Subject: Toggle-off absolute configuration while determining Murcko scaffold
Posted by nbehrnd on Wed, 21 Aug 2019 12:00:58 GMT
View Forum Message <> Reply to Message

Dear Thomas,

using both DW as well as third-party python module rdkit to determine the Murcko scaffolds of molecules including those with a stereogenic center, I noticed DataWarrior (DW) retains the stereochemical information in the trimmed fragment. Reading the same SMILES string as DW (5.0.0), rdkit (version 2019.1 with Python2) however trims this information off.

Question: Is there an option to instruct DW equally to 'forget' about this piece of information when writing the SMILES string?

Pristine SMILES string in question used in both programs:
C(=O)(C)O[C@H]1[C@H]([C@H](n2c3c(c(ncn3)N)nc2)O[C@@H]1CO)O

DW's output SMILES string about the Murcko scaffold:
C(C1)CO[C@H]1n1c2ncncc2nc1

MWE for processing with rdkit:

```
from rdkit.Chem.Scaffolds import MurckoScaffold
from rdkit.Chem import AllChem

source = 'C(=O)(C)O[C@H]1[C@H]([C@H](n2c3c(c(ncn3)N)nc2)O[C@@H]1CO)O'
mol = Chem.MolFromSmiles(source)
core = MurckoScaffold.GetScaffoldForMol(mol)
print(Chem.MolToSmiles(core))

>>> c1ncc2ncn(C3CCCO3)c2n1
```

which constitutionally is about the same molecular structure.

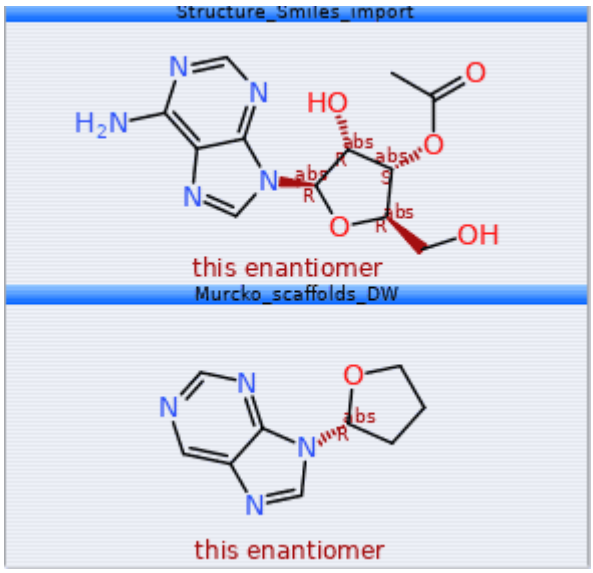
This may be more than a cosmetic issue, because the search for the Murcko scaffold of

Brc1ccc([C@@]2(CC(=O)CCC2)CN(=O)=O)cc1

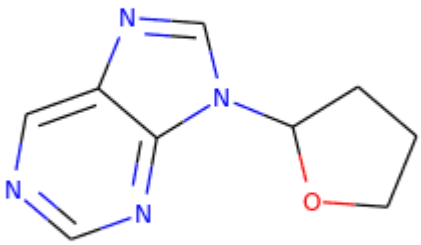
actually leads DW to invert the absolute configuration from initial R to now S (expressed by the lost of one @ in string O=C(CCC1)C[C@H]1c1ccccc1).

File Attachments

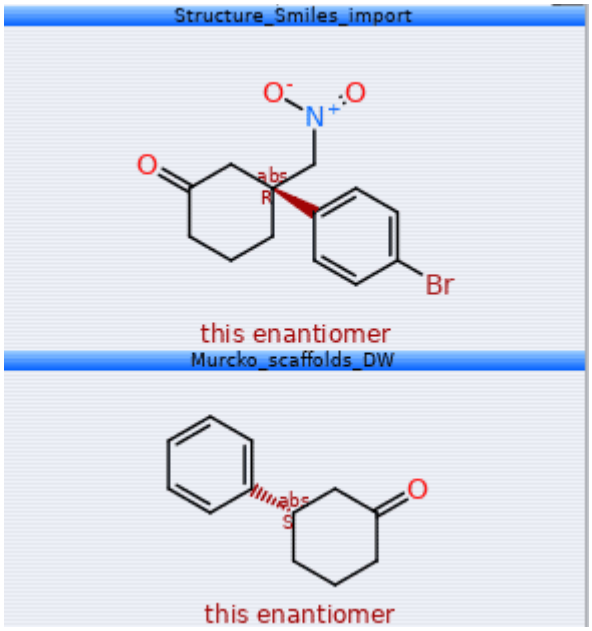
1) DWclip.png, downloaded 1311 times



2) rdkit_ACADOS.png, downloaded 1204 times



3) ABEGUL_DW_inversion.png, downloaded 1211 times


Structure_Smiles_import
O
N
O
O
abs
R
Br
this enantiomer
Murcko_scaffolds_DW
O
abs
S
this enantiomer