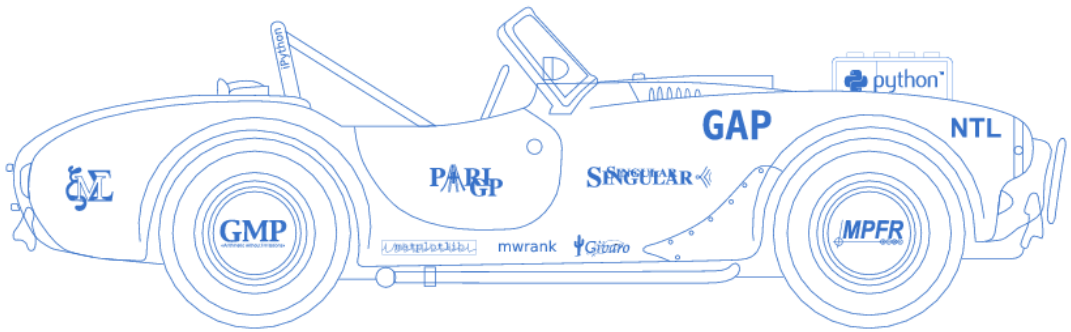
SAGE: Software for Algebra and Geometry Experimentation

William Stein

October 19, 2006, Current Problems Seminar at UW

http://modular.math.washington.edu/sage

SAGE
Building »The Car«



»Every free computer algebra system I've tried has

Outline

Introduction

Mathematics Software

Price for one **non-academic non-government** copy:

- Mathematica – **$1880 (or $3135)**
- Maple – **$1995**
- MATLAB - **$1900** (plus thousands of dollars for optional package),
- Magma is **$1150** (educational rate—the non-educational rate is much higher?).

After you graduate, this is what your employer or your student's employer might pay. These prices unduly inhibit math education and research in poor countries or schools.

The Problem

A Current Problem:

Create a powerful and free open source mathematics software package that beats the big commercial systems.

Create the "firefox" of math software. Take back the night. Statisticians did this with R. We can too.

This talk is about a project I initiated in January 2005 to address this problem.

Funding: use grant money, startup money, volunteer work, and **graduate and undergraduate student research** support. My "IP portfolio" is the existing world of free open source software.

History: Hecke 0.1 to SAGE 1.4.1

- **1997–1999:** Hecke – C++ math code for my thesis.
- **1999–2004:** Much MAGMA code for research.
- **Feb 2004:** Frustration with MAGMA.
- **Feb 2005:** I got job offers with **tenure** – **SAGE 0.1**
- **April 2005:** **Interfaces** to Mathematica, Magma, etc.
- **Feb 2006:** SAGE Days 1 workshop – **SAGE 1.0**
- **June 2006:** SIMUW workshop – **SAGE Notebook**
- **August 2006:** Grad student workshop – two week MSRI *graduate student* coding sprint.
- **October 2006:** SAGE Days 2 – workshop at UW.
- **Today:** SAGE now has a **wide range of functionality**. But **SAGE is not fast enough yet.**

Who is Writing SAGE?

I am a **number theorist** but SAGE has a **much wider** target area: **algebra, geometry, number theory, cryptography, numerical analysis, statistics, etc.**

Contributors Include: Martin Albrecht, **Tom Boothby**, **Robert Bradshaw**, Iftikhar Burhanuddin, Craig Citro, **Alex Clemesha**, John Cremona, Didier Deshommes, David Harvey, Naqi Jaffery, David Joyner, **Josh Kantor**, Kiran Kedlaya, David Kirkby, **Emily Kirkman**, David Kohel, Jon Hanke, **Robert Miller**, **Bobby Moretti**, Gregg Musiker, Fernando Perez, **Yi Qiang**, David Roe, Nathan Ryan, Kyle Schalm, Steven Sivek, Jaap Spies, Gonzalo Tornaria, Justin Walker, Mark Watkins, Joe Weening, Joe Wetherell

Implement it and send me a patch.

