

## *SARvision* Application Note: Construction of Fragment Libraries and Reagent Libraries.

### *Background:*

Fragment based drug design is based on the premise that molecules of interest in medicinal chemistry can be created by the assembly of chemotypes that have been proven to be viable as drugs, and avoiding those that have been found to have undesirable characteristics. We will use *SARvision* to identify chemotypes of interest and organize vendor catalogs around fragment like molecules that contain those scaffolds. Finally, we show how to divide the molecules according to reactive groups that can be used for the synthesis of new compounds that contain relevant chemotypes.

In the current example, *SARvision* is used to identify the scaffolds found in a drug database of ~3800 molecules. Using a subset of these fragments, those that meet user specified size requirements, a database of commercially molecules (Sigma-Aldrich) is mined to identify molecules that represent these fragments. In this study the desired fragment library that will be employed in structure based-fragment assembly strategies that use in NMR, X-ray and computational studies to build molecules from relevant fragments.

---

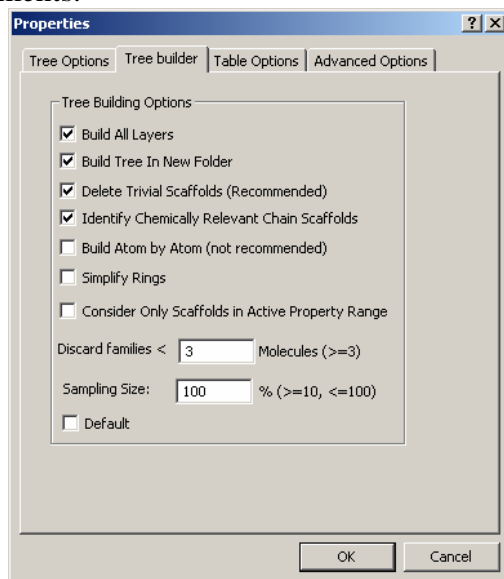
---

### *Application:*

#### *Generation of a medically relevant fragment library*

In the first step we will create a fragment library from the drug database of 3800 molecules using *SARvision*. The SD file containing the structures for these molecules is read into *SARvision*, and the scaffold generating algorithm is used to build a list of common scaffolds for this dataset. Subsetting the scaffolds based on size, removes all but the smallest fragments in the dataset, thus forming the basis a basis set of drug like fragments.

1. Launch a new instance of *SARvision*.
2. Under Main Menu:File->Import Dataset (\*.sdf) an chemistry structure file or library of molecules is selected for analysis from which to derive fragments. In the current example a dataset of ~3800 drug molecules (*drugDB301.sdf*) was used as medically relevant database. Any molecule set deemed relevant to a given project could be substituted here.
3. Under Main Menu:Tree->Scaffold Id Options: select the following options in the dialogue box. These options will cause the scaffold identification algorithm to build all layers of a hierarchical scaffold tree using both ring and chain fragments.



**Build All Layers:** Build the Complete Tree

**Build Tree in New Folder:** Creates a folder to contain the tree such that subsequent operations can be performed on this entire tree residing in this folder.

**Delete Trivial Scaffolds:** Deletes all trivial scaffolds. This includes fragments such as benzene, cyclohexane, cyclobutane and other fragments that have not heteroatoms or variations in bonds.

**Identify Chemically Relevant Chain Scaffolds:** This allows for the construction of scaffolds that contain only chain atoms and no rings. These must have some heteroatoms and/or variation in bonds. For example butane would not be considered.

**Discard families < 3 Molecules:** All scaffolds must have at least 3 representative molecules in the molecule dataset.

- Under Main Menu: **Tree->Identify Scaffolds** initiates the algorithm to identify scaffolds in the dataset of interest. This can take up to several hours to complete depending on the size and complexity of the dataset.
- When completed, right click on the folder that is generated by the algorithm and select **Scaffold Count**. In this example approximately **1770** unique scaffolds were identified.
- To remove the larger fragments, the scaffold tree was pruned to remove all fragments containing more than 2 rings. Right click on the folder and select **Prune Tree: 0-2 rings**. The end result for the drug database is now approximately **1390** scaffolds that contain at most 2 ring systems.
- The number of scaffolds was further reduced to only those that contained 12 atoms or less. Right click on the folder and **Prune Tree: 0-12 atoms**. This yielded **867** scaffolds or small molecule fragments.
- At this point the tree was converted from a hierarchical tree into a list by right click on the folder and selecting **Flatten Tree to List**. This step is optional, but does create a nonredundant list that can be exported to file, and imported into other applications as necessary (Main Menu: **Tree->Export To:sd file**).
- Optionally, Main Menu: **Tree->Sort Tree Nodes by: population** rank orders the scaffolds by the frequency by which they occur in this drug database.
- Finally, the workspace was saved for later use as **drug\_fragments.sarv** (Main Menu: **File->Save**).

