|  |
| --- |
| Liverpool ChiroChem Ltd. |
| User Manual |
| For CFMProtocol Package |

|  |
| --- |
| Charmaine Chu  MAR-2019 |

Table of Contents

[Introduction 2](#_Toc93485167)

[Protocol package content 2](#_Toc93485168)

[Required programs 2](#_Toc93485169)

[Concepts of the protocol package 3](#_Toc93485170)

[Selection of protocol 4](#_Toc93485171)

[Accessing the protocols 4](#_Toc93485172)

[Protocol Explanation 5](#_Toc93485173)

[Library curation 5](#_Toc93485174)

[Required file(s): 5](#_Toc93485175)

[Chemical functionality mapping using geometrical alignment 7](#_Toc93485176)

[Required file(s): 7](#_Toc93485177)

[Template generation 10](#_Toc93485178)

[Required file(s): 10](#_Toc93485179)

[Chemical functionality mapping using substructure alignment 14](#_Toc93485180)

[Required file(s): 14](#_Toc93485181)

[Post protocol processing 17](#_Toc93485182)

[DataWarrior files 17](#_Toc93485183)

[PyMOL files 17](#_Toc93485184)

[Output Explanation 18](#_Toc93485185)

[DataWarrior 18](#_Toc93485186)

[PyMOL 21](#_Toc93485187)

[Output library SD files 21](#_Toc93485188)

# Introduction

This protocol package aims to align molecules within a library, analyse their chemical functionalities within the 3D space occupied by the whole library and output 3D visual results of the analysis.

This document aims to provide details of the files within the package, how to use the protocols within the package and explanation of the output.

For a quick overview of the protocol packages, please read ReadMe.txt

# Protocol package content

* UserManual.pdf (this document)
* ReadMe.txt
* Library curation.xml
* Chemical functionality mapping using geometrical alignment.xml
* Template generation.xml
* Rings\_in\_Drugs.sdf
* Alignment\_Structures.sdf
* Chemical functionality mapping using substructure alignment.xml
* OverallVectorMacro.dwam
* MoleculeVectorMacro.dwam

# Required programs

* Pipeline Pilot
* PyMOL (Open Source: <https://pymol.org/>)
* DataWarrior (Open Source: <http://www.openmolecules.org/datawarrior/>)
* Jmol (Open Source: <http://jmol.sourceforge.net/>)(Optional)

# Concepts of the protocol package

With this protocol packages, there are several key concepts that should be understood before using the protocols in order to understand the settings for and outputs of the protocol

#### Chemical functionality

The section of a molecule which belongs to a particular pharmacophore, e.g. H-bond donors and acceptors, charged atoms, aliphatic chains, aromatic rings

#### Library 3D space

The 3D space occupied by the molecules within the output library

#### Templates

Ring structures that exist as a substructures of a molecular library which the molecules can be aligned to

#### Molecule of Interest

A molecule which acts as the reference point when viewing the output library in 3D

When templates are used for alignment, molecule of interest would only appear in the output if it can be aligned against the templates

#### 3D visualization outputs

The outputs which allow the visualization of the chemical functionality vectors of the output library in 3D space (details see [Output Explanation](#_Output_Explanation))

# Selection of protocol

Molecule Library

Are the any template available?

[Chemical functionality mapping using geometrical alignment](#_Chemical_functionality_mapping_1)

[Chemical functionality mapping using substructure alignment](#_Chemical_functionality_mapping)

[Template generation](#_Template_generation)

Alignment of molecules geometrically

Assessment of vectors within the library

Alignment of molecules against templates

Assessment of vectors within the library

Preparation of templates

No

Yes

# Accessing the protocols

1. Open Pipeline Pilot
2. File> Open Protocols...> (Select desired protocol)

# Protocol Explanation

## Chemical functionality mapping using geometrical alignment

### Required file(s):

* Chemical functionality mapping using geometrical alignment.xml
* OverallVectorMacro.dwam
* MoleculeVectorMacro.dwam

Chemical functionality mapping using geometrical alignment calculate the visualisation for selected chemical functionalities of geometrically aligned molecules from the input.

[Post protocol processing](#_Post_protocol_processing) is required to obtain the 3D visualization outputs.

