NODESTACK
ONLINE CONFERENCE

The Crazy-Cool Things you can do with Node.js

Nuno Job, Nodejitsu

@dscape

NODESTACK
ONLINE CONFERENCE

nodejitsu

NODE
FIRM

NODESTACK
ONLINE CONFERENCE

node JS ™

NODESTACK
ONLINE CONFERENCE

I HAVE NO SPECIAL TALENTS. I AM ONLY PASSIONATELY CURIOUS.

-ALBERT EINSTEIN

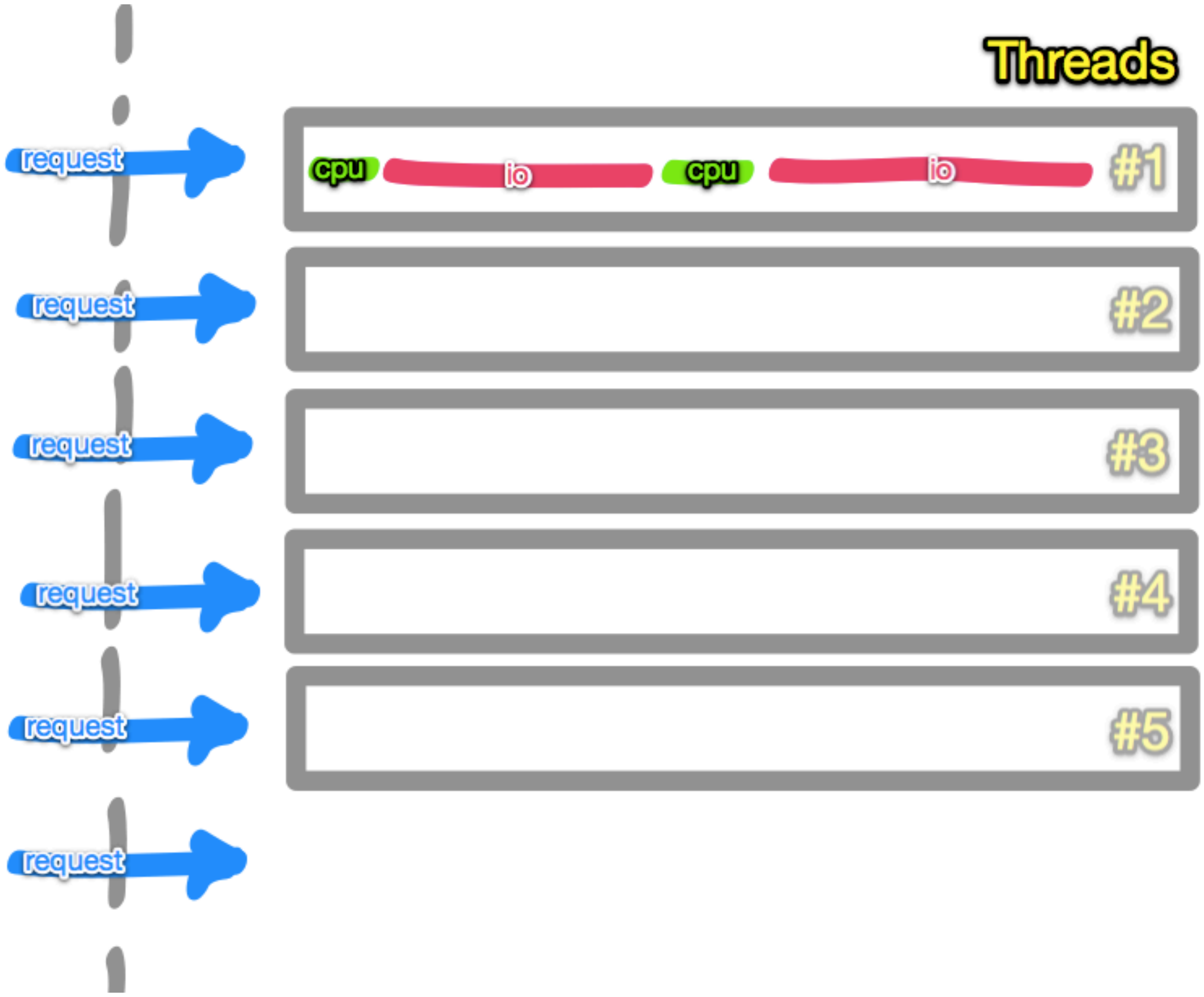
2+2

17x24

Fast calculations
Slow IO

Should we scale them the same way?

Apache


Threads
request
request
request
request
request
request
cpu
io
cpu
io
#1
#2
#3
#4
#5

flickr.com/photos/s4xton

Event loop

flickr.com/photos/liberato

Sample Program

- parse auth params,
  asynchronously **authenticate**,
  when you get a response execute this callback function
- auth callback executed,
  asynchronously **query** the db,
  when you get a response execute this callback function
- db callback executed,
  prepare html to **render** with info from db,
  send to user

