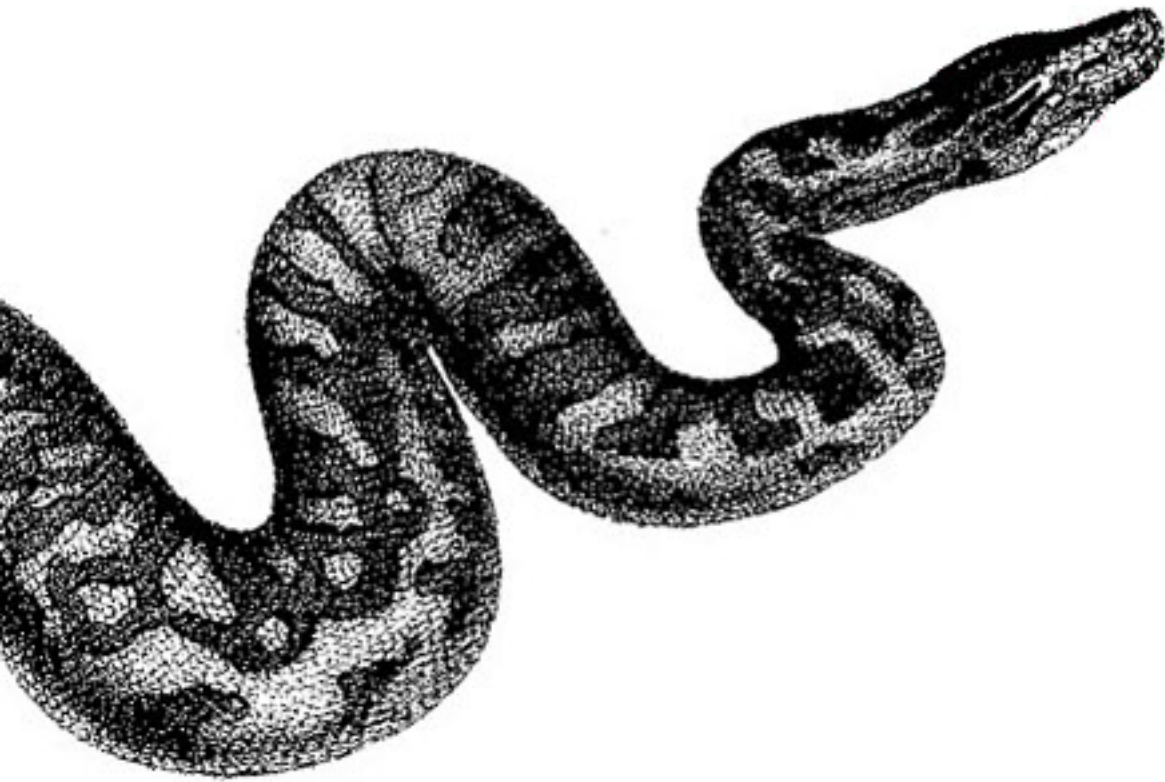


# Python: versões e ambientes



Luciano Ramalho  
[ramalho@python.pro.br](mailto:ramalho@python.pro.br)



# Linguagem de uso geral

- Internet: YouTube, Globo.com, Bitly, Mozilla...
- Computação gráfica: Disney, ILM, AutoDesk...
- Desktop: Dropbox, BitTorrent, OpenOffice...
- Operações: Google, Rackspace, Ubuntu, RedHat...
- Enterprise: IBM, Oracle, ERP5, OpenERP...
- Games, computação científica, segurança, etc...

