McGill – Molecular Forecaster Inc.

Forecaster Platform 1.1 – User Guide

21 February 2012

TABLE OF CONTENTS

DEVELOPMENT TEAM

Team leader: Prof. Nicolas Moitessier (nicolas.moitessier@mcgill.ca)

Development and support team leader: Dr. Eric Therrien (eric@molecularforecaster.com)

Developers: Dr. Nathanael Weill, Dr. Valérie Campagna-Slater, Andy Arrowsmith, Hugo Boyer.

The programs FITTED, SMART, PROCESS, SELECT, REDUCE and REACT, have been developed for the most part by Chris Corbeil (PhD 2009) and Pablo Englebienne (PhD 2009) as part of their graduate studies. The initial version of the platform FORECASTER has been developed by Sequence Technology Inc. IMPACTS was developed by Valérie Campagna-Slater (post-doc 2011). Further developments and improvements are currently ongoing.

We thank other group members (Janice Lawandi, Stephane De Cesco, Devin Lee, Mitchell Huot, Joris De Schutter, Melanie Burger and Rodrigo Mendoza-Sanchez), collaborators and users for their valuable feedback.

Please refer to this platform by citing the following publication:

"Integrating Medicinal Chemistry, Organic/Combinatorial Chemistry, and Computational Chemistry for the Discovery of Selective Estrogen Receptor Modulators with Forecaster, a Novel Platform for Drug Discovery" Eric Therrien, Pablo Englebienne, Andrew G. Arrowsmith, Rodrigo Mendoza-Sanchez, Christopher R. Corbeil, Nathanael Weill, Valérie Campagna-Slater, and Nicolas Moitessier, *Journal of Chemical Information and Modeling* **2012** *52* (1), 210-224.

Contact: info@molecularforecaster.com

Website: http://www.molecularforecaster.com

THE REGULAR USER INTERFACE

MINIMUM REQUIREMENTS FOR CLIENTS

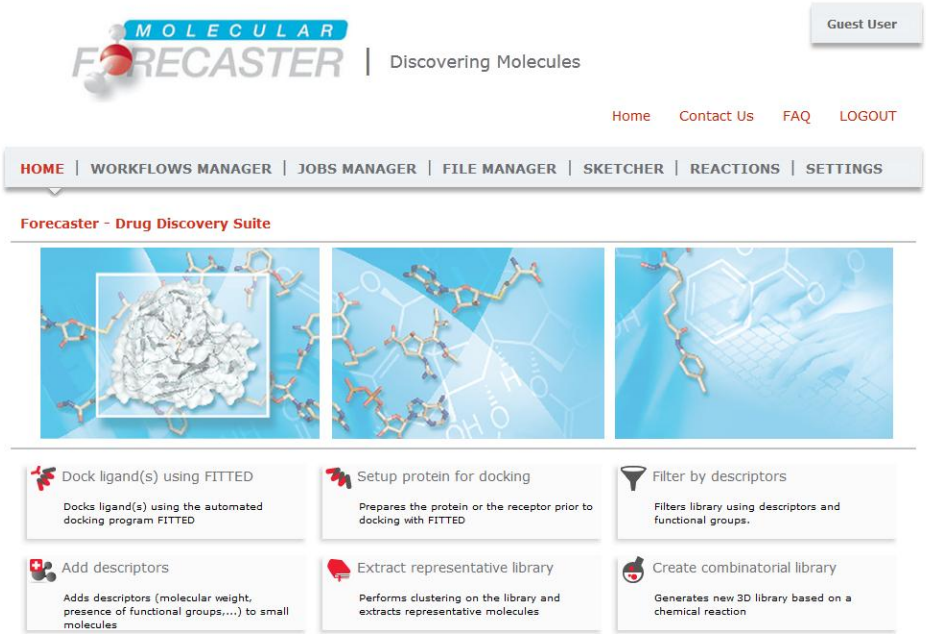
FORECASTER has been tested on Internet Explorer 7 and higher, Firefox 3.6 and higher, Chrome, and Safari 4 and higher. Other browsers including Internet Explorer 6 are not fully compatible with the interface FORECASTER. However, they can still be used but some logos, fonts and other visual details may be affected.

LOGIN INTERFACE

When loaded in a browser, FORECASTER will display the login interface. User inputs login name and password to log into the system.

HOME INTERFACE

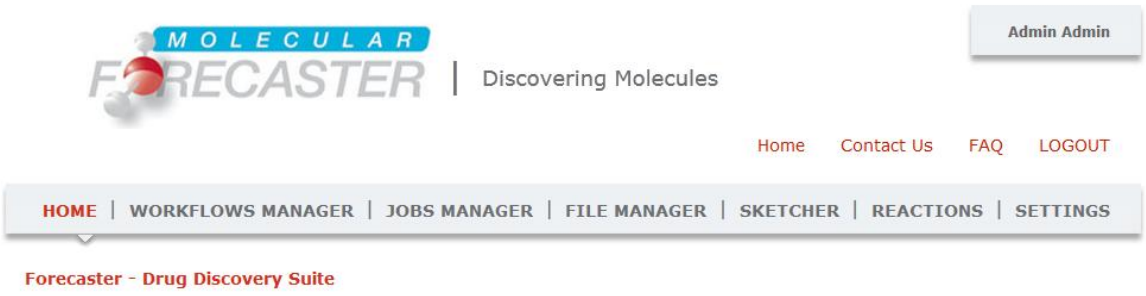
When logged in, the user will see the home page. This page presents the main features of the application.



WORKING WITH FORECASTER

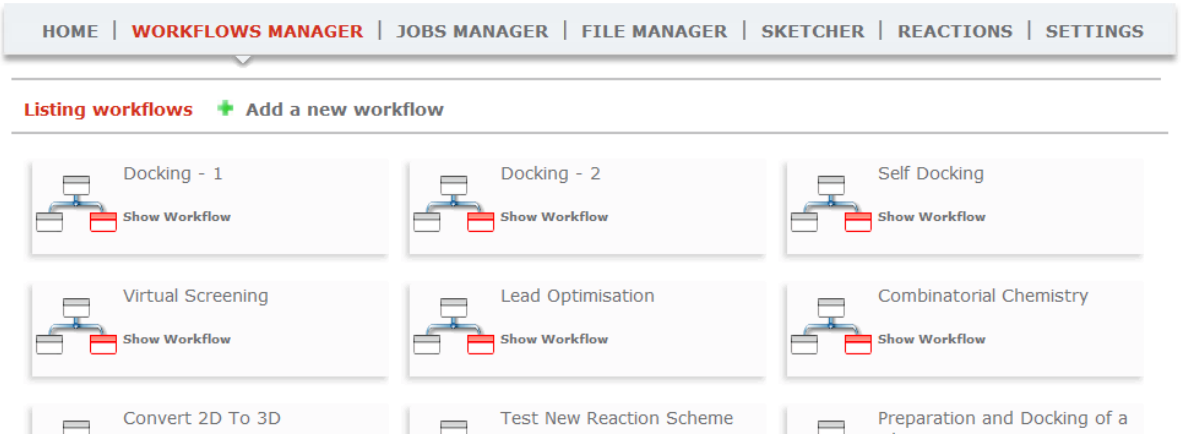
THE SECTIONS

When on the home page, the user can see the various sections (i.e., workflows manager, jobs manager) appearing on the top. Clicking on them can give access to the other functionalities as discussed below



WORKFLOW MANAGER

The workflow manager shows existing workflows.

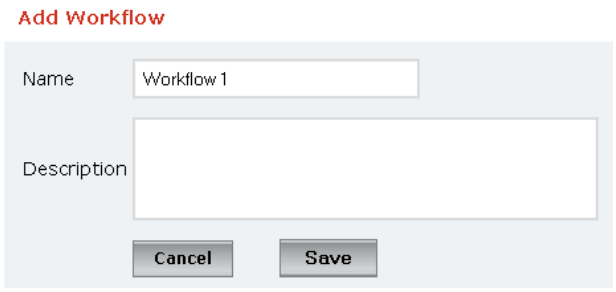


It allows users to view details of existing workflows and to create new ones. Users can only edit and delete workflows that have been created in addition to the out-of-the-box workflows. The workflows can be edited and deleted only when they are not used by any job. In the Figure below `Docking 1` is hard-coded and is given as an example by the developers and cannot be edited nor deleted. `Workflow 1` was created by a user and can be edited/modified and/or deleted.

To view the schematic details of an existing workflow, click on Show Workflow.

ADDING A NEW WORKFLOW

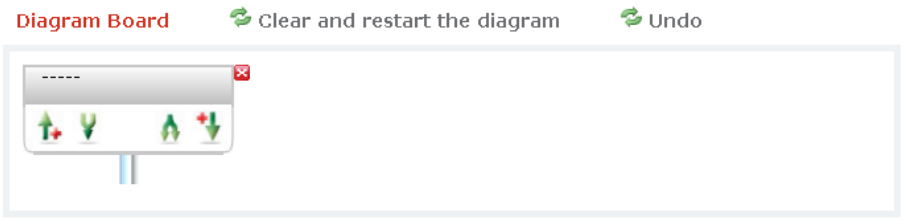
Clicking on the Add a new workflow icon allows the user to create a new workflow. A window appears where the user enters a name and a description of the new workflow to be created.



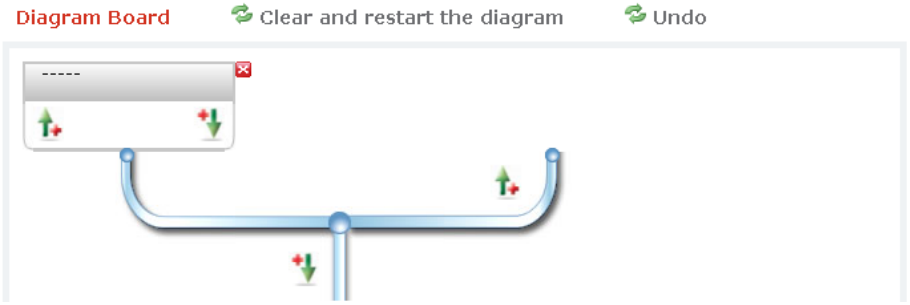
After clicking on Save, the user may create a diagram that represents the flow of actions to be executed by clicking on Start the diagram

BUILDING A WORKFLOW DIAGRAM

A new workflow diagram is opened by clicking on Start the diagram .

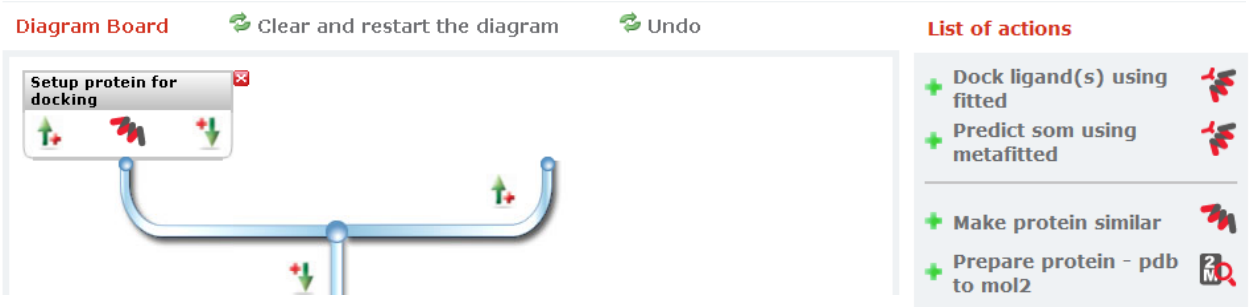


User can add boxes and connectors to the diagram by clicking on the green arrows. A creates a split, creates a merge, and create boxes above and below the current one. By clicking on the icon the user can delete a box from the diagram.