Synchronous vs. Asynchronous

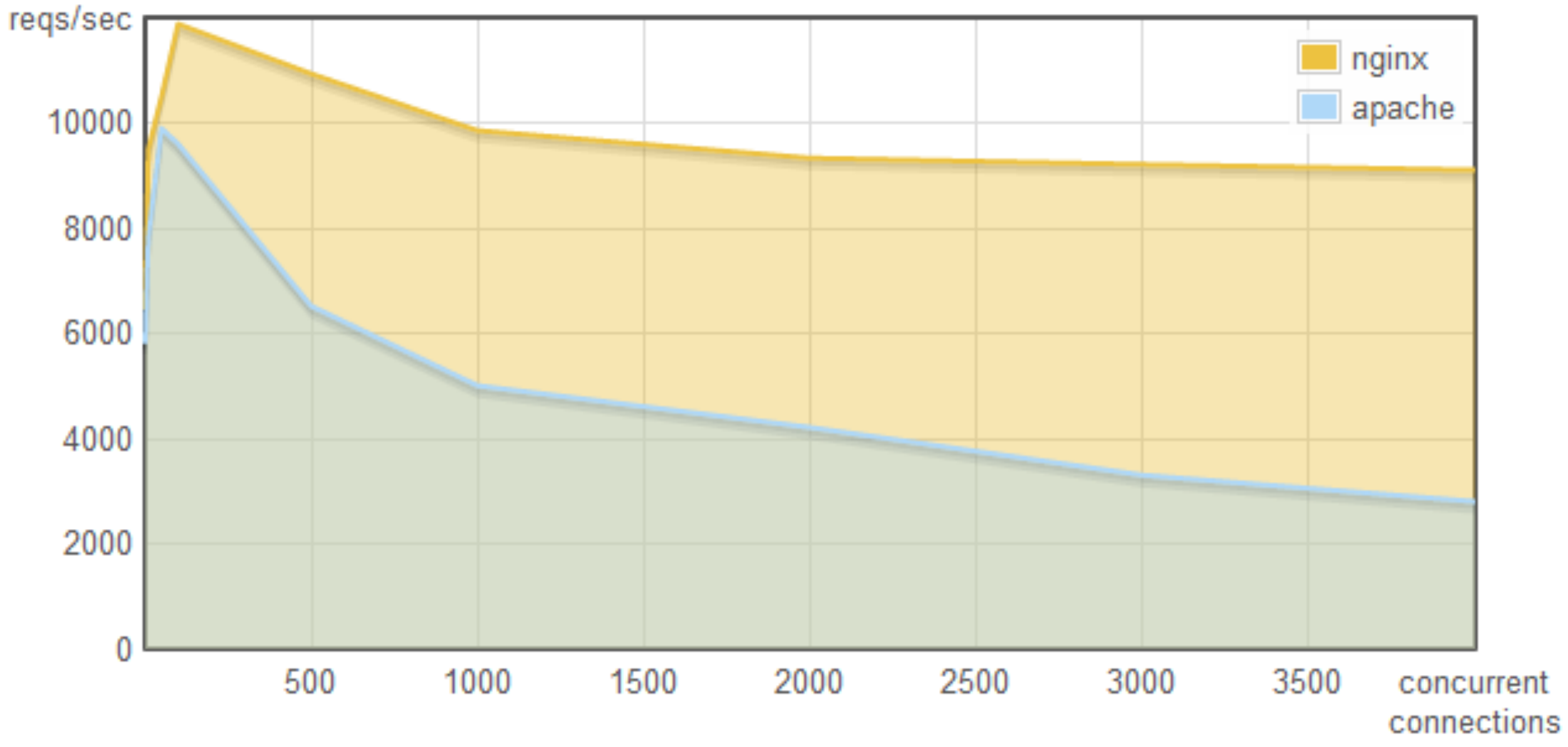
```
try {
 var auth = auth.authenticate(creds)
 // wait for io
 var user = sql.execute(
  “select * from users where id=”+
  auth.id)
 // wait for io
 response.send(render.user(user))
} catch (e) {
  response.send(“failed”)
}
```

```
auth.authenticate(creds,
function auth_cb(error, auth) {
 if(error) {
  return response.send(“auth”)
 }
 sql.execute(
 “select * from users where id=”+
 auth.id,
 function (error, user) {
  if(error) {
   response.send(“query”)
  }
  response.send(null, user)
 })
})
```

NODESTACK
ONLINE CONFERENCE

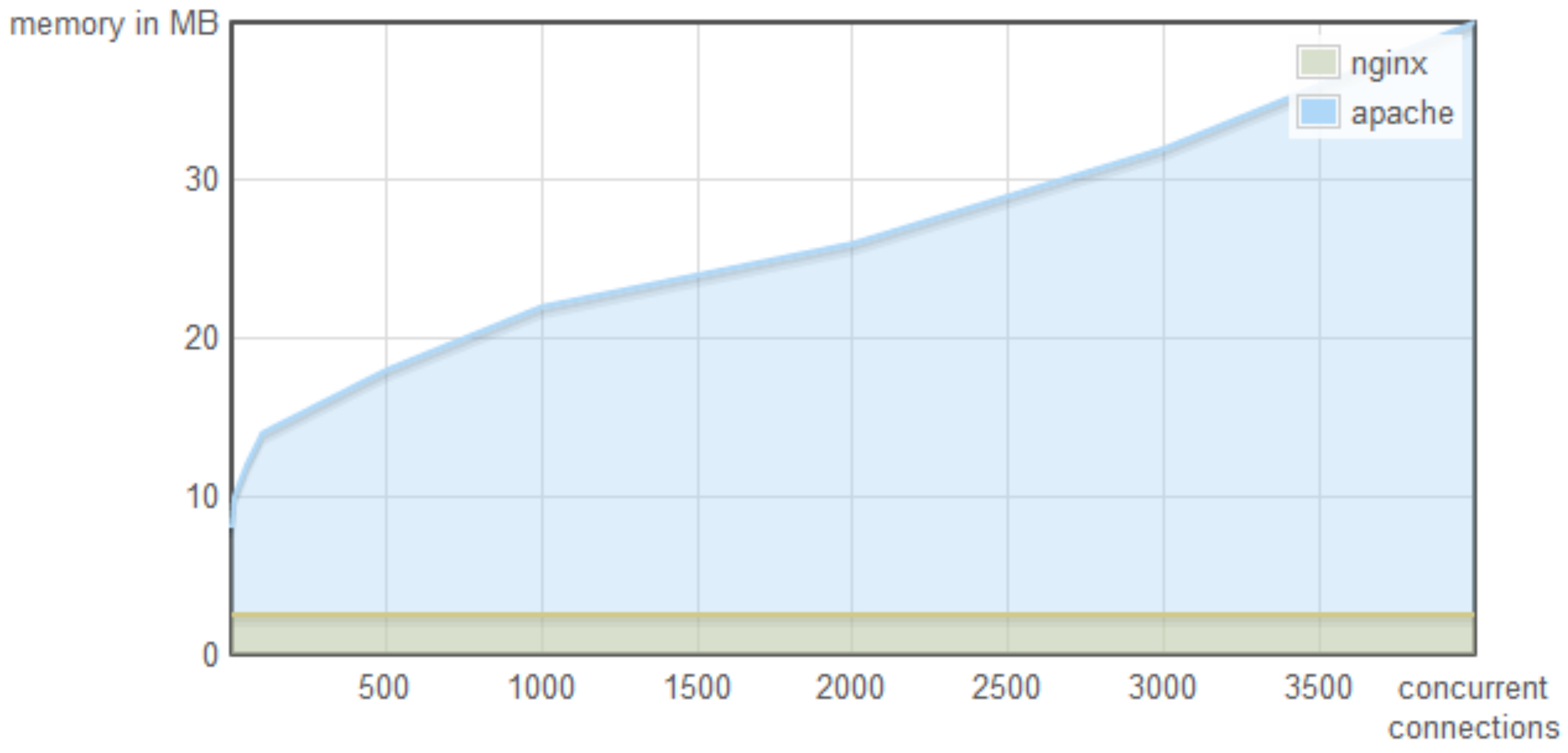
Does it matter?