[ABOUT](#)[SCHEDULE](#)[REGISTRATION](#)[VENUE](#)[SPONSORS](#)[JOBS FAIR](#)[BLOG](#)

# PYCON 2012

## SANTA CLARA, CA

PRESENTED BY

Google

heroku

Dropbox

**Tutorials****March 7<sup>th</sup> - 8<sup>th</sup>****Talks****March 9<sup>th</sup> - 11<sup>th</sup>****Sprints****March 12<sup>th</sup> - 15<sup>th</sup>**

facebook

GONDOR  
one **does** simply deploy

nasuni

Microsoft®

Eventbrite®

Events Made Easy

loggly



PLATINUM

PLATINUM

PLATINUM

PLATINUM

PLATINUM

GOLD

[Learn how to become a sponsor...](#)

## WHAT IS PYCON?

PyCon is the largest annual gathering for the community using and developing the open-source Python programming language. PyCon is organized by the Python community for the community. We try to keep registration far cheaper than most comparable technology conferences, to keep PyCon accessible to the widest group possible. PyCon is a [diverse conference](#) dedicated to providing an enjoyable experience to everyone. Our [code of conduct](#) is intended to help everyone maintain the PyCon spirit. We thank all attendees and staff for observing it.



# SPONSORS



# Sobre Python

- Linguagem dinâmica
  - compilador é parte do ambiente de runtime
- Tudo são objetos
  - Ex.: inteiros, funções, classes, exceções etc.
- Multi-plataforma
  - Interpretador, APIs, bytecode, GUIs etc.

# Implementações

- CPython: a principal, escrita em C
  - incluída na maioria das distros GNU Linux e no Mac OS X; vários pacotes para Windows
- Jython: escrita em Java
  - parte de IBM WebSphere e Oracle WebLogic
- IronPython: escrita em C#, .net CLR
  - desenvolvida pela Microsoft

# Implementações (2)

- PyPy: Python em Python
- Implementação em linguagem de alto-nível facilitando a inovação
- Desempenho 5x melhor que CPython
- Incompatível com as extensões em C

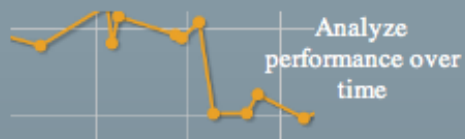


## Changes

|                  |         |        |     |
|------------------|---------|--------|-----|
| test_iteration   | 0.05111 | 2.12%  | -1  |
| ico              | 0.2226  | -3.30% | -3  |
| ingo             | 0.2842  | 0.90%  | -17 |
| itfire_cstringio | 4.4612  | 5.12%  | 5   |
| itstd_glo        | 0.0525  | -1.39% | -2  |
| itstd_names      | 0.0063  | -2.00% | -2  |
| itlib            | 10.1520 | 2.73%  | 2   |

