

Breve introducción a Sage



Miguel Angel Marco Buzunariz
PyConEs Zaragoza
8 de Noviembre de 2014

Importante

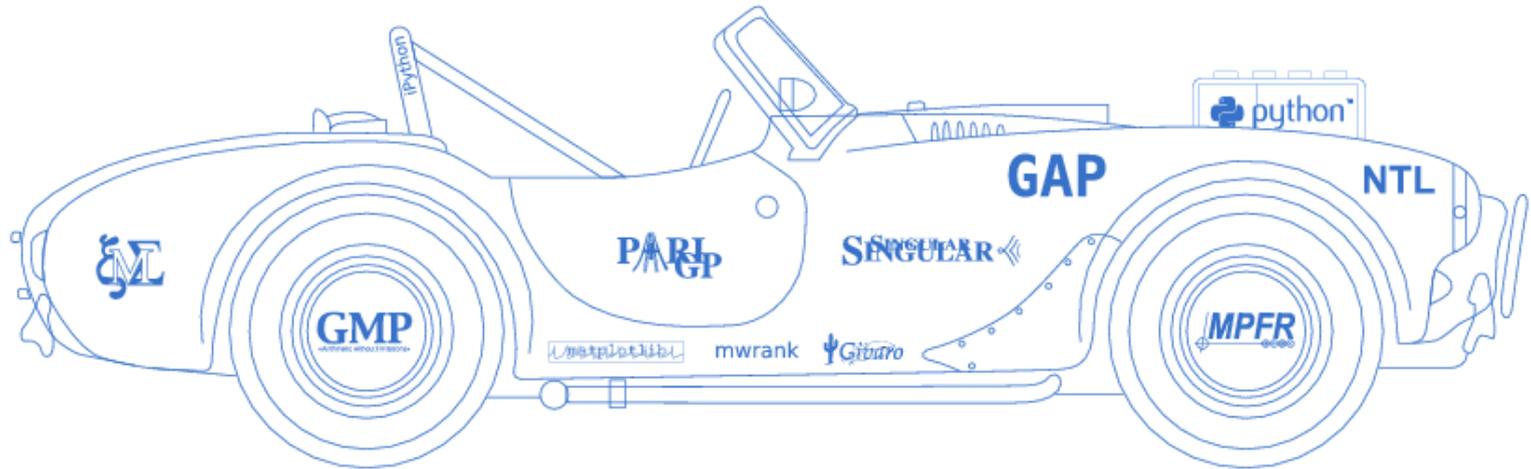


!!!!!! PREGUNTAS!!!!!!

¿Qué es Sage?

- Objetivo: crear una alternativa viable, libre y de código abierto a Magma, Maple, Mathematica y Matlab
- Interfaz web + línea de comando
- Representaciones gráficas (funciones, vectores...) en 2D/3D
- Aprovechando software libre ya existente

¿Qué es Sage?



»Every free computer algebra system I've tried has reinvented many times the wheel without being able to build the car.«

- No reinventar la rueda, sino construir el coche.

¿Qué es Sage?

- Lenguaje principal: Python
- Código crítico: pyrex/cython
- Teoría de números: Pari/GP, Flint, Givaro
- Teoría de grupos: GAP
- Manipulación simbólica: Maxima, SymPy
- Álgebra conmutativa: Singular, Polybori...

¿Qué es Sage?

- Álgebra lineal: ATLAS, ffblas-ffpack, iml, libm4ri, linbox...
- Teoría de grafos: Networkx
- Estadística: R
- Cálculo numérico: NumPy, SciPy...
- Representación gráfica: Matplotlib, jmol, Tachyon
- ...

¿Qué es Sage?