Some Fancy Hardware – A Collaboration Environment

http://sage.math.washington.edu

64GB RAM, **16 processor** Opteron server. You can browse the developer working directories over the web here!

What is SAGE?

What is SAGE?

1. **A Distribution** of free open source math software. 64MB source tarball that builds self-contained.
2. **New Code** that fill in gaps in functionality; implement new algorithms.
3. **A Unified Interfaces** to most mathematics software: MATLAB, Mathematica, Maple, etc.

SAGE runs on **Linux, OS X, and Windows**.

A Distribution

1. A Distribution

Basic Arithmetic	**GMP, NTL, MPFR, PARI**
Command Line	**IPython**
Commutative algebra	**Singular**
Database	**ZODB**
Graphical Interface	**jsmath, SAGE Notebook**
Graphics	**Matplotlib, Tachyon**
Group theory and combinatorics	**GAP**
Interactive programming language	**Python**
Networking	**Twisted**
Numerical computation	**GSL, Numeric, etc.**
Symbolic computation, calculus	**Maxima**

All core components are **free and open source**. You may change absolutely anything in SAGE or any of the libraries it relies on, build the new version and redistribute the result.

The SAGE Library – new code we've written

2. New Code

Over 100000 lines of Python and Pyrex code:

```
algebras     edu          lfunctions  monoids           sets
categories   ext          libs        plot              structure
coding       functions    matrix      quadratic_forms   tests
combinat     geometry     misc        rings
crypto       groups       modular     schemes
databases    interfaces   modules     server
```

```
UNIQUE Source Code Lines (including docstrings):
$ cat */*.py */*/*.py */*/*/*.py */*.pyx \
                */*/*.pyx */*/*.pyx |sort |uniq | wc -l
87513
UNIQUE Input Docstrings:
$ cat */*.py */*/*.py */*/*/*.py */*.pyx \
          */*/*.pyx */*/*.pyx |sort|uniq|grep "sage:" | wc -l
9759
```

A Unified Interface

3. A Unified Interface

- SAGE interfaces to:

  Axiom, GAP, GP/PARI, Kash, Macaulay2, Magma, Maple, Mathematica, MATLAB, Maxima, Octave, Singular, etc.

- Get access to the functionality of the other systems via interfaces.

- Get unified tab completion and online help even for many systems.

Overall Structure of SAGE

The Overall Structure of SAGE

- **Custom package management system** – 41 standard packages, and 29 optional ones. Automated upgrades.
- **Interactive command-line** interface – Python and IPython.
- **Graphical user interface** – an AJAX application: `http://sage.math.washington.edu:8100` or locally on your computer.
- **Underlying arithmetic** – from scratch; and built on top of standard libraries like GMP, PARI, GSL, etc. New code is written in Pyrex and Python.
- **Interfaces with other software** use **psuedo-tty**'s, which are extremely flexible.
- **Special purpose components** – e.g., Gfan (for computing Groebner fans), genus2reduction, GMP-ECM (for integer factorization), several *L*-functions programs.

Long-term Plans

Goals for SAGE 2.0

Main Goals for SAGE 2.0 (January 2007) – Speed up Arithmetic

See the trac server.

1. Make the **basic arithmetic**, e.g., finite fields, polynomials, etc., very fast.
2. Make the **everyday *exact* linear algebra** very fast.

This mainly involves **algorithm development**, because the only math software that does these things well now is **closed source**.

Next Year (2007) – SAGE 3.0: parallelism

Parallelism

1. Support for **parallelism** and use of it for algorithms, e.g., multimodular matrix multiply over $\mathbb{Q}$. Yi Qiang here at UW is working on something like GIMPS or Distributed.Net for SAGE.

2. I'm co-organizing a **workshop January 29–Feb 2** at MSRI on parallel computation.

Mathematics Software for Everybody

Questions?