

OpenEye
SCIENTIFIC

OpenEye Toolkits Release Notes – Java

Release 2026.1.0

Cadence Design Systems, Inc.

August 06, 2026

CONTENTS

1 Copyright and Trademarks **1**

2 Sample Code **3**

3 Citation **5**
- 3.1 Orion® . . . 5
- 3.2 Orion Floes . . . 5
- 3.3 Toolkits and Applications . . . 5
- 3.4 Publications for Bibliographies . . . 7
- 3.5 OpenEye MMDS Web Service . . . 8

4 Technology Licensing **9**
- 4.1 GCC . . . 81

5 OEToolkits 2026.1 **97**
- 5.1 Release Highlights 2026.1 . . . 97
- 5.2 General Notices . . . 102

6 Detailed Release Notes 2026.1 **103**
- 6.1 Bioisostere TK 5.0.0 . . . 103
- 6.2 OEChem TK 5.0.0 . . . 104
- 6.3 OEBio TK . . . 106
- 6.4 OEDepict TK 2.7.0 . . . 109
- 6.5 OEDocking TK 4.3.5 . . . 109
- 6.6 EON TK 3.1.3 . . . 110
- 6.7 FastROCS TK 2.3.2 . . . 110
- 6.8 Grapheme™ TK 1.5.5 . . . 110
- 6.9 GraphSim TK 2.7.1 . . . 111
- 6.10 Lexichem TK 2.9.5 . . . 111
- 6.11 OEMedChem TK 1.2.6 . . . 111
- 6.12 MolProp TK 2.7.0 . . . 111
- 6.13 OEFF TK 2.9.2 . . . 111
- 6.14 Omega TK 7.0.0 . . . 111
- 6.15 Quacpac TK 2.3.0 . . . 112
- 6.16 Saiph TK 1.0.1 . . . 112
- 6.17 Shape TK 3.10.0 . . . 112
- 6.18 SiteHopper TK 2.1.4 . . . 113
- 6.19 Spicoli TK 1.6.4 . . . 113
- 6.20 Spruce TK 1.9.0 . . . 113
- 6.21 Szmap TK 1.7.4 . . . 113
- 6.22 Szybki TK 2.9.2 . . . 113

6.23 Zap TK 2.5.3 . . . 113

7 Recent Release History **115**
7.1 OEToolkits 2025.2 . . . 115
7.2 OEToolkits 2025.1 . . . 118
7.3 OEToolkits 2024.2 . . . 120
7.4 OEToolkits 2024.1 . . . 125

8 Previous Release History **131**
8.1 OEToolkits 2023.2 . . . 131
8.2 OEToolkits 2023.1 . . . 133
8.3 Release Highlights 2022.2 . . . 134
8.4 Release Highlights 2022.1 . . . 138
8.5 Release Highlights 2021.2 . . . 142
8.6 Release Highlights 2021.1 . . . 144
8.7 Release Highlights 2020.2 . . . 148
8.8 Release Highlights 2020.1 . . . 153
8.9 OEToolkits 2019.Oct . . . 158
8.10 OEToolkits 2019.Apr . . . 160
8.11 OEToolkits 2018.Oct . . . 162
8.12 OEToolkits 2018.Feb . . . 164
8.13 OEToolkits 2017.Oct . . . 165
8.14 OEToolkits 2017.Jun . . . 167
8.15 OEToolkits 2017.Feb . . . 169
8.16 OEToolkits 2016.Oct . . . 171
8.17 OEToolkits 2016.Jun . . . 173
8.18 OEToolkits 2016.Feb . . . 175
8.19 OEToolkits 2015.Oct . . . 178
8.20 OEToolkits 2015.Jun . . . 181
8.21 OEToolkits 2015.Feb . . . 184
8.22 OEToolkits 2014.Oct . . . 187
8.23 OEToolkits 2014.Jun . . . 190
8.24 OEToolkits 2014.Feb . . . 191
8.25 OEToolkits 2013.Oct . . . 192
8.26 OEToolkits 2013.Jun . . . 193
8.27 OEToolkits 2013.Feb . . . 194
8.28 OEToolkits 2012.Oct . . . 195
8.29 OEToolkits 2012.Jun . . . 196
8.30 OEToolkits 2012.Feb . . . 197
8.31 OEToolkits 2011.Oct . . . 198
8.32 OEToolkits 2011.1 . . . 199
8.33 OEToolkits 1.7.4 . . . 201
8.34 OEToolkits 1.7.2 . . . 202
8.35 OEToolkits 1.7.1 . . . 203
8.36 OEToolkits 1.7.0 . . . 205

CHAPTER

ONE

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

TWO

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

THREE

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.

3.1 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.

3.2 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.

3.3 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.

3.4 Publications for Bibliographies

Please also cite these publications if the corresponding product was used in your research.

3.4.1 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

3.4.2 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

3.4.3 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

3.4.4 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

3.4.5 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

3.5 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

FOUR

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-runtim	https://github.com/conda-forge/antlr4-python3-runtim/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-stream	https://github.com/conda/conda-package-stream/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-framework.org/	https://github.com/encode/django-rest-framework 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-file#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
tifffile	https://github.com/cgohlke/tifffile/	https://github.com/cgohlke/tifffile/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

4.1 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.

4.1.1 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

4.1.2 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

CHAPTER

FIVE

OETOOLKITS 2026.1

5.1 Release Highlights 2026.1

5.1.1 OEChem TK: Peptide Informatics Support

This release delivers a comprehensive set of peptide informatics capabilities in **OEChem TK**, including reliable conversion between molecular structures and HELM representations. Support for custom monomer dictionaries enables seamless incorporation of proprietary nonstandard amino acids into peptide discovery workflows. The release also includes a built-in monomer dictionary containing more than 260 monomers commonly encountered in peptide chemistry projects.

In addition, enhanced visualization capabilities in **Grapheme TK** enable clear and intuitive representation of complex peptides, facilitating the interpretation and communication of diverse peptide chemistries.

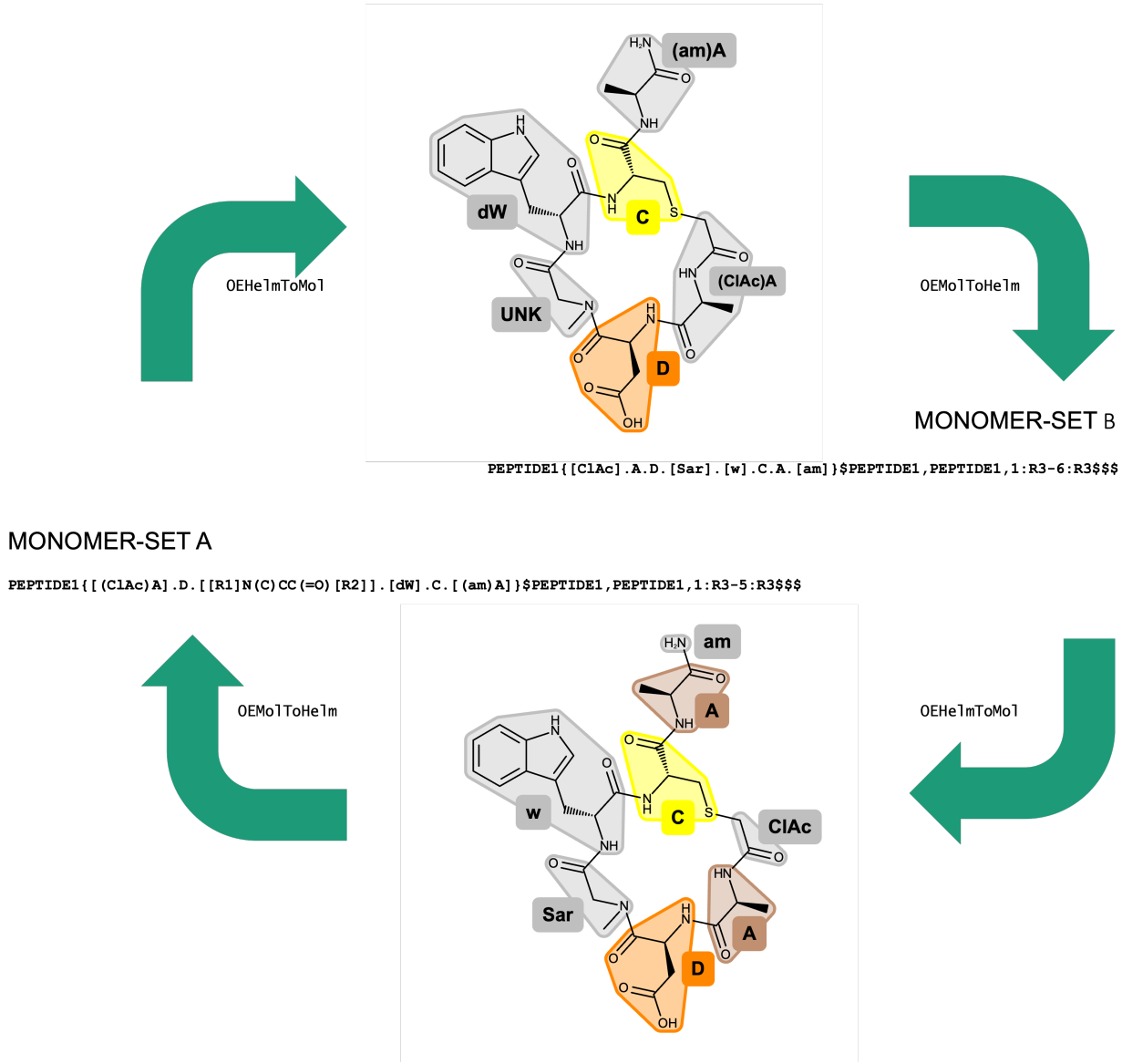


Fig. 1: **Figure 1.** Conversion between HELM strings generated with different monomer sets.

5.1.2 OMEGA: Torsion Driving with Thompson Sampling

Conformer generation using **OMEGA** with torsion driving now automatically uses Thompson sampling. The algorithm for Thompson sampling in **OMEGA** has been optimized for performance and extended to the generation of larger ensembles, such as those used in the `classic` and `pose` modes of **OMEGA**.

Thompson sampling is a Bayesian approach that drives conformer generation toward low energy structures, enabling conformer generation at speed without sacrificing accuracy. In general, the speed increase with Thompson sampling is larger when working with larger molecules and generating smaller ensembles. On a dataset of ~40K compounds, compiled from multiple vendor databases consisting of molecules of varied sizes, an overall 2x speedup over torsion driving without Thompson sampling is observed when generating conformers in the `classic` mode. Conformer generation for the same dataset using the `fastrocs` mode provides a 3x speedup over torsion driving without Thompson sampling. A comparison of timings for the dataset of the ~40K compounds from multiple vendor databases is shown in Figure 2.

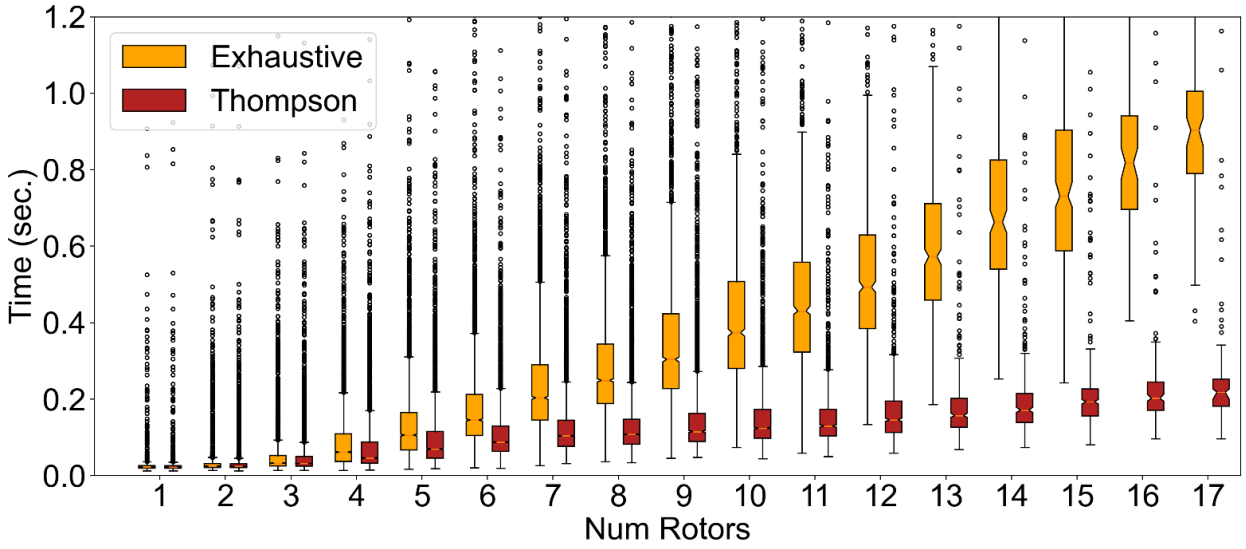


Fig. 2: **Figure 2.** Comparison of **OMEGA** performance in the `classic` mode between exhaustive and Thompson sampling. The box plots show the distribution of processing time binned by rotor count.

OMEGA maintains high accuracy while achieving this performance gain. As seen in Figure 3, both exhaustive and Thompson sampling methods in the `classic` mode reproduce conformations within 1 Å RMSD of the experimental conformation for 80% of ligands across the 6,000 PDB structures with HT (Highly Trustworthy) Iridium scores. Structure-based virtual screening performance is equivalent to that obtained from exhaustive sampling.

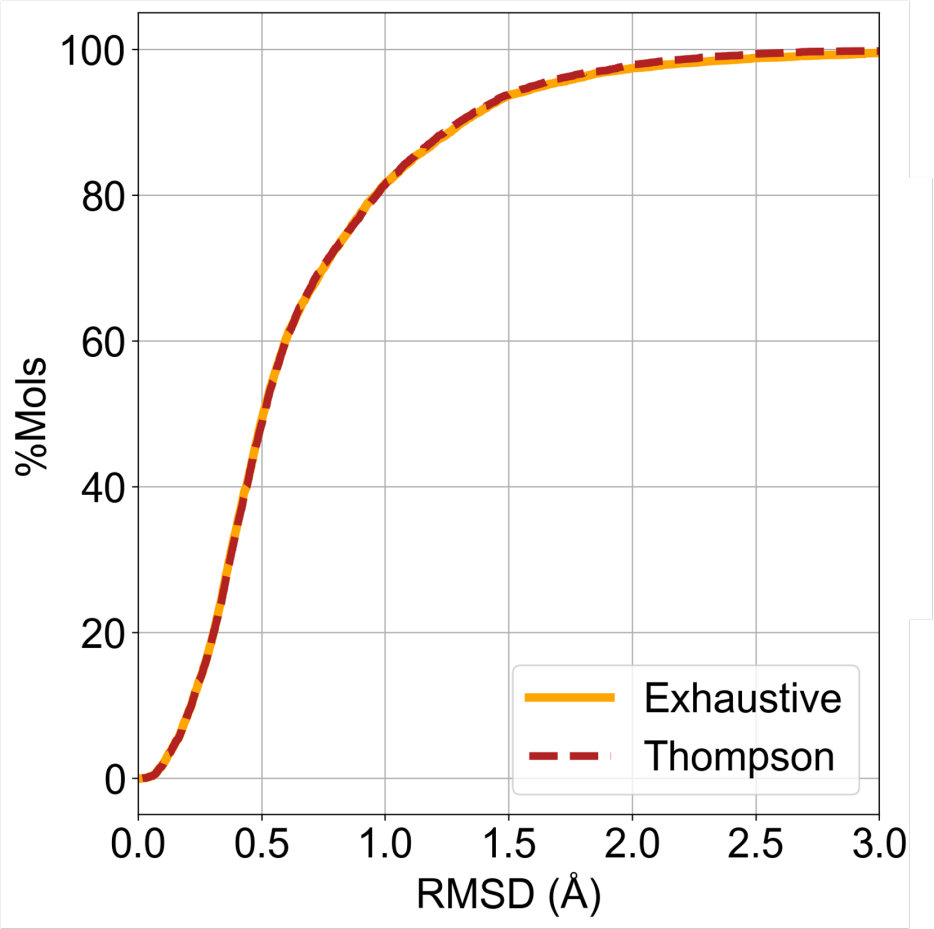


Fig. 3: **Figure 3.** Distribution of RMSD between the best pose from the generated ensemble and the crystal pose for the two sampling methods using `classic` mode in **OMEGA**.

5.1.3 Bioisostere TK: Multiquery Replacements

Bioisostere TK now includes new API functionality for simultaneous replacement of multiple query fragments in a molecule. With the addition of the new functionality, **Bioisostere TK** is uniquely positioned with tools to build applications for 3D generative de novo molecule design.

In addition to the ability to simultaneously replace multiple query fragments, tools have been added to obtain scores of multiple replacements from their single replacement counterparts, building a single hit list of **Brood** overlay scores obtained from multiple query replacements, building hit lists combining hits from different queries and multiple queries, and building clusters from the combined hit lists. All of the API designs are compatible for use with an arbitrary number of queries on a molecule, without any limitations. With the addition of each query, the searchable space can be expanded combinatorially, enabling exploration of increasingly larger chemical design spaces.

As shown in Figure 4, users can select specific fragments in a molecule of interest and generate targeted replacement suggestions.

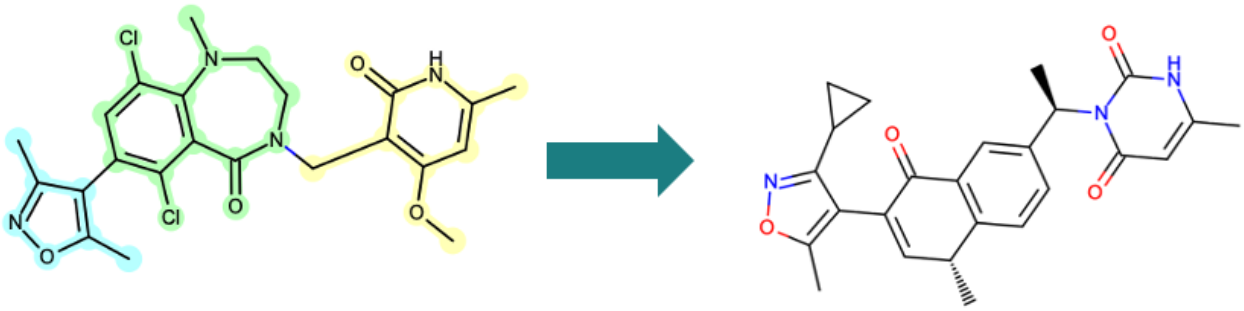


Fig. 4: **Figure 4.** De novo molecule design of an EZH2 inhibitor. The compound on the left highlights the input molecule being divided into three fragments to build three separate **Brood** queries. The compound on the right is a new molecule designed using **Bioisostere TK** to replace all three queries from the left.

5.1.4 Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.11 - 3.14	RHEL8/9/10, Ubuntu22/22-ARM/24	Win11	14, 15, 26
C++		RHEL8/9/10, Ubuntu22/22-ARM/24	Win11 (VS2022)	14, 15, 26
Java	8, 11, 21	RHEL8/9/10, Ubuntu22/22-ARM/24	Win11	14, 15, 26
C#			Win11 (VS2022)	

5.1.5 Related TK Versions

Toolkit	Version
Bioisostere TK	5.0.0
OEChem TK	5.0.0
OEDepict TK	2.7.0
OEDocking TK	4.3.5
Eon TK	3.1.3
FastROCS TK	2.3.2
OEFF TK	2.9.2
Grapheme TK	1.5.5
GraphSim TK	2.7.1
Lexichem TK	2.9.5
OEMedChem TK	1.2.6
MolProp TK	2.7.0
Omega TK	7.0.0
Quacpac TK	2.3.0
Saiph TK	1.0.1
Shape TK	3.10.0
SiteHopper TK	2.1.4
Spicoli TK	1.6.4
Spruce TK	1.9.0
Szmap TK	1.7.4
Szybki TK	2.9.2
Zap TK	2.5.3

5.2 General Notices

- This is the last release of the OpenEye Java and C# toolkits. Existing versions, including the current release versions, will continue to be available.
- Python Support:
  - OpenEye Python Cookbook is now available as a pip-installable package.
  - Support for Python 3.14 has been added in this release for all supported platforms.
  - Support for Python 3.10 has been dropped in this release.
  - Support for Python 3.11 will be dropped in the winter of 2027.
- Linux Support:
  - Support for Red Hat 10 has been added in this release.
  - Support for Ubuntu 26 will be added in the winter of 2027.
  - Support for Ubuntu 22 will be dropped in the winter of 2027.
- macOS Support:
  - Support for macOS 26 has been added in this release.
  - Support for macOS 13 has been dropped.
  - Support for all Intel-based Macs will be dropped in the winter of 2027.
- Support for Windows 10 has been dropped in this release.

CHAPTER

SIX

DETAILED RELEASE NOTES 2026.1

6.1 Bioisostere TK 5.0.0

6.1.1 New Features

- The following new classes have been added as preliminary API:
  - OEBroodBuildResult
  - OEBroodCluster
  - OEBroodClusterBuilder
  - OEBroodComboBuilder
  - OEHitlistBuilderOptions
  - OEHitlistBase
  - OEScoreHitlist
  - OEBroodHitlist
  - OEMolCTBuilder
  - OESimilarity
- The following new methods have been added to OEBroodQuery:
  - GetQueryID
  - SetQueryID
- A new constructor overload has been added to OEHitlistBuilder that takes a OEHitlistBuilderOptions as an argument.
- A new overload of Setup has been added to OEBroodDBFilter that takes a molecule as argument.
- The following new methods have been added to OEBroodQuery:
  - GetBuildRes
  - GetScores
- The following new methods have been added to OEBroodScore:
  - Add
  - GetQueryIDs
- The following new functions have been added:
  - OECreateFlipperOptions

  - OECreateFragFilter
  - OEIsFragment

- The following methods have been removed from the OEHitlistBuilder, as they are no longer relevant:
  - GetTrimmedCount
  - TrimHitlist
  - An overload of GetHits that took a vector of OEBroodHit as an argument.
  - GetRemovedHits

6.1.2 Minor Bug Fixes

- The OECreateBroodQuery method now keeps any incoming color atoms associated with the query selection, when available, and does not add any additional color atoms to the query. The behavior is unchanged when no color atoms are present with the incoming selection.
- A bug in fragment property filtering has been resolved. Now by setting the SetUseProperty option in OEBroodScoreOptions to true, you can activate the fragment filtering based on the properties defined by OEExtrinsicPropOptions and OEIntrinsicPropOptions. Previously, the filter was not being applied correctly, which could lead to unexpected fragments being included in the results.
- An issue has been fixed related to proper usage of SetRangeSize and SetRangeOffset so that fragment retrieval from the database during the overlay step now properly respects the specified range boundaries.
- A bug has been fixed where the color score reported by GetColorScore could exceed its valid range in ROCS mode, causing the combined score to exceed its range as well. The color Tanimoto is now computed consistently from the scored conformer.

6.2 OEChem TK 5.0.0

6.2.1 New Features

- The following preliminary APIs have been added for monomer, HELM, and peptide support.

  Classes:

  - `OEMonomer` — represents a monomer in a monomer set.
  - `OEMonomerData` — stores the definition of a monomer.
  - `OEMonomerSet` — a collection of monomers.
  - `OEMonomerValidationResult` — stores the result of monomer data validation.
  - `OEHelmParsingResult` — stores the result of HELM parsing.
  - `OEHelmGenerationResult` — stores the result of HELM generation.
  - `OEHelmGenerationOptions` — gives options for HELM generation.

  Predicates:

  - `OEIsInMonomerCodeSet` — filters monomers by code set.
  - `OEHasPolymerType` — filters monomers by polymer type.
  - `OEHasMonomerType` — filters monomers by monomer type.
  - `OEHasAminoAcidType` — filters monomers by amino acid type.
  - `OEIsMonomerGroup` — identifies groups that store monomer information.

- `OEHasMonomerGroup` — identifies groups that match a monomer group by code set and code.

Constants:

- `OEAminoAcidType` — amino acid type constants.
- `OEPolymerType` — polymer type constants.
- `OEMonomerType` — monomer type constants.
- `OEMonomerValidationReturnCode` — monomer validation return codes.
- `OEHelmParsingReturnCode` — HELM parsing return codes.
- `OEHelmGenerationReturnCode` — HELM generation return codes.
- `OEGroupType.Monomer` — monomer group type constant.
- `OEProperty.Monomer` — monomer property type constant.

Monomer data and validation functions:

- `OEIsValidMonomerData`
- `OEAminoAcidTypeToString`
- `OEPolymerTypeToString`
- `OEStringToPolymerType`
- `OEMonomerTypeToString`
- `OEStringToMonomerType`

Monomer set I/O functions:

- `OEReadMonomerSet`
- `OEWriteMonomerSet`
- `OELoadStandardMonomerSet`
- `OELoadOpenEyeMonomerSet`

Monomer perception and detection functions:

- `OEDetectMonomers`
- `OEGetPolymerType`
- `OEGetMonomerType`
- `OEGetAminoAcidType`
- `OEIsStandardMonomer`
- `OEGetStandardAnalog`
- `OECreatePeptideMonomerConnectionSmiles`
- `OEMonomerConnectionSmilesFromCXSmiles`
- `OEPerceivePeptideMonomerConnections`

HELM functions:

- `OEHelmToMol`
- `OEMolToHelm`
- `OEGetHelmChainNames`

  - `OEGetHelmMonomerCodes`

  Peptide perception functions:

  - `OEIsPeptide`
  - `OEPerceivePeptideBackbone`

- A new function, `OEHasRadii`, has been added to check if a molecule has radii assigned to all atoms. The function takes an optional boolean argument `all` (default: `true`) to specify whether to check for radii on every atom or whether to return early on the first atom without radii.
- More `OEIFlavor` constants have been added for better control of perception steps when reading mmCIF files.

6.2.2 Minor Bug Fixes

- When generating a substructure search database with `OECreateSubSearchDatabaseFile`, molecules with high graph complexity (such as https://www.rcsb.org/ligand/ZRW) are now skipped since they are unlikely to be useful for substructure searching and can cause significant performance degradation during database generation.
- The stack size of threads used in the `OECreateSubSearchDatabaseFile` function have been increased to avoid memory allocation issues when processing `.oeb` files with large number of generic data.
- An issue has been fixed in `OEGetTorsion` where residue perception was called on a const atom changing the underlying perception state of the molecule. The function now correctly returns an invalid value when the residue is not already perceived with a warning to perceive it.

6.2.3 Python-Specific Changes

- The `OEMolBase.NewGroup` method raises `TypeError` if invoked with an invalid argument type.
- The hashing function for OEResidue has been updated to work with multicharacter chain IDs.

6.3 OEBio TK

6.3.1 New Features

- The following preliminary APIs have been added to serializing and deserializing interaction hints into `.oeb` files.
  - `OEConstructInteractionHintContainer`
  - `OEDeleteInteractionsHintSerializationData`
  - `OEDeleteInteractionsHintSerializationIds`
  - `OEHasInteractionHintSerializationIds`
  - `OEHasInteractionsHintSerializationData`
  - `OESerializeInteractionsHintContainer`
  - `OESetInteractionsHintSerializationIds`
- The serialization API for interaction hints now supports OEDesignUnit directly. The following functions have new overloads accepting an `OEDesignUnit`:
  - `OEConstructInteractionHintContainer`
  - `OEDeleteInteractionsHintSerializationData`
  - `OEDeleteInteractionsHintSerializationIds`
  - `OEHasInteractionsHintSerializationData`

  - `OESetInteractionsHintSerializationIds`

- The OEInteractionHintContainer class now supports a new constructor overload that accepts an `OEDesignUnit`, assembling the receptor from the design unit components and the ligand (if present) automatically.
- Intramolecular interaction perception is now supported for a single component (either ligand or protein) by passing an empty molecule for the other component to the OEInteractionHintContainer constructor. For example, passing an empty molecule as the ligand argument will perceive only intramolecular interactions within the protein, and vice versa. This also applies when using the `OEDesignUnit` constructor overload: if a design unit has site residues but no ligand, intramolecular receptor interactions are perceived automatically.
- The following new predicates have been added for finer-grained filtering of interaction hints:
  - OEIsClashingHBondInteractionHint identifies clashing hydrogen bond interactions (i.e., `OEHBondInteractionHintType.ClashAcceptorAcceptor` and `OEHBondInteractionHintType.ClashDonorDonor` types).
  - OEIsUnpairedInteractionHint identifies all unpaired interactions, including unpaired hydrogen bond donors/acceptors and unpaired salt-bridge charges.
  - OEIsIntermolecularSaltBridgeInteractionHint identifies intermolecular (ligand-protein) salt-bridge interactions.
  - OEIsIntramolecularSaltBridgeInteractionHint identifies intramolecular salt-bridge interactions within the ligand or protein.
  - OEIsIntermolecularStackingInteractionHint identifies intermolecular (ligand-protein) pi- and T-stacking interactions.
  - OEIsIntramolecularStackingInteractionHint identifies intramolecular pi- and T-stacking interactions within the ligand or protein.
  - OEIsIntermolecularCationPiInteractionHint identifies intermolecular (ligand-protein) cation-pi interactions.
  - OEIsIntramolecularCationPiInteractionHint identifies intramolecular cation-pi interactions within the ligand or protein.
  - OEIsIntermolecularHalogenBondInteractionHint identifies intermolecular (ligand-protein) halogen bond interactions.
  - OEIsIntramolecularHalogenBondInteractionHint identifies intramolecular halogen bond interactions within the ligand or protein.
- Intramolecular stacking and cation-pi interactions are now perceived and reported. The following new type constants have been added:
  - `OEStackingInteractionHintType.IntramolecularLigandPi`
  - `OEStackingInteractionHintType.IntramolecularLigandT`
  - `OEStackingInteractionHintType.IntramolecularProteinPi`
  - `OEStackingInteractionHintType.IntramolecularProteinT`
  - `OECationPiInteractionHintType.IntramolecularLigand`
  - `OECationPiInteractionHintType.IntramolecularProtein`
- A new option has been added to OEPerceiveInteractionOptions to control the minimum stack offset required for a pi-stacking or T-stacking interaction to be reported:
  - `OEPerceiveInteractionOptions.SetMinStackOffset` / `OEPerceiveInteractionOptions.GetMinStackOffset`

The stack offset is defined as $\max(|\hat{c} \cdot \hat{n}_1|, |\hat{c} \cdot \hat{n}_2|)$, where $\hat{c}$ is the unit vector between the ring centers and $\hat{n}_1$ and $\hat{n}_2$ are the ring plane normals. This filter ensures that one ring sits above or below the other (i.e., a genuinely stacked arrangement) rather than side by side in the same plane. The default value is **0.5**.

- The following existing predicates have been updated to accept optional parameters that control whether intramolecular interactions and nonideal interactions are included in their results. All parameters default to `True` to preserve backward-compatible behavior where interactions were previously returned without distinction:
  - OEIsHBondInteractionHint — new constructor `OEIsHBondInteractionHint(bool includeIntra = true, bool includeNonIdeal = true)`
  - OEIsIntermolecularHBondInteractionHint — new constructor `OEIsIntermolecularHBondInteractionHint(bool includeNonIdeal = true)`
  - OEIsIntramolecularHBondInteractionHint — new constructor `OEIsIntramolecularHBondInteractionHint(bool includeNonIdeal = true)`
  - OEIsChelatorInteractionHint — new constructor `OEIsChelatorInteractionHint(bool includeIntra = true)`
  - OEIsSaltBridgeInteractionHint — new constructor `OEIsSaltBridgeInteractionHint(bool includeIntra = true)`
  - OEIsStackingInteractionHint — new constructor `OEIsStackingInteractionHint(bool includeIntra = true)`
  - OEIsCationPiInteractionHint — new constructor `OEIsCationPiInteractionHint(bool includeIntra = true)`
  - OEIsHalogenBondInteractionHint — new constructor `OEIsHalogenBondInteractionHint(bool includeIntra = true)`
- OEDesignUnit now supports an additional `SetSiteResidues` overload that accepts `std::vector<OEChem::OEResidue>`. The method validates that residues are present in the target, supports ligand coverage checks through `threshold` and `addBox`, supports optional component reassignment through `rearrangeComponents`, and updates site-associated cofactors through `siteSize`.
- A new overload to the function `OEAssignRadii` that operates on `OEDesignUnit` has been added. The function takes a mask argument and radii types to assign specific radii to specified components.
- The `OEDesignUnit` now tracks the bond order of covalent bonds between components.
- The dictionaries used in `OEResidueCategoryData` have been updated. Specifically, *GCP* has been added as a cofactor. In addition, polymers were reclassified based on how often they appear as a standalone in the wwPDB, so they can be targeted as ligands, or, if they appear frequently as part of larger complexes, they can be treated as cofactors.

6.3.2 Major Bug Fixes

- An issue has been fixed where using OEAltLocationFactory with the `OEAltLocFactOption.TrustInputBonds` option could lead to incorrect perception of formal charges for atoms that had an atom with multiple alternative locations bonded to it instead of proper reassignment of the implicit hydrogen count first.
- The OEIsHBondInteractionHint predicate previously returned `true` for clashing hydrogen bond interactions (`OEHBondInteractionHintType.ClashAcceptorAcceptor` and `OEHBondInteractionHintType.ClashDonorDonor`) and for unpaired hydrogen bond interactions (`OEHBondInteractionHintType.UnpairedLigandDonor`, `OEHBondInteractionHintType.UnpairedLigandAcceptor`, `OEHBondInteractionHintType.UnpairedProteinDonor`, and `OEHBondInteractionHintType.UnpairedProteinAcceptor`). These interaction types are now **excluded** from OEIsHBondInteractionHint, as they represent geometrically unfavorable or unpaired contacts rather than genuine hydrogen bonds. Use the new

OEIsClashingHBondInteractionHint predicate to retrieve clashing interactions, and the new OEIsUnpairedInteractionHint predicate for unpaired interactions (note: `OEIsUnpairedInteractionHint` also returns unpaired salt-bridge interactions).

- Similarly, OEIsIntermolecularHBondInteractionHint no longer includes clashing types, and OEIsIntramolecularHBondInteractionHint no longer includes unpaired types.
- The directionality of intramolecular hydrogen bond, salt bridge, and chelator interactions has been corrected:
  - Intramolecular hydrogen bond interactions now consistently report the donor as the begin fragment and the acceptor as the end fragment.
  - Intramolecular salt bridge interactions now consistently report the positive charge as the begin fragment and the negative charge as the end fragment.
  - Intramolecular chelator interactions now consistently report the metal as the begin fragment and the chelating atom as the end fragment.
- The `OEPerceiveInteractionOptions.GetMaxAtomPairDistance` has been updated to also include the MaxUnpairedDistance in the calculation of the maximum atom pair distance.

6.3.3 Minor Bug Fixes

- `OEGetActiveSiteInteractionHintTypes` no longer returns duplicate interaction hint types.
- An issue has been fixed where incorrect perception of stacking interactions would occur. The new feature mentioned above fixes this issue with an option to control the minimum stack offset required for a pi-stacking or T-stacking.
- An issue has been fixed where chelators were incorrectly perceived because of a missing angular term.
- An issue has been fixed where halogen interactions were incorrectly perceived because of a missing angular term.
- An issue has been fixed where, in the `OEIsCTerminalAtom` and `OEIsNTerminalAtom` methods, incorrect terminal atom identification could occur for nonstandard amino acid residues.

6.3.4 Python-Specific Changes

- Python wrapping of `OEInteractionHintContainer.GetMolecules` methods has been fixed.

6.4 OEDepict TK 2.7.0

6.4.1 New Features

- Two new methods, GetOptimizeMacrocycles and SetOptimizeMacrocycles, have been added to OEPrepareDepictionOptions to enable enhanced algorithmic depiction of macrocycle rings.

6.5 OEDocking TK 4.3.5

6.5.1 Minor Bug Fixes

- An issue has been fixed that was causing OEPosit to pick the wrong method in posing some ligands.

6.6 EON TK 3.1.3

6.6.1 New Features

- Two new methods, GetJoinRef and SetJoinRef, have been added to OEEonPBOptions to enable PB calculation on complexes in OEEonPBFunc.

6.7 FastROCS TK 2.3.2

Minor internal improvements have been made.

6.8 Grapheme™ TK 1.5.5

6.8.1 New Features

- The following preliminary APIs have been added for monomer and peptide depiction support.

  Classes:

  - `OEPeptideDisplayOptions` — options for peptide depiction.
  - `OEHighlightMonomerDisplayOptions` — options for monomer highlight depiction.
  - `OEMonomerColorDefault` — default monomer color functor.
  - `OEMonomerColorMonochrome` — monochrome monomer color functor.
  - `OEMonomerGlyphStyleBase` — base class for monomer glyph style functor.
  - `OEMonomerGlyphStyleDefault` — default monomer glyph style functor.
  - `OEMonomerGraphDisplayOptions` — options for monomer graph depiction.

  Functions:

  - `OEDrawMonomer` — draws a monomer image.
  - `OEDrawMonomerGraph` — draws a monomer graph of a peptide.
  - `OEHighlightMonomers` — highlights monomers in a peptide depiction.
- The `OERenderActiveSite` function now displays a counter on the top of the T- and pi-stacking interaction lines if multiple interactions exist between the aromatic ring of the ligand and the aromatic ring of a specific residue.
- The following APIs have been added to allow customization of ligand alignment in protein–ligand interaction depictions:
  - `OELigandAlignerBase`
  - `OE2DActiveSiteDisplayOptions.SetLigandAlignerFunctor` method
  - `OE2DActiveSiteDisplayOptions.SetOptimizeLigandOrientation` method

6.8.2 Minor Bug Fixes

- The `OEDrawResidues` function now depicts molecules with up to 300 heavy atoms.