Concurrency



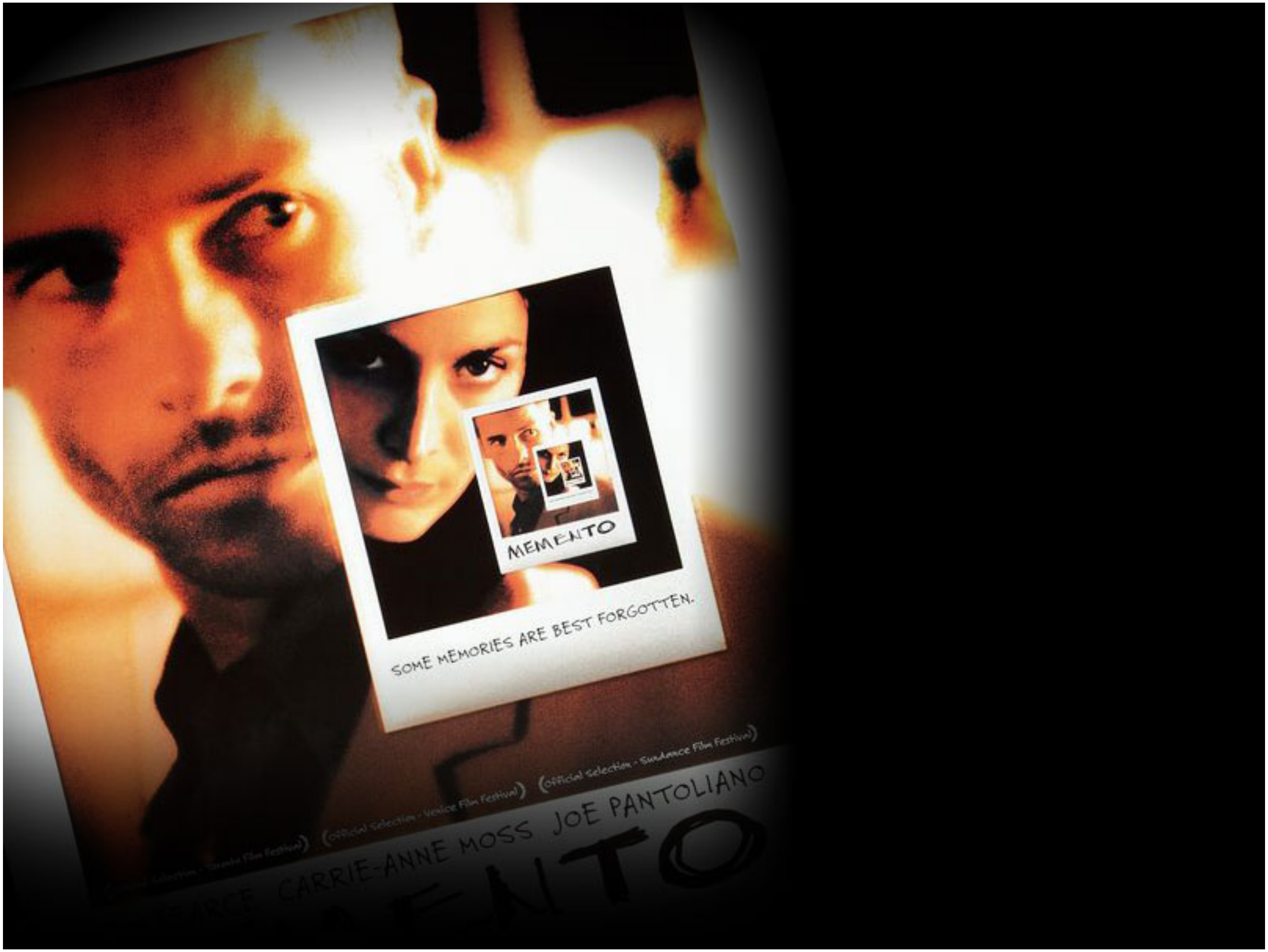
http://blog.webfaction.com/a-little-holiday-present

