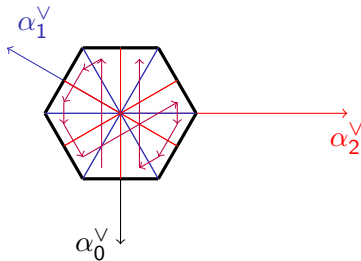


Sagemath

(Free Software for Mathematics)

<https://sagemath.org>



With slides from Nicolas Thiéry, Franco Saliola, Florent Hivert, Dan Drake, William Stein, ...

Sage is free software

Sage is free software

“Mission: Creating a viable free open source alternative to
Maple™, Mathematica™, Magma™ et MATLAB™”

...

Sage is free software: the GNU definition

A program is free software if the program's users have the four essential freedoms:

- 0 The freedom to run the program as you wish, for any purpose.
- 1 The freedom to study how the program works, and change it so it does your computing as you wish. Access to the source code is a precondition for this.
- 2 The freedom to redistribute copies so you can help your neighbor.
- 3 The freedom to distribute copies of your modified versions to others. By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

See <https://www.gnu.org/philosophy/free-sw.en.html>

Sage is free software: free as in “free beer”

Sage is free software: free as in “free beer”

- ▶ No need to be in a rich laboratory to be able to use it
- ▶ Technical and administrative simplicity
- ▶ Can be used by students at home
- ▶ Remote computation, large scale deployment
- ▶ Non discrimination
- ▶ Free access to non academics
- ▶ Free access in developing countries

Sage is free software: free as in “free beer”

- ▶ No need to be in a rich laboratory to be able to use it
- ▶ Technical and administrative simplicity
- ▶ Can be used by students at home
- ▶ Remote computation, large scale deployment
- ▶ Non discrimination
- ▶ Free access to non academics
- ▶ Free access in developing countries

But the production has a cost: *nothing prevents the rich to contribute.*

Sage is free software: access to the source code

“You can read Sylow’s Theorem and its proof in Huppert’s book in the library . . . then you can use Sylow’s Theorem for the rest of your life free of charge, but for many computer algebra systems license fees have to be paid regularly

With this situation two of the most basic rules of conduct in mathematics are violated: In mathematics information is passed on free of charge and everything is laid open for checking.”

— J. Neubüser (1993)
(GAP founder in 1986)

Sage is free software: access to the source code

- ▶ Teaching
- ▶ Reproducibility of scientific results
- ▶ Proof checking
- ▶ Control over the hypothesis, models and algorithms
- ▶ Analysis of bugs and unexpected behavior

Sage is free software: mutualization

Build the bike instead of reinventing the wheel:

Arbitrary precision arithmetic	MPIR (GMP), MPFR, MPFI, NTL
Algebra	GAP, Maxima, Singular, Givaro
Algebraic geometry	Singular, Macaulay2*
Arithmetic geometry	FLINT, PARI/GP, NTL, ecm
Courbes elliptiques et fonctions L	ECLib, mwrnk, ratpoints, SYMPOW, Lca
Symbolic computation	Pynac, Maxima, Sympy, giac*
Exact linear algebra	Linbox, IML
Numerical calculations	Blas (Atlas), Numpy, LAPACK
Numerical calculations	GSL, Scipy
Combinatorics	Symmetrica, Lrcalc, PALP, Coxeter 3, Che
Graph theory	NetworkX, Cliquer, Buckygen*, graphviz*, cvxopt, PPL, glpk, CBC*
Group theory	GAP
Game theory	Gambit*
Statistics	R, Rpy, pandas*
Cryptography	pycrypto, cryptominisat*

(* optional)

... and more!

Sage is free software: mutualization

- ▶ about 400k lines of code or doc
- ▶ about 7000 classes
- ▶ about 7000 functions
- ▶ Rich mathematical content: categories, combinatorics, graphs, number theory, ...
- ▶ Many new algorithms

Share the development to focus on the code specific to your own research.

Sage is free software: mutualization



me

1 developer

1k lines of code

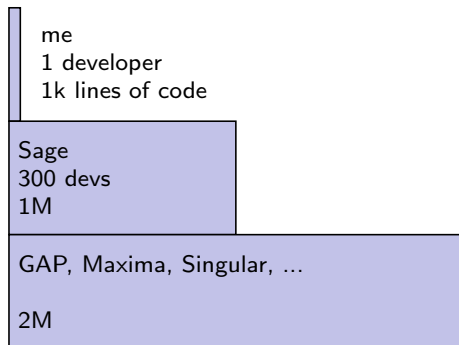
On the shoulders of a giant.

