innovative science • intuitive software

What's New in the Cresset Desktop Applications: Flare™, Forge™, Spark™

Giovanna Tedesco, Senior Product Manager

Overview of Cresset desktop applications

Ligand Based

- SAR analysis
  - Qualitative
  - Quantitative
- Bioactive conformation hypothesis
  - Beyond simple pharmacophores
- Interactive 3D molecule design
  - Informed, effective design

forge

- Scaffold hopping
- R-group design

spark

Structure Based

- Ligand design focus
  - Protein-ligand docking
  - Protein-ligand minimization
- Protein electrostatics
- Ligand electrostatics
- Electrostatic complementarity
- Water position and stability analysis
- Ligand binding energetics
- Customizable interface

flare

- What's the future for desktop applications?

Modern ligand-based workbench for SAR analysis and molecule design

Ligand-based workbench for SAR analysis and design

- Uses ligand alignment as a basis for molecule design and SAR analysis
- Ligands aligned and scored to a reference or ‘template’ using their electrostatic and shape properties
  - Electrostatics and shape
  - Substructure
  - Shape only
- Decipher SAR and communicate the results
  - Activity Atlas and Activity Miner
- Create predictive 3D-QSAR models
  - Field QSAR and Machine Learning models for regression and classification
- Build detailed hypothesis of bioactive conformation
  - Work in 3D on targets with no structural information
- Design new molecules in 3D
  - Property profile not just activity
- Virtually screen thousands of compounds on your desktop
  - Electrostatic/shape similarity or QSAR model predictions

Forge 10.6: Focus on models

- Released January 2019

- Focus on QSAR models
  - New Machine Learning methods
    - Regression
    - Classification
  - New Activity Cliffs summary method in Activity Atlas
  - Enhanced model support and viewing

- Lots of new or improved features
  - Improved Molecule Editor
  - Improved handling of large projects (1000+ ligands)
  - Auto-assign to roles using Murcko scaffolds
  - Enhanced filters (main GUI and Activity Miner)
  - Improved GUI (2D structures, Column management, z-clipping of protein only, improved 3D graphics….)

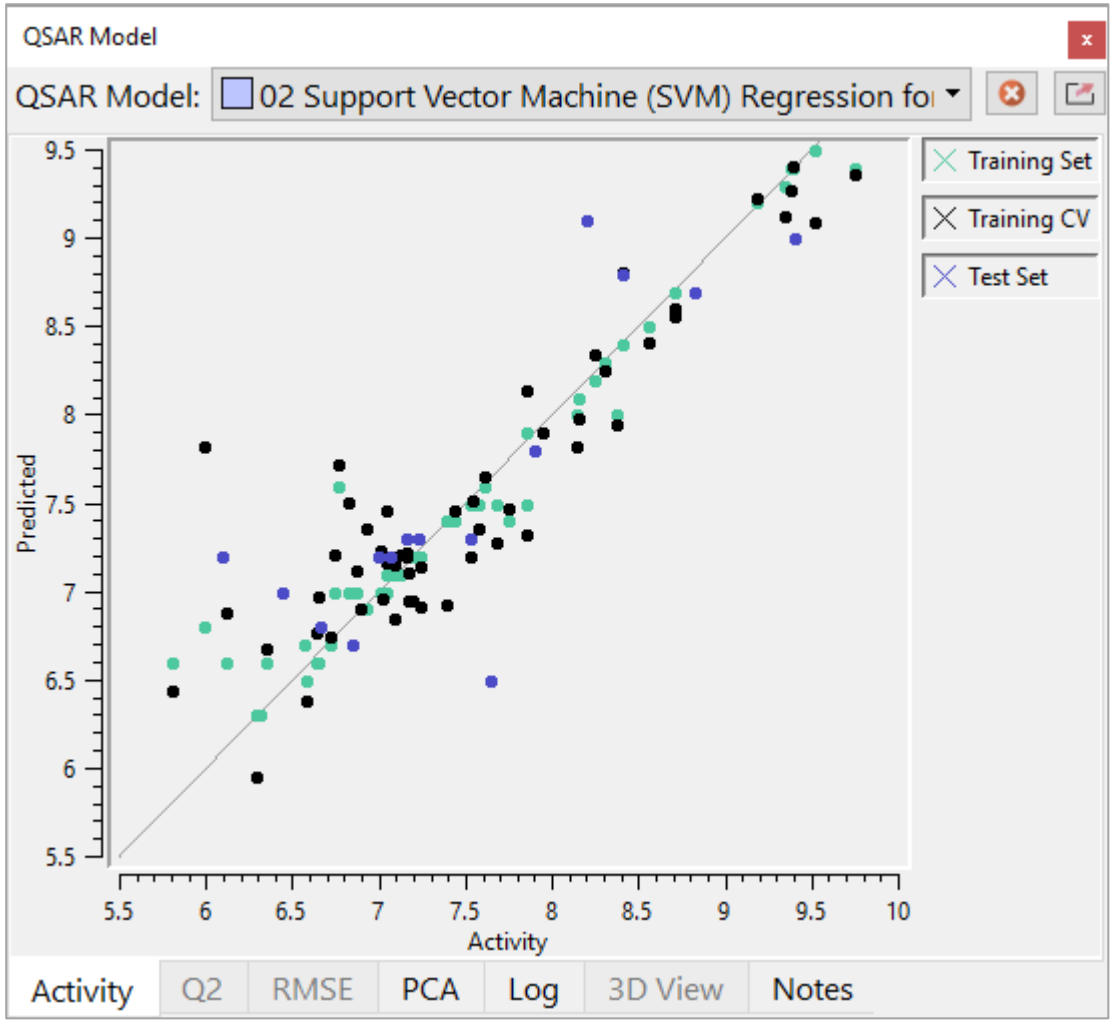
New Machine Learning models

ML regression models

- Predict activity for new compounds
  - Inform design of new ligands
  - Screen small libraries of compounds

- Support Vector Machine
- Relevance Vector Machine
- Random Forest
- k-Nearest Neighbors

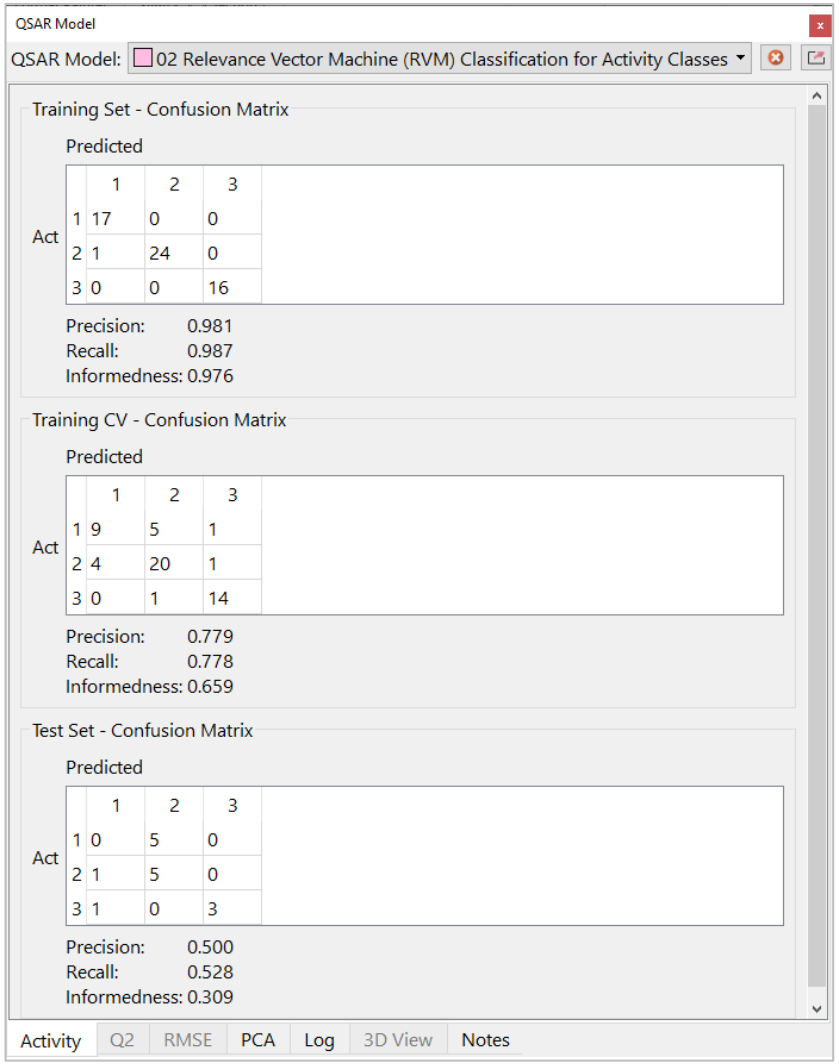
- Use 3D descriptors
  - Electrostatic and volume



New Machine Learning models

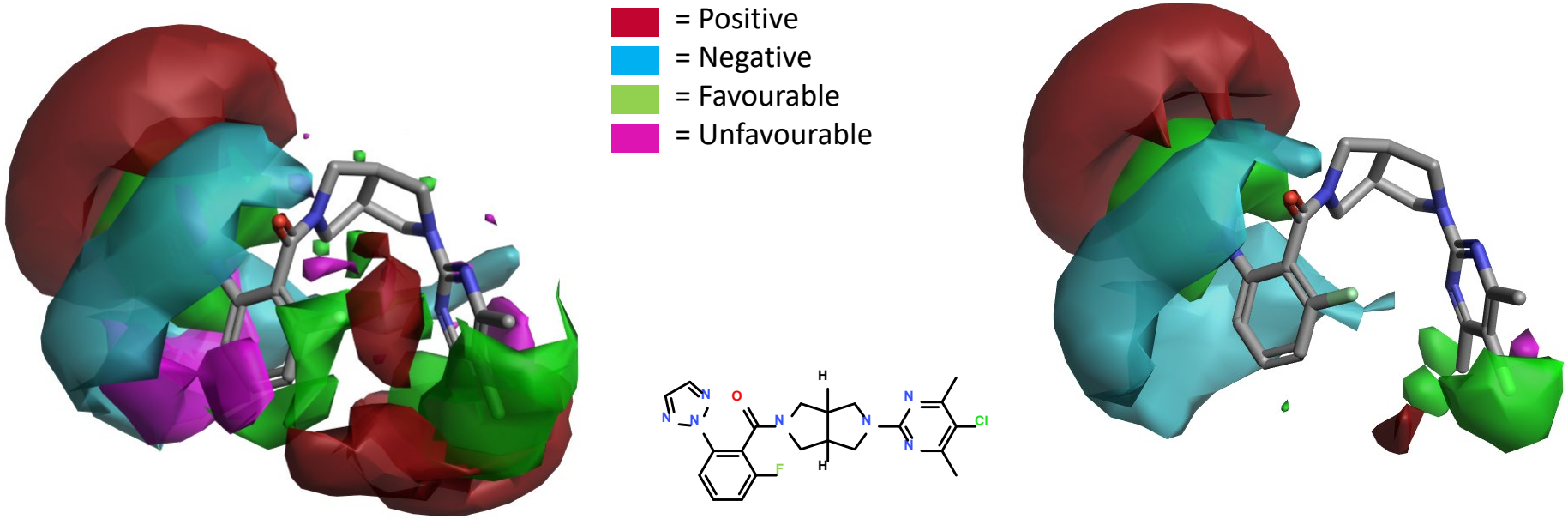
ML classification models

- Activity data is categorical
  - 'high', 'medium', ‘low’
  - activity data which can be converted into categories (e.g, % of inhibition data)
- Predict category for new compounds
  - Inform design of new ligands
  - Screen small libraries of compounds
- Support Vector Machine
- Relevance Vector Machine
- k-Nearest Neighbors
- Use 3D descriptors
  - Electrostatic and volume



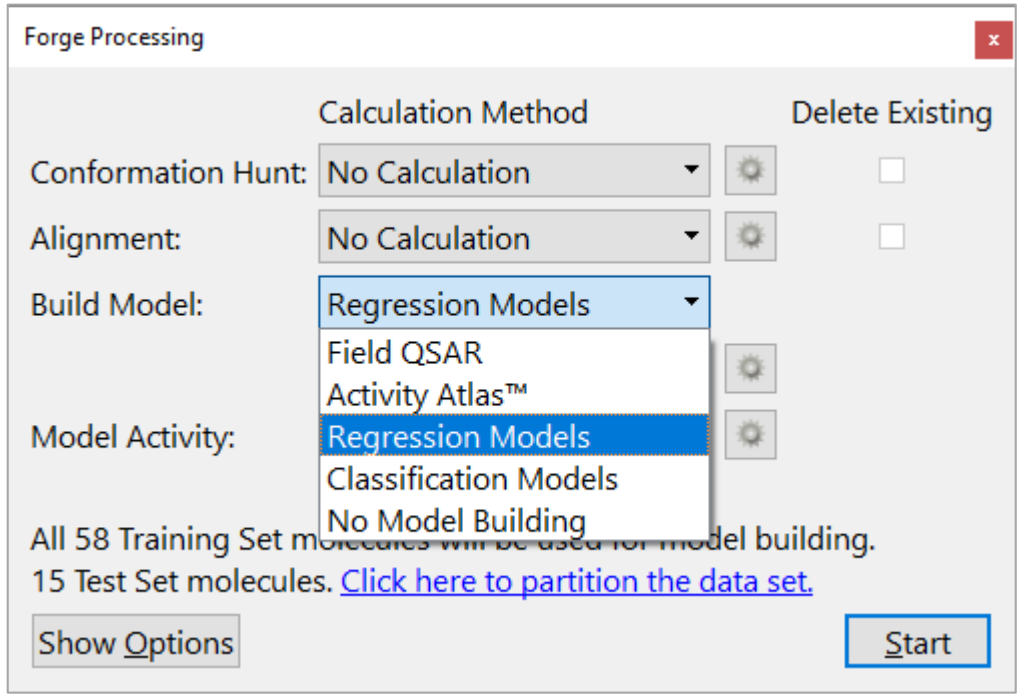
New summary of activity cliffs in Activity Atlas™

- More detailed SAR maps
  - Gives less importance to the strongest activity cliffs
  - Use for small to medium size data sets
- Original method still available
  - Focus on prevalent SAR signals
  - Use for larger data sets
    - e.g. patent data



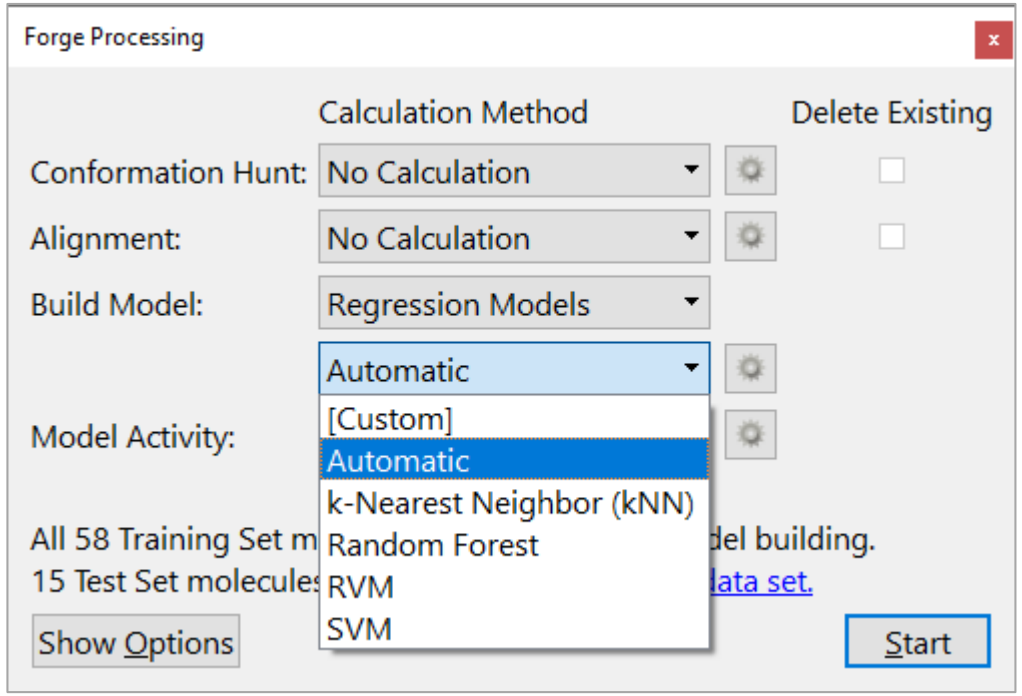
Improved GUI for model calculation

Choose the type of model



> Field QSAR, Activity Atlas, regression or classification?