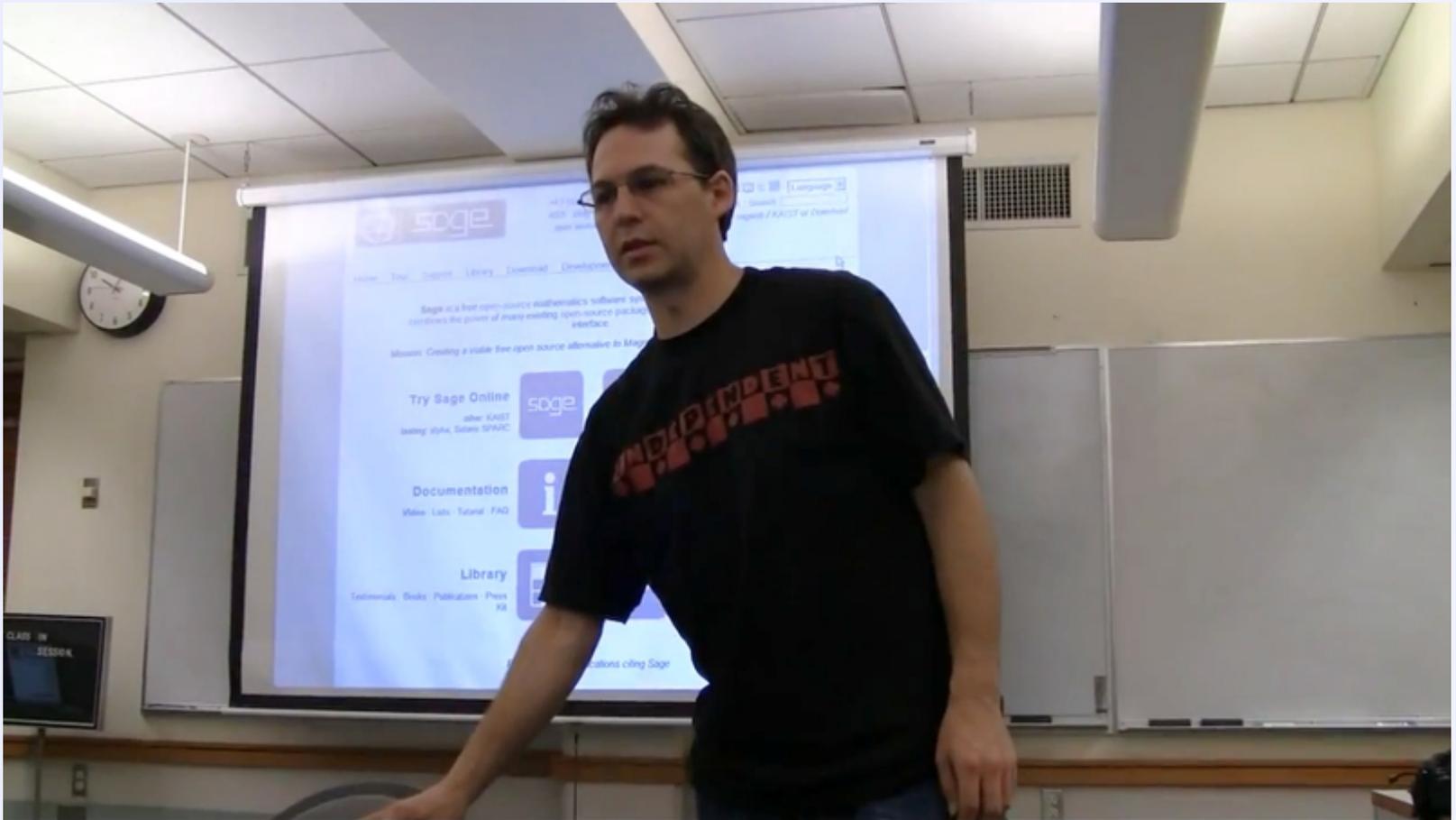
- Distribución de software libre matemático (GPLv3 en conjunto):
 - 92 paquetes estándar, 62 opcionales, 61 experimentales.
 - Scripts de instalación, manejo de dependencias...
- Biblioteca (Python) de funciones matemáticas (GPLv3+).
 - En muchos casos hacen uso de los paquetes de la distribución.
- Sistema de álgebra computacional (GPLv3+).
 - Interfaz web y línea de comandos.
 - Hace uso de la biblioteca (y por tanto de la distribución)

¿Qué es Sage?

- Paquetes mantenidos por la distribución:
 - Arch, Gentoo (overlay), Fedora, Ubuntu (PPA)
- Binarios descargables (descomprimir y ejecutar):
 - Ubuntu y Fedora (x86/x86_64/IA64/ARM/SPARK)
 - OSX (intel y PowerPC)
 - Solaris (x86)
- Código fuente (descomprimir, make, esperar y ejecutar)
 - Unix en general
 - Cygwin (con problemas)
- Windows mediante virtualización

Historia

- Iniciado en 2005 por William Stein



Historia

- Iniciado en 2005 por William Stein
- Había escrito código sobre curvas elípticas para Magma (propietario)
- En un momento dado, decidió iniciar una alternativa libre por:
 - Limitaciones con el lenguaje
 - Dificultad para compartir código con colaboradores
 - Imposible de modificar
 - Modelo de desarrollo más eficiente

Historia

“Puedes leer [un teorema] en [un libro] en la biblioteca [...] entonces puedes usarlo el resto de tu vida sin ningún cargo, pero para muchos sistemas de álgebra computacional hay que pagar licencias regularmente [...]. Pulsas botones y obtienes respuestas del mismo mofo que obtienes imágenes de tu televisor, pero no puedes controlar como se hicieron en ambos casos (...)