6.9 GraphSim TK 2.7.1

6.9.1 Minor Bug Fixes

- The hydrogen counts of neighbor atoms when R-groups are removed prior to fingerprint generation have been adjusted.

6.10 Lexichem TK 2.9.5

Minor internal improvements have been made.

6.11 OEMedChem TK 1.2.6

6.11.1 New Features

- A new overload of the OECreateIDStrings function has been added to enable speeding up **BROOD** hit list clustering when working with multiquery replacements.

6.12 MolProp TK 2.7.0

6.12.1 Minor Bug Fixes

- The SMARTS pattern of the halo_amine functional group has been fixed.

6.13 OEFF TK 2.9.2

6.13.1 Minor Bug Fixes

- The OEQuatAdaptor now passes the coordinates to the internal OEMolFunc when appropriate.

6.14 Omega TK 7.0.0

6.14.1 New Features

- The following changes have been made to extend Thompson sampling usage for torsion driving in OEOmega:
  - The Thompson sampling algorithm for torsion driving has been improved for robustness and speed.
  - The default value for SetUseThompson in OETorDriveOptions has been changed to `True`.
  - The SetUseThompson is honored even when SetUseGPU is set to `True`.
  - The preliminary API class `OEThompsonOptions` has been removed, as it is no longer relevant.
- The default energy window for torsion driving in Classic mode has been changed to a range-based value to better suit **OMEGA** with Thompson sampling. Similarly, the energy range for the Pose mode has also been adjusted.

6.14.2 Minor Bug Fixes

- OEMolBuilder now respects the MaxEnumConfs provided in the OEMolBuilderOptions even when an input molecule has more conformers than specified.
- Certain edge cases where the torsion driving algorithm would fail with a FailedTorDrive error message have been addressed.

6.15 Quacpac TK 2.3.0

Minor internal improvements have been made.

6.16 Saiph TK 1.0.1

6.16.1 Python-Specific Changes

- The following APIs are now properly wrapped in Python:
  - OESaiphGetArch
  - OESaiphGetLicensee
  - OESaiphGetPlatform
  - OESaiphGetRelease
  - OESaiphGetSite
  - OESaiphGetVersion
  - OESaiphIsLicensed

6.17 Shape TK 3.10.0

6.17.1 New Features

- When using the Overlay method in OEOverlay for overlay optimization, any of the inertial start methods now automatically trigger an additional as-is start if the optimization results in similarity measures less than the original similarity. This improves the default behavior of Overlay for some calculations.
- The following modifications have been made to OEOverlapPrepOptions to provide more flexibility in assigning color atoms during the Prep process in OEOverlapPrep:
  - Two new methods, SetRemoveColor and GetRemoveColor, have been added to enable removal or reassignment of color atoms.
  - The new methods AddDuplicateSmarts, GetDuplicateSmarts, and ClearDuplicateSmarts have been added to enable duplicate color atom removal based on provided SMARTS patterns.
  - Duplicate color atom removal is now more robust in keeping the central atoms instead of the first atom that it encounters and in placing the remaining atoms appropriately (at the center of corresponding clusters).
  - Defaults for SetRemoveDuplicate and SetDuplicateDistance have been changed from `False` and *0.001* to `True` and *0.0*, respectively. This improves robustness but keeps the default behavior unchanged.
- A new method, SetupFrame, has been added to OEStarts that simplifies setting up a reference frame for calculating the starts.

6.17.2 Minor Bug Fixes

- The overlay optimization in OEHermiteOverlay with the OEAsIsStarts start option has been updated to use Hermite expansions around the center of mass to prevent inconsistent overlaps when objects are far from the origin.
- Overlay optimization with the Overlay method in OEOverlay now captures the fit conformer self-overlap and color scores more accurately during the optimization.

6.18 SiteHopper TK 2.1.4

Minor internal improvements have been made.

6.19 Spicoli TK 1.6.4

Minor internal improvements have been made.

6.20 Spruce TK 1.9.0

6.20.1 New Features

- The maximum cluster size for partial residues built using OEBuildSidechains can now be set in OESidechainBuilderOptions using SetMaxClusterSize.
- The chemical component dictionary has been updated. However, if using mmCIF, the metadata will be used as primary information and the dictionary only engages if metadata is unavailable.

6.20.2 Minor Bug Fixes

- An issue has been fixed in the routine to reassign ambiguous water and metals from packing residues to the protein chain of interest. The issue caused HEME analogs and connected amino acids to be soaked into a given chain ID.
- An issue has been fixed to allow `OEHeterogenMetadata` to inform the component splitter during design unit generation and preparation. This also fixes an issue where the splitter would not allow reclassification of a metal ion to an excipient.
- An issue has been fixed that caused the option to collapse redundant alternate locations to improperly engage if there were multiple binding sites in a protein.
- An issue has been fixed that caused the option to collapse redundant alternate locations to improperly engage for different ligands in the same binding site.
- An issue has been fixed that caused `OEMutateResidues` to incorrectly leave the backbone nitrogen of a residue in a bad valence state when mutating from Proline.
- An issue has been fixed in `OEBuildSidechains` with the shell mutation option turned on that would leave side chains clashing with water molecules. The water molecules are now removed if the option to allow for removal of clashing waters is set to `True`.

6.21 Szmap TK 1.7.4

Minor internal improvements have been made.

6.22 Szybki TK 2.9.2

Minor internal improvements have been made.

6.23 Zap TK 2.5.3

6.23.1 Major Bug Fixes

- Salt concentration implementation in **Zap TK** has been corrected and made robust at higher salt concentrations.

- As a result of the salt concentration improvement, the rotational variation of Zap solvation energy at high salt concentrations has been substantially reduced.

CHAPTER

SEVEN

RECENT RELEASE HISTORY

7.1 OEToolkits 2025.2

7.1.1 Release Highlights 2025.2

Shape: Features Extension

New functionalities have been added to improve **Shape TK** and **ROCS** handling of queries beyond those derived from molecules. Major new features include using queries consisting only of color atoms or Gaussians (Figure 1), and the ability to search a database of grids as well as molecules. The ability to work with non-molecular queries has also been enhanced by introducing a new algorithm to generate Gaussians from a grid that can better capture sparse grids (Figure 2).

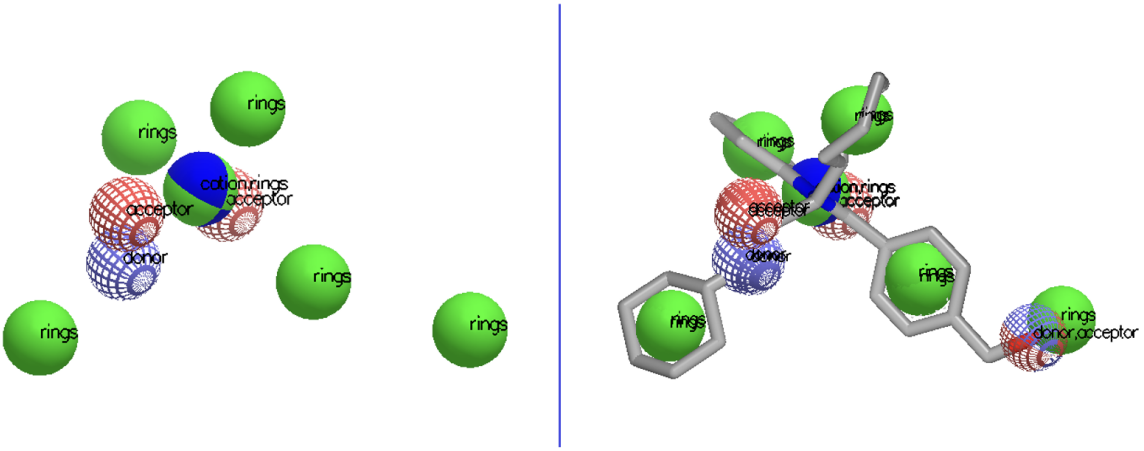


Fig. 1: **Figure 1.** A color-only query consisting only of color Gaussians (left). The overlay of a molecule on that color-only query (right).

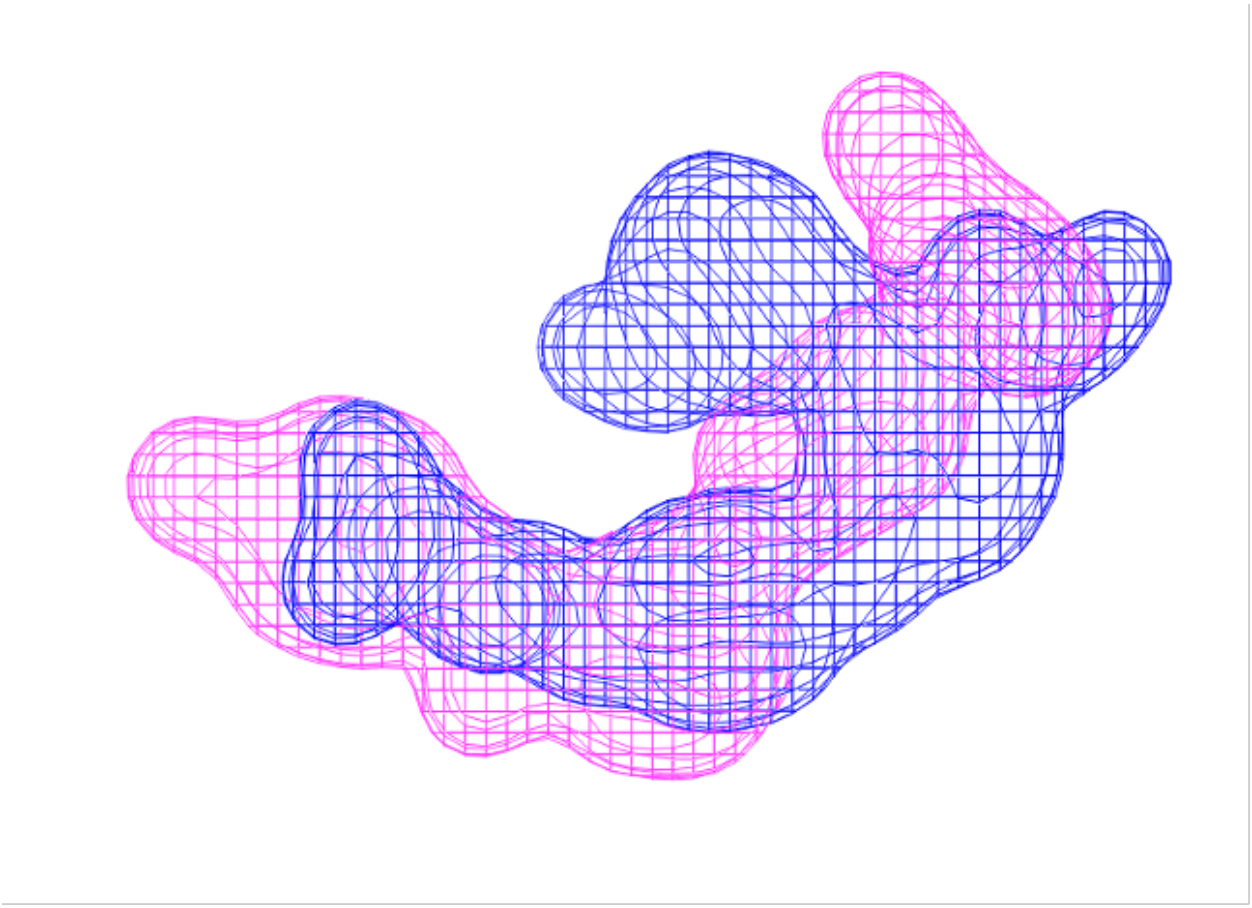


Fig. 2: **Figure 2.** The overlay of a grid on another grid as query.

Saiph TK: New Toolkit for ETL

This release introduces **Saiph TK**, a toolkit for extracting, transforming, and loading (ETL) files, bringing the existing Orion ETL functionality for file conversion into the toolkits, providing greater accessibility and improved performance.

Major functionality in the **Saiph TK** includes conversion of various molecule file formats, including SMILES and CXSMILES, SD, Spruce TK design unit files, and Shape TK query files, into a set of records. **Saiph TK** also provides functionality to interconvert CSV files and records, and to create a Pandas dataframe directly from any supported format, enabling smooth integration of molecule objects into machine learning in Python.

OEChem TK and Spruce TK: Improvements to CIF File Format Support

The Worldwide Protein Data Bank (wwPDB) has officially deprecated the legacy PDB file format in favor of mmCIF. The format has recently been expanded to include residue topology information, and the RCSB has worked to ensure that all deposited structures include these new mmCIF-specific categories. In alignment with wwPDB and RCSB recommendations, we now advise using the mmCIF format for all macromolecular structures.

To support the transition, the CIF file reader in **OEChem** has been improved particularly in regard to reading the mmCIF file format variant. New APIs have been added that allow parsing and modification of the mmCIF metadata. Improvements to CIF file writing has also been made to be more correct when header data is missing or has been manually altered.

The **Spruce** filter now leverages mmCIF header data to refine structure perception during standardization prior to preparation. Additionally, support has been added for reading structure factor CIF files into grids, replacing MTZ files

for electron density maps.

Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.10 - 3.13	RHEL8/9, Ubuntu22/22-ARM/24	Win10/11	13, 14, 15
C++		RHEL8/9, Ubuntu22/22-ARM/24	Win10/11 (VS2022)	13, 14, 15
Java	8, 11, 21	RHEL8/9, Ubuntu22/22-ARM/24	Win10/11	13, 14, 15
C#			Win10/11 (VS2022)	

Related TK Versions

Toolkit	Version
Bioisostere TK	4.2.1
OEChem TK	4.3.0
OEDepict TK	2.6.0
OEDocking TK	4.3.4
Eon TK	3.1.2
FastROCS TK	2.3.1
OEFF TK	2.9.1
Grapheme TK	1.5.4
Graphsim TK	2.7.0
Lexichem TK	2.9.4
OEMedChem TK	1.2.5
MolProp TK	2.6.7
Omega TK	6.1.1
Quacpac TK	2.2.7
Saiph TK	1.0.0
Shape TK	3.9.0
SiteHopper TK	2.1.3
Spicoli TK	1.6.3
Spruce TK	1.8.0
Szmap TK	1.7.3
Szybki TK	2.9.1
Zap TK	2.5.2

7.1.2 General Notices

- Support for Python 3.13 has been added for all supported platforms.
- Support for Python 3.10 will be dropped in the Spring 2026 release, and Python 3.11 will be dropped in the fall of 2026.
- Support for Python 3.14 will be added in the Spring 2026 release.
- Support for macOS 26 will be added in the Spring 2026 release, and support for macOS 13 will be dropped.
- Support for Windows 10 will be dropped in the Spring 2026 release.

7.2 OEToolkits 2025.1

7.2.1 OEToolkits 2025.1

Release Highlights 2025.1

FastROCS: Grid Query Support

The 2025.1 release adds support for grid shape queries in **FastROCS TK**, as is already available in the **Shape TK** and **ROCS**. The ability to use shape grids substantially expands the application of **FastROCS TK** to queries beyond those derived from molecules.

To assess the efficacy of grid-derived versus molecule-derived queries **ROCS** was run with a series of thirteen molecule-derived queries, as well as the corresponding grid-derived queries, against a database of 1M molecules and the top-ranked 10K hitlist for each of the 13 was retained. **FastROCS** was then run with the same molecule-derived and grid-derived queries, and the fraction of the top-ranked 10K hits in common with the reference ROCS hitlist was calculated (see Figure 1). The fraction of molecules in common is almost completely constant (compare the black and pink bars in Figure 1), whether FastROCS was used with a molecule-derived query or the corresponding grid-derived query.

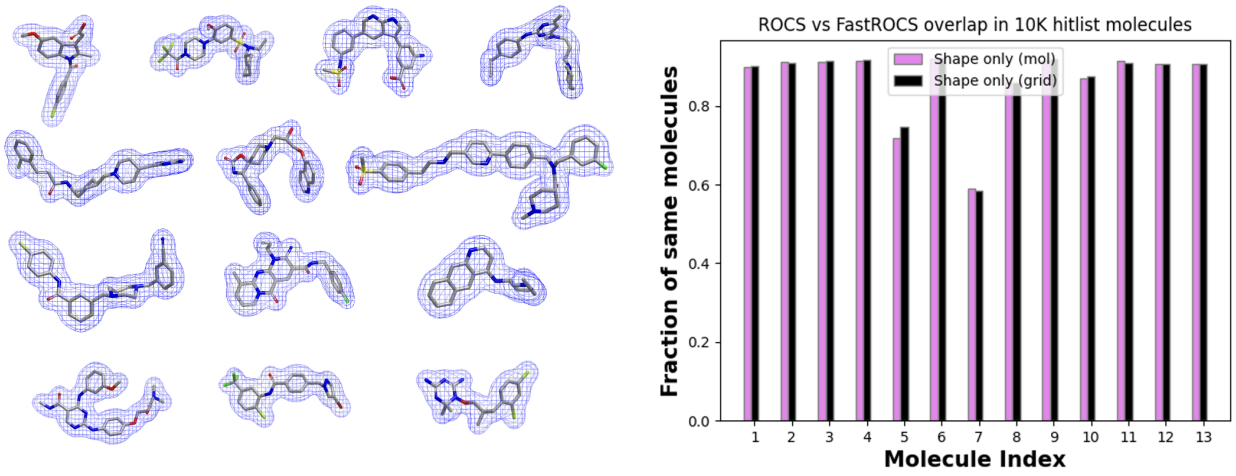


Fig. 3: **Figure 1.** Thirteen query molecules and their shape grids (left). A graph of **ROCS** versus **FastROCS** overlap in finding the same molecules in the hit list (right).

Bioisostere TK: Feature Extensions

New functionality has been added to **Bioisostere TK**, bringing all features of the **BROOD** application into the toolkit. Some of the major new features in the toolkit include the ability to build queries in the presence of the target protein binding site and/or a selectivity protein, as well as creating bridging queries that connect two molecules. New APIs have also been added to enable checking whether compounds similar to the analogs generated by **BROOD** exist in commercial or internal databases, allowing a chemist to consider molecules available for purchase before undertaking the synthesis of a given analog.

With all of the **BROOD** functionality now available in Bioisostere TK, **BROOD** has been reengineered with **Bioisostere TK** to simplify and streamline coordination with the toolkit, ensuring that consistent results are provided between **BROOD** and **Bioisostere TK**. The graphical application **vBROOD** has been also been rewired to the newly engineered **BROOD**.

Representation of both active site and selectivity proteins in **Bioisostere TK**, **BROOD**, and **vBROOD** has been modified to work with the **OEDesignUnit**.

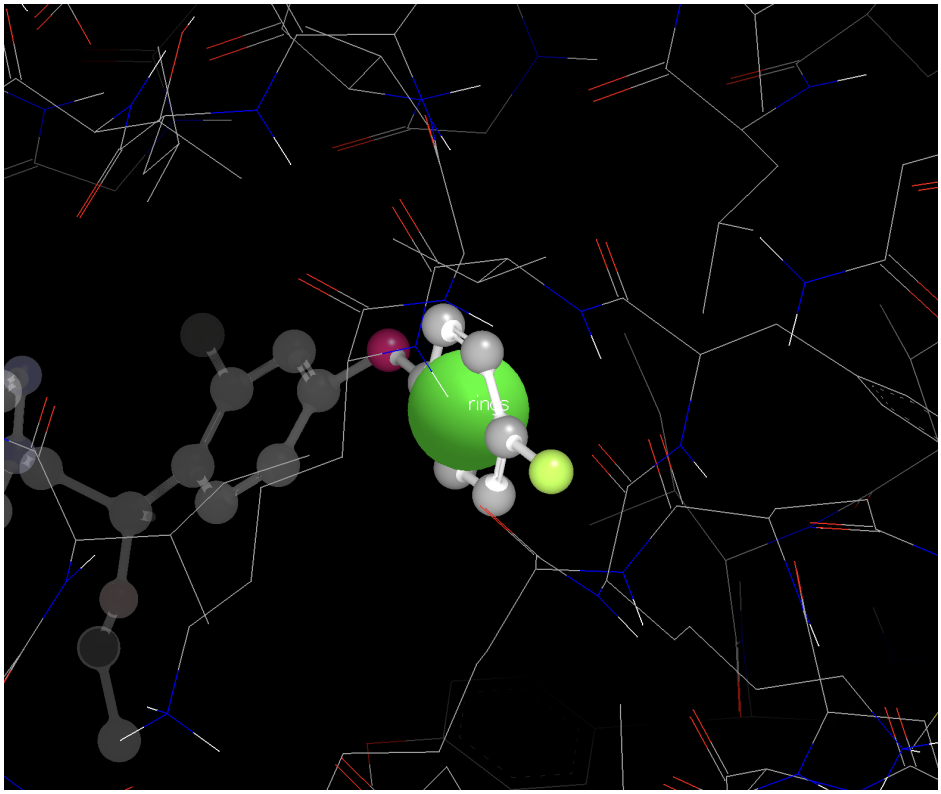


Fig. 4: **Figure 2.** Building a **BROOD** query in a protein active site.

SZYBKI: Robust Protein–Ligand Optimization

The robustness of ligand optimization in a protein active site has been improved in both **Szybki TK** and **SZYBKI**.

Optimization failures due to non-planarity in aromatic rings of optimized structures are now separated from those due to excess atomic overlap in the starting structures, allowing better handling of failed optimizations. Failures due to a low level of aromatic ring bending, which is a common artifact of force-field-based optimization, can now be ignored, producing a substantial increase in successful ligand optimizations. The criterion for flagging an optimized structure as having bent aromatic rings has also been modified. Overall, the revised algorithm allows more structures to pass through the minimization process, while still giving the user the ability to flag results with deformed aromatic rings.

Besides these algorithmic modifications, there are improvements in integrating optimization of **POSIT** generated pose and in assigning **AM1BCCELF10** charges to ligands, with **SZYBKI**.

Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.10 - 3.12	RHEL8/9, Ubuntu22/22-ARM/24	Win10/11	13, 14, 15
C++		RHEL8/9, Ubuntu22/22-ARM/24	Win10/11 (VS2022)	13, 14, 15
Java	8, 11, 20	RHEL8/9, Ubuntu22/22-ARM/24	Win10/11	13, 14, 15
C#			Win10/11 (VS2022)	

Related TK Versions

Toolkit	Version
Bioisostere TK	4.2.0
OEChem TK	4.2.0
OEDepict TK	2.5.6
OEDocking TK	4.3.3
FastROCS TK	2.3.0
OEFF TK	2.9.0
Grapheme TK	1.5.3
Graphsim TK	2.6.2
Lexichem TK	2.9.3
OEMedChem TK	1.2.4
MolProp TK	2.6.6
Omega TK	6.1.0
Quacpac TK	2.2.6
Shape TK	3.8.0
SiteHopper TK	2.1.2
Spicoli TK	1.6.2
Spruce TK	1.7.0
Szmap TK	1.7.2
Szybki TK	2.9.0
Zap TK	2.5.1

General Notices

- Support for Ubuntu 24.04 has been extended for all applications except VIDA. Support for VIDA will be added in a later release. Support for Ubuntu 24.04 for C++, Python, and Java toolkits was already added in the previous release. Support for Ubuntu 20.04 has been discontinued from this release.
- Support for macOS 15 has been added for C++, Python, and Java toolkits, as well as applications and **VIDA**.
- Support for Python 3.13 is not available in this release but will be added in a future release. Support for Python 3.9 has been discontinued from this release.

7.3 OEToolkits 2024.2

7.3.1 OEToolkits 2024.2.1

The **OEToolkits 2024.2.1** is a bug-fix of the **OEToolkits 2024.2.0** release.

- OEChem Notes
- Spruce Notes

7.3.2 OEToolkits 2024.2.0

Release Highlights 2024.2.0

OMEGA: Thompson Sampling for Torsion Driving

A Bayesian approach to explore conformer space with torsion driving has been added as an alternative to the exhaustive searching of the space in **OMEGA**. This uses the framework of Thompson sampling to quickly direct the conformer search towards the low energy structures, reducing the time required to find the lowest energy conformer. As such, Thompson sampling is particularly suitable for generating small to moderately sized ensembles, such as used in the

FastROCS or ROCS mode. Thompson sampling provides significant speedup for such ensemble generation, particularly for molecules with a large number of rotors.

Conformer ensembles for FastROCS usage are generated with **OMEGA** using both the standard exhaustive sampling and the newly implemented Thompson sampling, for ~5500 Iridium HT protein bound small molecule crystal structures. Performance between the two sampling methods is compared in terms of RMSD between the generated ensemble and the crystal pose, and in the time required to generate the ensembles. The results of the comparison, as shown in Figure 1 below, show that the accuracy of the generated ensembles, measured in RMSD, remains unaffected when using exhaustive versus Thompson sampling, whereas there is significant reduction in time usage to generate those ensembles when using Thompson sampling, especially as the size of the molecules grows.

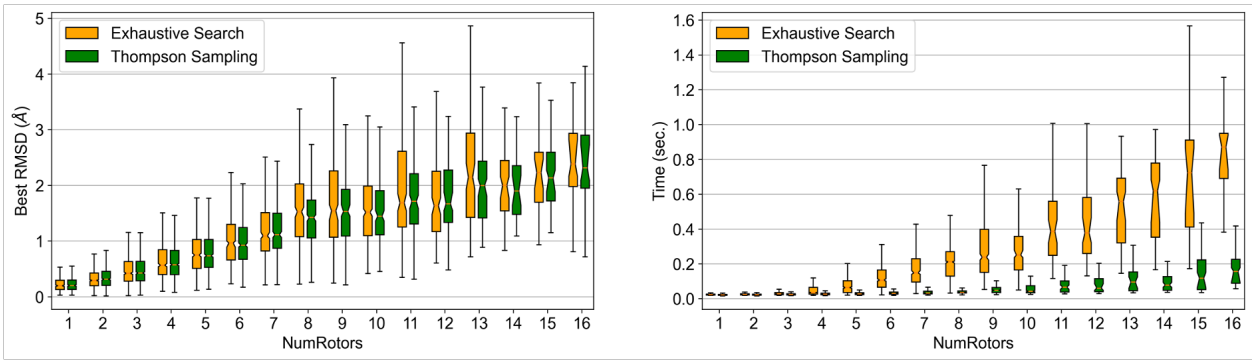


Fig. 5: **Figure 1.** Comparison of **OMEGA** performance between exhaustive and Thompson sampling.

Ligand-based virtual screening performance, using both the exhaustive and Thompson sampling generated ensembles, was also compared and displayed in Figure 2. The comparisons were performed using directory of useful decoys (DUD-Z) datasets containing 38 targets. The results ensure that there is no meaningful change in accuracy when using Thompson sampling for ensemble generation for usage in ligand-based virtual screening.

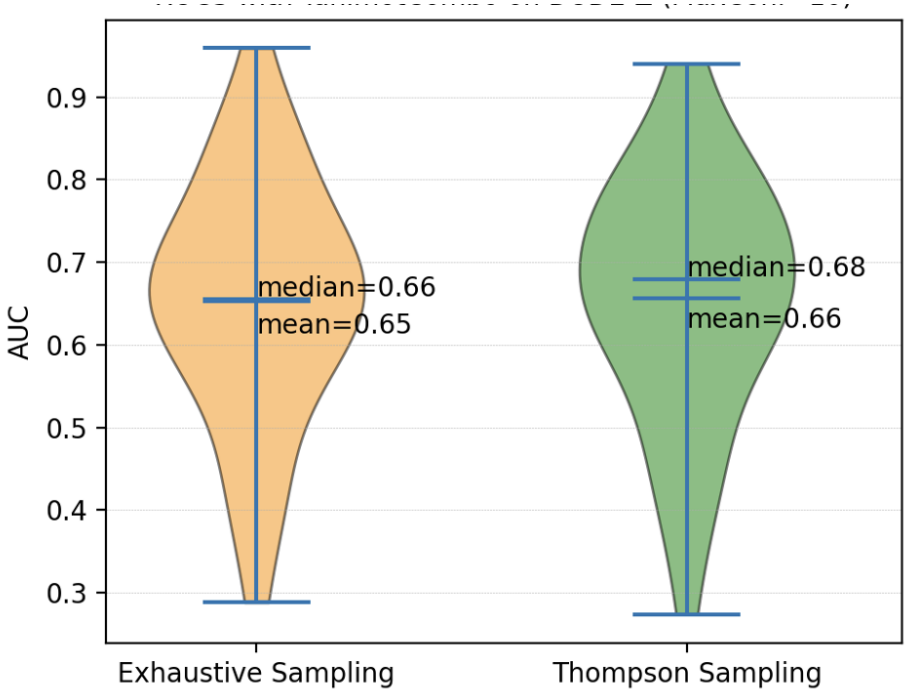


Fig. 6: **Figure 2.** Comparison of performance of ligand-based virtual screening success rate with conformer ensembles generated using exhaustive versus Thompson sampling.

BROOD: Bioisosteric Fragment Replacement in Macrocyclic Peptides

A new pose generation algorithm has been introduced in **Bioisostere TK**, as well as in **BROOD**, based on flexible overlay of shape, color, and force fields, as used in pose generation with **ShapeFit**. The new algorithm generates high quality poses of molecules that are obtained from fragment replacement in **Bioisostere TK** and **BROOD**. The robust pose generation algorithm enables usage of **Bioisostere TK** and **BROOD** for molecules with complex geometry, including macrocyclic peptides. Figure 3 shows an example of a residue fragment replacement in a macrocyclic peptide.

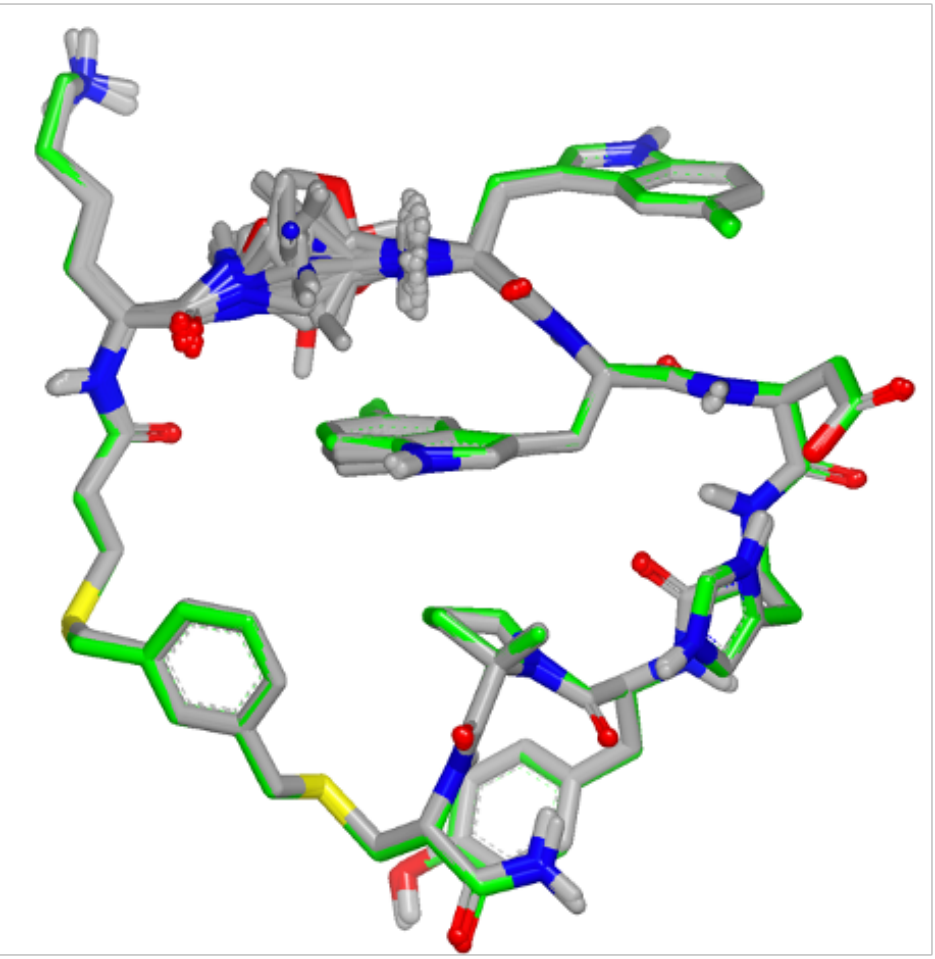


Fig. 7: **Figure 3.** Bioisosteric replacement of a residue fragment in a macrocyclic peptide using **BROOD**.

Eon TK: New Toolkit for Overlay Optimization with Shape and Charge Density

The 2024.2 release introduces **Eon TK**, a new toolkit for molecular similarity based on shape and electrostatics. This provides toolkit-level access to the existing **EON** application functionality. Following **EON**, the new toolkit has both molecular charge density and electrostatic potential as options to use as descriptors of electrostatic properties. Figure 4 shows an example of **EON** overlay with charge density visualization.

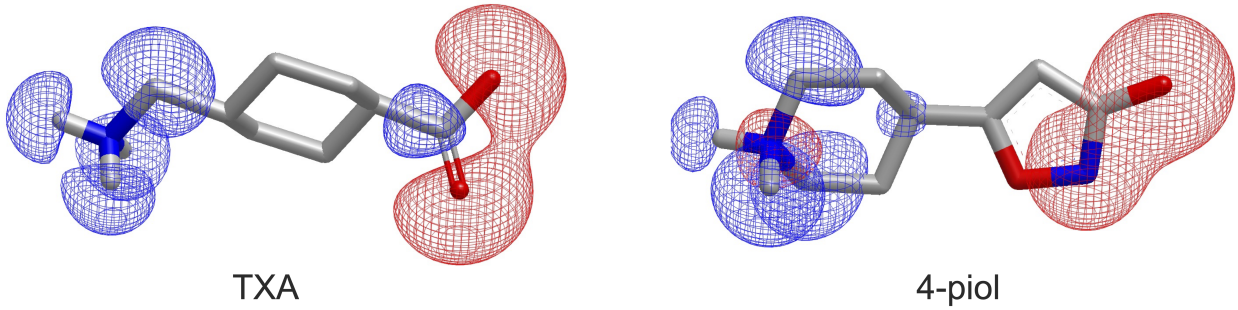


Fig. 8: **Figure 4. EON** shape charge density overlay between two molecules.

Eon TK provides tools for molecular overlay optimization with shape and charge density, similarity estimation based on shape and charge density or electrostatic potential, and corresponding hit list building.

Supported Platforms

Package	Versions	Linux	Windows	macOS (x64 and arm64)
Python	3.9 - 3.12	RHEL8/9, Ubuntu20/22/22-ARM/24	Win10/11	12, 13, 14
C++		RHEL8/9, Ubuntu20/22/22-ARM/24	Win10/11 (VS2022)	12, 13, 14
Java	8, 11, 20	RHEL8/9, Ubuntu20/22/22-ARM/24	Win10/11	12, 13, 14
C#			Win10/11 (VS2022)	

Related TK Versions

Toolkit	Version
Bioisostere TK	4.1.0
OEChem TK	4.1.1
OEDepict TK	2.5.5
OEDocking TK	4.3.2
FastROCS TK	2.2.7
OEFF TK	2.8.0
Grapheme TK	1.5.2
Graphsim TK	2.6.1
Lexichem TK	2.9.2
OEMedChem TK	1.2.3
MolProp TK	2.6.5
Omega TK	6.0.0
Quacpac TK	2.2.5
Shape TK	3.7.0
SiteHopper TK	2.1.1
Spicoli TK	1.6.1
Spruce TK	1.6.1
Szmap TK	1.7.1
Szybki TK	2.8.0
Zap TK	2.5.0

General Notices

- Support for Java 8 and 11 is now available on all platforms, including macOS arm64.
- Support for Ubuntu 24.04 has been added for C++, Python, and Java toolkits. Support for applications and VIDA will be added in the next release.
- Support for macOS 15 is not available in this release but will be added in the next release. This release will be the last to support macOS 12.
- Support for Python 3.13 is not available in this release but will be added in the next release.

- Support for Ubuntu 20.04 has been continued in this release, which will be the last to provide support for Ubuntu 20.04.

7.4 OEToolkits 2024.1

7.4.1 OEToolkits 2024.1.3

The **OEToolkits 2024.1.3** is a bug-fix of the **OEToolkits 2024.1.0** release.

- OEFF TK 2.7.1.3
- Szybki TK 2.7.1.3
- OEDocking TK 4.3.1.3
- GraphemeTM TK 1.5.1.3

7.4.2 OEToolkits 2024.1.0

Release Highlights 2024.1.0

EON: Overlay Optimization with Shape and Charge Density

EON, the tool for molecular similarity based on shape and electrostatics, has been reimagined with the addition of charge density as a descriptor for electrostatic properties. By replacing electrostatic potential with molecular charge density as the primary descriptor of electrostatic properties, **EON** no longer requires the input molecules to be pre-aligned with the query molecule. Instead, **EON** now aligns the input molecules using shape- and electrostatics-based overlay optimization, where electrostatics are modeled via molecular charge density. **EON** still has the option to use electrostatic potential as a measure of similarity, if desired. Figure 1 shows an example of **EON** overlay with charge density visualization.

