



# **Serveur d'applications à haute disponibilité**

17/09/2011 - Pyconfr  
~~Tarek Ziade~~ & Benoît Chesneau

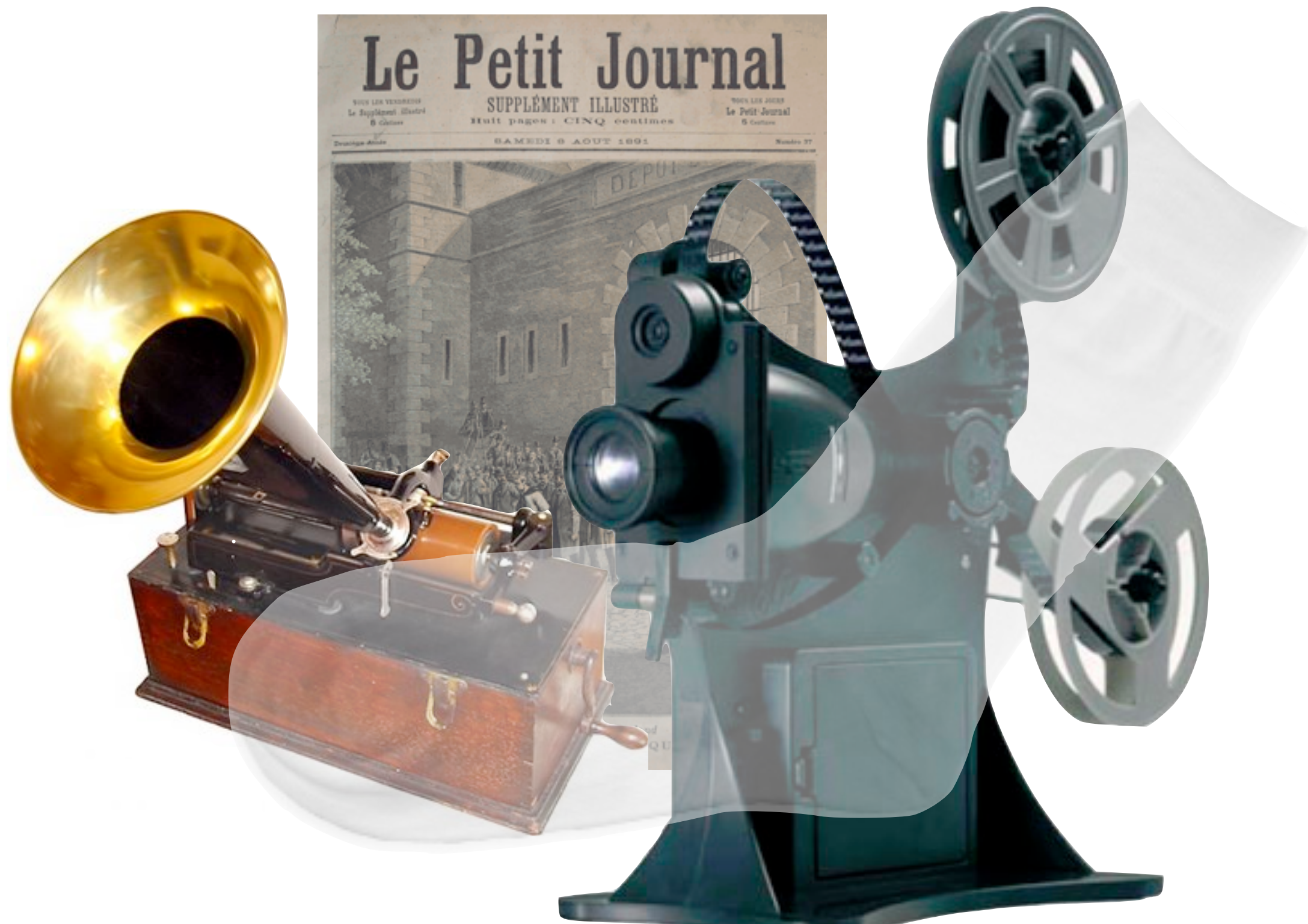
- Le problème c10k
- “It's time for web servers to handle ten thousand clients simultaneously, don't you think? After all, the web is a big place now.”
- <http://www.kegel.com/c10k.html>





Toujours plus de monde





Toujours plus de media

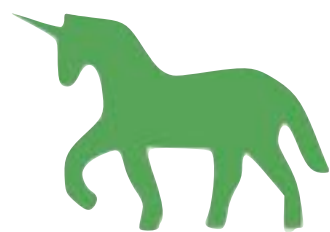
- Plusieurs requêtes par secondes
- Optimisations
  - Matériel
  - Temps de réponse
  - Concurrency
  - Impact de la base de données

**Haute disponibilité et concurrence**

APACHE  
HTTP SERVER



- 1993 - 1998 - CGI (et applets java...)
- Apache & Prefork - simple
- Multiprocessing - threads vs events



gunicorn

Un peu d'histoire



[Click here for advertising information - reach millions every month!](#)

Search  and Display the Results

Search with Digital's Alta Vista [ [Advanced Search](#) ] [ [Add URL](#) ]

[Download free demo versions of AltaVista Technology software](#)



---

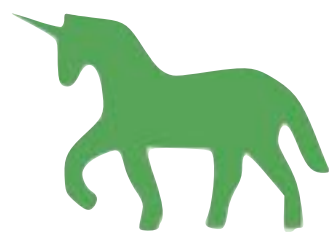
[[Creative](#)][[Search](#)][[Humor](#)][[Email](#)]

---

APACHE  
HTTP SERVER



- 1993 - 1998 - CGI (et applets java...)
- Apache & Prefork - simple
- Multiprocessing - threads vs events