A workflow represents a sequence of actions to be executed. **This sequence runs from left to right and from top to bottom**. In order to create a workflow, the user assigns actions to each

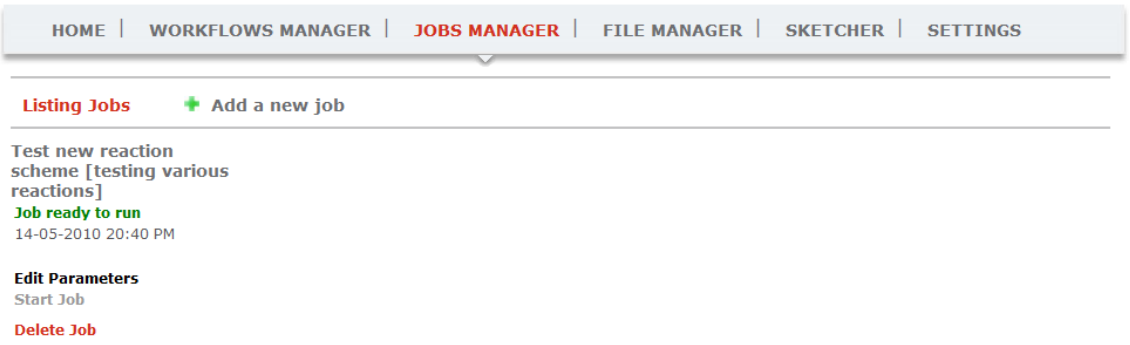
boxes by clicking on the box to be edited and clicking on the ✚ next to one of the actions located on the right hand-side or directly on the action label (i.e., Add descriptors).



Once the diagram is complete, user makes it active and available in the job manager section. The user created workflows become available to all the users on the platform. There is no validation for the compatibility of the actions within the workflow. The user needs to know which actions can be connected together (see the actions section below).

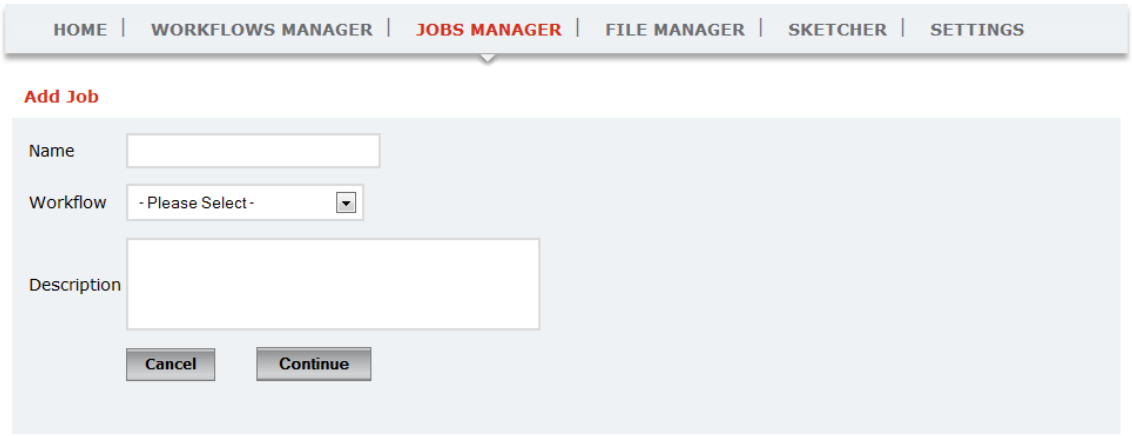
JOBS MANAGER

The job manager allows users to create a job based on a workflow. The first step is to parameterize the job, the second step is to execute the job, and the final step is to visualize the results.



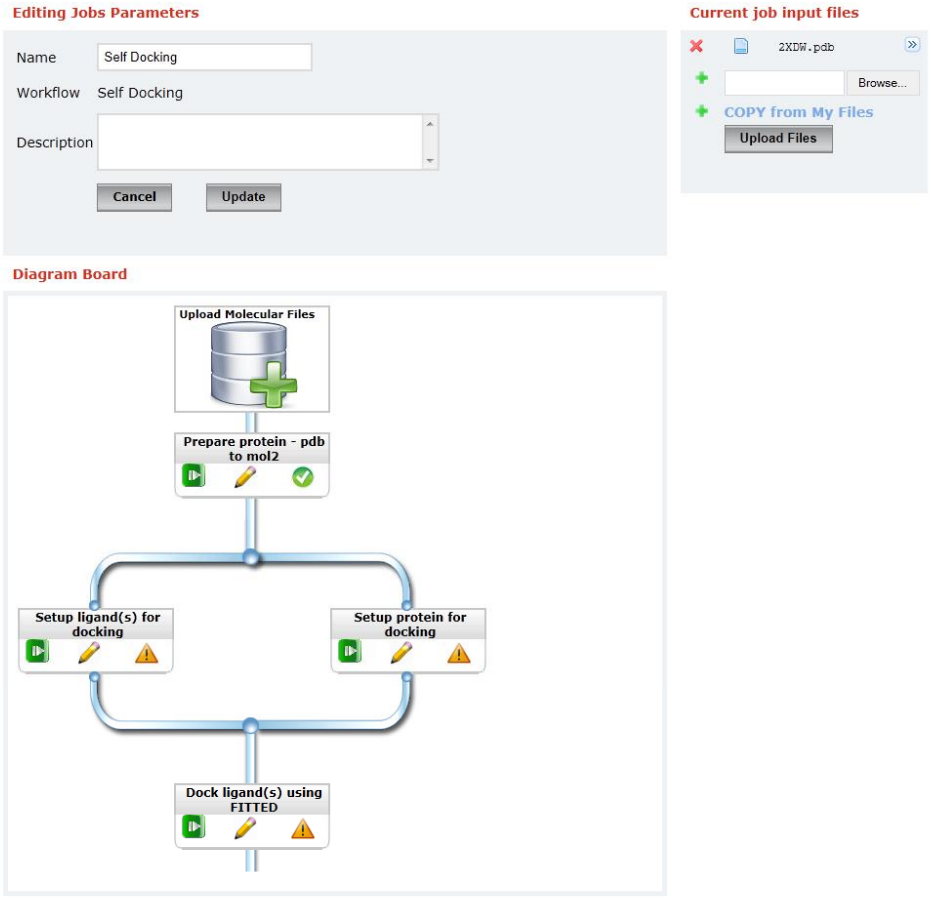
ADDING A NEW JOB

Users add new jobs by clicking on ✚ Add a new job icon, entering the job name, choosing the workflow to be used and clicking continue.

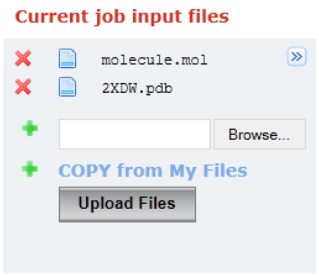


EDITING THE PARAMETERS AND SELECTING THE FILES TO PROCESS

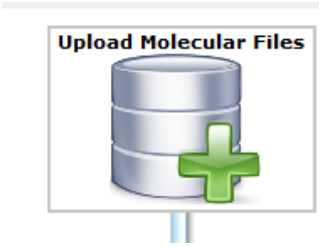
The editing screen displays the diagram created previously in the workflow manager. In this section, the user has to setup each action of the workflow. This setup is done using a form, which allows users to modify settings and select files to be processed in the job.

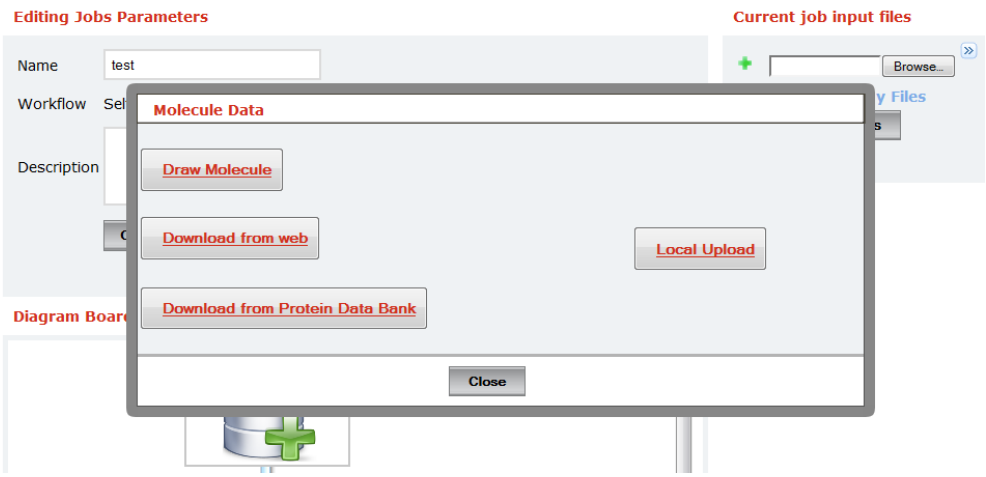


First, users have to upload files to be processed by directly uploading them to the system or by copying them from My Files folder. There are two possibilities to add files to the job. Files can be uploaded or transferred from My Files folder using the right-hand side bar.

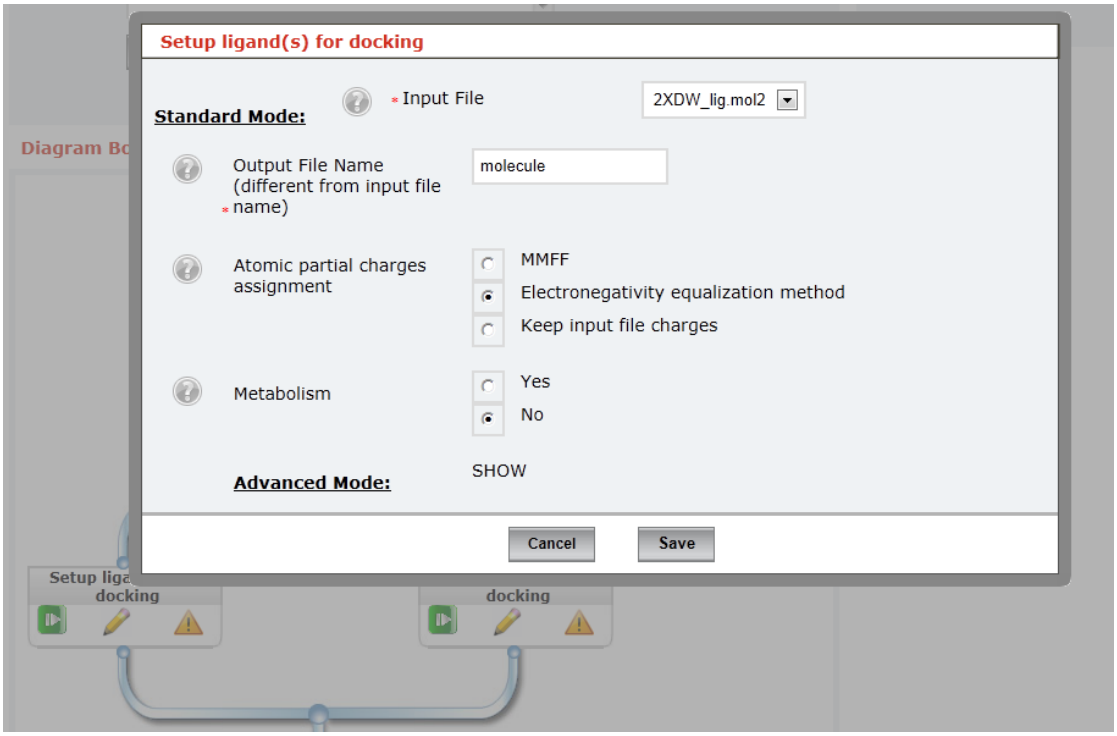


Alternatively, clicking on the container labeled Upload Molecular Files brings a window with different possibilities such as drawing directly the molecule, downloading from the web, retrieving from the PDB database or performing a local upload.

