The RDKit: History and Status

Gregory Landrum
NIBR IT
Novartis Institutes for BioMedical Research
Basel
RDKit UGM 2012, London
Acknowledgements

- **NIBR**
  - Tom Digby (Legal)
  - Richard Lewis (GDC)
  - Remy Evard (NIBR IT)
  - Andy Palmer (NIBR IT)
  - Nik Stiefl (GDC)
  - Peter Gedeck (GDC)
  - Manuel Schwarze (NIBR IT)
  - Eddie Cao (NIBR IT)

- **The RDKit Open-Source Community**

- **knime.com**
  - Michael Berthold
  - Bernd Wiswedel
  - Thorsten Meinl

- **PostgreSQL cartridge:**
  - Michael Stonebraker
  - Oleg Bartunov
  - Teodor Sigaev
  - Pavel Velikhov

- **Distributions:**
  - Gianluca Sforza (Fedora)
  - Michael Banck (Debian)
Overview

- History
- Where are we today?
- What next?
On 13/09/2007, Greg Landrum <greg.landrum@gmail.com> wrote:
> On 9/13/07, Noel O'Boyle <baoilleach@gmail.com> wrote:
> (4) Why haven't you publicised RDKit, if you don't mind me asking? For
> example, there is an excellent (if I do say so myself) website called
> Linux4Chemistry which lists the excellent (if you do say so yourself)
> YaEHMOP. Also there's the CCL mailing list. I only found RDKit because
> of trawling through the SF software map. Is this, um, shyness,
> intentional?

There are many components to the answer to this question. Some are:
1) Promotion isn't something I enjoy or am particularly good at.
2) I'm kind of afraid of having more users. I do a lot of this as a
free-time project and I'm afraid of spending all my time answering
questions. This is, of course, a bit stupid because if the whole open
source thing works then other people will pitch in and help with those
questions. For that to happen I need those other people as users,
which requires that I find them, which... it's a Catch 22
History and milestones

- 2000-2006: initial development work at Rational Discovery
- 2006: code open sourced and released on sourceforge.net
- 2007: First NIBR contribution (chemical reaction handling); Noel discovers the RDKit (=first rdkit-discuss post?)
- 2008: first POC of Java wrapper; Mac support added; SLN and Mol2 parsers;
- 2009: Morgan fingerprints; switch to cmake; switch to VF2 for SSS
- 2010: PostgreSQL cartridge; First iteration of the KNIME nodes; $RDBASE/Contrib appears; SaltRemover and FunctionalGroups code
- 2011: New Java wrappers; more functionality moved to C++; InChI support; Avalontools integration
- 2012: Speed improvements; MCS implementation; ??
Overview

- History
- Where are we today?
- What next?
RDKit: What is it?

- Python (2.x), Java, and C++ toolkit for cheminformatics
  - Core data structures and algorithms in C++
  - Python wrapper generated using Boost.Python
  - Java wrapper generated with SWIG

- Functionality:
  - 2D and 3D molecular operations
  - Descriptor generation for machine learning
  - Molecular database cartridge
  - Supports Mac/Windows/Linux

- History:
  - 2000-2006: Developed and used at Rational Discovery for building predictive models for ADME, Tox, biological activity
  - June 2006: Open-source (BSD license) release of software, Rational Discovery shuts down
  - to present: Open-source development continues, use within Novartis, contributions from Novartis back to open-source version
The RDKit “ecosystem”

Exact same algorithms/implementations accessible from many different endpoints
What can you do with it?

A laundry list

- Input/Output: SMILES/SMARTS, SDF, TDT, SLN\(^1\), Corina mol2\(^1\)
- “Cheminformatics”:
  - Substructure searching
  - Canonical SMILES
  - Chirality support (i.e. R/S or E/Z labeling)
  - Chemical transformations (e.g. remove matching substructures)
  - Chemical reactions
  - Molecular serialization (e.g. mol <-> text)
- 2D depiction, including constrained depiction
- 2D->3D conversion/conformational analysis via distance geometry
- UFF implementation for cleaning up structures
- Fingerprinting:
  Daylight-like, atom pairs, topological torsions, Morgan algorithm, “MACCS keys”, etc.
- Similarity/diversity picking (including fuzzy similarity)
- 2D pharmacophores\(^1\)
- Gasteiger-Marsili charges
- Hierarchical subgraph/fragment analysis
- RECAP and BRICS implementations
- Multi-molecule maximum common substructure\(^2\)

\(^{1}\) functional, but not great implementations

\(^{2}\) Contribution from A. Dalke
What can you do with it?