gunicorn

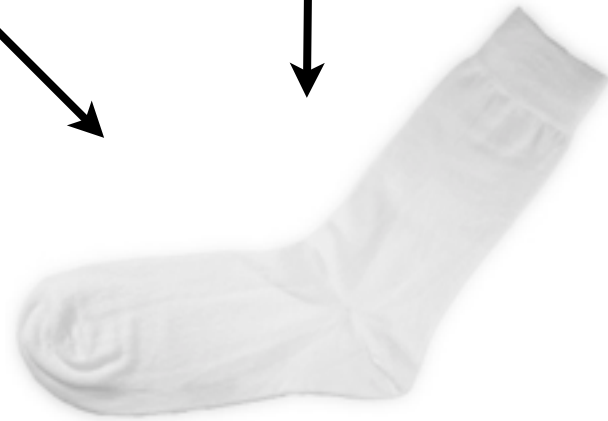
Un peu d'histoire



Clients



Socket



Port 80

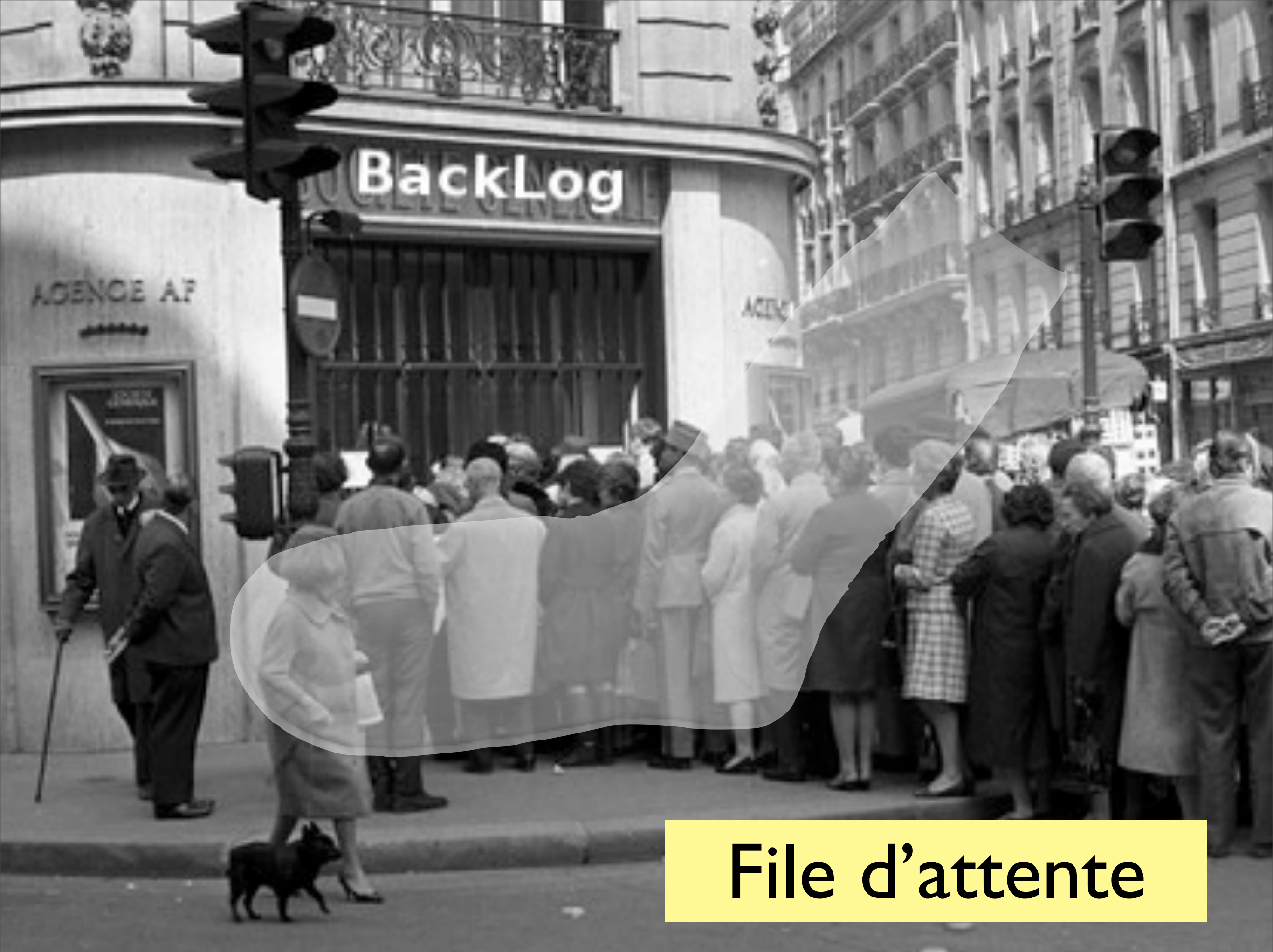
Worker



Application



Minitel



**File d'attente**



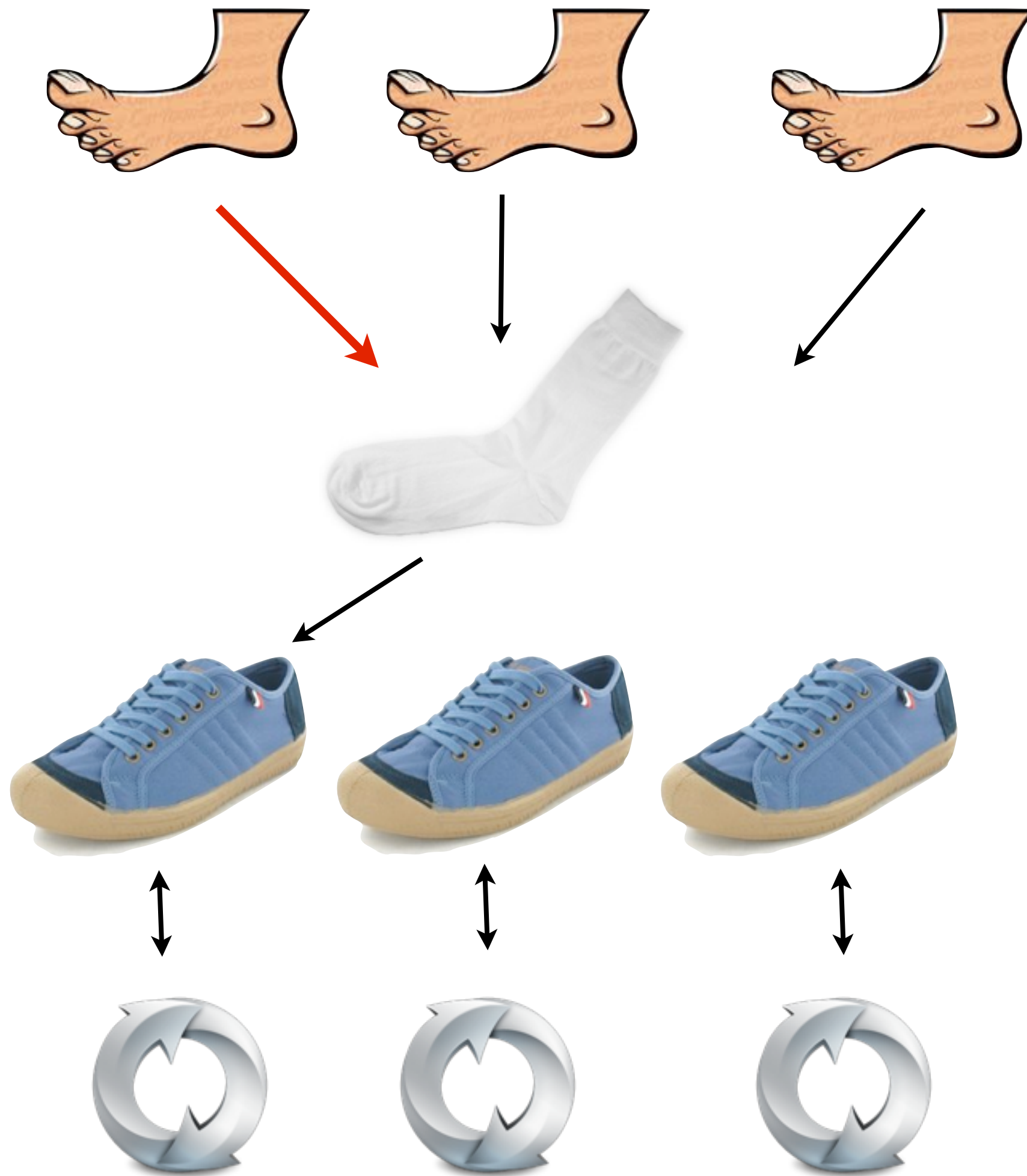


Moar