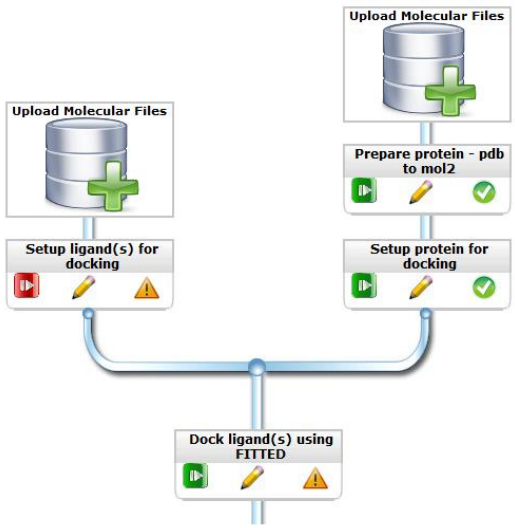




Once the files are uploaded, users can then configure actions. In a workflow, each action (i.e., box) outputs a result file which then becomes the input file for the following action. As the file does not exist yet (the job has not been run yet), an internal predictor guesses the name of these files. As a consequence, **the boxes should be configured from top to bottom**.

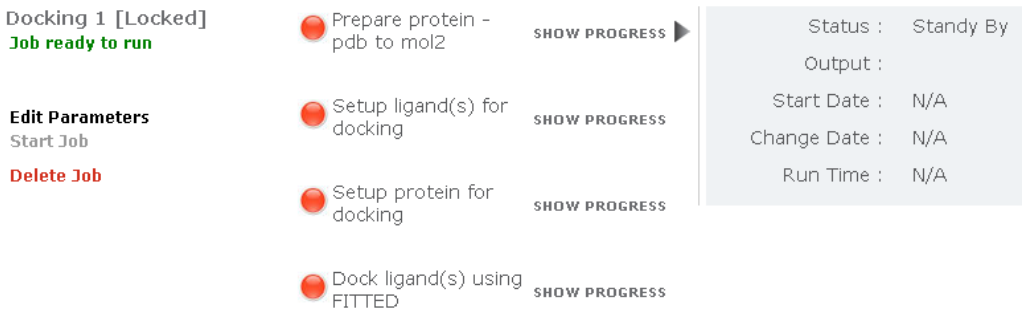


A **pause** feature is implemented which allows the user to stop the job after a specified action during the workflow so the output of an action can be visualized or edited. The job can then be **resumed** in order to proceed to completion. **This sequence runs from left to right and from top to bottom.** The figure below shows an example where the job will stop after the `Prepare protein – pdb to mol2` action from which the protein in mol2 and the ligand in mol2 are generated.



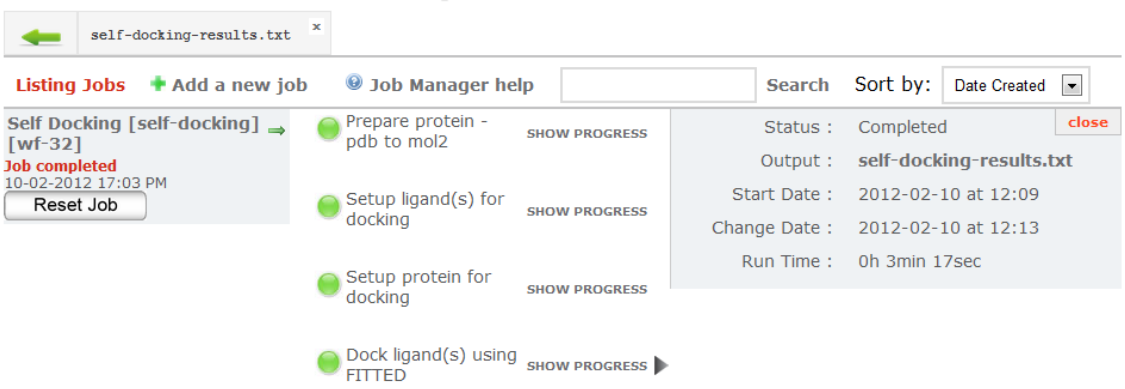
EXECUTING AND VIEWING JOBS

Once a job is correctly setup, it can be started. The job manager allows users to monitor the job and as each action completes its execution, color led will indicate its status. Output files can be viewed as soon as they are available.

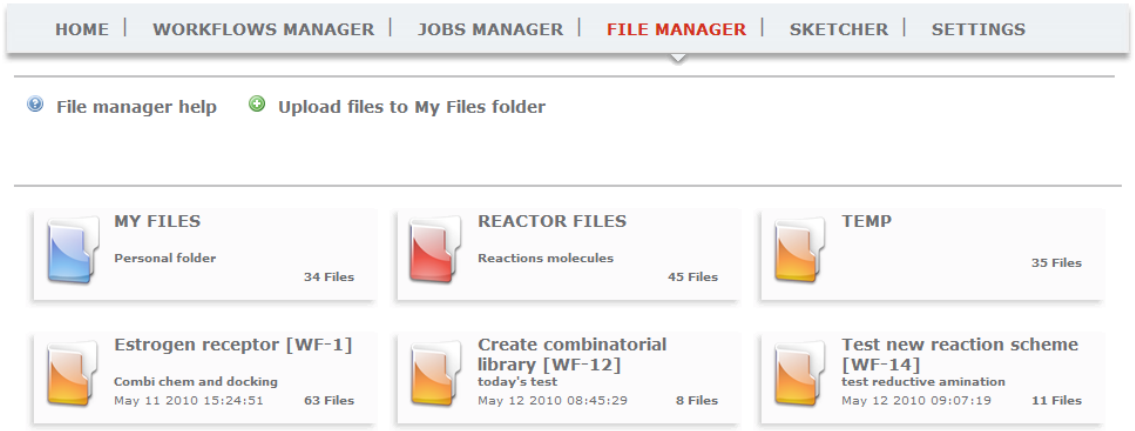


FILE PREVIEW

Once a job is completed (or running), the resulting files can be viewed directly in the job manager, unless the output is too large. In this latter case, the file should be viewed or downloaded in the file manager. A job can be stopped or reset at any time to allow users to re-parameterize and restart it at their convenience. When a job is running, a corresponding folder is created in the file manager where all input and output files are stored. Each job belongs to a particular user and cannot be viewed by other users.



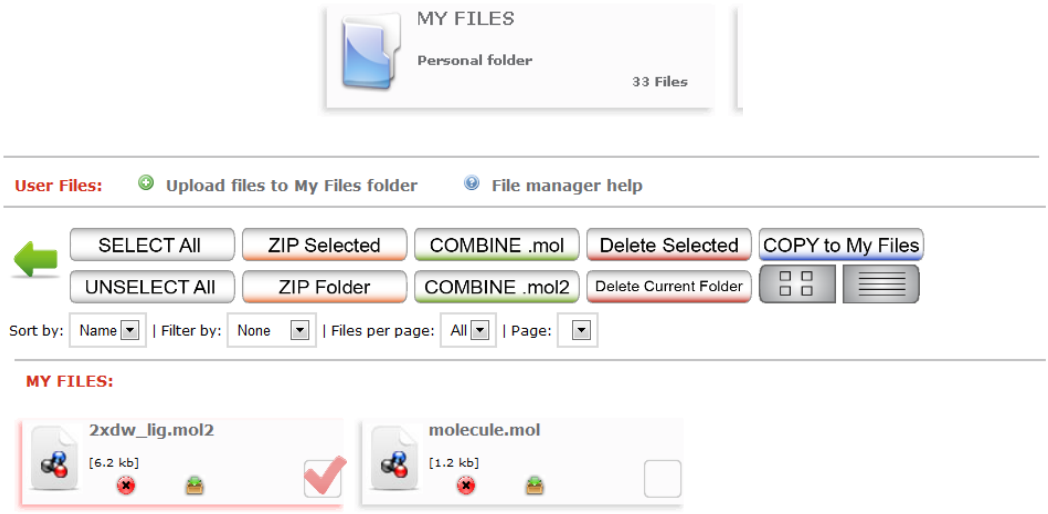
FILE MANAGER



The files in the file manager can be deleted or downloaded. Users can also zip the entire folder or just several selected files. In order to be visualized by the Jmol plugin, multiple mol2 files can be combined by clicking on the COMBINE .mol2 icon. The Combine Mol2 function assembles several mol2 files (i.e., a protein and a docked ligand) into a single file compatible with Jmol. The Icon and List View buttons allow users to change how they view their files.

FOLDER - MY FILES

Each user has a personal folder called “My Files” in which files from different jobs can be saved to be reused in other jobs. It is used as a transfer folder between a given user’s jobs.



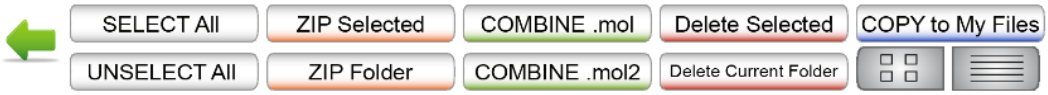
JOB-SPECIFIC FOLDERS

Files used by each job are saved in the job's folder using a naming convention such as WF-{id number} e.g. WF-8. The job-specific folder will then read "Not synchronized with the system" and it is at user's discretion to delete files contained in that folder.

FILES AND ACTIONS

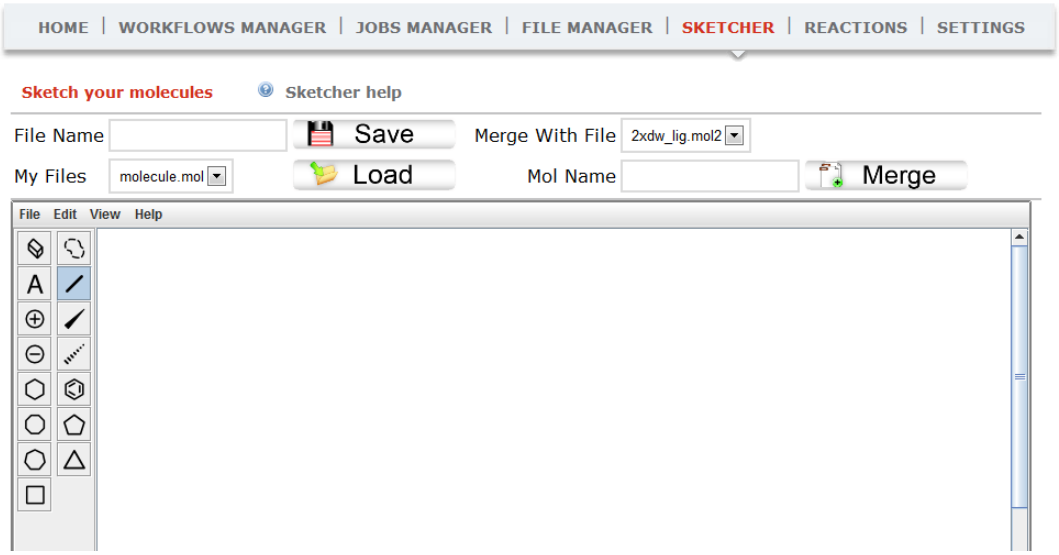
Files are viewed with different programs depending on their type. All text files are opened in a text editor directly integrated into the interface, while mol2 files are visualized using the Jmol java applet. Files that are bigger than one megabyte (1 Mb) must be downloaded in order to be viewed or edited.

Files located in all folders other than "My Files" folder can be copied to "My Files" by clicking on the "Copy to My Files" icon.