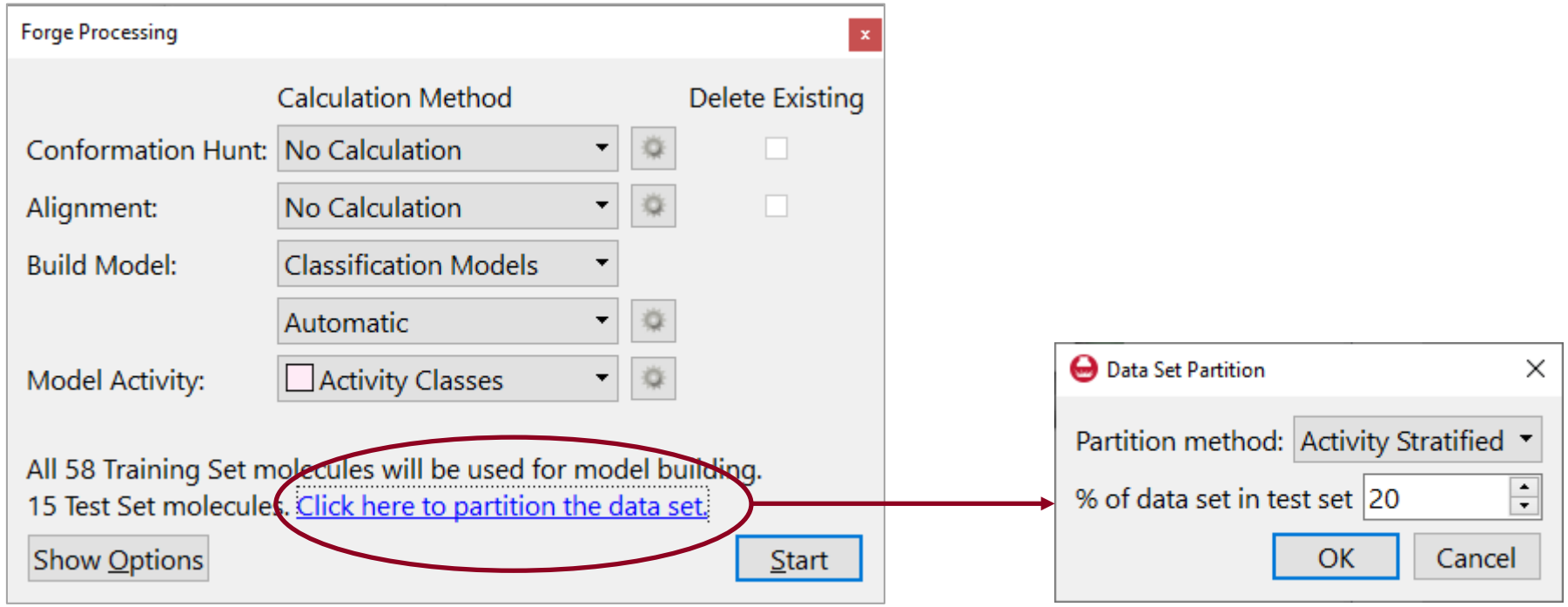
Choose the desired algorithm



> Automatic or choose one?

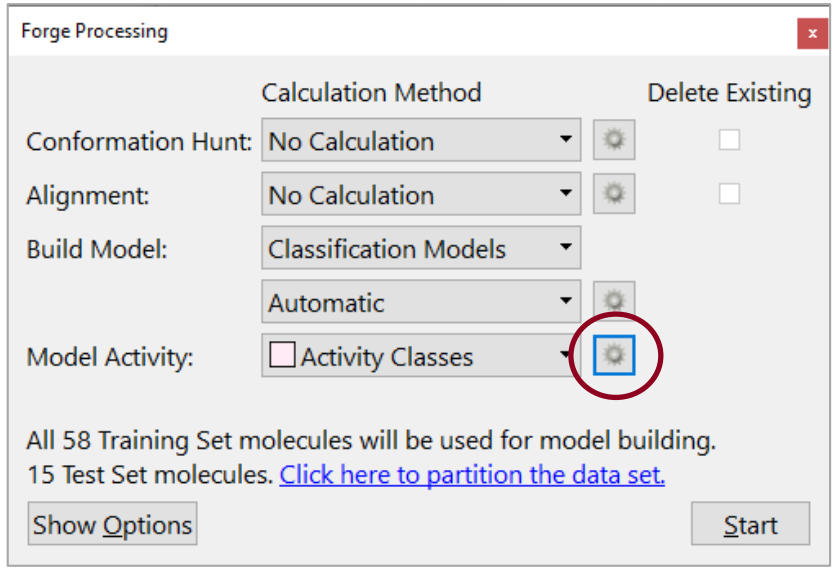
Improved GUI for model calculation

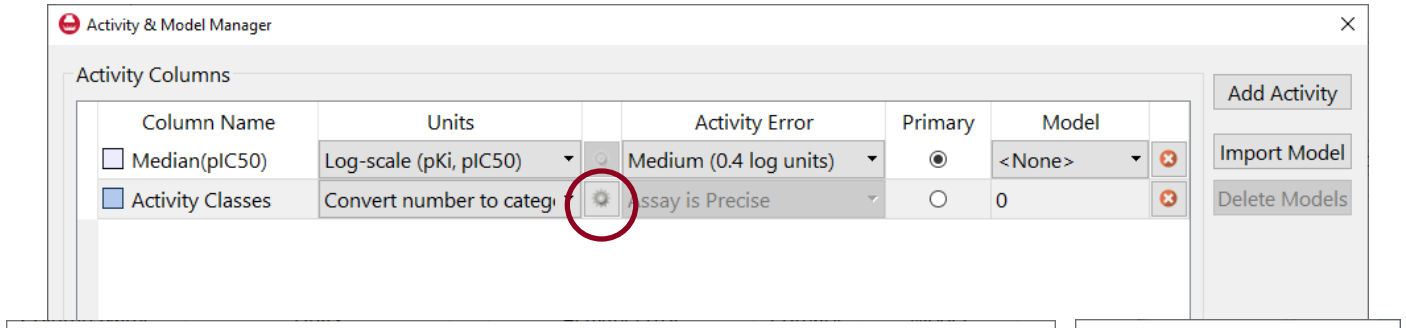
Partition the data set from the processing widget