## Fragment Database Derived from the Drug Database.

Structure	ID	Marketed	class	generic name	cas	molregno
	1		Antidepressant, Anxiolytic	EBALZOTAN [INN]	149494-37-1	
	10	1997	COGNITION ENHANCER	DONEPEZIL [INN]	120014-06-4	
	25	1992	ANTINEOPLASTIC	PENTOSTATIN [U,INN]	63677-95-2	
	31	1986	BRONCHODILATOR ANTI-ASTHMATIC	FORMOTEROL [INN]	73573-87-2	
	48	1996	ophthalmic	BRIMONIDINE [U,INN]	59003-90-4	
	49	1996	ANTIPARKINSONIAN	ROPINIROLE [INN]	91374-21-9	
	94		Antineoplastic, Adrenocortical suppress...	AMINOGLUTETHIMIDE [U,INN]	125-84-8	

Now that we have a medically relevant fragment library, a database of commercially available compounds was compiled. In this case, chemistry structure files (sd files) available from Sigma-Aldrich were used. In

the following steps, the molecules are loaded into a **SARvision** workspace, degeneracy in the database is removed, and any proteins and other catalog items without structures were removed.

1. Either select Main Menu: **File->New** or launch a new instance of **SARvision**.
2. The sd files were read in sequentially to obtain a total of ~56,000 rows of data in the table. Under Main Menu: **File->Import Molecule Dataset (sdf)**, the vendor files (*Aldrich.sdf*, *Sigma.sdf*, and *Fluka.sdf*) were each read into the program. Columns containing catalog numbers can be formatted by right clicking on the column header and selecting **Table->Format Column**: decimal points = 0, to display this data as integers.

Because Sigma-Aldrich is the combination of Sigma, Aldrich and Fluka, there is significant redundancy in the database. Using the remove duplicates function in the redundancy was removed to yield ~30,000 molecules. Under Main Menu: **Tools->Remove Duplicate: combine data** to build a nondegenerate database.

3. To further clean the database, molecular weight was calculated for the dataset (**Tools->calculate Molecular Properties: Molecular weight**). The table was sorted by MW (right click on column header MW and selecting **Sort Column** to sort all rows from smallest to largest molecules). All rows greater than 500 MW were removed (shift select all rows from 500MW to the end of the molecule table and right click on the row header and select **Delete row(s)**). All molecules with no structure or zero molecular weight were similarly deleted (first 75 rows). While in this case we chose to use only molecules with MW less than 500, in a similar manner it would be possible to implement the “rule of 3” and limit by MW, logP and number of hydrogen bond donors or acceptors.
4. Note that columns can be resized as necessary by clicking on the dividers between the column headers and dragging them to the desired size.
5. The data was saved to file (Main Menu: **File->save: CommercialLibrary.sarv**) for future work. Alternatively, structures and data can be exported to sdf (Main Menu: **Table->Export to SD file: Commercial library.sdf**).

---

---

### *Comparing fragment libraries:*

To compare libraries, the *CommercialLibrary.sarv* file was opened in **SARvision** and the fragments were imported as scaffolds from the *drug\_fragments.sarv*. For each fragment, molecules can be selected from the table that fit the scaffold.

1. The fragment tree was imported from the drug-fragment library file and the table was reconciled to the fragment list (Main Menu: **File->Import: Tree From Workspace (\*.sarv): drug\_fragments.sarv**). Below each fragment in the new fragment folder is the number of molecules in the dataset for which this fragment is a substructure. Note that fragments in the folder can be sorted by their respective populations, from most populated to least (Main Menu: **Tree->Sort Tree Nodes by: Population...deselect increasing**).
2. The molecules were sorted by molecular weight such that for each fragment selection, the smallest molecules appear at the top of table. To sort table by molecular weight, *right click* on the **MW** column header and select **Sort Column**. *Double clicking* on any fragment in the scaffold tree displays those molecules for which it is a substructure in the molecule table.

**Fragment library (left side) with a selected fragment highlighted. Molecule table (right side) displaying molecules for which the selected fragment is a substructure.**