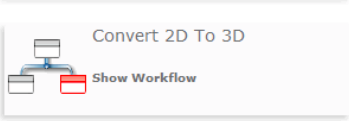
SKETCHER

The sketcher is a tool that allows users to draw molecules using the ChemWriter java applet.



SAVING AND LOADING FILES

Users can save all molecules they draw using the sketcher in their "My Files" folder. The platform provides an action called "Convert 2D to 3D" which converts the two dimensional drawings ("mol" or "sdf" file format) into three dimensional mol2 files that can subsequently be used as inputs in workflows.



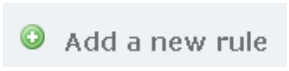
Any molecule that is drawn and saved can later be loaded and reopened in the applet for viewing and editing.

REACTIONS MANAGER

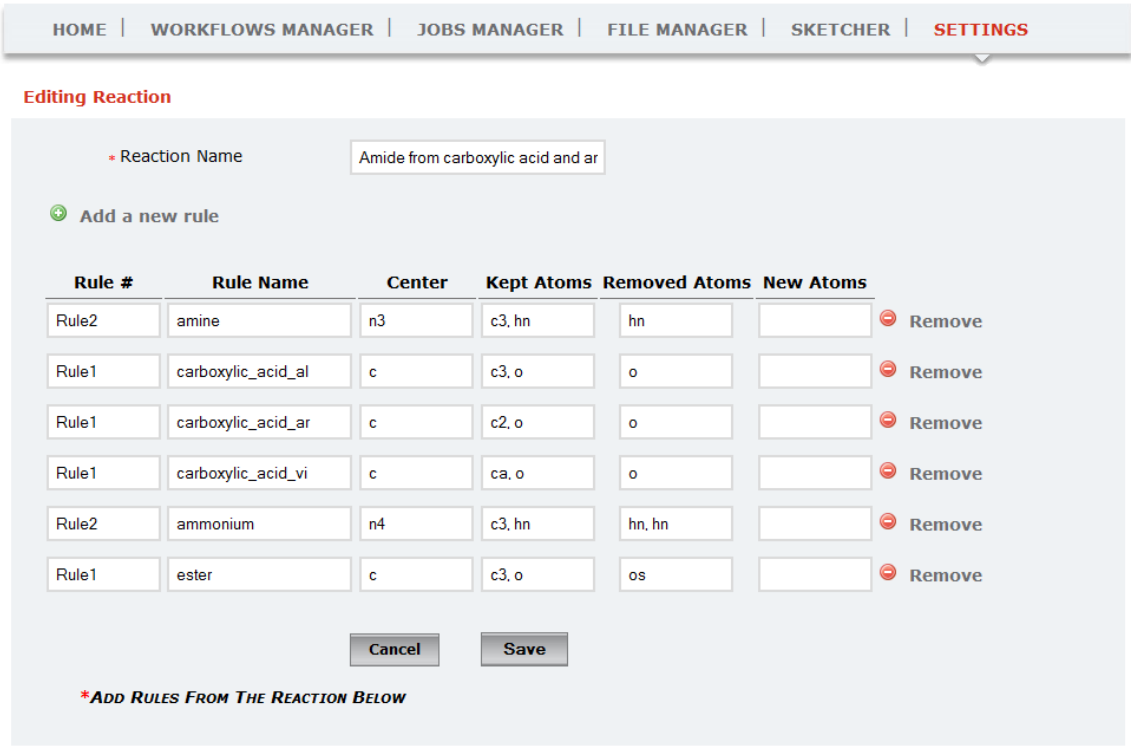
The reactions manager is a tool that allows users to define reaction rules used in the "Create Combinatorial Library" plugin. **This section might experience bugs; we are still making improvements in the automated reaction rules definitions.**

CREATING REACTIONS

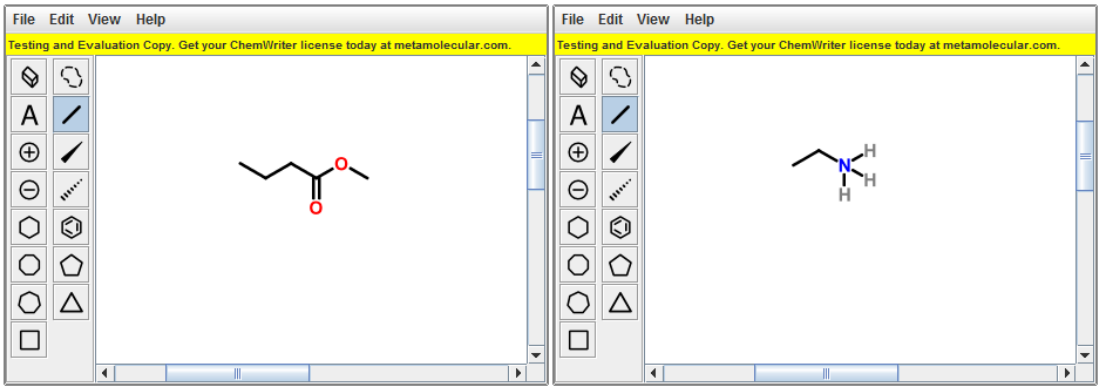
Users have two options when it comes to creating new reaction rules. They can either write the rules themselves or use the ChemWriter applet to draw the reaction.

After assigning a name to the new reaction, user can add as many rules as he likes by clicking on 

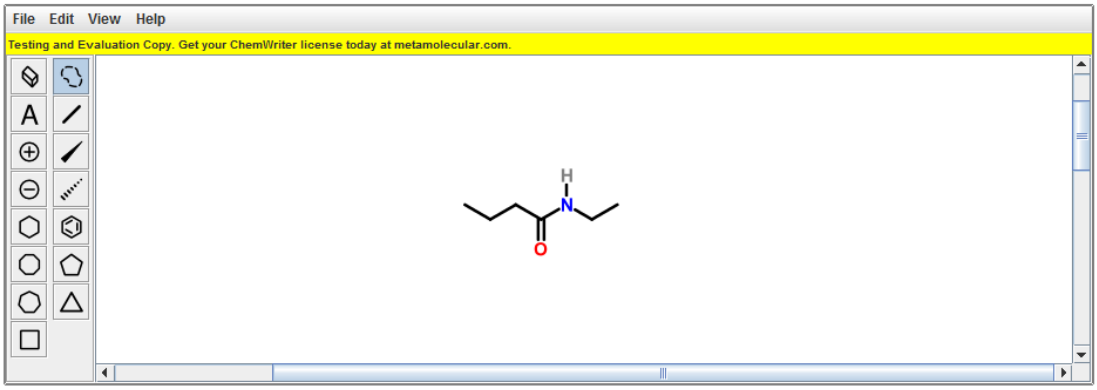
and filling out the appropriate fields. The rule number defines which reactant the rule applies to, the rule name is usually the reactant functional group, the center is the reacting atom center of the molecule, the kept atoms are the atoms that remain in the product of the reaction, the removed atoms are the atoms that are deleted by the reaction, new atoms are atoms the atoms that are created by the reaction. Once the reaction is defined and saved, it can be used in the Create Combinatorial Library plugin.



Alternatively, users can draw and save each reactant and their product using ChemWriter, click on Save all to save the drawn structures and have the rules generated automatically by clicking on *Add Rules From The Reaction Below



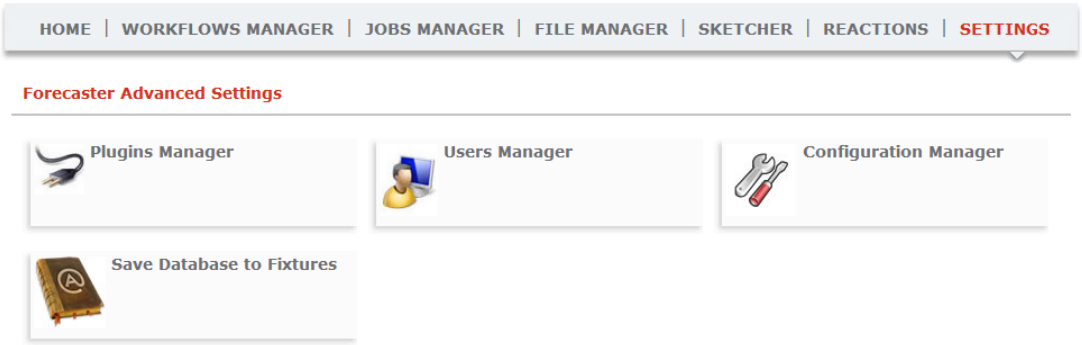
Draw Product #1



In the current version, the use of the java applet has limitations. Most of the coupling reactions (the most widely used reactions when building libraries) should work. However, rules for transesterification (replacement of an alkoxide group by another alkoxide group) are not found by the algorithm currently implemented. Similarly, reactions including two reacting atoms (such as double bond reduction) and metals (Grignard addition) are not yet considered. These issues will be addressed in the next version. If the user cannot produce the rules using the applet, the rules can be set manually. Otherwise, contact Molecular Forecaster (info@molecularforecaster.com) for the development of new rules.

Other limitations are regioselectivity (if more than one functional group of a kind is found, the first one will react) and stereoselectivity (it a chiral center is formed, a single isomer is produced). Stereoselectivity will be considered in the future.

ADMINISTRATOR INTERFACE - SETTINGS



USER MANAGER

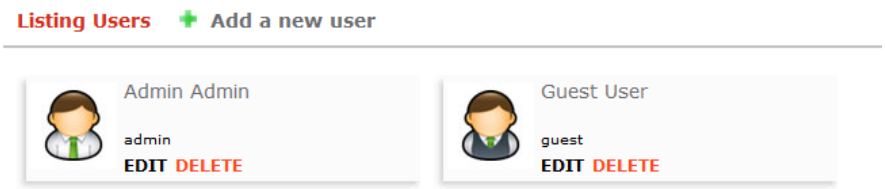
The user manager is located under the settings menu tab. It allows the administrator to manage users and their roles in the platform.

USER ROLES AND ACCESS

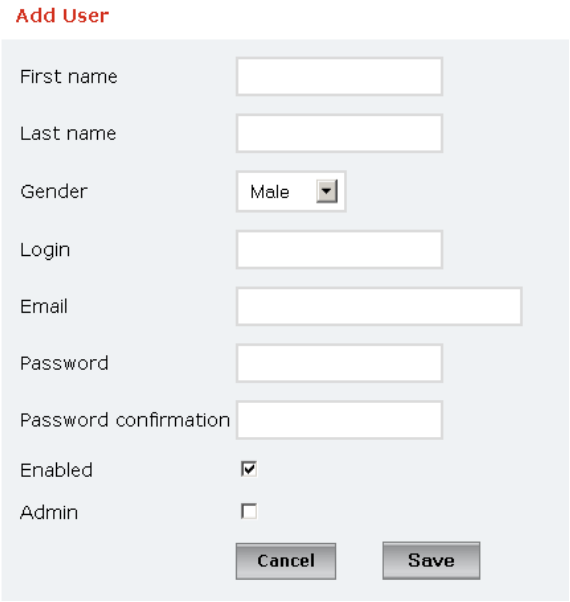
Each user has a role that is assigned by the platform administrator. Depending on their roles, users are given authorization to access certain features of the program.

ADDING NEW USERS

Only the program administrator can add or delete users and edit access settings for existing ones.