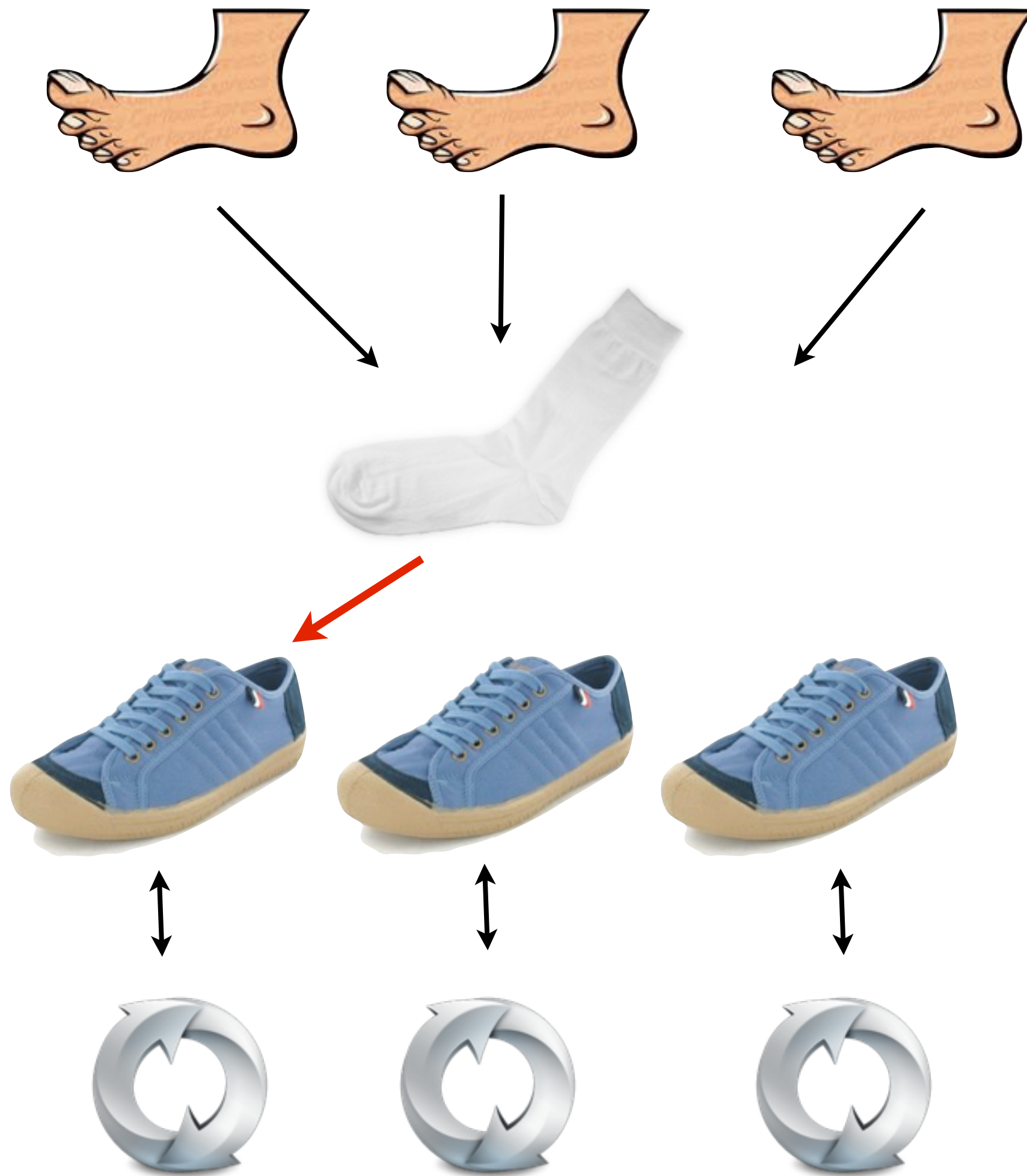




Fork

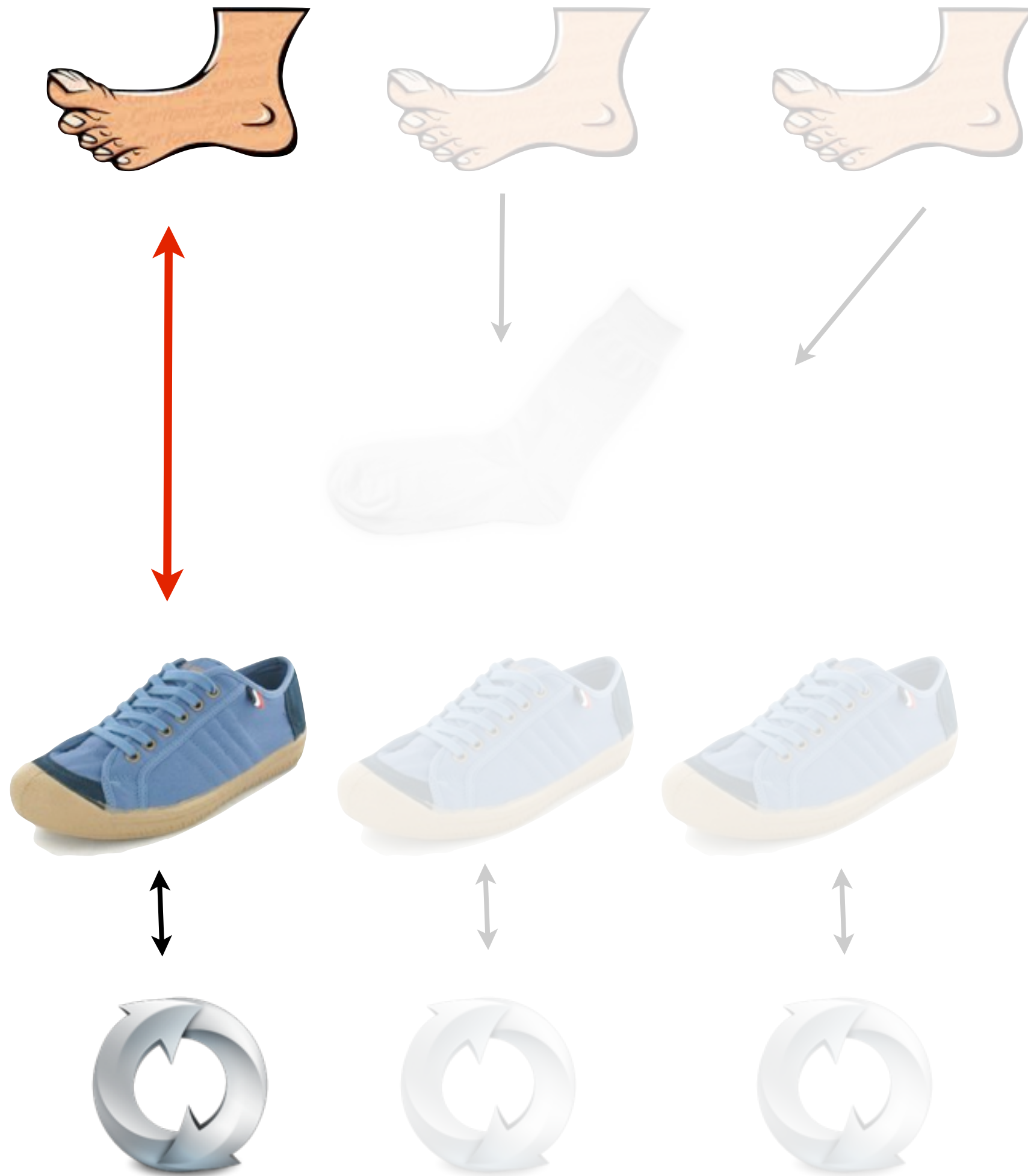


Fork

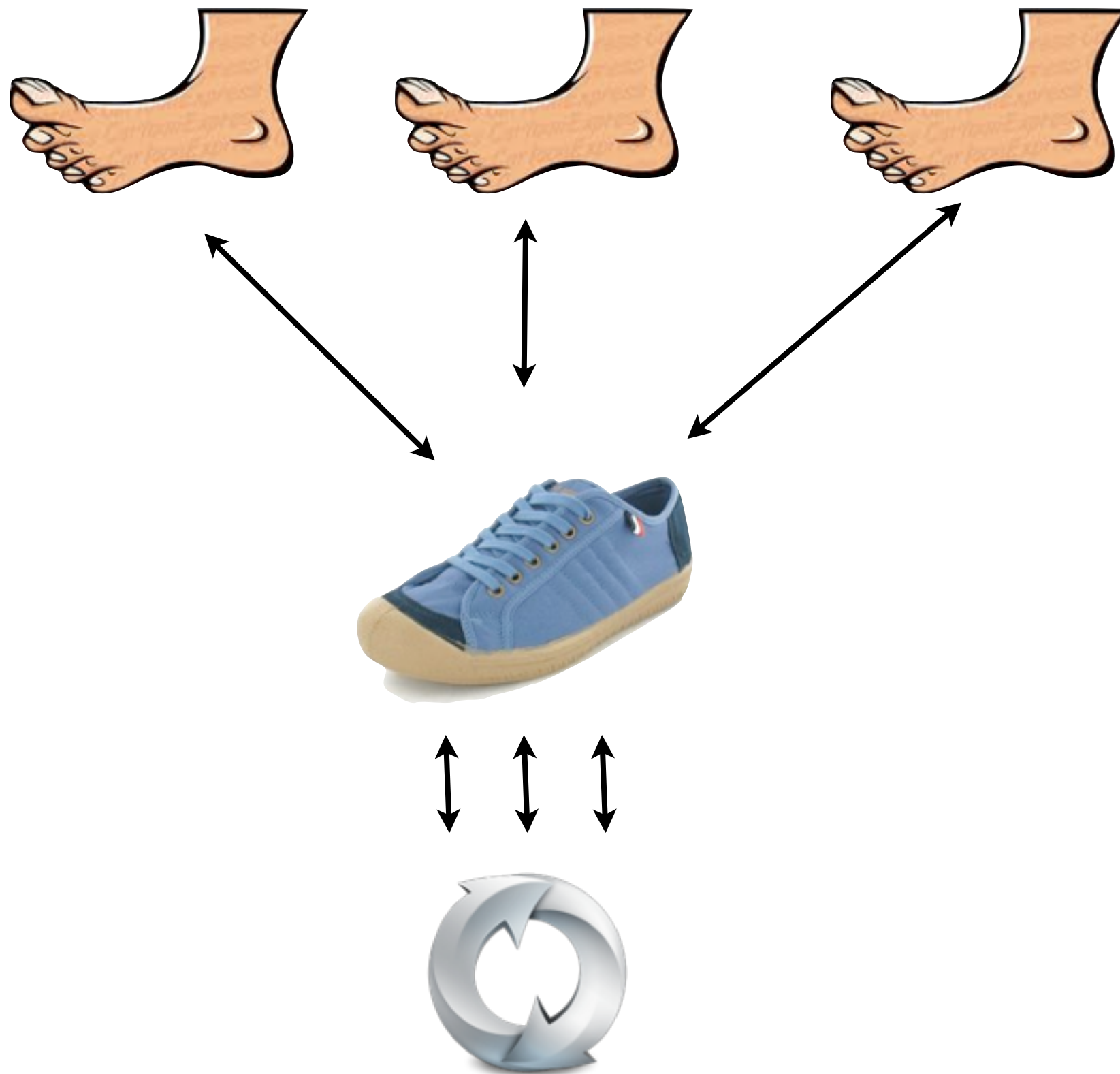


Fork





Fork



# Multiprocessing

- Threads ou Events ?
- Apache, Nginx, Node.js, .... (Erlang)



- MPM Prefork

```
$ ab -n 1000000 -c 250 http://127.0.0.1/  
$ ps ax|grep apache2|wc -l  
152
```

- MPM Worker (hybride prefork + threads par requêtes)

```
$ ps ax|grep apache2|wc -l  
5
```



# Apache

- 1 master process, plusieurs workers
- Prefork
- Accepte plusieurs connections par worker
- Évènements