Track performance changes in the latest revisions

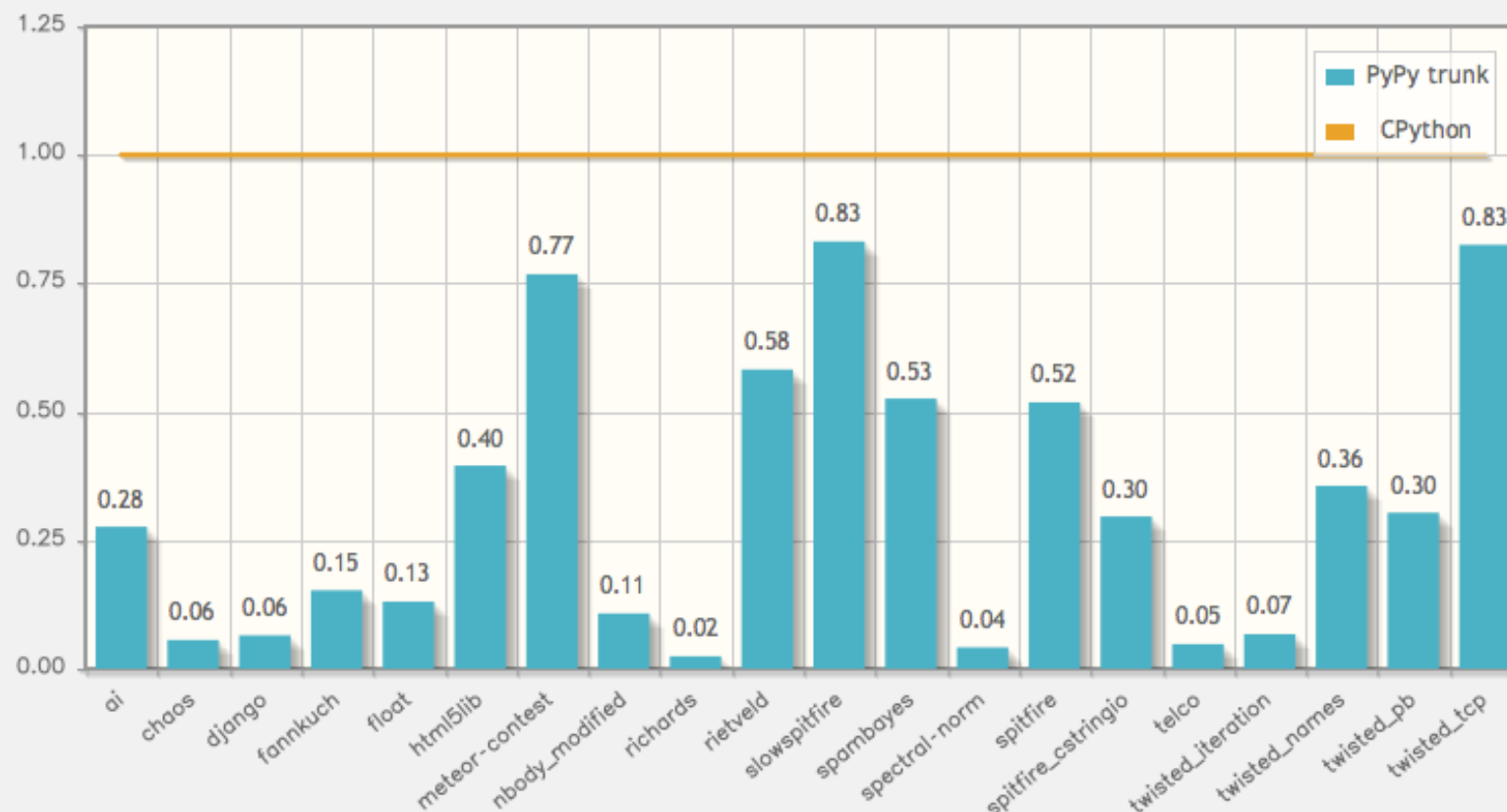
## Timeline



## Comparison



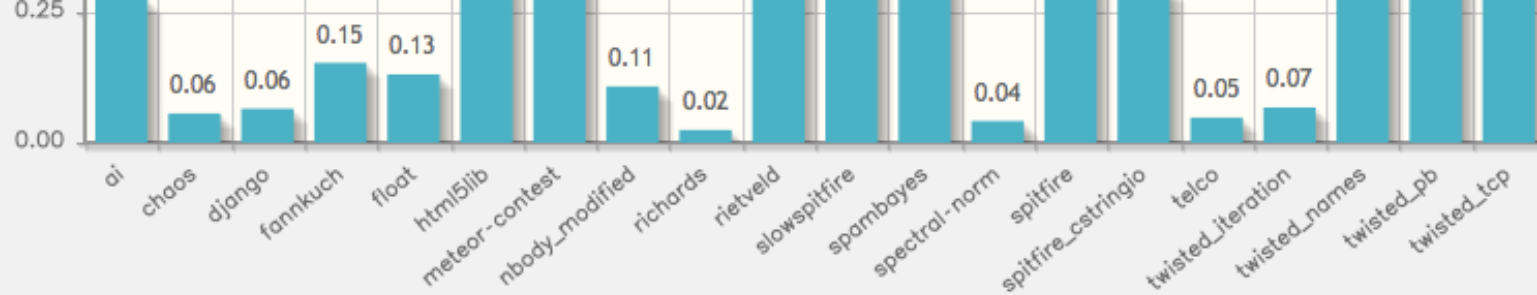
## How fast is PyPy?



Plot 1: The above plot represents PyPy trunk (with JIT) benchmark times normalized to CPython. Smaller is better.

It depends greatly on the type of task being performed. The geometric average of all benchmarks is 0.20 or **5.1 times faster** than CPython.