Chemical functionality mapping using geometrical alignment has the following settings:

|  |  |  |
| --- | --- | --- |
| Parameter | Options / Inputs | Explanation / Notes |
| InputLibrary | Location of the SD file containing the input library |  |
| MoleculeOfInterest | * Yes * No | Presence of Molecule of Interest in the 3D visualization ouputs  Default: No |
| Source | Location of the SD file containing the Molecule of Interest | The file should only contain one molecule |
| OutputDirectory | The folder directory to save all the output files | Recommended to create a new folder for each run |
| VisualisationType | * DataWarrior * DataWarrior&PyMOL | The type of visualisation output the protocol prepare for (require post protocol processing)  \*Inclusion of PyMOL output preparation increase runtime of the protocol  Default: DataWarrior&PyMOL |

|  |  |  |
| --- | --- | --- |
| Parameter | Options / Inputs | Explanation / Notes |
| OutputType |  | The output type for the PyMOL and DataWarrior output  Default: Everything |
| * Overall | Output files which create an overall chemical functionality visualisation of all aligned molecules |
| * Molecules | Output files which create chemical functionality visualisation for each molecule individually |
| * Everything | Output files for both Overall and Molecules |
| RemoveDuplicates | * True * False | Remove duplicates within the input library  Default: False |
| Sampling | * True * False | Sampling of the aligned molecules  Default: False |
| NumSamples | Any Integer | Maximum number of samples to select |
| Ionisation |  | Ionisation state of the molecules for analysis  Default: True - At specific pH and Original state |
| * True - All Possibilities | Return states of the molecules ionised at various pH (between 0-14) and discard the original states |
| * True - At Specific pH | Ionise the molecules at a specific pH (between 0-14) and discard the original states |
| * True - At Specific pH and Original state | Ionise the molecules at a specific pH (between 0-14) and keep the original states |
| * False | Molecules not ionised and the original states are kept |
| pH | Any value between 0.0 – 14.0 | pH value to use in the calculation of the ionisation state  Default: 7.4 |

|  |  |  |
| --- | --- | --- |
| Parameter | Options / Inputs | Explanation / Notes |
| Features | * H-Bond Donors * H-Bond Acceptors * Charged Atoms * Pi-systems * Hydrophobics | Chemical functionalities to analyse within the library  Default: All selected |

## Template generation

### Required file(s):

* Template generation.xml
* Alignment\_Structures.sdf
  + A list of ring structures to align desired templates against
* Rings\_in\_Drugs.sdf
  + Generated using supporting information of Rings in Drugs, Richard D. Taylor, Malcolm MacCoss, and Alastair D. G. Lawson, Journal of Medicinal Chemistry 2014 57 (14), 5845-5859
  + Aligned against Alignment\_Structures and Cn rotational axis information is added and processed

One or more templates are required to run [Chemical functionality mapping using substructure alignment](#_Chemical_functionality_mapping). In order for the protocol to have comparable outputs, the template(s) have to be aligned into a sensible orientation. Cn rotational axis information of each template are also desired for the optimal results.

Template generation is the protocol for accomplishing these requirements.

Template generation has 2 mode of actions:

|  |  |
| --- | --- |
| Mode | Explanation |
| TemplateAlignment | Process desired templates into correct orientation for Chemical functionality mapping using substructure alignment and Library comparison |
| CnProcessing | Process newly added Cn rotational axis information of within the file after TemplateAlignment  **(Optional)\*1** |

TemplateAlignment mode has the following settings:

