RDKit: State of the Toolkit

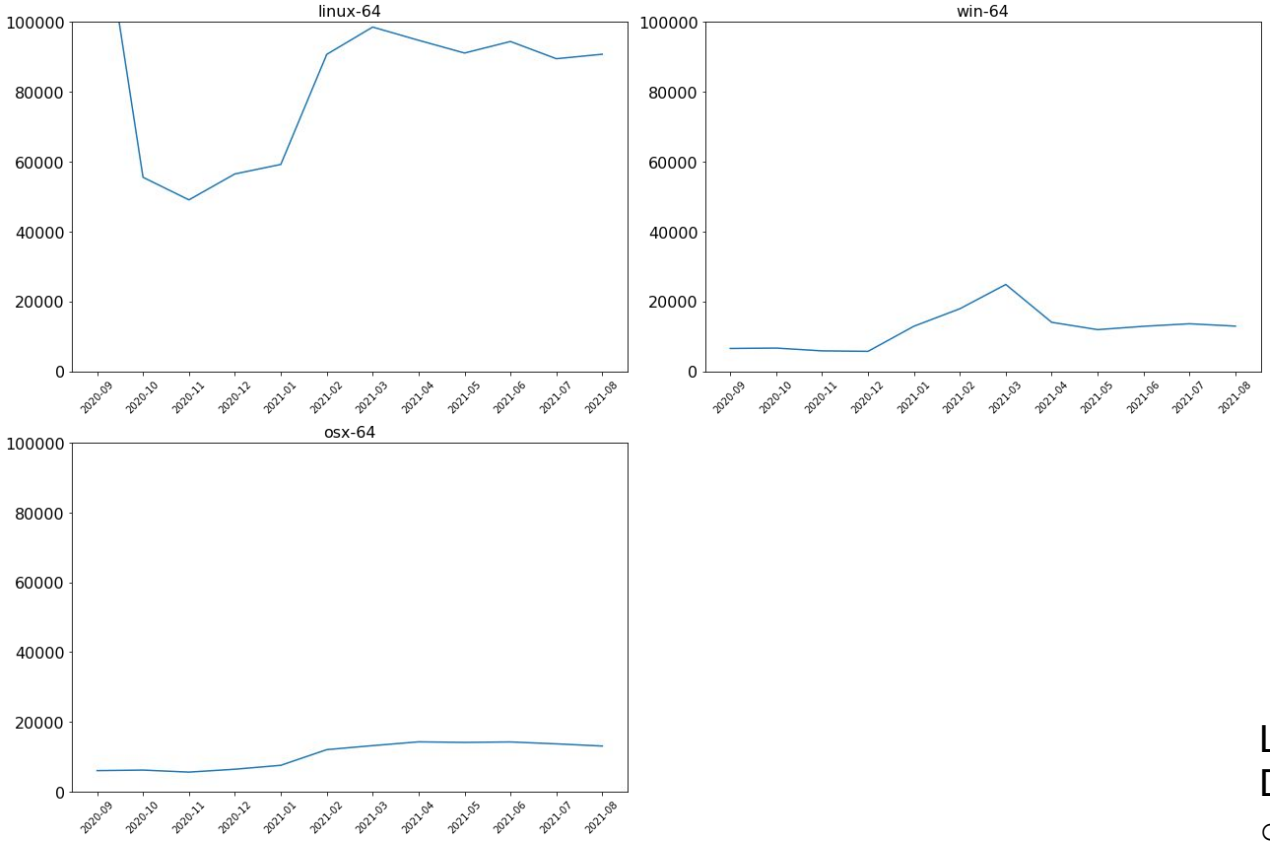
2021 UGM edition

Greg Landrum
@dr_greg_landrum

Adoption / usage