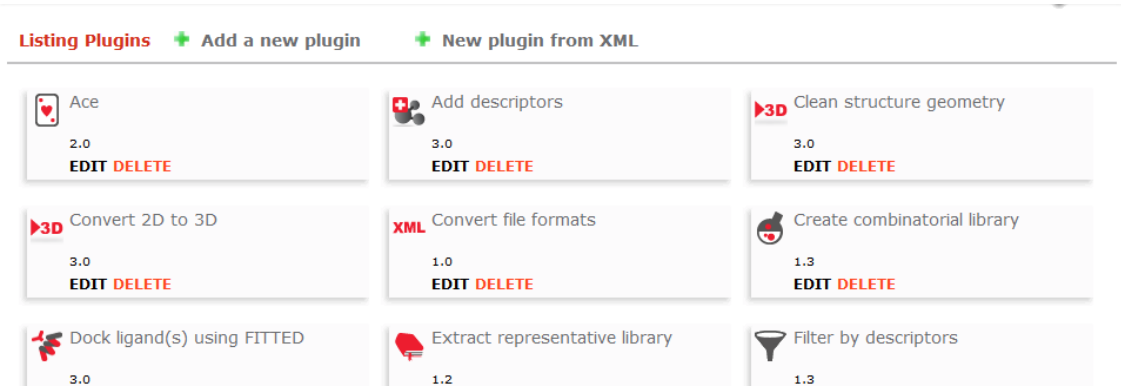


Clicking on the ✚ next to “Add a new user” allows the admin to create a new user. By checking the “Enabled” box, the admin allows the new user to access the application.



PLUGINS MANAGER

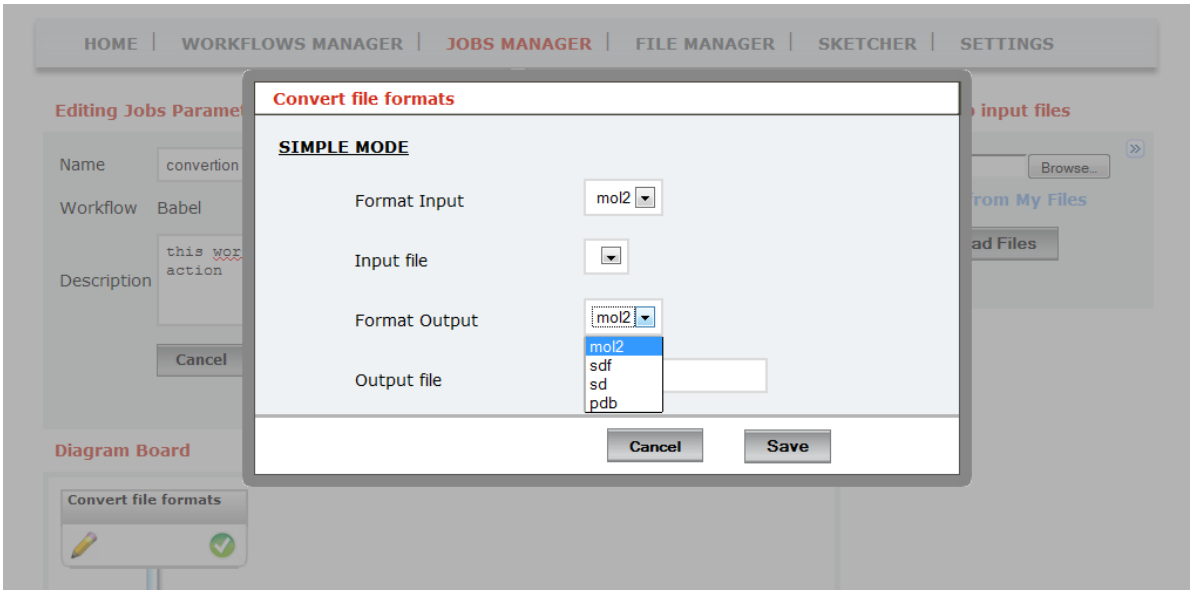
The plugins interface allows users to modify simple information about existing plugins or to delete them.



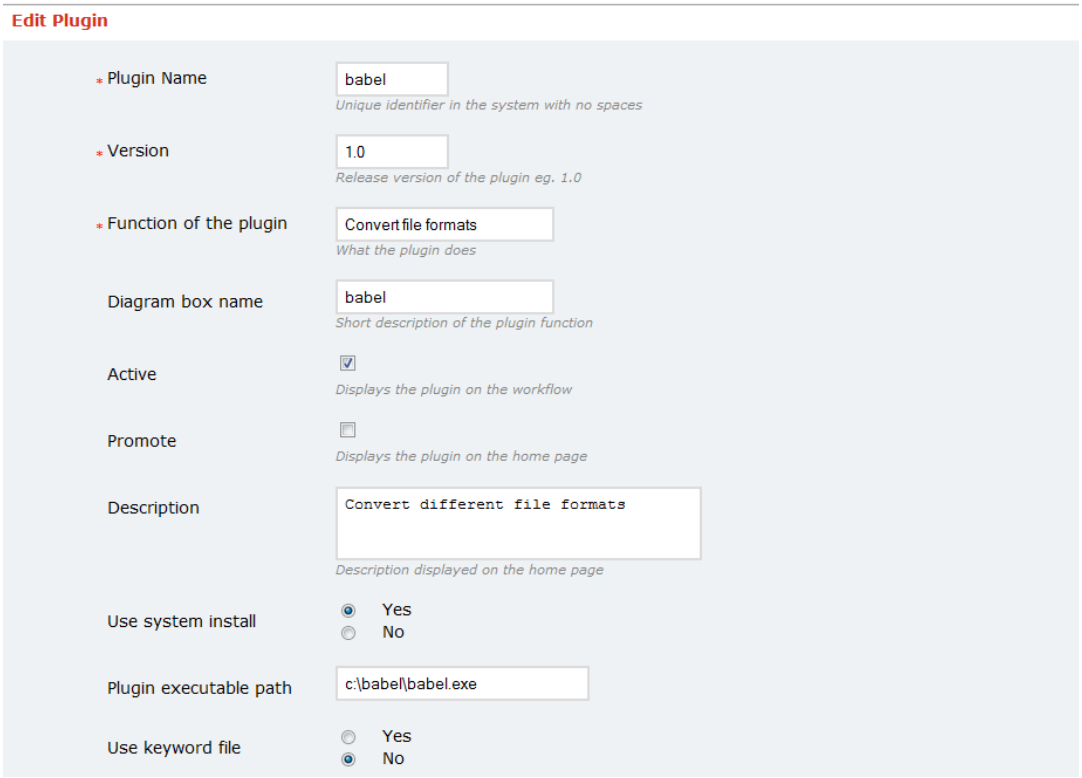
New actions (programs) can also be integrated into the application. When creating a new plugin, users define all the necessary parameters required for the plugin to work inside of a workflow, such as name, version, and executable files.

ADDING NEW PLUGINS

Users can add new plugins (i.e., new actions) by clicking on Add a new plugin and defining fields in the plugin interface. In the example described below an action called “convert file format” will be created. This action can next be used in any workflow. This action uses babel as an executable and we would like to create a form to set it up when preparing jobs. The created form is shown below while the preparation of this form and its implementation as an action is given in the subsequent section.



This form is associated to an action (with a name), an executable (i.e., program) and keywords or arguments (if the executable uses a command line). By clicking on Add a new plugin and defining the following window appears and needs to be filled.



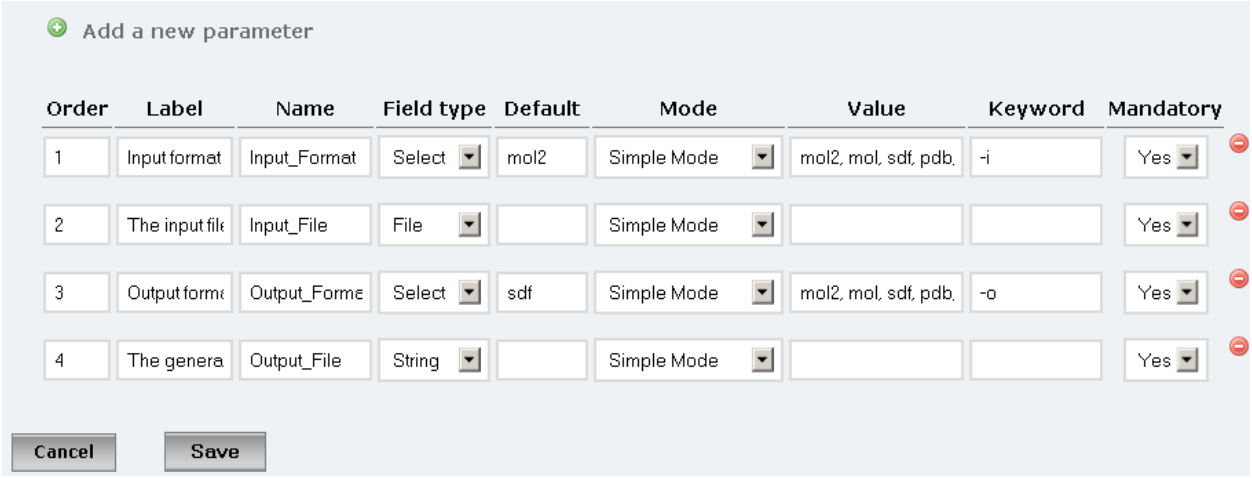
Each new plugin has to be assigned a unique name and version. The function of the program may also be described. Additional details about the plugin's purpose may be added in the "diagram box name" field. Users can choose to promote the plugin by checking off the "promote" box, which will ensure that the new plugin is shown on the home page. A description of the plugin's function that will appear on the home page can also be added.

If the plugin's executable file is located on the platform's server, users can use the system install, by checking off "yes" next to "Use system install" and providing a path to the file. If the plugin's executable file is not located on the platform's server, users have to provide it and upload it to the platform.

Some programs work by reading the parameters from a keyword file (usually a text file) while others uses various arguments in command line. To instruct the platform to either write a command line or a keyword file, users choose whether or not the arguments for the executable file are located in a keyword file. Selecting "yes" means that arguments are written inside of the keyword file. Selecting "no" means that arguments are specified directly on the command line.

ADDING PARAMETERS

Users add plugin parameters by clicking on Add a new parameter and filling out the appropriate fields. Each parameter represents a field in the job manager edit screen.



Order – specify the order in which fields appear on a form

Label – describe the field. This label will be used as label in the form

Name – specify internal field name in a single word, no spaces.

Field type – there are seven possible field types that can be defined.

Check box, radio button or select – when defining one of these field types, the options must be specified in the value field, separated by commas, and the default option will be selected.

String or text – is a text input to be filled out by the user.

File – when this field type is defined, a file is selected from a drop-down menu containing a list of files uploaded to the job using the new plugin.

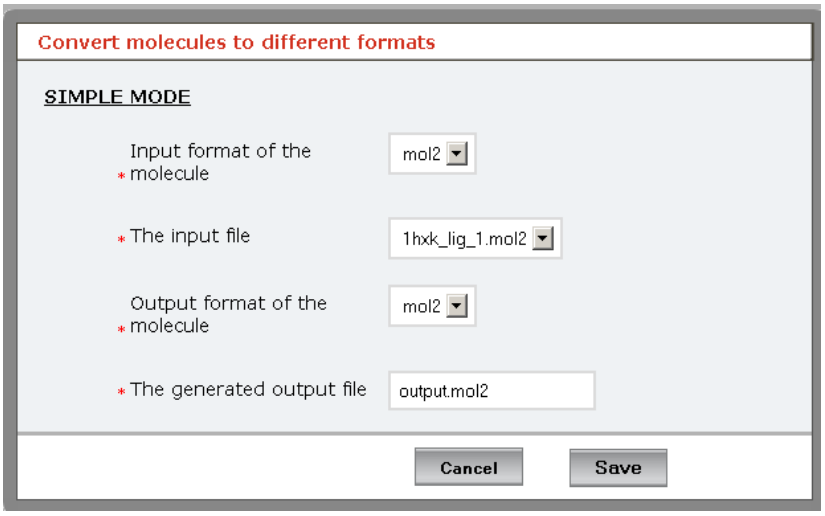
Hidden – a hidden field contains the keyword value that will be used and that cannot be modified by the user.

