CHICAGO
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE 2015

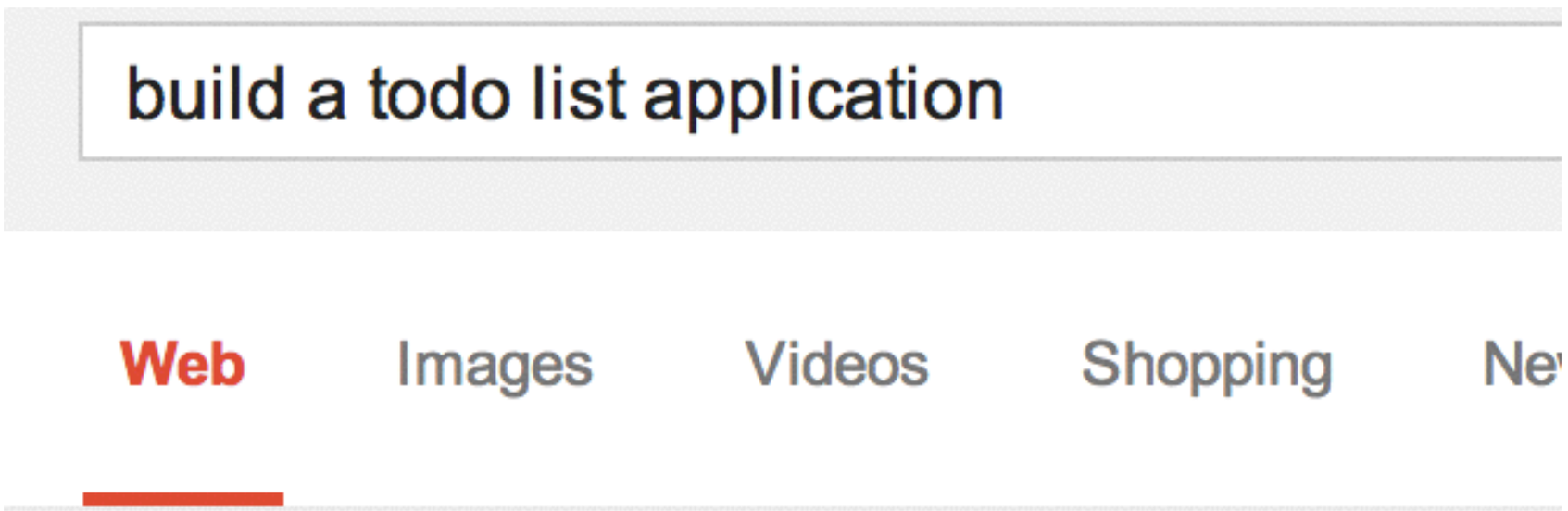
goto; conference

From Homogeneous Monolith to Radically Heterogeneous Microservices Architecture

Chad Fowler


Chad Fowler
Systems Euthanizer
@chadfowler

HELLO WORLD!!!!!!!!!!!!!!!!!


build a todo list application
Web
Images
Videos
Shopping
Ne
About 228,000,000 results (0.97 seconds)

What is Wunderlist?

Wunderlist is available on iPhone, iPad, Mac, Web, Android, Windows and Windows Phone

The wait is over,
it’s time to unwrap Wunderlist 2

December 18, 2012 - 427 Comments

Just in time for Christmas, we’re super excited to deliver you Wunderlist 2. Available on iPhone, Mac, Android, Windows and the Web, Santa’s little helpers, aka our entire team, have worked tirelessly day and night to bring this early Christmas gift to you. So, without much further ado let’s rip off the gift wrap and take a closer look at Wunderlist 2 - your beautiful and simple to-do list. When you first unwrap Wunderlist 2 you’ll notice it’s available for your iPhone, Mac, Android, Windows PC and Web Browser. Select your device of choice and let’s get started. Once you login we have a...

Apartments & Bed and breal | AWS | WunderAPI (Production) – Ne

https://rpm.newrelic.com/accounts/52168/applications/1157400

New Relic.