A laundry list, cntd

- Feature maps
- Shape-based similarity
- Molecule-molecule alignment
- Shape-based alignment (subshape alignment) ¹
- Integration with PyMOL for 3D visualization
- Database integration
- Molecular descriptor library:
  - Topological (κ3, Balaban J, etc.)
  - Electrotopological state (Estate)
  - clogP, MR (Wildman and Crippen approach)
  - “MOE like” VSA descriptors
  - Feature-map vectors
- Machine Learning:
  - Clustering (hierarchical)
  - Information theory (Shannon entropy, information gain, etc.)
  - Decision trees, naïve Bayes¹, kNN¹
  - Bagging, random forests
  - Infrastructure (data splitting, shuffling, enrichment plots, serializable models, etc.)

¹ functional, but not great implementations
RDKit: Where is it?

- Web page: http://www.rdkit.org
- Sourceforge: svn repository, bug tracker, mailing lists, downloads
  - http://sourceforge.net/projects/rdkit
- Google code: wiki, downloads
  - http://code.google.com/p/rdkit/
- Releases: quarterly
- Licensing: new BSD
- Documentation:
  - HTML/PDF “Getting Started” documentation
  - in-code docs extracted by either doxygen (C++) or epydoc (python)
- Getting help:
  - Check the wiki and “Getting Started” document
  - The rdkit-discuss mailing list
RDKit: Documentation?

The documentation:

Built using Python’s standard docs tool: Sphinx
Sample section from introductory docs:

**Reading and Writing Molecules**

**Reading single molecules**
The majority of the basic molecular functionality is found in module `rdkit.Chem`:

```python
>>> from rdkit import Chem
```

Individual molecules can be constructed using a variety of approaches:

```python
>>> m = Chem.MolFromSmiles('Cc1ccc1')
>>> m = Chem.MolFromMolFile('data/input.mol')
>>> stringWithMolData = file('data/input.mol', 'r').read()
>>> m = Chem.MolFromMolBlock(stringWithMolData)
```

All of these functions return a `Mol` object on success:

```python
>>> m
<rdkit.Chem.rdchem.Mol object at 0x...>
```

Note: docs that include python code snippets are tested.
RDKit: Documentation?

The documents that come out of the tutorials here are going to be another good source of information.
RDKit: Who is using it?

- Hard to say with any certainty
- ~500 downloads of each new version
- Active contributors to the mailing list from:
  - Big pharma
  - Small pharma/biotech
  - Software/Services
  - Academia
- Starting to see contributions coming from the community (wiki pages, code patches, changes to the build system, etc.) as well as active use in other systems.
- Community contributions for packaging:
  - rpms/debs for Fedora/Debian linux
  - homebrew recipe for MacOS
Sustainability of the RDKit

... thinking about the bus problem

- This clearly isn’t just a hobby project any more
- Used internally in NIBR in multiple production systems
- Starting to get contributions from outside
- I’m no longer the only one answering questions on the mailing list
Overview

- History
- Where are we today?
- What next?
What’s next?

- We’ll decide some of that here in the round-table sessions

- Obvious candidates:
  - Further performance improvements
  - Improved documentation
  - Move more code into C++ (allows access from Knime and the cartridge)
to be aware of...

Dice Holdings buys Geeknet websites for $20M
Dice Holdings acquires Geeknet online media business for $20 million

NEW YORK (AP) -- Dice Holdings Inc. said Tuesday that it acquired Geeknet Inc.'s online media business, including its Slashdot and SourceForge websites, for $20 million in cash.

The New York-based careers website company said the acquisition of the technology websites is part of its strategy of providing content and services geared toward technology professionals.

It’s not clear what this means for sourceforge
Thanks!

C++:
Core data structures and algorithms

web
app

script

CLI

Postgres
SQL

Knime

Java

app

Python

Boost.Python

RDKit

SWIG