Historia

Con esta situación se **violan dos de las reglas de conducta más básicas de la matemática**: En las matemáticas, **la información se comparte gratuitamente**, y todo **se expone públicamente para ser comprobado**. No aplicar estas reglas a los sistemas de álgebra computacional usados para la investigación matemática [...] significa avanzar en una dirección de lo más indeseable. Peor: ¿podemos esperar que alguien se crea el resultado de un programa que no se le permite ver? Más aún: ¿realmente queremos cobrar a nuestros colegas en Moldavia varios años de su salario por su sistema de álgebra computacional? [...] E incluso: Si [unos matemáticos] tuvieran que pagar una licencia por usar una implementación de sus ideas [sobre ciertos objetos matemáticos], ¿no deberían tener derecho por tanto a cobrar una licencia por usar sus ideas en la implementación?

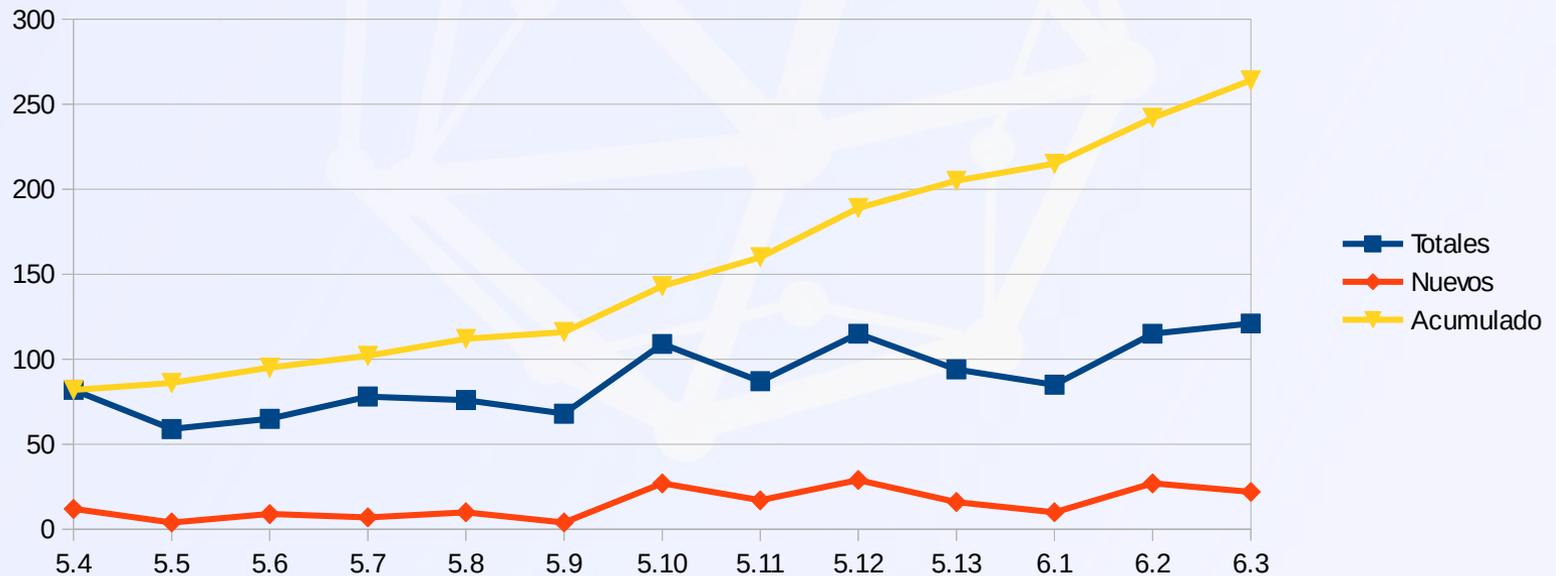
J. Neubüser in 1993

Historia