Mode – fields can appear in the simple or the advanced mode. The simple mode contains common parameters, modified by most users. Parameters that are less likely to be changed or that are used by more advanced users are placed in the advanced mode.

Keyword – is the name that will be used in the keyword file. It will be followed by the value selected or written in the field, depending on the field type.

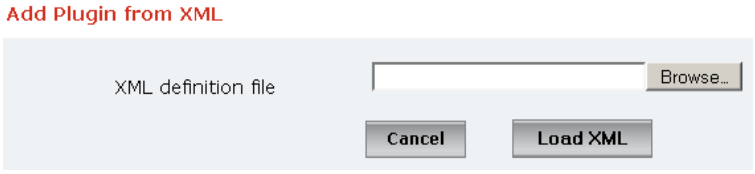
Mandatory – specify whether or not filling out a field is mandatory.

Once the form is saved it can be used to edit parameters used in a job.



ADDING NEW PLUGINS FROM XML

Users can also create new plugins by uploading and editing their own xml files in the system. They may do so by clicking on New plugin from XML. Adding a plugin from xml means that users can import an existing or sample definition of a plugin and modify it according to their needs.



The platform provides a sample file to help in the integration process. The file is located in the doc folder of the application. Once an xml file is loaded, the user gets the new plugin screen, identical to the one that appears when a plugin is created from scratch.

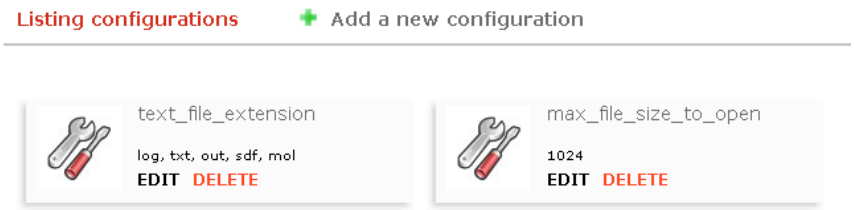
CONFIGURATIONS MANAGER

The configurations manager can be found in the setting menu tab. It allows the administrator to edit and create new configurations used in the platform.

DEFAULT SETTINGS

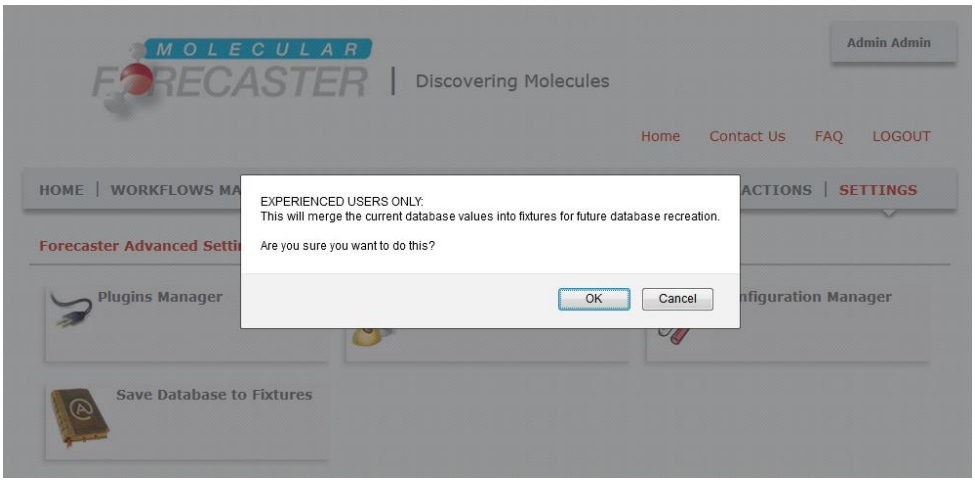
The current settings define the file types that are recognized and can be opened by the application's text editor. The default settings also identify the maximum file size that can be opened by the application without having to be downloaded.

The platform's administrator can add settings by clicking on Add a new configuration .

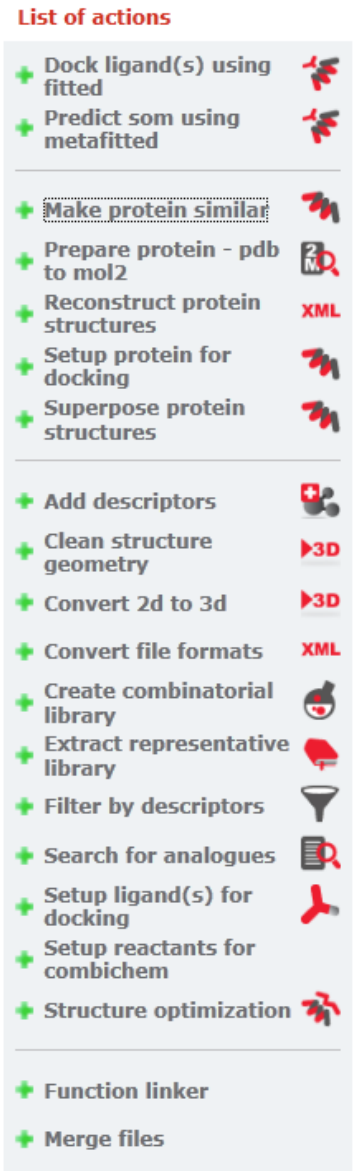


SAVE DATABASE TO FIXTURES

This option allows the administrator to store specified database information into files (called fixtures) which is used to re-create the database after an update of the platform. This action saves information about the workflows, jobs, users, plugins, reactions, and configurations that were modified since the first deployment.



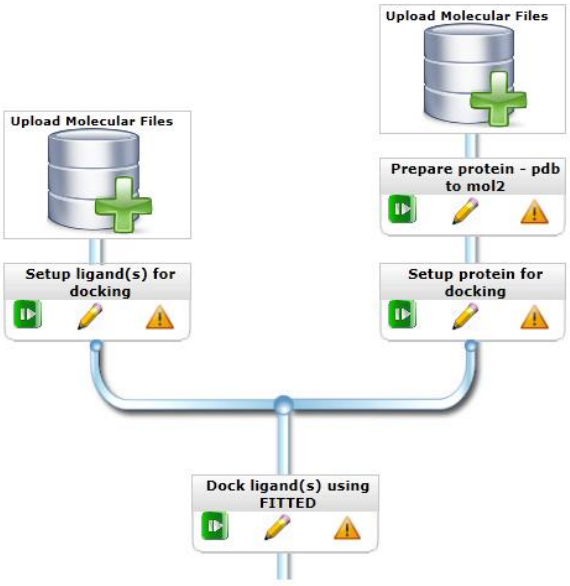
ACTIONS



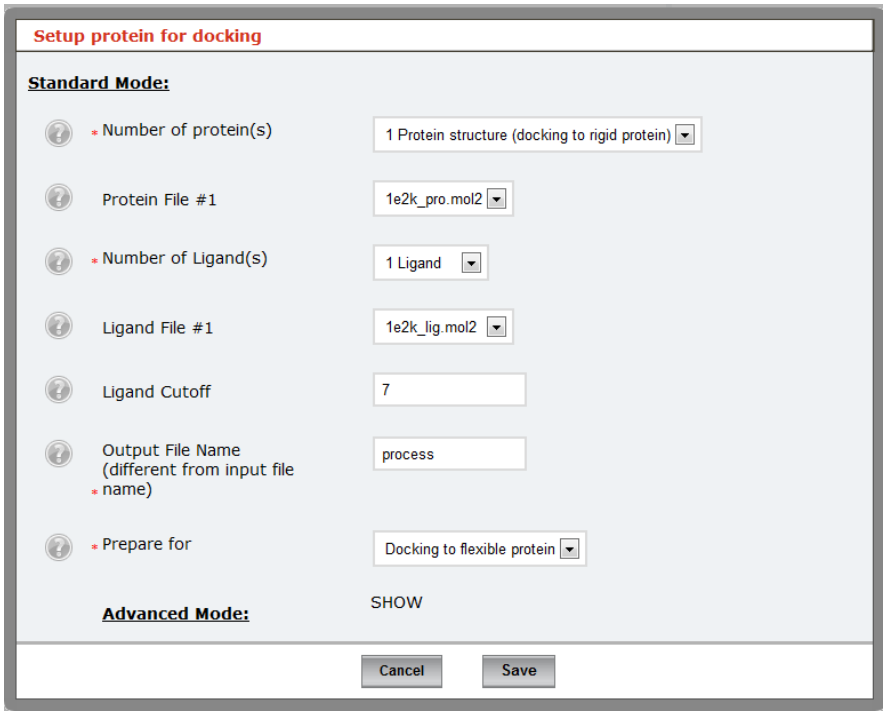
A number of actions are already integrated in the FORECASTER platform. Details about these actions are given below:

DOCK LIGAND(S) USING FITTED

The action `Dock ligand(s) using Fitted` takes a ligand or a library of ligands and docks it to a protein. It requires that the ligand(s) are setup properly using the action `Setup ligand(s) for docking` and that the protein has been prepared as well using the actions `Prepare protein – pdb to mol2` and `Setup protein for docking` as shown below. The format of the input files is therefore taken care of by the platform.



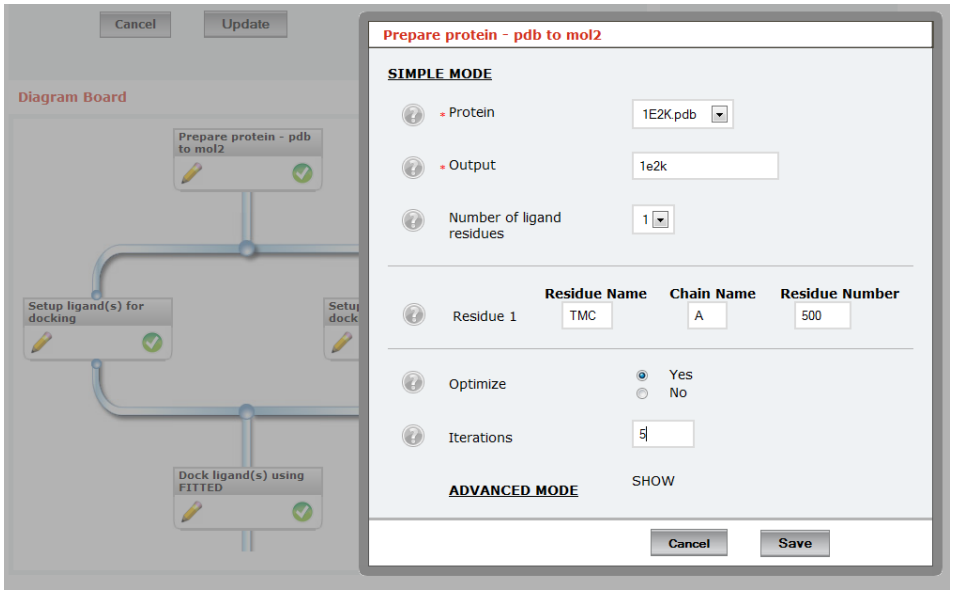
When setting this action in Job manager, a few boxes have to be filled. The number of proteins to be used in the flexible mode is defined in the first box. The protein file name is guessed by the interface as `Setup protein for docking` should be used in a preceding box.