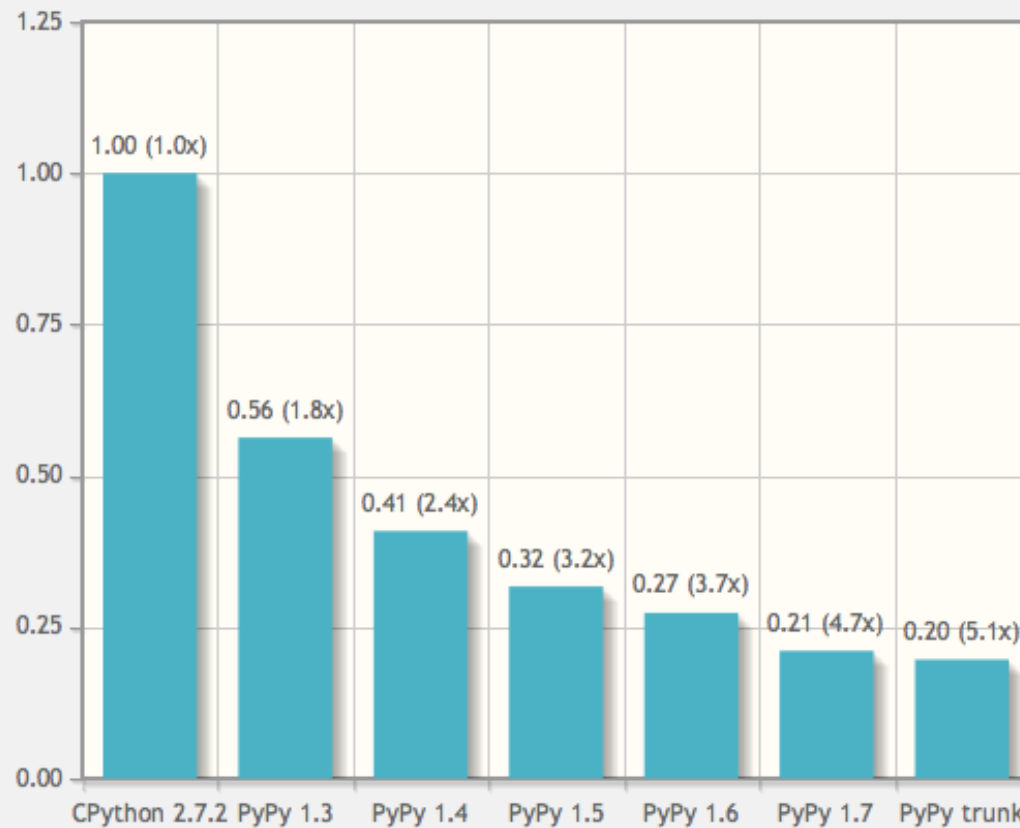




Plot 1: The above plot represents PyPy trunk (with JIT) benchmark times normalized to CPython. Smaller is better.

It depends greatly on the type of task being performed. The geometric average of all benchmarks is 0.20 or **5.1 times faster** than CPython