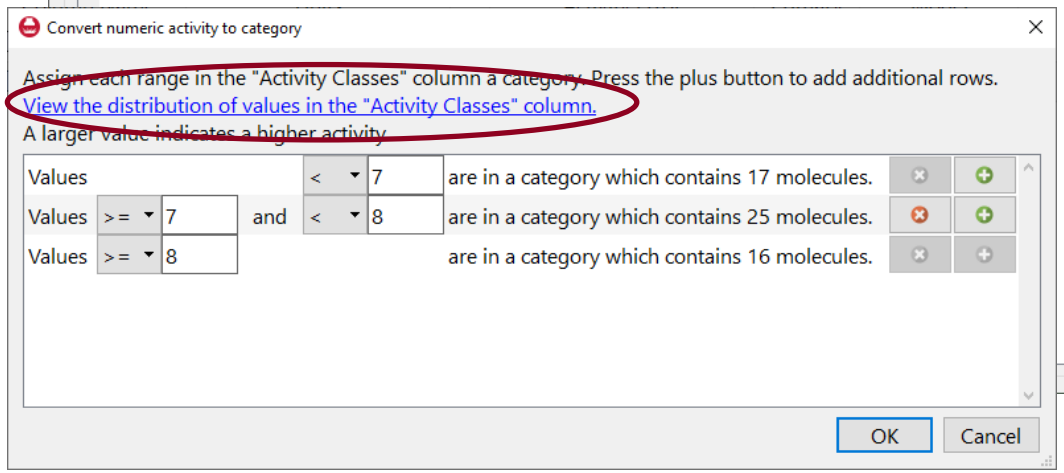


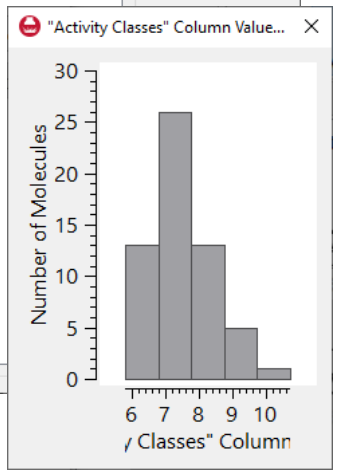
Improved GUI for model calculation

Manage activity data



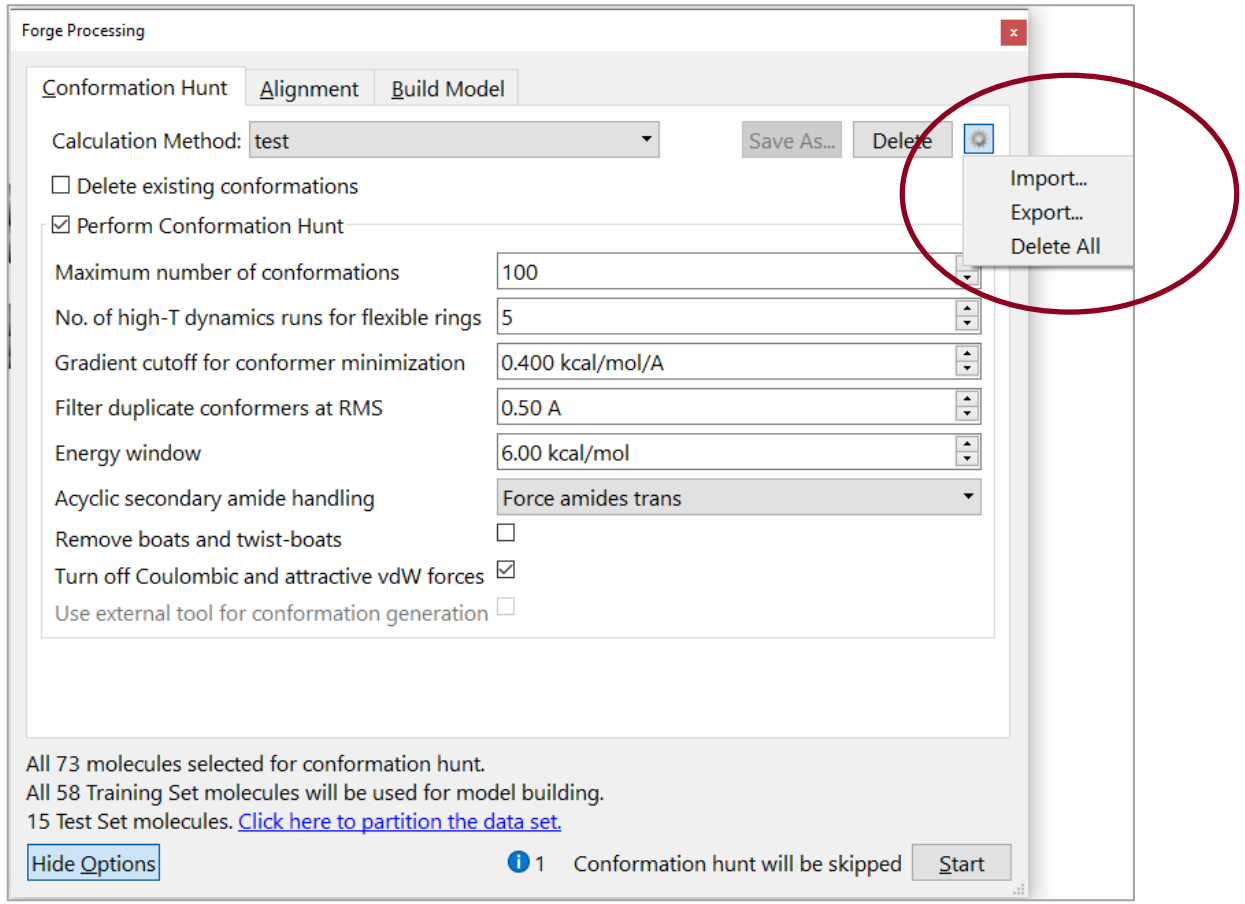




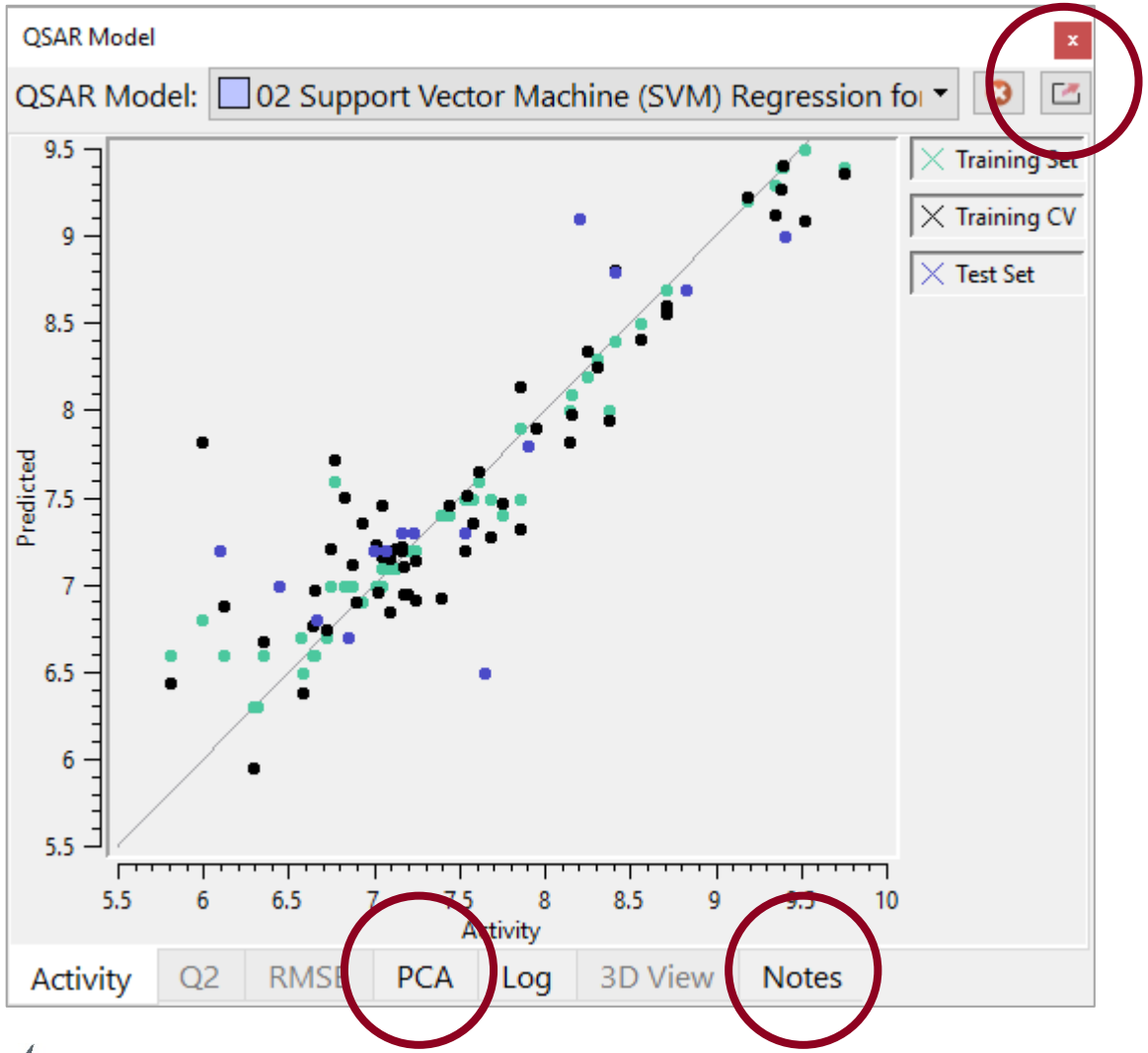


Improved GUI for model calculation

Import/Export custom settings

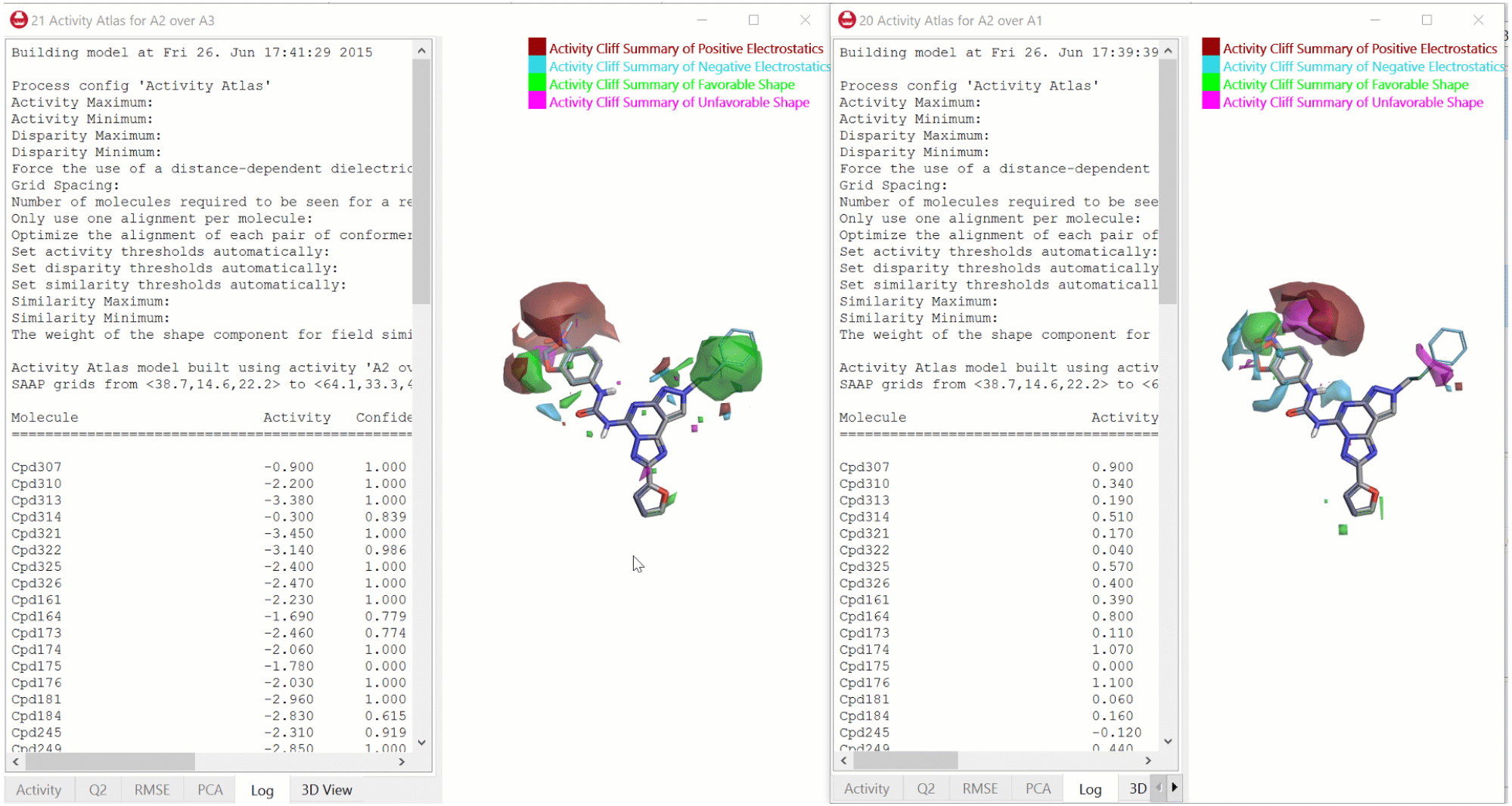


Improved GUI for displaying model results



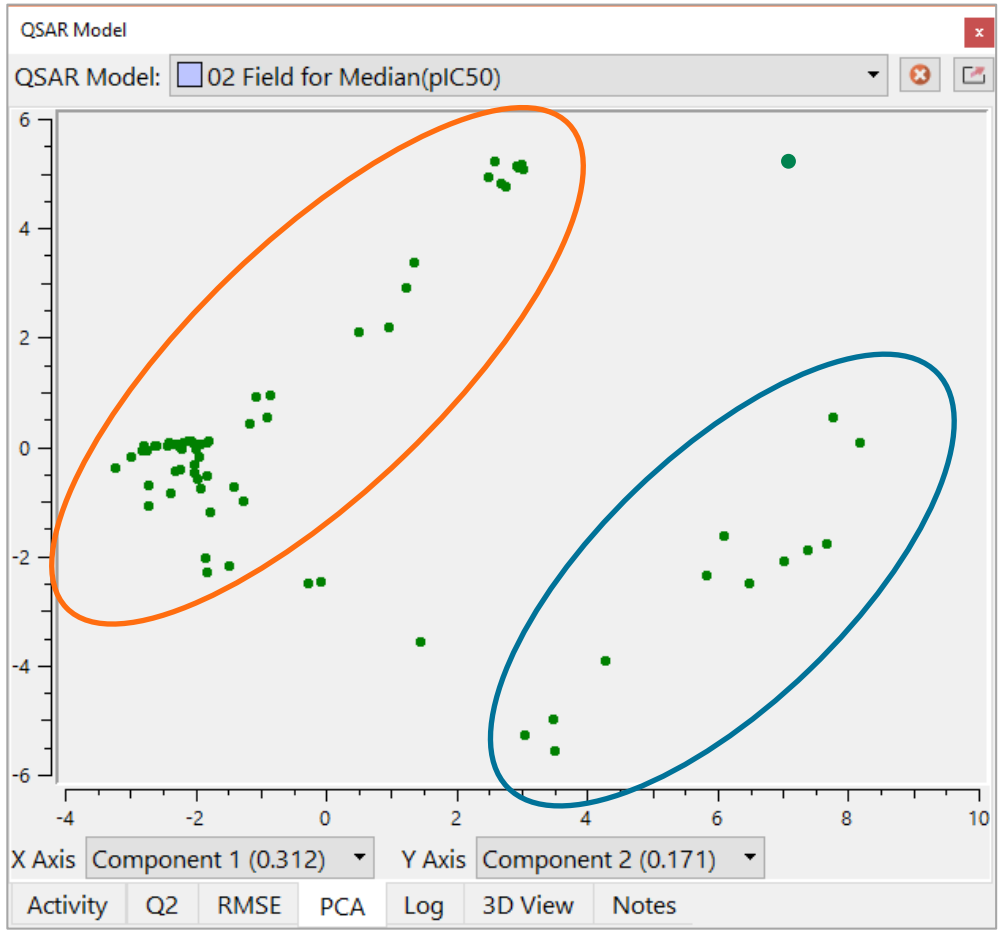
- View multiple models side by side
  - Compare in 3D
  - Compare log files
- Create Notes for models
- PCA plots to spot subgroups or outliers in your dataset