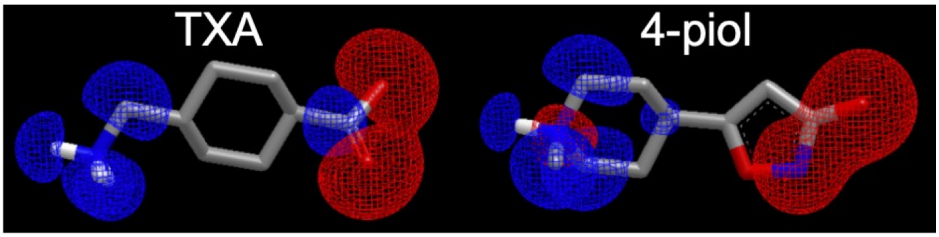


Fig. 9: **Figure 1. EON** shape charge density overlay between two molecules.

Even with the addition of overlay optimization within the workflow, the new **EON** is now significantly faster compared to the previous version; and yet, as a tool for virtual screening with shape and charge similarity, the behavior of **EON** remains the same. When compared to **ROCS**, **EON** lags slightly when measured in terms of virtual screening success rates but excels in finding hits that are more chemically diverse.

Performance of the previous and current versions of **EON**, as compared to **ROCS**, are displayed in Figure 2. The comparisons were performed using multi-query directory of useful decoys (MDUD) datasets containing 38 targets. The current version of **EON** overlays and scores with shape and charge similarity. A command line flag `-potential` supports the rescoring of shape- and charge-aligned conformers with shape and electrostatic potential similarity. The current default version of **EON** is now almost as fast as **ROCS**, as can be seen in the right-hand plot below.

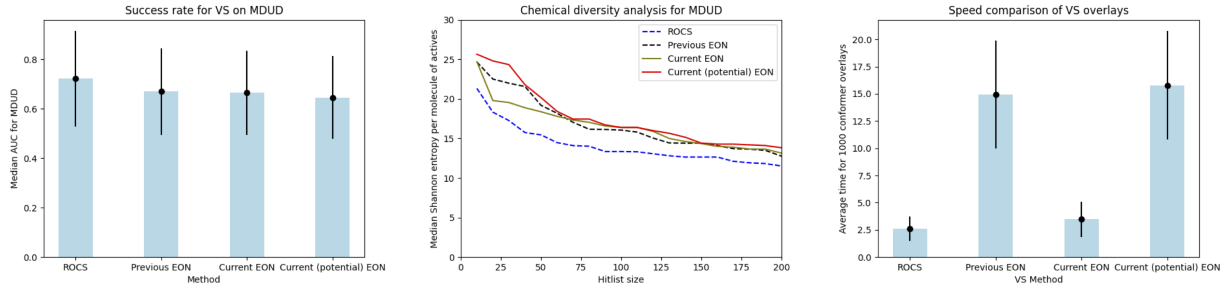


Fig. 10: **Figure 2.** Comparison of performance of **EON** and **ROCS** for virtual screening success rate (left); for chemical diversity of actives found (middle); and for the speed of calculations (right). "Current" refers to the default version using shape and charge density for similarity, and "current (potential)" refers to **EON** using shape and electrostatics.

FLYNN: Ligand fitting for Cryo-EM

FLYNN, the tool for fitting small molecules to electron density, has been extended to support density maps generated through **cryo-electron microscopy (cryo-EM)**.

FLYNN is a powerful tool for ligand fitting that uses adiabatic fitting to generate the best-fitting, lowest strain conformers consistent with experimental electron densities and structural models. This tool was one of the first to enable high-quality, physics-informed pockets and poses from X-ray crystallographic models. **FLYNN** has now been updated to enable researchers to access the same powerful insights from maps and models generated through cryo-electron microscopy (cryo-EM).

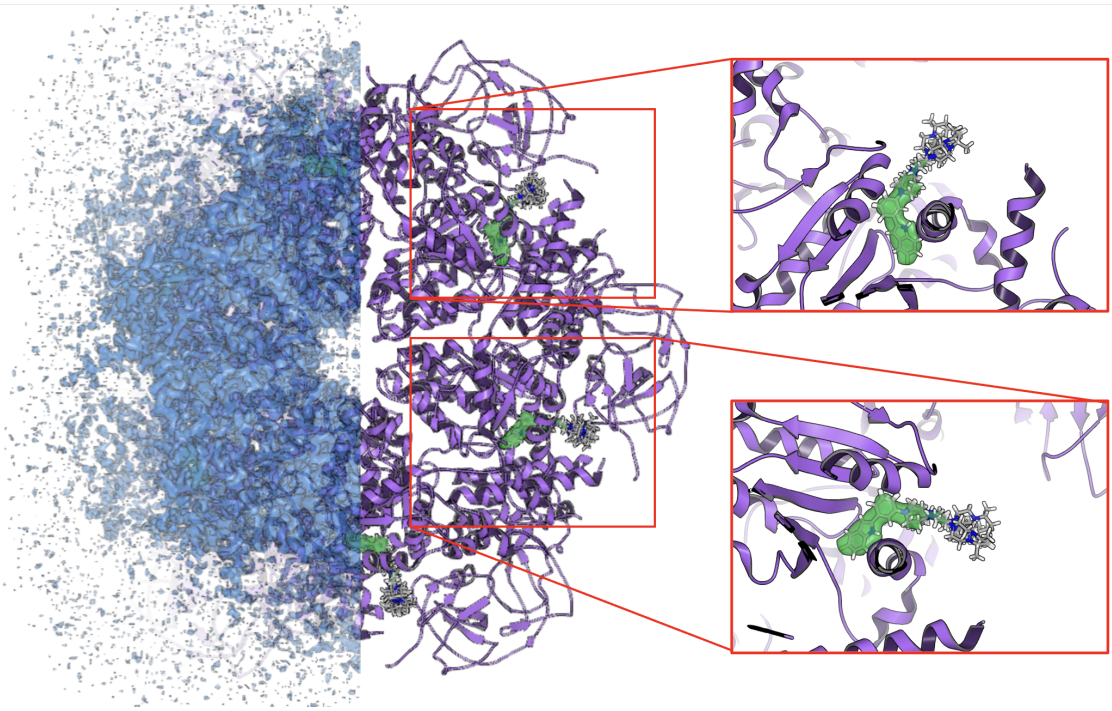


Fig. 11: **Figure 3.** Ligand fitting with **FLYNN** to predict binding pockets and poses consistent with electron density from cryo-electron microscopy (cryo-EM).

Hydrogen Placement Updates in SPRUCE

In the 2024.1 release of **SPRUCE**, several improvements have been made to OEPlaceHydrogens to handle metal cation chelation effects on the nearby protonation states, specifically for ligands. The changes shown in Figure 5 below highlight new state and optimized geometries for sulfonamide, hydroxamate, phosphate, and borate groups. In addition, changes have been made around geometries for sp^3 secondary amines. Lastly, improvements have been added in OEProtonateDesignUnit to allow for protonation state changes related to the detection of unfavorable acceptor–acceptor clashes, resulting in a more favorable hydrogen bond network.

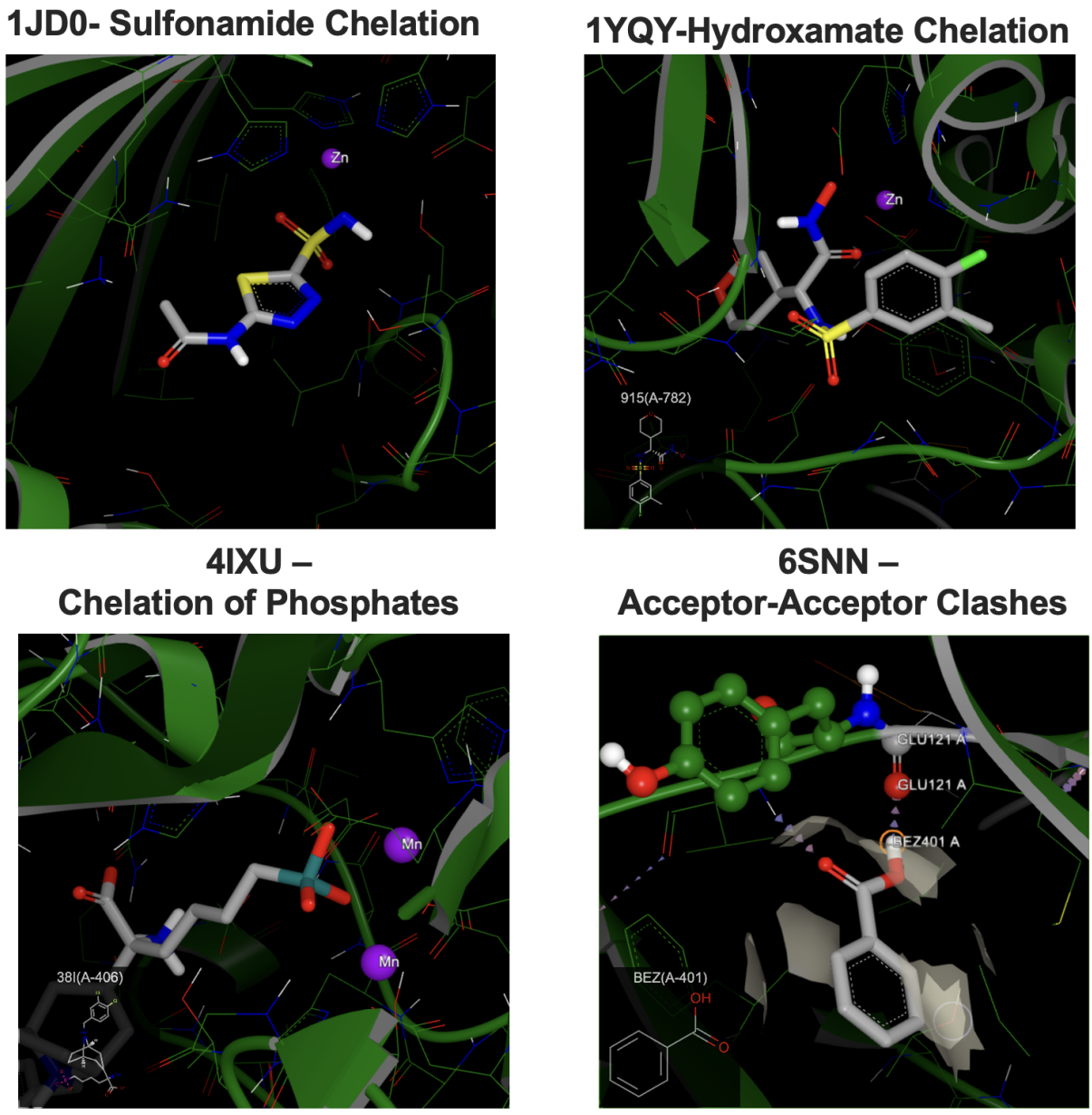


Fig. 12: **Figure 4.** Examples of improved chelator interactions and deprotonation rules for different moieties: hydroxamate, sulfonamide, and phosphates/borates. Also, on the bottom right, an example of improved interaction networks with acceptor–acceptor heuristic fixes.

Bioisostere TK: New Toolkit for Bioisosteric Fragment Replacement

The 2024.1 release introduces **Bioisostere TK**, a new toolkit for bioisosteric analog generation. It involves replacing a portion of a lead compound with fragments that have similar shape and electrostatics but with potentially novel connectivity and chemistry. This provides toolkit level access to the existing **BROOD** application functionality.

Bioisostere TK also brings several additional functionalities. At its core are tools for fragment-based modeling, such as fragment creation, fragment conformer generation, and fragment alignment and similarity in 3-dimensional space. **Bioisostere TK** brings OpenEye's shape-, color-, and electrostatics-based molecular similarity tools into fragment-based similarity. It also exposes tools for the generation of 3D molecule conformers with fragment replacement.

An example of fragment alignment with **Bioisostere TK** based on shape and color is shown in Fopeigure 3. The attachment points in both the reference and fit fragments are marked in purple. The fragment overlay highlights how special attention is paid to the overlay of the attachment points.

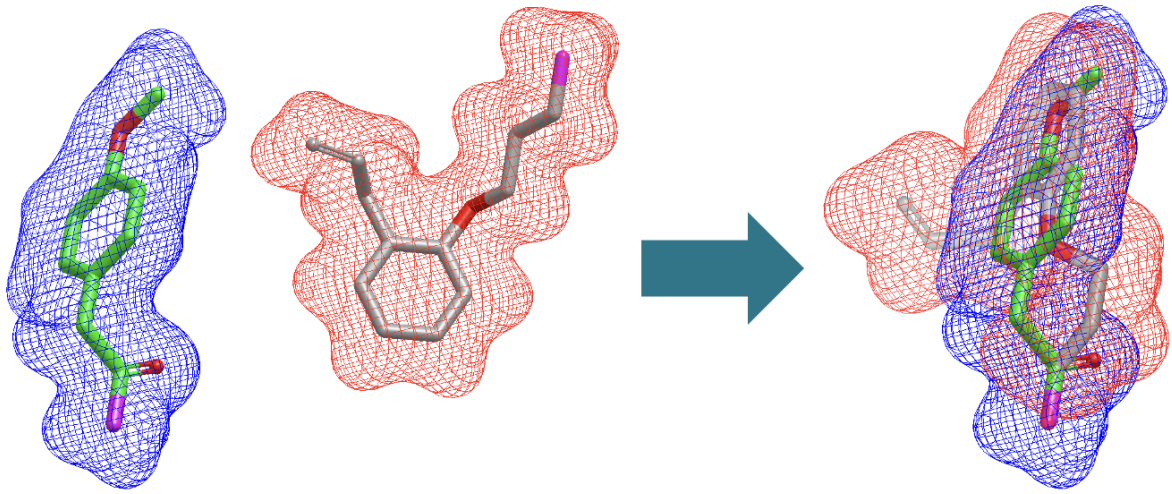


Fig. 13: **Figure 5.** Fragment overlay between a reference and a fit fragment.

Supported Platforms

Package	Versions	Linux	Windows	macOS (x64 and arm64)
Python	3.9 - 3.12	RHEL8/9, Ubuntu20/20-ARM/22	Win10/11	12, 13, 14
C++		RHEL8/9, Ubuntu20/20-ARM/22	Win10/11 (VS2022)	12, 13, 14
Java	8, 11, 20	RHEL8/9, Ubuntu20/20-ARM/22	Win10/11	12, 13, 14 (only Java 20 support for arm64)
C#			Win10/11 (VS2022)	

Related TK Versions

Toolkit	Version
Bioisostere TK	4.0.0
OEChem TK	4.1.0
OEDepict TK	2.5.4
OEDocking TK	4.3.1
FastROCS TK	2.2.6
OEFF TK	2.7.1
Grapheme TK	1.5.1
Graphsim TK	2.6.0
Lexichem TK	2.9.1
OEMedChem TK	1.2.2
MolProp TK	2.6.4
Omega TK	5.1.0
Quacpac TK	2.2.4
Shape TK	3.6.2
SiteHopper TK	2.1.0
Spicoli TK	1.6.0
Spruce TK	1.6.0
Szmap TK	1.7.0
Szybki TK	2.7.1
Zap TK	2.4.7

General Notices

- Support for Java 20 is now available on all platforms.
- Support for macOS 14 has been added.
- Python 3.12 is now supported on all platforms.
- This release will be the last to support Ubuntu 20.04. Support for Ubuntu 24.04 will be added in the next release.

CHAPTER

EIGHT

PREVIOUS RELEASE HISTORY

8.1 OEToolkits 2023.2

8.1.1 OEToolkits 2023.2.0

Release Highlights

- Sage force field (Sage 2.1.0), from the Open Force Field Consortium, is now available in **OMEGA** as a complement to MMFF94.
- The defaults for Sage force field from the Open Force Field Consortium have been updated to the latest 2.1.0 version, everywhere Sage is supported.
- The built-in fragment library in **OMEGA** has been updated; expanding on 80 K fragments in the previous library, the new fragment library contains approximately 540 K fragments.
- Stereo-enumeration options in **Flipper** have been improved to allow fine-grained control over enumerating different types of atom and bond stereocenters in an uncoupled manner.
- The torsion scanning algorithm in **SZYBKI** has been improved by adding the ability to optimize in internal coordinate space.
- Protein–ligand optimization with FF14SB-OpenFF in **SZYBKI** has been improved for performance and robustness.
- Support for PDBx/mmCIF in **OEChem TK** has been improved, following the format updates to PDB entries by wwPDB. The MMCIF reader now runs on Gemmi version 0.6.2.
- A preliminary API has been added to provide reaction automapping ability, based on the maximum common substructure, enhancing reaction support in **OEChem TK**.
- Python toolkit packaging has been improved to use pyproject.toml to declare metadata and dependencies.

Supported Platforms

Package	Versions	Linux	Windows	macOS (x64 and arm64)
Python	3.9, 3.10, 3.11	RHEL8/9, Ubuntu20/20-ARM/22	Win10/11	12, 13
C++		RHEL8/9, Ubuntu20/20-ARM/22	Win10/11 (VS2022)	12, 13
Java	8, 11	RHEL8/9, Ubuntu20/20-ARM/22	Win10/11	12, 13
C#			Win10/11 (VS2022)	

Related TK Versions

Toolkit	Version
OEChem TK	4.0.0
OEDepict TK	2.5.3
OEDocking TK	4.3.0
FastROCS TK	2.2.5
OEFF TK	2.7.0
Grapheme TK	1.5.0
Graphsim TK	2.5.7
Lexichem TK	2.9.0
OEMedChem TK	1.2.1
MolProp TK	2.6.3
Omega TK	5.0.0
Quacpac TK	2.2.3
Shape TK	3.6.1
Sitehopper TK	2.0.4
Spicoli TK	1.5.7
Spruce TK	1.5.3
Szmap TK	1.6.7
Szybki TK	2.7.0
Zap TK	2.4.6

8.1.2 General Notices

- Support for MacOS arm64 Java is now available.
- Python toolkits now use pyproject.toml to declare their metadata and dependencies, and no longer use the deprecated setup.py method.
- Python 3.11 is now fully supported on Windows.
- This is the last release to support MacOS 12. Support for MacOS 14 will be added in the next release.
- Python 3.12 will not be fully supported until the next release.
- This is the last release to support Python 3.9

8.2 OEToolkits 2023.1

8.2.1 OEToolkits 2023.1.1

The **OEToolkits 2023.1.1** is a bug-fix of the **OEToolkits 2023.1.0** release.

- OEFF TK 2.5.2.1
- Szybki TK 2.6.0.1

8.2.2 OEToolkits 2023.1.0

Release Highlights

Supported Platforms

Package	Versions	Linux	Windows	macOS (x64 and arm64)
Python	3.9, 3.10, 3.11	RHEL8/9, Ubuntu20/20-ARM/22	Win10/11 [no python 3.11]	12, 13
C++		RHEL8/9, Ubuntu20/20-ARM/22	Win10/11 (VS2017/19/22)	12, 13
Java	8, 11	RHEL8/9, Ubuntu20/20-ARM/22	Win10/11	12, 13
C#			Win10/11 (VS2017/19/22)	

Related TK Versions

Toolkit	Version
OEChem TK	3.4.0
OEDepict TK	2.5.2
OEDocking TK	4.2.1
FastROCS TK	2.2.4
OEFF TK	2.5.2
Grapheme TK	1.4.7
Graphsim TK	2.5.6
Lexichem TK	2.8.1
OEMedChem TK	1.2.0
MolProp TK	2.6.2
Omega TK	4.2.2
Quacpac TK	2.2.2
Shape TK	3.6.0
Spicoli TK	1.5.6
Spruce TK	1.5.2
Szmap TK	1.6.6
Szybki TK	2.6.0
Zap TK	2.4.5

8.2.3 General Notices

8.3 Release Highlights 2022.2

8.3.1 OEToolkits 2022.2.2

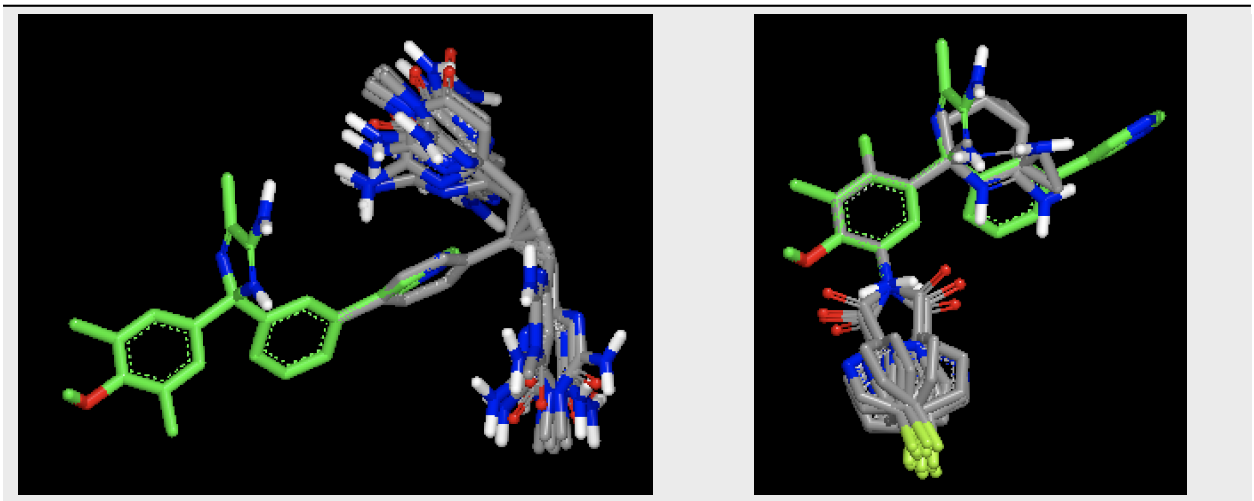
The **OEToolkits 2022.2.2** is a bug-fix of the **OEToolkits 2022.2.1** release.

8.3.2 OEToolkits 2022.2.1

8.3.3 MCS based fix during OMEGA Conformer Generation

The ability to constrain a fragment of the molecules during conformer generation with torsion driving in OMEGA and the OMEGA toolkit has been extended to include fixing based on maximum common subgraph (MCS) match. In this mode of fixing, the MCS between the template `fixmol` and the generated molecule is determined, and subsequently that common subgraph portion of the generated molecule is fixed to the proved conformer of the `fixmol`.

The MCS based fixing during conformer generation can be turned on by setting `OEConfFixOptions.SetFixMCS` to `true` for use in the OMEGA toolkit. Additional API points have also been added to allow control during MCS search. A new flag `-fixMCS` is available in the OMEGA application to facilitate this.



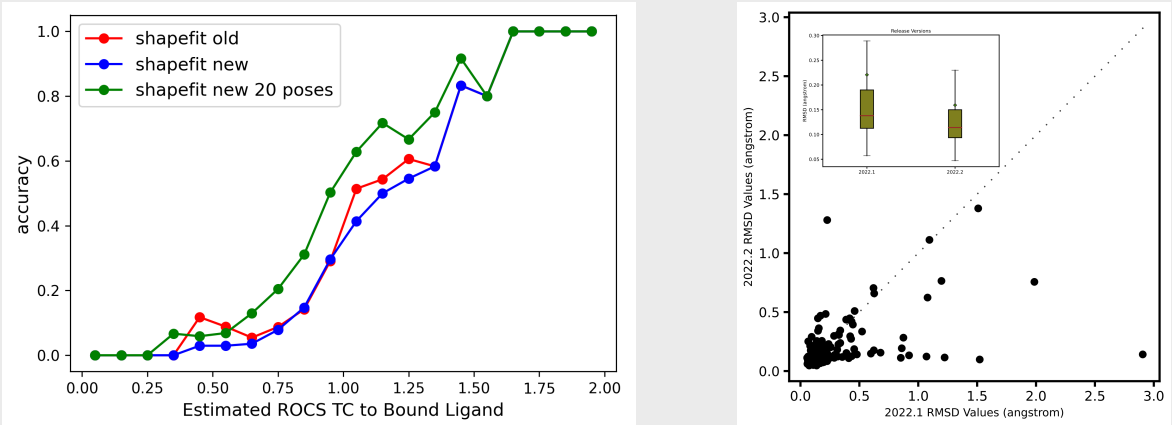
OMEGA generated conformers for two different molecules, with MCS based fix against a single bound ligand pose. Figure shows that with MCS based fix, two different fragments are fixed during conformer generation of the two different molecules.

8.3.4 New ShapeFit Algorithm for Pose Prediction

The algorithm for pose prediction with the `OEPositMethod.SHAPEFIT` method has been modified. The new algorithm simultaneously optimizes the shape and chemical similarity between the fit molecule and bound ligand, and intra-molecular force-field energies of the fit molecule. The new simultaneous optimization algorithm replaces the previous adiabatic optimization algorithm and is more robust. The new algorithm also offers flexibility to use different forcefields and uses the `OELigandFFType.SAGE` force field by default. Another advantage of the new algorithm is that it is capable of producing multiple distinct poses when desired. The new algorithm is automatically reflected in both the OEPosit toolkit and the POSIT application. `ShapeFit` is also now available in the OEDocking TK as an independent API OEShapeFit.

Cross-docking experiments based on the 22 diverse kinase types presented in [Tuccinardi-2010] shows that `ShapeFit` accuracy is unaffected between the previous and the new algorithm (when accuracy is measured as RMSD <=2.0). A balanced cross-docking set of 20,000 points, where points are equally distributed over the various similarity regions

and kinase types, was used. However, we see clear indication that generating multiple poses helps increase the pose prediction accuracy. A separate self-docking study based on the highly trustworthy iridium dataset ([Warren-2012]) with 280 ligand-protein structures shows that the new algorithm predicts poses that are closer to the x-ray crystallographic poses.



Left: Comparison of ShapeFit accuracy in a cross-docking experiment between the existing and the new algorithm. The x-axis shows the tanimotocombo similarity between the ligand and the bound ligand present in protein complex. The Y-axis shows the fraction of accurately predicted poses based on being within 2 angstroms of the experimental pose. Right: Comparison of ShapeFit accuracy in a self-docking experiment, between the existing and the new algorithm. Axes show the RMSD of generated poses with respect to the experimental pose.

8.3.5 Protein-ligand Optimization with ff14SB forcefield and PB solvent model

Protein-ligand Optimization with `ff14SB-OpenFF` force field and Poisson-Boltzmann (PB) solvent model is now available both in SZYBKI and the SZYBKI toolkit.

In the **SZYBKI TK**, the functionality is available through the `OEFixedProteinLigandOptimizer` and the `OEFlexProteinLigandOptimizer` APIs, and can be enabled by setting `OESolventModel.PB` as the value in `OEProteinLigandOptOptions.SetSolventModel`. In **SZYBKI** both the OptLigandInDU and OptimizeDU applications now accept `pb` as value for `-solventModel`.

Optimization of 244 highly trustworthy x-ray structures of protein-ligand complexes from the iridium dataset ([Warren-2012]) with and without PB solvation and with `ff14SB-Sage` forcefield shows better accuracy for the optimized ligand poses when PB solvation is turned on. Accuracy is measured as the RMSD between the crystallographic ligand pose and the optimized pose. On an average PB solvation optimized structures have an RMSD of 0.31 angstroms, compared to a value of 0.55 angstroms when explicit solvation is not turned on. A paired T-test suggests that the average improvement of 0.24 angstroms is statistically significant with a p value of 0.00001. Turing on the solvation effects also removes any significant deviations (RMSD > 2 angstroms) of pose, as was seen for a few cases when optimization was performed in vacuum.

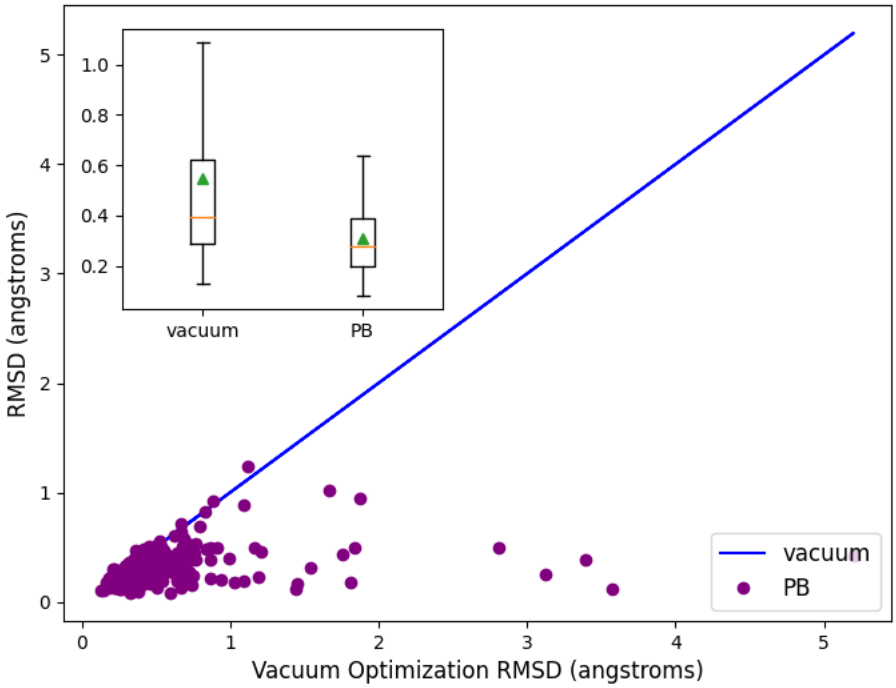
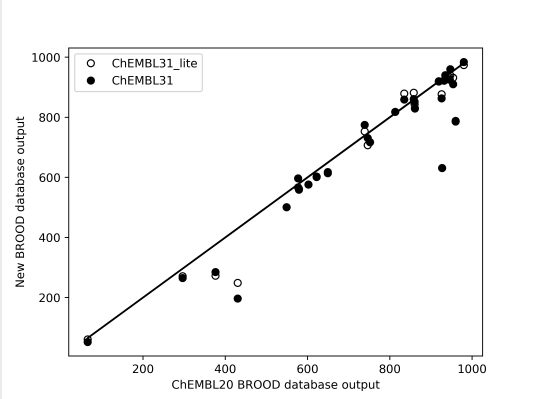


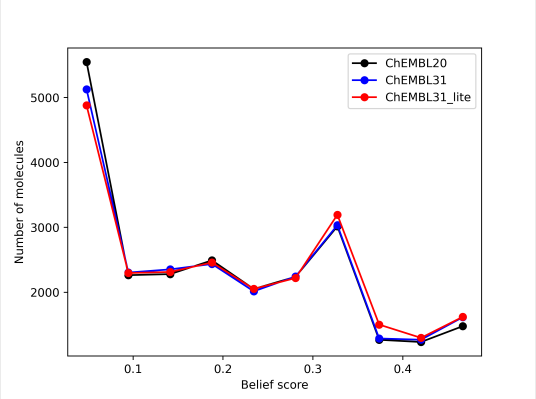
Fig. 1: **Comparison of RMSD for Flexible protein-ligand optimization of x-ray structures with and without PB solvation and with ff14SB-Sage forcefield.**

8.3.6 New BROOD Fragment Databases

Two new BROOD fragment databases, `brood-database-ChEMBL31` and `brood-database-ChEMBL31_lite`, built from the latest version of ChEMBL, have been generated and are made available. The `brood-database-ChEMBL31` contains all possible fragments up to 3 attachment points, whereas the `brood-database-ChEMBL31_lite` is curated to prioritize fragments with medicinal relevance.

Limited validation, with 29 different queries across 14 molecules, shows that the number of hits of drug-like molecules obtained from the new databases are within 1% of those generated from the existing database `brood-database-chembl-3.0.0`, when a maximum of 1000 hits are requested. Comparison of belief score, a measure of potential activity of the generated hits, also shows that the hits generated from the new databases are at least as good as those generated from the existing database.





Left: Comparison of the number of hits of druglike molecules using the brood-database-chembl-3.0.0 (ChEMBL20), brood-database-ChEMBL31 (ChEMBL31) and brood-database-ChEMBL31_lite (ChEMBL31_lite) fragment databases. Right: Comparison of the belief score distribution of all the hits generated using the ChEMBL20, ChEMBL31 and ChEMBL31_lite fragment databases.

Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.7, 3.8, 3.9, 3.10	RHEL7/8, Ubuntu20/20-ARM/22	Win10/11	11, 12
C++		RHEL7/8, Ubuntu20/20-ARM/22	Win10/11 (VS2017/19/22)	11, 12
Java	1.8, 11	RHEL7/8, Ubuntu20/20-ARM/22	Win10/11	11, 12
C#			Win10/11 (VS2017/19/22)	

Related TK Versions

Toolkit	Version
OEChem TK	3.3.1
OEDepict TK	2.5.2
OEDocking TK	4.2.0
FastROCS TK	2.2.3
OEFF TK	2.5.1
Grapheme TK	1.4.6
Graphsim TK	2.5.5
Lexichem TK	2.8.0
OEMedChem TK	1.1.7
MolProp TK	2.6.1
Omega TK	4.2.1
Quacpac TK	2.2.1
Shape TK	3.5.1
Spicoli TK	1.5.5
Spruce TK	1.5.1
Szmap TK	1.6.5
Szybki TK	2.5.1
Zap TK	2.4.4

General Notices

- Support for macOS 12 has been added.
- This is the last release to support macOS 11. Support for macOS 13 will be added in the next release.
- Support for Ubuntu22 has been added. Support for Ubuntu18 has been dropped.
- This is the last release to support RHEL7. Support for RHEL9 will be added in the next release.
- This is the last release to support Visual Studio 2017 and Visual Studio 2019.

The detailed release notes consist of these highlights plus the individual release notes and the release history.

8.4 Release Highlights 2022.1

8.4.1 Enhanced Stereo Information Support

The 2022.1 release adds support for enhanced stereochemistry designations in molecules, defining relationships between stereocenters. Enhanced stereochemistry information on any molecule is honored by `OEFlipper` during stereo enumeration when asked to do so by setting appropriate options. Enhanced stereochemistry support has also been extended to both the OMEGA and the Flipper applications.

Support for CXSMILES as a valid molecular input format via `OEFormat.CXSMILES` has been added. CXSMILES is specifically intended to support the import of Enamine download structure files containing enhanced stereochemistry designations. While many other features are possibly provided in the CXSMILES appendix information, only the enhanced stereochemistry group information is supported at this time. Please contact support@eyesopen.com with specific needs for additional CXSMILES features to allow prioritization in future releases.

OMEGA and **Flipper** applications can now accept molecule files in CXSMILES format.

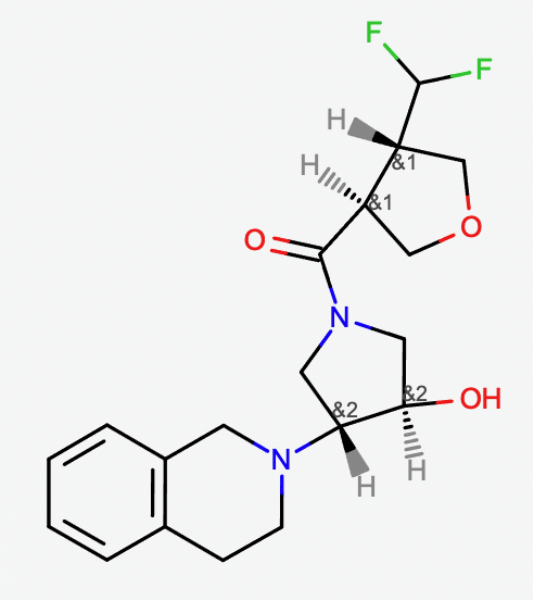


Fig. 2: **Molecule with enhanced stereo designation. The molecule has 16 stereoisomers when stereocenters are treated independently and only 4 stereoisomers when the enhanced stereo is enabled.**

8.4.2 Sheffield Solvation Model for Proteins

The simple two parameter Sheffield Solvation Model, as introduced by Grant et al. ([Grant-2007]) for calculating the electrostatic component of molecular solvation energy, is a semi-empirical model that performs well for small molecules and requires few computational resources. This model does not work well for macromolecules like proteins and complexes, where many of the atoms are hidden or only partially exposed to the solvent.

A modified three parameter Sheffield Solvation Model is parameterized and implemented in this release, where solvent exposure is treated by correcting the effective atomic radii with the introduction of a third additional parameter. The new model is available as part of the `OESheffield` API in the **OEFF toolkit**. The model is also available for use through the `OEFixedProteinLigandOptimizer` and the `OEFlexProteinLigandOptimizer` APIs in the **SZYBKI toolkit**, and the OptLigandInDU and OptimizeDU applications in **SZYBKI**.

Electrostatic solvation energies for protein conformers are calculated using both the new Sheffield Solvation Model and the Poisson–Boltzmann (PB) model, and the difference in solvation energies between conformers of the same protein is compared between the two models. A set of 232 NMR proteins and peptide structures were used for this study that produced 41491 solvation energy differences between pairs of conformations. The results are compared in the image below.