Volume annual | Account settings | My preferences | 8 | Help | Logout

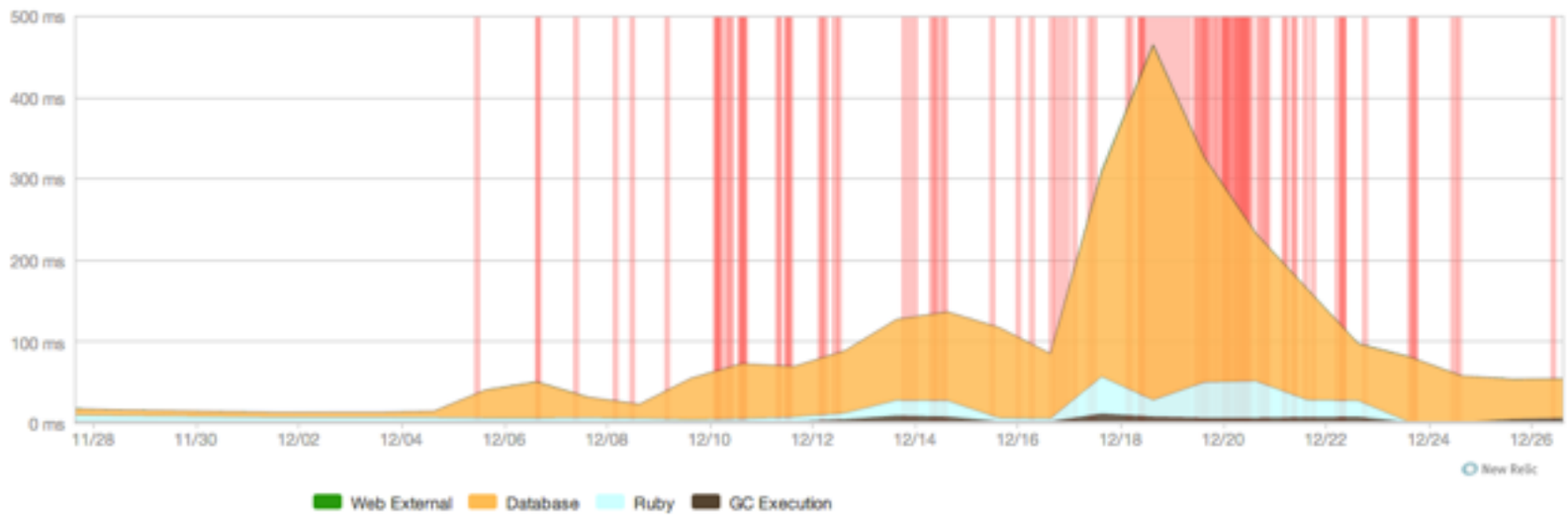
WunderAPI (Production) | Monitoring | Events | Reports | Settings | Servers: All servers (18) | Last 1 month: ending 12/27 03:22

Overview | Map | Web transactions | Database | External services | Background tasks | Geography | Browsers

App server response time (ms)

App server | Browser

155 ms
Average



Apdex score

0.94 [0.2]
App server

Throughput (rpm)

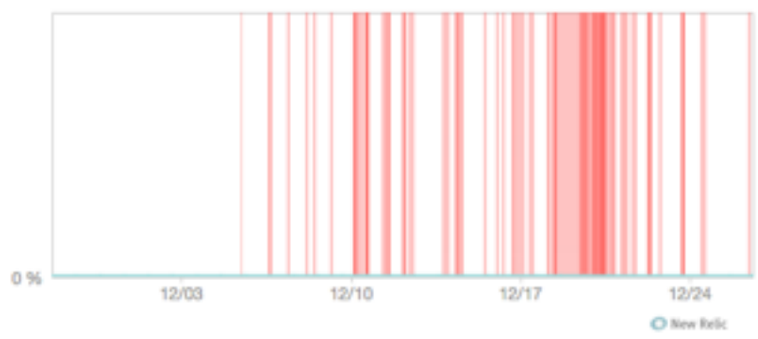
11,800
Average

Compare with yesterday and last week

Loading transactions ...

Error rate →

0.0001%



Recent events

All

YESTERDAY

8:58 Alert on WunderAPI (Production)
Details →

8:41 26 timeout ping failures over 7 minutes
Details →

THIS WEEK

13:13 Mon Alert on WunderAPI (Production)
Details →

12:55 Mon 21 timeout ping failures over 6 minutes
Details →

11:16 Mon Alert on WunderAPI (Production)
Details →

121 timeout ping failures over 31 minutes
Details →

193 servers →	Apdex score	Resp. time	Throughput	CPU usage	Memory
ip-10-208-11-106 1 instance	0.85 0.2	541 ms	8 rpm	1 %	186 MB
ip-10-208-11-233					

WunderAPI (Production) | End user 1.61 s 1.33 ppm | App server 46.3 ms ▼ 25.6k rpm 0.00 err%

Every thing is fine!

