

Introduction à



Marc MEZZAROBBA

CNRS, LIP6, Université Paris 6

Qu'est-ce que Sage ?

« *Une alternative libre viable à Magma,
Maple, Mathematica et Matlab* »

« *Construire la voiture au lieu
de réinventer la roue* »



Essayer Sage



<http://sagemath.org/>
GNU GPL



<http://sagenb.org/>



<http://cloud.sagemath.com/>



















Qu'est-ce que Sage ?

- 1 Une distribution
- 2 Une bibliothèque Python
- 3 Un système interactif



Une distribution

```
$ wget http://mirror/sagemath/src/sage-6.3.tar.gz  
&& tar xfz sage-6.3.tar.gz && cd sage-6.3 && make  
$ ./sage
```

ATLAS • *boehm_gc* •  cddlib • cephes • cliquer
cvxopt •  ython • ECL • eclib • ecm • *f2c* • fplll
FLINT • GAP • gfan •  Glibto • GLPK • GSL
IML • IP[y]: ipython •  LAPACK • lcalc
 LinBox •  matplotlib •  Maxima • M4RI
MPC • MPFI •  MPFR • mpmath • networkx • NTL
 NumPy • PALP •  PARI •  PolyBoRi •  Pynac
 python •  R •  SciPy •  SINGULAR •  symmetrica
sympow •  tachyon • zn_poly • + *d'autres...*

+ ~ 60 paquets optionnels + ~ 60 paquets expérimentaux

Une bibliothèque Python

<code>sage.algebras</code>	<code>sage.logic</code>
<code>sage.calculus</code>	<code>sage.matrix</code>
<code>sage.categories</code>	<code>sage.modular</code>
<code>sage.coding</code>	<code>sage.modules</code>
<code>sage.combinat</code>	<code>sage.monoids</code>
<code>sage.crypto</code>	<code>sage.numerical</code>
<code>sage.databases</code>	<code>sage.parallel</code>
<code>sage.finance</code>	<code>sage.plot</code>
<code>sage.functions</code>	<code>sage.rings</code>
<code>sage.geometry</code>	<code>sage.sat</code>
<code>sage.graphs</code>	<code>sage.schemes</code>
<code>sage.groups</code>	<code>sage.sets</code>
<code>sage.homology</code>	<code>sage.stats</code>
<code>sage.interfaces</code>	<code>sage.symbolic</code>
<code>sage.lfunctions</code>	<code>...</code>

- S'appuie sur les logiciels tiers embarqués
- \simeq 600 000 lignes de code spécifique (hors doc + tests)



Une bibliothèque Python

<code>sage.algebras</code>	<code>sage.logic</code>
<code>sage.calculus</code>	<code>sage.matrix</code>
<code>sage.categories</code>	<code>sage.modular</code>
<code>sage.coding</code>	<code>sage.modules</code>
<code>sage.combinat</code>	<code>sage.monoids</code>
<code>sage.crypto</code>	<code>sage.numerical</code>
<code>sage.databases</code>	<code>sage.parallel</code>
<code>sage.finance</code>	<code>sage.plot</code>
<code>sage.functions</code>	<code>sage.rings</code>
<code>sage.geometry</code>	<code>sage.sat</code>
<code>sage.graphs</code>	<code>sage.schemes</code>
<code>sage.groups</code>	<code>sage.sets</code>
<code>sage.homology</code>	<code>sage.stats</code>
<code>sage.interfaces</code>	<code>sage.symbolic</code>
<code>sage.lfunctions</code>	<code>...</code>

- S'appuie sur les logiciels tiers embarqués
- \simeq 600 000 lignes de code spécifique (hors doc + tests)



Une bibliothèque Python

<code>sage.algebras</code>	<code>sage.logic</code>
<code>sage.calculus</code>	<code>sage.matrix</code>
<code>sage.categories</code>	<code>sage.modular</code>
<code>sage.coding</code>	<code>sage.modules</code>
<code>sage.combinat</code>	<code>sage.monoids</code>
<code>sage.crypto</code>	<code>sage.numerical</code>
<code>sage.databases</code>	<code>sage.parallel</code>
<code>sage.finance</code>	<code>sage.plot</code>
<code>sage.functions</code>	<code>sage.rings</code>
<code>sage.geometry</code>	<code>sage.sat</code>
<code>sage.graphs</code>	<code>sage.schemes</code>
<code>sage.groups</code>	<code>sage.sets</code>
<code>sage.homology</code>	<code>sage.stats</code>
<code>sage.interfaces</code>	<code>sage.symbolic</code>
<code>sage.lfunctions</code>	<code>...</code>

- S'appuie sur les logiciels tiers embarqués
- \simeq 600 000 lignes de code spécifique (hors doc + tests)



Une bibliothèque Python

`sage.algebras`

`sage.calculus`

`sage.categories`

`sage.coding`

`sage.combinat`

`sage.crypto`

`sage.databases`

`sage.finance`

`sage.functions`

`sage.geometry`

`sage.graphs`

`sage.groups`

`sage.homology`

`sage.interfaces`

`sage.lfunctions`

`sage.logic`

`sage.matrix`

`sage.modular`

`sage.modules`

`sage.monoids`

`sage.numerical`

`sage.parallel`

`sage.plot`

`sage.rings`

`sage.sat`

`sage.schemes`

`sage.sets`

`sage.stats`

`sage.symbolic`

...