View multiple models side by side

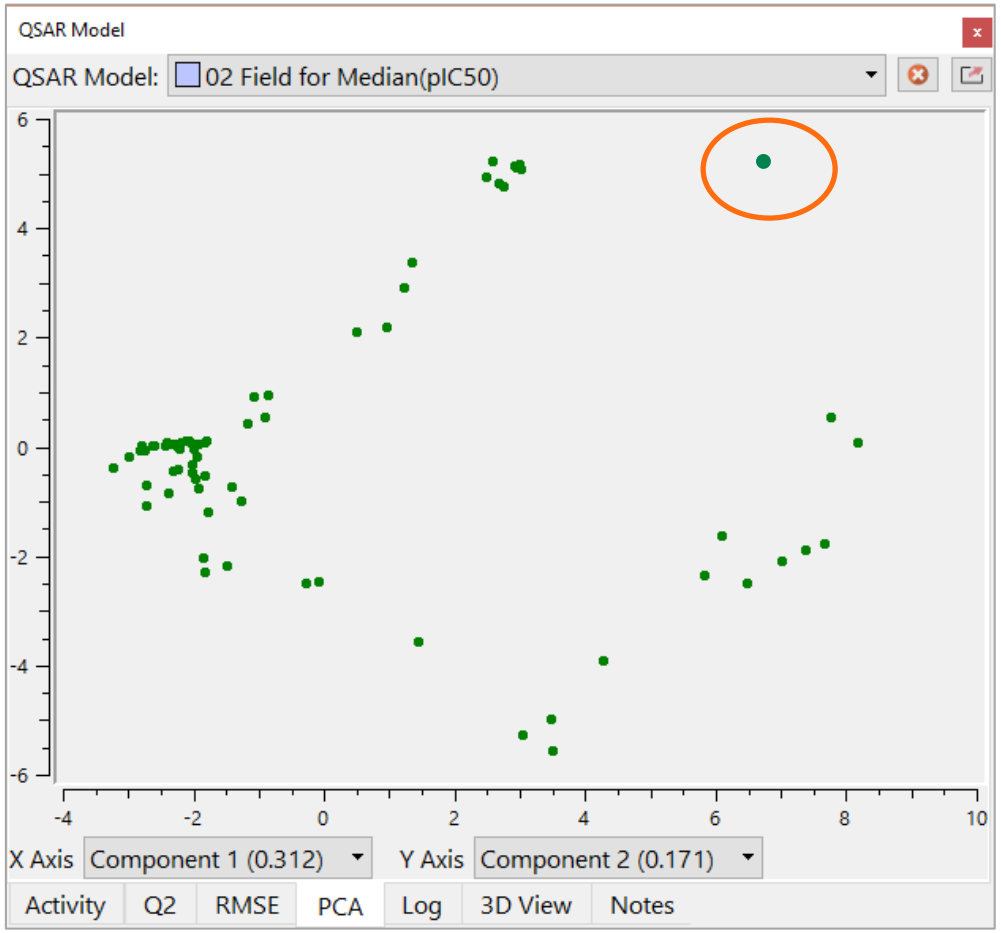


New PCA tab

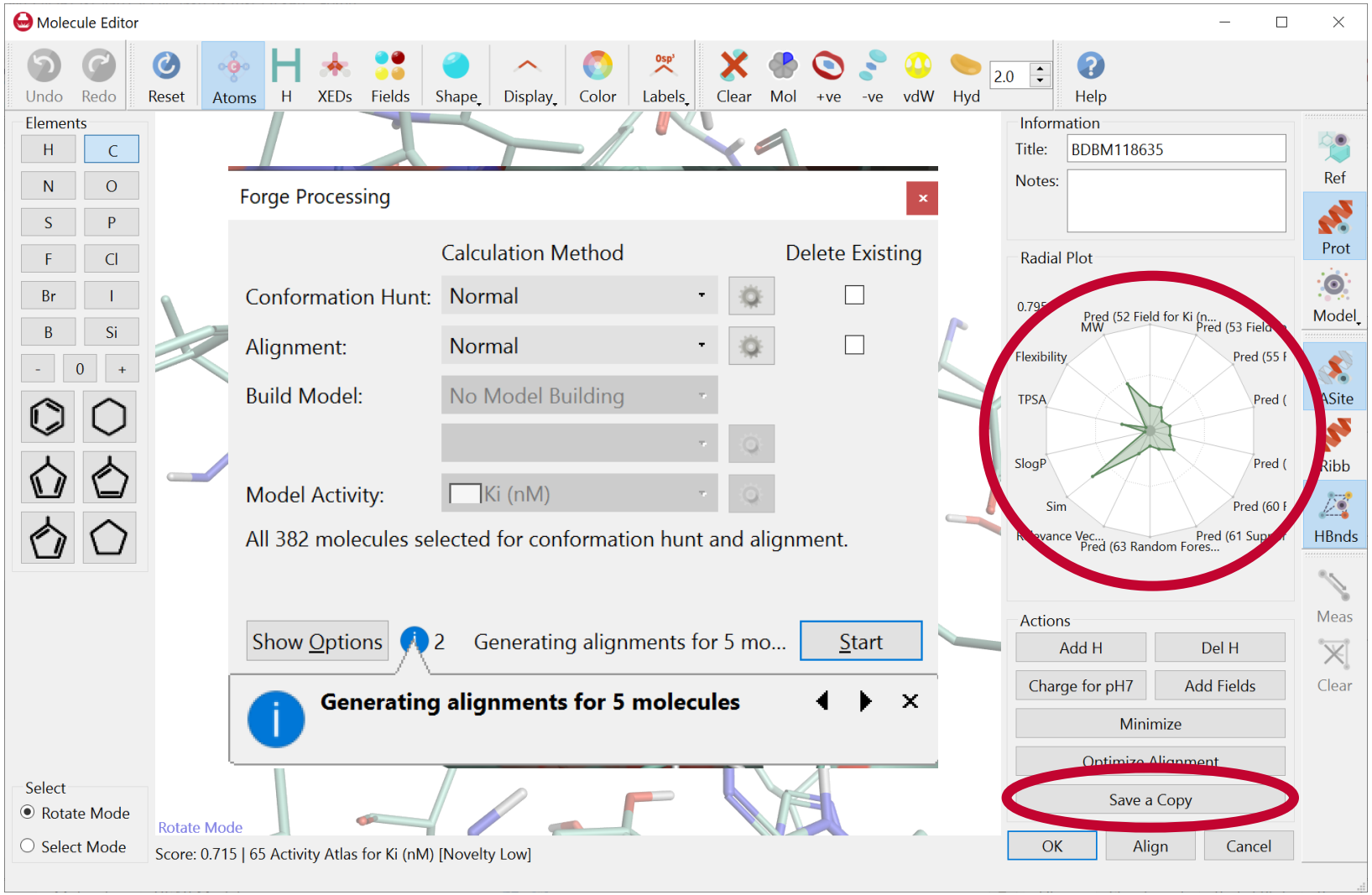
Useful to spot groupings



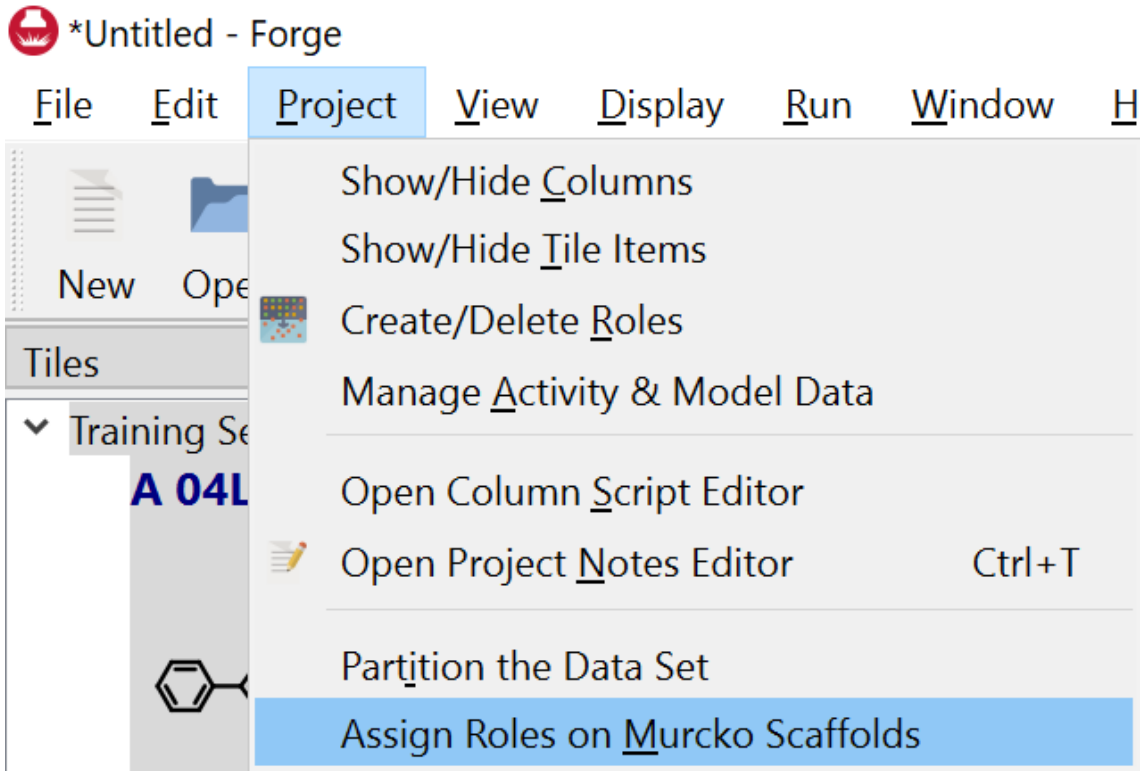
Useful to spot outliers



Improved editor display and behavior



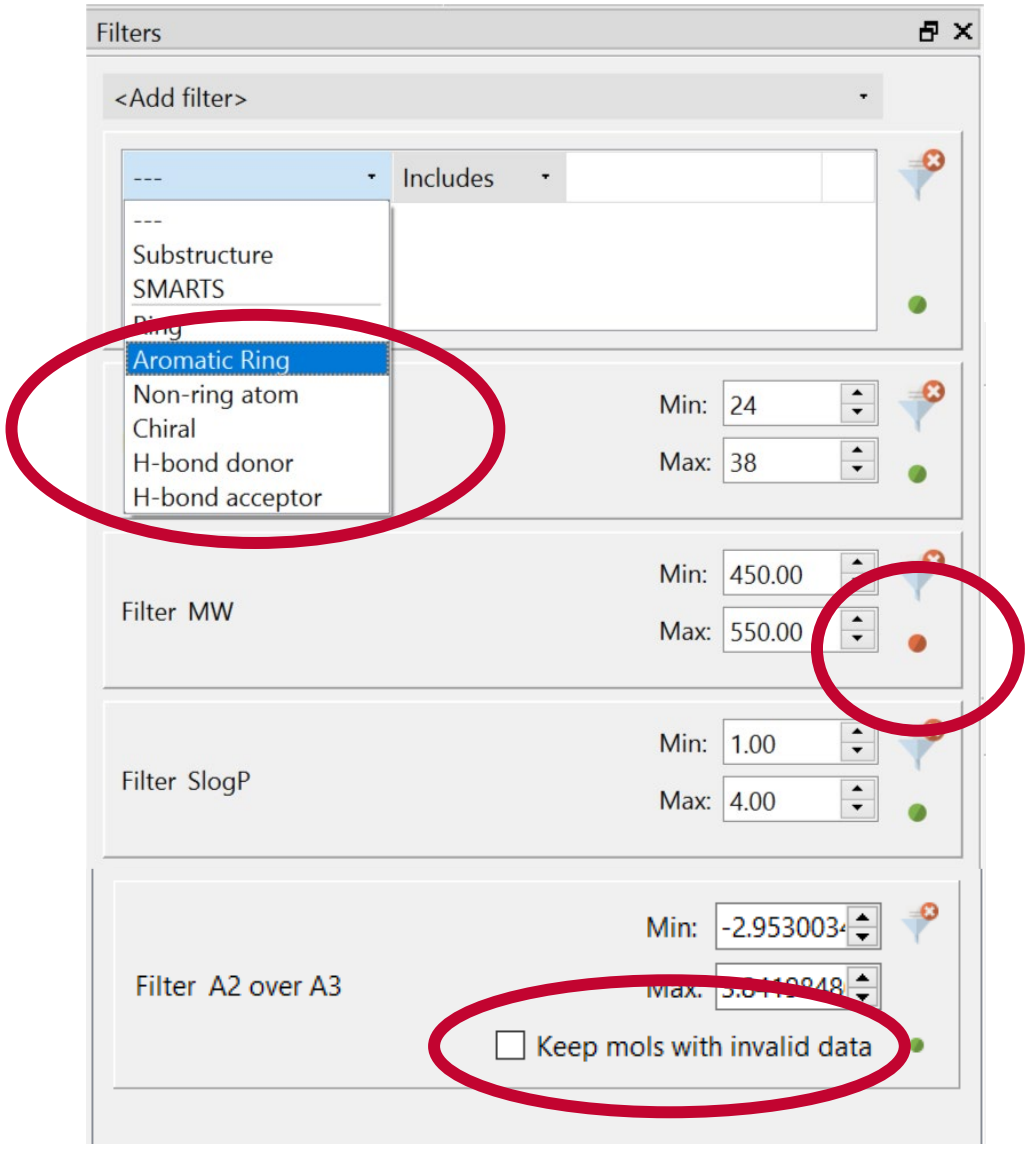
Automatically assign compounds to project roles



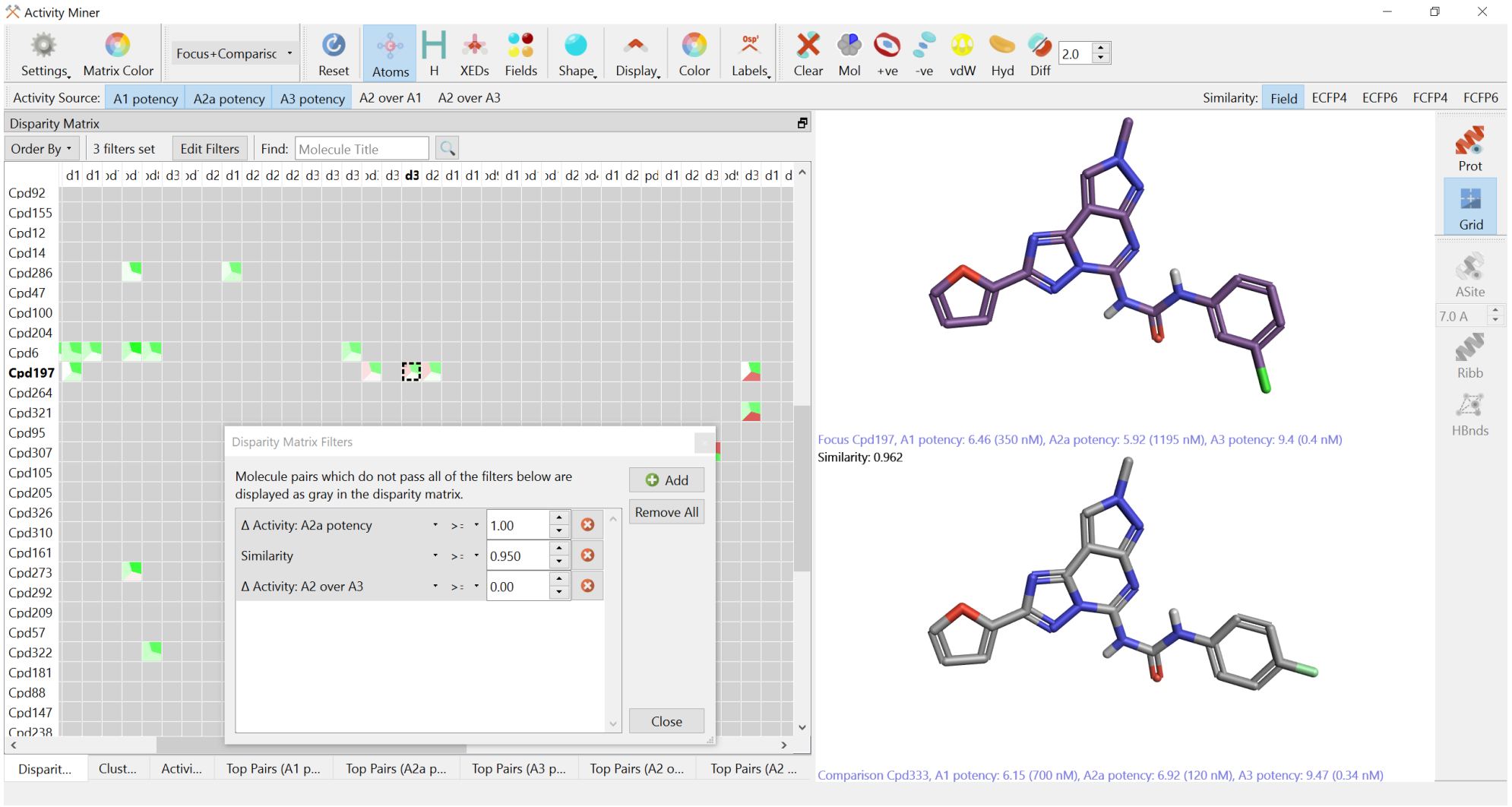
- Murcko Scaffold C=C(C1CC1)C2CCC[C@H](C2)c3cc(c4ccccn43)-c5ccc(cc5)C(Cc6ccccc6)=C (1 mol)
- Murcko Scaffold C=C(C1CCCCC1)c2ccc(CC=3C(CC=C(C3)c4cc(CC(c5ccccc5)=C)ccc4)=C)cc2 (1 mol)
- Murcko Scaffold C=C(C1CCCCC1)c2ccc(Cc3cc(-c4cc(CC(c5ccccc5)=C)ccc4)cc6c3cc[nH]6)cc2 (1 mol)
- Murcko Scaffold C=C(C1CCCCC1)c2ccc(Cc3cc(cn4cccc34)-c5cccc(c5)CC(c6ccccc6)=C)cc2 (2 mols)
- Murcko Scaffold C=C(C1CCCCC1)c2ccc(cc2)CC=3C(CC=C(c4cc(C5C=Cc6ccccc6C5=C)ccc4)C3)=C (1 mol)
- Murcko Scaffold C=C(C1CCCCC1)c2ccc(cc2)CC=3C(CC=C(c4cc(CC(c5ccccc5)=C)ccc4)C3)=C (1 mol)
- Murcko Scaffold C=C(C1CCc2c3ccccc3[nH]c2C1)c4c(CCc5ccc6ccccc6c5)cc[nH]4 (1 mol)
- Murcko Scaffold C=C(Cc1cccc(c1)C(Cc2ccc3c(c2)cc([nH]3)-c4ccccc4)=C)/C=C/c5ccccc5 (1 mol)
- Murcko Scaffold C=C(c1cc2ccccc2[nH]1)Cc3cccc(C4=CCC(C(Cc5cn6c(CCCC6)c5)=C4)=C)c3 (1 mol)
- Murcko Scaffold C=C(c1ccc([nH]1)Cc2cccc(C3CCCCC3)c2)Cc4ccccc4 (2 mols)
- Murcko Scaffold C=C1C(c2cc(ccc2)C3=CCC(C(=C3)Cc4ccc(C5CCCCC5)cc4)=C)C=Cc6c1ccc(C7CC7)c6 (2 mols)
- Murcko Scaffold C=C1C(c2cc(ccc2)C3=CCC(C=C3)=C)C=Cc4ccccc41 (1 mol)
- Murcko Scaffold C=C1C(c2ccc(cc2)Cc3ccccc3)=C4C=CCC=C4C1[C@@H]5CCCC5 (1 mol)
- Murcko Scaffold C=C1C(c2cccc(-c3cccc4c3c5CCCCc5[nH]4)c2)C(Cc6ccccc61)=C (1 mol)
- Murcko Scaffold C=C1C(c2cccc(-c3cccc4c3c5ccccc5[nH]4)c2)C=Cc6ccccc16 (1 mol)
- Murcko Scaffold C=C1C(c2cccc(C3=CCC(C(=C3)Cc4ccccc4)=C)c2)C=Cc5ccccc51 (1 mol)
- Murcko Scaffold C=C1CC/C(c2ccc(C3CCCCC3)cc12)C\c4c[nH]cc4 (1 mol)
- Murcko Scaffold C=C1CC=C(c2cc(C3C=Cc4ccccc4C3=C)ccc2)C=C1Cc5ccc(C6CCCCC6)cc5 (1 mol)
- Murcko Scaffold C=C1CCCC[C@@H]1c2ccc(cc2)CC=3C(CC=C(c4cc(CC(c5cc6c([nH]5)CCCC6)=C)ccc4)C3)=C (2 mols)
- Murcko Scaffold C=C1C[C@@H]2C=C(Cc3ccccc3)C=CC2=C[C@H]1c4ccccc4 (1 mol)
- Murcko Scaffold C=C1Cc2cc(Cc3ccccc3)ccc2C=C1Cc4ccccc4 (1 mol)
- Murcko Scaffold C=C1c2ccccc2CCC1c3cc(ccc3)C4=CCC(C(=C4)Cc5ccc(C(C6CCCCC6)=C)cc5)=C (1 mol)
- Murcko Scaffold [nH]1ccc2ccc(-c3cccc4ccccc43)cc21 (1 mol)
- Murcko Scaffold c1(Cc2cccc(Cc3ccc(cc3)Cc4ccccc4)c2)ccccc1 (1 mol)
- Murcko Scaffold c1(Cc2cccc(Cc3ccccc3)c2)ccccc1 (1 mol)
- Murcko Scaffold c1(cccc2ccccc21)-c3ccccc3 (1 mol)
- Murcko Scaffold c12ccccc2cccc1 (1 mol)
- Murcko Scaffold n1(C2CCCC2)cc(c3ccccc31)-c4ccc(cc4)Cc5ccccc5 (1 mol)
- Murcko Scaffold n1([C@H]2CCCCC2)cc(-c3ccc(cc3)Cc4ccccc4)c5ccccc51 (2 mols)