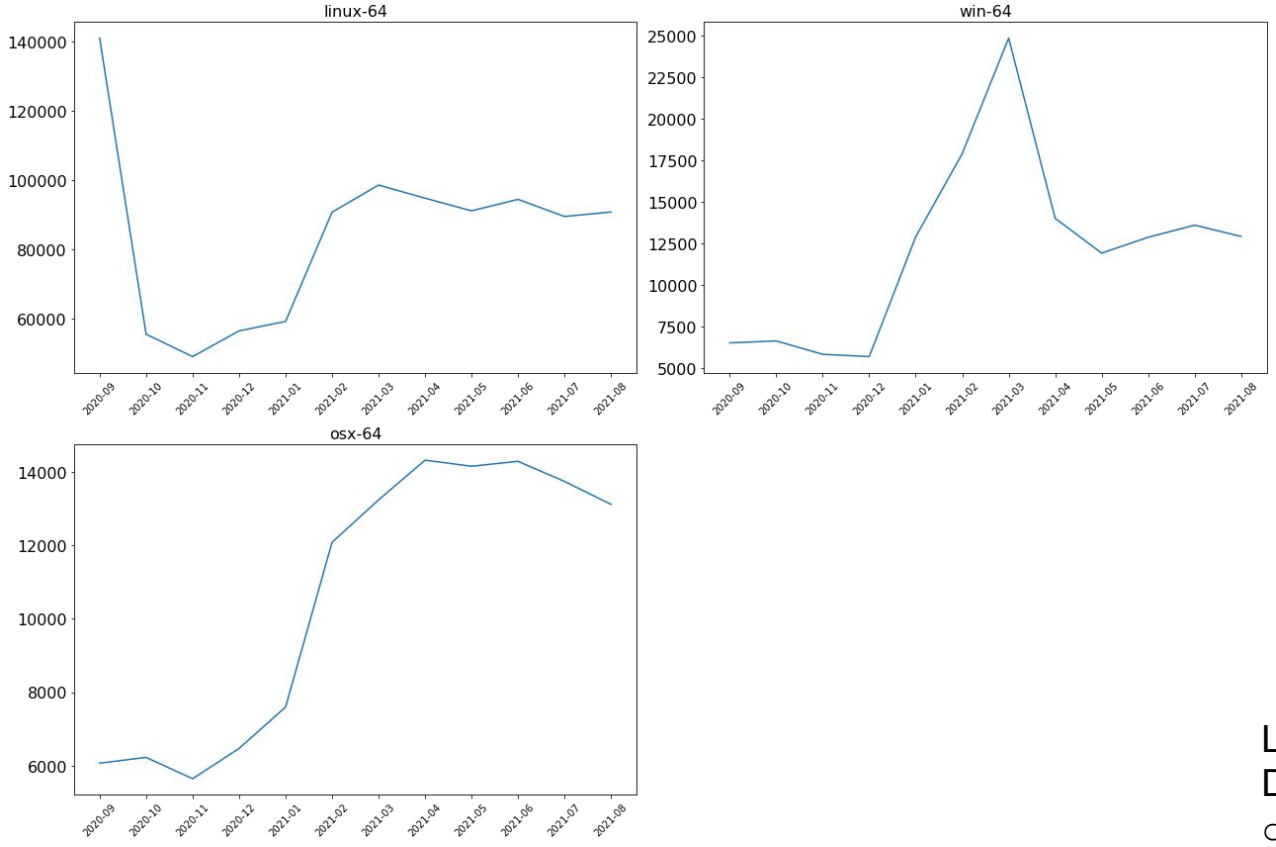
Always tricky to figure out with open source tools, but let’s try.

Usage: Conda install counts (by operating system)