Sage is free software: mutualization



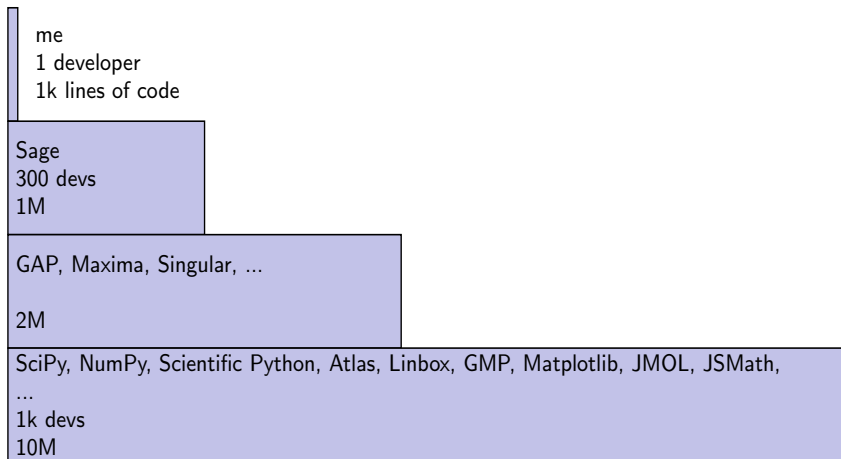
On the shoulders of a giant.

Sage is free software: mutualization



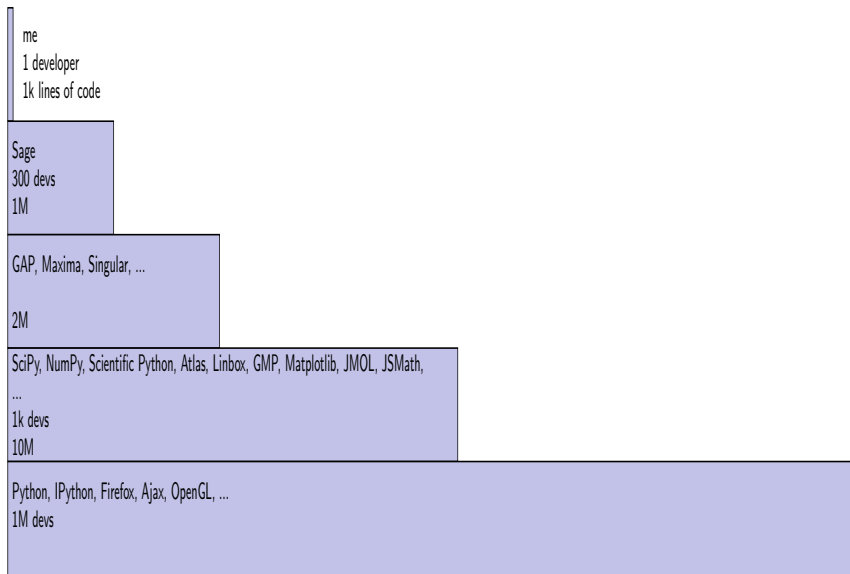
On the shoulders of a giant.

Sage is free software: mutualization



On the shoulders of a giant.

Sage is free software: mutualization



On the shoulders of a giant.

Sage is free software: in both directions

Freedom to improve and publish one's improvements

- ▶ Adaptation to local needs (dialects, conventions)
- ▶ Specific developments
- ▶ Bug fixes
- ▶ Empowering of users
- ▶ Mutualisation of efforts
- ▶ Importance of communities

Sage is free software: in both directions

- ▶ A tarball on your webpage far is better than nothing, as easy as:
 - ▶ `tar czvf my-code.tgz my-code/`
 - ▶ `scp my-code.tgz my-lab:my-homepage/.`
- ▶ public git repository
- ▶ `pip/R/gap/npm/...` package
- ▶ integration into a low-level library
- ▶ integration into Sage

Sage is free software: in both directions

Beyond the tarball that rots on your webpage.

- ▶ Your code will be read (at least once), and even used!
- ▶ Compilation problems on strange machines will be dealt with.
- ▶ Your code enters a collective world, and will be maintained if it is good (or trashed).
- ▶ The future PhD candidates in your field can position themselves with respect to your work, and build on top of it.
- ▶ Note: it is important to let the permanent researchers do the well-known code and maintenance, so that PhD and postdoc can focus on their own research and algorithms.
- ▶ Remember that Python is only the interface language, if you have a fast optimized library written in C (for example), it also can enter Sage or one of the specialized libraries it uses.