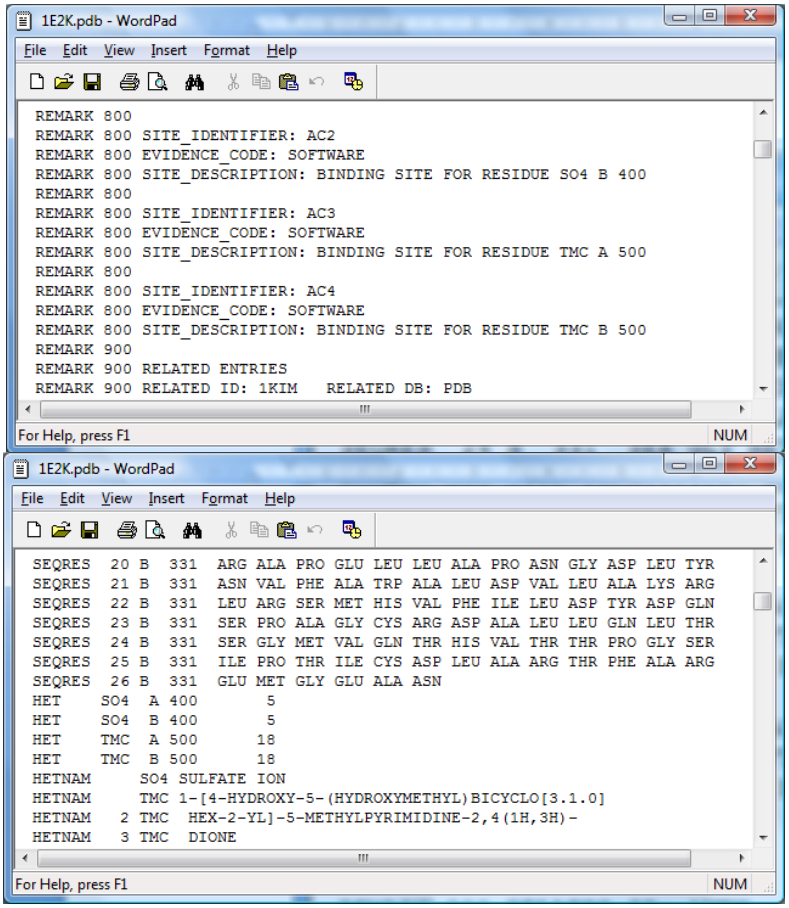
PREPARE PROTEIN – PDB TO MOL2

This action takes a protein pdb file, add hydrogens to the protein (according to the residue pKa), and to water oxygen atoms, searches for the optimal rotamers for asparagines, glutamines and histidines, reconstructs and optimizes missing side chains, extract the ligand

and finally outputs a protein file and a ligand file in mol2 format. This format is appropriate for most programs and required by most actions in FORECASTER as shown below.

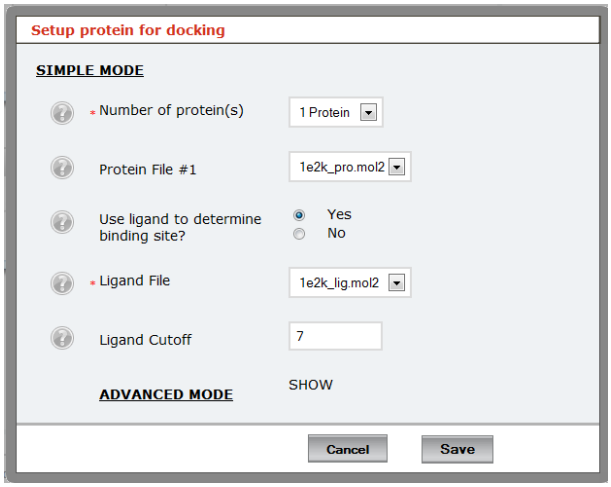


When setting this action in job manager, the user will have to provide the ligand name (i.e., TMC A 500 below). This information can be found in the pdb file.



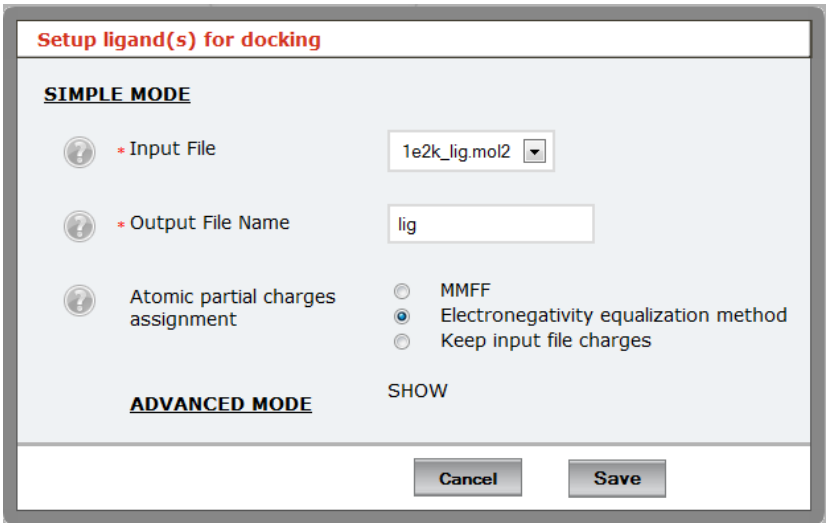
SETUP PROTEIN FOR DOCKING

This action is required to prepare the necessary files for FITTED to work. In the publications describing FITTED [1-3], we refer to the use of a program PROCESS which is the core of this action. It requires a protein mol2 file as input. Providing a ligand mol2 file helps identifying the binding site. The ligand file can be obtained by the action `Prepare protein – pdb to mol2`.



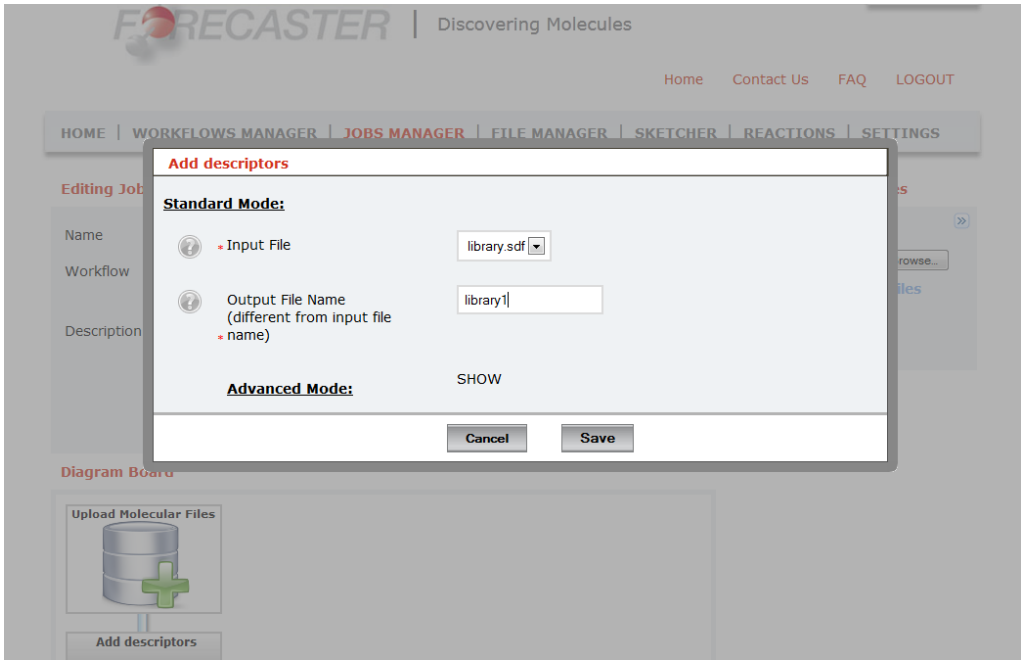
SETUP LIGAND(S) FOR DOCKING.

In the publications describing FITTED [1-3], we referred to SMART as a program to setup the ligand files. This action has been built from this program. The ligand must have hydrogen atoms added when given to this action (the user may use `Convert 2D to 3D` to add hydrogens or obtain this ligand from the `Prepare protein – pdb to mol2` action). sdf and mol2 formats are accepted and automatically detected by the action.



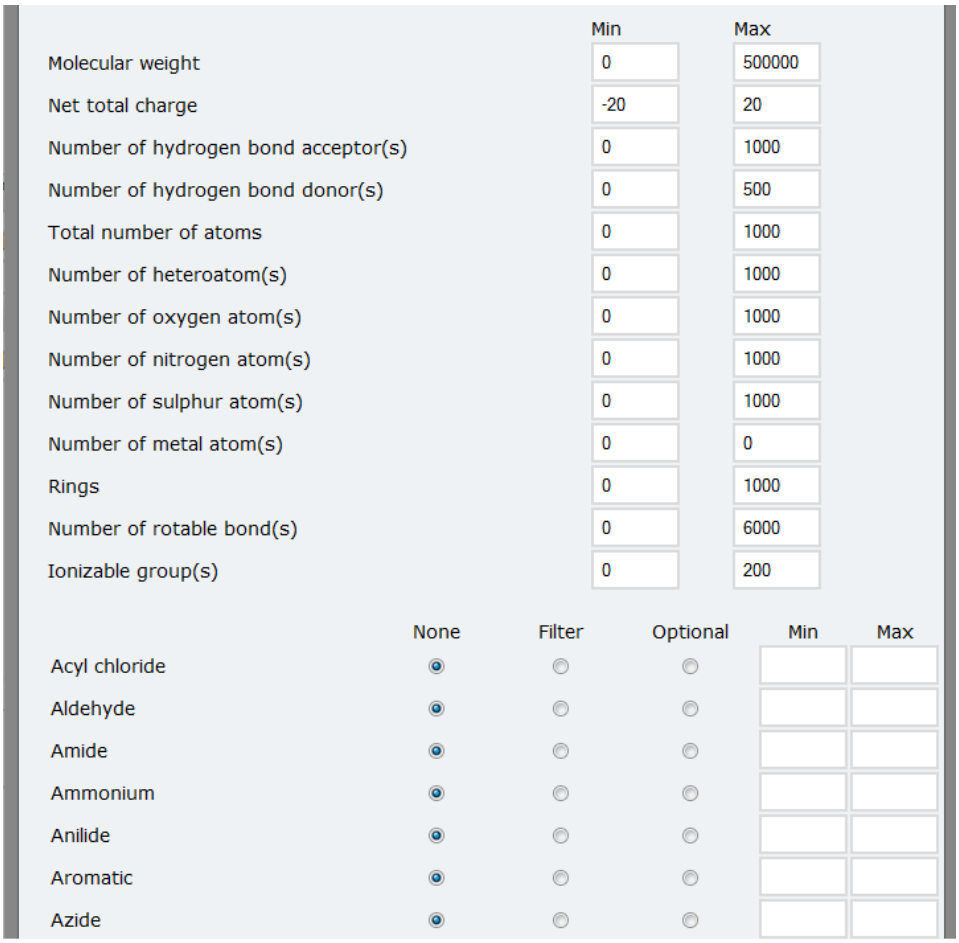
ADD DESCRIPTORS

As described in ref. 2, SMART can add descriptors that can be used for further filtering. This action has been built from this program. The ligand must be in 3D and have hydrogen atoms added when given to this action. Formats such as sdf and mol2 are accepted and automatically detected by the action. Within the `add descriptors` action, a number of descriptors and functional groups are automatically identified and added as a bitstring in the resulting mol2 output file.



FILTER BY DESCRIPTORS

`Filter by descriptors` uses the descriptors prepared using the action mentioned above and keep only the molecules with the appropriate properties.