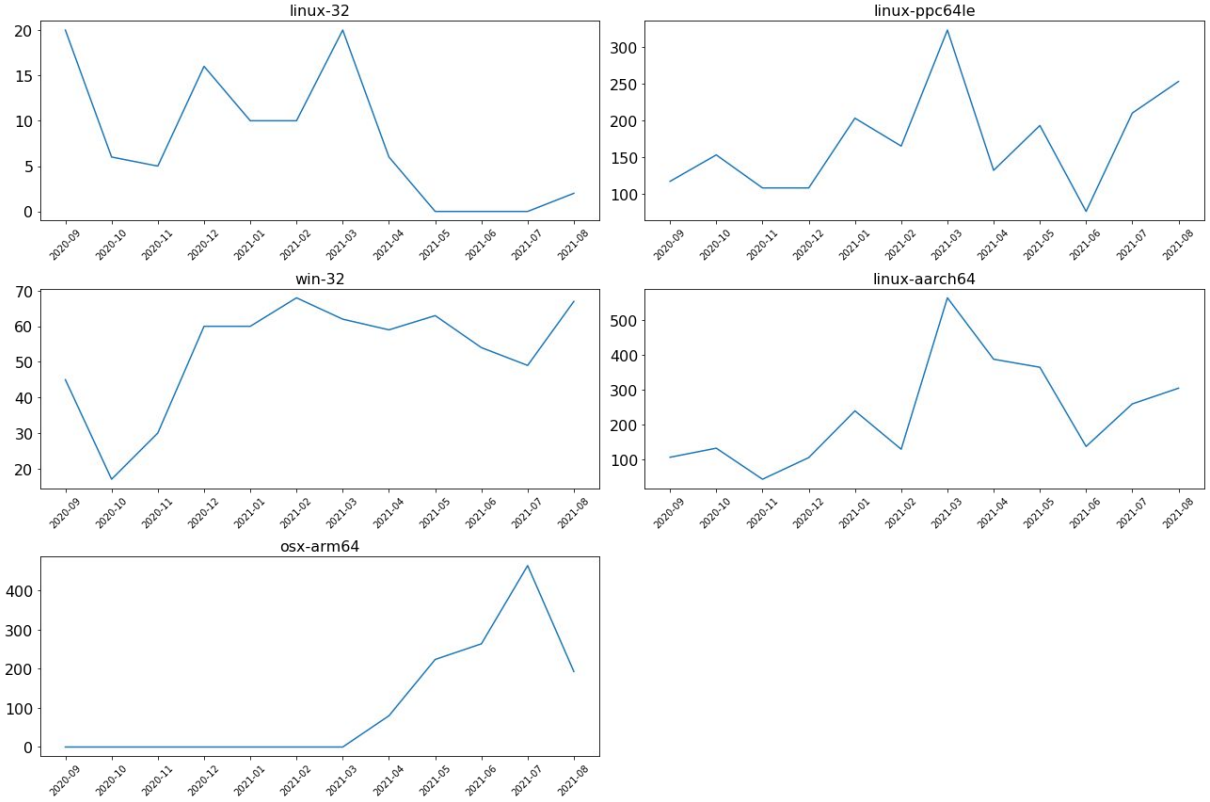
Last 12 months
Data collected using the `condastats` package

Usage: Conda install counts (by operating system)



Last 12 months
Data collected using the `condastats` package

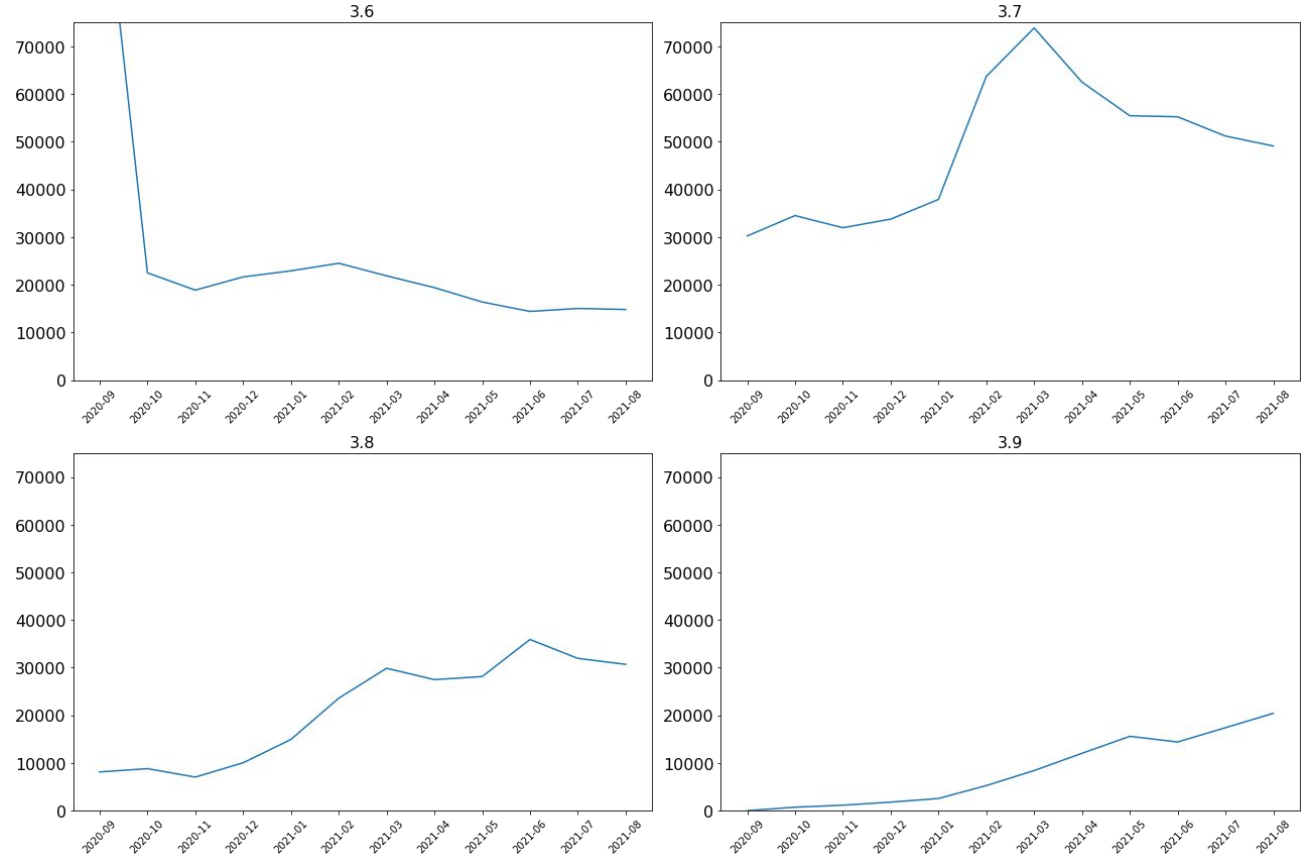
Usage: Conda install counts (by operating system)



