

Python is only slow if you use it wrong

Avery Pennarun <apenwarr@gmail.com>

Google Inc.

*This presentation absolutely, positively, in no way at all,
could ever possibly begin to reflect the opinion of my employer*

Who says?

- bup: backup software with 80 megs/sec thruput - in python
 - <http://github.com/apenwarr/bup>
- sshuttle: VPN software that easily handles 802.11g/n speeds - in python
 - <http://github.com/apenwarr/sshuttle>

The Easiest Way to Use Python Wrong

TIGHT INNER LOOPS.

A line of python code is **80-100x slower** than a line of C code.

```
s = open('file').read()  
for char in s:  
    ...
```

JUST SAY NO.

The Easiest Way to Make Python Fast

- Use regexes and C modules.
- No such thing as “100% pure python.”
- Forget about swig: writing C modules is easy.
- **python + C** together is, so far, the winning combination.
- C is simple; python is simple; pypy is hard.

The Other Way to use Python Wrong

- Computation threads
 - Worthless because of GIL
- Threads are ok for I/O
- `fork()` works great for both
- C modules that use threads are fine

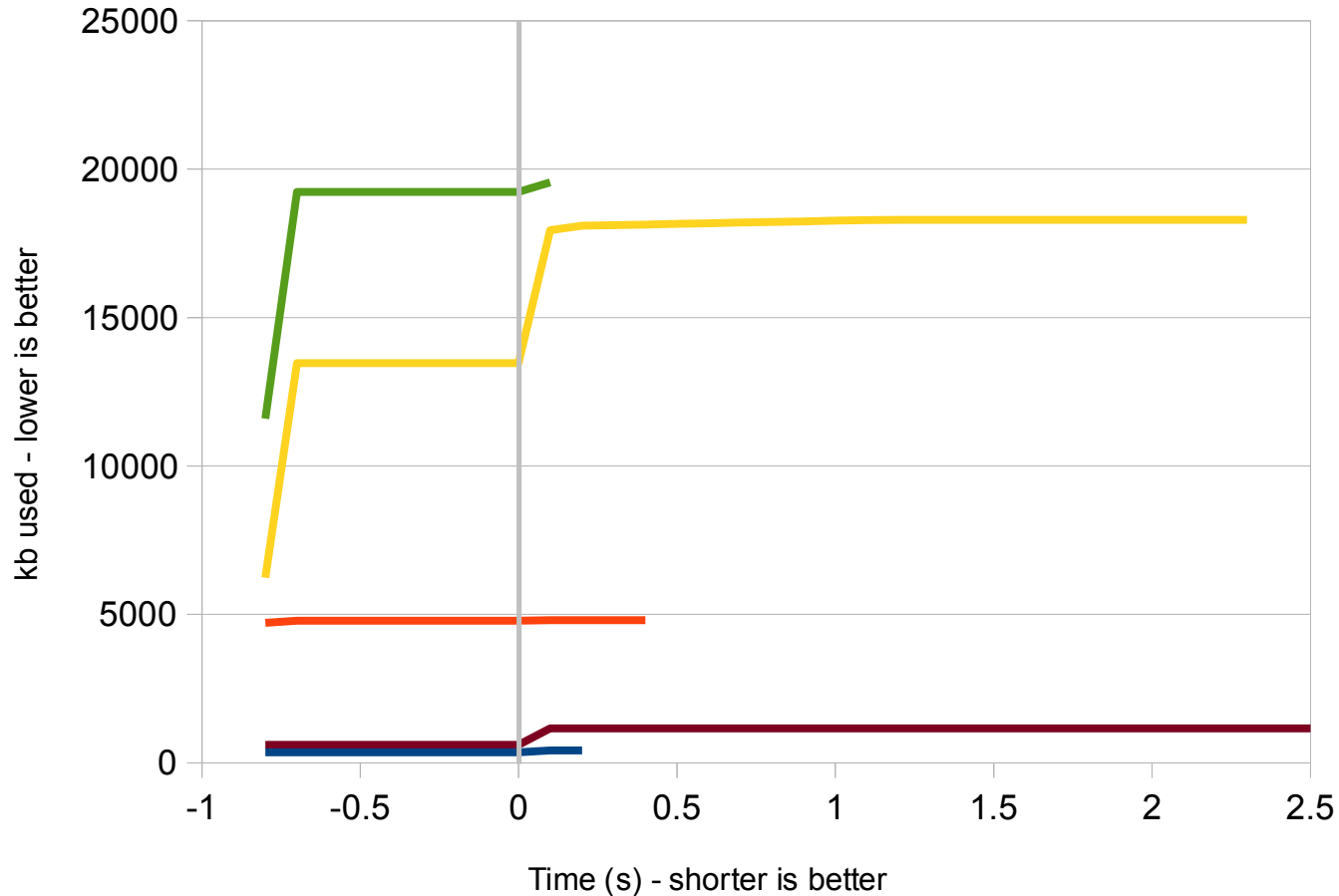
Garbage Collection

RefCounting... and threads

- A bad combo
- You would need locks around every single variable access
- One reason why removing the GIL is almost impossible
- There are tricks...