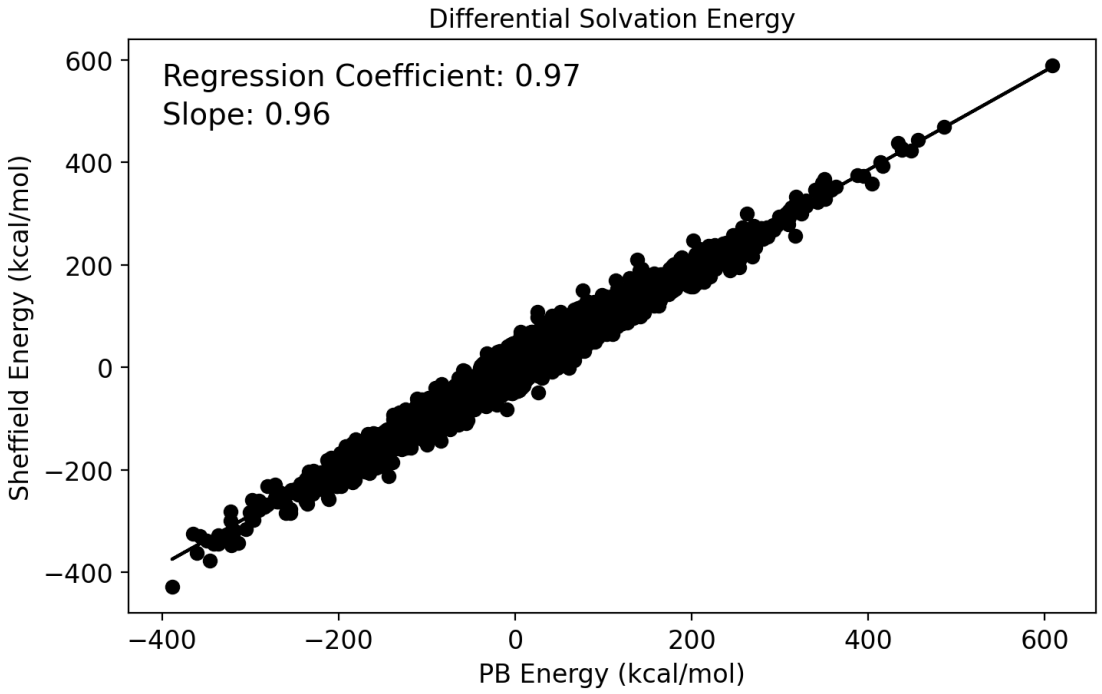


Fig. 3: **Comparison between Sheffield and PB of differential electrostatic solvation energies between protein conformers.**

8.4.3 Spruce Filter and Design Unit Validation

New functionality for preprocessing inputs and validation of the output `Design Units` has been added to **Spruce TK** in the form of two preliminary API classes `OESpruceFilter` and `OEValidateDesignUnit`. These classes are also used by the **SPRUCE** application to check and fix, if possible, inputs and to validate the resulting outputs.

`OESpruceFilter` performs a standardization and a filter check structures to use before prepping this structure in **SPRUCE**. It automatically checks and fixes invalid component names, invalid residue states,and invalid metal bonds. It also checks for invalid CRYST1 from the header file, low overlap between the electron density map and the structure, invalid sequence alignment, blank chain IDs, and invalid alternate locations. If the structure does not pass these tests, Spruce Filter will fail the structure.

The `OEValidateDesignUnit` function performs a check on the design units generated by **SPRUCE**. It will report if the provided DU contains partial side chains, missing loops, implicit hydrogens, incorrect partial charges, termini that are not capped properly, heavy atoms overlapping with each other, and broken chains.

8.4.4 Multistate Heuristic pKa Model

The multistate heuristic pKa model is an extension of the existing pKa model in **OpenEye** software. The existing model generates only one probable ionization state based on pKa estimated from about 80 rules. With the development of this multistate pKa model, we are enhancing the implementation with many more unique functional patterns in the form of pKa rules, identified from OpenEye's internal pKa database, allowing generation of all ionization states possible at physiological pH.

The model described herein uses SMARTS-based heuristics to assess the pKa of acidic or basic functional groups of molecules. For all atoms of a query molecule, pKa is assessed as it belongs to one of three classes: Acidic (pKa < 6.4), Basic (pKa > 8.4), and Neutral (6.4 < pKa < 8.4). According to the assessed pKa class for each atom, it is either ionized and/or un-ionized at pH 7.4. Any group with Acidic or Basic pKa range will result in one state, ionized or

un-ionized and a group with Neutral pKa range will result in both states, ionized and un-ionized. Based on how many such functional groups there are in the molecule, and how many states are generated for them, multiple molecules can be generated for all possible combinations of all possible ionization states of acid/base groups.

The multistate heuristic pKa model is currently only available in **Quacpac TK** as preliminary API classes `OEMultistatepKaModel` and `OEMultistatepKaModelOptions`.

Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.7, 3.8, 3.9, 3.10	RHEL7/8, Ubuntu18/20/20-ARM	Win10/11	10.15, 11, 12
C++		RHEL7/8, Ubuntu18/20/20-ARM	Win10/11 (VS2017/19/22)	10.15, 11, 12
Java	1.8, 11	RHEL7/8, Ubuntu18/20/20-ARM	Win10/11	10.15, 11, 12
C#			Win10/11 (VS2017/19/22)	

Related TK Versions

Toolkit	Version
OEChem TK	3.3.0
OEDepict TK	2.5.0
OEDocking TK	4.1.2
FastROCS TK	2.2.2
OEFF TK	2.5.0
Grapheme TK	1.4.5
Graphsim TK	2.5.4
Lexichem TK	2.7.5
OEMedChem TK	1.1.6
MolProp TK	2.6.0
Omega TK	4.2.0
Quacpac TK	2.2.0
Shape TK	3.5.0
Spicoli TK	1.5.4
Spruce TK	1.5.0
Szmap TK	1.6.4
Szybki TK	2.5.0
Zap TK	2.4.3

General Notices

- Support for macOS 12 has been added. macOS 10.14 is no longer supported.
- This is the last release to support macOS 10.15.
- A GCC 7.3 C++ toolkit package has been added for RHEL7.
- Support for Windows 11 has been added.
- Support for Visual Studio 2022 has been added.

- Support for Apple M1 architecture has been added for C++ and Python toolkits.
- Support for Linux ARM architectures has been added.
- This is the last release to support Ubuntu 18.04. Support for Ubuntu 22.04 will be added in the next release.

For detailed release notes, see *Version 2022.1.0 Release Notes*.

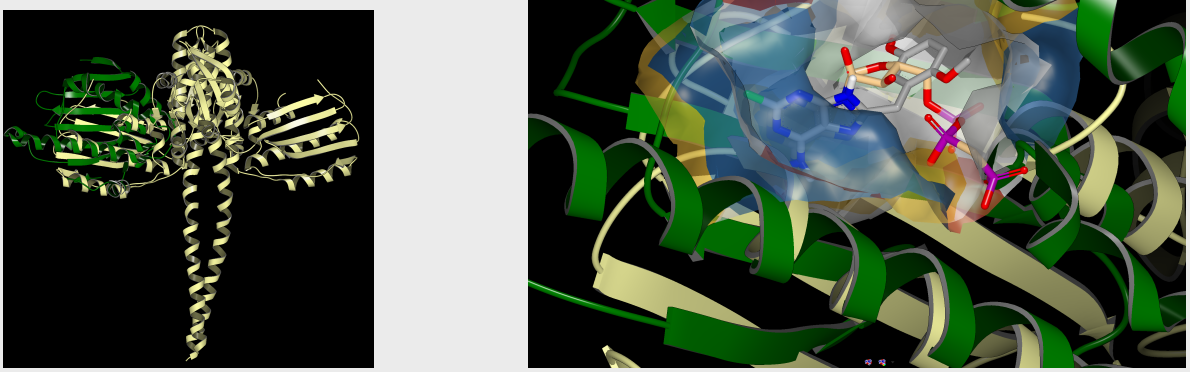
8.5 Release Highlights 2021.2

8.5.1 SiteHopper: New Toolkit for Protein Binding Site Comparison

The 2021.2 release introduces **SiteHopper TK**, a new toolkit for searching a database containing design units for proteins with similar binding sites to a query design unit. The new **SiteHopper TK** provides toolkit level access to the **SiteHopper** application functionality released in 2021.1.

Besides providing toolkit level access, the 2021.2 release also brings several other improvements to **SiteHopper**. Efficiency of database building in **SiteHopper** is enhanced with the use of a multi-threaded approach. Furthermore, search is now possible on the CPU, enabling cross platform support of the toolkit and application. On Linux machines with an NVIDIA graphics card, the GPU is used as a pre-screening method, speeding up the search dramatically. A feature to exclude results based on a similarity threshold has been added that allows for more diversity in the protein targets with similar binding sites. The database storage format has also been revised.

An example of a **SiteHopper** produced hit is shown in the figure below. 1UYG is a structure of human heat shock protein 90-alpha, bound to 8-(2,5-dimethoxy-benzyl)-2-fluoro-9H-purin-6-ylamine. 5IUN is a structure of the DesK-DesR complex, bound to AMP-PCP. The image on the left shows the overlay of 1UYG (green) and 5UIN (light yellow), zoomed out to show the major structural differences between the two proteins. The image on the right zooms in on the binding sites, with a surface showing the type of residues present in each binding site. Blue represents acidic, red represents basic, yellow represents polar, and white represents non polar. Despite a sequence similarity of only 46%, 1UYG and 5IUN may be targetable by similar ligands, as they have very similar steric and electrostatic properties in their binding sites.



1UYG human hsp 90 (green) overlaid with 5UIN N-formyltransferase (yellow), full struture view (left), binding site view (right).

8.5.2 SZYBKI: OpenFF-Sage force field support

The latest force field, **Sage**, from the Open Force Field Initiative has been added to **OEFF TK**, **Szybki TK**, and **SZYBKI**. The major features of the **Sage** force field are outlined here.

New force field parameter classes OESageParams and OEFF14SBSageParams have been added to the toolkit to support the **Sage** parameters. Three new force field classes, OESage, OEFF14SBSage, and OEFF14SBSageComplex, have also been introduced. The **Sage** force field is now the default for free energy analysis of conformation ensembles with OEFreeFormConf, and the corresponding application **Freeform**. The protein-ligand forcefield, **FF14SBSage**, is now the default for protein-ligand optimization in OEFixedProteinLigandOptimizer and OEFlexProteinLigandOptimizer, OptimizeDU, and OptLigandInDU.

The **Sage** force field can be used in **SZYBKI** for ligand calculations, with the `-ff sage_openff2.0.0` command line option, and for calculations involving proteins using `-ff ff14sb_sage`.

8.5.3 OEChem: MMCIF and CIF writers

Functionality to write Crystallographic Information Files (CIF and MMCIF) has been added to the **OEChem TK** in the 2021.2 release. Similar to writing other molecule file formats in **OEChem TK**, CIF and MMCIF files can be written using the high level `OEWriteMolecule` function. Low level functionality for advanced users is also provided through a new `OEWriteCIFFile` function.

If the molecule does not contain residue information, it is written as a small molecule CIF file if the data is available to do so (i.e. space group information). If residue information is available, the writer checks the number of residues and the molecule is either written as a chemical component dictionary entry (in the event there is a single residue) or as a macromolecular CIF file (MMCIF).

The small molecule CIF reader has also been improved for performance and robustness, and round-tripping these files has been tested and validated using the checkCIF facility from the International Union of Crystallography (IUCR). The MMCIF parser based on Gemmi has also been updated, and items parsed from PDB metadata have been expanded for proper processing of the biomolecules with **Spruce**.

Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.7, 3.8, 3.9	RHEL7/8, Ubuntu18/20	Win10	10.14, 10.15, 11
C++		RHEL7/8, Ubuntu18/20	Win10 (VS2017, VS2019)	10.14, 10.15, 11
Java	1.8, 11	RHEL7/8, Ubuntu18/20	Win10	10.14, 10.15, 11
C#			Win10 (VS2017, VS2019)	

Related TK Versions

Toolkit	Version
OEChem TK	3.2.0
OEDepict TK	2.4.7
OEDocking TK	4.1.1
FastROCS TK	2.2.1
OEFF TK	2.4.0
Grapheme TK	1.4.4
Graphsim TK	2.5.3
Lexichem TK	2.7.4
OEMedChem TK	1.1.5
MolProp TK	2.5.7
Omega TK	4.1.2
Quacpac TK	2.1.3
Shape TK	3.4.3
Spicoli TK	1.5.3
Spruce TK	1.4.0
Szmap TK	1.6.3
Szybki TK	2.4.0
Zap TK	2.4.2

General Notices

- This is the last release to support macOS 10.14. Support for macOS 12 will be added in the next release.
- A gcc 7.3 C++ toolkit package has been added for RHEL7.
- Support for Windows 11 will be added in the next release.

8.6 Release Highlights 2021.1

8.6.1 OEDocking TK: Receptors in Design Unit

OEReceptor as an integral part of an OEDesignUnit, with properly prepped structures and contained in an `OEDU` file, is fully integrated in all workflows. Receptors in an OEDesignUnit can now be created using the **Make Receptor** GUI application, as well as the **SPRUCE** and **ReceptorInDU** command-line applications.

The **Make Receptor** GUI application has been extended to be a fully functional DesignUnit builder in addition to building receptors. A new DesignUnit can be created from a `PDB/MMCIF` file (with or without an associated `MTZ` file), or by combining molecules from various molecule files of any format. **SPRUCE** options are exposed through a graphical interface giving users control over the level of preparation the structure should go through. Existing receptors, both in `OEDU` or `OEB` format, can also be opened and edited as desired in **Make Receptor**.

`OEMakeReceptor` now automatically detects protein constraints. The detected protein constraints are kept disabled by default, and can be completely turned off if desired, functionality that was previously only available in the **Make Receptor** application. Both the detected constraints and any custom constraints can be enabled and edited via the `OEMakeReceptor` interface, or using commands from OEReceptorProteinConstraint and OEReceptorCustomConstraints classes in the toolkit.

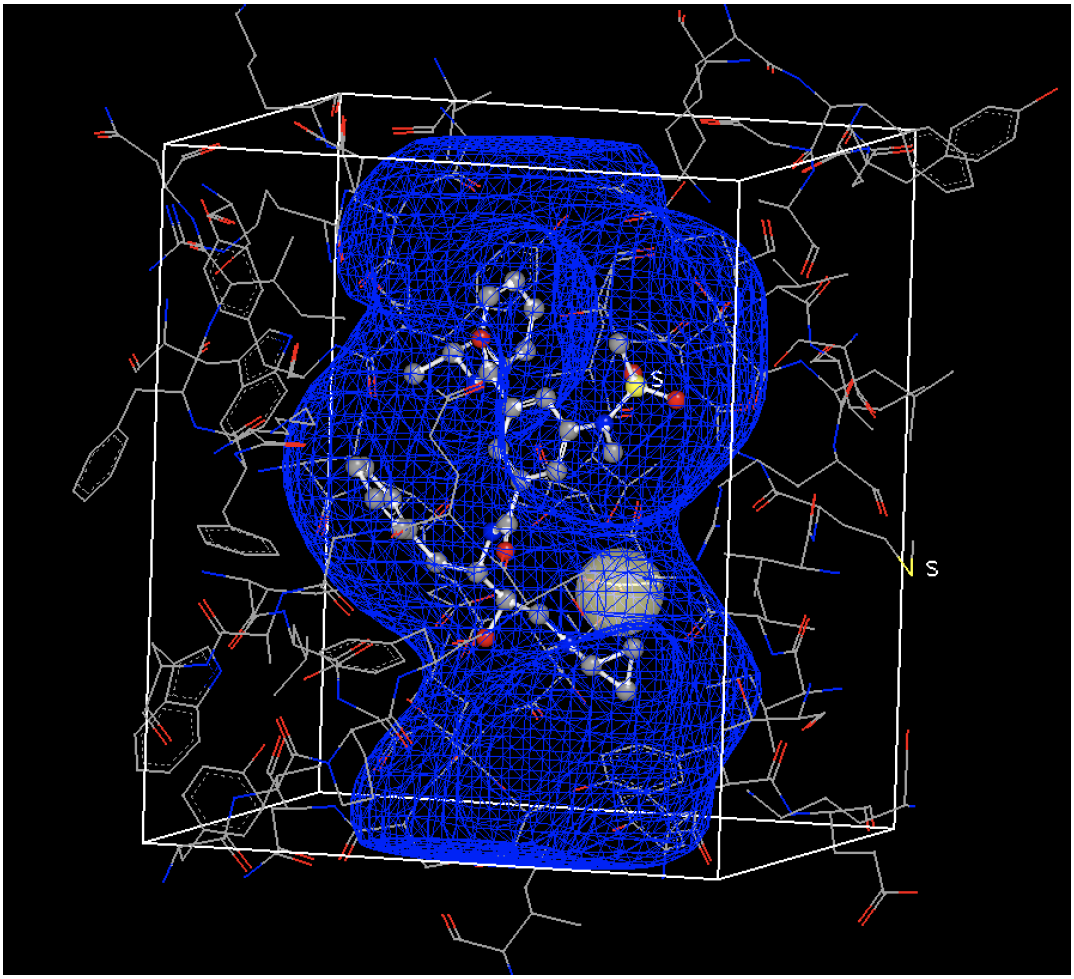


Fig. 4: **OEDocking receptor for the 3TPP BACE1 Complexed with Inhibitor with target structure, bound ligand, and a protein constraint. Displayed in the Make Receptor viewer.**

Having the receptors as part of the OEDesignUnit with properly prepped structures makes it easier for the docked and posed structures to be used in further downstream modeling, which is especially necessary when working with protein force fields like `FF14SB`. The new receptors with properly prepped protein also enabled use of modern force fields like `FF14SB` and `Parsley` for structure optimization in flexible OEPosit.

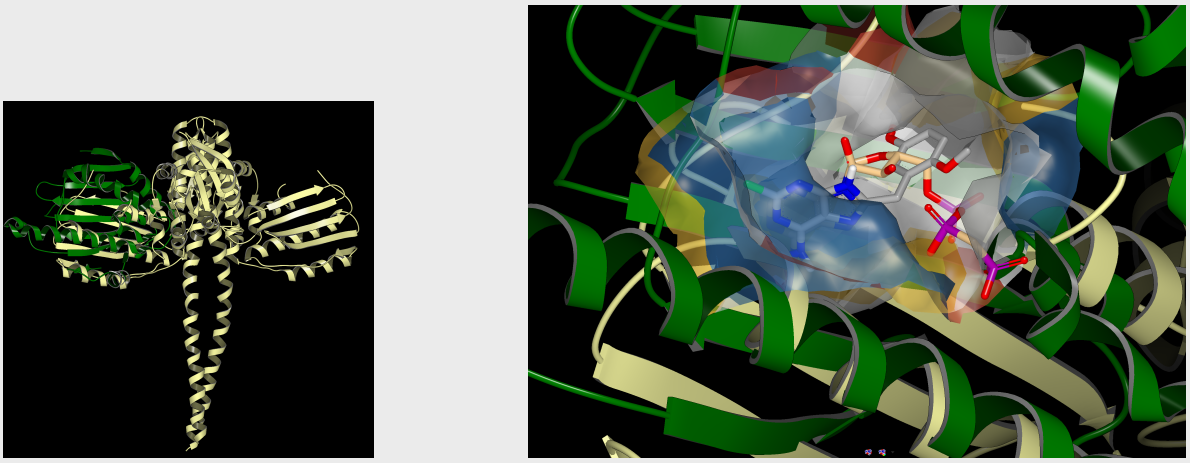
8.6.2 OESiteHopper: Application Suite for Rapid Protein Binding Site Comparison

The 2021.1 release introduces **SiteHopper**, which can search a database containing hundreds of thousands of proteins in a few minutes for proteins with similar binding sites to a query protein. **SiteHopper** will output the most similar biomolecules, overlaid by binding site, for visualization and analysis. A visual representation of the binding site is also output. **SiteHopper** can also create databases for searching from a series of biomolecular structures in `OEDU` format.

SiteHopper finds proteins with similar binding sites which searching by sequence similarity would overlook. An example is shown in the figure below. 1UYG is a structure of human heat shock protein 90-alpha, bound to 8-(2,5-dimethoxy-benzyl)-2-fluoro-9H-purin-6-ylamine. 5IUN is a structure of the DesK-DesR complex, bound to AMP-PCP.

The image on the left shows the overlay of 1UYG (green) and 5UIN (light yellow), zoomed out to show the major structural differences between the two proteins. The image on the right zooms in on the binding sites, with a surface showing the type of residues present in each binding site. Blue represents acidic, red represents basic, yellow represents polar, and white represents non polar. Despite a sequence similarity of only 46%, 1UYG and 5IUN may be targetable

by similar ligands, as they have very similar steric and electrostatic properties in their binding sites.



1UYG human hsp 90 (green) overlaid with 5UIN N-formyltransferase (yellow), full struture view (left) and binding site view (right).

8.6.3 VIDA 5: Major Update and OEDesignUnit Support

VIDA has undergone major changes including:

- Upgrade to Python3.
- Upgrade to Qt5 with PySide2 providing for user-interface improvements.
- Updated to run on the same platforms as OpenEye Toolkits and Applications.

This allows us to release **VIDA** along side the toolkits and applications in our twice-yearly releases.

In addition, **VIDA** now has support for MMCIF files and improvements to I/O from other file format types. **VIDA** can also read prepared protein structures in OEDesignUnit format produced with **SPRUCE** and **Make Receptor** along with the receptor objects that are now stored on these design units.

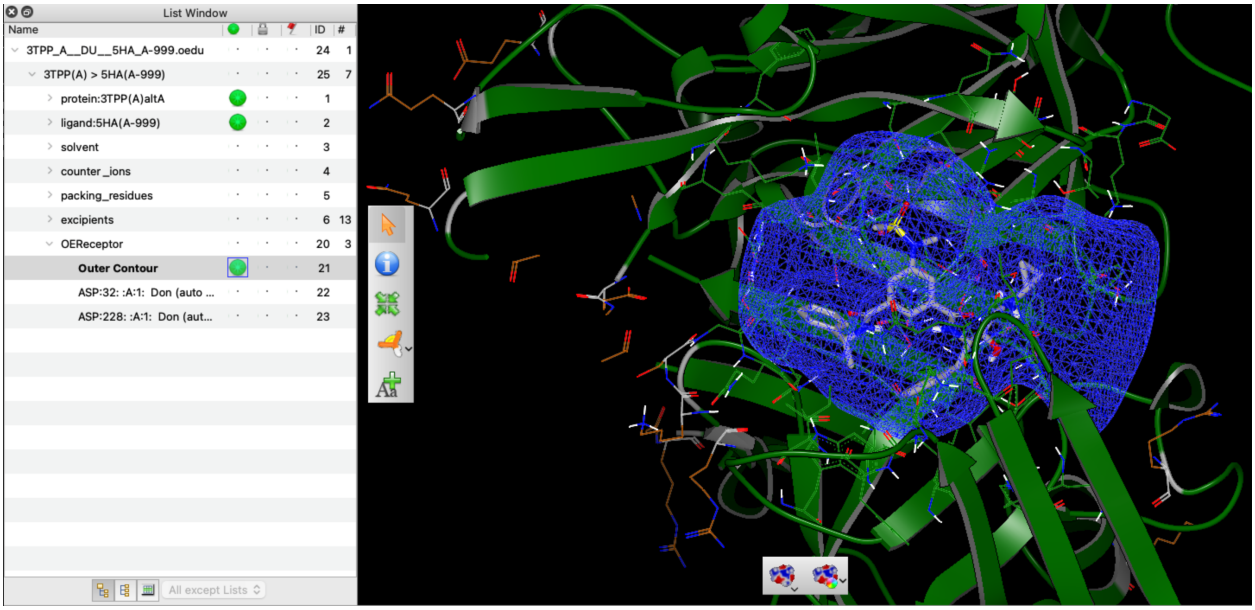


Fig. 5: **VIDA DesignUnit Support with receptors showing the prepared beta-secretase structure 3TPP.**

8.6.4 Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.7, 3.8, 3.9	RHEL7/8, Ubuntu18/20	Win10	10.14, 10.15, 11
C++		RHEL7/8, Ubuntu18/20	Win10 (VS2017, VS2019)	10.14, 10.15, 11
Java	1.8, 11	RHEL7/8, Ubuntu18/20	Win10	10.14, 10.15, 11
C#			Win10 (VS2017, VS2019)	

8.6.5 Related TK Versions

Toolkit	Version
OEChem TK	3.1.1
OEDepict TK	2.4.6
OEDocking TK	4.1.0
FastROCS TK	2.2.0
OEFF TK	2.3.1
Grapheme TK	1.4.3
Graphsim TK	2.5.2
Lexichem TK	2.7.3
OEMedChem TK	1.1.4
MolProp TK	2.5.6
Omega TK	4.1.0
Quacpac TK	2.1.2
Shape TK	3.4.2
Spicoli TK	1.5.2
Spruce TK	1.3.0
Szmap TK	1.6.2
Szybki TK	2.3.1
Zap TK	2.4.1

8.6.6 General Notices

- C++ CUDA-enabled toolkits and features are now supported on RHEL8 and Ubuntu20.
- Support for GCC 9.x has been added. Support for GCC 4.x and GCC 5.x has been dropped.
- Support for Python 3.6 has been dropped.
- Support for VS2015 has been dropped.
- Support for MacOS 11 has been added. Support for MacOS 10.13 has been dropped.
- OpenEye applications and toolkits have not been optimized for the M1 Mac but run under Rosetta 2.

8.7 Release Highlights 2020.2

8.7.1 OEToolkits 2020.2.2

The **OEToolkits 2020.2.2** is a bug-fix of the **OEToolkits 2020.2** release. Click on the below links for more information about the bug fixes in this release:

- OEChem TK 3.1.0.2
- OEDocking TK 4.0.0.2
- Quacpac TK 2.1.1.2

8.7.2 Omega TK: New fragment library

A new enhanced fragment library has been created for use in **Omega TK**. The new fragment library consists of more than 500,000 fragments that have been generated using the MMFF force field.

This new library is designed to reduce the run time of large-scale conformer generation, and therefore is not built in to **Omega TK** but is available as a separate download. To run **Omega TK** with the enhanced library, simply use the OEOmega method `OEOmega.AddFragLib`.

The fragments have been collated from commonly used electronic databases, such as Enamine Building Blocks. Consequently, dramatic runtime performance improvements can be seen for the Enamine Real dataset, among others, since less time is spent building fragments on the fly. Runtime performance improvements can be seen for all datasets; the Platinum and Enamine Real sets show particular improvement and run ~50% faster (benchmarks run on Ubuntu 16 with Intel(R) Xeon(R) Gold 6128 CPU 3.40GHz and NVIDIA Tesla V100 GPU).

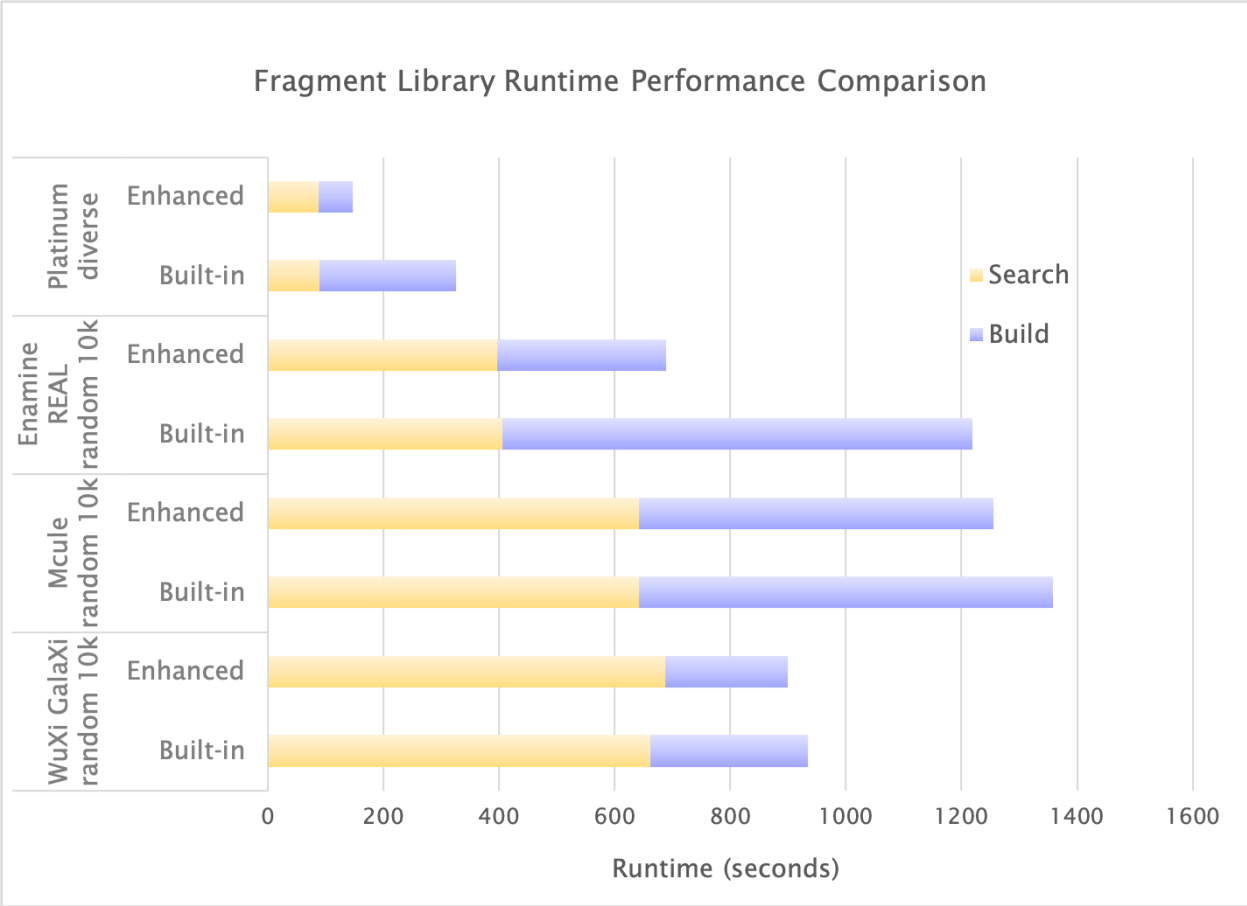


Fig. 6: **GPU-OMEGA runtime performance using the built-in fragment library compared to the new enhanced fragment library. Benchmark datasets are a filtered set of 2359 molecules of the Platinum dataset and 10,000 randomly chosen molecules of the Enamine Real, Mcule and WuXi databases.**

The enhanced fragment library generates the same high-quality conformers as the built-in library, as shown by the minimum RMSDs of the filtered Platinum dataset.

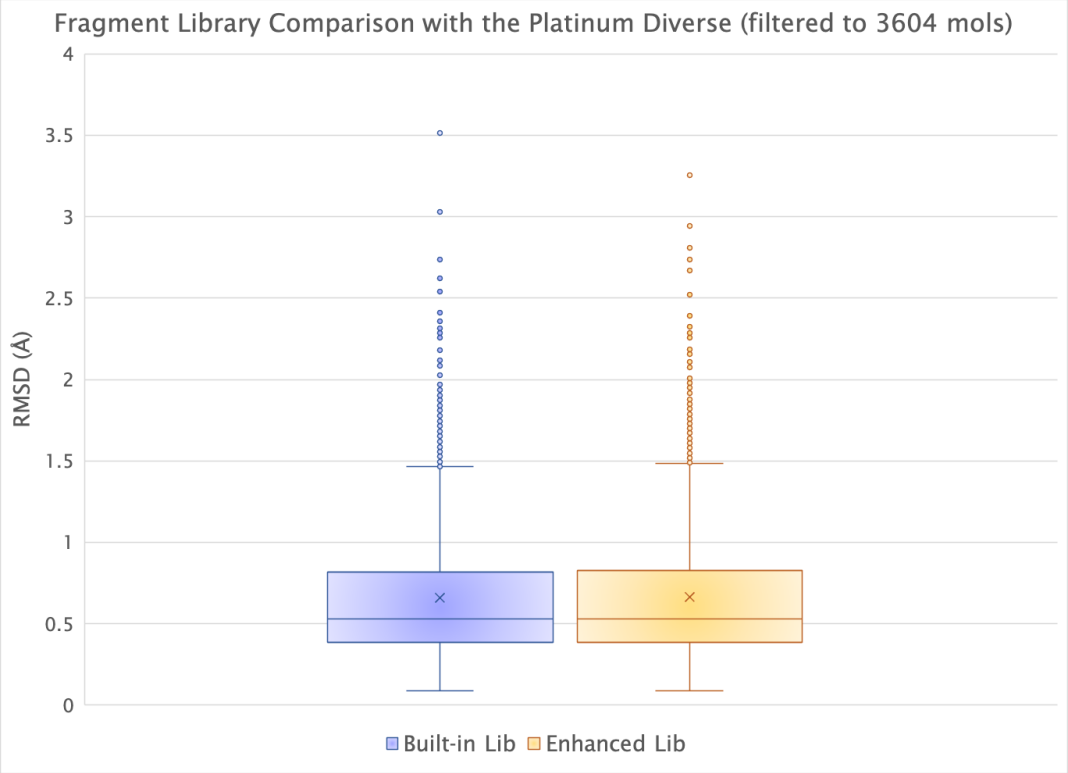


Fig. 7: **Box plots showing minimum RMSDs of a filtered set of 2359 molecules of the Platinum dataset with the built-in fragment library and the new enhanced fragment library.**

Datasets can be downloaded from:

- Mcule
- Enamine Real
- Platinum Dataset
- WuXi

We are eager to get your feedback on the new enhanced fragment library. Please share your experience with us at support@eyesopen.com.

8.7.3 OEDocking TK: Improved receptors

Receptors used in **OEDocking TK** classes, including **FRED**, **HYBRID**, and **POSIT**, have been improved to take advantage of properly prepared structures from **SPRUCE**. A new OEReceptor object has been introduced and is created and contained within an OEDesignUnit. Since an OEReceptor is now an integral part of an OEDesignUnit, it is saved into an `OEDU` file along with a design unit. There is no separate I/O for the receptors.

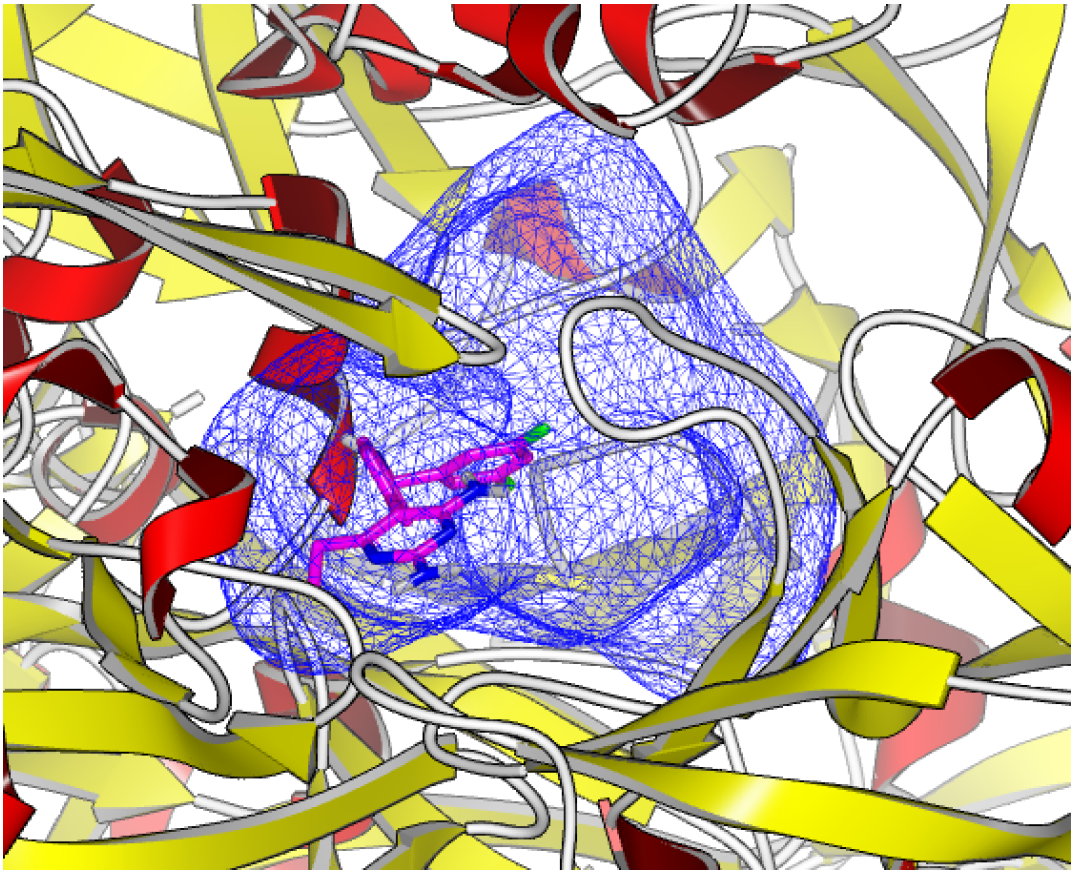


Fig. 8: **OEDocking receptor, along with the target structure and the bound ligand, for the 2IKO Human Renin Complexed with Inhibitor.**

Having the receptors as part of the design unit with properly prepped structures makes it easier for the docked and posed structures to be used in further downstream modeling, which is especially necessary when working with protein force fields like `FF14SB`. The new receptors with properly prepped protein also enabled use of modern force fields like `FF14SB` and `Parsley` for structure optimization in flexible OEPosit.

8.7.4 Szybki TK: A new protein force field

The `AMBER FF14SB` protein force field has been implemented in **OEFF TK**. This new force field has also been made available in **SZYBKI** for optimizing both protein and protein-ligand complexes.