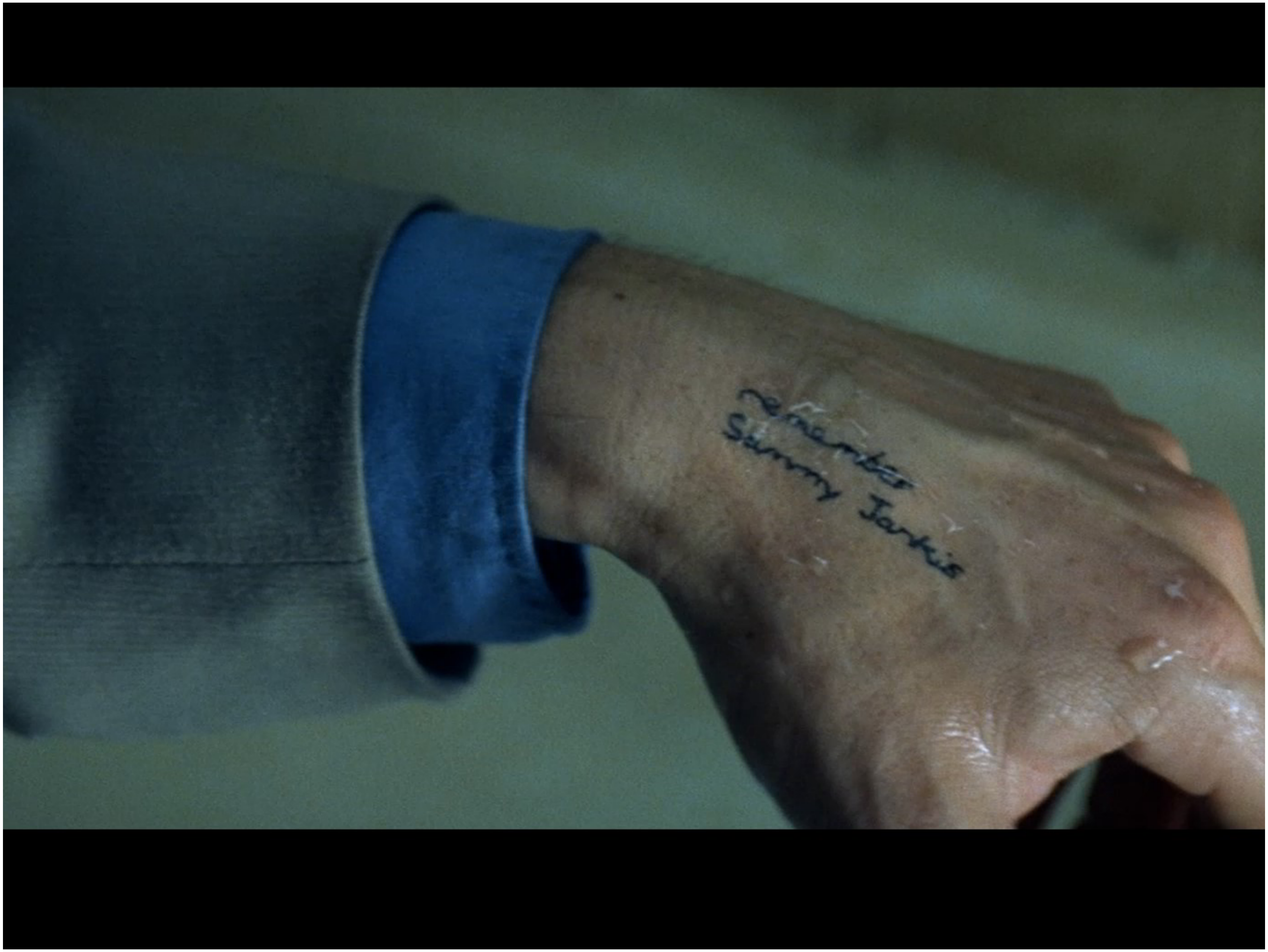
Memory Usage

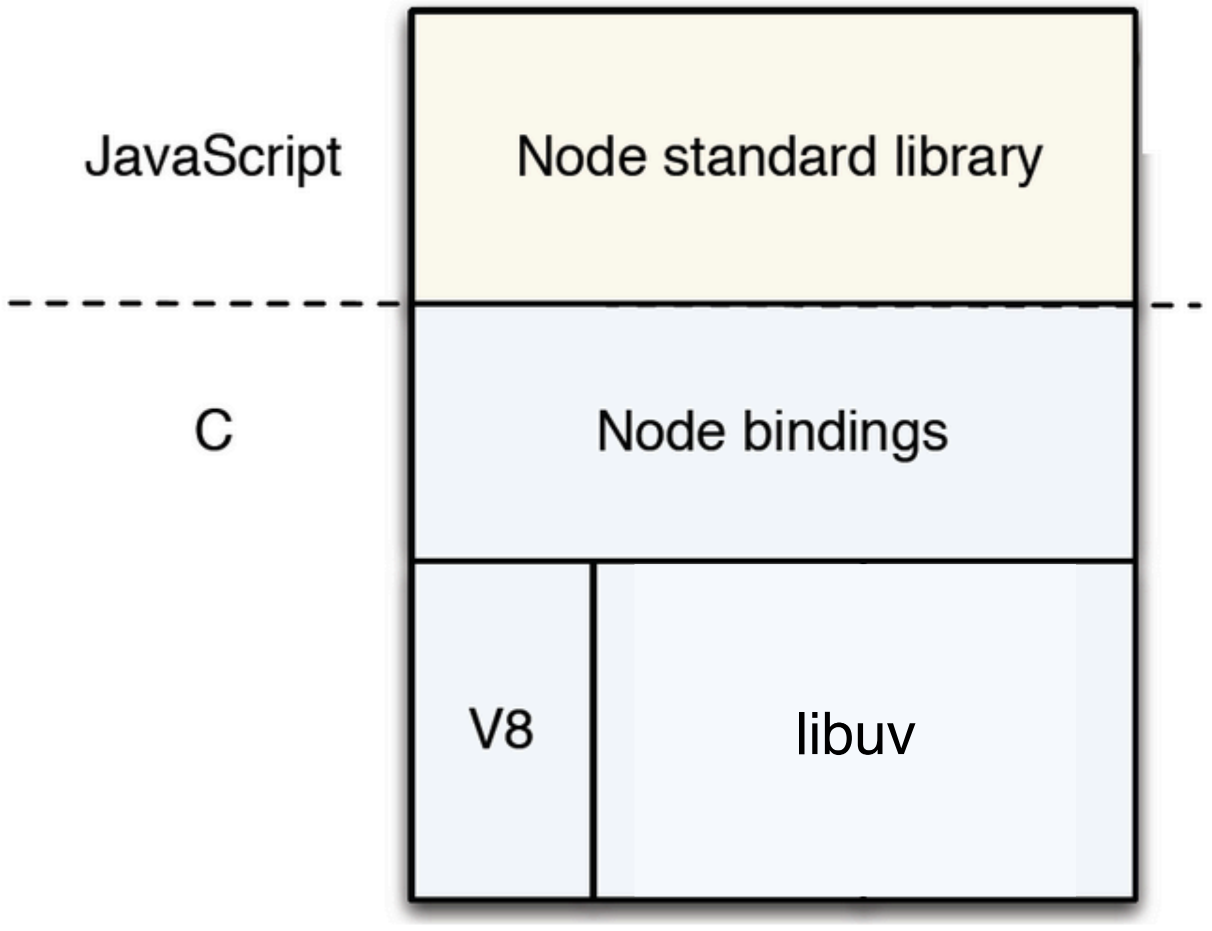


http://blog.webfaction.com/a-little-holiday-present

Yes


MEMENTO
SOME MEMORIES ARE BEST FORGOTTEN.
(Official Selection - Toronto Film Festival)
(Official Selection - Venice Film Festival)
(Official Selection - Sundance Film Festival)
CARRIE-ANNE MOSS
JOE PANTOLIANO