Python is not a garbage collected language (*)

MemTest array[1] Benchmark



— C — Go — python — java-client — pypy

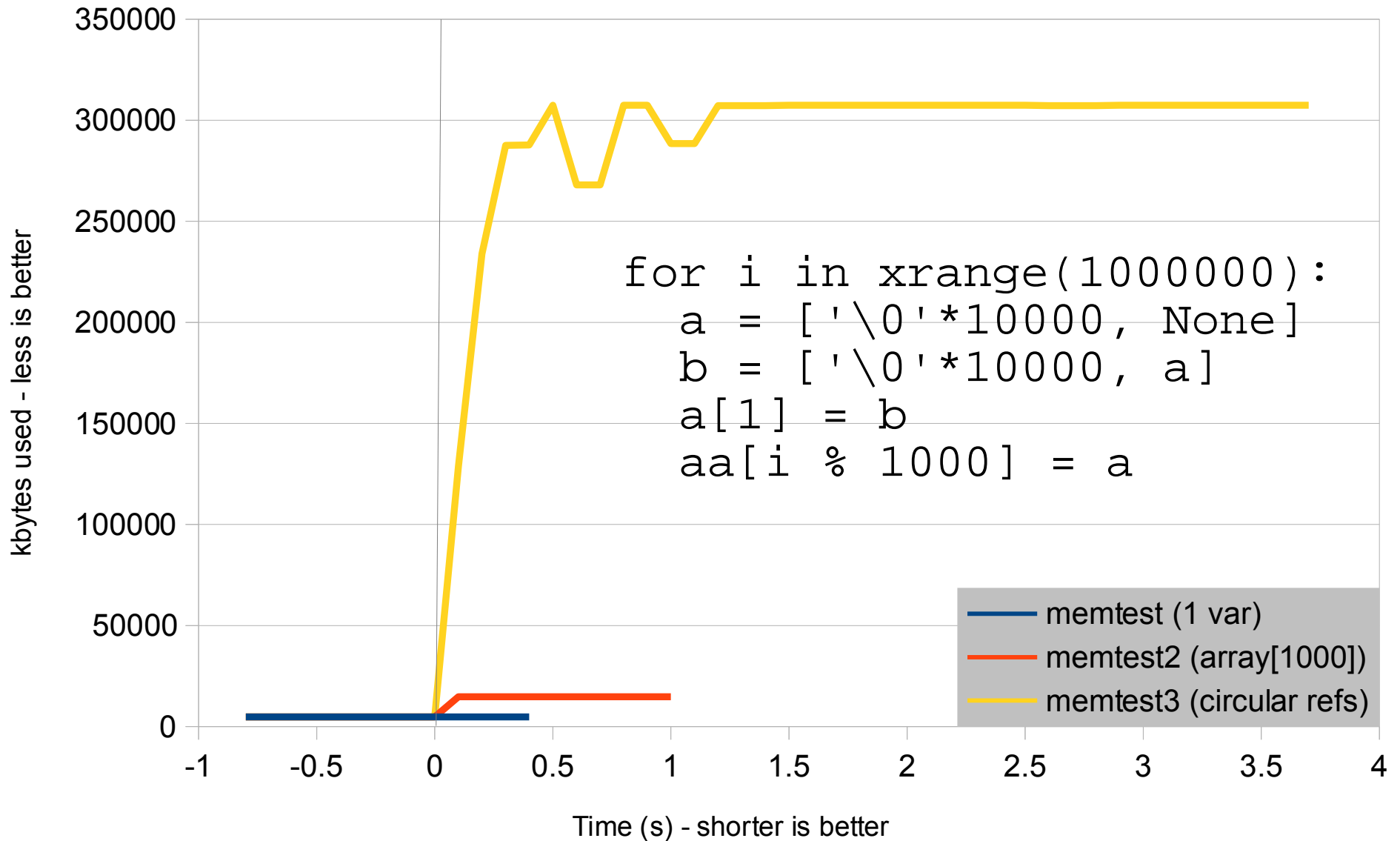
```
for i in xrange(1000000):  
    a = '\0'*10000
```


Java is a garbage collected language

MemTest array[1] Benchmark



(*) Except sometimes python is a garbage collected language



Getting the Most out of Python's GC

JUST AVOID IT AT ALL COSTS.