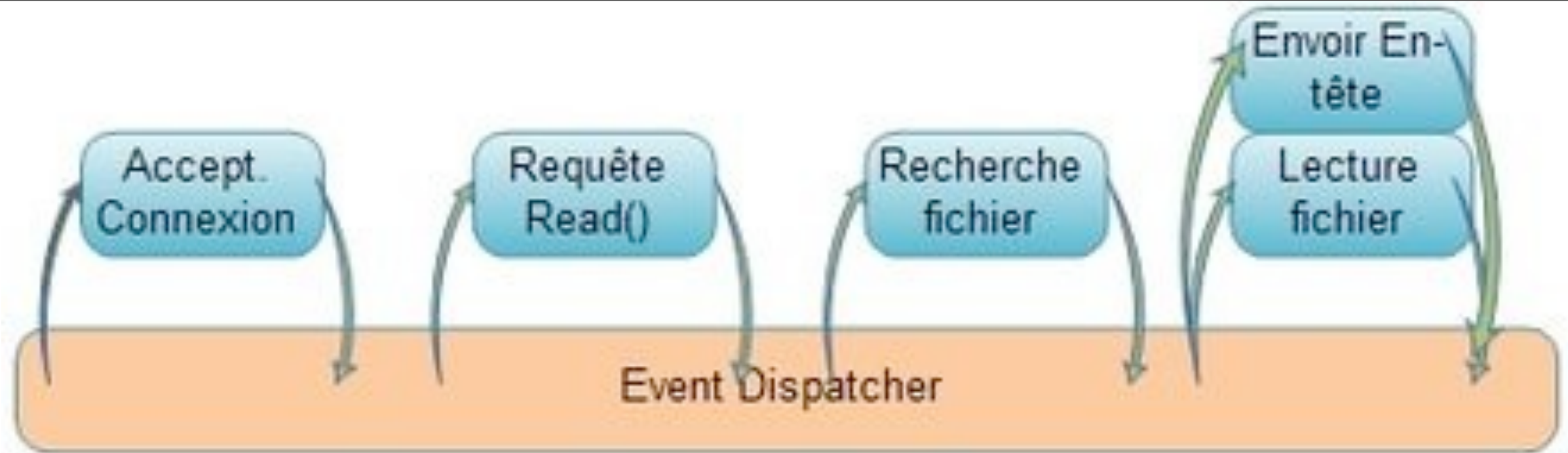


- 1 seul thread
- 1 boucle d'évènements (READ/WRITE)
- Context switching



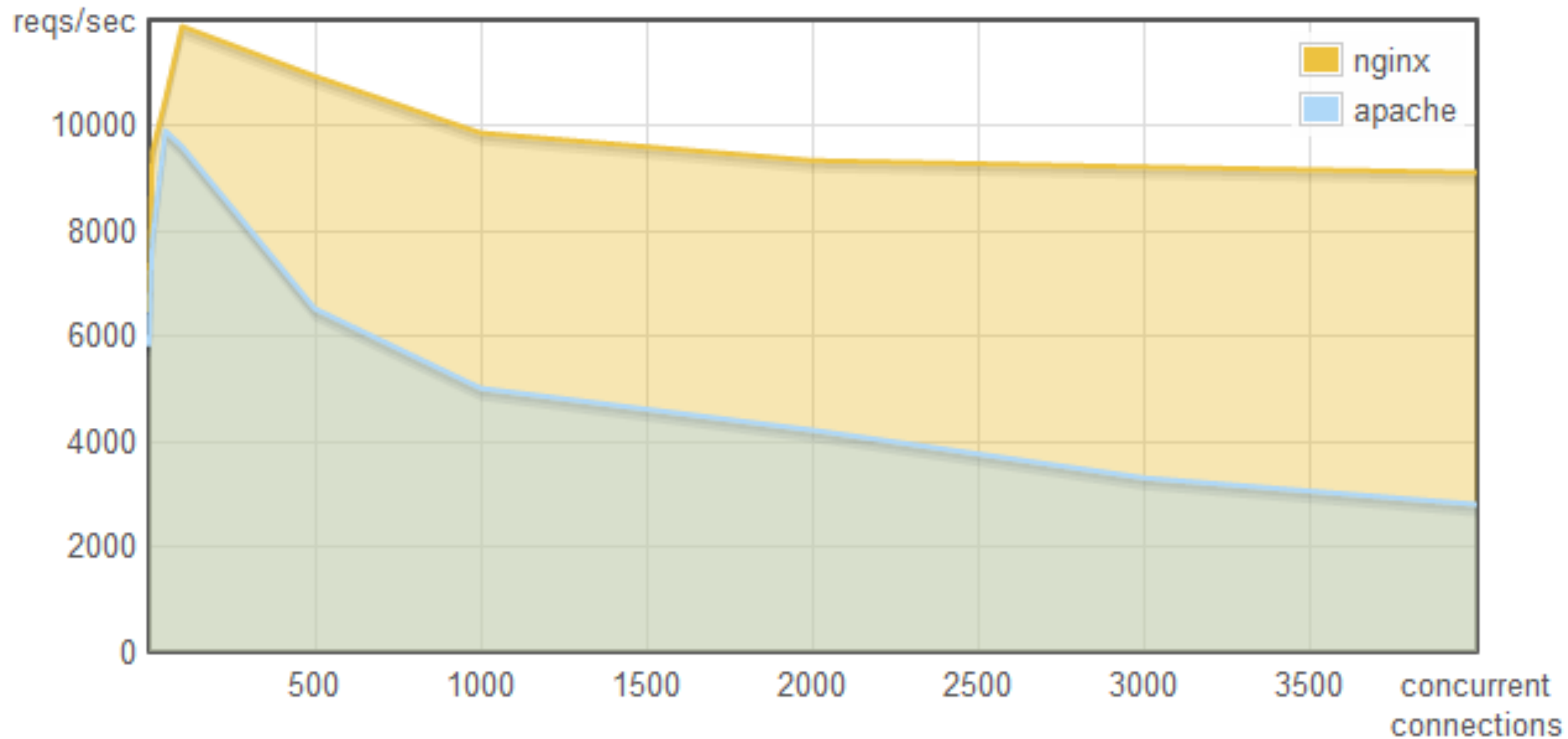
Event Loop



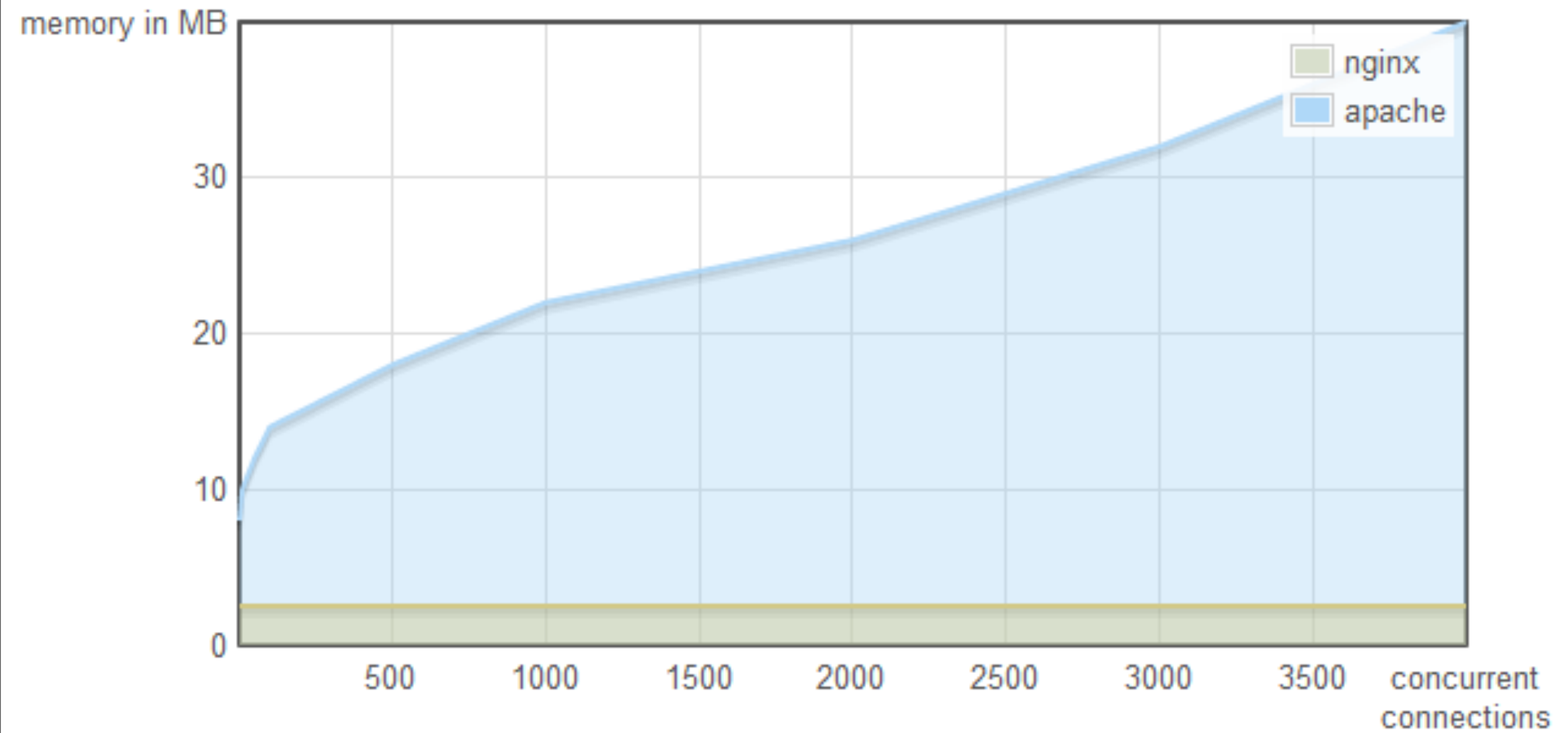


- Nginx VS Apache
- <http://blog.webfaction.com/a-little-holiday-present>

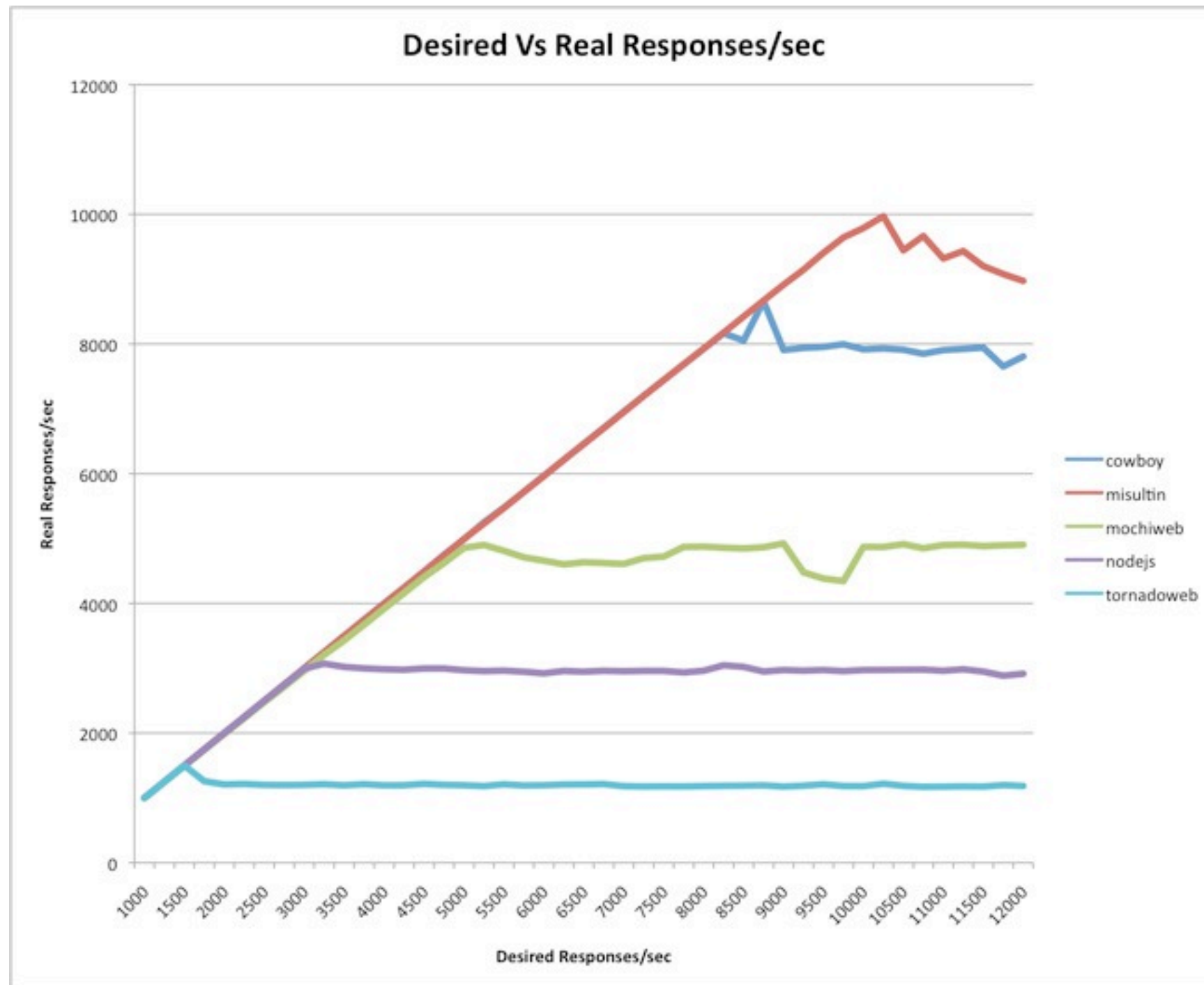
# Nginx versus Apache (with the worker-MPM) for serving a small static file:



# Usage de