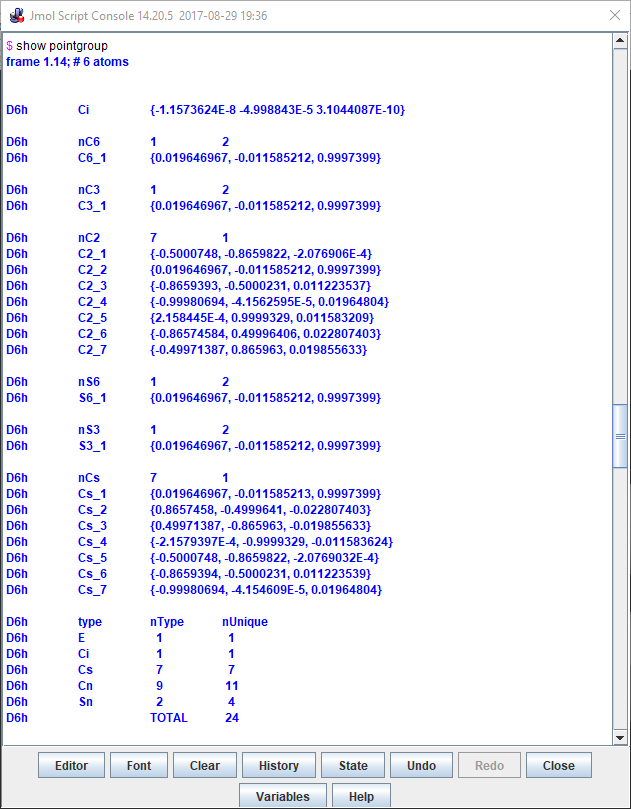
|  |  |  |
| --- | --- | --- |
| Parameter | Options / Inputs | Explanation / Notes |
| Templates | * SD * SMILES * SD&SMILES | Type of files(s) to input the desired template(s) into the protocol |
| SDTemplates | Location of the SD file containing the desired templates |  |
| SMILESStrings | SMILES strings of the desired templates | The strings have to be inputted as instructed within the parameter help text |

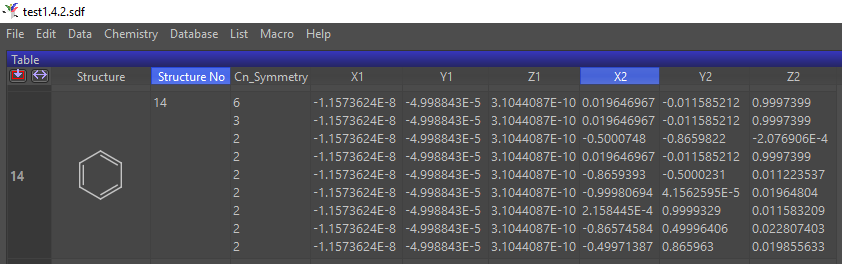
| Parameter | Options / Inputs | Explanation / Notes |
| --- | --- | --- |
| Method |  | Method of preparing the inputted template(s)  Default: SelectFromLibrary |
| * StandardAlignment | Align the inputted template(s) against a list of structures from Alignment\_Structures.sdf |
| * SelectFromLibrary | Select the inputted template(s) from the Template Library (Rings\_in\_Drugs.sdf)  Inputted templates not available from the Template Library are aligned via StandardAlignment |
| AlignmentTemplates | Location of Alignment\_Strucure.sdf |  |
| TemplateLibrary | Location of Rings\_in\_Drugs.sdf |  |
| AlignedTemplateDestination | Destination to save the outputting file | Where the aligned desire templates will be saved to  Ensure the file ends in **.sdf** |

**\*1** It is possible the outputted SD file at this stage does not contain all the Cn rotational axis information for each aligned templates. In such case, the missing Cn rotational axis information can be added to the SD file by the following steps:

1. Open Jmol
2. File> Open> (Aligned Template SD file)
3. File> Console...
4. Open DataWarrior
5. File> Open> (Aligned Template SD file)
6. Data> Add Empty Columns...
7. Add the following columns if not already available (column type= Text):
   * Cn\_Symmetry
   * X1
   * Y1
   * Z1
   * X2
   * Y2
   * Z2
8. Repeat the following steps for each template molecules:
   1. Input 'show pointgroup' within the Jmol Script Console
   2. Input the following values displayed within the Jmol Script Console (if available), each on a separated line:
      * x value of 'Cx\_n' under Cn\_Symmetry (where x is a number)
      * coordinates of 'Ci' or 'center' under X1, Y1 and Z1 respectively
      * coordinated of 'Cx\_n' under X2, Y2 and Z2 respectively

Example: Benzene





1. File> Save Special> SD-File...
2. Rename file (optional) and click 'Save'
3. Ensure the following settings and click 'OK':
   * Structure column: Structure
   * SD-file version: Version 3
   * Atom coordinates: 3D if available
   * Compound name column: <Automatic>

