OEMakeReceptor, 73
OEDocking::OEMakeReceptorDeprecated, 92
OEDocking::OEMakeReceptorOptions, 42
    Constructor, 42
    GetAutoConstraints, 43
    GetBoxExtension, 43
    GetBoxMol, 43
    GetNegativeImageType, 43
    GetTargetMask, 43
    GetTargetPred, 43
    SetAutoConstraints, 43
    SetBoxExtension, 43
    SetBoxMol, 44
    SetNegativeImageType, 44
    SetTargetMask, 44
    SetTargetPred, 44
OEDocking::OENegativeImageType, 65
OEDocking::OENegativeImageType::Default, 66
OEDocking::OENegativeImageType::INVALID, 66
OEDocking::OENegativeImageType::LargeShape, 65
OEDocking::OENegativeImageType::SmallShape, 66
OEDocking::OENegativeImageType::StandardShape, 65
OEDocking::OENegativeImageTypeConfigure, 95
OEDocking::OENegativeImageTypeGetName, 73
OEDocking::OENegativeImageTypeGetValue, 73
OEDocking::OEPoseOptimizer, 44
    Constructor, 44
    Optimize, 44
OEDocking::OEPoseOptimizerOptions, 45
    Constructor, 45
    GetFlexRange, 45
    GetForceField, 45
    GetNonBondCutoff, 45
    SetFlexRange, 45
    SetForceField, 45
    SetNonBondCutoff, 46
OEDocking::OEPoseRelaxMode, 66
OEDocking::OEPoseRelaxMode::ALL, 66
OEDocking::OEPoseRelaxMode::CLASHED, 66
OEDocking::OEPoseRelaxMode::NONE, 66
OEDocking::OEPosit, 46
    AddReceptor, 46
    ClearReceptors, 47
    Constructor, 46
    Dock, 47
    DockMultiConformerMolecule, 48
    GetHighScoresAreBetter, 47
    GetInvalidScore, 47
    GetName, 47
    Initialize, 48
    IsInitialized, 48
    MethodChoice, 48
    RankDesignUnits, 48
    RankReceptors, 49
    ScoreLigand, 49
OEDocking::OEPositMethod, 66
OEDocking::OEPositMethod::ALL, 66
OEDocking::OEPositMethod::FRED, 66
OEDocking::OEPositMethod::HYBRID, 67
OEDocking::OEPositMethod::SHAPEFIT, 67
OEDocking::OEPositMethod::UNKNOWN, 67
OEDocking::OEPositMethodGetName, 73
OEDocking::OEPositOptions, 49
    Constructor, 50
    GetAllowedClashType, 50
    GetBestReceptorPoseOnly, 50
    GetExcludeSelf, 50
    GetFlexiOverlayOptions, 50
    GetFullConformationSearch, 50
    GetIgnoreNitrogenStereo, 51
    GetMinProbability, 51
    GetPoserCacheCount, 51
    GetPoseRelaxMode, 51
    GetPoseRelaxOptions, 51
    GetPositMethods, 51
    GetTorLib, 51
    SetAllowedClashType, 51
    SetBestReceptorPoseOnly, 52
    SetExcludeSelf, 52
    SetFlexiOverlayOptions, 52
    SetFullConformationSearch, 52
    SetIgnoreNitrogenStereo, 52
    SetMinProbability, 52
    SetPoserCacheCount, 53
    SetPoseRelaxMode, 53
    SetPoseRelaxOptions, 53
    SetPositMethods, 53
    SetTorLib, 53
OEDocking::OEPositProbability, 53
    Constructor, 54
    Estimate, 54
    PoseEstimate, 54
    SetupRef, 54
OEDocking::OEPositResults, 54
    Constructors, 54
    GetNumPoses, 55
    GetReturnCode, 55
    GetSinglePoseResults, 55
OEDocking::OEProteinConstraint, 88
    GetAtom, 88
    GetEnabled, 88

