Fast Python? Don't Bother

Russel Winder

russel@winder.org.uk
https://www.russel.org.uk
@russel_winder

There is a rumour I only come to PyCon UK to talk about languages other than Python...

...this is not true...

...Well not entirely.

A gentle rant, but...

...a bit of a rant nonetheless.

Python is...

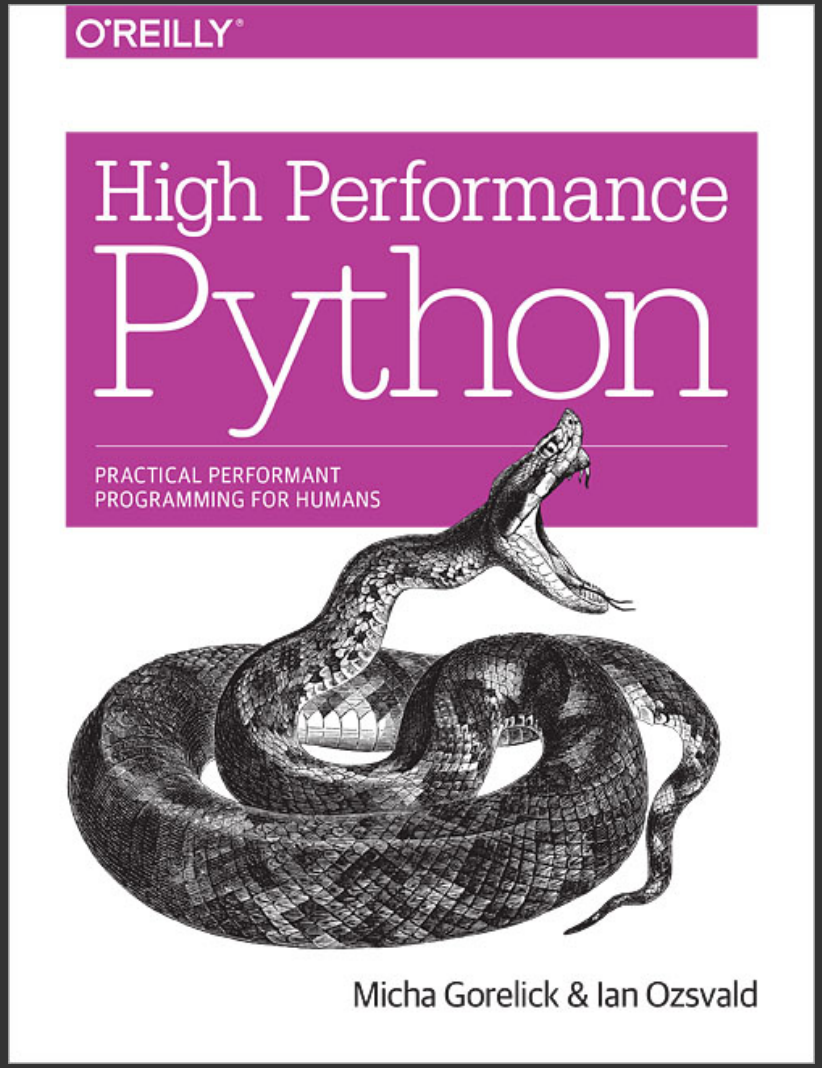
- A dynamically typed (or not) language.
- Quite popular, in data science, and data visualisation.
- Quite popular for sys. admin.
- Quite popular for web-y stuff.
- Having it's development driven by IDEs.

Type signatures.

Slow

Python is not...

- A high performance computational language.
- A language in which you **can** write parallel programs.



Maybe the GIL is a good thing,

it stops us using Python when we shouldn't.

Why?

Parallelism

So how to do computation?

- Cython
- Numba
- NumPy

Cython

- Write Python and annotate the code to make it appear like C so it is actually C by source-to-source translation.

Hats off to Pyrex.

Why not just write C if you want to write C?

Because C is a programming language for writing operating systems, not applications?

What we want is Fortran or even FORTRAN.

Or not.

Numba

- Write Python code and decorate with decorator so as to perform non-Python translation via LLVM to native code.

Does Python really have native code semantics?

Does RPython really have native code semantics?

No

NumPy

- And the whole SciPy, Pandas, etc. kit and caboodle.
- A C subsystem, providing opaque data types and operations well integrated with Python as an API.

It's just a C system with a nice API.

Lots of niceness to the API.

Is C good enough for computation

in the modern era?

No

Why?

Parallelism

Multi-multicore processors

GPGPU

~~OpenGL~~

Vulkan

~~CUDA~~

Actors

CSP

Dataflow

Data Parallelism

Parallel Processing

- Threads and thread pools:
  - C — but very low level.
  - C++ — better than C.
  - Java: Akka, Quasar, Hadoop, Apache Spark, GPars.
  - D — actors and data parallelism.
- Partitioned Global Address Space (PGAS):
  - Chapel
  - X10

There are man, many more: Pony, Nim, etc.

Where does Python fit in?

It doesn't for the computation...

...but there is a way forward:

Microservices Architecture

Processes communicating over a network.

Processes on same computer communicating using inter-process communication.

Processes communicating is the only way pure Python code can create parallelism so it must be Pythonic.

Intermix Python processes with non-Python ones.

Intermix:

- Python

- C
- C++
- Rust
- D
- Chapel

Peek at D and Chapel.

https://github.com/russel/Pi_Quadrature

Follow Up...

Short hands-on workshop:

Monday 2016-09-19T14:30+01:00

Room A

Fast Python? Don't Bother

Russel Winder

russel@winder.org.uk
https://www.russel.org.uk
@russel_winder