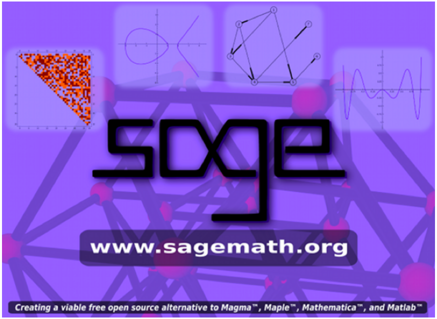
Introduction to Sage

William Stein
University of Washington



February 26, 2011

The Sage Project

Mission Statement

Create a viable **free open source** alternative to Magma, Maple, Mathematica, and Matlab

1. *Mathematical features* of all of Magma, Maple, Mathematica, and Matlab
2. A notebook interface
3. Many books (undergraduate/grad curriculum)

History

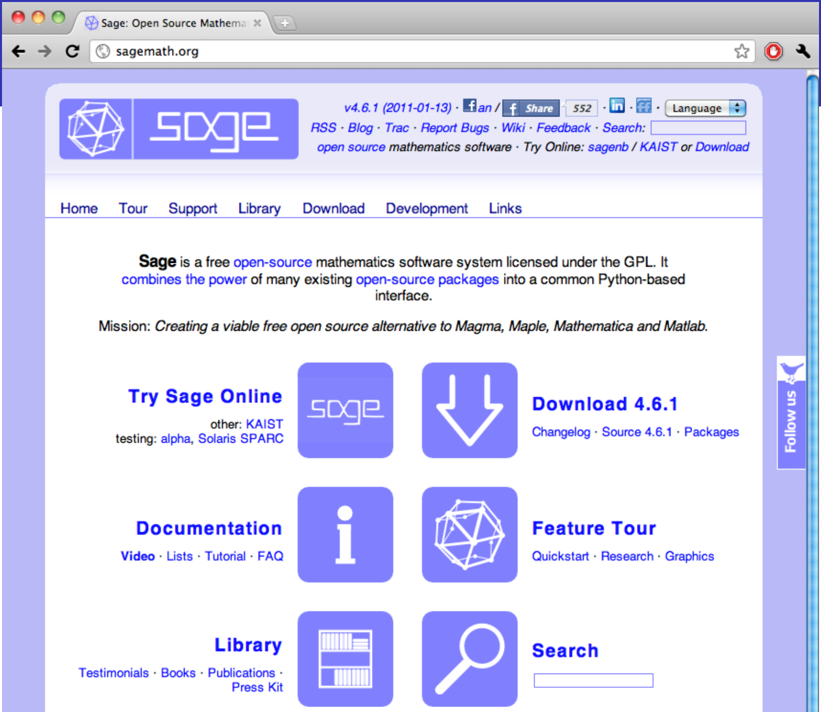
Sage Timeline

- **2005:** I released Sage 0.1... long year of very hard work.
- **2006:** *(2 Sage Days)* Sage is not just for number theory!
- **2007:** *(4 Sage Days)* Win prize—tons of publicity; 100% test requirements and peer review of all code; industry funding (Google, Microsoft).
- **2008:** *(7 Sage Days)* Release managers besides me.
- **2009:** *(8 Sage Days)* Better quality; more developers.
- **2010:** *(13 Sage Days)* More people; serious NSF support.
- **2011:** Revamping web interface (due to new funding), undergrad curriculum materials, new research tools.

What is Sage?

Sage

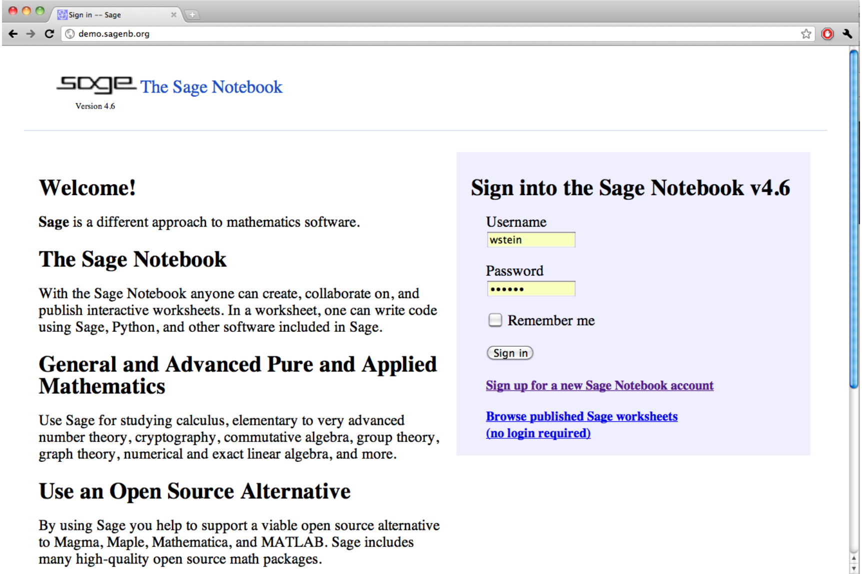
- **Python:** a mainstream programming language
- **Distribution:** over 90 open source packages
- **Interfaces:** smoothly combine packages
- **New code:** implements novel algorithms; over half million lines written by several hundred people.


