

Ghent University in OpenDreamKit

Jeroen Demeyer

UGent added to OpenDreamKit

- ▶ Jeroen Demeyer applied for a “ingénieur de recherche” position at Paris-Sud to work for OpenDreamKit.
- ▶ Jeroen was recruited 1 March 2016, but there were a lot of administrative difficulties because he did not want to move to France.
- ▶ To solve this, it was decided to add UGent as partner of OpenDreamKit, starting at 1 July 2016.
- ▶ UGent has a single participant, Jeroen Demeyer, who will work 28.4 PM at UGent for OpenDreamKit (after having worked 2 PM at Paris-Sud).

Who is Jeroen Demeyer?

- ▶ SageMath developer since July 2010 (Sage Days 23 in Leiden).
Main contributions:
 - ▶ Build system.
 - ▶ PARI interface.
 - ▶ Coercion model, core arithmetic and comparison infrastructure.
 - ▶ Lines of code written for Sage according to GitHub: –125 115.
- ▶ SageMath release manager from January 2011 (Sage 4.6.1) to December 2013 (Sage 5.13).
- ▶ Main developer of cysignals: a Cython package for interrupt, signal, error and memory handling.
- ▶ Main developer of pari_jupyter: a kernel for the Jupyter Notebook running PARI/GP.

D 3.2: Understand and document SageMathCloud backend code

Some initial attempts to run SageMathCloud on a personal laptop.

- ▶ There is documentation from upstream on how to do this.
- ▶ Managed to run the SMC server, but starting projects did not work.
- ▶ Various issues reported upstream: upstream updated some documentation. TODO: try again now!

D 3.10: Packaging for major Linux distributions

There is a big effort going on to package Sage in Debian. So far, things are looking well: the next Debian release should have a Sage package.

UGent mainly helped to make Sage easier to package. In particular:

- ▶ Change the build system of the Sage Python library to use a more standard `setup.py` file.
- ▶ Stop hardcoding the installation directory of Sage: allow `SAGE_LOCAL` and other directories to be customized.
- ▶ Many more small fixes. . .

D 4.1: Python/Cython bindings for PARI and its integration in Sage

This deliverable is about splitting off the PARI interface of Sage to a separate package `cypari2`.

- ▶ This turned out to be much harder than originally anticipated: the PARI bindings were quite closely tied to Sage.
- ▶ Two most important problems:
 - ▶ Interrupt (CTRL-C) and error handling using the `sig_on()` mechanism.
 - ▶ The coercion model which is needed to do arithmetic between PARI elements and other Sage elements.

D 4.1: Python/Cython bindings for PARI and its integration in Sage

This deliverable has been split in 4 sub-tasks:

1. Split off interrupt/signal/error handling to a separate package `cysignals`: DONE.
2. Refactor the Sage coercion model. Add better support for non-Sage types, such that the PARI bindings no longer need the coercion model. DONE.
3. Split off the remaining Sage-specific parts of the PARI interface (for example, conversion from/to Sage types). DONE but not yet merged in Sage.
4. Finally, actually split off the PARI bindings from Sage as a new package. Easy, almost done.

D 4.1: Python/Cython bindings for PARI and its integration in Sage

This deliverable has been split in 4 sub-tasks:

1. Split off interrupt/signal/error handling to a separate package `cysignals`: DONE.
2. Refactor the Sage coercion model. Add better support for non-Sage types, such that the PARI bindings no longer need the coercion model. DONE.
3. Split off the remaining Sage-specific parts of the PARI interface (for example, conversion from/to Sage types). DONE but not yet merged in Sage.
4. Finally, actually split off the PARI bindings from Sage as a new package. Easy, almost done.

D 4.4: Basic Jupyter interface for GAP, PARI/GP, SageMath, Singular

Done for PARI/GP: Python package `pari_jupyter` on PyPI.

- ▶ Supports all GP functions.
- ▶ Supports history and TAB-completion as in GP.
- ▶ TODO: discuss with Jupyter upstream about the proper way to install kernels (in particular kernel specs) such that all kernels can use the same way.

D 4.7: Full featured Jupyter interface for GAP, PARI/GP, Singular

Partially done for PARI/GP: Python package `pari_jupyter` on PyPI.

- ▶ Plotting is supported using SVG images. This required adding SVG plotting functionality, which is proposed to upstream PARI but not yet merged.
- ▶ TODO: syntax highlighting, long help, break loop.

D 4.13: Refactorisation of SageMath's Sphinx documentation system

Big task, but many things have been done:

- ▶ Sphinx in Sage has been upgraded to version 1.4.1 (latest upstream is version 1.5.1).
- ▶ A lot of cleaning up and minor refactoring has been done.
- ▶ Reduce the amount of Sage-specific stuff in docbuilder: a lot of progress but a lot remains to do. This is mostly about Cython, so coordination with Python/Cython upstream might be needed.
- ▶ Some work has been done with upstream to reduce the memory footprint of Sphinx.