Once the file is saved, go back to the protocol. Set the Template generation Mode to CnProcessing and run with the following settings:

|  |  |  |
| --- | --- | --- |
| Parameter | Options / Inputs | Explanation / Notes |
| AlignedSDFile | Location of the SD file with added Cn rotational axis information | The SD file must have be through the above steps |
| ProcessedAlignedTemplates | Destination to save the outputting file | Where the aligned templates with added Cn rotational axis information will be saved to  Ensure the file ends in **.sdf** |

## Chemical functionality mapping using substructure alignment

### Required file(s):

* Chemical functionality mapping using substructure alignment.xml
* OverallVectorMacro.dwam
* MoleculeVectorMacro.dwam
* (SD file generated from [Template generation](#_Template_generation)) or Ring\_in\_Drugs.sdf

Chemical functionality mapping using substructure alignment calculate the visualisation for selected chemical functionalities of template aligned molecules from the input library.

[Post protocol processing](#_Post_protocol_processing) is required to obtain the 3D visualization outputs.

Chemical functionality mapping using substructure alignment has the following settings:

| Parameter | Options / Inputs | Explanation / Notes |
| --- | --- | --- |
| Template | Location of the SD file containing the aligned templates | The SD file needs to be generated through Template generation  or Rings\_in\_Drugs.sdf can be used directly |
| InputLibrary | Location of the SD file containing the input library |  |
| MoleculeOfInterest | * Yes * No | Presence of Molecule of Interest in the 3D visualization outputs (only if it can be aligned against the templates)  Default: No |
| Source | Location of the SD file containing the Molecule of Interest | The file should only contain one molecule |

|  |  |  |
| --- | --- | --- |
| Parameter | Options / Inputs | Explanation / Notes |
| VisualisationType | * DataWarrior * DataWarrior&PyMOL | The type of visualisation output the protocol prepare for (require post protocol processing)  \*Inclusion of PyMOL output preparation increase runtime of the protocol  Default: DataWarrior&PyMOL |
| OutputType |  | The output type for the PyMOL and DataWarrior output  Default: Everything |
| * Overall | Output files which create an overall chemical functionality visualisation of all aligned molecules |
| * Molecules | Output files which create chemical functionality visualisation for each molecule individually |
| * Everything | Output files for both Overall and Molecules |
| RemoveDuplicates | * True * False | Remove duplicates within the input library  Default: False |
| Sampling | * True * False | Sampling of the aligned molecules  Default: False |
| NumSamples | Any Integer | Maximum number of samples to select |

|  |  |  |
| --- | --- | --- |
| Parameter | Options / Inputs | Explanation / Notes |
| Ionisation |  | Ionisation state of the molecules for analysis  Default: True - At specific pH and Original state |
| * True - All Possibilities | Return states of the molecules ionised at various pH (between 0-14) and discard the original states |
| * True - At Specific pH | Ionise the molecules at a specific pH (between 0-14) and discard the original states |
| * True - At Specific pH and Original state | Ionise the molecules at a specific pH (between 0-14) and keep the original states |
| * False | Molecules not ionised and the original states are kept |
| pH | Any value between 0.0 – 14.0 | pH value to use in the calculation of the ionisation state  Default: 7.4 |
| Features | * H-Bond Donors * H-Bond Acceptors * Charged Atoms * Pi-systems * Hydrophobics | Chemical functionalities to analyse within the library  Default: All selected |

## Post protocol processing

Chemical functionality mapping using geometrical alignment, Chemical functionality mapping using substructure alignment and Library comparison with substructure alignment requires post protocol processing to obtain the final 3D visualisaion outputs within DataWarrior and/or PyMOL through the following steps:

### DataWarrior files

(If OutputType = Overall or Everything)

1. File> Open...> OverallVectors.csv
2. File> Open Special> Run Macro...> OverallVectorMacro.dwam
3. File> Save As...> OverallVector.dwar

(If OutputType = Molecule or Everything)

1. File> Open...>MoleculeVectors.sdf
2. File> Open Special> Run Macro...> MoleculeVectorMacro.dwam
3. File> Save As...>MoleculeVector.dwar

\*Error expected if more than 1 templates are available in the output library and/or molecule of interest is not provided\*