Improved molecule filters

- Prepared SMARTS patterns for
  - Aromatic ring atom
  - Ring / Not ring
  - H-bond donor/acceptor
  - Chiral
- Option to enable/disable a specific filter
- Improved handling of molecules with invalid data in filters and radial plots



Enhanced filtering in Activity Miner™



Introducing Forge Design

> Goodbye Torch, welcome Forge Design

> New licensing level of Forge that fully reproduces Torch functionality

> Existing Torch customers get additional functionality
  > Enhanced Activity Miner
  > Field differences

Future plans for Forge

> … more at the end

Fresh insights into structure-based design

Fresh insights into structure-based design

> Cresset workbench for structure-based design

> Analyse proteins and inform new molecule design

> New scientific capabilities extend the range of SBDD methods available

> Robust enabling capabilities to support new workflows

> Modern, extensible user interface that is easy to use and easy to extend

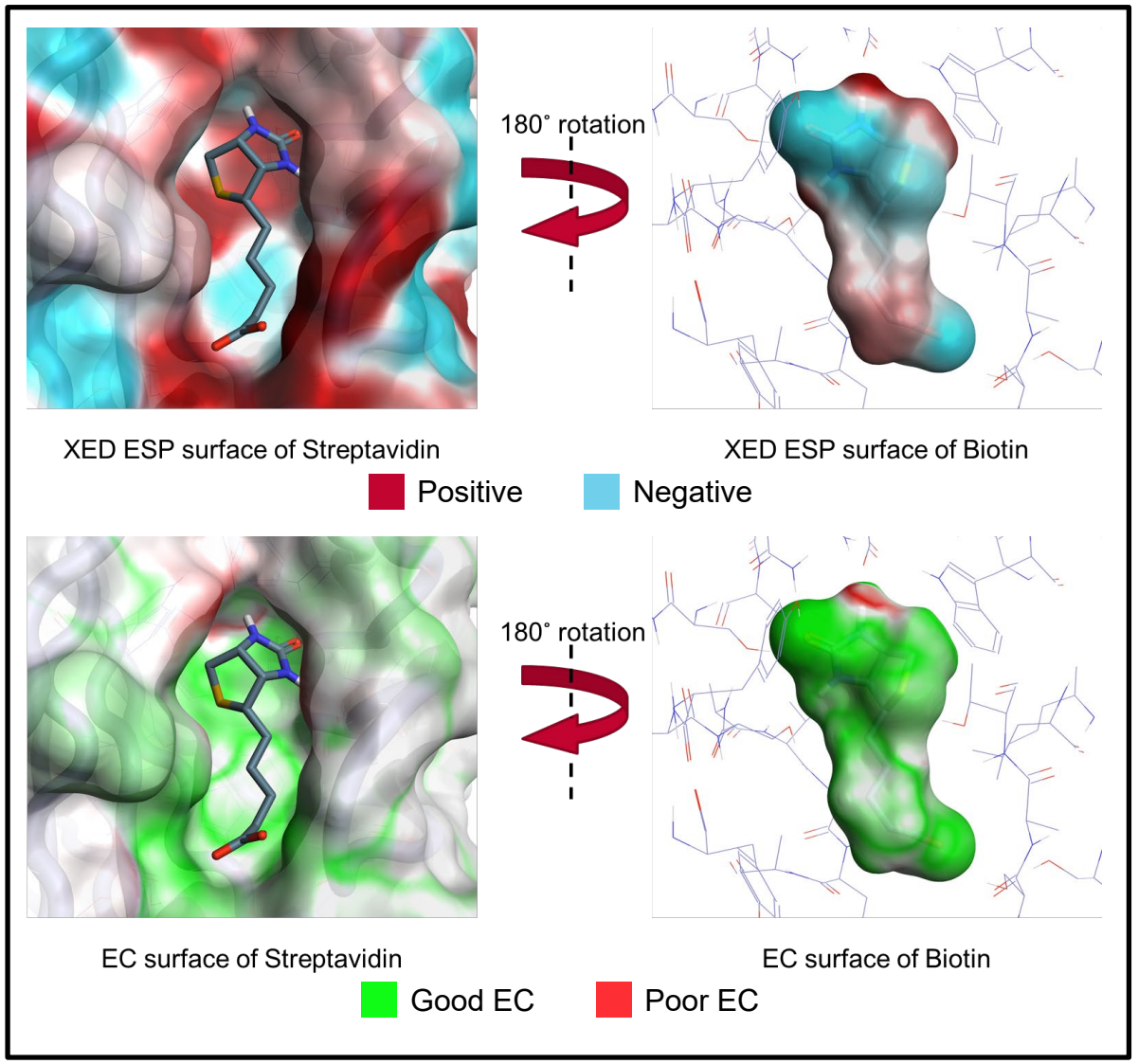
> Customisable, scriptable, deployable science through Flare API and python command line

Flare V2.0: Focus on electrostatics and ligand workflows

- Released June 2018
- Electrostatic Complementarity
  - Scoring
  - Maps
- Ensemble docking
  - Handling active site flexibility
- Improvements on ligand workflows
  - Radial plots/Multi-parametric scoring
  - Filtering using properties, tags or sub-structure
  - Storyboard
- Flare Python® API
  - Most functions scriptable using Python
  - Command-line access through pyflare

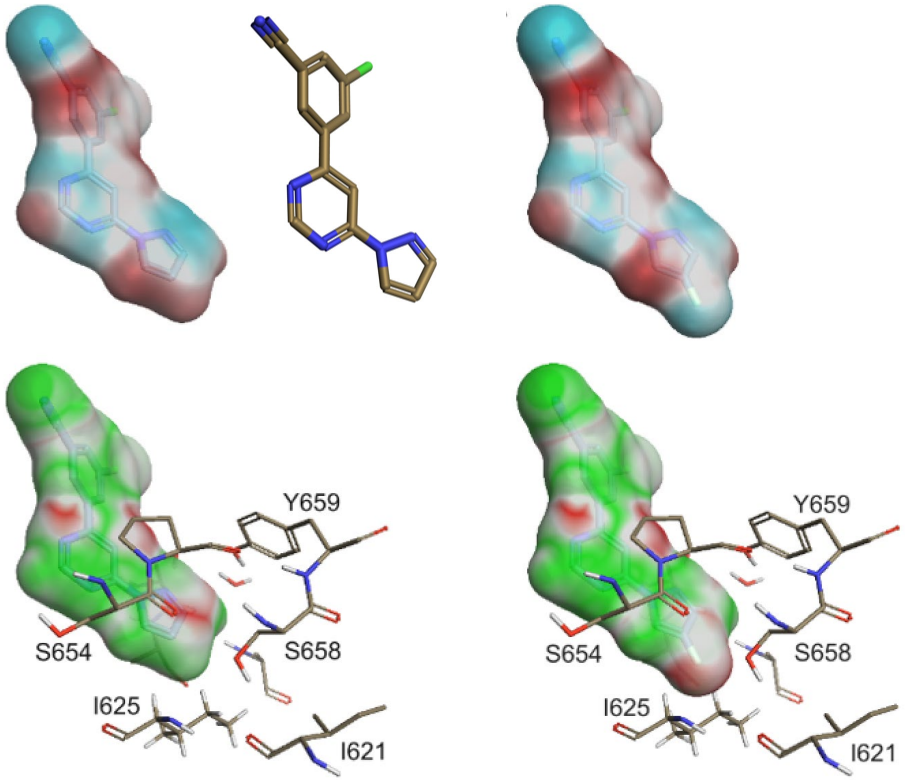
Electrostatic Complementarity™ as a guide for molecule design

- Electrostatic interactions between ligands and proteins are an important factor in recognition and binding energetics
- Assessing Electrostatic Complementarity (EC) provides
  - Insight of why ligand bind
  - Inform molecular design
  - Predict activity
- Dedicated algorithm to calculate and display where electrostatics are complementary



EC maps and scores inform ligand design –mGluR5

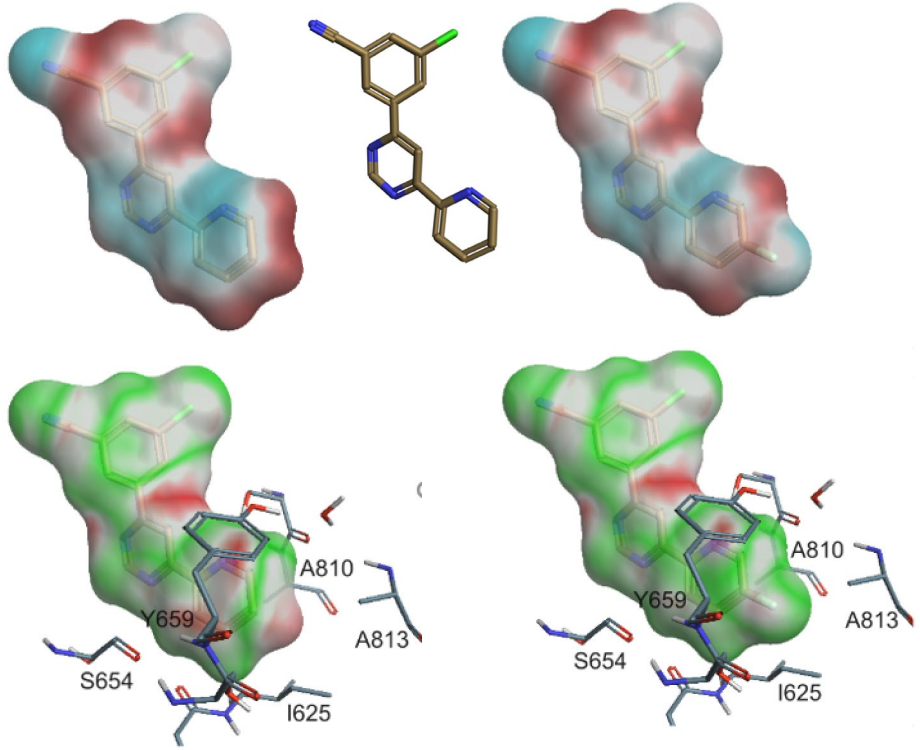
A



Compound 13
pKi = 8.4
SCI = 0.40

Compound 17
7.7
0.38

B



Compound 21
pKi = 8.5
SCI = 0.40

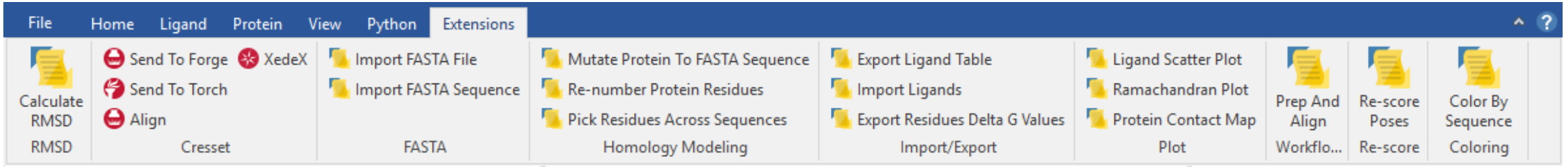
Compound 25
9.3
0.46

Christopher *et al., J. Med. Chem.* 2015, **58**, 6653−6664

Python API for Flare

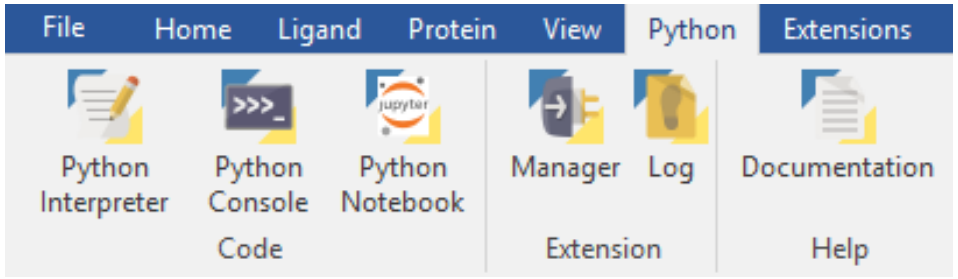
- The vast majority of Flare's functions are now scriptable using Python
  - File I/O
  - Docking
  - GUI
  - WaterSwap
  - 3D-RISM
  - Protein prep, alignment, superposition
  - etc
- New 'pyflare' executable provides command-line access
- Three collections of Python examples and extensions
- Command line extensions
  - Cresset developed scripts to use Flare from the command line (pyflare)
  - https://gitlab.com/cresset/flare-python-pyflare
- Python Extensions
  - Useful new functionality for all
  - https://gitlab.com/cresset/flare-python-extensions
- Developer examples
  - Templates to demonstrate key principles and functionality
  - https://gitlab.com/cresset/flare-python-developer