la mémoire







<http://www.ostinelli.net/a-comparison-between-misultin-mochiweb-cowboy-nodejs-and-tornadoweb/>

Choix

- WSGI (PEP 3333)
- Prefork - socket partagé
- async, sync workers (gevent, eventlet, tornado, ...)
- S'intègre facilement avec django, paster, ...



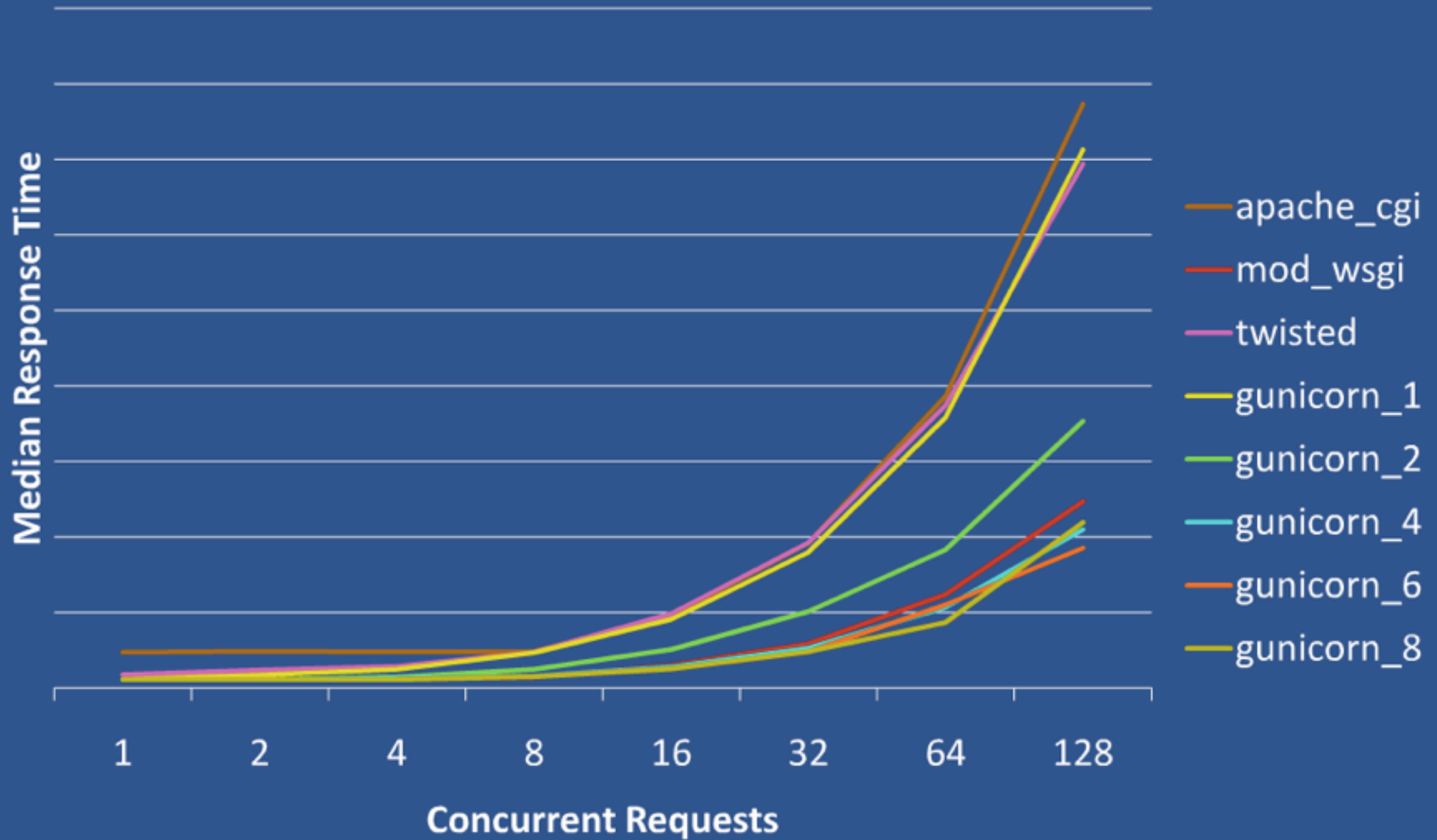
- HTTP Stream
- Upgrade à chaud “à la nginx”
- Ligne de commande simple:

```
$ gunicorn -w 3 test:app
```