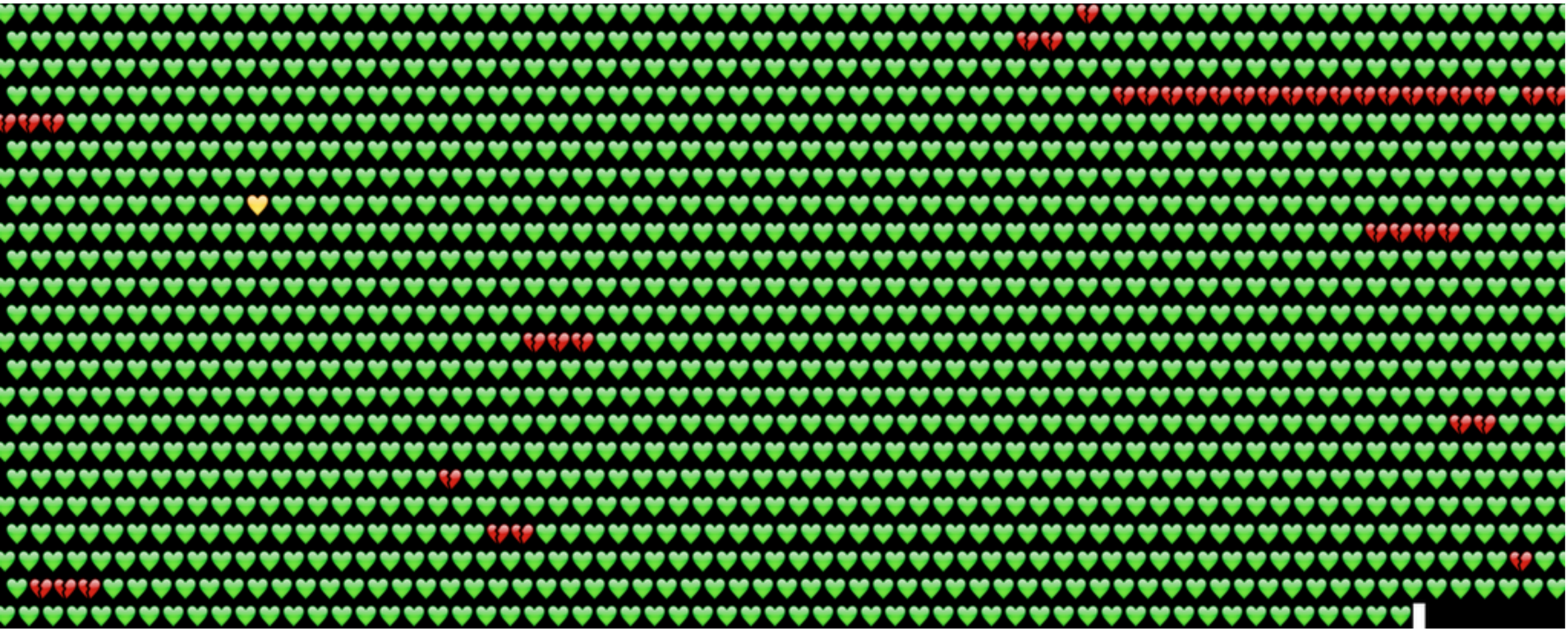
OD

One Monolithic Application

One Monolithic Database

One (*Backend*)
Language (*Ruby*)

