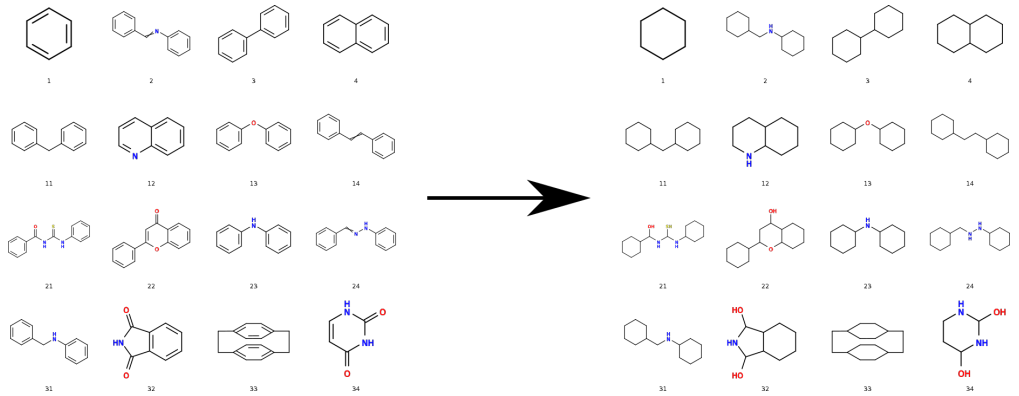
1 Background

The Bemis-Murcko scaffold[1] provided by `DataWarrior`[2] retains information about bond order and chirality. There are instances where retaining only information about which atoms are connected with each other, but not their bond order is desired. This corresponds to the assumption 'there are only single bonds'. `DataWarrior` equally offers the export of Bemis-Murcko skeleton, however this simplifies e.g. the scaffold about an imidazole into one of cyclopentane.



2 Typical use

The script is to be used on the CLI of Python3 with the mandatory parameter about listing file containing the SMILES to work with:

```
python saturate_MurckoScaffolds.py [listing_file.txt]
```

to generate `saturated_Murcko_scaffold.csv` as permanent record. The script equally works with the legacy of Python2.17, too.

3 Example

For a collection of organic materials, the Bemis-Murcko scaffolds was extracted with `DataWarrior` (release 5.0.0 for Linux) as listing `Murcko_scaffolds_with_bond_order.txt`. The "saturation" was obtained by

```
python saturate_MurckoScaffolds.py Murcko_scaffolds_with_bond_order.txt
```

yielding `saturated_Murcko_scaffold.csv`. The affect of this operation on the SMILES strings deposit in the listings are easy to see (fig. 1).

For a visual survey about the scaffolds, `openbabel`[3] allows a quick check on the CLI by writing a `.svg`, here as:

[1] Bemis GW, Murcko MA *J. Med. Chem.* 1996, **39**, 2887-2893, doi 10.1021/jm9602928

[2] Sander T, Freyss J, von Korff M, Rufener C, *J. Chem. Inf. Model.* 2015, **55**, 460-473, doi 10.1021/ci500588j. The program, (c) 2002–2019 by Idorsia Pharmaceuticals Ltd., is freely available under `http://www.openmolecules.org`.

[3] www.openbabel.org. This outline is based on release 2.4.1 (Nov 12, 2018) provided by Linux Xubuntu 18.04.2 LTS.

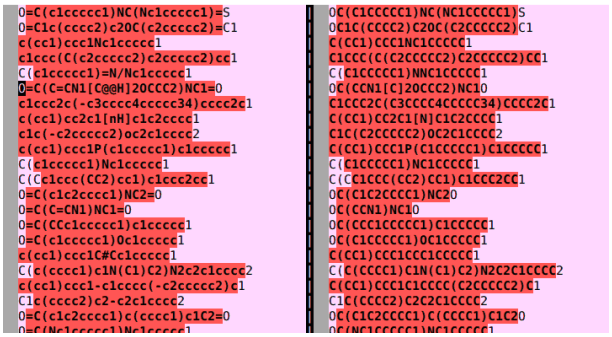


Figure 1: DiffView of the SMILES strings of a Murcko scaffold *prior* (left hand column) or *after* an «artifical saturation» (right hand column). Note the removal of explicit bond order indicators, e.g. double bond (equality sign), triple bond bond (octohorpe), or about implicit aromatization (lower case -> upper case for atoms of carbon, nitrogen (depicted); oxygen or sulfur (not depicted). At the same time, stereochemical indicators are removed, too (e.g., at-signs).

```
obabel -ismi Murcko_scaffolds_with_bond_order.txt -O
↪ Murcko_scaffolds_with_bond_order.svg -xc10 -xr12 -xl --addinindex
```

This formats the output as an array of 10 columns (`-xc10`) by 12 rows (`-xr12`) with a grid (`-xl`), where the entries are labeled in order of their appearance in the input file (`--addinindex`). If using the GUI of `openbabel` instead of the CLI, the later optional parameter is called `Append input index to title`.[4]

The `.svg` may be post-processed further to yield a `.pdf` or a `.png`, for example `cairosvg`[5] by a call of

```
cairopdf Murcko_scaffold_with_bond_order.svg -o Murcko_scaffold_with_bond_order.pdf
```

Alternatively, the `.svg` may be processed in programs like `inkscape`.[6]

As desired, this "artificial saturation" of double / triple / aromatic bonds retains the information about which atoms are directly connected with each other. `openbabel`'s algorithm to display the molecular structures deals surprisingly well even with sometimes complicated motifs (e.g., the scaffold of cyclophane [entry #33], sparteine [#38], or adamantane [#50]).

[4] By default, `openbabel` attributes element specific colors. Especially if the output is print black-and-white, the labels about atoms like hydrogen, silicon, sulfur, phosphor, then might be barely intelligible, especially at low scale. It is possible to toggle off the element-colors by adding `-xu` as additional parameter of `openbabel`'s conversion after the definition of the output file.

[5] `https://cairosvg.org/`

[6] `https://inkscape.org`