Less common operating systems / hardware combos

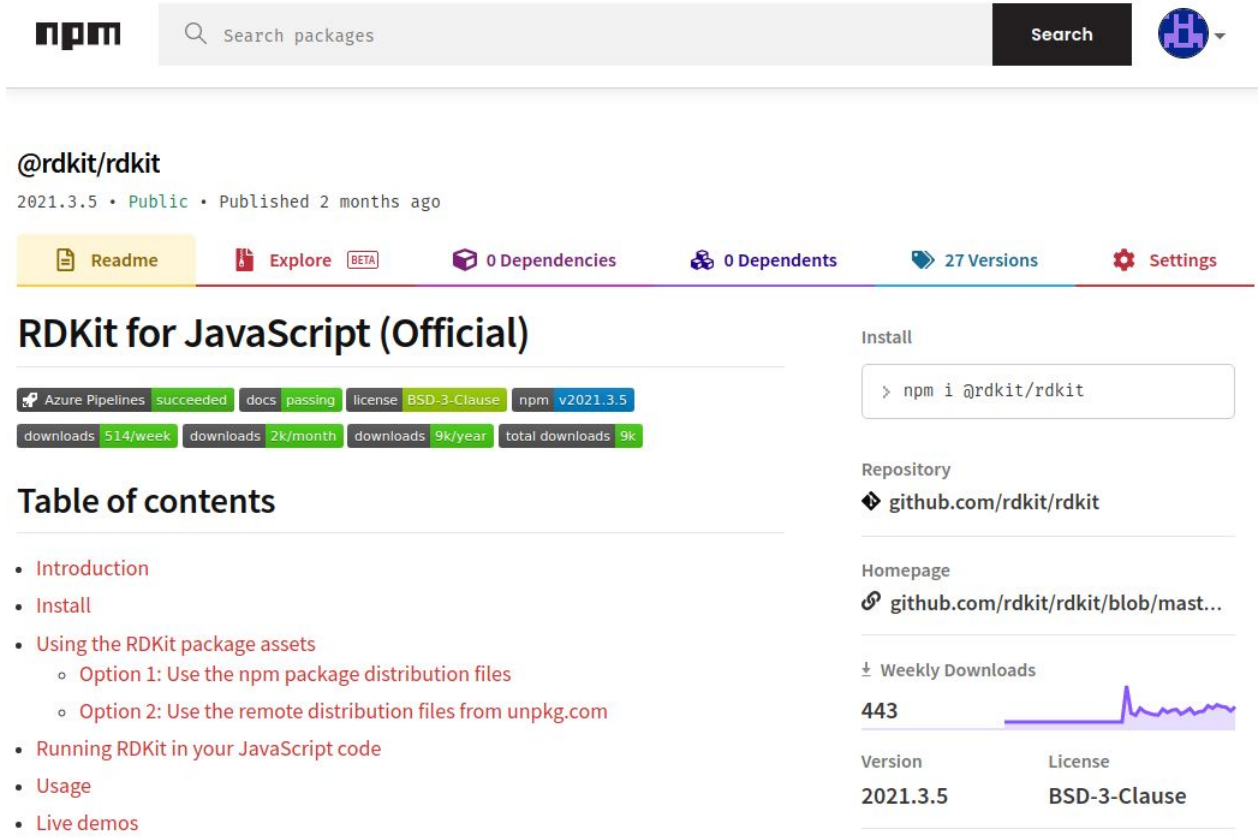
Last 12 months
Data collected using the `condastats` package