Test suite slow and brittle



Caching required for survival

Deployment was terrifying


Logic
buried in
abstractions

Fear

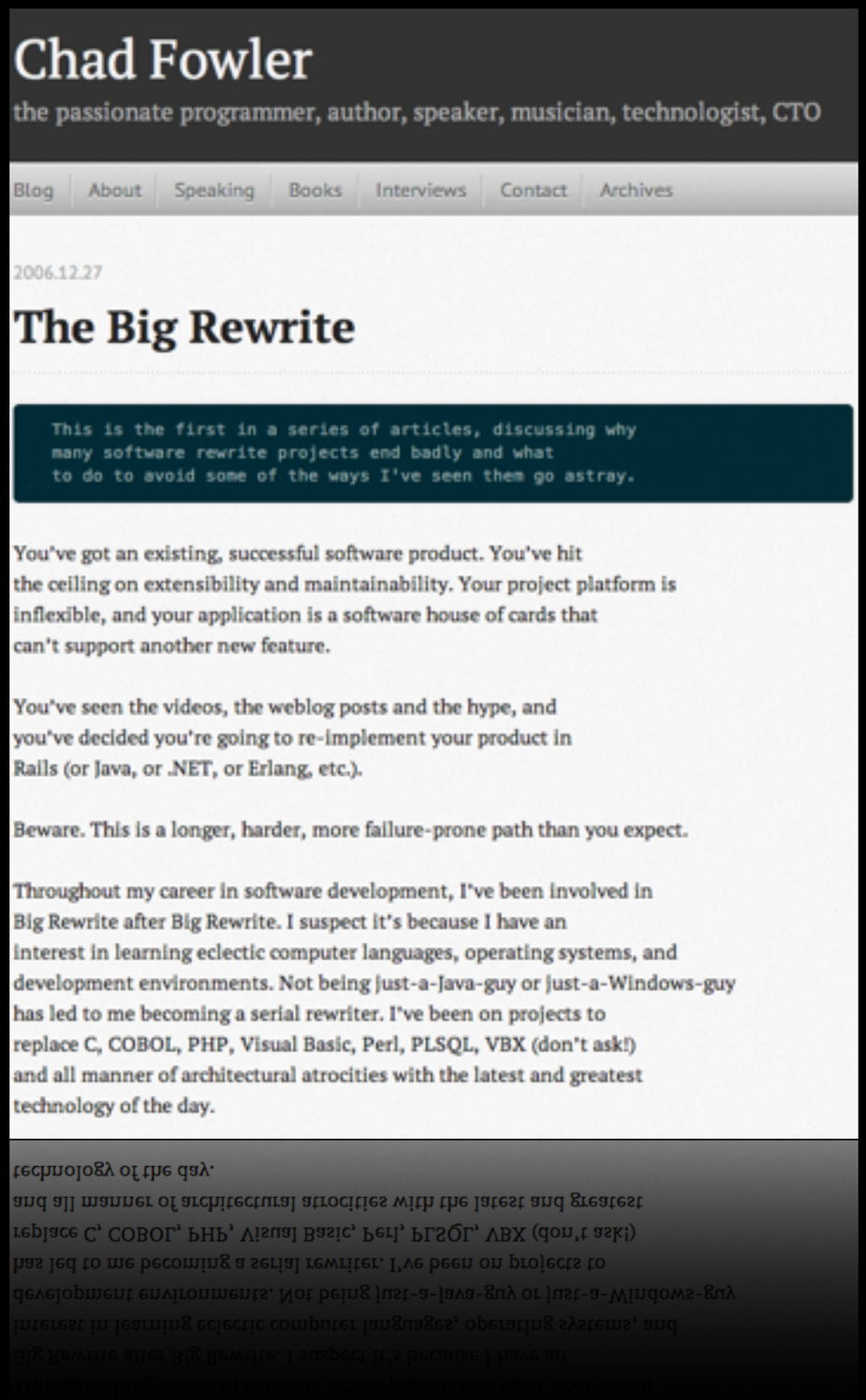
Uncertainty

Doubt

Legacy

@chadfowler

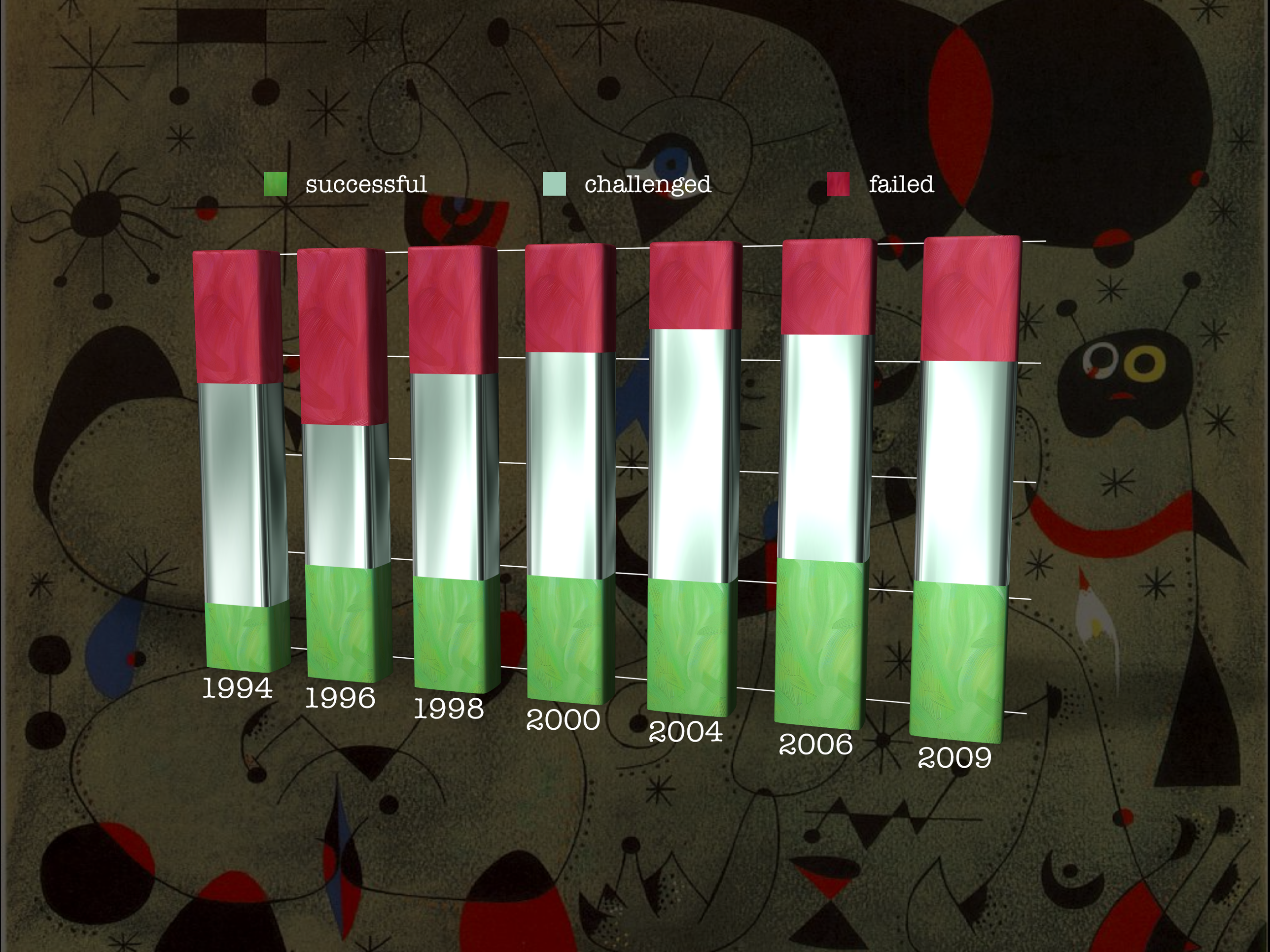
Systems Euthanizer

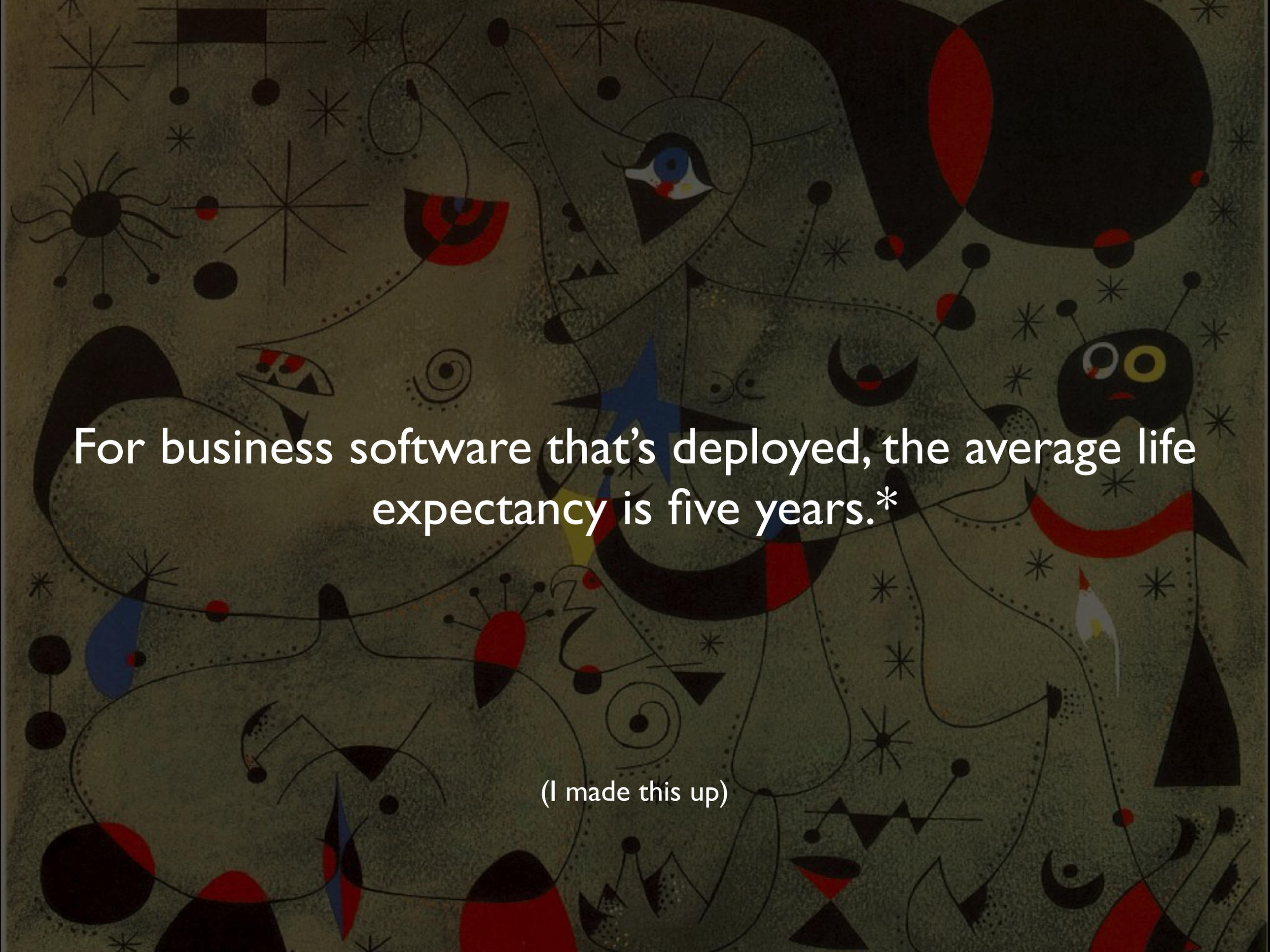