`FF14SB` is the most widely used protein force field for molecular dynamics or any other force field-based calculations. It is also used in Orion molecular dynamics package, and is the community gold standard for such calculations.

8.7.5 Supported Platforms

Package	Versions	Linux	Windows	macOS
Python	3.6, 3.7, 3.8	RHEL7/8, Ubuntu18/20	Win10	10.13, 10.14, 10.15
C++		RHEL7/8, Ubuntu18/20	Win10 (VS2017, VS2019)	10.13, 10.14, 10.15
Java	1.8, 11	RHEL7/8, Ubuntu18/20	Win10	10.13, 10.14, 10.15
C#			Win10 (VS2015, VS2017, VS2019)	

8.7.6 Related TK Versions

Toolkit	Version
OEChem TK	3.1.0
OEDepict TK	2.4.5
OEDocking TK	4.0.0
FastROCS TK	2.1.0
OEFF TK	2.3.0
Grapheme TK	1.4.2
Graphsim TK	2.5.1
Lexichem TK	2.7.2
OEMedChem TK	1.1.3
MolProp TK	2.5.5
Omega TK	4.1.0
Quacpac TK	2.1.1
Shape TK	3.4.1
Spicoli TK	1.5.1
Spruce TK	1.2.0
Szmap TK	1.6.1
Szybki TK	2.3.0
Zap TK	2.4.0

8.7.7 General Notices

- Support for Ubuntu20 has been added. Ubuntu16 is no longer supported.
- C++ CUDA-enabled toolkits and features are now supported on RHEL8 and Ubuntu18.
- This is the last release to support GCC 4.x and GCC 5.x. GCC 7.4 will be the minimum GCC supported in the next release.
- This is the last release to support Python 3.6.
- This is the last release to support VS2015.

8.8 Release Highlights 2020.1

8.8.1 OEToolkits 2020.1.1

The **OEToolkits 2020.1.1** is a bug-fix of the **OEToolkits 2020.1** release.

- OEChem TK 3.0.0.6
- Omega TK 4.0.0.6

8.8.2 OEToolkits 2020.1

The **OEToolkits 2020.1** is a bug-fix of the **OEToolkits 2020.0** release.

- OEChem TK 3.0.0.5
- OEDocking TK 3.5.0.5

8.8.3 OEToolkits 2020.0

OpenEye Toolkits and **Applications** are now integrated into a single release cycle. This integration ensures that the tools delivered are more consistent across various product levels and, in keeping with **OpenEye** standards, are robust and efficient. In addition, the integration reduces release process management time, allows more time to deliver new features, and facilitates delivering bug-fix releases for critical issues when needed.

Omega TK: GPU-Omega - GPU-accelerated torsion driving

The 2020.0 release introduces **GPU-Omega** to the **Omega TK**. **GPU-Omega** provides accelerated torsion driving by utilizing a GPU (Graphics Processing Unit). **GPU-Omega** is integrated into the existing **Omega TK**; no additional user actions are required to make use of this new feature. **GPU-Omega** detects any supported Nvidia GPU on Linux platforms and accelerates torsion driving automatically unless explicitly suppressed through the option `OETorDriveOptions.SetUseGPU`. **GPU-Omega** is available with all sampling modes except `macrocycle`, which does not use the torsion driving method for conformer generation. For more details on **GPU-Omega**, including prerequisites and additional requirements for use with `dense` mode, please see GPU-Omega.

GPU-Omega performs at the same high level of accuracy as **Omega TK**, as shown below using a filtered subset of the Platinum dataset [Friedrich-2017] [Hawkins-2020]:

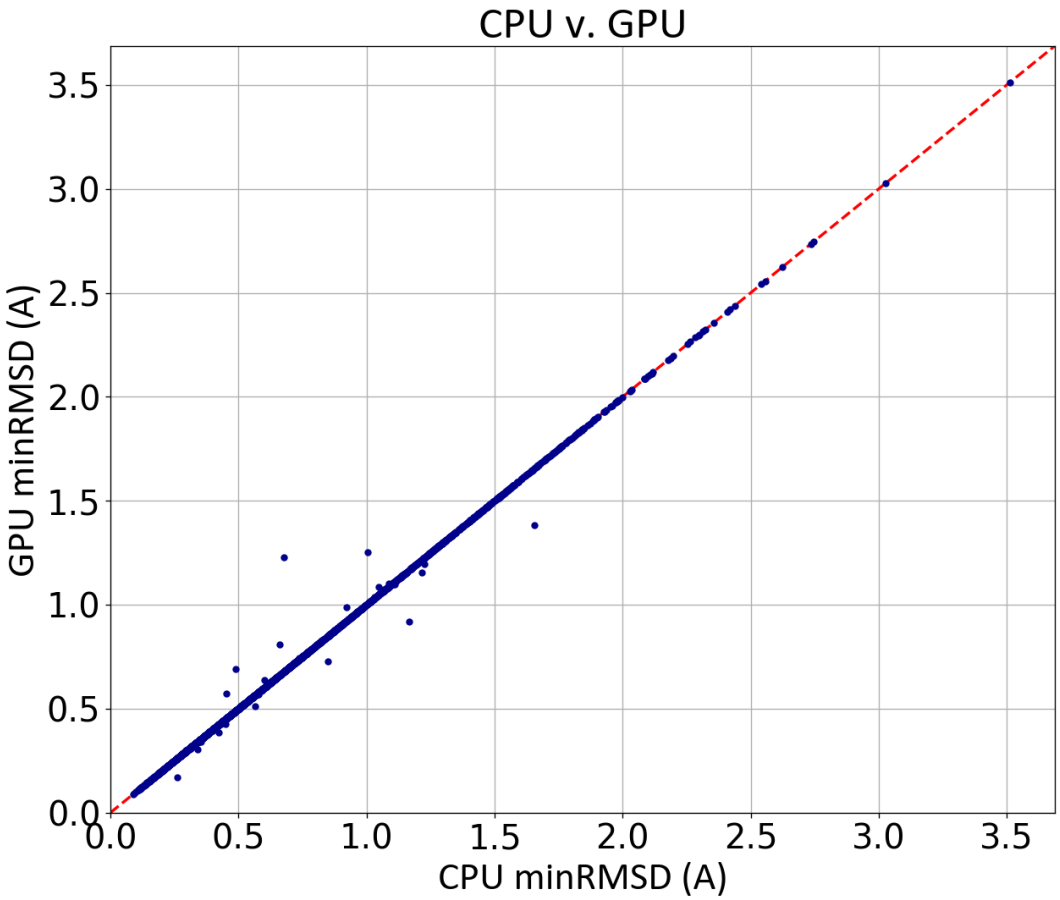


Fig. 9: **Comparison of the minimum RMSDs of CPU- and GPU-generated conformers against the experimental coordinates of a filtered subset of the Platinum dataset**

GPU-Omega was benchmarked against the CPU on AWS to demonstrate the achievable performance on a P3 instance, which houses an Nvidia Tesla V100 GPU. **GPU-Omega** was benchmarked with all supported sampling modes (`sparse`, `dense pose`, `rocs`, and `fastrocs`) using a subset of ~6000 molecules from the GSK TCAMs dataset [Gamo-2010]. The most impressive speed-up is seen with the `dense` sampling mode, with a reduction in time from 120 hours to less than 4 hours with **GPU-Omega**.

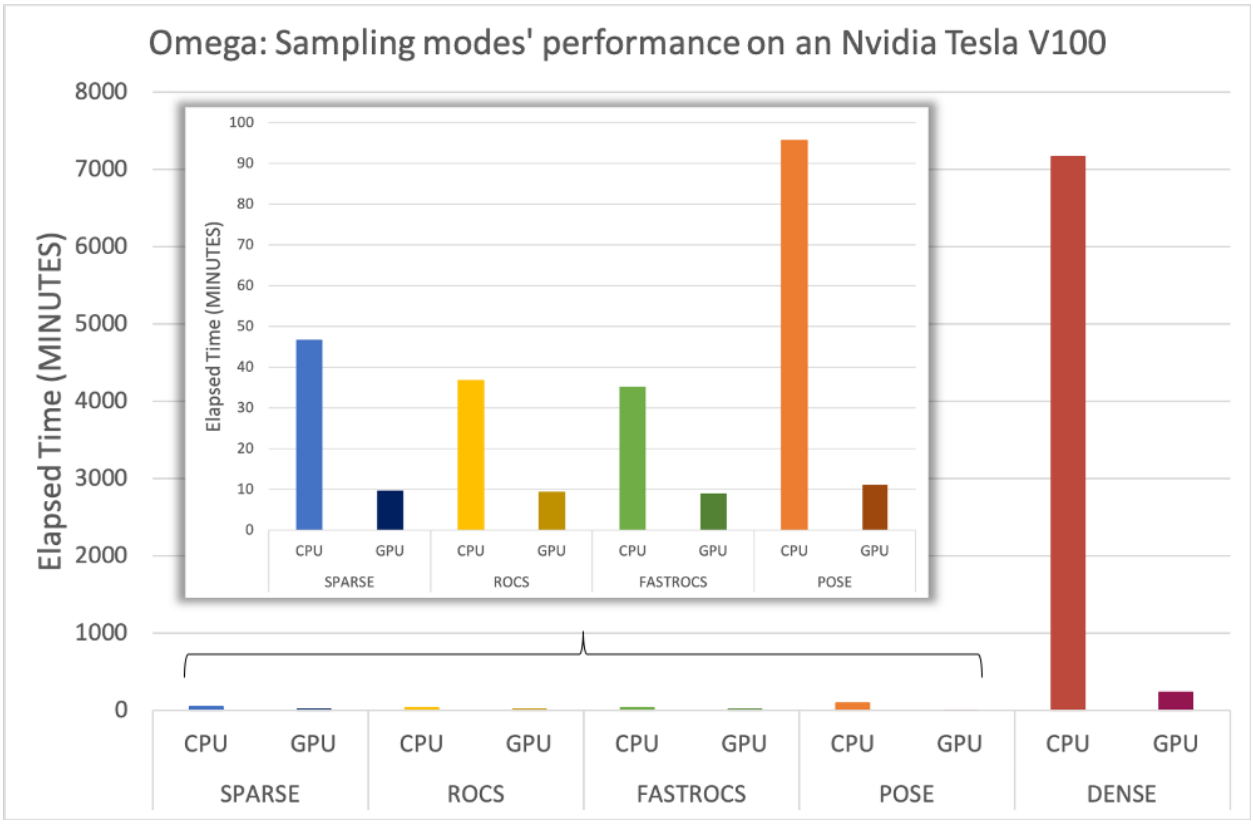


Fig. 10: **Elapsed time comparison between Omega and GPU-Omega on an AWS P3 instance when sampling a subset of the GSK-TCAMS dataset with sparse, rocs, fastrocs, pose, and dense modes.**

FastROCS TK: Color optimization on GPU

Overlay optimization by color, in addition to shape, has been added to **FastROCS**. The color adds chemistry in the superposition and similarity analysis process, and thus facilitates identifying compounds that are similar both in shape and chemistry. **FastROCS** with color optimization was carried out on K80, M60, and V100 instances on AWS GPU instances and compared against those without color optimization:

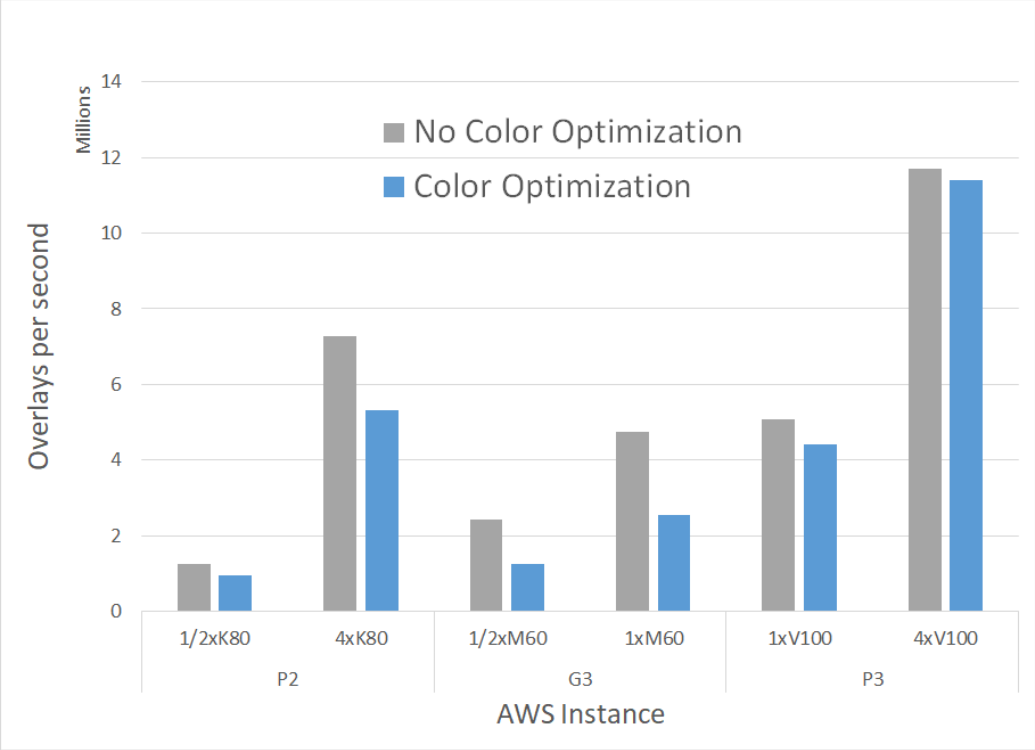


Fig. 11: **Performance comparison for FastROCS with and without color optimization in the 2020.0 release, using OpenEye's standard benchmark dataset of 14M conformers on AWS.**

Previous versions of **FastROCS** optimized over the shape score and followed that with a single point calculation for color. This can result in overlooking promising molecules. An example of the effect of optimizing over color and shape overlap rather than just shape overlap is shown below for XIAP BIR3 [Ndubaku-2009]:

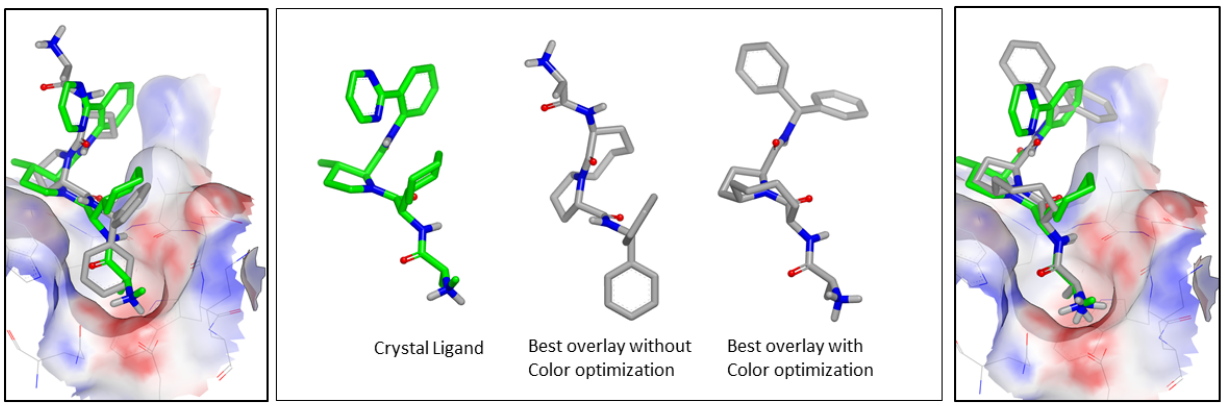


Fig. 12: **Example of the improvement using color optimization in FastROCS**

On the far left is the XIAP BIR3 binding site with its crystallographic ligand in green. In the center is the crystallographic ligand, as well two copies of a known active ligand from the DUDE dataset [Mysinger-2012], with the best overlay from **FastROCS** without and with color optimization, respectively. Without color optimization, the ligand is not predicted to have high shape or chemical similarity. When color optimization is used, a high degree of chemical

and volumetric overlap is predicted, as would be expected given that this compound is a known active. On the far right is the overlay in the binding site of XIAP BIR3, occupying a similar pose to the crystallographic ligand.

Spruce TK: Protein Loop Modeling

The 2020.0 release adds protein loop modeling to **Spruce TK**. This feature is available with the lower level APIs `OEBuildSingleLoop` and `OEBuildLoops`. It is also incorporated as an option in the high level structure preparation API `OEMakeDesignUnit`.

The approach relies on a template database built from structures in the public RCSB Protein Data Bank. The database, which has been built for modeling loops that are 4-20 residues long by default, is available for download in a platform-independent format. An application in the **SPRUCE** bundle allows appending additional structures to the existing database. For example, proprietary in-house structures can be added, or the database can be completely rebuilt from scratch with different options.

This approach has been validated on a public dataset from Rossi et al. [Rossi-2007]. The results shown below were generated by building the excised loops back into the dataset, where the specific structures themselves were not part of the template database, and used the same success criterion as in the paper (<2.5A backbone RMSD). The results are generally good and the median RMSD is 0.6-0.7A (blue dots), irrespective of the length of the modeled loop. The unsuccessful cases (gray bar) have a large effect on the average RMSD (green dots), as loops are chosen that deviate significantly from the expected. Only in a single case of the ~200 loops modeled was a result not produced (red bar) that survived the search and filtering steps.

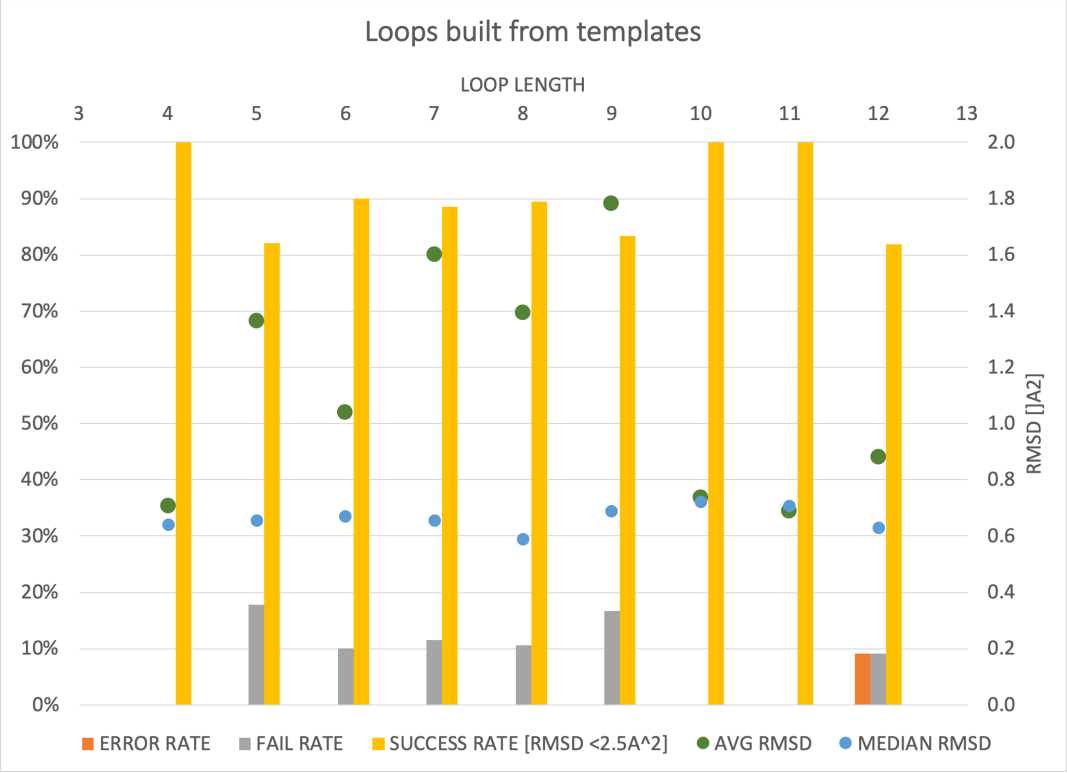


Fig. 13: **Performance assessment building loops ranging from 4-12 residues in length. Errors are when no loop was produced, failures when the backbone RMSD is larger than 2.5A from the original structure.**

For the full list of new preliminary APIs, see Spruce TK 1.1.0 Release Notes.

OEChemTK: Data record preliminary API

A preliminary API for **OERecord** and its associated classes has been added to **OEChem TK**. **OERecord** is a data container for storing and transmitting strongly typed data and its associated metadata. A pure Python implementation of **OERecord** has been in use by **ORION** and **Floe**; the new C++ implementation improves performance and memory management, and makes data records available to **OpenEye** toolkit code. The API is only available in C++ and Python at this time.

8.8.4 General Notices

- Support for macOS Catalina 10.15 has been added. MacOS 10.12 is no longer supported.
- RHEL8 is supported across all toolkits except for the C++ CUDA-enabled toolkits and features. **FastROCS TK**, **GPU-Omega TK**, and **GraphSim TK's** CUDA fast fingerprints are not supported.
- C++ CUDA-enabled toolkits and features are not available on Ubuntu18. **FastROCS TK**, **GPU-Omega TK**, and **GraphSim TK's** CUDA fast fingerprints are not supported.
- C++ toolkits are now available for GCC 8.x.
- Support for Python 3.8 has been added for all supported platforms. Python 3.6 is no longer supported for Windows toolkits.
- Java toolkits are now available for macOS.
- Visual Studio 2019 toolkits are now available for C++ and C#.
- This is the last release to support Ubuntu16.04. Support for Ubuntu20.04 will be added in the next release.
- Memory management in the Java toolkits no longer relies on the JVM to clean up objects. The new memory management significantly improves the Java toolkits performance, however requires users to manage memory. See Best Practices for Java for current best practices.

8.9 OEToolkits 2019.Oct

8.9.1 Spruce TK: A new toolkit for biomolecular structure preparation

The 2019.Oct release introduces **Spruce TK**. **Spruce TK** takes experimental biomolecular structure data in either PDB or mmCIF formats, and prepares the structures for use with downstream modeling applications, such as docking with **FRED** or **HYBRID**, pose prediction with **POSIT**, identifying favorable and unfavorable waters with **SZMAP** or free energy calculations using molecular dynamics simulations.

With **Spruce TK**, users can easily prepare proteins using either C++ or Python APIs (see `OEMakeDesignUnits`), where the output of the file is the recently developed OEDesignUnit class and is serialized to a new OEDU file format. The prepared system is componentized into molecular categories including *protein*, *ligand*, *nucleic*, *solvent*, and many other molecular types common to biomolecular experiments. By default, hydrogens are added to systems using a tautomer search for any ligand, cofactor, or other nonstandard residue molecules. In addition, the system is split into distinct biological units, and all final design unit structures are superposed into the common frame of reference of a parent structure. For more details, see Spruce theory.

Spruce TK has been tested across a broad range of industry-relevant targets. It has been used in virtual screening and molecular dynamics simulations in **Orion**, as well as to prepare structures for loading into the MacroMolecular Data Service (MMDS).

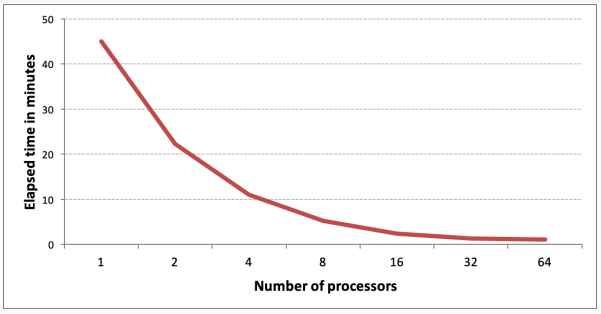
8.9.2 OEChem TK: Fast Substructure Search

Substructure search is a fundamental task in cheminformatics. **OEChem TK** now provides improved substructure search capability, allowing users to search tens of millions of molecules in seconds.

OEChem TK uses the screen-based approach to accelerate the search process. Screens are bit vectors encoding global and local features of molecules. They are prebuilt and can be used to rapidly eliminate structures that could not be matched to a query molecule. Only molecules that pass the screen are subjected to the computationally more expensive atom-by-atom matching for validation. **OEChem TK** provides three types of built-in screens based on the source of the query molecule: `MDL`, `SMARTS`, and `Molecule`. All screens are rigorously tested using diverse sets of queries and multiple molecule databases to ensure that they do not eliminate true positive matches.

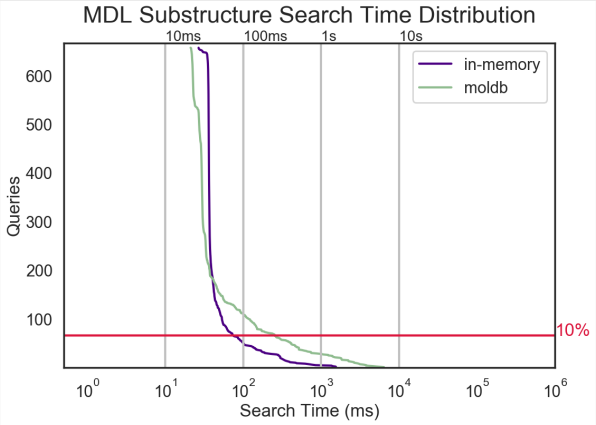
The new multi-threaded substructure search can be performed in two modes: `in-memory` and `molecule-database`. The `in-memory` mode provides the fastest way to search a dataset, but it is memory-intensive as it holds both screens and molecules in memory. The `molecule-database` mode keeps only the screens in memory; only molecules with unmatching screens have to be loaded into the memory. This method can be slower but uses significantly less memory, allowing users to search larger datasets.

screen generation performance



- *generating 1M MDL 896-bit long screens*
- *benchmarked on r5.16xlarge AWS instance*
- *multi-threaded screen generation*

substructure search performance



- Enamine REAL Diverse 24.1M
- in-house diverse set of 650 MDL queries
- *benchmarked on r5.16xlarge AWS instance*
- *multi-threaded search using 64CPU*

See the Molecule Searching chapter of **OEChem TK** for examples using the new OESubSearchDatabase class. For the full list of new preliminary APIs, see OEChem TK 2.3.0.

8.9.3 OEFF TK: SMIRNOFF

The 2019.Oct release incorporates the SMIRNOFF small molecule force field from the Open Force Field Initiative into **OEFF TK**. The force field can handle almost all pharmaceutically relevant chemical space.

The SMIRNOFF force field differs from traditional force fields principally in the data representation of the chemistry. It uses extended SMARTS strings for parameters instead of atom types, greatly reducing the number of interdependent parameters. It currently adopts the functional form of the AMBER force field, but the Open Force Field Initiative is actively examining improvements to this. **OEFF TK** intends to include the most up-to-date version of the force field in its own releases whenever possible.

8.9.4 General Notices

- This is the last release to support macOS 10.12. Full support for macOS 10.15 will be added in the next release.
- This is the last release to support RHEL6. Full support for RHEL8 will be added in the next release.
- The 2019.Oct release skips Java support for macOS. For Java support on RHEL6, please contact support@eyesopen.com.
- A new search engine that provides better hits and shows results in a much cleaner format has been added to the documents.
- The 2019.Oct release now uses SWIG 3.0.12 for generating C#, Java, and Python wrappers. C# and Python use SWIG's built-in support for wrapping C++ std::vector and string types.

8.10 OEToolkits 2019.Apr

8.10.1 FastROCS TK: Color calculation on GPU

Changes to the **FastROCS** algorithm, including porting the static color calculation to GPU, have resulted in a more than 2x performance improvement of **FastROCS** on some GPU systems. Benchmarking was carried out on AWS GPU instances and compared to the 2018.Oct version of **FastROCS TK**:

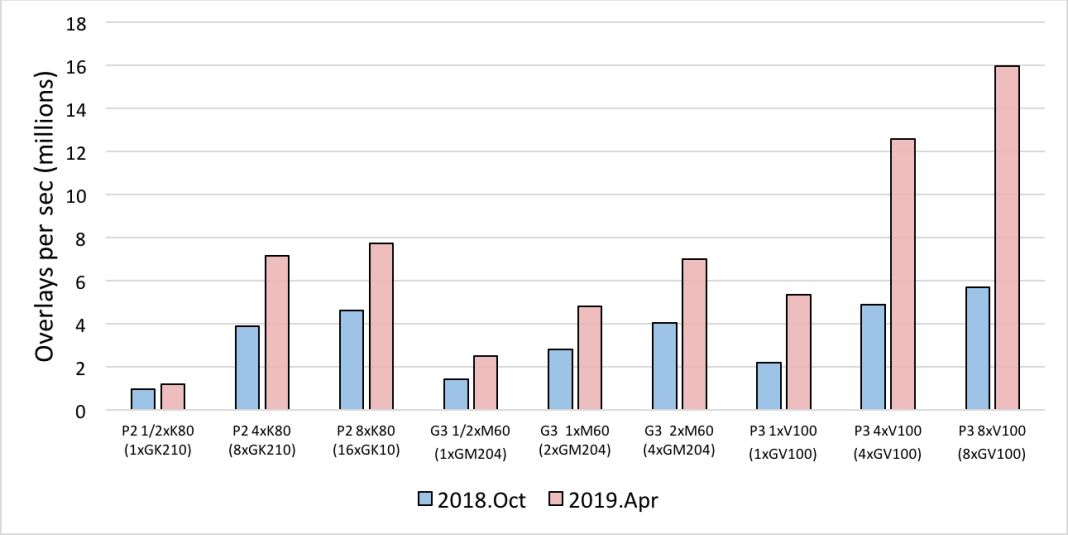


Fig. 14: **Performance comparison between the 2018.Oct release and the 2019.Apr release using OpenEye's standard benchmark dataset of 14M conformers on AWS.**

For some systems, the standard benchmark dataset of 14M conformers is not large enough to fully saturate the GPUs. To demonstrate the potential performance of larger databases, a 2 billion conformer subset of Enamine was searched on suitable AWS instances. FastROCS searched the 2 billion conformer database on an AWS P3 instance with 8x Nvidia Tesla V100 GPUs at a rate of over 44 million overlays per second:

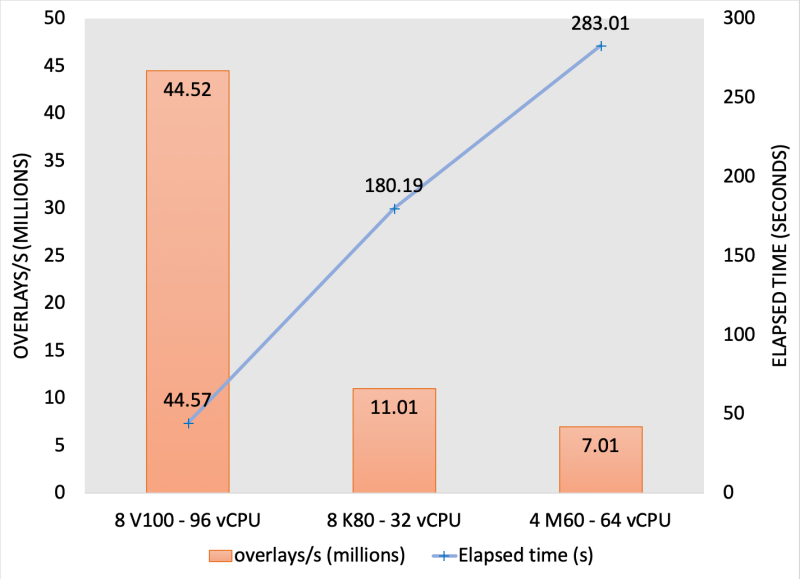


Fig. 15: **Performance of a 2 billion conformer subset of Enamine on suitable AWS instances.**

8.10.2 OEChem TK: OEZ compressed file format

A new compressed file format with the extension `oez` has been added to **OEChem TK**. This format uses the Zstandard algorithm to efficiently compress individual multi-conformer molecules, making it well-suited to work with OEMolDatabase for fast read-only random access to molecules. See more details in the Compressed Input and Output section of the **OEChem TK documentation**.

8.10.3 OEGrapheme TK: Ramachandran plot

Following our tradition of providing easily comprehensible visualization of complex data, **OEGrapheme TK** can now depict Ramachandran plots ([Ramachandran-1963]). These plots provide an easy way to examine the distribution of backbone dihedral angles in a particular protein. For instance, structure quality can be assessed by identifying residues whose backbone ψ are outside energetically accessible regions. **Grapheme TK** uses the data and classifications available in **OEBio TK**, which were extracted from the open-source Computational Crystallography Toolbox [Grosse-Kunstleve-2002].

8.10.4 General Notices

- **OpenEye** currently uses OpenJDK8 to build its Java packages. We are currently exploring support for OpenJDK 11.
- The 2019.Apr release skips Java support for macOS. For Java support on RHEL6, please contact support@eyesopen.com.
- This release adds support for macOS 10.14. OS X 10.11 is no longer supported.
- This release adds support for Ubuntu 18. Ubuntu 14 is no longer supported.
- Visual Studio 2017 support has been added for C++ and C#. This is the last release to support Visual Studio 2013, as well as 32-bit Windows for C++ and C#.
- Full support has been added for Python 3.7. Python 3.5 for Windows is no longer supported.
- The following prerequisites have been introduced so that **OpenEye** can implement algorithms in a more modern and efficient C++11 style:

- GCC 4.8 is now the oldest targeted compiler for toolkits. RHEL6 with the default GCC 4.4 compiler is no longer supported.
- The minimum recommended GNU binutils version is 2.26 for the C++ single-build Linux packages.
- The minimum CMake version required to build the C++ examples is now 3.1.3 on all platforms.

8.11 OEToolkits 2018.Oct

8.11.1 Omega TK: Macrocycle

A new distance geometry-based, conformational space-sampling method has been added to **Omega TK** as a preliminary API. This approach opens up a new avenue for exploring conformer ensembles in **Omega TK**. Although the distance geometry-based sampling is slower than the traditional torsion driving-based approach, it works substantially better for ensemble generation of macrocyclic molecules. This sampling method can also be applied to linear and small ring-containing molecules.

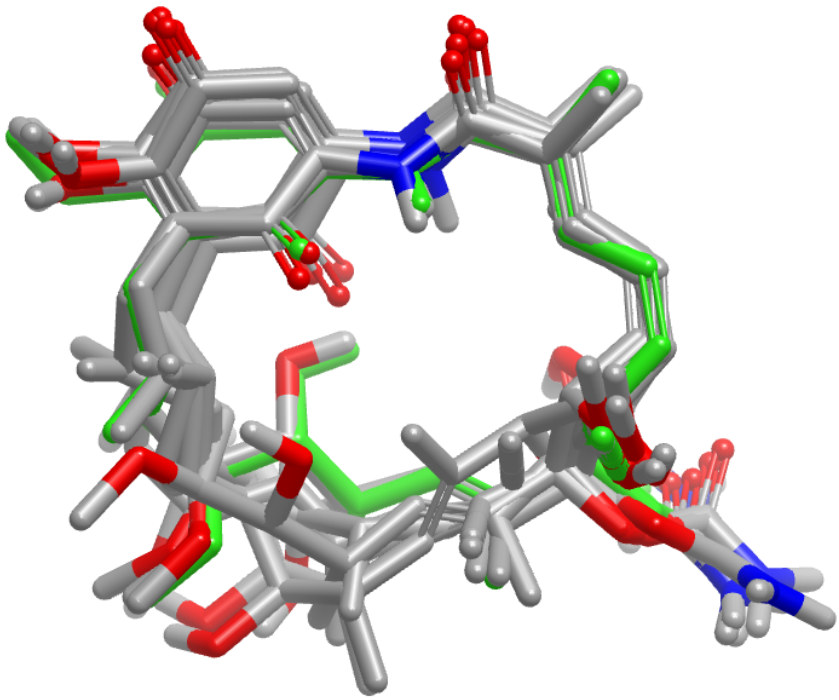


Fig. 16: **Conformer ensemble subset for geldanamycin (PDB: 1YET) generated using the macrocycle method**

8.11.2 FastROCS TK: C++ and Java Support

Previously, **FastROCS TK** was available only in Python. This release extends support to C++ and Java. All classes, functions, and constants are wrapped and available for use. Several example scripts now have C++ and Java versions to showcase **FastROCS TK** functionalities. Additionally, users running **FastROCS TK** on Tesla GPUs will see a 10-30% speed-up over the previous version, depending on the system.

8.11.3 Quacpac TK: Tautomers

This release adds several improvements to the tautomer functionality in **Quacpac TK**. Tautomer handling in molecular modeling and cheminformatics usually includes three important functions:

- Generating a canonical protomer representation for duplicate-free storage
- Generating a single tautomer suitable for visualization by chemists and modelers
- Generating an ensemble of aqueous-phase tautomers suitable for molecular modeling

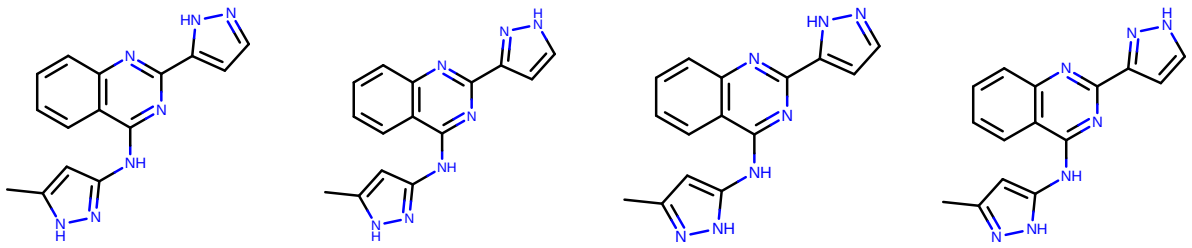


Fig. 17: **Example of generating an ensemble of biologically relevant tautomers**

This release provides new API to cover each of these use-cases. The algorithm has also been improved to handle memory usage when a large number of possible tautomeric forms are present.