### PyMOL files

There are 2 methods for post protocol processing of the PyMOL output:

|  |  |
| --- | --- |
| Method |  |
| 1 | Simply open OverallScript.pml and/or MoleculeScript.pml within the output directory generated by the protocol  \*\*Ensure PyMOL is installed\*\* |
| 2 | (If OutputType = Overall or Everything)   * File> Run...> OverallScript.pml   (If OutputType = Molecule or Everything)   * File> Run...> MoleculeScript.pml   \*\*File> Reinitialize in between the runs if both files are to be ran\*\* |

Note: The post protocol processing of the PyMOL output need to be completed without any file path changes including renaming of the outputted folder

# Output Explanation

There are 4 outputs available at the end of Chemical functionality mapping using geometrical alignment, Chemical functionality mapping using substructure alignment and Library comparison with substructure alignment for the output library

## DataWarrior

Requires post protocol processing to obtain the final output

Depending on the settings of the original protocol, the following two outputs are available:

|  |  |
| --- | --- |
| File | Description |
| OverallVectors.dwar | Output of the library vectors within the library 3D space which accounts for the frequency of their appearance within the output library |
| MoleculeVectors.dwar | User end output of the library vectors within the library 3D space where they can be selected to find which molecules within the output library provide such vector |

Depending on the settings of the original protocol and the output file, the following data are included:

| Data | Explanation |
| --- | --- |
| (Molecule) Name | Type of structure the vector originates from   * Library Molecules * Origin * Template * MoleculeOfInterest |
| MoleculeNumber | Molecule identification number |
| StructureNumber | Molecule structure after Cn axis rotation and optional ionisation identification number |
| Group | Group which the data entry belongs   * Reference   + Reference point(s) for the visualisation such as the origin, molecule of interest or a single template * Template   + Templates used for alignment within the output library * VectorsFromTemplates   + Vectors within the output library originated from the alignment templates * VectorsFromLibraryMolecules   + Vectors within the output library not originated from the alignment templates but the library molecules |
| TemplateNumber | The template reference which the molecule aligned against |
| Type | Type of chemical functionalities:   * HBondDonorHeteroatom * HBondDonorHydrogen * HBondAcceptors * ChargedAtoms * Pi-systems * Hydrophobics   Or reference point label:   * Template * Origin * MoleculeOfInterest |

|  |  |
| --- | --- |
| Data | Explanation |
| Type2 | Type of Hydrophobics functionalities:   * HydrophobicTerminal (hydrophobic terminals) * HydrophobicChain (alphatic chains) * HydrophobicRing (hydrophobic rings) * HydrophobicArRing (hydrophobic aromatic rings) * HydrophobicS (hydrophobic Sulphur atoms)   Or reference point label:   * Template * Origin * MoleculeOfInterest |
| Type3 | Type of ChargedAtoms functionalities:   * PositiveAtom (positively charged atoms) * NegativeAtom (negatively charged atoms)   Or reference point label:   * Template * Origin * MoleculeOfInterest |
| Size | Size (number of atoms) of the given chemical functionality |
| X, Y, Z | Coordinates of the vector |
| Frequency | How many times the vector exist within the output library |

## PyMOL

Requires post protocol processing to obtain the final output

Depending on the settings of the original protocol, the following two outputs are available:

|  |  |
| --- | --- |
| File | Description |
| OverallSession.pse | Overall chemical functionality vector visualisation of all aligned molecules |
| MoleculeSession.pse | Chemical functionality vector visualisation for each molecule individually |

Depending on the settings of the original protocol and the output file, the following objects are available and can be turned on/off as desired:

| Object | Description |
| --- | --- |
| Molecules | Output library molecules, including the templates |
| MoleculeOfInterest | Molecule of interest |
| Template | The single template used for alignment within the output library or a single carbon atom representing the origin |
| HBondDonors | Arrows representing the H-Bond Donors where heteroatom→hydrogen |
| HBondAcceptors | Light blue spheres representing the H-Bond Acceptors |
| PositiveAtoms | Red spheres representing the positive atoms |
| NegativeAtoms | Blue spheres representing the negative atoms |
| AromaticRings | Yellow rings representing the aromatic rings |
| Hydrophobics | Grey spheres representing the hydrophobic groups |
| Surface | Coloured surfaces of the molecules |

## Output library SD files

There are 2 output library SD files:

|  |  |
| --- | --- |
| File | Description |
| Molecules.sdf | The SD file of the output library with templates |
| OutputLibrary.sdf | The SD file of the output library without the templates |