Usage: Conda install counts (by python version)



Last 12 months
Data collected using the `condastats` package

Something new: npm packaging



Thanks to Michel Moreau for getting this set up!

Other adoption measures

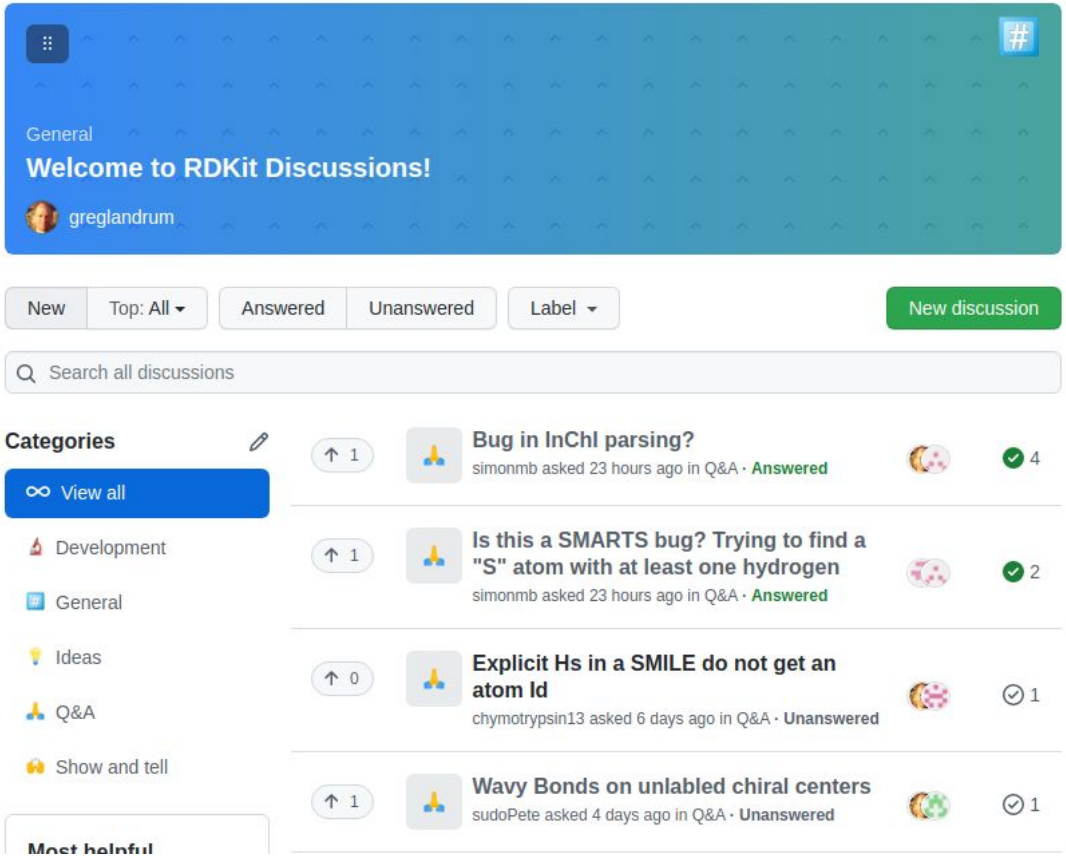
- Mailing lists: ~600 messages to rdkit-discuss from 2020.10.11- 2021.10.12
- Google scholar: >1200 hits for "rdkit" in 2020, >1400 so far in 2021
- Searching github for `from rdkit import Chem` returns >30000 code results
- Each of the last eight UGMs at capacity with 40-100+ attendees