Structure	MW	CAT_NO	CAT_NO_FULL	Source	COPYRIGHT	Selected
<chem>NC(=O)c1ccccc1</chem>	121.137040	425443 12370 135528 150762 491037 82009 389337	B2009	Aldrich FLUKA Aldrich Aldrich Sigma Aldrich	(C) 2001 SIGMA-ALDRICH CO.	1
<chem>NC(=O)c1ccc(O)cc1</chem>	121.137040	A8620	A8620	Sigma	(C) 2001 SIGMA-ALDRICH CO.	1
<chem>OC(=O)c1ccccc1</chem>	122.121700	460002 468074 60065 548932 277746 217158 12366 12353 12960 87521 12349 83250 83375 87690 109479 87697 201162 9896 290009 9896 427603 431888 491861 88027 242381 62443 87359 109169 71301 227277 71296 103334	87521 83250 83375 88027	Aldrich Aldrich FLUKA Aldrich Aldrich FLUKA FLUKA FLUKA Sigma FLUKA Sigma Sigma Sigma FLUKA Aldrich FLUKA Aldrich Aldrich Sigma Aldrich FLUKA FLUKA Aldrich FLUKA Aldrich	(C) 2001 SIGMA-ALDRICH CO.	1
<chem>OC(=O)c1ccc(O)cc1</chem>	122.121700	84162 04160 S366		FLUKA FLUKA Aldrich	(C) 2001 SIGMA-ALDRICH CO.	

- At this point in the data analysis, it is useful to start selecting molecules to build a final library that can be purchased from the vendors. In this example, a column was added to the table labeled **Selected** (Main Menu: **Table->Add Column->Data: Selected**). For each molecule in the table that is of interest, a number can be entered into the cell to differentiate it from the others. By *double clicking* on different fragments and selecting representative molecules this way, a library can be rapidly assembled.
- Upon completion of molecule selection, the table can be filtered for only those molecules that contain a numeric entry in this new column: **Selected** (**Data->Set Active Property Data: lower = 1, upper = 1**). **Dialog to select a property range to filter the molecular table. In this example, col:Selected is filtered for all rows that contain a '1'.**

Then by *clicking* the **ALL** button on the toolbar at the top, a complete set of only those molecules that have been selected (contain a numeric entry in the **Selected** column) will be displayed. This table of molecules and data can in turn be exported to Word, Excel or sd file.

Reagent libraries were similarly constructed for reagents that have specific reactive chemical motifs. In this example, the chemical motif was added directly to the scaffold tree and the fragment library was imported and moved under the motif. In this case, **SARvision** first identify all molecules with the desired chemical motif and then distributes these molecules to its children nodes, the fragments.

- The reactive chemical motifs: COOH, NH<sub>2</sub> and N=C=O were drawn into the scaffold tree (right click on scaffold tree background and **Add Scaffold**).
- The fragment library was imported for each chemical motif from *drug\_fragments.sarv* (Main Menu: **File->Import Tree from Workspace**). By dragging the folder containing the scaffolds under the each reagent reestablishes the substructure relationships between the fragments and the molecules. In the case shown below, the table displays molecules that first contain an amino group (-NH<sub>2</sub>) and second contain the selected fragment. In this way, a chemist can select medicinally relevant reagents drawn from fragments from the drug database.

### Amine reagent library organized by fragments found in the drug database.

The screenshot shows the SARvision 2.7 interface. On the left, a scaffold tree is visible, organized into folders for 'Carboxylic Acid Reagents' (4177) and 'Amine Reagents' (3699). Under 'Amine Reagents', there are several 'ScaffoldTree' folders (e.g., ScaffoldTree1 1750, ScaffoldTree3 1705, ScaffoldTree6 1297). On the right, a data table displays a list of molecules. The table has the following columns: Structure, MW, CAT\_NO, CAT\_NO\_FULL, Source, COPYRIGHT, and Selected. The table contains 11 rows of data, each representing a molecule with its corresponding chemical structure and metadata.

Structure	MW	CAT_NO	CAT_NO_FULL	Source	COPYRIGHT	Selected
		241261 50056 50052		Aldrich FLUKA FLUKA		
	75.110240	9280 110248		FLUKA Aldrich	(C) 2001 SIGMA-ALDRICH CO.	
	75.110240	5230 A4916 297662	A4916	FLUKA Sigma Aldrich	(C) 2001 SIGMA-ALDRICH CO.	
	75.110240	239844 9290 452572 A76400		Aldrich FLUKA Aldrich Aldrich	(C) 2001 SIGMA-ALDRICH CO.	
	75.110240	5225 A76206		FLUKA Aldrich	(C) 2001 SIGMA-ALDRICH CO.	
	75.110240	9283 A1531 238856	A1531	FLUKA Sigma Aldrich	(C) 2001 SIGMA-ALDRICH CO.	
	75.110240	64740 241067 143693		FLUKA Aldrich Aldrich	(C) 2001 SIGMA-ALDRICH CO.	
	75.110240	9281 238864		FLUKA Aldrich	(C) 2001 SIGMA-ALDRICH CO.	
	75.110240	406694 A2005 406686 406678 406651 406643 192171 406635 406627 6340	A2005	Aldrich Sigma Aldrich Aldrich Aldrich Aldrich Aldrich Aldrich Aldrich FLUKA	(C) 2001 SIGMA-ALDRICH CO.	

This final SARvision file is available for SV2.6 and SV2.7 file format:

***Fragments\_Reagents\_CommercialLibrary.sarv.***