Python Extensions

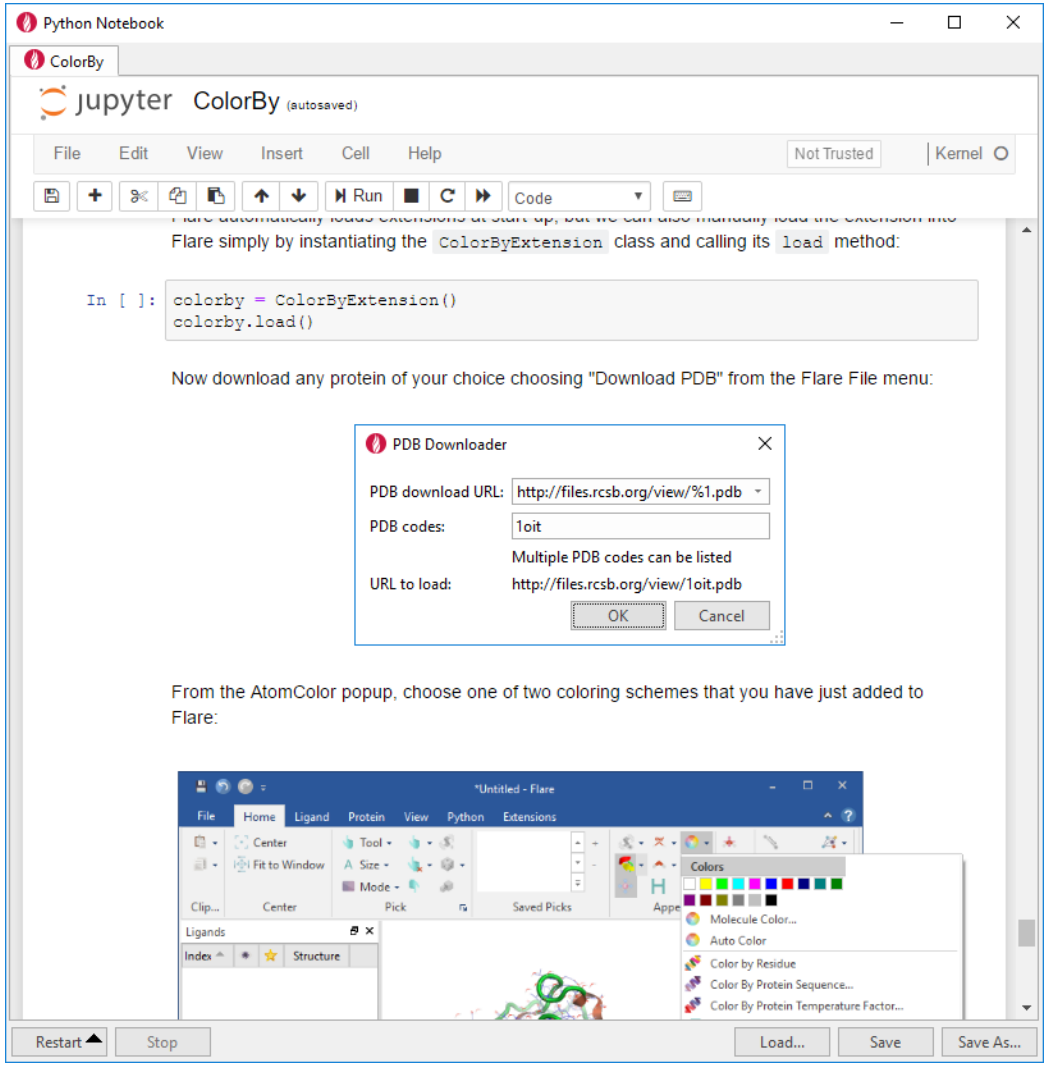


- https://gitlab.com/cresset/flare-python-extensions
  - Cresset supported extensions
  - Significantly expand Flare functionality for free
  - Available for download to all Flare users from a GitLab repository

Python Jupyter notebook



- The ideal environment for Python enthusiasts
- Provides a comfortable Python shell environment:
  - History
  - TAB completion
  - Syntax highlighting
  - Embedding of images, etc.
  - Editing and running individual code cells
  - … and more



Introducing Flare Viewer

Released November 2018

> Free, viewer version of Flare
> Replaces TorchLite

Capabilities

> View protein ligand interactions
> View ligand electrostatics
> Superpose proteins and visually compare
> Edit ligands
> Calculate physico-chemical properties for each ligand
> Radial plots
> Filters
> Storyboard

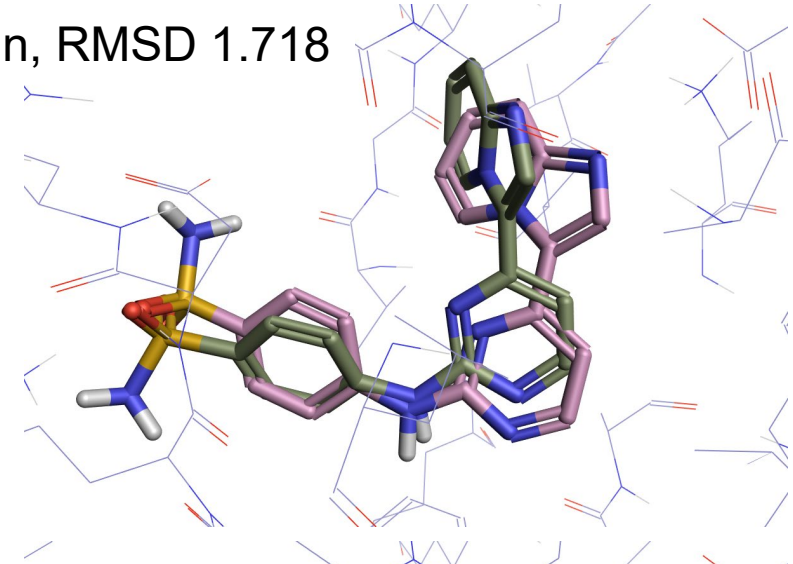
A sneak peek of Flare V3

- Release date: Autumn 2019
- FEP
- Improved docking
  - Improved ‘Accurate’ algorithm
  - Template docking
  - Covalent docking
- Molecular Dynamics
- ‘Forge Design-like’ ligand alignment
- Faster minimization algorithm
- Improved atom typing in protein prep
- Improved handling of larger datasets
- Lots of GUI new features and improvements
  - Edit ligand and proteins the main window
  - Ribbons for DNA
  - Ribbon colors and transparency
  - Improved Contacts panel
  - Secondary structure in the Alignment table
  - Filtering the Alignment table
  - Sequence similarity window
  - Tile view
  - Export filters
  - New Help tab
  - Atom custom labels
  - Synchronize 3D view
  - Send/Retrieve from Blaze
  - …….

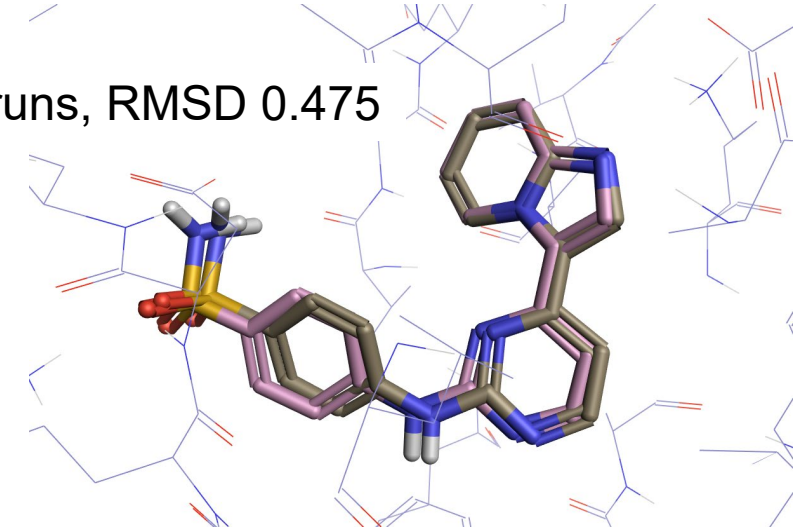
Improved ‘accurate’ docking algorithm

- The docking engine in Flare is based on a Genetic Algorithm
  - Non-deterministic
  - Repeat runs may give different results
- New improved algorithm expands the GA exploration
  - Run repeat dockings
  - Keep the best poses from each run
  - Merge and re-score the separate pools of poses from each run
  - Keeps the best poses overall
- ‘Accurate’ mode slightly slower but gives improved results

Single run, RMSD 1.718



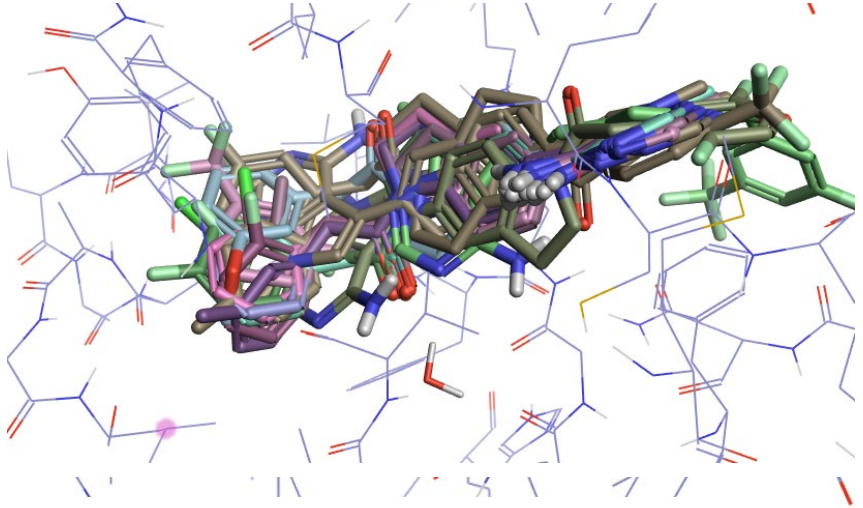
Repeat runs, RMSD 0.475



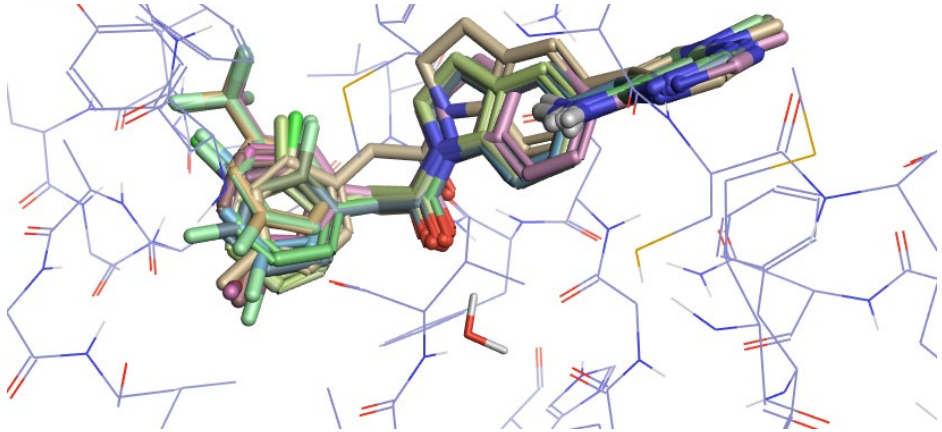
Template docking

- Choice of 3D starting conformation may have strong influence on docking results
- Template docking aligns by substructure the molecules to be docked to a template molecule
- Uses the aligned conformation to ‘seed’ the docking run
- Useful to dock congeneric ligands to a known template pose