remember
Sammy Jankis


JavaScript
Node standard library
C
Node bindings
V8
libuv

Node Core

NODESTACK
ONLINE CONFERENCE

- Programmable interface to network protocols and some helpers

- TCP
- UDP
- HTTP
- DNS

A non-black box approach

```
var http = require('http');

http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
}).listen(1337, '127.0.0.1');

console.log('Server running');
```

Node Standard Idioms

```
var foo =
  fs.createReadStream('fot.txt');

foo.on('data', function (chunk) {
  process.stdout.write(chunk);
});

foo.on('error', function (err) {
  console.log(err.message);
});
```

```
fs.readFile('foo.txt', 'utf8',
function (err, data) {
  if (err) {
    return console.log(err.message);
  }
  console.log(data);
});
```

NODESTACK
ONLINE CONFERENCE

Streams are to time as arrays are to space

@JedSchmidt

A simple reverse proxy

```
var http = require('http');
var request = require('request');

http.createServer(function (req, res) {
  console.log(req.url);
  req.pipe(request('http://nodestack.org' +
    req.url)).pipe(res);
}).listen(1337);
```

Pros and cons

- Architecture for scaling io based applications, e.g. networking
- Plays very well with V8
- “Fire and Forget” forces developers to save meaningful state as stack trace is lost
- Developers must learn a new event driven programming paradigm