8.11.4 General Notices

- The release cycle of **OpenEye Toolkits** has been changed. See the Release Cycle section for more information.
- This is the last release to support GCC 4.4 on RHEL6. For future toolkit releases, GCC 4.8 will be the oldest targeted compiler.
- This is the last release to support OS X 10.11. Full support for macOS 10.14 will be added in the next release.
- This release adds support for Ubuntu 18 and will be the last release to support Ubuntu 14.
- This release adds support for Python 3.7 on macOS and Linux. On Windows, Python 3.7 will be supported in the next release that will also drop support for Python 3.5.
- **FastROCS TK** no longer supports Nvidia Tesla, GeForce, and Quadro cards with a compute capability of less than 3.0. This includes all GPUs with Fermi architecture (e.g., Tesla C2050 and GeForce GTX 430). Users running one of these older GPUs who cannot upgrade their hardware should not update to this version of **FastROCS TK**.
- **Spruce TK** now requires a specific license. For information on **Spruce TK** licensing, please visit https://www.eyesopen.com/contact.
- Improvements have been made to the Python 3 toolkits with regard to wrapping optimization. This results in Python 3 performance being comparable to Python 2.

8.12 OEToolkits 2018.Feb

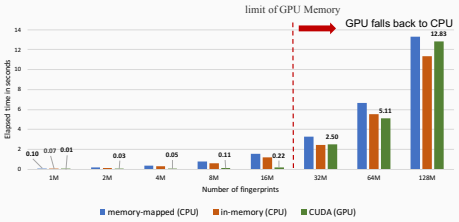
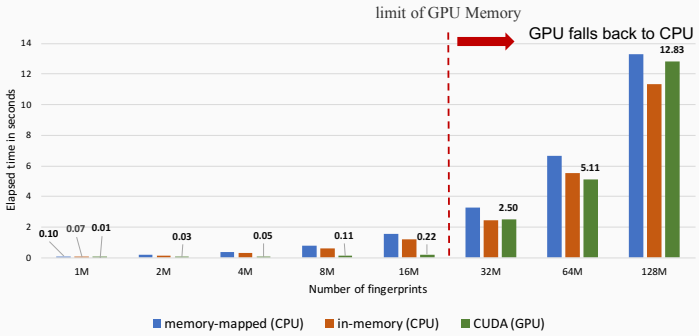
8.12.1 GraphSim TK: GPU-Accelerated Fast Fingerprint Similarity Searches

GraphSim TK now supports CUDA for even faster fingerprint searching, allowing users to perform molecular similarity calculations up to *200x faster* than the in-memory fast fingerprint mode (see the NxN similarity score calculations graph below).

CUDA mode involves preloading all fingerprints into GPU memory prior to performing similarity calculations. Searches are limited by GPU memory availability and will fall back to the memory-mapped CPU mode if the entire set of fingerprints cannot be preloaded into the GPU memory. Despite this limitation, this represents the fastest way to perform similarity searches.

The most effective way to use CUDA fingerprint searches is to perform multiple searches with a pre-loaded database on a dedicated GPU machine. If CUDA mode is selected but a GPU is not present on the system, the search will simply fall back to memory-mapped mode.

The new `OEFastFPDatabase.GetHistogram` method performs an NxN search of a database and returns a histogram of scores. This is valuable information for understanding the diversity of a database without consuming excess disk space. Using `OEFastFPDatabase.GetHistogram` in CUDA mode will afford *2.5 billion* searches per second (benchmarked on an AWS P3 instance - NVIDIA Tesla V100) compared with *12.5 million* searches per second when using memory-mapped mode on the CPU.

Performance of an NxN similarity score calculation	Performance of a fingerprint search (load time not included)
	
• *16M 4096-bit long tree fingerprints = 8GB data* • *CPU benchmarked on m4.10xlarge AWS instance (single threaded)* • *GPU benchmarked on p3.8xlarge AWS instance* • *In-memory timings are extrapolated from shorter runs*	• *128M 4096-bit long tree fingerprints = 64GB data* • *retrieving the 10 most similar molecules* • *CPU benchmarked on m4.10xlarge AWS instance (single threaded)* • *GPU benchmarked on p3.8xlarge AWS instance*

See the GraphSim TK Examples chapter of the **OEGraphSim TK** documentation for more information and examples using the new OEFastFPDatabase class.

8.12.2 Grapheme TK : Unpaired Interaction Map

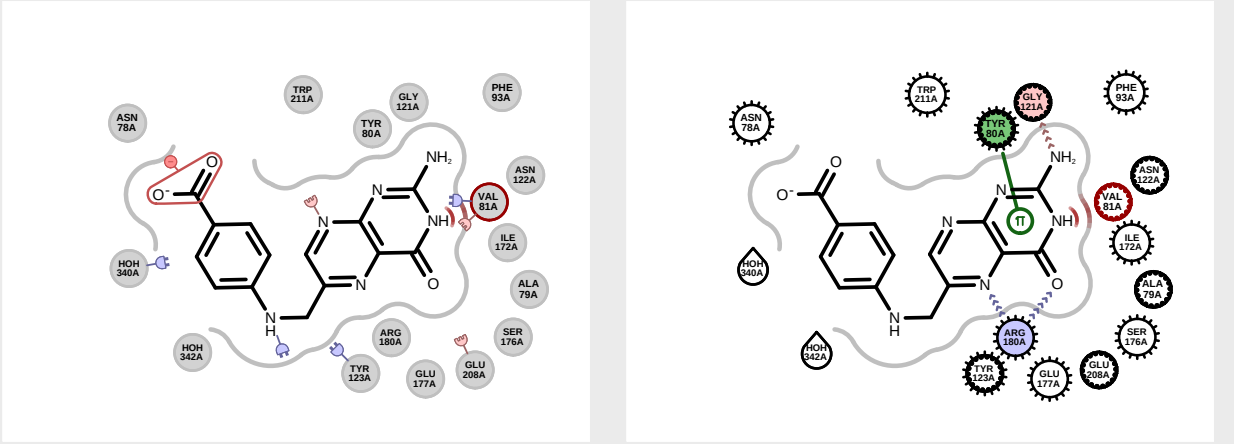
Grapheme TK now includes the ability to generate unpaired interaction maps (see the example below). An unpaired interaction map provides a complementary view to the more common active site interaction depiction. While the interaction map (on the right) depicts interactions between the ligand and protein, the unpaired map (on the left) illustrates interactions that could contribute to binding but are not formed in the complex. Together, these two maps of the protein-

ligand binding site provide insights into protein-ligand interactions and communicate complex 3D structural results in a directly actionable way.

In the unpaired interaction map, different linker types are used to mark unpaired acceptor and donor hydrogen bond interactions (see the figure on the left). Note that since these interactions are unpaired, the direction of the linkers has no real spatial meaning. Ligand linkers are directed away from the ligand, while protein linkers are directed towards the ligand.

The geometric parameters used to perceive unpaired interactions are customizable to fit user needs.

Table 1: **Unpaired interaction map (left) and active site depiction (right) generated after hydrogen placement (PDB: 1BR6)**



8.12.3 Conda Packages for Python

The 2018.Feb release is the first release to fully support Conda packages for Python toolkits on all supported platforms. The toolkits are available as a package named `openeye-toolkits` in the `openeye` Conda channel on the default `conda.anaconda.org` server. Versions are available for Python 3.5 and higher on Windows and Python 3.4 and higher on Linux and macOS. On Linux and macOS, users can install the toolkits into a Conda Python environment with `conda install -c openeye openeye-toolkits`. On Windows, Anaconda Navigator is a convenient way to manage Conda packages.

See Python Installation for more information.

8.12.4 General Notices

- All **OpenEye Toolkit** Python code examples now follow the PEP8 standard.
- The 2018.Feb release fully supports macOS High Sierra 10.13.
- mac OS 10.10 is no longer supported.

8.13 OEToolkits 2017.Oct

8.13.1 Shape TK: Version 2.0.0

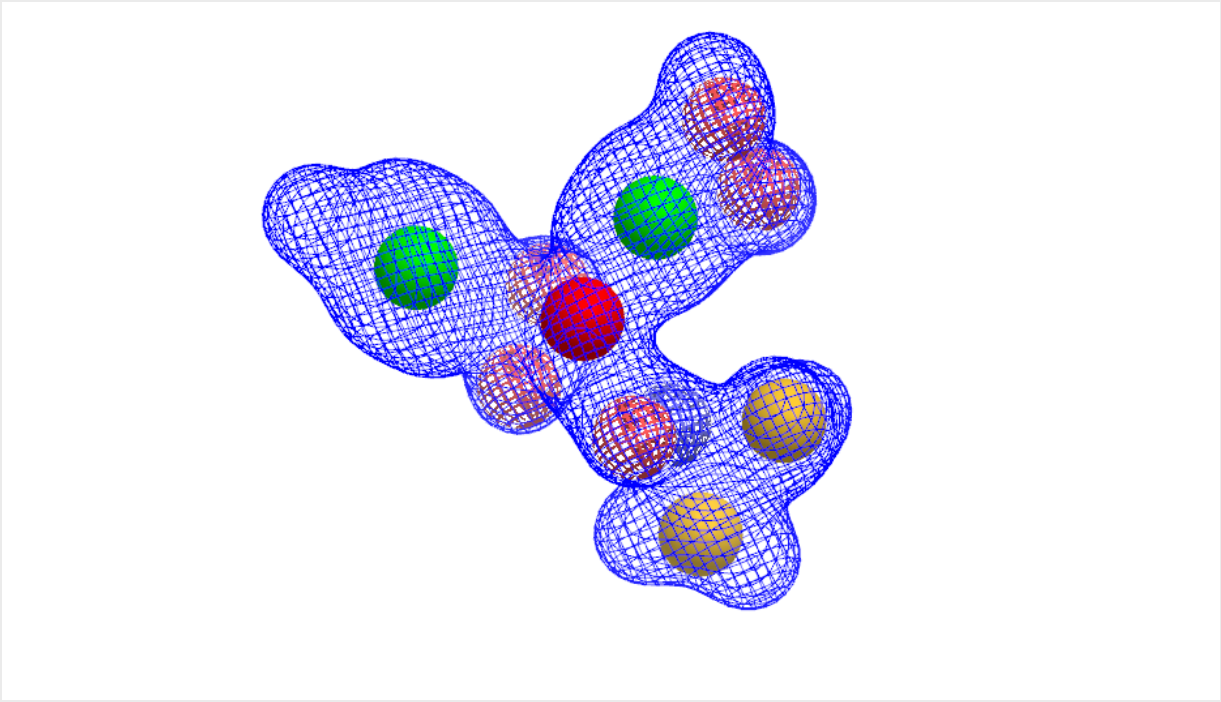
The 2017.Oct release introduces **Shape TK** version 2.0.0, which is completely redesigned with a new API that is more flexible and easily extended with new functionality. **Shape TK** version 2.0.0 is now thread-safe and is able to:

- calculate color using grid- or proxy grid-based analytic methods

- calculate shape overlap and color score between two fixed-in-space objects with a single calculation using the OEOverlapFunc class
- combine force fields from the **OEFF TK** with shape and/or color for overlay optimization
- obtain top hits from a database against a given query using the OEROCS class
- create a query combining any combination of a molecule, a grid, and multiple shape and/or color Gaussians using the OEShapeQuery class (see example below)

Preliminary testing of functionality from **Shape TK** version 2.0.0 indicates that overlay optimization is up to 10% faster than the *2017.Jun* version.

Table 2: **Shape query with shape grid and color Gaussians**



8.13.2 Omega TK: New Torsion Library

A new torsion library has been added to **Omega TK** based on a probabilistic analysis of torsion angles in the Cambridge Structural Database (CSD) by Guba et al. [Guba-2016] This torsion library, which contains more than 500 entries, complements the existing torsion library, which is based mainly on analysis of structures found in the Protein Data Bank (PDB). Replacement of the existing library with the new library has four important effects on OMEGA's performance:

- nearly complete elimination of torsion values not well represented in the CSD
- significantly improved run time
- substantial reduction in the size of the conformer ensemble
- small trade-off increase in the median RMSD for solid state structures (both from the PDB and the CSD)

8.13.3 OEDepict TK and Grapheme TK: Interactive SVG Images

The 2017.Oct release expands these toolkits' ability to assemble sets of graphical objects in SVG images.

Images can present great deal of information very quickly, but those that try to convey too much information at once can look convoluted. The solution lies in generating interactive images that on the surface are simple but that can reveal extra information on request. This release allows users to mouse over simple images to view additional embedded information.

In the example below, the sequence-level depiction of a peptide is easily comprehensible due to its simplicity (see image below). The more detailed atomic-level representation of the amino acid components of the peptide can be revealed on mouse-over without cluttering the simplified view.

For more information, see the Generating Interactive SVG Images chapter of the **OEDepict TK** documentation.

8.13.4 General Notices

- On macOS Python, `single-build` distributions are now installed by default when using the meta-installer via `pip install openeye-toolkits`.
- The Python verification tests now use `pytest` rather than `nosetest` as the test discovery tool. `pytest` options can be passed to the `openeye.examples.openeye_tests` module to control the tests and level of debugging information provided.
- This is the last release to support Java 1.6.
- This is the last release to support Python 2.7.

8.14 OEToolkits 2017.Jun

8.14.1 Shape TK: Hermite

The 2017.Jun release introduces the OEHermite class, which allows a molecule to be expanded into a Hermite representation. An initial suggested area of application is the representation of protein shape. The advantages of representing proteins by Hermites include:

- smooth representations that capture any level of detail, from the atomistic (many coefficients, see the picture on the right) to the very coarse (few coefficients, as in the middle picture),
- the ability to store these representations in a compact manner,
- the ability to transform these representations easily, and
- very fast overlap calculations between proteins.

Below are examples of low- and high-resolution Hermite representation images of dihydrofolate reductase (DHFR):

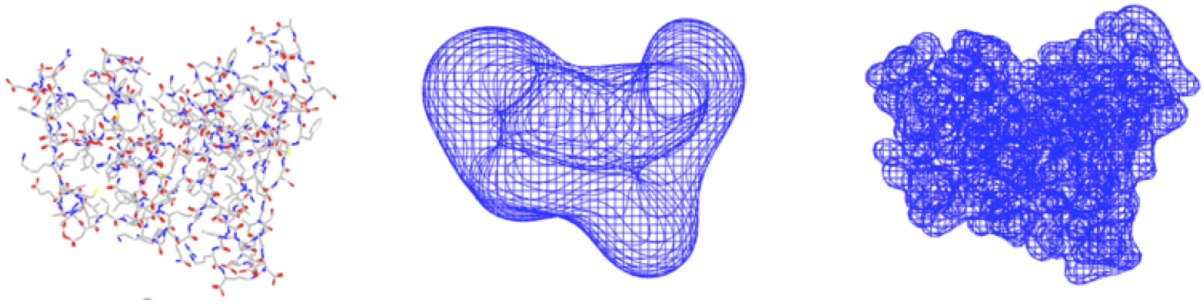


Fig. 18: **Exact shape versus Hermite Representation**

OpenEye will explore the potential of Hermite representations; we invite users to do likewise with this exciting new addition to the OpenEye toolkits. See the Shape from Hermite Representation chapter of the **Shape TK** documentation for more information.

8.14.2 OEDepict TK and Grapheme TK: Interactive SVG images

Scalable Vector Graphics (SVG) image format is dominant in web applications due to its wide browser support and scalability. An SVG image contains a list of graphical objects that can be grouped, styled, and transformed together. This enables the generation of interactive images defined in the SVG XML elements or via scripting that accesses the SVG Document Object Model (DOM).

The new Preliminary API available in **OEDepict TK** provides low-level functionality that enables grouping of graphical elements in SVG images that can be manipulated externally by web developers. High-level API entrypoints are also available to generate self-contained SVG images that show or hide different elements of the image based on mouse events.

For more information see the Generating Interactive SVG Images chapter of the **OEDepict TK** documentation.

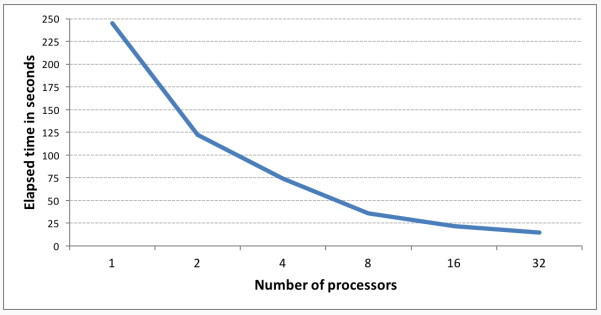
8.14.3 GraphSim TK: Fast Fingerprint Similarity Search

OEGraphSim TK now provides improved fingerprint search capability, allowing users to perform molecule similarity searches on large databases up to 10x faster than previous versions.

Fingerprint methods are used for the computation of similarity scores. When searching large virtual databases, performance can be problematic. To address this issue, **OEGraphSim TK** now uses popcount, a hardware instruction available on modern CPUs, to provide significant performance improvements for computing similarity.

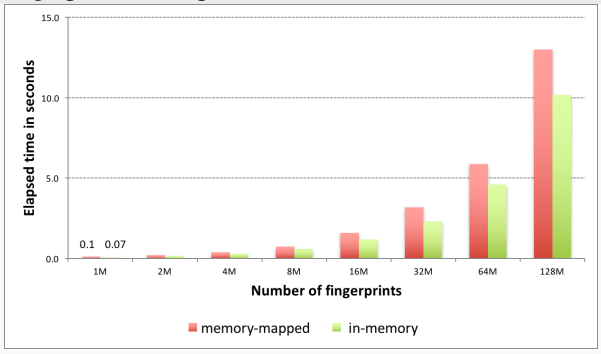
The new fingerprint search can be performed in two modes: *in-memory* and *memory-mapped*. The in-memory mode involves preloading all fingerprints into memory and performing the search. While this represents the fastest way to perform similarity searches once the fingerprints are loaded, searches are limited by memory availability. The memory-mapped mode has no load time penalty or memory limitation but the search itself takes more time.

fingerprint generation performance



- *generating 1M 4096-bit long tree fingerprints*
- *benchmarked on m4.10xlarge AWS instance*
- *multi-threaded fingerprint generation*

fingerprint search performance



- *128M 4096-bit long tree fingerprints = 64GB data*
- *benchmarked on m4.10xlarge AWS instance*
- *single-threaded search on CPU*

See the GraphSim TK Examples chapter of the **OEGraphSim TK** documentation for more information and examples using the new OEFastFPDatabase class.

8.14.4 General Notices

- On Linux, Python `single-build` distributions are now installed by default when using the meta-installer via `pip install openeye-toolkits`. If a problem is encountered with the `single-build` packages, users can set the environment variable `OE_PIP_ARCH=old` before the `pip install` step.
- Python 3.6 is supported in the 2017.Jun release.
- This is the last release to support 32-bit Java.
- This is the last release to support Red Hat Enterprise Linux version 5. As of March 31, 2017, RHEL5 has reached end-of-production support.

Note

OpenEye continues to monitor and collect feedback from our customers on the need to support Python 2 wrappers. Our current plan is to phase out support for Python 2 on all platforms by the 2017.Oct release. As this is a substantial change for our customers, we are willing to help with migration. Please contact support@eyesopen.com for more details.

8.15 OEToolkits 2017.Feb

8.15.1 Alternative Starting Points for Shape Overlap Optimization in FastROCS TK

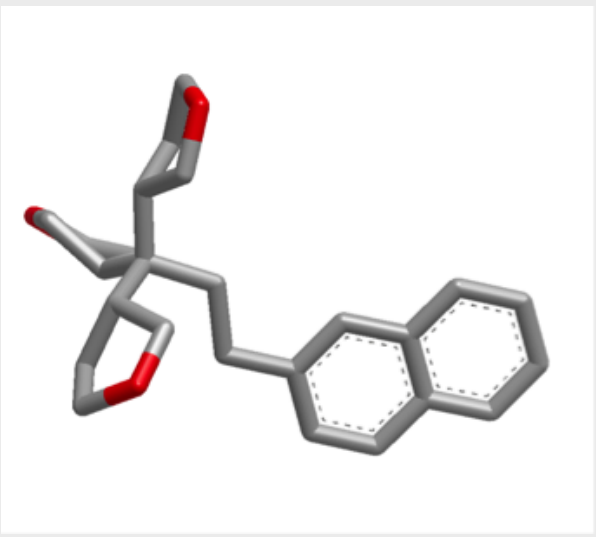
This release of **FastROCS TK** extends its capability to find the best possible shape match between two conformers by introducing the ability to use custom starting points for the shape search. For some systems, using these alternative starting points can lead to more accurate shape scoring as it enables broader sampling of the molecular volume overlap.

By default, the shape search in **FastROCS TK** starts from 4 poses of the database molecule aligned around the inertial axes. With the `UserInertialStarts` option, **FastROCS TK** translates the database molecule to user-defined starting coordinates. With the `InertialAtHeavyAtoms` option, the database molecule is translated to each heavy atom of the query molecule. In both cases, the shape overlap is optimized for 4 unique inertial poses at each location.

The images below illustrate a shape search between caffeine query and a naphthalene derivative.

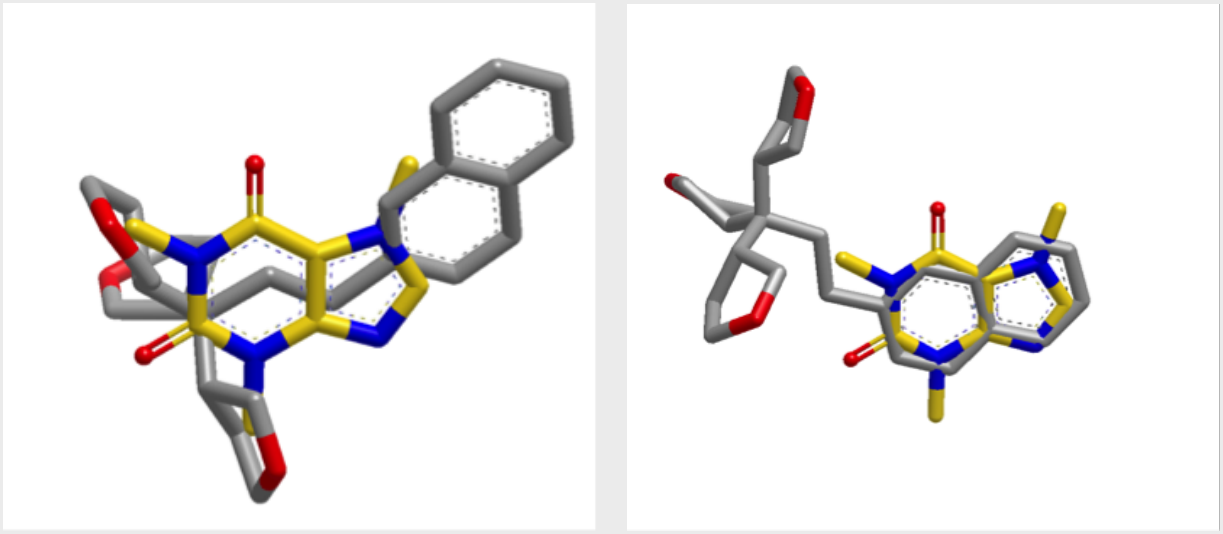
Table 3: **Caffeine query and naphthalene derivative**





The default shape search results in a poor overlap between the two molecules (left). When the new `InertialAtHeavyAtoms` option is used, the resulting overlap is much more favorable (right).

Table 4: **Best overlap: default vs. ``InertialAtHeavyAtoms`` starting points**



In general, these options provide the ability to fine-tune **FastROCS TK** searching based on specific criteria.

8.15.2 Charging Engine

A new charging function, `OEAssignCharges`, has been added. This function introduces the concept of a `charge engine`, a flexible, open-ended options class that defines a charging method. Charge engines have been written for every currently supported charging method as well as the new method `OEAM1BCCELF10Charges`.

8.15.3 New Maximum Common Substructure (MCS) Similarity Capability

A new class, OEMCSFragDatabase, has been added that allows pairwise MCS similarity scores to be computed for a query molecule against a set of indexed target structures. The indexing algorithm uses a fragmentation approach to reduce the computational requirements of traditional `NxN` structure comparisons for MCS scoring. Tanimoto and Tversky similarity scoring functions are provided, as well as the ability to implement custom scoring functions.

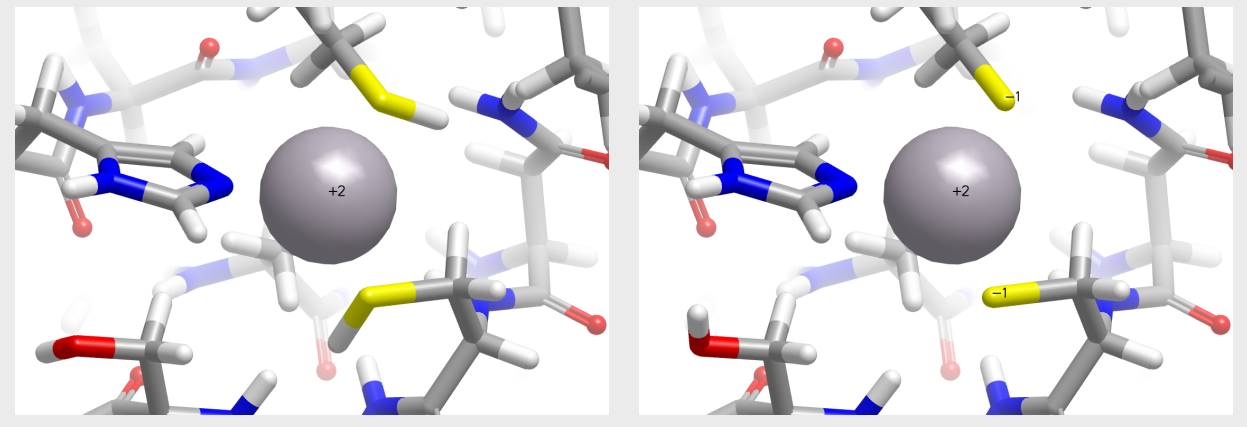
8.15.4 Protein Preparation Update

In the 2016.Oct release, new functionality for hydrogen placement was added as part of a protein preparation workflow. This new functionality has been significantly improved.

A new scoring system, `MMFF-NIE` ("MMFF Neighbor Interaction Energies"), has been added to `OEPlaceHydrogens` that significantly improves optimization of hydrogen bonding networks and clash avoidance.

In addition, significant improvements have been made in the placement of hydrogens around metal complexes. In the following example, cysteine sulfurs are now properly perceived as deprotonated in the zinc finger domain of the nucleocapsid protein NCp8 (PDB: 2A51).

Table 5: **2016.Oct vs 2017.Feb cysteines in zinc finger domain**



8.15.5 General Notices

- For Python, the new environment variable `OE_PIP_ARCH` can be set to override the Python package download. `OE_ARCH` now only applies to application installation. This should mitigate conflicts between Python and application package installation.
- On Linux, Python `single-build` distributions are now installed by default when using the meta-installer via `pip install openeye-toolkits`. If a problem is encountered with the `single-build` packages, users can set the environment variable `OE_PIP_ARCH=old` before the `pip install` step.
- The Java wrappers are now built using Java 1.8 in 1.6 compatibility mode. Users should see no impact.
- The 2017.Feb release fully supports macOS Sierra 10.12.
- Python 3.4 on Windows is no longer supported.
- Python 3.6 will be supported in the 2017.Jun release.

Note

The default Python version installed under Anaconda/Miniconda is Python 3.6. Until the 2017.Jun release is available, users should override the default Python version with `conda create python=3.5 ...` during the conda environment creation step.

Note

OpenEye continues to monitor and collect feedback from our customers on the need to support Python 2 wrappers. Our current plan is to phase out support for Python 2 on Windows by the 2017.Oct release and phase out support on other platforms at a later date. As this is a substantial change for our customers, we are willing to help with migration. Please contact support@eyesopen.com for more details.

8.16 OEToolkits 2016.Oct

With this release, **OpenEye** continues to focus on protein handling and visualization. In addition, numerous improvements to the toolkits have been made.

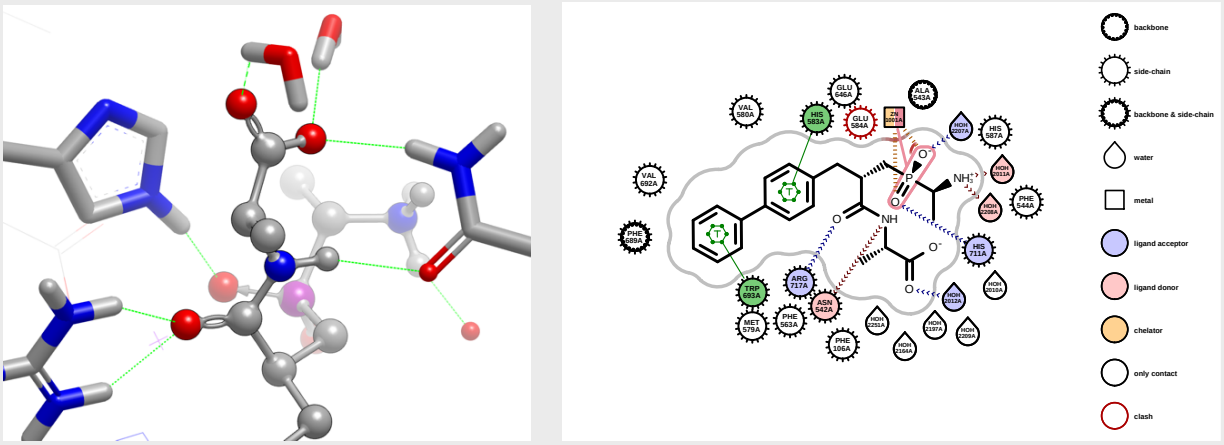
8.16.1 Protein Preparation

A new function, `OEPlaceHydrogens`, has been introduced to the **OpenEye** protein preparation workflow in **OEBio TK**. This function positions hydrogens in a molecule to build hydrogen bonding networks and avoid clashes. It can also flip functional groups that are sometimes modeled incorrectly, such as histidine, asparagine, and glutamine sidechains. Previously, **OpenEye** recommended that users run the open source application Reduce to perform these functions. `OEPlaceHydrogens` is easier to use and has broader applicability because it uses SMARTS patterns rather than relying on naming conventions.

This release represents an initial implementation of this functionality; it is under continuing development. A new example program, Preparing a Protein, demonstrates this functionality.

For more information, see the Protein Preparation section.

Table 6: **Example of a hydrogen bonding network in 1R1H along with the corresponding protein-ligand visualization**



8.16.2 Protein-Ligand Interaction

This release of **OEBio TK** includes significant improvements to the perception of potential protein-ligand interactions, intramolecular interactions, and missing potential interactions.

A new type of interaction, `Halogen Bonds`, has been added to **OEBio TK**. It joins `Hydrogen Bonds`, `Salt Bridges`, `Chelation`, and `Pi- and T-aromatic Stacking` in the set of perceived interaction types. These interactions provide qualitative hints about biological interactions that may be important for binding, and can be helpful for visualizing interactions and clashes.

An interface for modifying the geometric definitions for each interaction has also been introduced. While the default bond geometries have been carefully constructed using literature data (see below), users can now customize interaction parameters to fit their needs.

1. Kumar, S. and Nussinov, R., Biophysical Journal, 83:1595-1612, 2002.
2. Cavallo, G., Metrangolo, P., Milani, R., Pilati, T., Priimagi, A., Resnati, G., Terraneo, G., Chem. Rev. 2478-2601, 2016.
3. Bissantz, C., Kuhn, B., Stahl, M., J. Med. Chem., 53:5061-5084, 2010.
4. Marcou, G., Rognan, D., J. Chem. Inf. Model., 47:195-207, 2007.

8.16.3 FastROCS TK CUDA Implementation

The internal GPU implementation of **FastROCS** has been ported from OpenCL to CUDA. Although this does not change the results returned by **FastROCS** and should not impact the deployment of **FastROCS TK** on customer machines, it does allow **OpenEye** to better support NVIDIA hardware and facilitates our ability to improve performance and add GPU functionality.

8.16.4 General Notices

- The 2016.Oct release adds support for Ubuntu16 for C++, Java, Python 2.7, and Python 3.x.
- The 2016.Oct release is the last release to support OSX 10.9. The 2017.Feb release will add full support for macOS Sierra 10.12.

> **Note**
>
> **OpenEye** continues to monitor and collect feedback from our customers on the need to support Python 2 wrappers. Our current plan is to phase out support for Python 2 on Windows by the 2017.Oct release and phase out support on other platforms at a later date. As this is a substantial change for our customers, we are willing to help with migration. Please contact support@eyesopen.com for more details.

8.17 OEToolkits 2016.Jun

8.17.1 Single Build Distributions

OpenEye is pleased to announce the release of a Linux distribution built for maximum compatibility across multiple Linux versions. This will simplify the deployment of toolkits and toolkit programs across heterogeneous Linux environments. A toolkit executable built using the `single-build` distribution will run unmodified on most modern Linux systems, eliminating the need to build executables for each Linux release or compiler version. Currently, this `single-build` release is available for C++ and Python in parallel with our normal version-specific releases. In the future, we plan to make this build strategy available for all **OpenEye**-supported platforms.

For more information, see the Single-Build Distributions (Linux) section for C++ and the Common Linux Single-Build Installation section for Python.

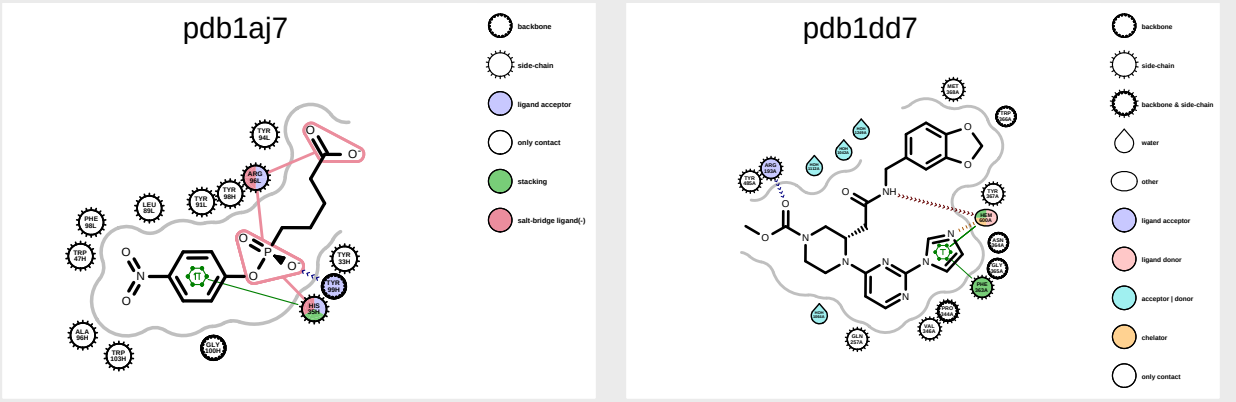
8.17.2 Protein-Ligand Visualization

Grapheme TK provides representation schemes that allow clear and coherent visualization of complex chemical information. Its most important function is to project the intricate interactions of a protein-ligand complex into a 2D diagram visualizing key interactions and properties that chemists can use to make fast and effective judgments.

The receptor-ligand visualization introduced in *2015.Feb* has been extended to visualize two new interaction types: salt-bridges and stacking interactions (Pi and T). Accordingly, the color scheme used to depict these interactions has been revamped.

Grapheme TK has also introduced the capability to select and highlight various parts of the protein-ligand 2D depiction.

Table 7: **Examples of new protein-ligand visualization in Grapheme TK**



Note

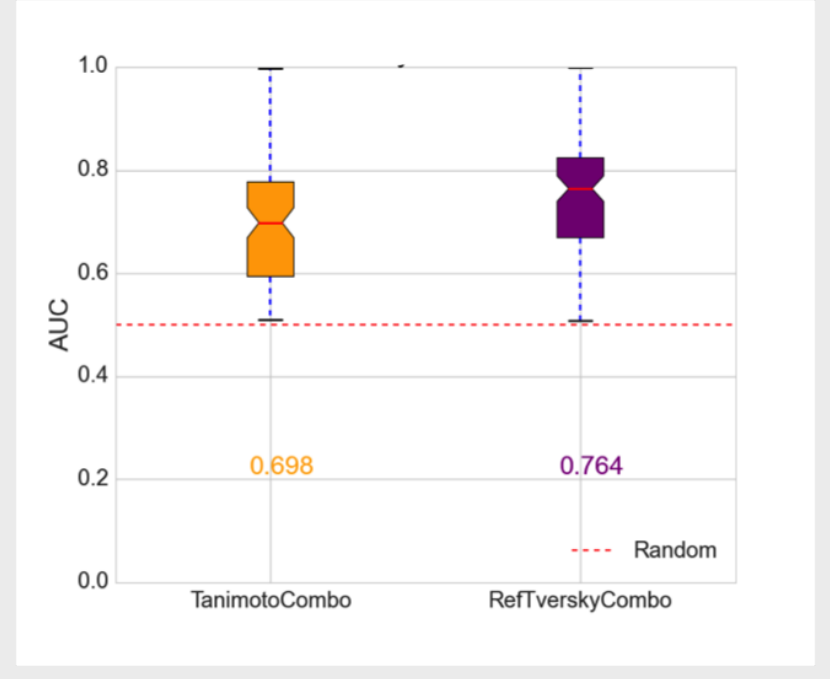
As per customers' request, the OEFragmentNetwork class and its related API have been renamed and deprecated. The words "fragment," "network," and "connection" appearing in the names of classes and functions designed to store receptor-ligand interactions has been found to be confusing. The primary class is now called OEInteractionHintContainer. For more details, see the Deprecated OEFragmentNetwork and related API section. The new API retains all previous functionalities and adds the capability to handle the new interaction types.