Contributions to github issue tracker in the last year

greglandrum jasondbiggs bp-kelley ricrogz ptosco UnixJunkie IchiruTake mbanck yurivict tdudgeon paconius gmseabra evansenter e-kwsm bzoracler SPKorhonen MichelML ErikCVik zhandn ys0123 wopozka wjm41 sirbiscuit shayakhmetov sargelavoie ryszard314159 rvianello rmrmg rexdwyer rd0102 qq87258580 pschwllr nicogam nbehrnd nanhuayu nadzhou mwojcikowski msudhanshu10 mrcblt mjw99 mily33 mayankBIL matthewyclark malteseunderdog maksbotan macraild ljn917 likhangy kienerj khoivan88 kheyer kekoburgo kazuyaujihara kapilaGIT kangway jw-feng jones-gareth jinpan jhjensen2 iwatobipen imxj i-tub hsiaoyi0504 hgarrereyn hadim goldwind-ting ghiander eloyfelix eguidotti d-b-w chrisbutch chmnk chaoyan1037 cchang373 cbouy brilee bokertof autodataming andt88 amintavakol adelenelai ZontaNicola Zax0NL XFaIT RobinFrcd Plancalkuele MahaveerSatra LivC182 LanceKnight KramerChristian JinghanH IngvarLa DavidACosgrove ChinzoD AustinApple Andy-Wilkinson AlDossetter

That's 97 different people

Community (update)



Usage in other open-source projects (updated 2021)

- **Shape-IT - shape-based alignment**
- **DockOnSurf - high-throughput code to find stable geometries for molecules on surfaces**
- **https://datamol.io/ - A Python library to intuitively manipulate molecules.**
- **Scopy - Python library for desirable HTS/VS database design**
- ChEMBL Structure Pipeline - ChEMBL protocols used to standardise and salt strip molecules.
- FPSim2 - Simple package for fast molecular similarity searches.
- stk (docs, paper) - a Python library for building, manipulating, analyzing and automatic design of molecules.
- OpenFF - Open source approach for better force fields
- gpusimilarity - GPU implementation of fingerprint similarity searching
- Samson Connect - Software for adaptive modeling and simulation of nanosystems
- mol_frame - Chemical Structure Handling for Dask and Pandas DataFrames
- mmpdb 2.0 - matched molecular pair database generation and analysis
- CheTo - Chemical topic modeling
- OCEAN - web-tool for target-prediction of chemical structures which uses ChEMBL as datasource
- Coot - software for macromolecular model building, model completion and validation
- DeepChem - deep learning toolkit for drug discovery
- sdf2ppt - Reads an SDFile and displays molecules as image grid in powerpoint/openoffice presentation.
- chemfp
- PYPL - Simple cartridge that lets you call Python scripts from Oracle PL/SQL.
- WONKA - Tool for analysis and interrogation of protein-ligand crystal structures
- OOMMPPAA - Tool for directed synthesis and data analysis based on protein-ligand crystal structures
- chemicalite - SQLite integration for the RDKit
- django-rdkit - Django integration for the RDKit
- … more ...

New projects are in bold

Sustainability: the bus problem

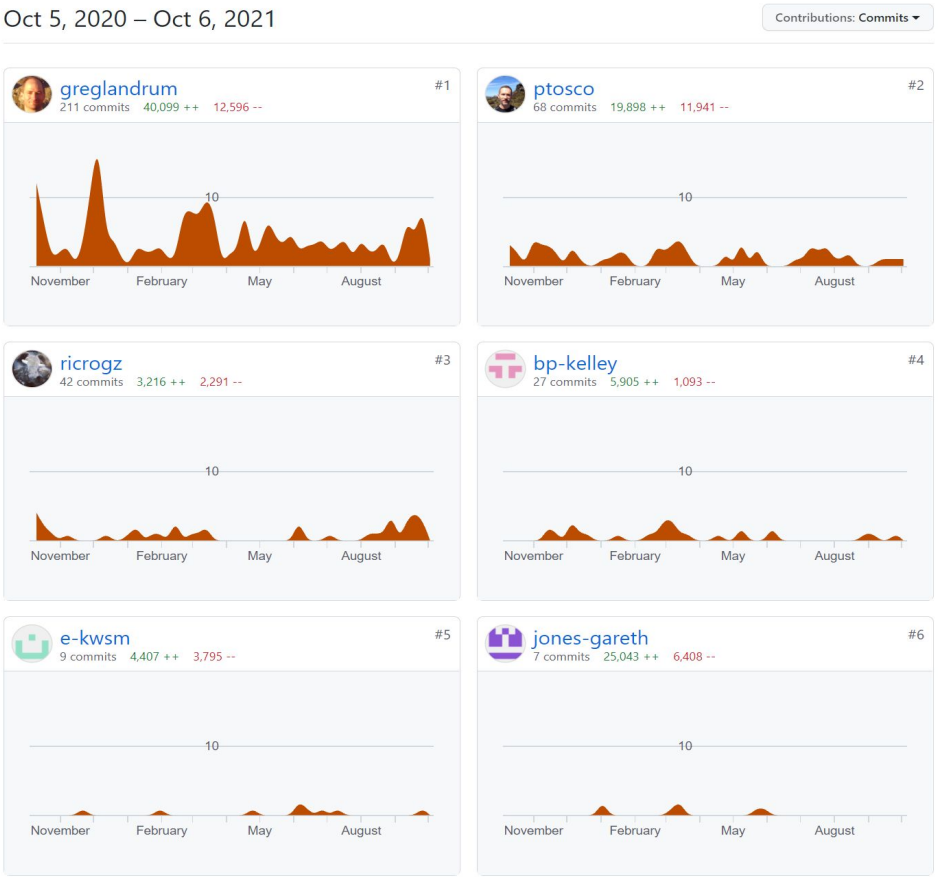
https://commons.wikimedia.org/wiki/File:Postauto_susten.jpg