NODESTACK
ONLINE CONFERENCE

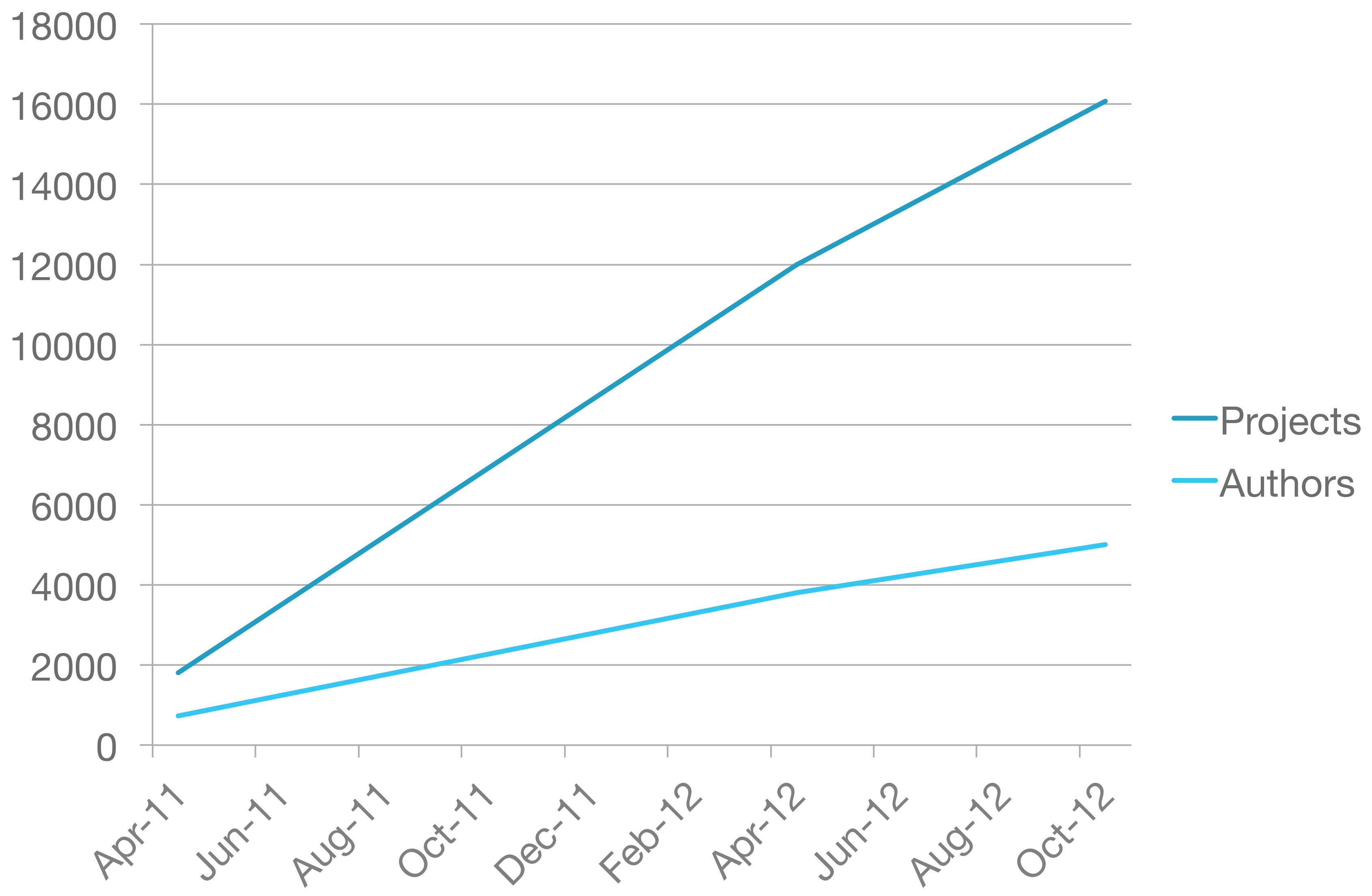
Evolution

NODESTACK
ONLINE CONFERENCE

npm

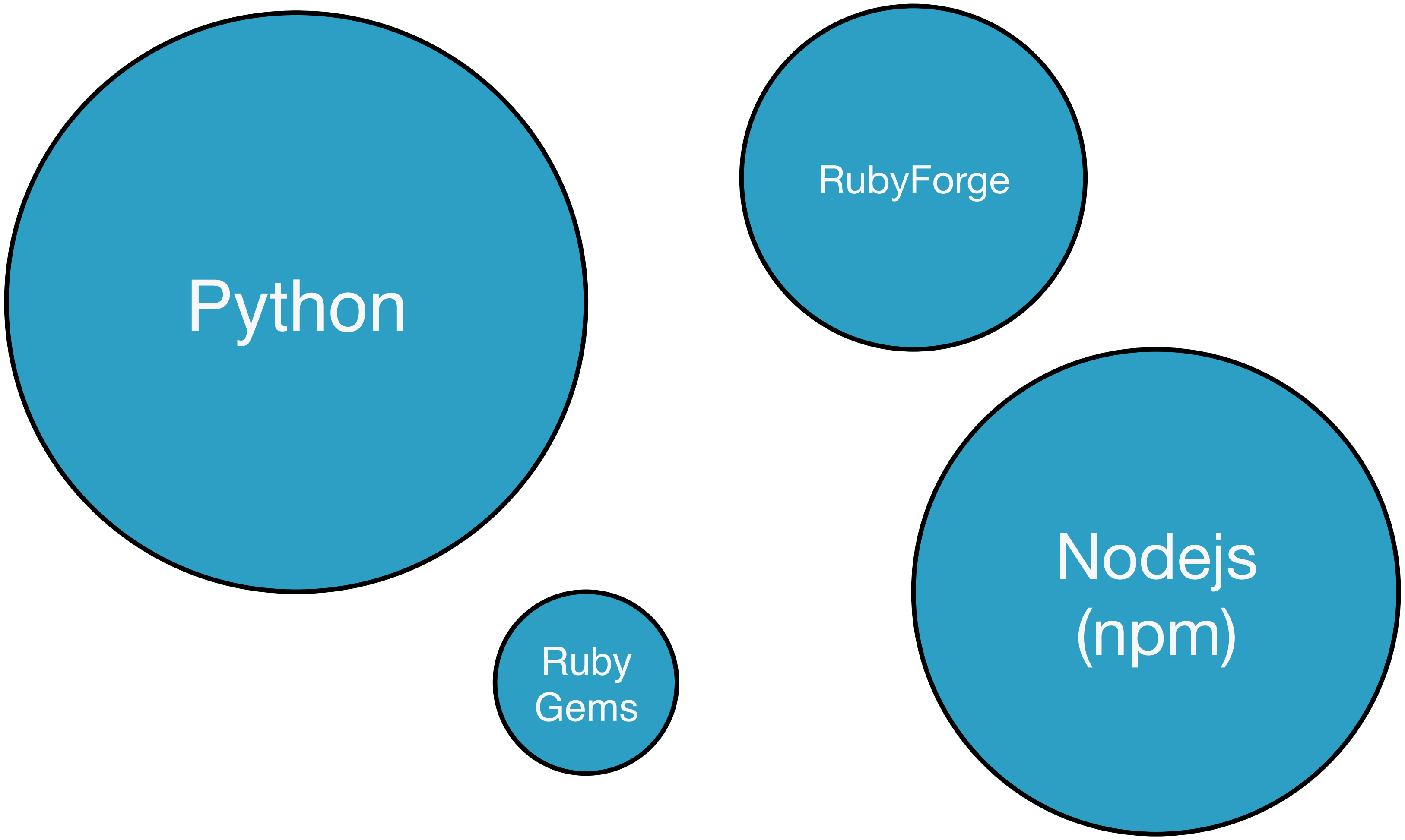
Growth

NODESTACK
ONLINE CONFERENCE



How big is big?