Normal docking, 1 run, 2D input conformations

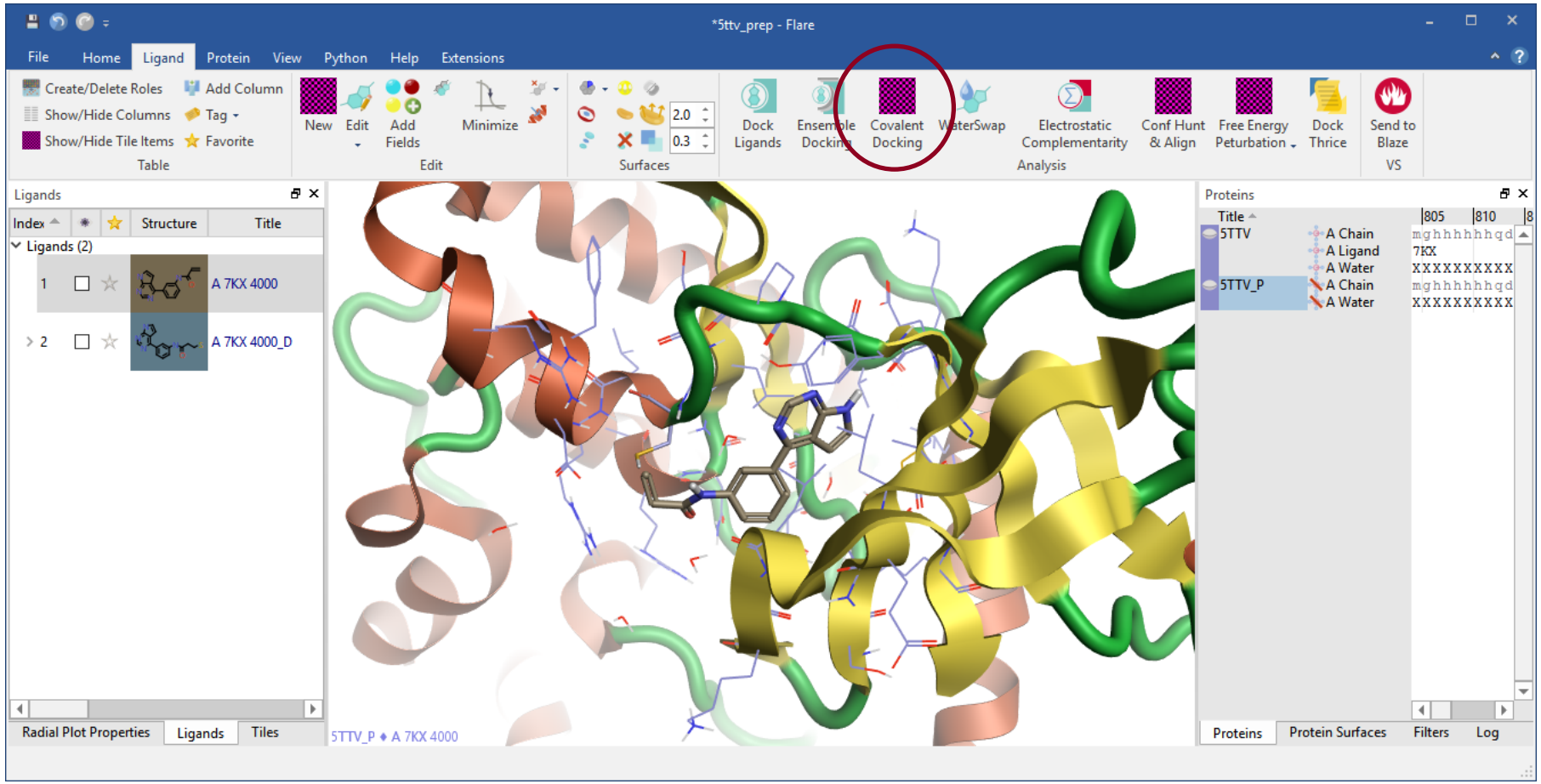


Template docking, 2D input conformations



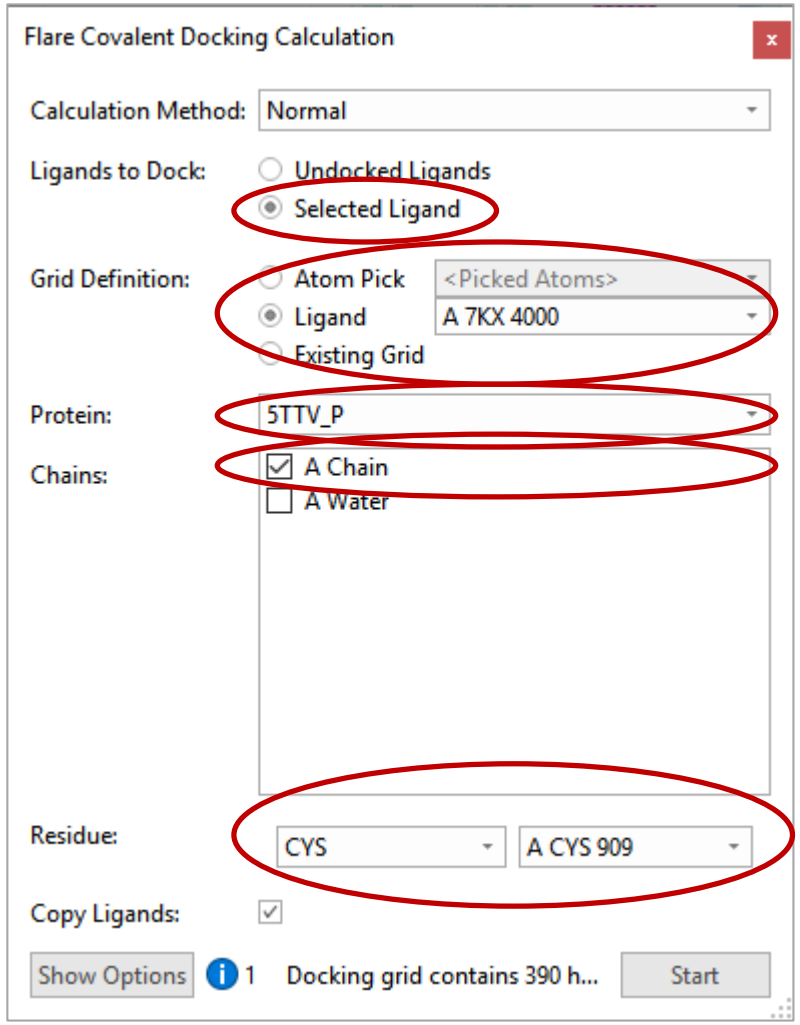
Covalent docking

> New button in the Ligands tab

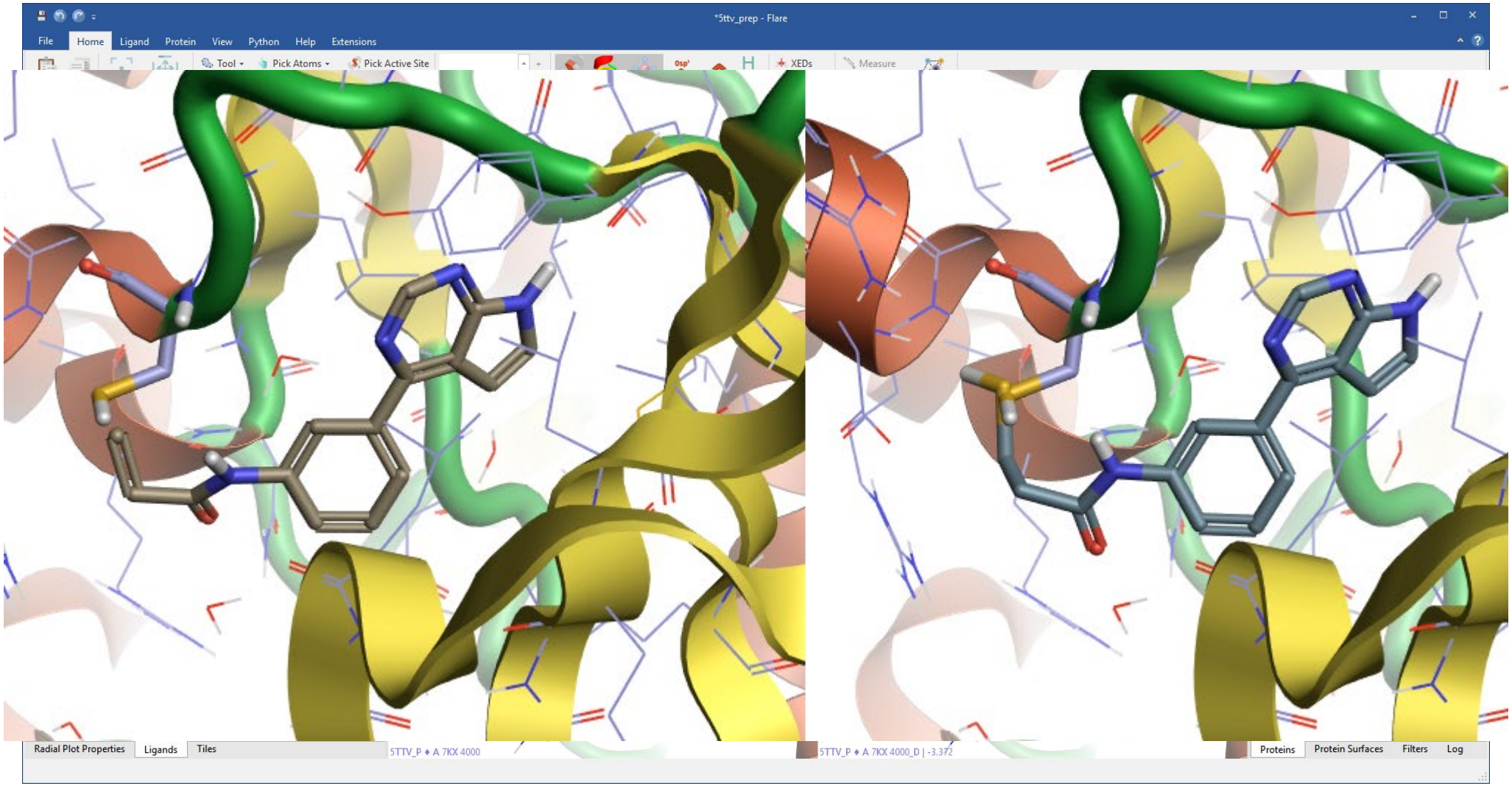


Covalent docking

- New button in the Ligands tab
- Choose ligands to dock
  - Must have covalent warhead
- Choose protein to dock into
  - Choose chains to include
  - Define active site as per standard docking
- Choose covalent docking residue
  - Use pull down menus

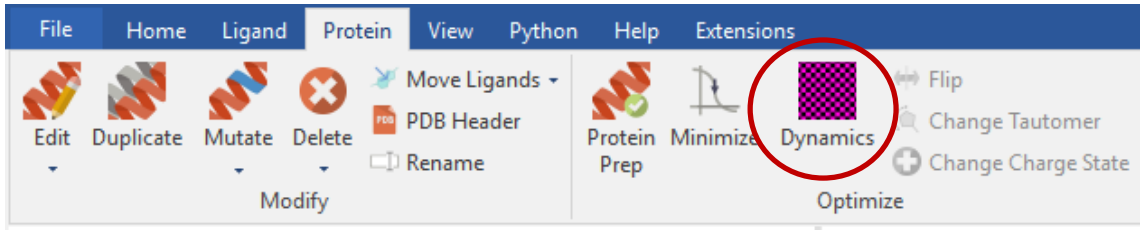


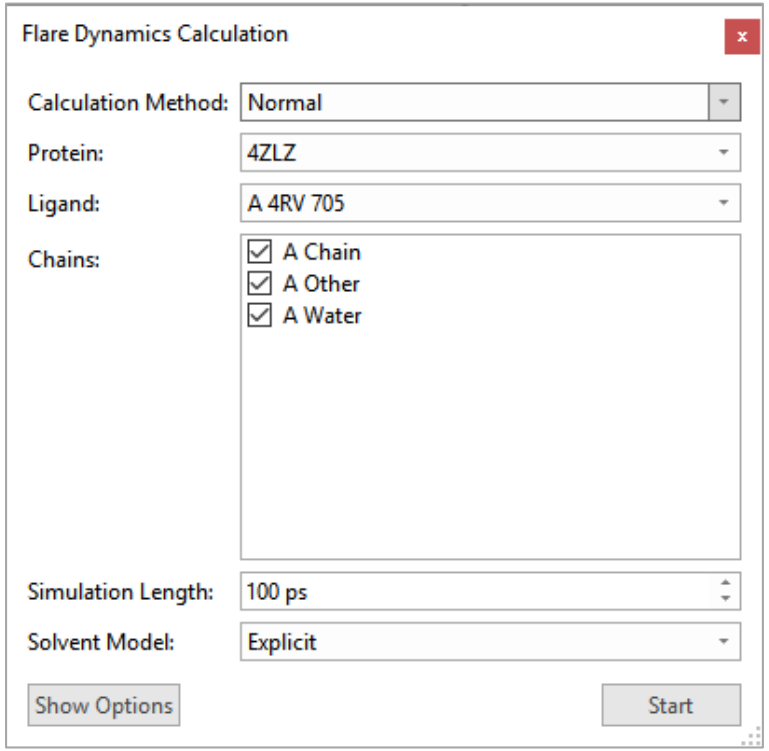
Covalent docking



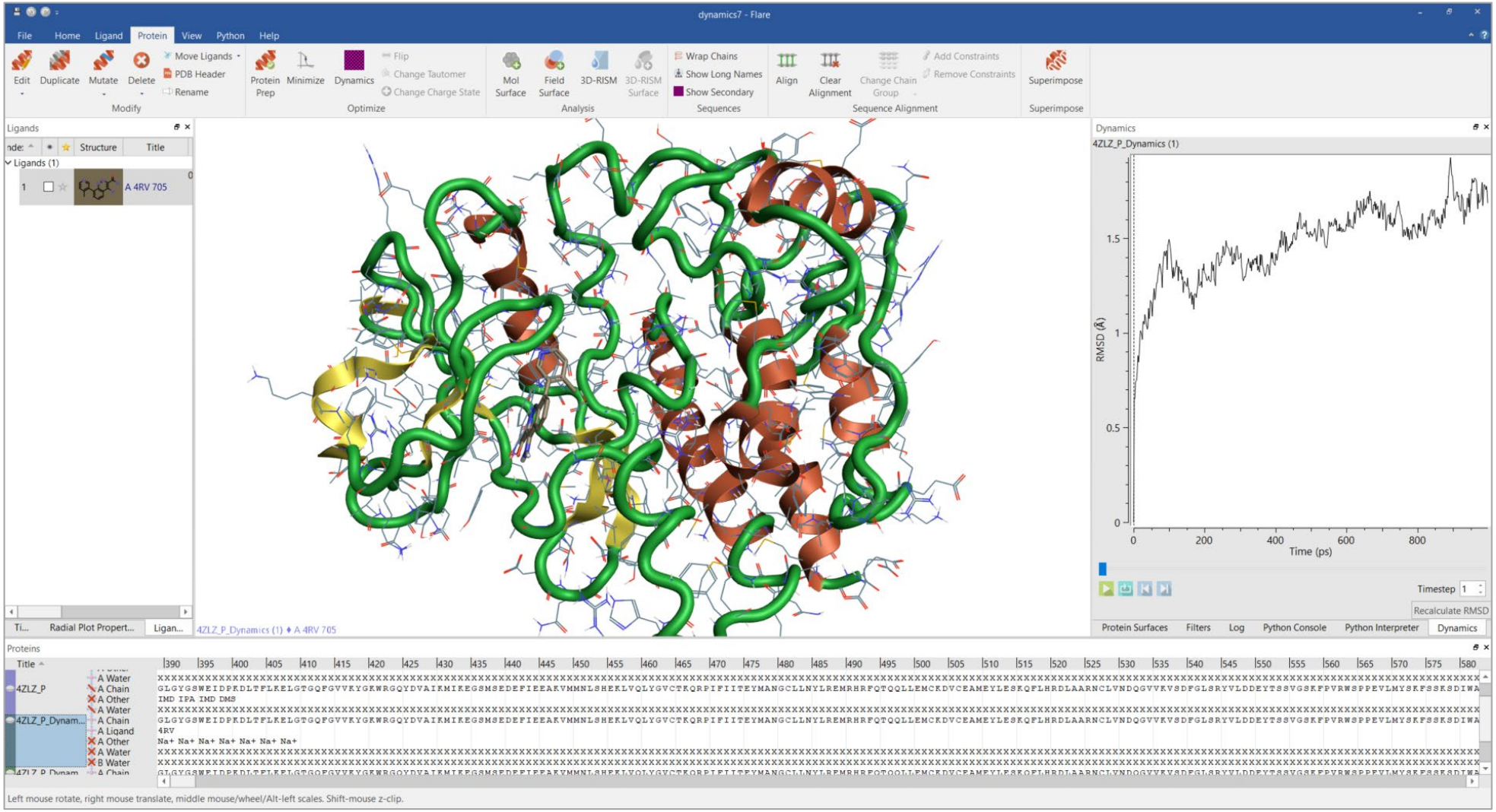
Molecular Dynamics

- Based on OpenMM/AMBER
- Flare V3 implementation:
  - Run dynamics
  - View results
    - Plot of RMSD
    - Player to browse through frames
    - Save frames to project
    - View log
- More to come!

