

PyPy

Maciej Fijalkowski
merlinux GmbH

PyCon PL 2008

October 18 2008



O czym bede opowiadal

- Co to jest PyPy?
- Dlaczego?
- Co chcemy osiagnac?
- Jak nam idzie?

Kim jestem?

- Maciej Fijałkowski
- od około 2 lat zajmuje się głównie PyPy
- pracuje dla Merlinix GmbH
- nie umiem ustawić polskich liter...

Co to jest PyPy?

- interpreter pythona napisany w pythonie
- framework do pisania interpreterow dynamicznych jezykow
- open source software
- dzis bede mowil glownie o interpreterze

Po co nam to?

- CPython jest fajny, ale ciezko zmienic pewne decyzje (np garbage collector)
- Napisany w C
- Srodowisko, ktore umie pythona, a do rozwijania jezyka potrzebni sa ludzie biegli w C
- Projekty takie jak Psyco czy stackless ciezko jest utrzymac w zgodnosci z najnowszym Pythonem.
- Problem z podzialem srodowiska (CPython, Jython, IronPython)

Wiec co?

- Napisac interpreter pythona w pythonie!
- Zdefiniowac podzbior pythona ktory sie nadaje do statycznej kompilacji
- Skompilowac go do C/LLVM/JVM/.NET
- My uzywamy okrojonego pythona do pisania interpretera, wiec ty juz nie musisz!

Metodologia

- Test Driven Developement
- Organizujemy sprinty raz na 2/3 miesiace
- Rozproszona grupa (glownie Europa)
- Mnostwo testow, nocne testy (buildbot)

Status na dziś

- 2.5
- ctypes (np pysqlite-ctypes, pyglet, pymunk ...)
- django
- twisted/nevow
- wiekszosc testow CPython'a
- sympy
- twój program...

Czego brak

- czesci modulow (napisanych w C)
- bibliotek ktore maja czesci w C (np PIL, pygame, numpy, ...)
- wydajnosci

Zabawy z odpalaniem programow

- przyklady

Zabawy z odpalaniem programow

- lekcja: nie istnieje nic co jest zbyt obskurne zeby ludzi na tym polegali

Wydajność

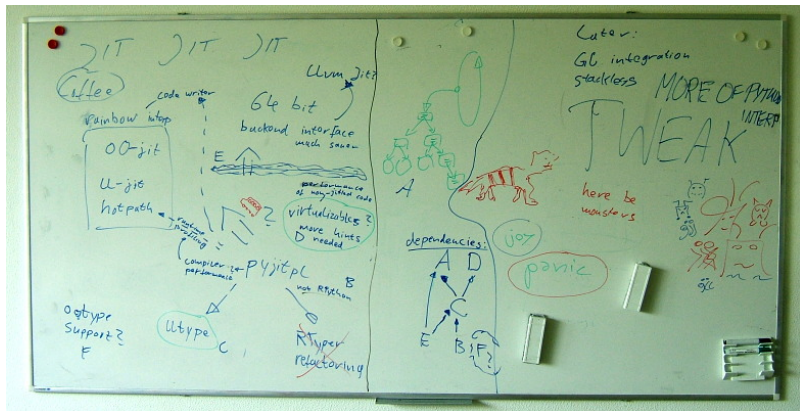
- PyPy jest od 20% szybszy do 4x wolniejszy, zależy od aplikacji
- gcbench - szybszy (lepsze Garbage Collector)
- nadzieja w dynamicznej kompilacji

Wydajnosć - GC

- jeden z większych sukcesów PyPy
- w miarę szybki GC (wolniejszy niż np. JVM)
- trochę szybciej od CPython'a, mimo że alokujemy dużo więcej.
- możliwość zmiany, dla specjalnych zastosowań

Wydajnosć - JIT

- nie jest jeszcze gotowy

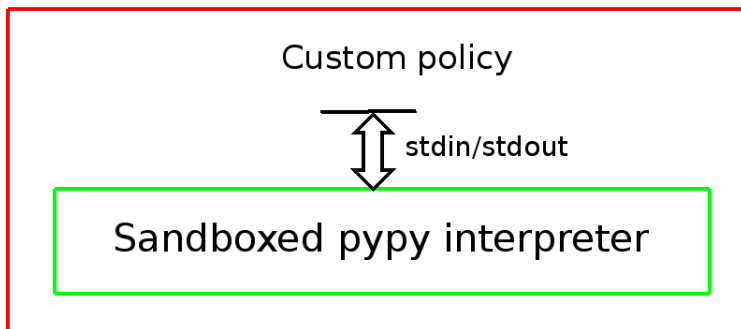


Sandbox

- pelen python
- dowolna polityka bezpieczeństwa
- mozliwosc limitowania CPU i pamieci
- duzo trudniejszy do segfaultowania od CPython'a
- demo

Sandbox - jak to dziala

Normal Python interpreter



Maemo

- Mniejsze zużycie pamięci per-objekt niż cpython
- Możliwość wpięcia specjalnego Garbage Collectora
- Szybszy czas startu interpretera
- Ale np. większy rozmiar kodu, wolniejszy kompilator

Przyszlosc

- JIT!
- Pelne wsparcie dla 2.5
- Zorganizowanie wiekszej ilosci funduszy
- Mozesz pomoc! (to nie jest takie trudne)

Q&A

- <http://codespeak.net/pypy>
- <http://morepypy.blogspot.com>
- można sciagnac te slajdy z internetu