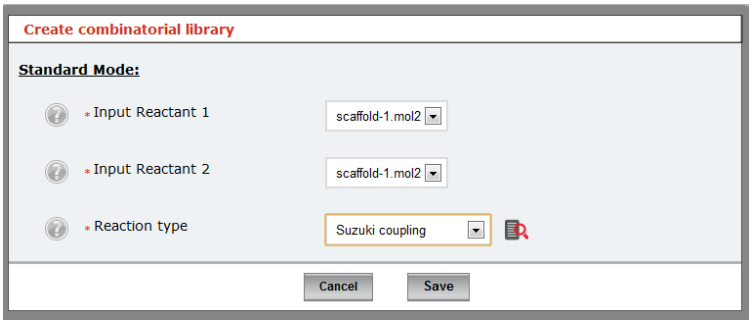


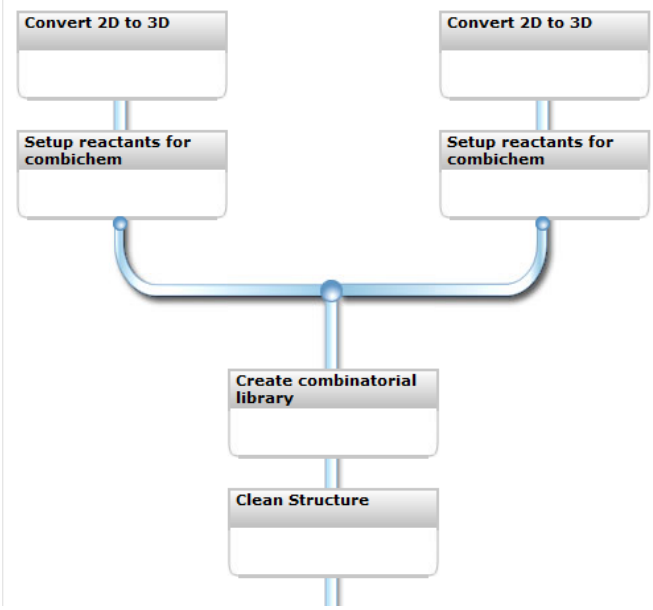
SETUP REACTANTS FOR COMBICHEM

This action prepares the reactants for further processing with the program REACT implemented in the action below. The dialog box is similar to the action `add descriptors`.

CREATE COMBINATORIAL LIBRARY

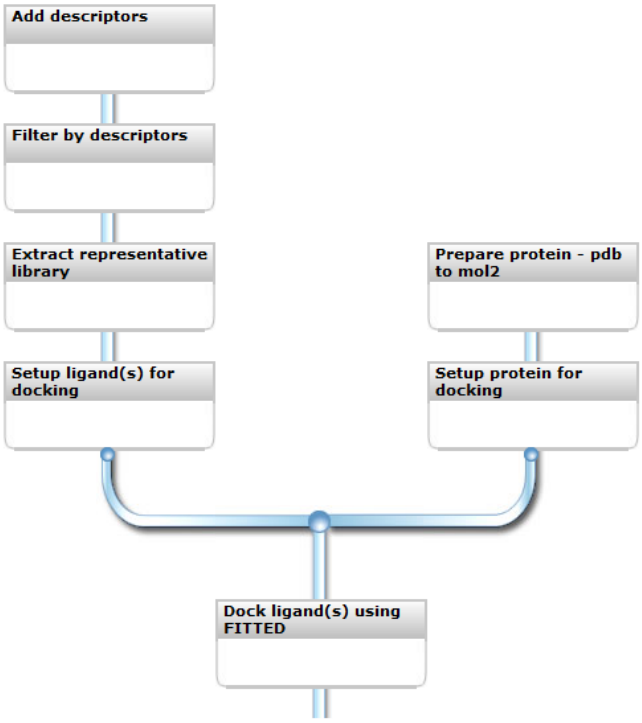
This action is based on our program REACT that takes two libraries of chemicals and a reaction scheme to prepare a combinatorial library. The reactants name are guessed by the interface as `Setup reactants for combichem` should be used in a preceding box as shown in the Figure below.

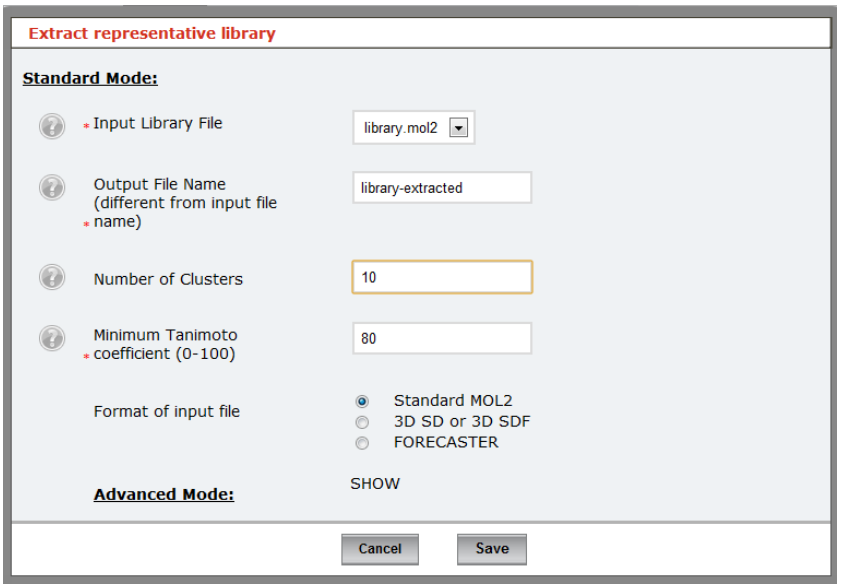




EXTRACT REPRESENTATIVE LIBRARY

This action takes a library, clusters the molecules by similarity and outputs a smaller library. The clustering allows the user to remove similar molecules (e.g., to keep the same diversity while reducing the number of molecules). The number of clusters is the maximum number of molecules extracted based on the value of the Tanimoto coefficient used.





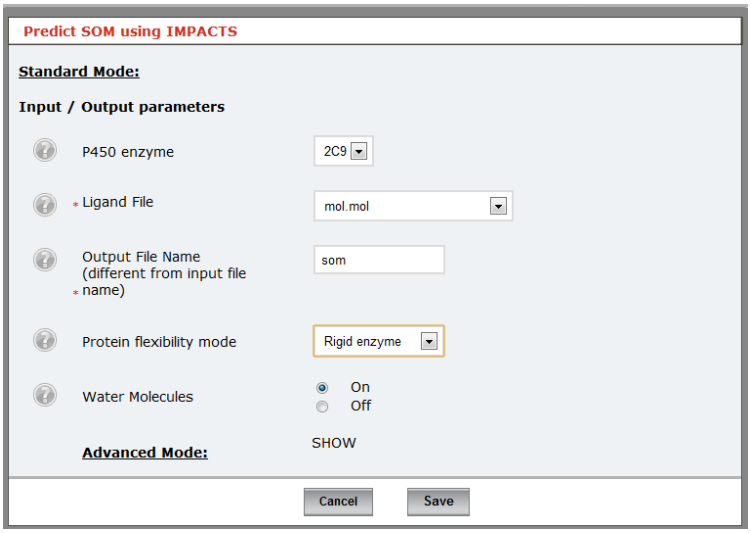
SEARCH FOR ANALOGUES

This action takes a library and a hit molecule and creates a new library of molecules from the library that are similar to the hit. This action is not yet fully implemented in workflows. However, it can be used alone.

PREDICT SOM USING IMPACTS

This action uses the IMPACTS program predicts the most likely site(s) of reaction and transition state (TS) structures of small molecules when reacting with the CYP heme as the activated iron-oxygen species. It uses a significantly modified version of our FITTED docking program to predict CYP-mediated metabolism of small molecules.

The action contains a single box, the molecule can be drawn in 2D within the sketcher (or uploaded). A 3D format is also supported (single or multi-mol2 files). The user can choose between the four implemented CYP 450 namely: 1A2, 2D6, 3C9 and 3A4.



CLEAN STRUCTURE

This action cleans (add missing hydrogens,…) and optimizes the input structure(s) through energy minimization (optional).

CONVERT 2D TO 3D

This action takes the input structure(s), adds missing hydrogens, converts from 2D to 3D and can optimize through energy minimization. The file formats compatible with this action are .mol and .sdf (2d).

FUNCTION LINKER

This is not a real action. This action adds a connector (i.e., a vertical bar) to the workflow.

STRUCTURE OPTIMIZATION

This action optimizes structures that have been setup for docking (conjugate gradient minimization).

CONVERT FILE FORMATS

This action uses babel to convert a structure from one format (i.e., mol2) to another one (i.e., sdf).

FORMAT COMPATIBILITY TABLE

The following table gives details about which file formats are expected and compatible for the actions within Forecaster.

Actions	Programs	Input file formats	Output file format
Dock ligand(s) using fitted	Fitted	mol2*	ligand: mol2, sdf protein#: mol2, pdb
Predict som using impacts	Impacts	mol2*	mol2
Make protein similar	Prepare	pdb	pdb
Prepare protein - pdb to mol2	Prepare	pdb	mol2
Reconstruct protein structures	Prepare	pdb	pdb
Setup protein for docking	Process	mol2	mol2*
Superpose protein structures	Prepare	pdb	pdb
Add descriptors	Smart	sdf3D, mol2	mol2*

Clean structure geometry	Smart/Minimize	mol2	mol2
Convert 2d to 3d	Prepare/Smart/Minimize	sdf[2D], mol	mol2
Create combinatorial library	React	mol2*	mol2
Extract representative library	Select	mol2	mol2
Filter by descriptors	Reduce	mol2*	mol2
Search for analogues	Select	mol2	mol2
Setup ligand(s) for docking	Smart	sdf[3D], mol2	mol2*
Setup reactants for combichem	Smart	sdf[3D], mol2	mol2*
Structure optimization	Smart/Minimize	mol2	mol2

* modified, not standard mol2 file; # when flexible docking is performed, the composite protein structure is provided in mol2 and pdb.

REFERENCES

1. Christopher R. Corbeil, Pablo Englebienne, and Nicolas Moitessier. Docking Ligands into Flexible and Solvated Macromolecules. 1. Development and Validation of FITTED 1.0. *J. Chem. Inf. Model.*, **2007**, *47* (2), pp 435–449.
2. Christopher R. Corbeil, Pablo Englebienne, Constantin G. Yannopoulos, Laval Chan, Sanjoy K. Das, Darius Bilimoria, Lucille L'Heureux and Nicolas Moitessier. Docking Ligands into Flexible and Solvated Macromolecules. 2. Development and Application of FITTED 1.5 to the Virtual Screening of Potential HCV Polymerase Inhibitors. *J. Chem. Inf. Model.*, **2008**, 48 (4), pp 902–909.
3. Christopher R. Corbeil and Nicolas Moitessier. Docking Ligands into Flexible and Solvated Macromolecules. 3. Impact of Input Ligand Conformation, Protein Flexibility, and Water Molecules on the Accuracy of Docking Programs. *J. Chem. Inf. Model.*, **2009**, 49 (4), pp 997–1009
4. Pablo Englebienne and Nicolas Moitessier. Docking Ligands into Flexible and Solvated Macromolecules. 4. Are Popular Scoring Functions Accurate for this Class of Proteins? *J. Chem. Inf. Model.*, **2009**, 49 (6), pp 1568–1580
5. Roxanne Kieltyka, Pablo Englebienne, Johans Fakhoury, Chantal Autexier, Nicolas Moitessier and Hanadi F. Sleiman. A Platinum Supramolecular Square as an Effective G-Quadruplex Binder and Telomerase Inhibitor. *J. Am. Chem. Soc.*, **2008**, 130 (31), pp 10040–10041
6. Pablo Englebienne and Nicolas Moitessier. Docking Ligands into Flexible and Solvated Macromolecules. 5. Force-Field-Based Prediction of Binding Affinities of Ligands to Proteins. *J. Chem. Inf. Model.*, **2009**, 49, pp 2564–2571
7. Janice Lawandi, Sylvestre Toumieux, Valentine Seyer, Philip Campbell, Sabine Thielges, Lucienne Juillerat-Jeanneret and Nicolas Moitessier. Constrained Peptidomimetics Reveal Detailed Geometric Requirements of Covalent Prolyl Oligopeptidase Inhibitors. *J. Med. Chem.*, **2009**, 52 (21), pp 6672–6684