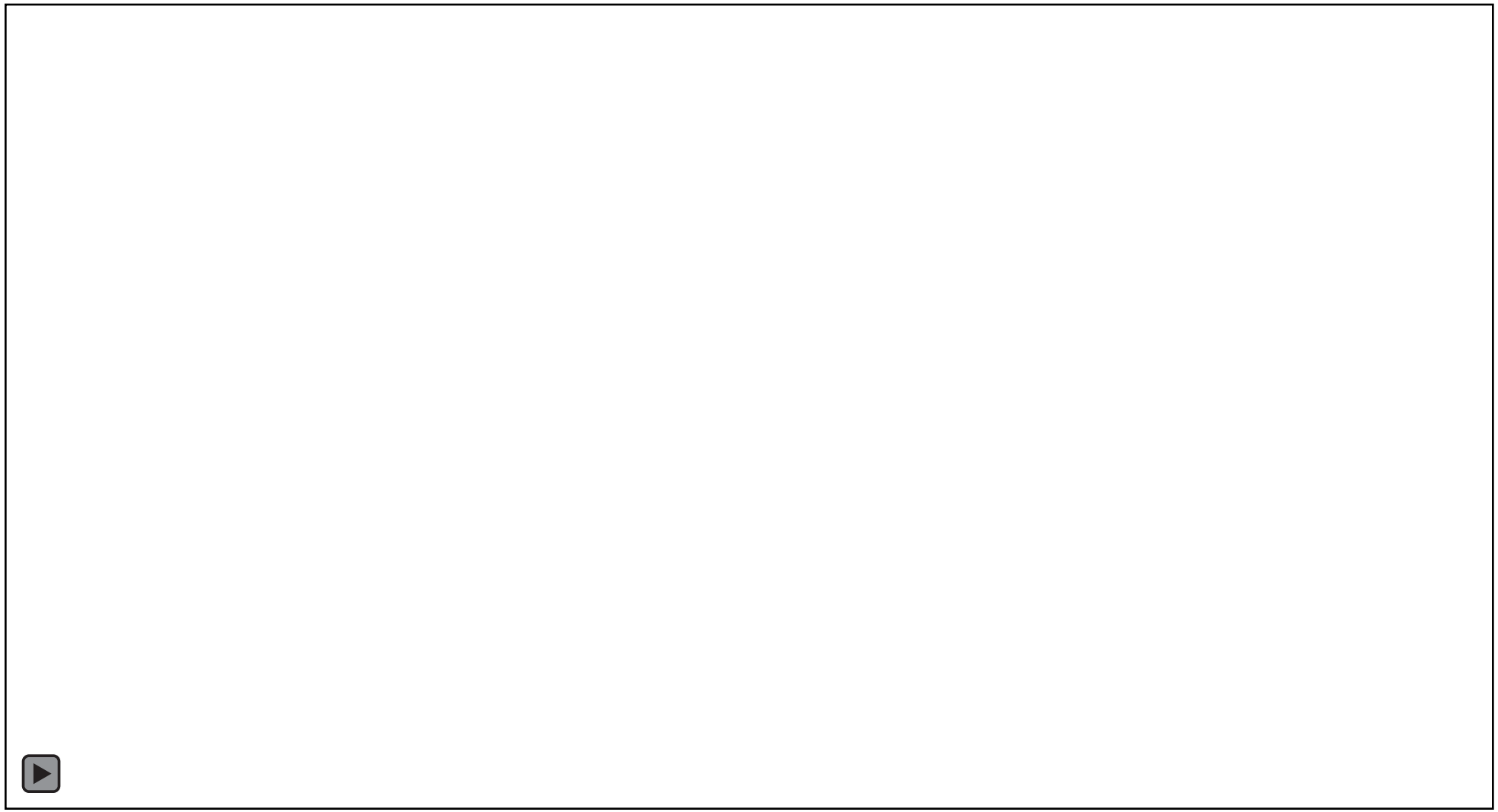




Molecular Dynamics results



Editing in the main window



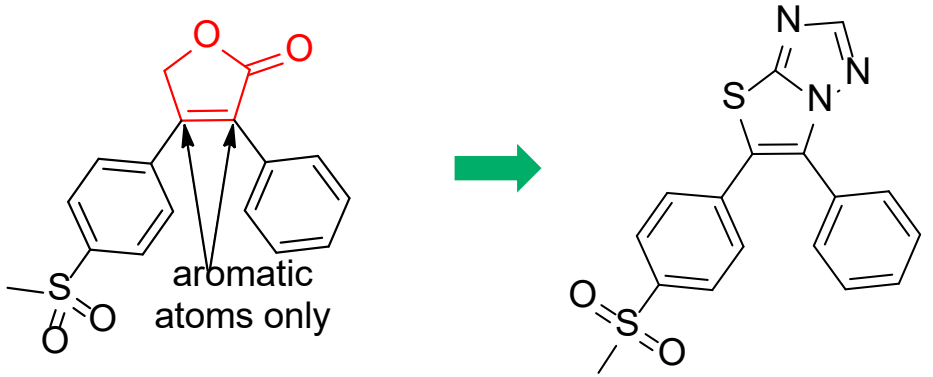
Highly innovative new ideas for your project

spark

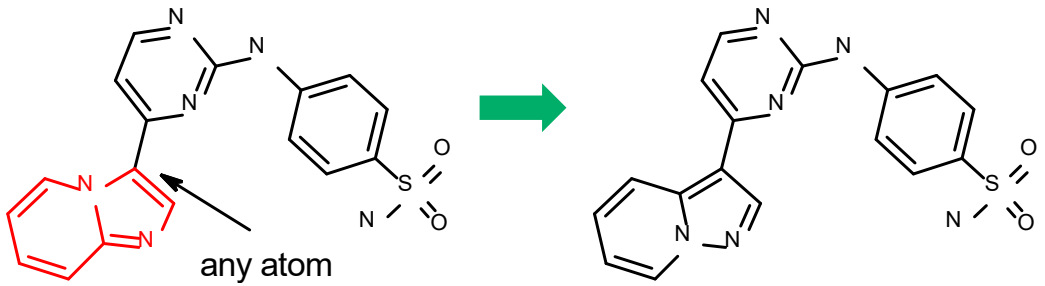
'The best bioisostere application on the market'

- Scaffold hopping
  - Using fragments of real molecules or commercially available reagents
- R-group exploration
  - Using commercially available reagents
  - Monthly updates for updated availability information
- Advanced bioisostere replacement experiments
  - Replace in one molecule, score against a second
  - Fragment growing and linking
  - Crystallographic water replacement
  - Macrocyclization

Scaffold hopping



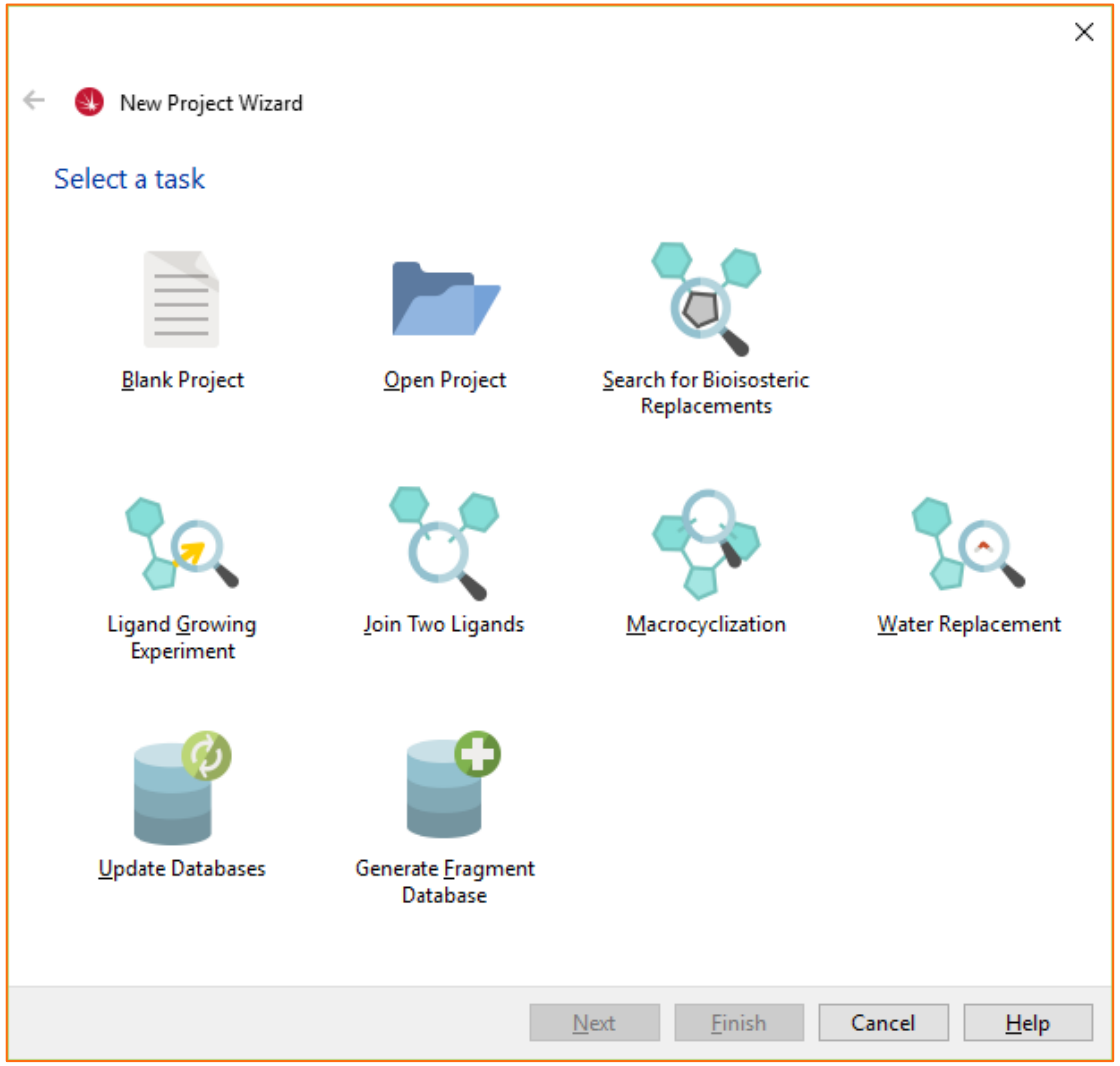
R-group replacement



Spark V10.5

> Released January 2018

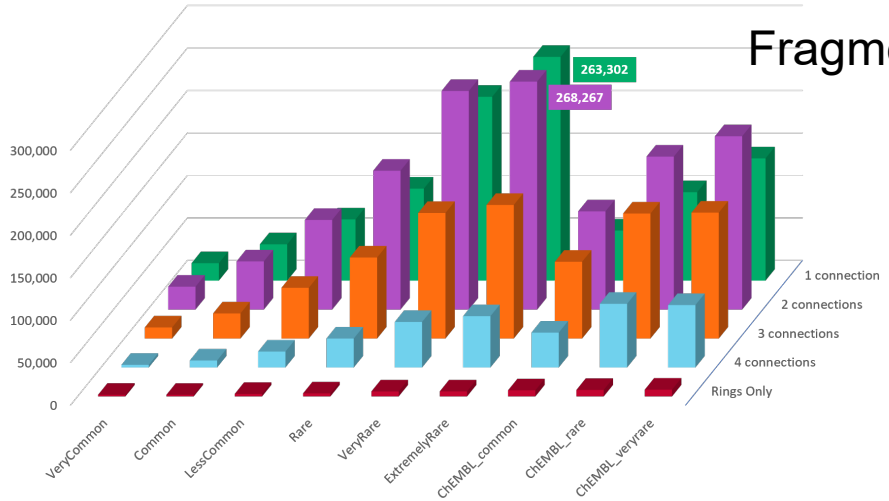
> New wizards for more complex experiments



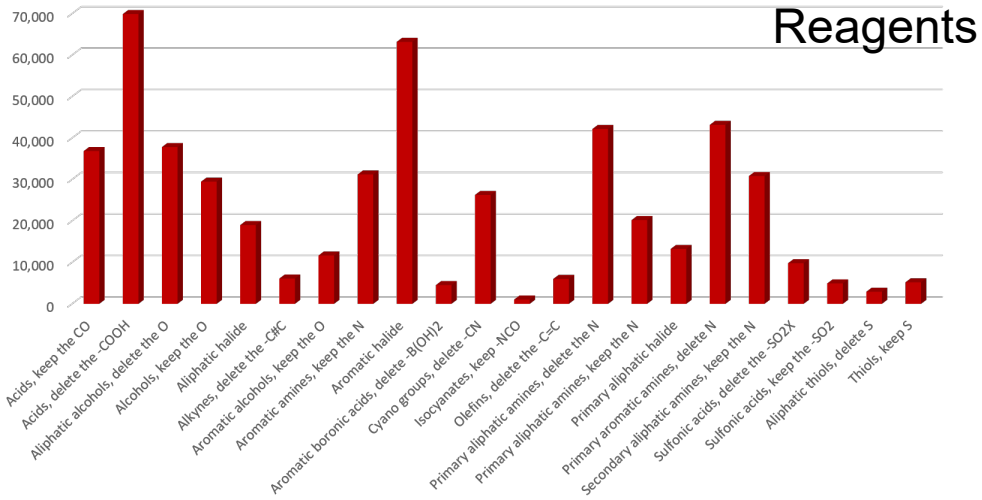
Spark V10.5

- Released January 2018
- New wizards for more complex experiments
- New databases – now distributing 3M unique fragments
- Other improvements
  - Enhanced Spark database update functionality
  - Pharmacophores
  - Additional similarity metrics – Tanimoto & Tversky
  - ‘Flexibility’ filter during the search
  - Annotated Storyboard
  - Query Editor
  - New 3D graphic engine

Fragments



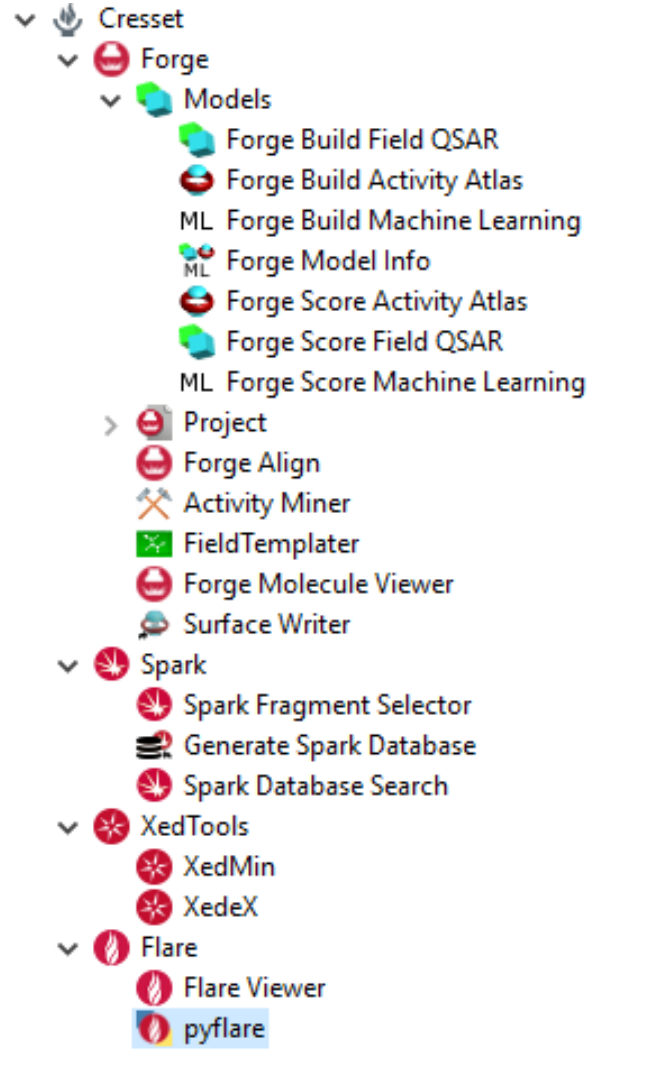
Reagents



Cresset KNIME nodes and Pipeline Pilot components

> Released April/May 2019

> New and enhanced KNIME nodes and PP components to access most recent Forge, Flare and Spark functionality



Forge and Spark: what's next?

Building together the products of the future

What our users say

I never remember which product I should use

I want to compare Activity Atlas maps against multiple proteins, to understand selectivity

Can I set Spark to make a H-bond interaction with this residue?

I want to do all my modelling work from the same GUI

How do I post-process Spark results in Flare?

I want to use ligand alignment in Flare

Can I import EC scores in Forge?

I want a docking-scoring-function-like capability to Spark to allow it to score replacements directly against the protein

I want to generate conformers in Forge using the MMFF94 RDKit implementation

How do I display cation-pi interactions in Forge?

I use both ligand- and structure-based approaches in my projects

Exporting radial plot properties in Forge is great. Can I have it also in Spark?

We listen!

- We hear you calling for:
  - New science
  - Simplicity of product offering
  - Integration across products
  - Easier access to our methods
  - Easier access to third party and in-house methods
  - Flexibility of licensing
  - User friendliness

Our vision

> Integrate Forge and Spark into Flare

> The advantages:
  > Simplification of product offering
  > A single, integrated GUI
  > Ligand- and structure-based methods working in synergy
  > Full access to all functionality through the Python API
  > Modern, user friendly architecture
  > Flexible, modular license structure

> The challenges:
  > Safeguard the user experience
  > Keep it easy to use
  > Keep learning curve at a minimum

cresset®
software

innovative science • intuitive software

Let’s talk. We listen.

Many thanks to:

Tim

The development and science team (Mark, Nigel, Ken, Paolo, Rosie, Matthias, Martin, Mary M, Joy, Andy S, Andy V)

cressetgroup