GetName, 89
GetType, 88
SetAtom, 89
SetEnabled, 89
SetName, 89
SetType, 89
OEDocking::OEProteinConstraintType, 67
OEDocking::OEProteinConstraintType::Acceptor, 67
OEDocking::OEProteinConstraintType::Chelator, 67
OEDocking::OEProteinConstraintType::Contact, 67
OEDocking::OEProteinConstraintType::Donor, 67
OEDocking::OEProteinConstraintType::Lipophilic, 67
OEDocking::OEProteinConstraintType::Unknown, 67
OEDocking::OEReadReceptorFile, 95
OEDocking::OEReadReceptorFromByteArray, 95
OEDocking::OEReceptorAddCachedScore, 96
OEDocking::OEReceptorAddExtraMol, 96
OEDocking::OEReceptorClear, 96
OEDocking::OEReceptorClearBoundLigand, 97
OEDocking::OEReceptorClearCachedScore, 97
OEDocking::OEReceptorClearCustomConstraints, 97
OEDocking::OEReceptorClearExtraMols, 97
OEDocking::OEReceptorClearProteinConstraint, 97
OEDocking::OEReceptorClearProteinConstraints, 97
OEDocking::OEReceptorGetBoundLigand, 97
OEDocking::OEReceptorGetCustomConstraints, 97
OEDocking::OEReceptorGetExtraMols, 98
OEDocking::OEReceptorGetInnerContourLevel, 98
OEDocking::OEReceptorGetNegativeImageGrid, 98
OEDocking::OEReceptorGetOuterContourLevel, 98
OEDocking::OEReceptorGetProteinConstraints, 98
OEDocking::OEReceptorHasBoundLigand, 98
OEDocking::OEReceptorHasCachedScore, 99
OEDocking::OEReceptorHasCustomConstraints, 99
OEDocking::OEReceptorHasExtraMols, 99
OEDocking::OEReceptorHasInnerContourLevel, 99
OEDocking::OEReceptorHasNegativeImageGrid, 99
OEDocking::OEReceptorHasOuterContourLevel, 99
OEDocking::OEReceptorHasProteinConstraints, 99
OEDocking::OEReceptorSetBoundLigand, 100
OEDocking::OEReceptorSetCachedScore, 100
OEDocking::OEReceptorSetCustomConstraints, 100
OEDocking::OEReceptorSetInnerContourLevel, 100
OEDocking::OEReceptorSetOuterContourLevel, 100
OEDocking::OEReceptorSetProteinConstraint, 101
OEDocking::OEScore, 55
AnnotatePose, 58
CacheScoringSetup, 58
Constructor, 55
GetComponentNames, 56
GetHighScoresAreBetter, 56
GetName, 56
Initialize, 56
IsInitialized, 55
ScoreAtom, 56
ScoreAtomComponent, 57
ScoreLigand, 56
ScoreLigandComponent, 57
SystematicSolidBodyOptimize, 57
OEDocking::OEScoreType, 68
OEDocking::OEScoreType::Chemgauss, 68
OEDocking::OEScoreType::Chemgauss3, 68
OEDocking::OEScoreType::Chemgauss4, 68
OEDocking::OEScoreType::Chemscore, 68
OEDocking::OEScoreType::Default, 68
OEDocking::OEScoreType::INVALID, 68
OEDocking::OEScoreType::PLP, 68
OEDocking::OEScoreType::Shapegauss, 68
OEDocking::OEScoreTypeConfigure, 101
OEDocking::OEScoreTypeGetName, 73
OEDocking::OEScoreTypeGetValue, 73
OEDocking::OESearchResolution, 68
OEDocking::OESearchResolution::Default, 70
OEDocking::OESearchResolution::High, 69
OEDocking::OESearchResolution::INVALID, 70
OEDocking::OESearchResolution::Low, 69
OEDocking::OESearchResolution::Standard, 69
OEDocking::OESearchResolutionConfigure, 101
OEDocking::OESearchResolutionGetName, 74
OEDocking::OESearchResolutionGetValue, 74
OEDocking::OESetEnergyScore, 74
OEDocking::OESetEnergyScoreComponent, 75
OEDocking::OESetScore, 76
OEDocking::OESetScoreComponent, 77
OEDocking::OESetSDScore, 75
OEDocking::OESetSDScoreComponent, 76
OEDocking::OESetupBox, 77, 102
OEDocking::OESetupBoxCenterAndExtents, 103
OEDocking::OEShapeFit, 78
Constructor, 78
Fit, 79
SetupRef, 78
OEDocking::OEShapeFitOptions, 79

Constructor, 79
CreateCopy, 79
GetFlexiOverlapOptions, 80
GetFullConformationSearch, 79
GetMaxLocalStrain, 80
GetMaxOptSteps, 80
GetRigidOverlayOptions, 80
GetTorLib, 80
SetFlexiOverlapOptions, 81
SetFullConformationSearch, 80
SetMaxLocalStrain, 80
SetMaxOptSteps, 80
SetRigidOverlayOptions, 81
SetTorLib, 81
OEDocking::OEShapeFitResults, 81
Constructor, 81
GetPose, 82
GetScore, 82
OEDocking::OESinglePoseResult, 59
Constructors, 59
GetDesignUnit, 59
GetHasClash, 60
GetNumClashes, 60
GetNumContacts, 60
GetPose, 60
GetPositMethod, 60
GetProbability, 61
GetReceptorIndex, 61
GetRelaxAttempted, 61
GetRelaxed, 61
GetRelaxStatus, 61
GetRelaxStatusStr, 61
GetTargetMask, 60
GetTargetPred, 60
OEDocking::OESphereArea, 103
OEDocking::OESphereBase, 89
Destructor, 89
GetRad, 90
GetX, 89
GetY, 90
GetZ, 90
SetCenter, 90
SetRad, 90
OEDocking::OESphereVolume, 103
OEDocking::OEWriteReceptorFile, 103
OEDocking::OEWriteReceptorToByteArray, 103
OEModels::OEROCSQueryModelBuilder, 104
AddLigand, 105
Build, 105
ClearLigands, 105
Constructor, 104
GetROCSQueryModelOptions, 105
SetLigands, 105
OEModels::OEROCSQueryModelOptions, 105
Constructor, 106
CreateCopy, 106
GetColorForceField, 106
GetMaxMoleculesPerModel, 106
GetMergeColorAtoms, 106
GetNumModels, 106
SetColorForceField, 107
SetMaxMoleculesPerModel, 107
SetMergeColorAtoms, 107
SetNumModels, 107
operator()
OEDocking::OEAlignReceptorFunction, 82
Optimize
OEDocking::OEPoseOptimizer, 44