- Break circular references by hand when you're done.
- Better still: use the **weakref** module.

Deterministic destructors

- Quiz: does this program work on win32?

```
open('file', 'w').write('hello')  
open('file', 'w').write('world')
```

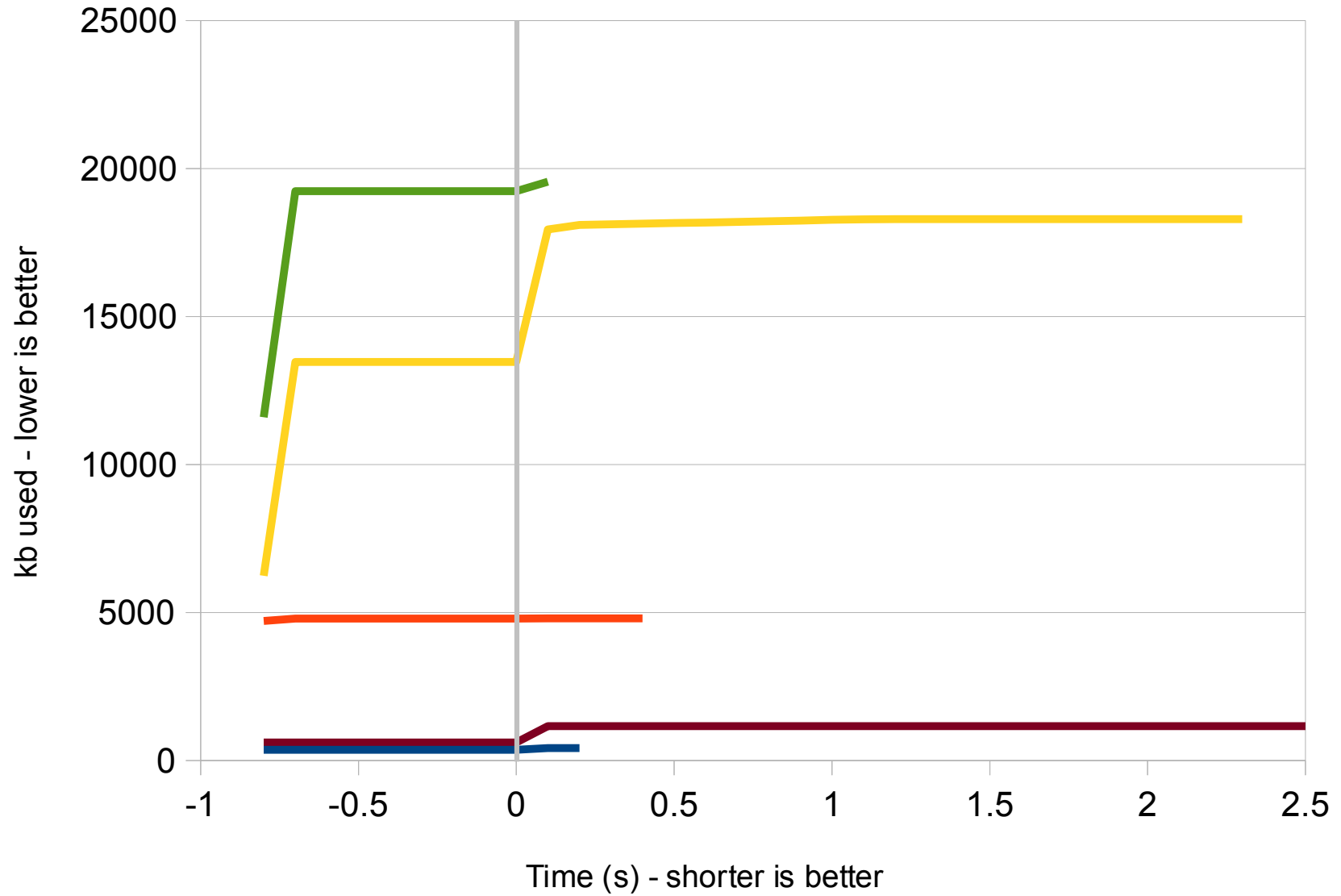
- With “real” gc, you have to manually manage resources:
 - files
 - database handles
 - sockets
 - locks

Ruffians & Vagabonds are trying to take away your Deterministic Destructors!