- greins (<https://github.com/meebo/greins>)



# Performance





**GUNICORN!!!!!!**



asynchrone ou  
synchrone

# twisted

- librairie asynchrone
- Évènements & callbacks
- serveur twisted: 1 seule boucle d'évènements

# gevent & eventlet

- Bibliothèques réseaux
- api synchrone
- monkey patch
- coroutines
- Le code tourne toujours de manière asynchrone



```
from gevent import pywsgi

def handle(environ, start_response):
    start_response('200 OK', [('Content-Type', 'text/html')])
    yield '<html><body><h1>Yo!</h1></body></html>'

server = pywsgi.WSGIServer(('127.0.0.1', 1234), handle)
print "Serving on http://127.0.0.1:1234..."
server.serve_forever()
```





EVE<sup>®</sup>  
ONLINE

---

TYRANNIS



- Python stackless
- ou `py.py --withmod-_stackless`

<http://www.grant-olson.net/python/intro-to-stackless-python>

- ulimit est ton ami
- somaxcon
- ...

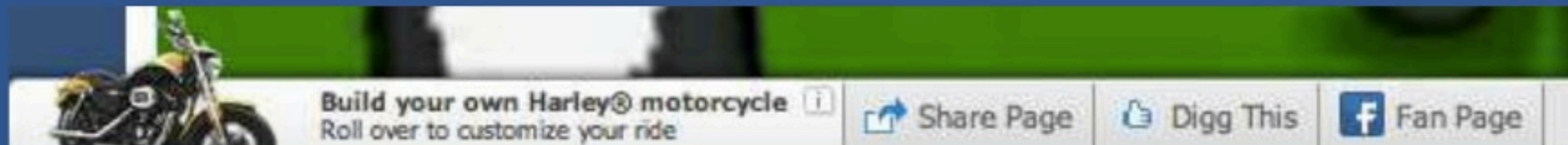
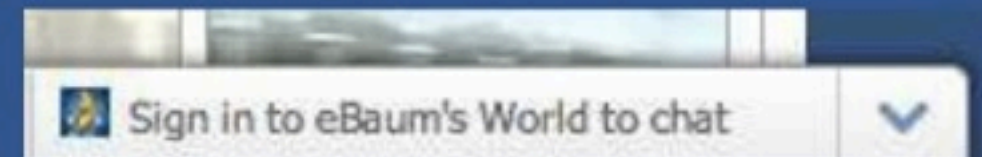


# Tuning



# Meebo Bar

- 1000+ sites
- Quantcast: 197 MM monthly uniques\*
- LOTS of pageviews
- LOTS of ad requests



\* <http://bit.ly/xAPXx>



# **Firefox Sync**

- 10 webheads
- 83 mysql
- 2 openldap masters
- 4 openldap slaves
- ~1.5 millions registered users
- ~550 RPS pour le storage





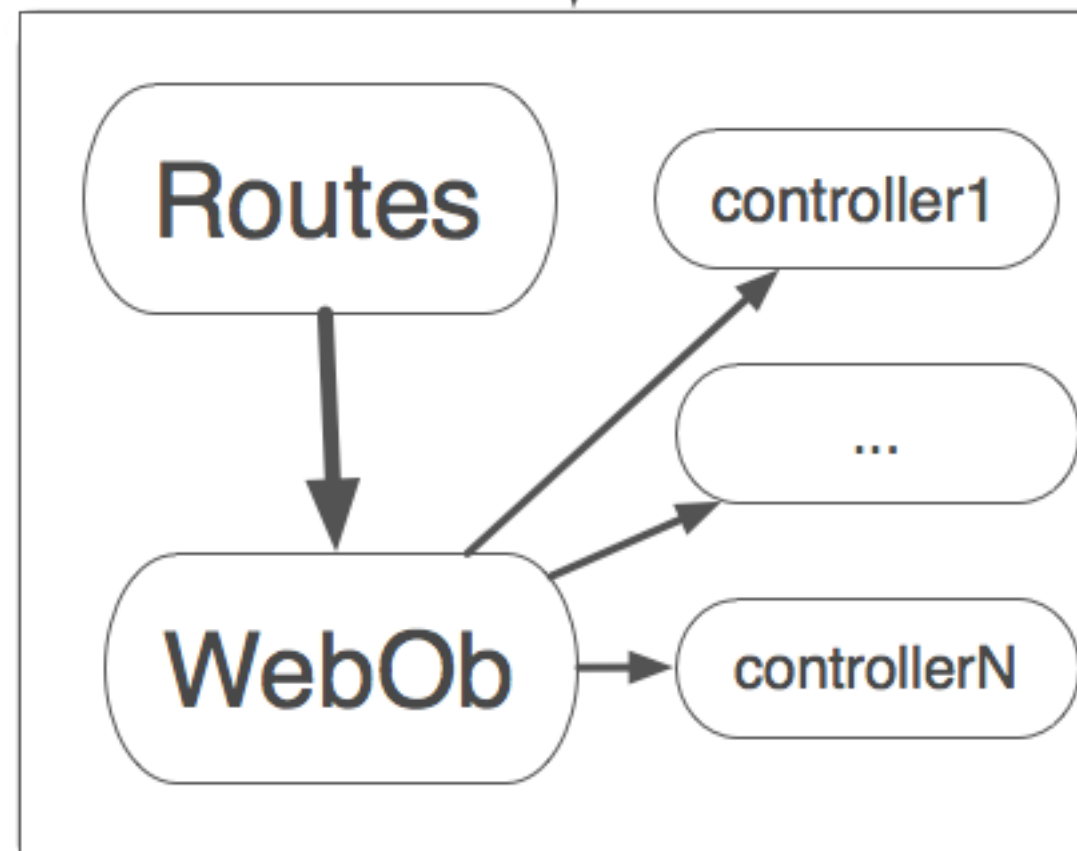
TCP



gunicorn

wsgi

ServiceApp



# Déploiement

- libcloud (<http://libcloud.apache.org/>):  
deployer sur les “clouds”
- juju (<https://juju.ubuntu.com/>) : ubuntu  
cloud (openstack + ensemble ou ec2)

Déployer



- silverlining (<http://cloudsilverlining.org/>)
- kraftwerk (<http://kraftwerk-wsgi.org>)