P

PoseEstimate
OEDocking::OEPositProbability, 54
posemolecules.java
Example Code, 23

R

RankDesignUnits
OEDocking::OEPosit, 48
RankReceptors
OEDocking::OEPosit, 49
receptoroutercontourvolume.java
Example Code, 29
rescoreposes.java
Example Code, 21

S

ScoreAtom
OEDocking::OEDock, 40
OEDocking::OEScore, 56
ScoreAtomComponent
OEDocking::OEDock, 40
OEDocking::OEScore, 57
ScoreLigand
OEDocking::OEDock, 40
OEDocking::OEPosit, 49
OEDocking::OEScore, 56
ScoreLigandComponent
OEDocking::OEDock, 40
OEDocking::OEScore, 57
SetAllowedClashType
OEDocking::OEPositOptions, 51
SetAtom
OEDocking::OEProteinConstraint, 89
SetAutoConstraints
OEDocking::OEMakeReceptorOptions, 43
SetBestReceptorPoseOnly
OEDocking::OEPositOptions, 52
SetBoxExtension

OEDocking::OEMakeReceptorOptions, 43
SetBoxMol
OEDocking::OEMakeReceptorOptions, 44
SetCenter
OEDocking::OESphereBase, 90
SetColorForceField
OEModels::OEROCSQueryModelOptions, 107
SetEnabled
OEDocking::OEFeature, 88
OEDocking::OEProteinConstraint, 89
SetExcludeSelf
OEDocking::OEPositOptions, 52
SetFeatureName
OEDocking::OEFeature, 87
SetFlexiOverlapOptions
OEDocking::OEShapeFitOptions, 81
SetFlexiOverlayOptions
OEDocking::OEPositOptions, 52
SetFlexRange
OEDocking::OEPoseOptimizerOptions, 45
SetForceField
OEDocking::OEPoseOptimizerOptions, 45
SetFullConformationSearch
OEDocking::OEPositOptions, 52
OEDocking::OEShapeFitOptions, 80
SetIgnoreNitrogenStereo
OEDocking::OEPositOptions, 52
SetLigands
OEModels::OEROCSQueryModelBuilder, 105
SetMaxLocalStrain
OEDocking::OEShapeFitOptions, 80
SetMaxMoleculesPerModel
OEModels::OEROCSQueryModelOptions, 107
SetMaxOptSteps
OEDocking::OEShapeFitOptions, 80
SetMergeColorAtoms
OEModels::OEROCSQueryModelOptions, 107
SetMinProbability
OEDocking::OEPositOptions, 52
SetName
OEDocking::OEProteinConstraint, 89
SetNegativeImageType
OEDocking::OEMakeReceptorOptions, 44
SetNonBondCutoff
OEDocking::OEPoseOptimizerOptions, 46
SetNumModels
OEModels::OEROCSQueryModelOptions, 107
setoutercontourvolume.java
Example Code, 32
SetPoserCacheCount
OEDocking::OEPositOptions, 53
SetPoseRelaxMode
OEDocking::OEPositOptions, 53
SetPoseRelaxOptions
OEDocking::OEPositOptions, 53
SetPositMethods
OEDocking::OEPositOptions, 53
SetRad
OEDocking::OESphereBase, 90
SetResolution
OEDocking::OEDockOptions, 37
SetRigidOverlayOptions
OEDocking::OEShapeFitOptions, 81
SetScoreMethod
OEDocking::OEDockOptions, 37
SetTargetMask
OEDocking::OEMakeReceptorOptions, 44
SetTargetPred
OEDocking::OEMakeReceptorOptions, 44
SetTorLib
OEDocking::OEPositOptions, 53
OEDocking::OEShapeFitOptions, 81
SetType
OEDocking::OEProteinConstraint, 89
Setup
OEDocking::OEBoxBase, 84
SetupRef
OEDocking::OEPositProbability, 54
OEDocking::OEShapeFit, 78
SetupTarget
OEDocking::OEAlignReceptorFunction, 82
SystematicSolidBodyOptimize
OEDocking::OEScore, 57

T

toggleinnercontour.java
Example Code, 30