NODESTACK
ONLINE CONFERENCE

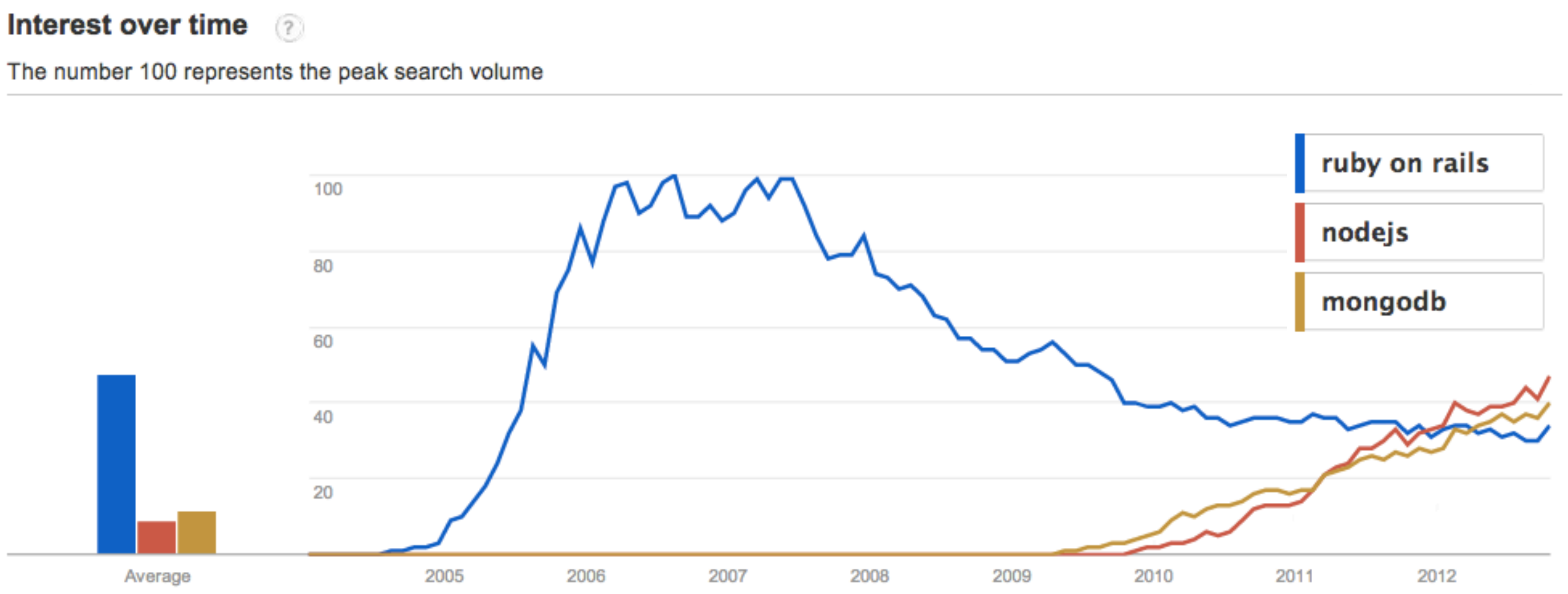


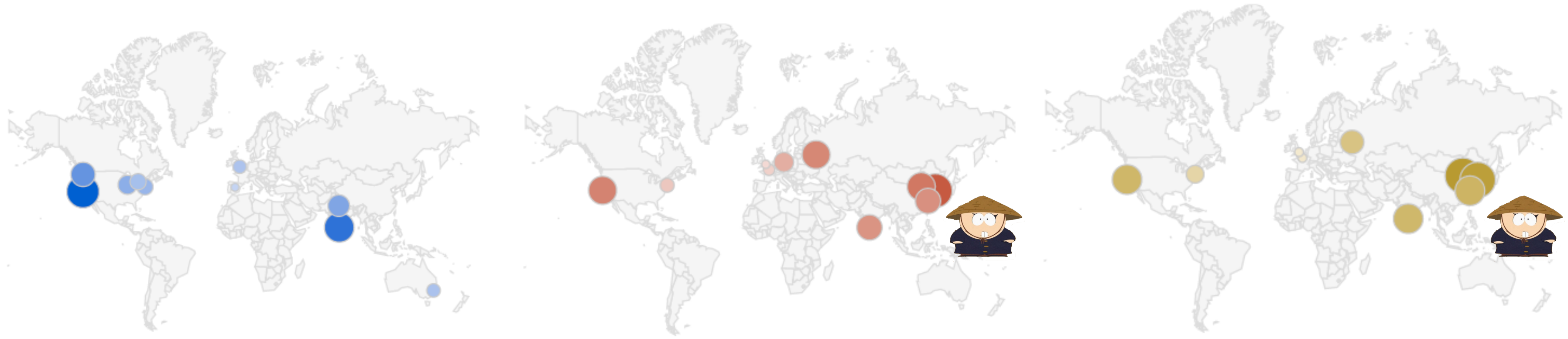
npm secrets

- Super easy to publish
- State of the art package management software
- Adoption of standard idioms makes module creators and users know what interfaces to expect