- Some people claim “real gc” is the “right solution”
- But what they mean is “it's the easiest way to do python on the java or .net runtime”
- “with” statement is powerful, but not good enough

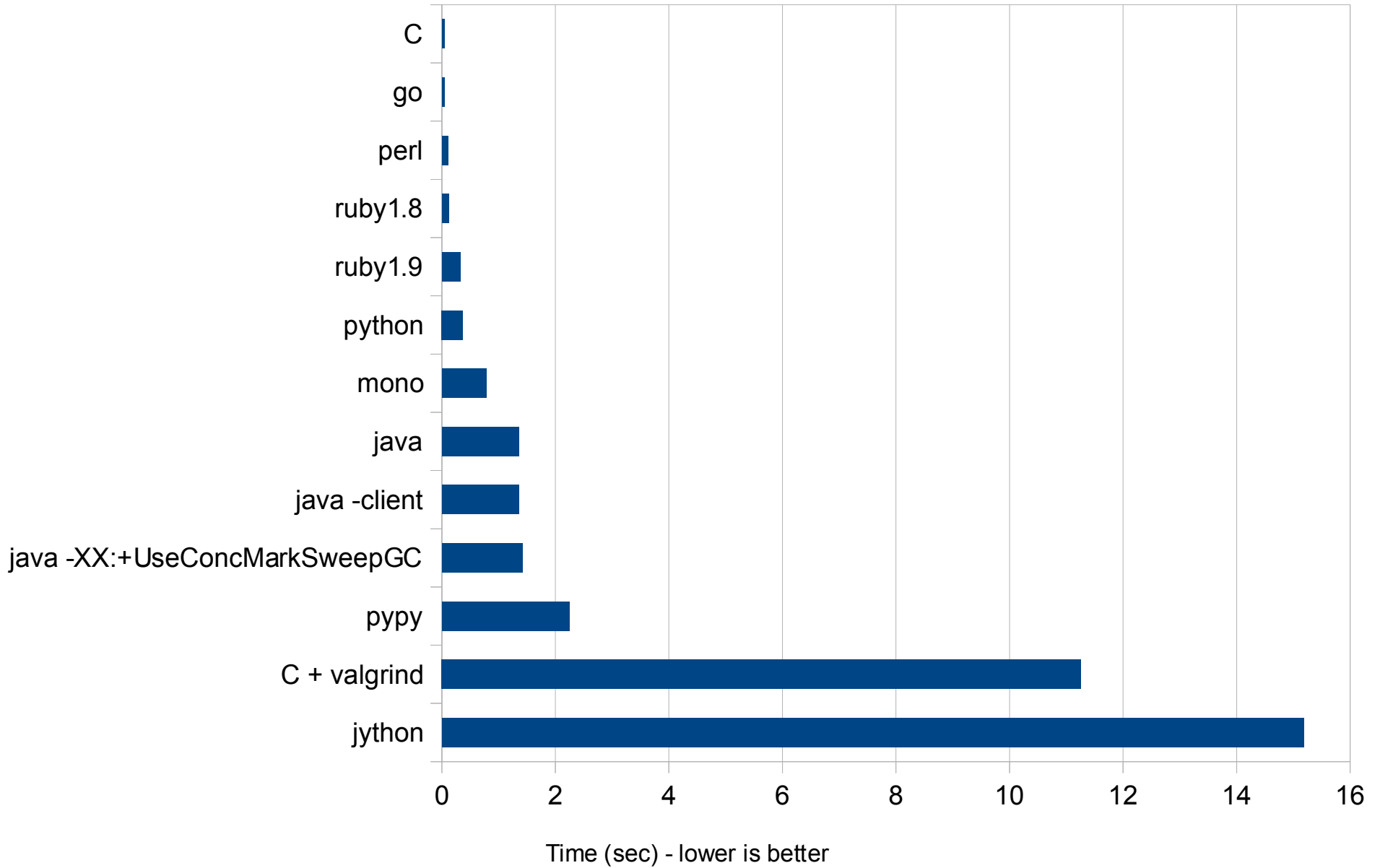
Interpreted vs. JIT

MemTest array[1] Benchmark



— C — Go — python — java-client — pypy

Hello World Benchmark - HelloMark (tm)



** many git commands run in about 2x the time of C hello world*

.pyc files

- are awesome

Summary

- Love reccounting, hate gc
- Don't write tight inner loops: that's what C is for
- If you need a JIT, you're doing it wrong
 - ...even if the JIT is good
- Let's keep working on that startup time

<http://github.com/apenwarr/avebench>