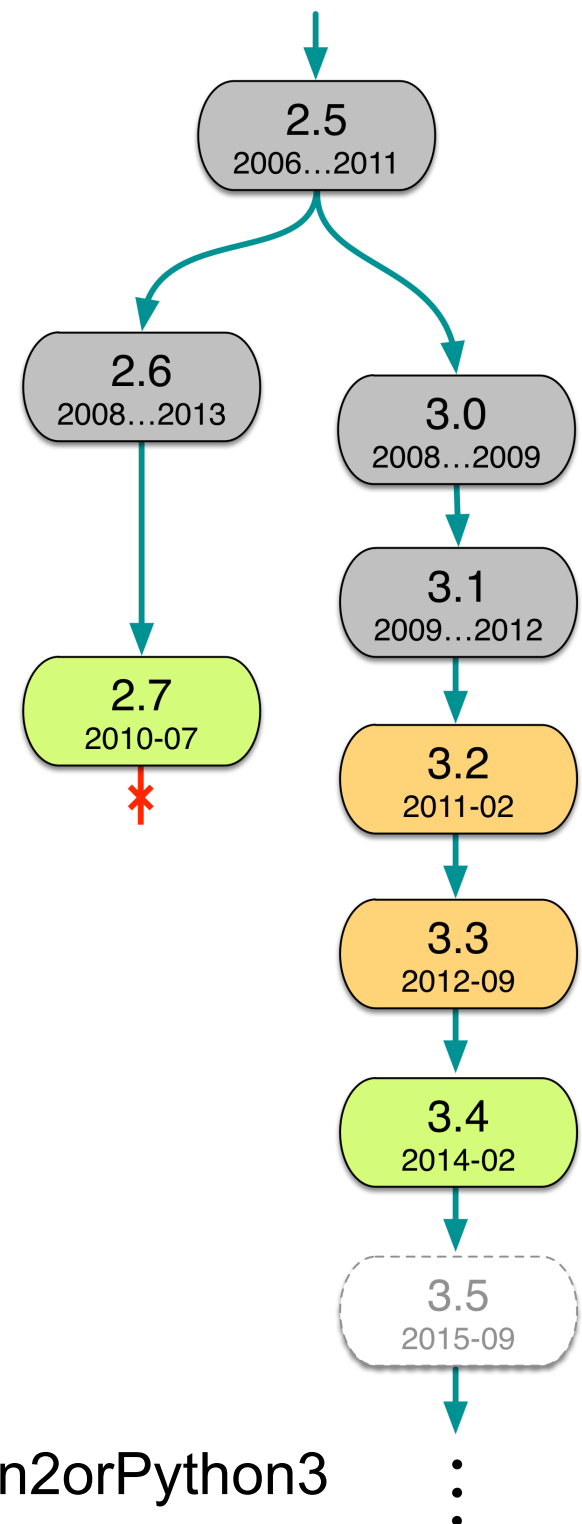
### How has PyPy performance evolved over time?



Plot 2: Geometric averages of normalized times, out of 20 benchmarks. Smaller is better. "times faster" inside parenthesis

# Python: evolução

- Versões do CPython em jan/2015
- 2.6, 3.1 e anteriores: end-of-life
- 3.2 e 3.3: em manutenção
- 2.7: versão final da série 2.x, será mantida até 2020
- 2.7, 3.4: versões atuais
- 3.5: beta previsto para mai/2015, lançamento em set/2015



# CPython: instaladores oficiais

- GNU Linux: pré-instalado em quase toda distro
  - usar gerenciador de pacotes da distro ou compilar (é fácil)
- Windows
  - Python.org: instaladores MSI para 32 e 64 bits
- Mac OS: CPython 2.7 pré-instalado (64 bits)
  - Instaladores 32 bits (x86 e PPC), 64 bits (x86) ou compilar (é fácil depois de instalar o Xcode)

# CPython: instaladores + extras

- Instaladores com muitos pacotes pré-instalados
- Multi-plataforma (Linux, OSX, Windows):
  - Enthought Canopy
  - Anaconda Python
- Somente Windows
  - WinPython
  - PythonXY
- Outros:
  - <https://www.python.org/download/alternatives/>



# Consoles interativos

- python
- integrado em IDEs
- iPython
- bpython
- IDLE



# IDEs, algumas opções

- PyDev (Eclipse)
- PyCharm (JetBrains) \$\$\$\*
- Komodo Edit
- Komodo IDE \$\$\$
- TextMate \$ - OSX
- SublimeText \$
- WingIDE \$\$\$

# Em vez de IDE: The Unix environment

- Linux ou OSX ou qualquer Unix:
  - janelas de editor, console e navegador ou sua aplicação
  - alternar entre janelas com alt-tab: simples, poderoso e eficaz
- Emacs: python-mode.el
- vi:  
[http://www.vex.net/~x/python\\_and\\_vim.html](http://www.vex.net/~x/python_and_vim.html)
- Geany: um Gedit mais esperto para lidar com vários arquivos



# Configure o editor para...

- Indentar com 4 caracteres de espaço ao usar a tecla TAB ou comandos de indentação multi-linha
- Salvar tabs como 4 espaços, nunca como tabs
- Limpar brancos no final das linhas ao salvar
- Indentação inteligente:
  - preservar indentação da linha acima
  - indentar automaticamente após: if, elif, else, for, while, try, except, finally, def, class, with