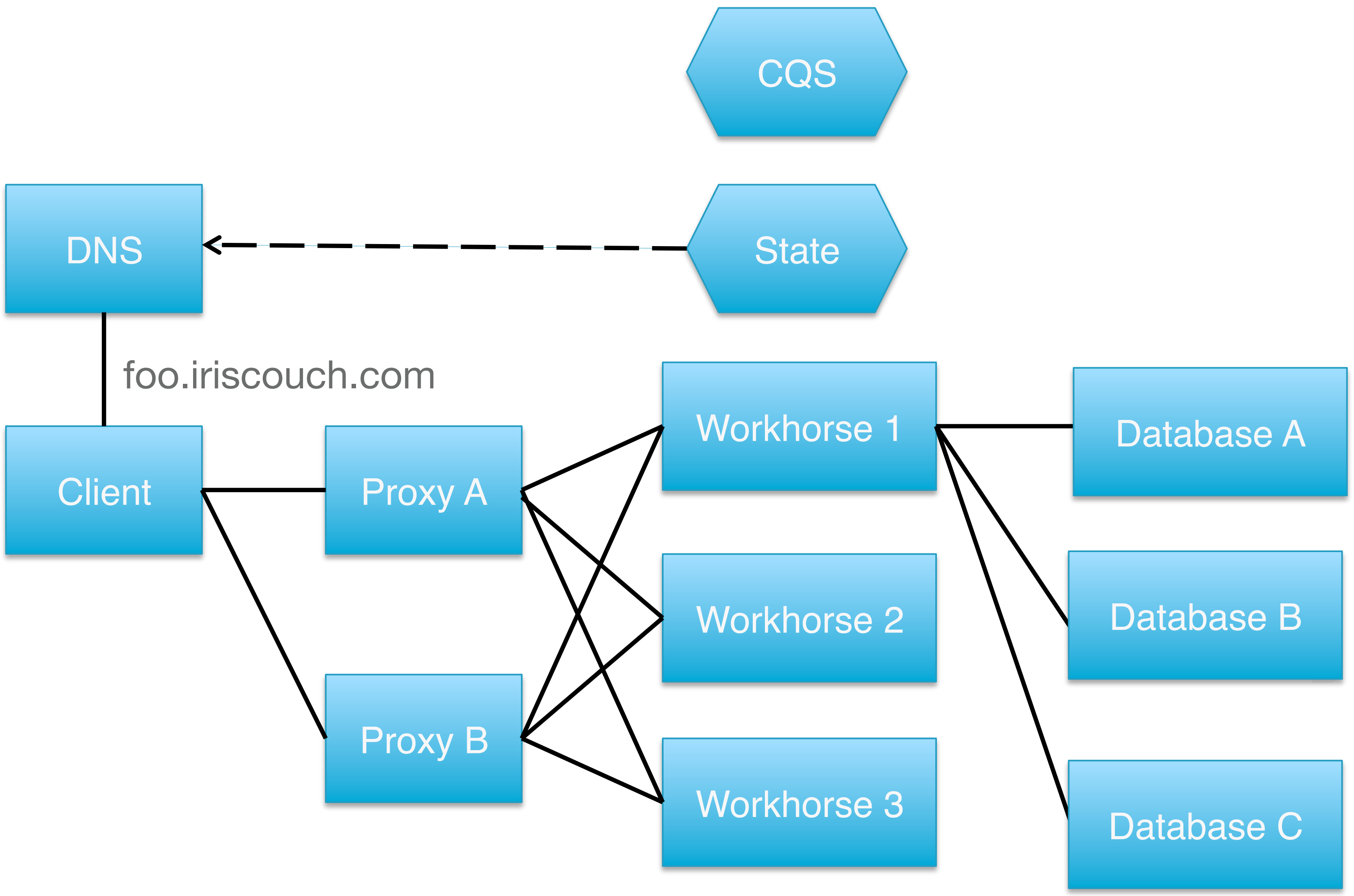
flickr.com/photos/31246066@N04

Growth


Interest over time
The number 100 represents the peak search volume
ruby on rails
nodejs
mongodb
100
80
60
40
20
Average
2005
2006
2007
2008
2009
2010
2011
2012



Iris
Couch

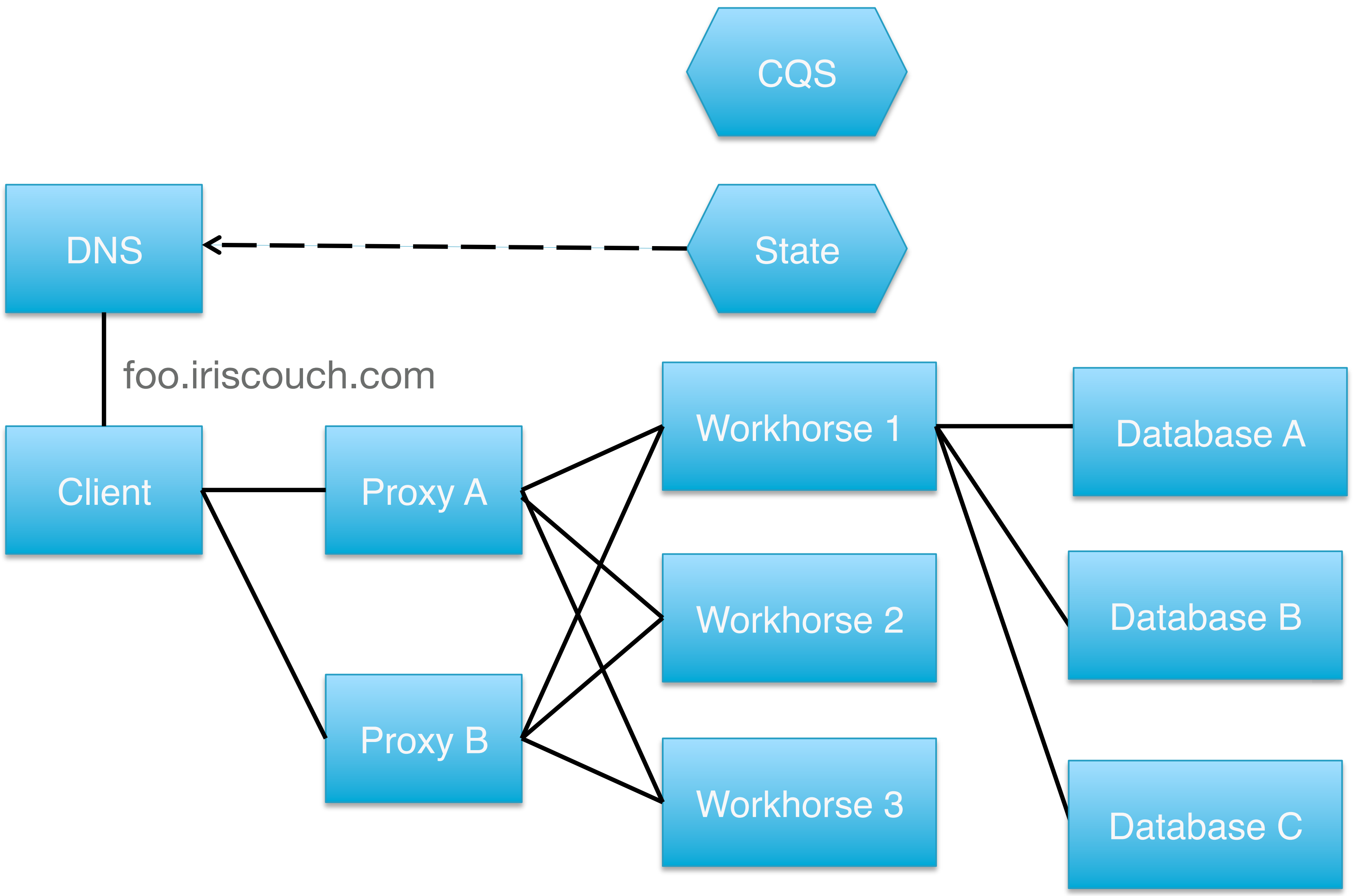

CQS
DNS
State
foo.iriscouch.com
Client
Proxy A
Proxy B
Workhorse 1
Workhorse 2
Workhorse 3
Database A
Database B
Database C

FLIGHT
RECORDER
DO NOT
OPEN

Black boxes

NODESTACK
ONLINE CONFERENCE

```
var named = require('named');
named.createServer(function(req, res) {
  res.end('1.2.3.4');
}).listen(5353, '127.0.0.1');
console.log('Server running')
```



NODESTACK
ONLINE CONFERENCE

Navda
a.header
Parrot
17:00 / 22:09
CC

NODESTACK
ONLINE CONFERENCE

In other words, I am now in control of a flying web server.

@FelixGe

Summary

NODESTACK
ONLINE CONFERENCE

- Extremely efficient networking applications
- Fast javascript runtime (V8)
- Rapid growth in both packages and community
- No black boxes
- Robots

Thank you

NODESTACK
ONLINE CONFERENCE

@dscape