8.17.3 Tversky Similarity Scoring in FastROCS TK

FastROCS TK has supported only Tanimoto (symmetric) similarity scoring since its inception. This release adds Tversky similarity scoring to **FastROCS TK**. Using Tversky instead of Tanimoto scoring can lead to a higher retrieval of actives in searches, as indicated by our large-scale comparison experiment using the DUDE database (see the chart below).

Other significant improvements in **FastROCS TK** include the reduced memory consumption of OEShapeDatabase to support color scoring and support for the new `361.*` driver. This new driver has a noticeable performance improvement (approximately 10%), so upgrading is recommended.

Table 8: **Retrieval rates using Tanimoto and Tversky similarity measures**



8.17.4 General Notices

- This 2016.Jun release supports Python 3.5 on Windows with Visual Studio 2015. Python 3.3 on Windows is no longer supported.
- This 2016.Jun release supports Visual Studio 2015 for C++ and C# toolkits. Visual Studio 2010 and Visual Studio 2012 are no longer supported.
- This 2016.Jun release is the last to support Ubuntu 12 for all toolkits. The 2016.Oct release will add Ubuntu 16 support.
- This 2016.Jun release is the last to support 32-bit RedHat 5 for all toolkits.
- This 2016.Jun release is the last to support Python 2.6 on RedHat 6.

Note

OpenEye is planning to phase out Python 2 support by the October, 2017 release. As this is a substantial change for us and our customers, we are willing to help with code migration, either with advice or hands-on work with your code-base. Please contact support@eyesopen.com for more details.

8.18 OEToolkits 2016.Feb

8.18.1 OpenEye Toolkit Security

OpenEye toolkits are increasingly being used in web services that require protection from malicious users. The most obvious attack vector against the **OpenEye** toolkits is file format parsing since scientific file formats are complex and often underdefined.

OpenEye has closed a significant number of vulnerabilities related to the parsing of supported file formats: molecules, grids, surfaces, receptors, etc. In addition, state-of-the-art testing and detection techniques are now continuously run on the **OpenEye** code base to ensure a high level of security going forward.

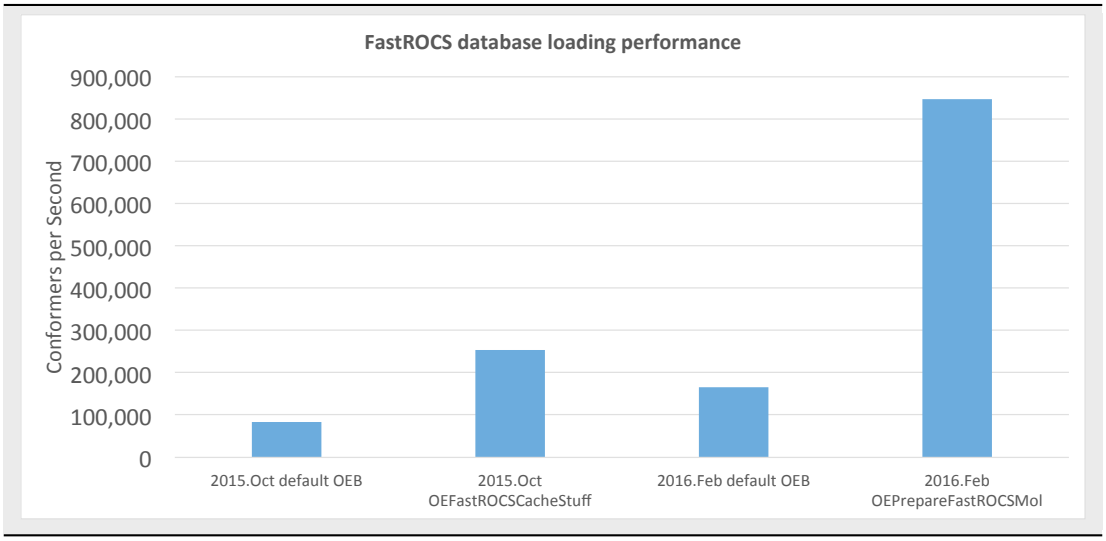
Security is a task requiring continuous vigilance and **OpenEye** will continue to make this effort a high priority.

8.18.2 FastROCS TK: Database Loading Performance Improvement

The utility of **FastROCS** has always been hampered by the amount of physical memory available in GPU machines. The computational chemistry community is continuously pushing the boundaries of **FastROCS** by searching larger and larger virtual libraries of compounds. In addition, **FastROCS** servers are becoming an increasingly important component of many infrastructures and must be able to be rapidly brought up and down on demand. Both these problems come down to the fact that loading a database into memory is often between 10 and 20 times slower than performing an actual **FastROCS** search on that database.

This problem has been addressed in this release by dramatically increasing the performance when loading a dataset into memory for eventual **FastROCS** processing. `OEPrepareFastROCSMol` generates additional information to be written to OEB that is used to rapidly read molecules into memory.

Using `OEPrepareFastROCSMol` on a molecule can result in load times that are nearly 10 times faster than previous uncached OEB file loads. Depending on the ratio of CPU and GPU compute power and conformers per molecule, this can mean that loading a dataset can now be just as fast as the **FastROCS** search itself.

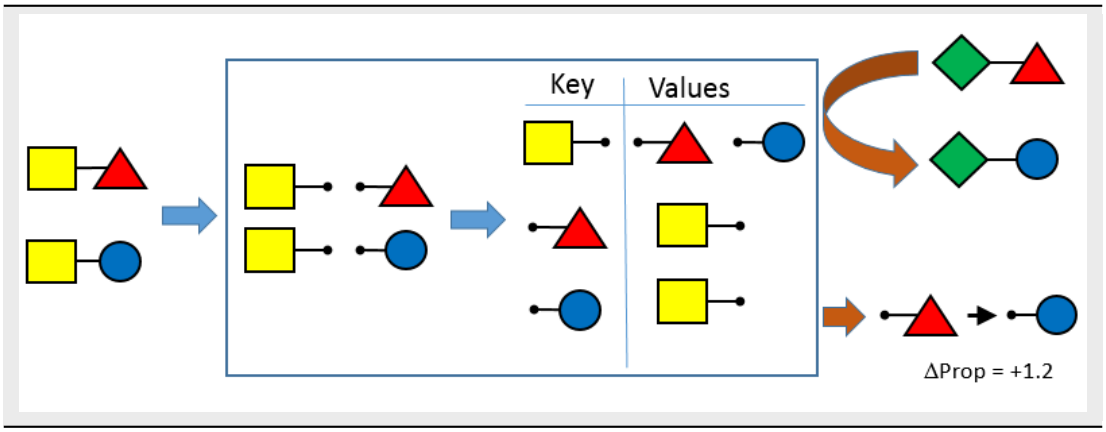


8.18.3 OEMedChem TK

OpenEye is pleased to announce the first official release of **OEMedChem TK**. This medicinal chemistry functionality was previously only available in beta form.

OEMedChem TK provides the ability to index a set of input structures and identify matched molecular pairs. Matched molecular pair analysis is becoming recognized as a powerful tool for the extraction of the effects of chemical changes on a property or properties of interest in large data sets.

Table 9: *Fragmentation and Indexing Approach to Matched Molecular Pair Analyses*



Additional **OEMedChem TK** capabilities include fragmentation and perception, similarity measures, belief theory, and molecular complexity measures. For the full list of capabilities, see the OEMedChem functionalities section.

Note

If you would like to utilize the functionalities of **OEMedChem TK**, please visit https://www.eyesopen.com/contact for a license.

8.18.4 General Notices

- This 2016.Feb release is the last to support Ubuntu 12 for all toolkits.
- OSX 10.8 is no longer supported, but support has been added for OSX 10.11.
- This 2016.Feb release is the last to support Visual Studio 2012 for C++ and C# toolkits and Visual Studio 2010 for Python 3.3 toolkit.
- This 2016.Feb release supports Python 3.5 for the following platforms: OSX 10.10, OSX 10.11, Ubuntu 12, Ubuntu 14, RedHat 6, and RedHat 7.
- The 2016.Jun release is expected to support Python 3.5 on Windows with Visual Studio 2015.
- The 2016.Jun release will add C++ support for Visual Studio 2015.

The Python 2 distributions of **OpenEye Python Toolkits** can now handle both ascii and unicode Python strings for all APIs that accept string arguments. For example, the `OESmilesToMol` function can now parse the following SMILES strings without throwing a 'TypeError' exception:

```
smiles_str = "[N+](=O)([O-])([O-]) nitrát"
smiles_unicode = u"[N+](=O)([O-])([O-]) nitrát"
```

Previously, the unicode strings would work in Python 3, but not in Python 2, making it difficult to write code that was compatible with both major versions of Python at the same time. This major change should make porting to Python 3 much easier for **OpenEye** toolkit users.

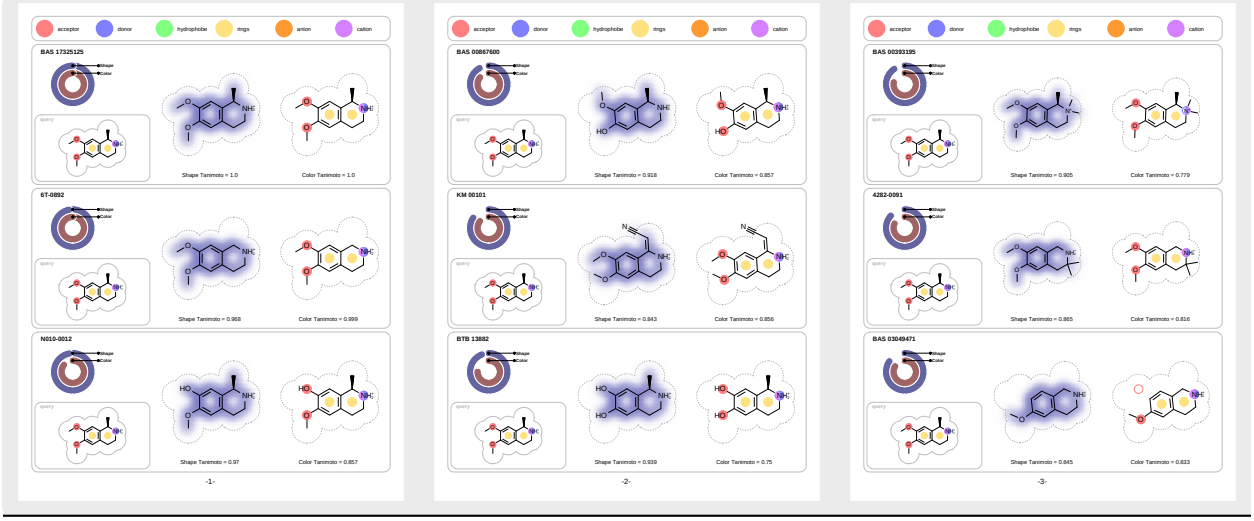
8.19 OEToolkits 2015.Oct

8.19.1 FastROCS TK

OpenEye is pleased to announce that **FastROCS** is joining the pantheon of **OpenEye** toolkits as **FastROCS TK**. Shape technology continues to be at the center of **OpenEye's** scientific mission, and the unprecedented performance gains of **FastROCS TK** continues to enable new science to be explored throughout the molecular modeling community.

> **Note**
>
> **FastROCS TK** is only available in Python on 64-bit Linux platforms.

Table 10: **Example of depicting the results of FastROCS TK** (See code example in Grapheme TK)



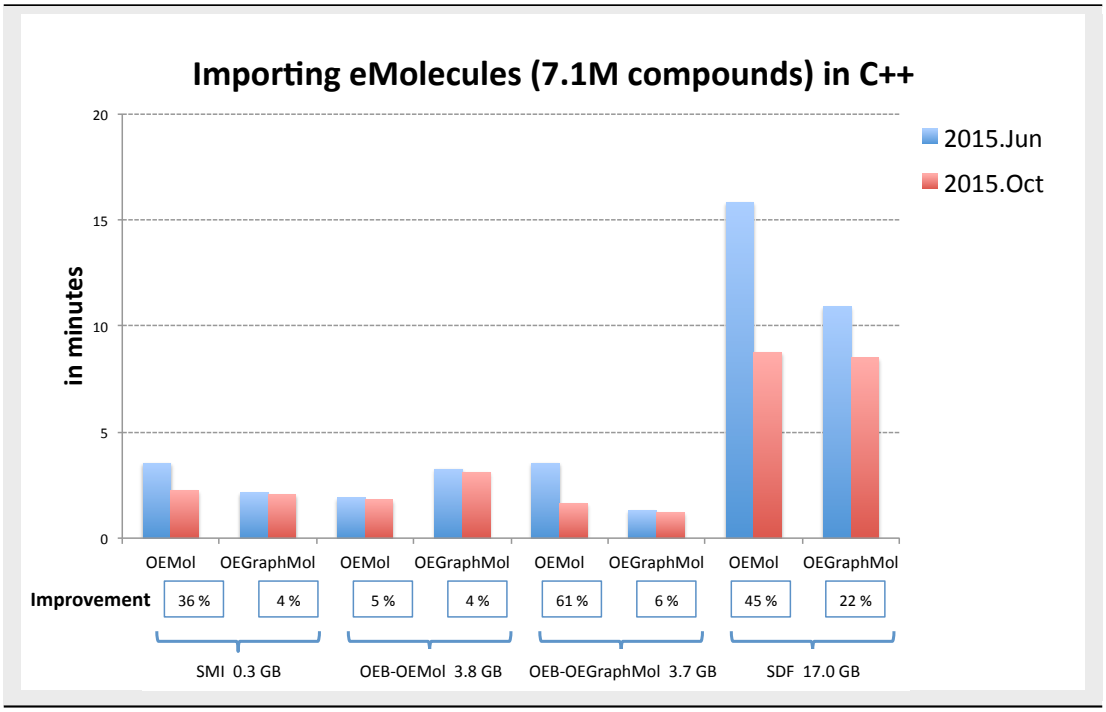
Warning

If your **FastROCS** license was created before May 19th, 2015, please visit https://www.eyesopen.com/contact for a new license. **FastROCS TK** requires that the relevant "python" feature be turned on for access to **FastROCS TK**.

8.19.2 OEChem: Molecule Reading Performance Improvement

In addition to extending the capability of **OpenEye's** toolkits, improving the performance of their existing functionalities is also important to us. We continually work to increase the performance of the molecule readers in **OEChem TK** to accommodate our field's ever-growing molecule databases. Significant effort has been made to accelerate reading molecules into OEMol. The graph below illustrates the overall results of the code optimizations performed since the previous release (*2015.Jun*).

Table 11: **Performance improvement in C++**. See **OEChem TK**'s detailed release notes for the other languages.

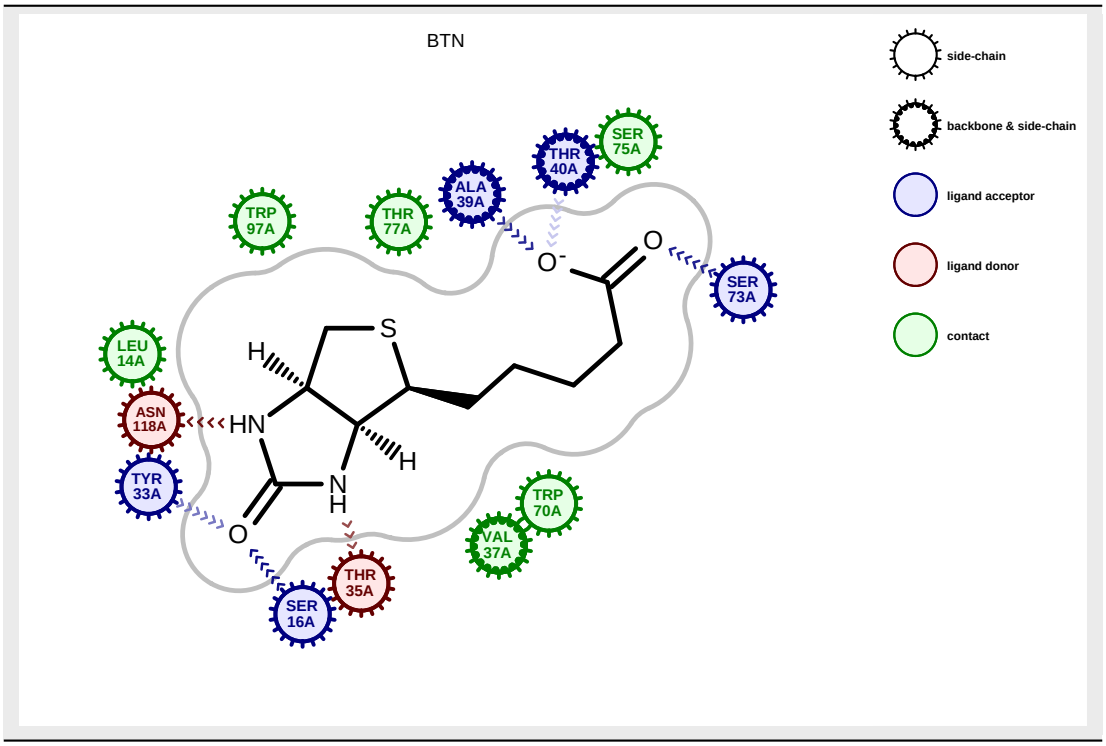


Note

- OEB-OEMol - **OpenEye** OEBinary file format stores molecules as OEMol (multi-conformer molecule representation)
- OEB-OEGraphMol - **OpenEye** OEBinary file format stores molecules as OEGraphMol (single-conformer molecule representation)

8.19.3 OEBio: Fragment Network

The *2015.Feb* release introduced the visualization of receptor-ligand interactions by **Grapheme TK**. The following *2015.Jun* release provided automatic identification of ligands by **OEBio TK**. This release extends these capabilities by providing read-only access to the fragment network data structure that can store interactions perceived by **OEDocking TK**. Several classes and predicates that provide convenient ways to examine various interactions types between a ligand and a protein have been also introduced.



For more details about using these new APIs, see the example in the Perceive and Print Protein-Ligand Interactions section.

8.19.4 General Notices

- **OpenEye Python Toolkits'** installation command has been changed to:

```
$ pip install -i https://pypi.anaconda.org/OpenEye/simple OpenEye-toolkits
...
$ oecheminfo.py
Installed OEChem version: |oechemversion| platform: ubuntu-14.04-g++4.8-x64 built:␣
↪|builddate|
...
```

- This is the last release to support **FastROCS TK** on Ubuntu 12.
- The 2016.Feb release will be the last release to support Ubuntu 12 for all toolkits.
- The 2015.Oct release is the last to support OSX 10.8.

8.20 OEToolkits 2015.Jun

8.20.1 PDB Splitting

OEBio TK includes a powerful new API for automatically classifying and separating the various components of a macromolecular complex by their functional roles. This new API makes it easy to excise a ligand or protein/cofactor complex automatically or identify binding-site residues. The process is also fully customizable to allow internal definitions of the ligand to augment the splitting algorithm. For more information, please see the Splitting Macromolecular Complexes documentation.

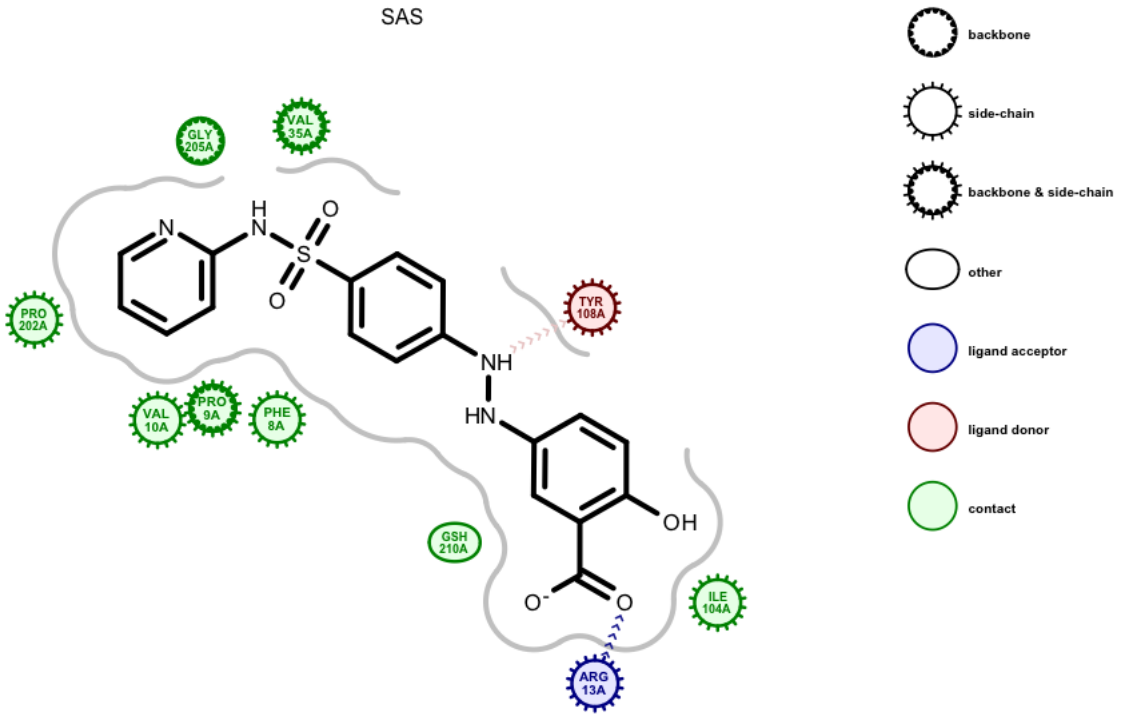
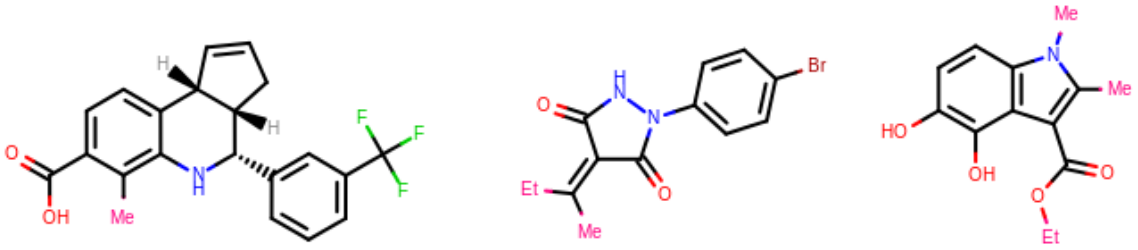


Fig. 19: The above Sulfasalazine ligand was automatically identified in the glutathione S-transferase binding site by using the `OESplitMolComplex` function. The image was created by running the complex2img Grapheme TK example on the 13GS PDB entry. Previous versions required an external definition of the ligand; this new version bypasses that requirement for ease of use.

8.20.2 PAINS Filter

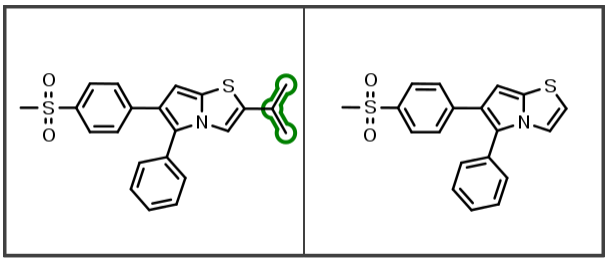
Pan Assay Interference Compounds (PAINS) filter was added to **OEMolProp TK** by adapting the supplementary material from the original [Baell-2010] paper. The PAINS filter can be used as a prefilter set for screening or, more interestingly, as a post-processing filter to identify possible protein-reactive or other assay-confounding functionalities within hits. Any hits so identified will require more care to determine if the measured activity is valid. The PAINS filters can give valuable insights into lead optimization priorities.



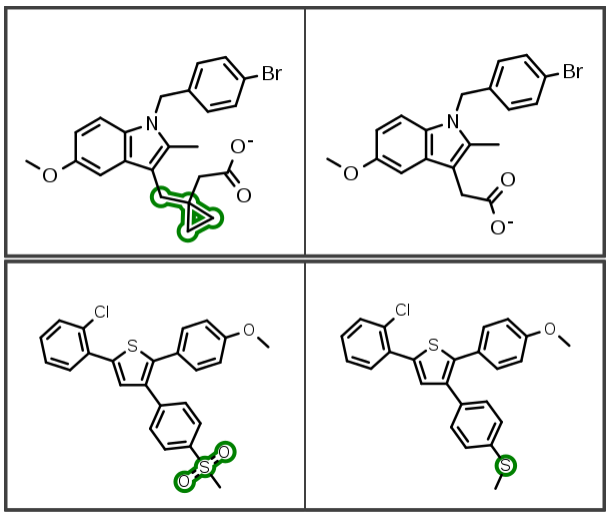
8.20.3 Matched Molecular Pair Improvements

There is a wealth of information stored in a chemical series. Extracting this property change information from chemical analogs is at the heart of a matched molecular pair analysis. One challenge is capturing molecular differences corresponding to substitution of a hydrogen with a larger group in an unsupervised analysis approach. This new indexing capability generates significantly more matched molecular pairs during the analysis and yields additional information about substituent effects on molecular properties. More matched pairs means more information is being extracted from the chemical series.

This next beta release of **OEMedChem TK** improves the matched molecular pair analysis engine to robustly and efficiently support the identification of molecular pairs due to differences in hydrogen-atom substitutions:

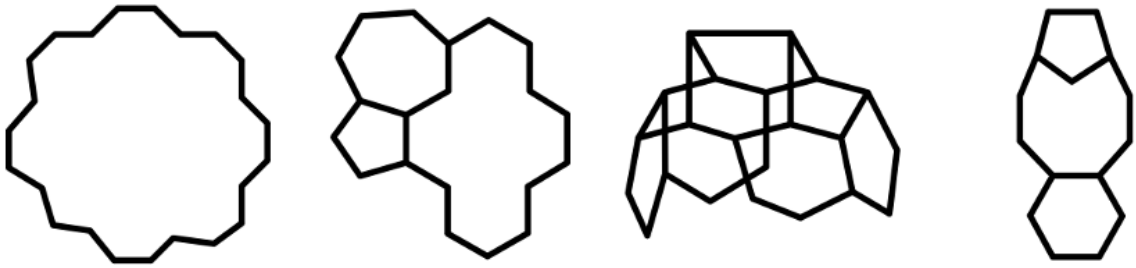


Additionally, indexing improvements also identify new linker insertions and modifications:



8.20.4 Custom Ring Template Dictionaries

Depiction of complex molecules in 2D is both a science and an art. For this reason, **OEChem TK** can now utilize custom ring template dictionaries specified by the user. This allows **OpenEye Toolkits** to adapt immediately to changing project demands. **OpenEye** will gladly continue to incorporate ring templates submitted to us, but this new feature will be more flexible. The Custom Ring Template Dictionary Guide explains the tools and process to start using custom internal dictionaries.



8.20.5 Anaconda Support

We want to work where you do. At **OpenEye**, we support all major platforms and Continuum's Anaconda is becoming a major platform for scientific Python. Anaconda is a Python distribution that installs commonly used scientific packages by default. **OpenEye Python Toolkits** have added binary support for Anaconda.

> **Note**
>
> Using the PIP installation mechanism described below greatly simplifies this compatibility; the user should not even notice that Anaconda Python requires a different binary.

8.20.6 One-Command Python Installation

OpenEye Python Toolkits can now be installed with one command:

```
$ pip install -i https://pypi.binstar.org/OpenEye/simple OpenEye-toolkits
...
$ oecheminfo.py
Installed OEChem version: |oechemversion| platform: ubuntu-14.04-g++4.8-x64 built:␣
↪|builddate|
...
```

See the Quick Start guide to learn how to make this command even shorter:

```
pip install OpenEye-toolkits
```

The command above will automatically select the correct version and operating system in a way that is natural for any pure Python project.

- The Python toolkit package name has been changed from `OpenEye` to `OpenEye-toolkits`. This will change how the previous toolkit is uninstalled from a `virtualenv`.
- `examples`, `docexamples`, and `tests` are now found under the `openeye` module in the `OpenEye-toolkits` package. This allows examples and integration testing to be performed easily after installing the `OpenEye-toolkits` package like a proper Python package. Run the following command to execute the integration tests:

```
(OpenEye)$ python -m openeye.examples.openeye_tests
```

In addition to **OpenEye** examples being placed in the Python `bin` or `Scripts` directory, examples can also be run through the Python interpreter's `-m` option:

```
(OpenEye)$ python -m openeye.examples.oechem.oecheminfo
```

- py2exe should now work. **OpenEye Python Toolkits** will no longer use modules that are blacklisted by py2exe.

8.20.7 General Notices

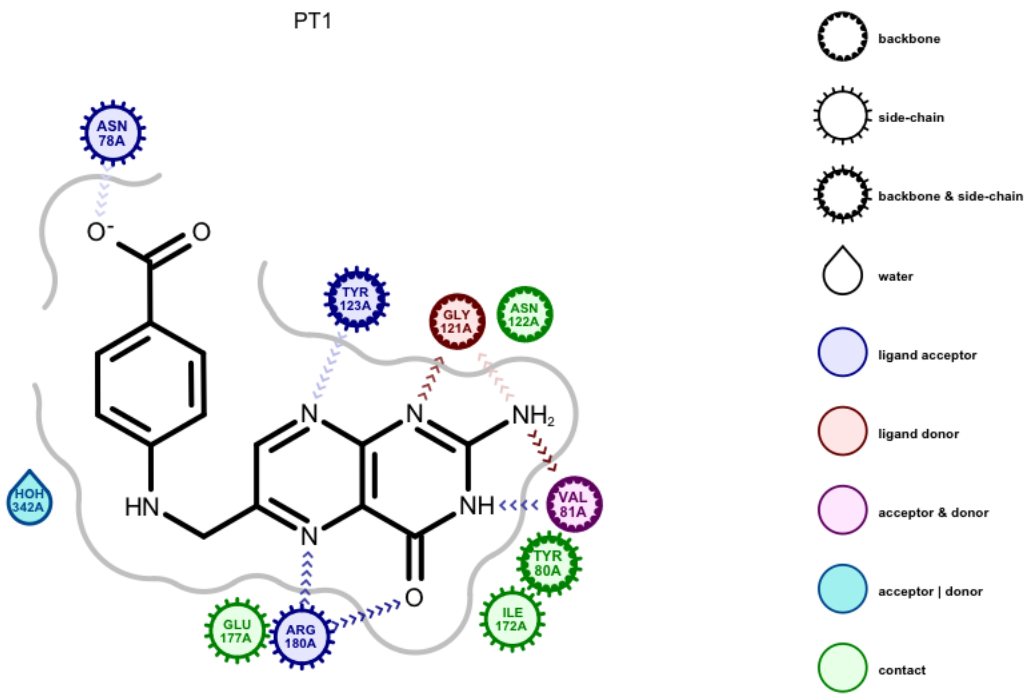
- This is the last release to support **Visual Studio 2010** for C++.
- This is the last release to support **CMake 2.x** in the C++ distributions; all future releases will use **CMake 3.1+**. The CMake files shipped with the C++ toolkits have been made compatible in both **CMake 2.x** and **CMake 3.1+**, but compatibility with **CMake 2.x** will not be tested after this release.

8.21 OEToolkits 2015.Feb

This release of the **OpenEye Toolkits** contains many new features and bug fixes. The most important are highlighted here.

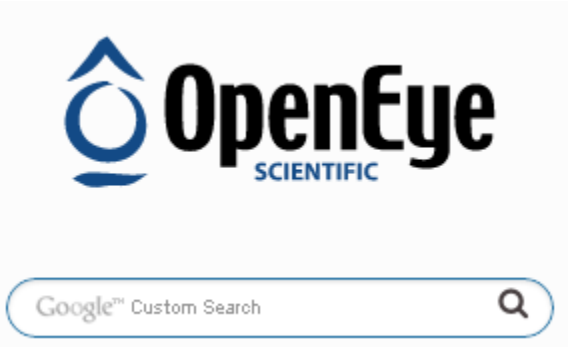
8.21.1 Protein-ligand Rendering

Grapheme TK now includes a powerful new API for the display of 3D information. The interactions between a protein and a ligand provide crucial information to the pursuit of new drugs. To facilitate the deciphering of these complex interactions, **OpenEye** is releasing the ability to render this 3D information in a 2D diagram. Currently, only **OEDocking TK** interactions can be rendered by **Grapheme TK**. However, this feature will be fully extensible through the **OEBio TK**. If you wish to display a particular type of interaction and would like to contribute to the development, please contact support@eyesopen.com.



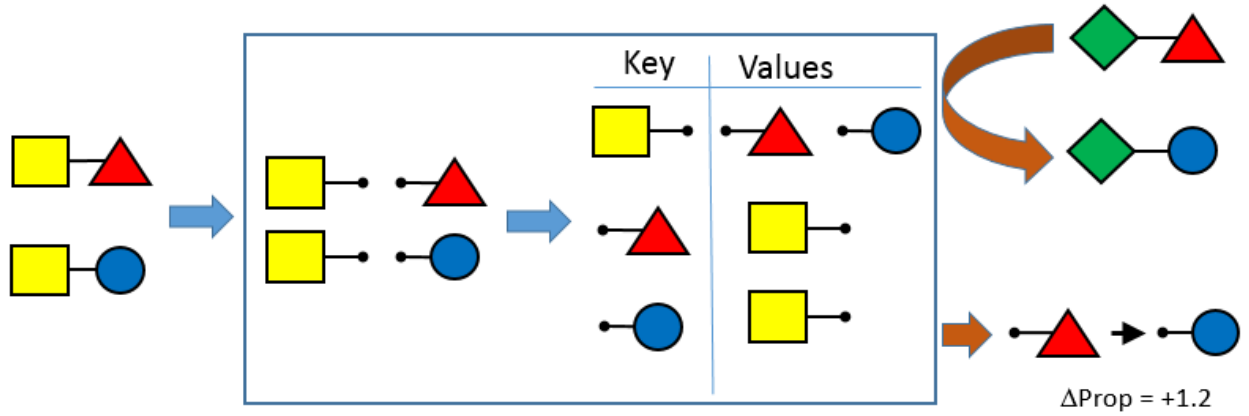
8.21.2 Documentation Refresh

To improve usability and maintain a high level of quality, the **OpenEye Toolkit** documentation has been refreshed over the last year. In addition to a new logo and style, significant functionality has been added. All **OpenEye Toolkits** will now cross-reference, leading users to the correct toolkit while searching for functionality. In addition, searching the HTML documentation can now be performed with Google, offering significantly better search results than the previous HTML documentation. If you are an avid user of the PDF documentation, we strongly encourage you to take another look at the HTML documentation.



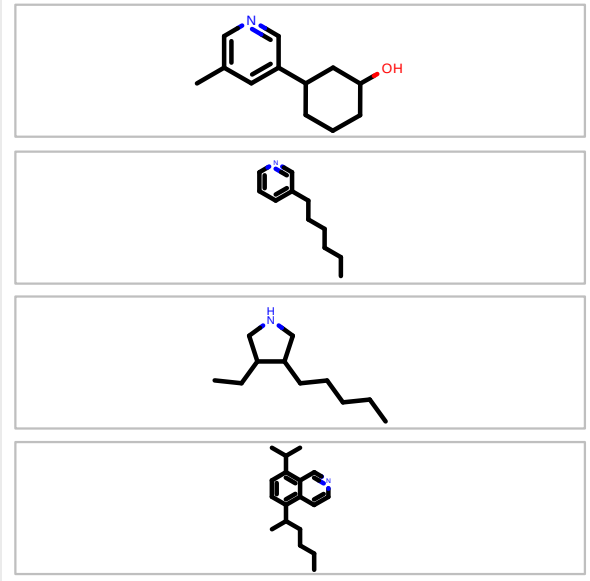
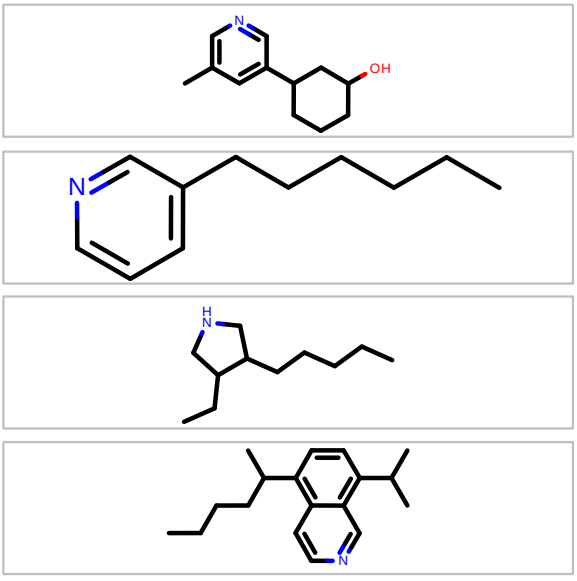
8.21.3 Matched Pairs Transformations

This release continues the incorporation of community suggestions in the **OEMedChem TK**. In this release, the matched molecular pairs functionality has been augmented to include easy access to the transformations themselves in native form, allowing for retrieval and manipulation of transformations for many more matched pair use cases.



