**Classes for 2D Molecular Drawing in RDKit**

**Introduction**

This document describes some new classes for drawing molecules in C++ RDKit programs.

Up until at least 2014\_09\_2, drawing in a C++ program involved using function DrawMol in

$RDBASE/Code/GraphMol/MolDrawing/MolDrawing.h, which encoded drawing commands into a bespoke code expressed as a vector of ints. The drawings were not complete, not least because they didn't show chirality. They also just described which lines and text to draw where. It was very difficult to use the drawing code in, for example, Qt, to create interactive drawings where atoms could easily be selected by a user. That was the main motivation for these classes - the ability to locate atoms easily in the reference frame of the drawing canvas for annotation, selection etc.

There is a completely separate and more complete set of drawing modules written for the Python bindings.

The design philosophy for these classes mimics that of the original OEDepict library from OpenEye Scientific Software ([www.eyesopen.com](http://www.eyesopen.com/)) and is also similar to that using in the RDKit Python bindings.

**The MolDraw2D class**

This is a virtual base class containing the functions necessary for drawing an arbitrary molecule. It contains several pure virtual functions, such as drawLine and drawChar which must be defined in derived concrete classes for drawing using a particular drawing package such as Qt or Cairo. The molecule is drawn using calls to these functions. Drawing is done by calling the function DrawMolecule, passing in a constant reference to an ROMol and an optional vector of ints giving atoms to be highlighted, or a map of triples of floats indexed by atom index. The ROMol must already have 2D coordinates.

As well as functions for drawing the molecule, MolDraw2D contains functions getDrawCoords and getAtomCoords which transform coordinates from the molecule reference frame to the drawing canvas reference frame and *vice versa*.

**The MolDraw2DQt class**

This is a concrete class derived from MolDraw2D which draws into one of Qt's QPainter objects. As well as being useful in its own right, it demonstrates how to use the new drawing system.

**Program RDKitSV**

This is a simple Qt program that demonstrates the use of the new classes in an application. It is intended to help with the creation and debugging of SMARTS definitions. A file of molecules, in SMILES or SDF format, is read and displayed. SMARTS can be read from file or created within the application, and matched against the molecules. Two display panels will then show the molecules for which the SMARTS matched, and those for which they didn't. In the former case, the matching atoms are highlighted, and, so as to demonstrate the use of getDrawCoords, a small orange rectangle is also drawn over each matched atom. The class **MolDisplay2DWidget** also shows how a mousePressEvent is used to select an atom. MolDisplay2DWidget contains an instance of a MolDraw2DQt, which is used to draw the molecule into a QPainter created in MolDisplay2DWidget::paintEvent.