Sustainability: the bus problem

RDKit maintainers:

- Greg
- Brian Kelley (Relay Therapeutics)
- Ricardo Rodriguez (Schrödinger)
- Paolo Tosco (Novartis)

Most frequent code contributors in the last year



Merged pull request contributors in the last year

bzoracler, hjuinj, David Cosgrove, sailfish009, Paolo Tosco, cespos, slchan3, Rocco Moretti, Maximilian Greil, charly828, Niels Kristian Kjærgård Madsen, John Konecny, jungb-basf, Steven Kearnes, Kit Choi, Eloy Félix, Hadrien Mary, Zhijiang Yang, Daniel Paoliello, jasondbiggs, Riccardo Vianello, Maximilian Peters, Eisuke Kawashima, Greg Landrum, Braxton, Dan N, NadineSchneider, Michel Moreau, Kazuya Ujihara, JLVarjo, Steve Roughley, sirbiscuit, SPKorhonen, Maciej Wójcikowski, pkubaj, Vincent F. Scalfani, Gareth Jones, magattaca, Rachel Walker, Ric, Brian Kelley, Jin Pan, Luca Naef, Matt Swain

That's 44 different people

Maintenance work in the last year

We started tracking maintenance/cleanup work with the 2019.09 release.

For the 2021.03 and 2021.09 releases, there have been >40 “cleanup” issues/PRs merged:

Greg Landrum 21
Paolo Tosco 7
Ric 5
Eisuke Kawashima 5
jasondbiggs 2
John Konecny 1
Gareth Jones 1
Dan N 1
Brian Kelley 1

Roadmap

Future work tends to be determined by what's needed for active projects or requests that come out of the community. So there's not much of a roadmap.

Still, some parts of the way forward are pretty obvious...

Making sure all the pieces required to build a good compound registration system are there

Making sure all the pieces required to build a good corporate chemical database are there

Better support for polymers and organometallics

Performance improvements

Ongoing refactoring and code cleanup

Some things are hard...

Technology changes (i.e. taking advantage of new C++ or Python versions) is tricky: which operating systems/compilers are people using?

Is it safe to remove old code that seems peripheral or redundant with functionality provided better by other packages?

There are some larger API changes to clean up old mistakes that it would be nice to make, but we can’t just make arbitrary changes

Really, really want to avoid the Python 2/Python 3 situation

… what we're doing about it

Try to minimize hard external dependencies

Be *very* conservative about language versions/features

Announce deprecations at least one major release in advance

"Backwards incompatible changes" doc

Version-compatibility report (for commercial support customers)

Ok, enough of that, let's look at what's new

I'm trying an experiment this year and using mybinder.org to create an interactive version of the What's New notebook:
https://mybinder.org/v2/gh/rdkit/UGM_2021/main?urlpath=tree/Notebooks
or https://tinyurl.com/RDKit-Notebook-2021
(the link is in the discord chat)

The notebook itself is also in the in the UGM github repo:
https://github.com/rdkit/UGM_2021

You can install the RDKit nightly build for Python 3.8/Linux that I used with:

```
conda install -c rdkit/label/beta rdkit
```