8.21.4 Optimized Depiction Orientation

Molecular depictions are a common element in any chemistry environment. However, they can waste a lot of screen real estate, especially when displaying extended or oddly shaped molecules. To maximize screen space, the **OEDepict TK** now provides the ability to optimize for either horizontal or vertical rendering of the molecule coordinates. For example: to display molecules in a spreadsheet, horizontal depictions will conserve the most vertical whitespace; to display molecules in the sidebar of a webpage, vertical rendering will conserve the most horizontal whitespace.

OEDepictOrientation.Default	OEDepictOrientation.Horizontal
	

8.21.5 General Notices

- **SuSe 11** is no longer supported.
- **Visual Studio 2008** for C++ and C# toolkits is no longer supported.
- The 2015.Jun release will be the last to support **Visual Studio 2010** for C++.
- **OSX 10.10** Yosemite support has been added.

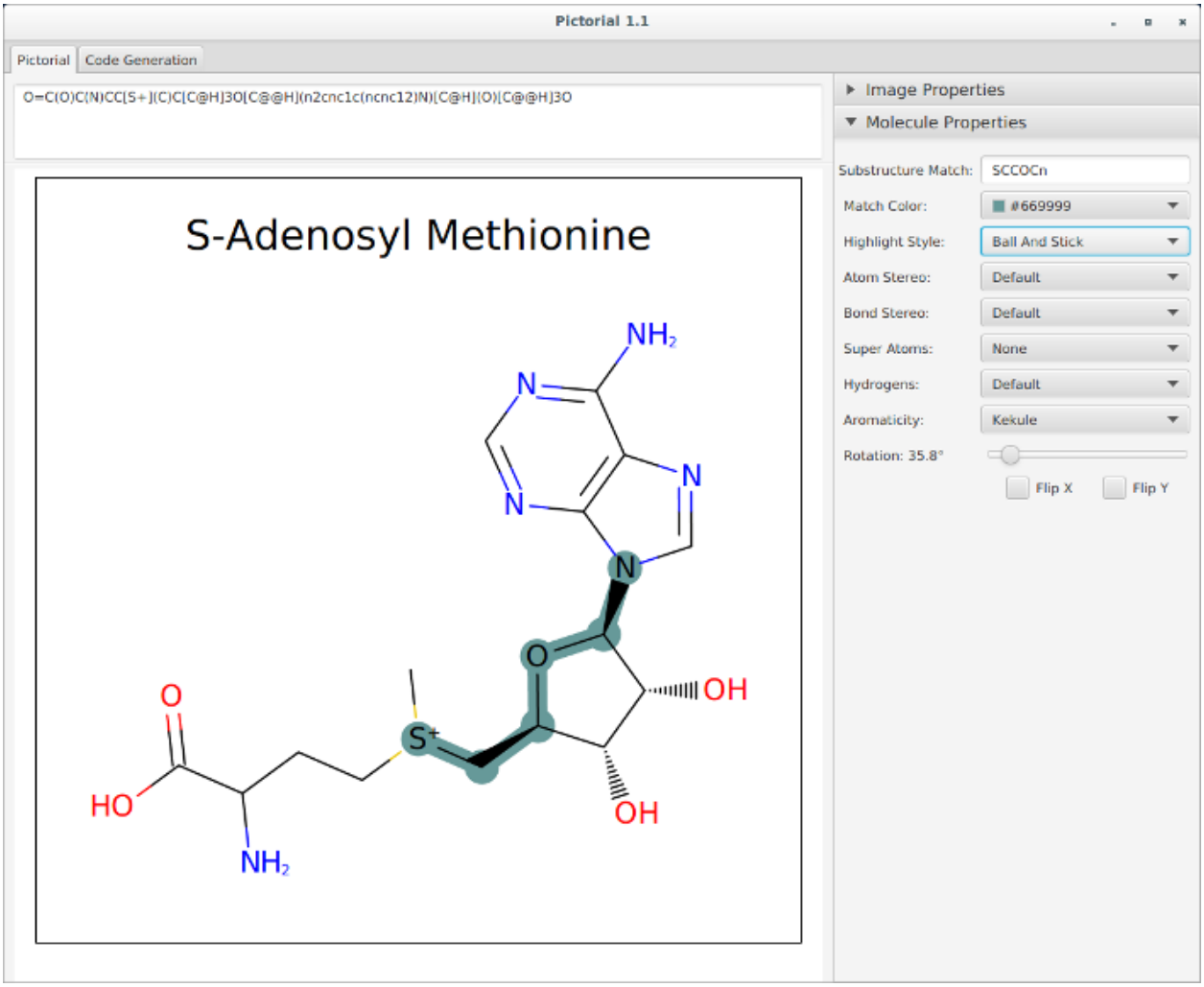
- Babel is a command line tool for molecular file format translation, incorporating a wide range of OEChem options. The application code is now open source and can be found here: https://github.com/oess/OEBabel
- **OpenEye** Python toolkits are now built with SWIG 3.0.2. Users who write and compile their own **OpenEye** C++ toolkit extensions will need to upgrade accordingly. An example of compiling a custom extension to the Python toolkits is given here: https://github.com/oess/oepython_extension.

8.22 OEToolkits 2014.Oct

This release of the OpenEye toolkits contains many new features and bug fixes. The most important have been highlighted here, the full list can be accessed below.

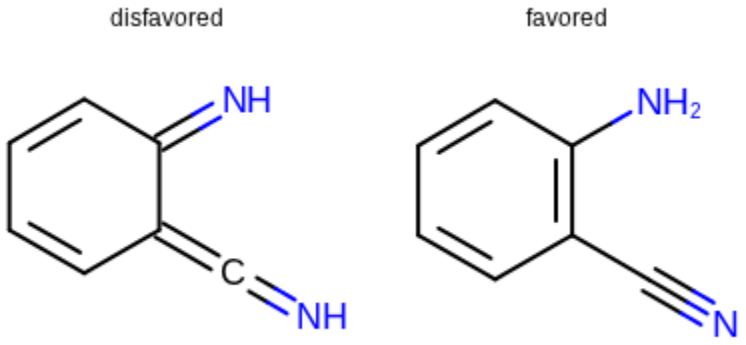
8.22.1 Pictorial application for automatically writing OEDepict code

OEDepict TK is a powerful API for the "2D" depiction of small molecules. While such depictions are universal in chemistry there are many variants in drawing style, the majority of which are captured in OEDepict's extensive set of options. To aid navigation through the many options available we now provide a small application, **Pictorial**, written in Java, that will autogenerate code that can then be used in your own programming. The figure below is an illustration of the interface.



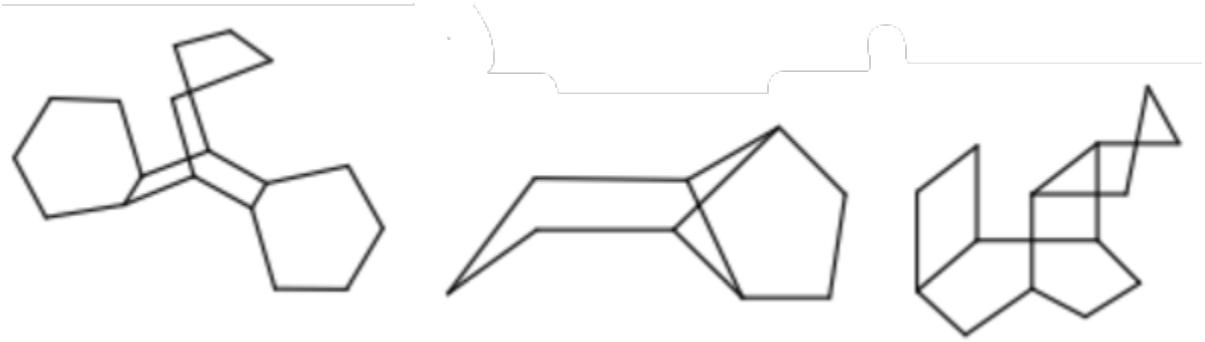
8.22.2 Major improvements to Quacpac's protomer assignment algorithms

Quacpac TK is a set of tools for the assignment of charges, ionization states and proton positions. The latter has been significantly improved to produce more reasonable tautomeric forms, for instance by disfavoring exocyclic bonds adjacent to aromatic bonds (e.g. favoring aromatic forms). An example is depicted below.



8.22.3 Major improvements to depiction of complex ring systems

Depiction of complex molecules in 2D is both a science and an art. Algorithms help with the majority of cases but sometimes they have to be supplemented with the 'art' of hand-drawn templates. In our continuing quest to provide the best 2D depiction of molecules, this release includes an additional 550 such templates to **OEDepict** and 2D coordinate generation in **OEChem**. Some examples are shown below.



8.22.4 Multi-precision conformer handling

There are times where **precision** is important and times when, due to memory consumption, it gets in the way of computation. As such, this release of the toolkits includes **two new levels of precision**. In OEChem the basic molecular object OEMol can now store conformer coordinates as 16-bit 'half-floats', 64-bit double precision number or long doubles (typically 80 bits although storage requirements may vary). Half-floats have five bits reserved for the exponent, eleven for the fraction. Examples of where it might be useful include shape representation (robust to small coordinate changes) or compact storage. Extra-precision is useful when interfacing with quantum programs or for storage in rotor-offset form in OEB files.

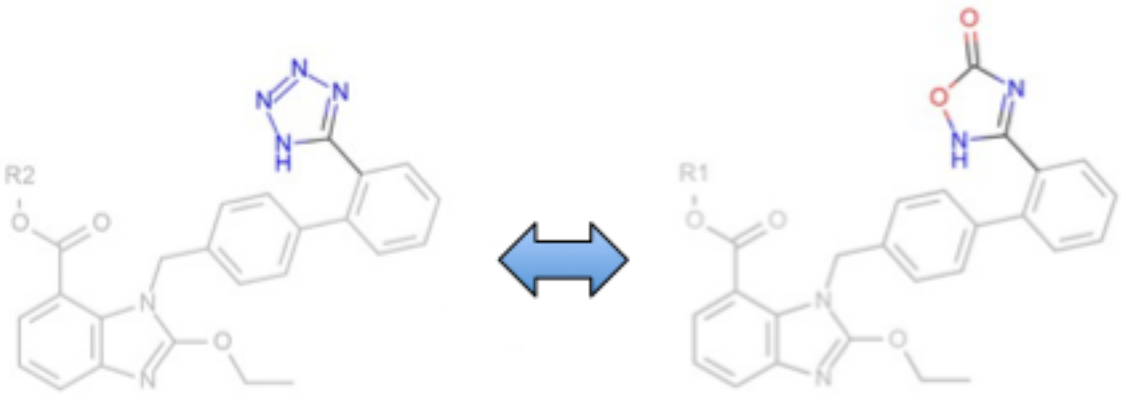
> **Warning**
>
> C++ users may notice that *OEChem::OEMCMolBaseT* and *OEChem::OEConfBaseT* have been renamed to OEMCMolBase and OEConfBase respectively. This change was to enable the above feature and to ease the use of the documentation by Python, Java, and C# users who should see no effect from this change.

8.22.5 New force field in Szybki

Szybki TK encompasses our capabilities for molecular optimization. This version of the toolkit includes our latest advances in the calculation of intermolecular interactions, i.e. the combination of dispersion (attractive), Fermi overlap (repulsion) and Coulombic interactions (either). These are captured in the IEFF formulation, which represents the first two interaction terms in more detail than typical VdW forms. In addition, Coulombic interactions are calculated using atom-centered point multipoles, as output by quantum chemistry codes. OpenEye provides facilities to input these into OEB files for proteins and ligands, allowing a new level of sophistication in the evaluation of complex interactions. In addition, the Szybki toolkit also includes a new function call to allow for specifically constraining a single torsion and new methods to assign flexibility to residues during protein-ligand optimization.

8.22.6 Matched Pairs analysis beta API added to OEMedChem TK

This release includes an exciting beta of **OEMedChem** which includes containing facilities for "Matched Pair" analysis. The matched pair concept has resonated in the medicinal chemistry community because it provides a simple method of comparing two molecules, i.e. ascertaining whether they differ in chemically obvious ways, e.g. a swap of a fluorine for a chlorine or a benzamide for a benzamidine, and then using historical data on such transformations to make predictions on property differences. The functions within the OEMedChem toolkit in this release are capable of searching datasets for such pairings, allowing for such prior expectations to be tabulated. We stress this is a beta release feature and look forward to incorporating community suggestions and requests.



8.22.7 Major bug fixes

Warning

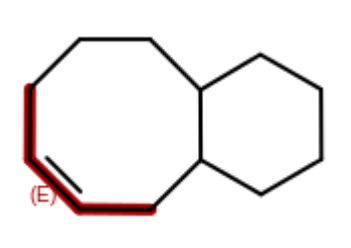
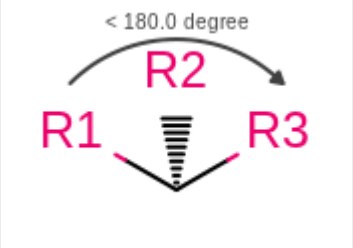
It is strongly recommended for users of the 2014.Feb and 2014.Jun releases to upgrade to the 2014.Oct release because of the following major bug fixes for the SDF, MDL, and CDX file formats.

The addition of 2D coordinate generation for the SDF, MDL, and CDX file formats in the 2014.Feb release, **OEChem 2.0.0**, created the possibility of erroneously changing stereo chemistry in the following cases:

1. Bond stereo chemistry in macrocycles
2. Atom stereo from wedges or hashes between atoms with an angle less than 180 degrees

The 2D coordinate generation can not always generate coordinates that represent the required stereo chemistry. For these cases, **OEChem** will now leave the coordinates in the resulting file all zero. This will make file readers choose to honor the stereo marks instead of the 2D coordinates.

Table 12: **Examples of stereo chemistry problems addressed in 2014.Oct**

[1]	[2]
	

8.22.8 General notices

- The following is no longer supported on RedHat Enterprise Linux 5:
  - Python 2.6
  - gcc 4.3
  - gcc 4.4
- **Ubuntu 10.04** is no longer supported.
- This will be the last release to support **SuSe 11**.
- **OSX 10.7** is no longer supported.
- This will be the last release to support **Visual Studio 2008** for C++ and C# toolkits.

Several notable **improvements** have been made to the **Java** distributions:

- The OEJavaImageBase class was missing some of the newer drawing primitives added recently to the OEImageBase class. The OEJavaImageBase class now fully implements the complete OEImageBase abstract base class to allow drawing into the native Java `Graphics2D` object: http://docs.oracle.com/javase/7/docs/api/java/awt/Graphics2D.html
- New API's which support Java collections were added to the molecule objects:

OEMolBase	OEMCMolBase
java.util.List<OEAtomBase> GetAtomList()	java.util.List<OEConfBase> GetConfList()
java.util.List<OEBondBase> GetBondList()	
java.util.List<OEGroupBase> GetGroupList()	

- The Picto example was removed from the distribution, and was replaced with a Java application, Pictorial.

8.23 OEToolkits 2014.Jun

This release of the OpenEye toolkits is focused on stability and new platform support. The last release, 2014.Feb, was a major feature release introducing numerous new features. This release focused on fixing many bugs and improving the overall stability of the OpenEye toolkits.

There is still a major new feature being added in this release:

- FreeForm API added to **Szybki TK**

The following changes were made across all the toolkits:

- Java and C# should see greatly improved stability. All OpenEye toolkits will now be much more conservative with the use of stack based memory allocation. Stack memory allocation was causing non-deterministic crashes in environments with limited stack space, e.g., Java and C#.

Several notable **improvements** have been made to the **Java** distributions:

- Java 8 is fully supported by the OpenEye Java toolkits.
- Multiple OpenEye jars can be combined together to make platform independent runnable jars. The architecture of the JVM is now detected, and the appropriate shared library will be loaded.
- OpenEye jars are now cryptographically signed. Use *jarsigner -verify <jarname>* to verify the jar.

General notices:

- **Windows XP** is no longer a supported version of Microsoft Windows.
- Added support for **Ubuntu 14.04**, a long term support distribution that OpenEye will support for 4 years.
- This release will be the last release to support **Ubuntu 10.04**.
- Added support for **RHEL 7**. This is the next major release from RedHat that will be supported by OpenEye for at least 5 years. OpenEye will be very rapidly phasing out support for **RHEL 5**. **RHEL 5** users should be planning their upgrade now.

  To start the phase out, this will be the last release to support the following on **RHEL 5**:

  - Python 2.6
  - gcc 4.3
  - gcc 4.4
- This release will be the last release to support **OSX 10.7**.
- The next release, 2014.Oct, will be the last release to support **Visual Studio 2008** for C++ and C# toolkits.
- Deprecated APIs will now throw a warning message once, and only once, for the lifetime of the process.
- Documentation examples for Java and Python distributions should now have documentation markup removed.

8.24 OEToolkits 2014.Feb

This release marks a major milestone for **OEChem**: 10 years as an OpenEye product! In recognition of that, we are releasing **OEChem** 2.0 to clean up some rough edges and bring in some important new features. We are committed to maintaining a stable **OEChem** API and thus the vast majority of programs written with previous versions of **OEChem** should continue to be supported in the **OEChem** 2.x series. The only "backwards breaking" functional change that everyone will see is something that we feel all users will appreciate:

> **Warning**
>
> **'.smi' will now output molecules with stereochemistry.**

A few of the new major features in **OEChem** are:

- `OEGenerate2DCoordinates` added for 2D coordinate generation and is automatically called when writing `.sdf`, `.mdl`, and `.cdx` files when the associated molecule had no coordinates.
- Support for `.csv`, comma-separated values, added as a supported molecule file format.
- Fast random access to all **OEChem** file formats through the new OEMolDatabase class.

- `OEAddLicenseFromHttp` added to Python, Java, and .NET toolkits to enable fetching license information from a server instead of the standard text file. This is not currently supported in C++.

This release also features the addition of **two new toolkits** to the distribution:

- **Szmap TK** – this provides access to the core functionality used by the **SZMAP** application and enables the calculation of solvent energies and probabilities at arbitrary points in space. It also provides access to probe geometry for output and display.
- **OEMedChem TK** – this is still in the experimental stages and is not yet officially supported; however, it provides many useful and interesting tools that we wanted to make available while we finalize the API and functionality set. As such, the existing API may (and most likely will) change in a future toolkit release.

Several notable **improvements** have been made to the **Java** distributions:

- The layout of the Java package has been simplified. In general the directory structure was flattened out. Unzipping the distribution now creates a directory: OpenEye-Java-Version-Platform-Arch. This directory includes the following subdirectories and files: lib, doc, examples, verify, and README.txt. The OpenEye jar is under the lib directory. The OEChem Quick Start Guide and Java Best Practices document are in the docs directory. Run verify.py in the verify directory to ensure a working distribution.
- All Java APIs that take primitive arrays as arguments have been optimized.
- The Java toolkits no longer run a separate finalizer thread to automatically `close()` file-like objects as it was not guaranteed to run `close()` for all objects. Users are required to call `close()` for any object that provides that method.
- The Java toolkits will no longer use `OESwigBase` and `OENativePointer` to circumvent the Java garbage collector. These constructs are no longer necessary with the newer versions of Java and were creating a small overhead in newer versions of Java.

General notices:

- Added support for **OSX 10.9** Mavericks.
- The next toolkit release, 2014.Jun, will be the last release to support **OSX 10.7**.
- This release will be the last release to support **OSX 10.6**.
- The next toolkit release, 2014.Jun, will be the last release to support 64-bit **Ubuntu 10.04**.
- **GCC 4.8.2** support added for **RHEL6**. GCC 4.8.1 had a bug that made it impossible to compile OpenEye header files. Please use 4.8.2+.
- Experimental support for **Python 3.3** added.

8.25 OEToolkits 2013.Oct

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.9.3

OEBrood TK
0.9.0

OEDepict TK
2.2.1

OEDocking TK
1.2.1

Grapheme TK
1.1.0

GraphSim TK
2.0.6

Grid TK
1.4.4

Lexichem TK
2.3.0

MolProp TK
2.1.7

Omega TK
2.5.3

Quacpac TK
1.6.4

Shape TK
1.9.2

Spicoli TK
1.2.2

Szybki TK
1.7.5

Zap TK
2.1.6

- Two more important "idiomatic differences" have been added to the Introduction section of the Java Quick Start guide.
  - Calling close is required for any object that provides that method.
  - Explicitly calling delete is recommended on objects from certain libraries.
- The Java OpenEye toolkits will now allocate memory from the Java heap once again. The Java OpenEye toolkits should not exceed the memory usage specified by the Java -Xmx flag. The problem first appeared in the 2011.Oct release when the JNI_OnLoad symbol was made hidden.
- Many Java functions and methods that returned iterators are now usable from Java. These methods used to return opaque SWIGTYPE_P_<class_name> objects.
- A new Java example was added: Picto. This example requires JavaFX 2.2 and Java version 1.7.0.21 or higher. Please refer to the JavaFX 2.2 requirements here. To run this example execute *ant picto* in the directory that contains the examples jar file.
- No OpenEye toolkit will call `OEErrorHandler.Fatal` anymore. All instances have been replaced with `OEErrorHandler.Error` allowing end-users the ability to control whether programs should exit on error with `OEErrorHandler.SetStrict`. It is recommended to use `OEThrow.SetStrict(false)` inside long running services like web servers.

8.26 OEToolkits 2013.Jun

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.9.2

OEDepict TK
2.2.0

OEDocking TK
1.2.0

Grapheme TK
1.0.6

GraphSim TK
2.0.5

Grid TK
1.4.3

Lexichem TK
2.2.3

MolProp TK
2.1.6

Omega TK
2.5.2

Quacpac TK
1.6.2

Shape TK
1.9.1

Spicoli TK
1.2.1

Szybki TK
1.7.4

Zap TK
2.1.6

- Most toolkit distributions do not ship with documentation anymore due to the sheer size of the documentation. Only the Python and C# Windows installers will continue to contain documentation. Documentation is still available as a separate download. Most users appear to rely on the OpenEye online documentation, that will continue to be maintained.
- A test script was added that will automatically run several of the included examples to ensure the Java toolkit distribution is running correctly. See the readme.txt file in the `openeye/java/verify` directory.
- This release will be the last release to support SuSe 10.

> **Note**
>
> The last release of the Java toolkits to support SuSe 10 was the 2012.Jun compatibility version.

8.27 OEToolkits 2013.Feb

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.9.1

OEDepict TK
2.1.0

OEDocking TK
1.1.4

Grapheme TK
1.0.5

GraphSim TK
2.0.4

Grid TK
1.4.2

Lexichem TK
2.2.2

MolProp TK
2.1.5

Omega TK
2.5.1

Quacpac TK
1.6.1

Shape TK
1.9.0

Spicoli TK
1.2.0

Szybki TK
1.7.3

Zap TK
2.1.5

- OSX 10.8 support added for C++, Python, and Java. Note, C++ and Python are built with the Clang 4 compiler.
- Visual Studio 2012 support added for C++ and C#.
- OSX 10.6 32-bit python is no longer supported.
- The next release, 2013.Jun, will be the last release to support SuSe 10.

8.28 OEToolkits 2012.Oct

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.9.0

OEDepict TK
2.0.4

OEDocking TK
1.1.3

Grapheme TK
1.0.4

GraphSim TK
2.0.3

Grid TK
1.4.1

Lexichem TK
2.2.0

MolProp TK
2.1.4

Omega TK
2.5.0

Quacpac TK
1.6.0

Shape TK
1.8.3

Spicoli TK
1.1.3

Szybki TK
1.7.2

Zap TK
2.1.4

- `*GetSite` and `*GetLicensee` free functions exposed by each toolkit now directly return a string to be more usable in all wrapped languages.

8.29 OEToolkits 2012.Jun

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.8.0

OEDepict TK
2.0.3

OEDocking TK
1.1.2

Grapheme TK
1.0.3

GraphSim TK
2.0.2

Grid TK
1.4.0

Lexichem TK
2.1.2

MolProp TK
2.1.3

Omega TK
2.4.6

Quacpac TK
1.5.2

Shape TK
1.8.2

Spicoli TK
1.1.2

Szybki TK
1.7.1

Zap TK
2.1.3

- This release brings about the following changes in platform support:
  - Added support for 64-bit Ubuntu 12.04 LTS. A reminder that 32-bit will not be supported on future linux distributions.
  - Last release to support Visual Studio 2003. Please upgrade to Visual Studio 2008 or 2010.
  - Last release to support Python 2.5 on Windows. Please upgrade to Python 2.6 or 2.7.
  - Last release to support OSX 10.5. Please upgrade to OSX 10.6 or 10.7.

Warning

OEChem has undergone a major revision change from 1.7.7 to 1.8.0 to indicate that bugs were fixed in isomeric canonical smiles generation that could potentially change the smiles generated. A benchmark over 24.0 million unique compounds only yielded changes in 42 of the compound's isomeric canonical smiles. These 42 compounds fall into the following classes:

- broken porphyrins
- cross linked organometallics
- cross linked high molecular weight macrocycles
- trivalent and tetravalent oxygens
- polycyclic hydrazine dyes

See the OEChem release notes for a more specific description of these minor algorithmic changes.

8.30 OEToolkits 2012.Feb

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.7.7

OEDepict TK
2.0.2

OEDocking TK
1.1.1

Grapheme TK
1.0.2

GraphSim TK
2.0.1

Grid TK
1.3.5

Lexichem TK
2.1.1

MolProp TK
2.1.2

Omega TK
2.4.5

Quacpac TK
1.5.0

Shape TK
1.8.1

Spicoli TK
1.1.1

Szybki TK
1.7.0

Zap TK
2.1.2

All the toolkits now use an extension of the MMFF94 force field for tricoordinate boron compounds. Please refer to the Omega TK and Szybki TK release notes for a more complete description of what types of boron are supported.

8.31 OEToolkits 2011.Oct

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.7.6

OEDepict TK
2.0.1

OEDocking TK
1.1.0

Grapheme TK
1.0.1

GraphSim TK
2.0.0

Grid TK
1.3.5

Lexichem TK
2.1.0

MolProp TK
2.1.2

Omega TK
2.4.4

Quacpac TK
1.5.0

Shape TK
1.8.1

Spicoli TK
1.1.1

Szybki TK
1.6.0

Zap TK
2.1.2

This release is meant to be a minimal bug fix release to address a few key issues in the *2011.1* release.

- The *2011.1* release left internal debugging code turned on during all builds. The result was spurious application failures for corner case conditions. In general, the *2011.Oct* version should run faster due to this being turned off.
- The package wide version number has been tweaked due to customer feedback to include an explicit month as the 2nd field to avoid confusion with the version number looking like a date. The version number will now be structured as the following:

  [year].[month].[build]

  year
  is the year the package was released

  month
  is the month the package was released

  build
  is an arbitrary number used for internally tracking the exact state of the package

- Added the username to the temporary directory used by the new Java JAR scheme. This fixes a bug where two users on Linux could get a permission conflict resulting in a failure to load the shared libraries.

8.32 OEToolkits 2011.1

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.7.5

OEDepict TK
2.0.0

OEDocking TK
1.1.0

Grapheme TK
1.0.0

GraphSim TK
2.0.0

Grid TK
1.3.5

Lexichem TK
2.1.0

MolProp TK
2.1.2

Omega TK
2.4.4

Quacpac TK
1.5.0

Shape TK
1.8.1

Spicoli TK
1.1.1

Szybki TK
1.6.0

Zap TK
2.1.2

- C# support added across all the toolkits. See the *C# Quick Start* guide to get started using this new language binding.
- Renaming *Ogham TK* to *OEDepict TK*. *OEDepict TK 2.0* is a major redesign of this core rendering library to enable new technology advances in 2D molecular communication.

Warning

The *Ogham TK 1.x* API is deprecated. The API is still available in *OEDepict TK 2.0*, but will be removed in a future release.

- Introducing a major new library, *Grapheme TK*, based upon *OEDepict TK 2.0*.
- Changing *OEToolkits* package version to no longer track the version of *OEChem* inside the package. The version number will now be structured as the following:

  [year].[release].[build]

  year
  is the year the package was released

  release
  is a sequentially increasing number marking the release within the year

  build
  is an arbitrary number used for internally tracking the exact state of the package

- Changed the default visibility of symbols in *OpenEye* libraries to be hidden. This can drastically lower the number of symbols exported from a given library improving compile times and shared library load time.
- When building with Microsoft Visual Studio, the OpenEye header files will enforce that you are building with the correct version of the compiler.
- The Java toolkits now ship as a single JAR file to ease use. Note, if a smaller JAR file is desired a custom JAR file can be created for only a subset of toolkits that can also work across platforms. See the new *Java Quick Start* guide for more information.
- Note that there are two sets of distributions for Linux. For the most part, if you are using a newer version of Linux (RHEL 5, RHEL 6, SLE 11 and Ubuntu 10.04 LTS), the regular distributions marked "Linux-x64" and "Linux-x86" are what you want. These are built with a newer version of GCC and provided a noticeable increase in performance over previous releases.

But, for RHEL 4 and SLED 10, these new releases are not compatible. For these two older Linux platforms, we've provided a pair of downloads labeled "Linux-x64-compat" and "Linux-x86-compat". Note that these "compat" releases will work on all the supported Linux platforms, albeit with a noticeable decrease in performance.

8.33 OEToolkits 1.7.4

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.7.4

GraphSim TK
1.0.0

Grid TK
1.3.4

Lexichem TK
2.0.2

MolProp TK
2.1.1

Ogham TK
1.7.1

Omega TK
2.4.1

Quacpac TK
1.4.1

Shape TK
1.8.0

Spicoli TK
1.1.0

Szybki TK
1.5.0

Zap TK
2.1.1

Docking TK
1.0.0

- This release of the OpenEye toolkit suite introduces a major new library, the :Docking TK:.
- The license error handling has been significantly improved to provide more descriptive error messages for why exactly a given license has failed. So instead of seeing "General failure to decrypt" as the license failure reason it will actually report "expired" or "version number" if that is the reason.
- The `OE*IsLicensed` functions provided in each library will now report when the license expired if it is already expired instead of just erroring out.

Note

The above two changes do not affect licenses at all. Old and new licenses should continue to work exactly the same.

8.34 OEToolkits 1.7.2

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.7.2

GraphSim TK
1.0.0

Grid TK
1.3.3

Lexichem TK
2.0.0

MolProp TK
2.1.0

Ogham TK
1.7.1

Omega TK
2.4.0

Quacpac TK
1.4.1

Shape TK
1.7.2

Spicoli TK
1.1.0

Szybki TK
1.4.0

Zap TK
2.1.1

- There are two major additions to the OpenEye toolkit suite with this release:

  GraphSim TK

  A toolkit for doing 2D molecule similarity. The methods currently implemented are:

  - Path based fingerprints
  - MACCS key fingerprints
  - LINGOS

  MolProp TK

  A toolkit derived from OpenEye's FILTER technology. This toolkit is useful for calculating an extensive list 2D molecular properties and building custom filters around these properties.

  > **Note**
  >
  > The version number of *MolProp TK* will track with the FILTER application to indicate similarities.

- This release of OEChem introduces many bug fixes and new features provided they didn't break source-level compatibility with 1.7.0.

- This release completes the transition from the old LaTeX based documentation system to the new reStructuredText based system. This includes better support for Python and Java in the manuals as separate manuals for every library.

The following release notes are only for the java toolkit:

- JDK1.5 would sometimes crash while loading the 1.7.0 libraries. This bug was introduced in 1.7.0 with the change to make all OpenEye library memory allocation use memory from the Java heap. However, JDK1.5 has a known bug http://bugs.sun.com/view_bug.do?bug_id=6253386 which doesn't allow accessing the JVM during library initialization. Therefore, we now detect when the OpenEye libraries are being loaded by JDK1.5 and turn off Java heap allocation during library initialization. The amount of memory allocated during library initialization is usually negligible, so this shouldn't allow programs to exceed the Java heap restrictions very much.

Warning

This fix introduces a rare corner case that could cause the OpenEye libraries to leak memory. It will only occur if the following sequence of events occurs:

Table 13: **Race Condition Causing Memory Leak**

Thread 1	Thread 2
Load *OEChem*	
Allocate *OEChem* objects	
	Start loading another OpenEye library (e.g. OEDepict)
Deallocate *OEChem* objects	
This thread is terminated causing the *OEChem* objects to leak	
	Finish loading library

If this occurs *OEChem* will throw the following warning:

```
Leaking memory (0x00000000) because OEUseJavaHeap is false
Contact OpenEye support if this message is seen
```

Note

None of this is possible in JDK1.6.

- *OEChem::OEIter::Target* and *OEChem::OEIter::next* will now raise `NoSuchElementException` to better adhere to the `Iterable` interface.
- Quacpac: Added support for writing a custom OETautomerMolFunction or OETyperMolFunction in Java by deriving from the base class.

8.35 OEToolkits 1.7.1

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.7.1

GraphSim TK
1.0.0

Grid TK
1.3.3

Lexichem TK
2.1.0

MolProp TK
Ogham TK
Omega TK
1.4.0

Quacpac TK
1.4.1

Shape TK
1.4.0

Spicoli TK
1.1.0

Szybki TK
Zap TK
2.1.1

- There are two major additions to the OpenEye toolkit suite with this release:

  GraphSim TK

  A toolkit for doing 2D molecule similarity. The methods currently implemented are:

  – Path based fingerprints
  – MACCS key fingerprints
  – LINGOS

  MolProp TK

  A toolkit derived from OpenEye's FILTER technology. This toolkit is useful for calculating an extensive list 2D molecular properties and building custom filters around these properties.

  Note

  The version number of *MolProp TK* will track with the FILTER application to indicate similarities.

- This release of OEChem introduces many bug fixes and new features provided they didn't break source-level compatibility with 1.7.0.
- This release completes the transition from the old LaTeX based documentation system to the new reStructuredText based system. This includes better support for Python and Java in the manuals as separate manuals for every library.
- JDK1.5 would sometimes crash while loading the 1.7.0 libraries. This bug was introduced in 1.7.0 with the change to make all OpenEye library memory allocation use memory from the Java heap. However, JDK1.5 has a known bug http://bugs.sun.com/view_bug.do?bug_id=6253386 which doesn't allow accessing the JVM during library initialization. Therefore, we now detect when the OpenEye libraries are being loaded by JDK1.5 and turn off Java heap allocation during library initialization. The amount of memory allocated during library

initialization is usually negligible, so this shouldn't allow programs to exceed the Java heap restrictions very much.

Warning

This fix introduces a rare corner case that could cause the OpenEye libraries to leak memory. It will only occur if the following sequence of events occurs:

Table 14: **Race Condition Causing Memory Leak**

Thread 1	Thread 2
Load *OEChem*	
Allocate *OEChem* objects	
	Start loading another OpenEye library (e.g. OEDepict)
Deallocate *OEChem* objects	
This thread is terminated causing the *OEChem* objects to leak	
	Finish loading library

If this occurs *OEChem* will throw the following warning:

```
Leaking memory (0x00000000) because OEUseJavaHeap is false
Contact OpenEye support if this message is seen
```

Note

None of this is possible in JDK1.6.

- *OEChem::OEIter::Target* and *OEChem::OEIter::next* will now raise `NoSuchElementException` to better adhere to the `Iterable` interface.
- Quacpac: Added support for writing a custom OETautomerMolFunction or OETyperMolFunction in Java by deriving from the base class.

8.36 OEToolkits 1.7.0

This is a new release of the OpenEye Toolkits with versions of the following libraries:

OEChem TK
1.7.0

Grid TK
1.3.2

Lexichem TK
1.9.0

Ogham TK
1.7.0

Omega TK
2.3.3

Quacpac TK
1.4.0

Shape TK
1.7.1

Spicoli TK
1.0.2

Szybki TK
1.3.4

Zap TK
2.1.1

- This release starts the transition from the old LaTeX based documentation system to the new reStructuredText based system.
- Starting with *OEChem* 1.7.0 the core OpenEye toolkits (`OEPlatform`, `OESystem`, `OEChem`, `OEBio`) are now considered "thread-safe". The Thread Safety section in the *OEChem* theory manual goes into more depth of what it means to be "thread-safe". Many minor changes were made to achieve this, only the major ones are listed in the release notes. No guarantee is made about any other library.
- *OEChem* objects no longer rely on finalizers to clean themselves up (*i.e.*, free memory). A separate thread is spawned to specifically handle *OEChem* object deletion. This reduces the risk of creating objects faster than the Java garbage collector can free them, usually leading to the process either running very slow, or crashing due to low memory.
- *OEChem* will allocate all memory from the Java heap. This makes *OEChem* adhere to any memory restrictions placed on the JVM. It also allows for more reliable `OutOfMemory` exceptions to be thrown back to Java code in the event the JVM heap runs out of memory.
- In previous versions, many functions had arguments of the type SWIGTYPE_p_float or SWIGTYPE_p_unsigned_int. These were opaque C++ pointers that could be created from the OE*Array family of classes. To simplify use, many functions that formerly took a SWIGTYPE_p_* now just take the corresponding OE*Array class instance directly. In previous versions, the user had to directly call the `cast()` method, but that is done inside the wrapper now.
- Many example programs have been renamed and cleaned up to provide better support across all three languages.
- Documentation example code is given a descriptive name instead of an out of date chapter and section number.