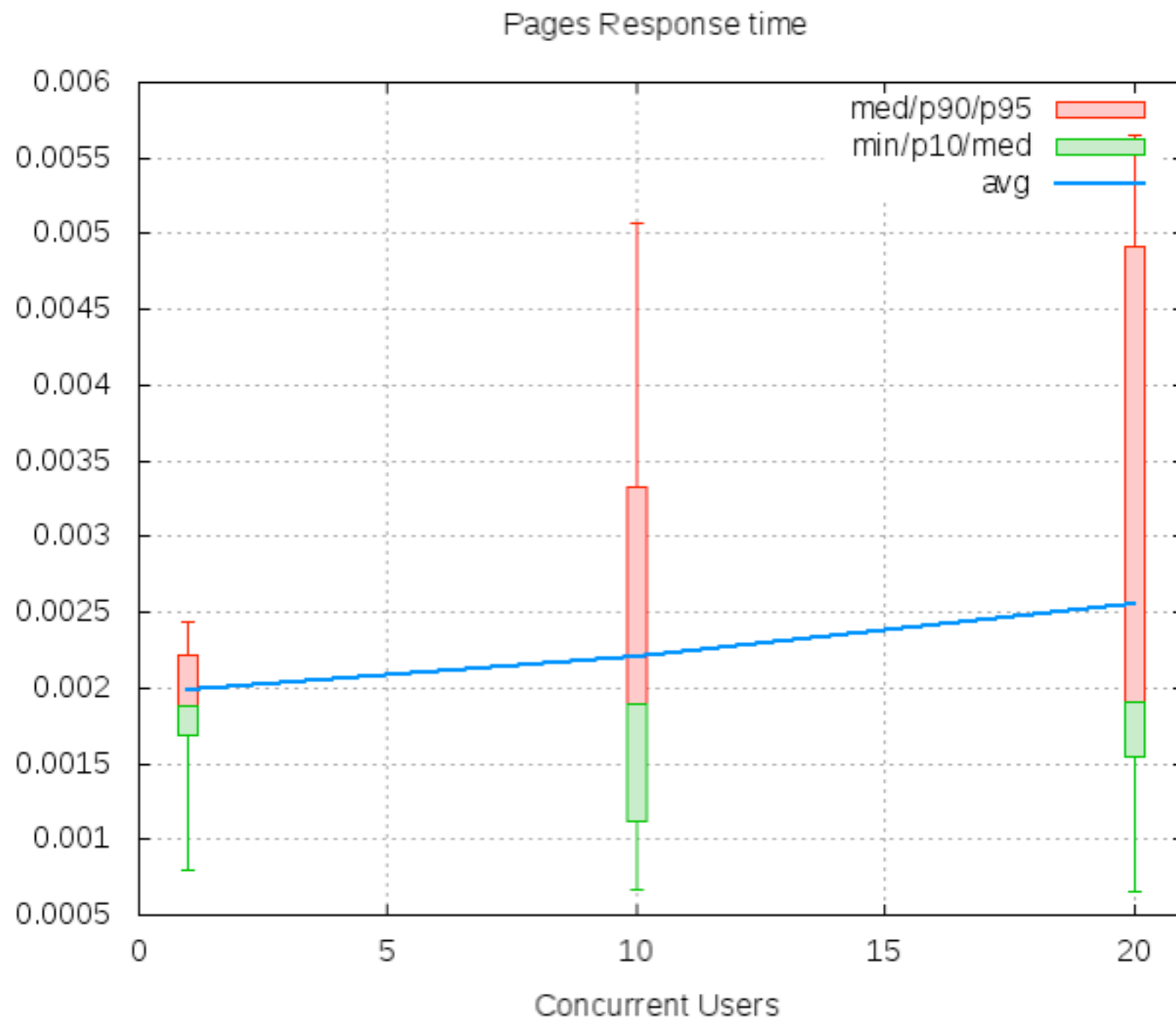
Déployer

- Apache Bench (ab)
- httpperf
- Apache Jmeter
- Grinder
- Funkload

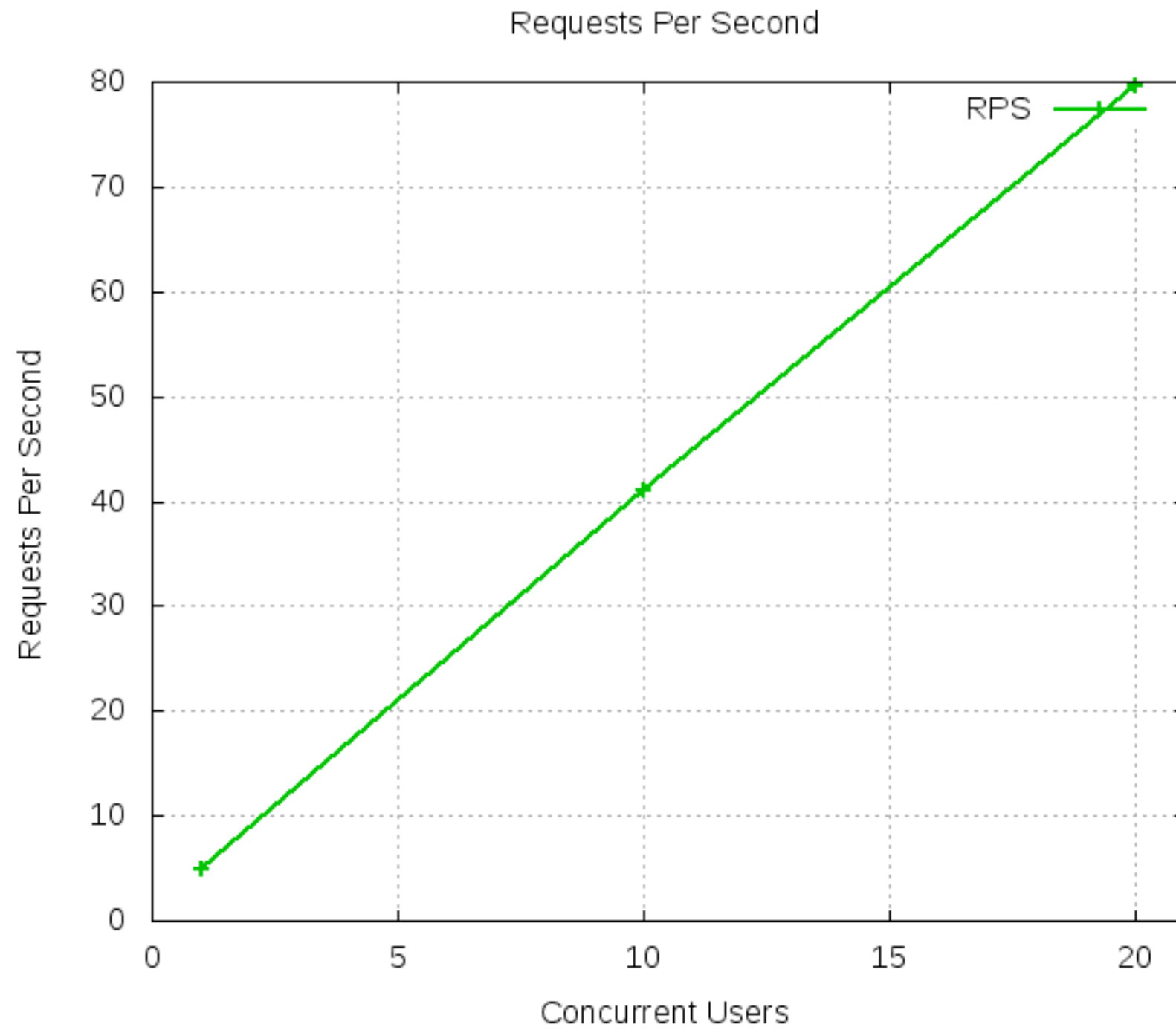


# The Grinder

Bench



Funkload



Funkload

# Merci!

- @benoitc
- @tarek\_ziade

Moar ???