- S'appuie sur les logiciels tiers embarqués
- \simeq 600 000 lignes de code spécifique (hors doc + tests)



Une bibliothèque Python

<code>sage.algebras</code>	<code>sage.logic</code>
<code>sage.calculus</code>	<code>sage.matrix</code>
<code>sage.categories</code>	<code>sage.modular</code>
<code>sage.coding</code>	<code>sage.modules</code>
<code>sage.combinat</code>	<code>sage.monoids</code>
<code>sage.crypto</code>	<code>sage.numerical</code>
<code>sage.databases</code>	<code>sage.parallel</code>
<code>sage.finance</code>	<code>sage.plot</code>
<code>sage.functions</code>	<code>sage.rings</code>
<code>sage.geometry</code>	<code>sage.sat</code>
<code>sage.graphs</code>	<code>sage.schemes</code>
<code>sage.groups</code>	<code>sage.sets</code>
<code>sage.homology</code>	<code>sage.stats</code>
<code>sage.interfaces</code>	<code>sage.symbolic</code>
<code>sage.lfunctions</code>	<code>...</code>

- S'appuie sur les logiciels tiers embarqués
- \simeq 600 000 lignes de code spécifique (hors doc + tests)



Une bibliothèque Python

<code>sage.algebras</code>	<code>sage.logic</code>
<code>sage.calculus</code>	<code>sage.matrix</code>
<code>sage.categories</code>	<code>sage.modular</code>
<code>sage.coding</code>	<code>sage.modules</code>
<code>sage.combinat</code>	<code>sage.monoids</code>
<code>sage.crypto</code>	<code>sage.numerical</code>
<code>sage.databases</code>	<code>sage.parallel</code>
<code>sage.finance</code>	<code>sage.plot</code>
<code>sage.functions</code>	<code>sage.rings</code>
<code>sage.geometry</code>	<code>sage.sat</code>
<code>sage.graphs</code>	<code>sage.schemes</code>
<code>sage.groups</code>	<code>sage.sets</code>
<code>sage.homology</code>	<code>sage.stats</code>
<code>sage.interfaces</code>	<code>sage.symbolic</code>
<code>sage.lfunctions</code>	<code>...</code>

- S'appuie sur les logiciels tiers embarqués
- \simeq 600 000 lignes de code spécifique (hors doc + tests)



Un système interactif

```
-$ sage
```

```
| Sage Version 4.6.2, Release Date: 2011-02-25 |  
| Type notebook() for the GUI, and license() for information. |
```

```
sage: taylor(exp(x), x, 0, 5)  
1/120*x^5 + 1/24*x^4 + 1/6*x^3 + 1/2*x^2 + x + 1  
sage:  
sage: MatrixSpace(RR,5,3).random_element()  
[-0.570390764900653  0.521446993576251 -0.950894560265950]  
[-0.942431942330060  0.254122819002693  0.916721924359961]  
[-0.195702504102615 -0.350489870318781 -0.214359534055980]  
[ 0.487076746020482  0.461116221981387 -0.665179594662514]  
[ 0.180194930460366  0.616390883848273 -0.389309976296204]  
sage:  
sage: import urllib2  
sage: f = urllib2.urlopen("http://sagemath.org/")  
sage: f.read(121)  
'<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" \n"http  
://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">'  
sage:
```

Enveloppes (2) (Sage) - Iceweasel

Iceweasel [01] Enveloppes (2) (Sage)

The Sage admin [Toggle](#) | [Home](#) | [Published](#)
Notebook [Log](#) | [Settings](#) | [Help](#)
Version 4.6.2 [Report a Problem](#) | [Sign out](#)

Enveloppes (2) [Save](#) [Save & quit](#) [Discard & quit](#)

last edited on May 15, 2011 09:31 PM by admin

File... Action... Data... sage Typeset

[Print](#) [Worksheet](#) [Edit](#) [Text](#) [Undo](#) [Share](#) [Publish](#)

R.<x,y,t> = QQ[]; eq = x^2+(y-t)^2-1/2*(t^2 + 1) eq

$$x^2 + y^2 - 2yt + \frac{1}{2}t^2 - \frac{1}{2}$$

```
fig_circles = add((eq(t=k/5)*QQ[x,y]).plot()  
for k in (-15..15))  
options = {'aspect_ratio': 1, 'xmin': -2,  
'xmax': 2, 'ymin': -3, 'ymax': 3, 'frame':  
True, 'axes': False, 'fontsize': 8}  
fig_circles.show(**options)
```

evaluate

jsMath