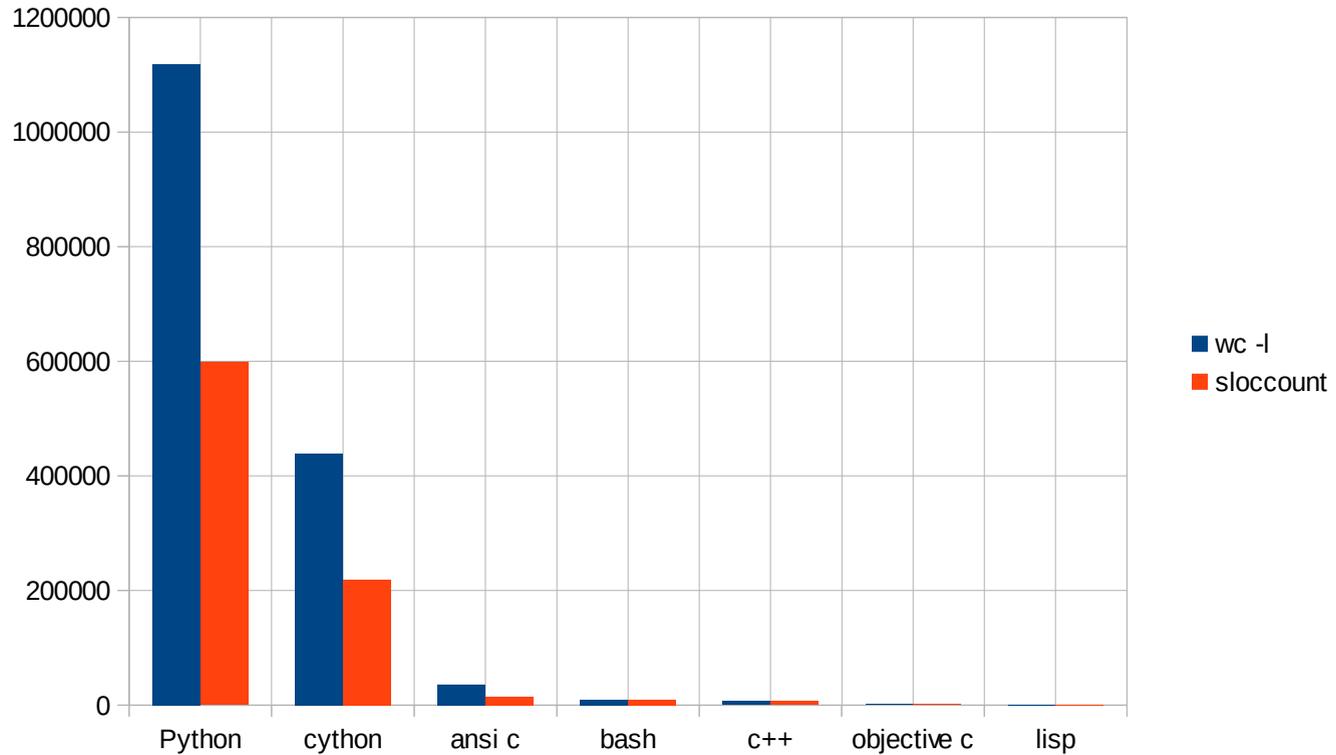
- Python:
 - Lenguaje moderno (comparado con Magma)
 - Mantenido por la comunidad
 - Ecosistema muy rico
 - Fácil de leer (no especialistas en programación)
 - Se puede extender con C/Pyrex/cython para tareas críticas
 - Adecuado como “pegamento” de otros sistemas

Desarrollo

- Desarrollo en comunidad + Sage Days.
- Release early, release often (unos dos meses)



Desarrollo



• 238 años-persona, \$32.163.081 (David A. Wheeler's 'SLOCCount')

Desarrollo

- Comunidad de voluntarios
- Modelo abierto
- Revisión por pares
- Políticas estrictas de documentación y doctests
- Fácil paso de usuario a desarrollador:
 - Mismo lenguaje
 - Herramientas incluidas (código fuente, git, scripts para sphinx y doctest...)

Desarrollo

- Comunicación entre desarrolladores:
 - Listas de correo (support, devel, release, flame)
 - Tickets en trac server
 - Release manager rotatorio
 - Premio anual Jaap Spies
 - Sage Days
 - GSoC
- 