Sage: Open Source Mathema
sagemath.org
sage
v4.6.1 (2011-01-13) · fan / Share 552 · · · Language
RSS · Blog · Trac · Report Bugs · Wiki · Feedback · Search:
open source mathematics software · Try Online: sagenb / KAIST or Download
Home Tour Support Library Download Development Links
Sage is a free open-source mathematics software system licensed under the GPL. It combines the power of many existing open-source packages into a common Python-based interface.
Mission: Creating a viable free open source alternative to Magma, Maple, Mathematica and Matlab.
Try Sage Online
other: KAIST
testing: alpha, Solaris SPARC
sage
Download 4.6.1
Changelog · Source 4.6.1 · Packages
Documentation
Video · Lists · Tutorial · FAQ
Feature Tour
Quickstart · Research · Graphics
Library
Testimonials · Books · Publications · Press Kit
Search
Follow us

47,942 Website Visits *this* month (Feb 2011)

Sage has interest from all over, but is still relatively small...
(No advertising yet; all word of mouth/grassroots.)

The Sage Notebook



MAA talk on Sage - 20min

MAA Talk on Sage

William Stein

February 26, 2011, Santa Rosa

Factor

Factoring an integer:

```
factor(2012)
```

$2^2 \cdot 503$

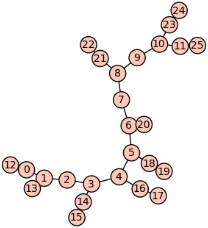
Factoring a symbolic expression:

```
var('x,y')
factor(x^8 - y^2*e^(2*x))
```

$(x^4 - ye^x)(x^4 + ye^x)$

Graph Theory

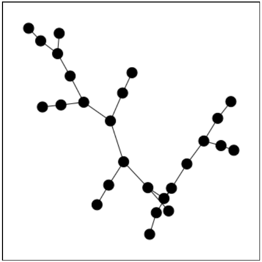
```
set_random_seed(1); G = graphs.RandomLobster(7, .6, .3); show(G)
```




```
G.automorphism_group()
```

$\langle (11,23)(24,25) \rangle$

```
graph_editor(G)
```



live:

variable name: G

strength:

length:

help

Save Close

```
G.is_planar()
```

True

Symbolic Integrals

```
integrate(sin(x)*tan(x), x)
```

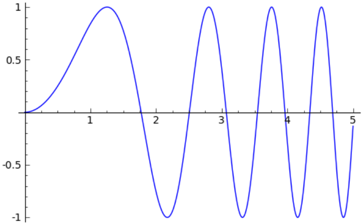
$$-\frac{1}{2}\log\left(\sin\left(x\right) - 1\right) + \frac{1}{2}\log\left(\sin\left(x\right) + 1\right) - \sin\left(x\right)$$

```
f = 1/sqrt(x^2 + 2*x - 1)
f.integrate(x)
```

$$\log\left(2\,x + 2\,\sqrt{x^{2} + 2\,x - 1} + 2\right)$$

Plotting Functions

```
plot(sin(x^2), (x,0,5))
```

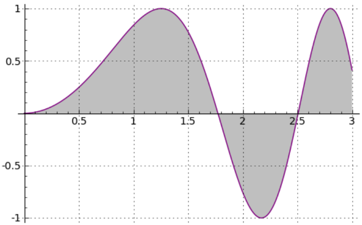



```
var('x')
@interact
def h(f=sin(x^2), grid=True, t=(1..20)):
    show(plot(f, (x, 0, 3),
                  thickness=t, color='purple', fill=True,
gridlines=grid))
```

f sin(x^2)

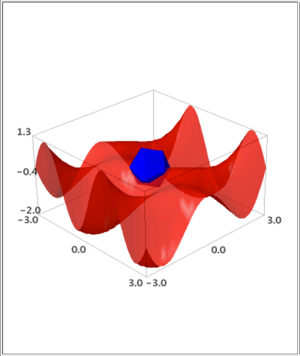
grid

t 1



Plotting a Function in 3D

```
f(x,y) = sin(x - y)*y*cos(x)
plot3d(f, (x,-3,3), (y,-3,3), opacity=.7, color='red') +
icosahedron(color='blue')
```



[Wake this 3-D view]
[Click to Sleep this 3-D view]
[Help for Jmol 3-D viewer]

Applet

Functions

State

3-D display size:

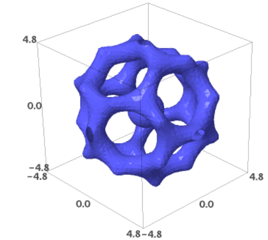
Medium (400px)

- [Arbitrarily resizable in own window]
- [Get static image to save]

☐ Spin on

An implicit 3D plot:

```
T = RDF(golden_ratio)
var('x,y,z')
p = (2 - (cos(x + T*y) + cos(x - T*y) + cos(y + T*z) +
          cos(y - T*z) + cos(z - T*x) + cos(z + T*x)))
r = 4.77
implicit_plot3d(p, (x, -r, r), (y, -r, r), (z, -r, r),
plot_points=40)
```



[Wake this 3-D view]
[Click to Sleep this 3-D view]
[Help for Jmol 3-D viewer]

Applet

Functions

State

3-D display size:

Medium (400px)

- [Arbitrarily resizable in own window]
- [Get static image to save]

☐ Spin on

Image Compression

```
import pylab; import numpy
A_image = numpy.mean(pylab.imread(DATA + 'santarosa.png'), 2)
u,s,v = numpy.linalg.svd(A_image); S = numpy.zeros( A_image.shape
)
S[:len(s),:len(s)] = numpy.diag(s)
n = A_image.shape[0]
@interact
def svd_image(i = ("Eigenvalues (quality)",(20,
(1..A_image.shape[0]//2)))):
    A_approx = numpy.dot(numpy.dot(u[:,:i], S[:i,:i]), v[:i,:])
    g = graphics_array([matrix_plot(A_approx),
matrix_plot(A_image)])
    show(g, axes=False, figsize=(10,6))
```

```
    html("Compressed to %.1f%% of size using %s eigenvalues."%(
        100*(2.0*i*n+i)/(n*n), i))
```

Eigenvalues (quality) 20

Compressed to 14.1% of size using 20 eigenvalues.