1 *a legacy from a great aunt*: BEQUEST, inheritance, heritage, endowment, gift, patrimony, settlement, birthright; formal benefaction.

"legacy"

2 *a legacy of the wars*: CONSEQUENCE, effect, upshot, spin-off, repercussion, aftermath, by-product, result.


successful
challenged
failed
1994
1996
1998
2000
2004
2006
2009



For business software that's deployed, the average life expectancy is five years.*

(I made this up)

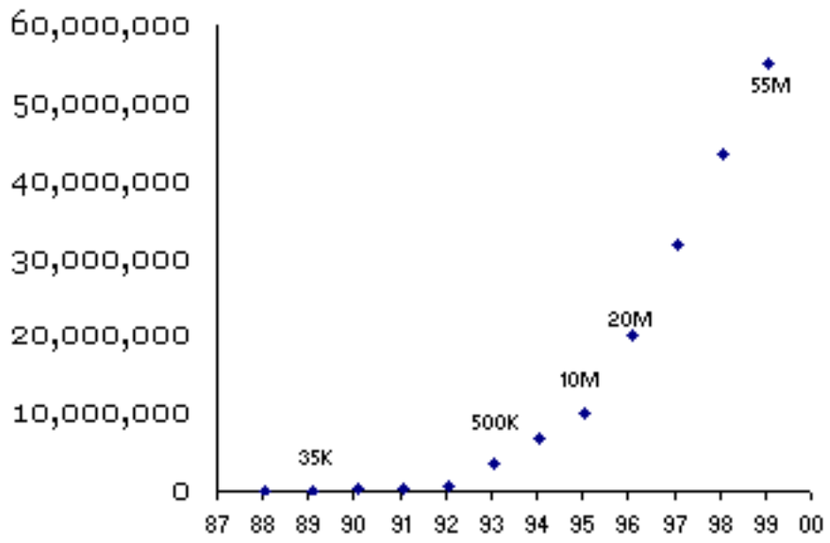
Joel on Software

Good Software Takes Ten Years. Get Used To it.

by Joel Spolsky

Saturday, July 21, 2001

Have a look at this little chart:



How do you CREATE Legacy software?

http://www.dreamsongs.com/Files/DesignBeyondHumanAbilitiesSimp.pdf

richard p. gabriel

Design Beyond Human Abilities

"Biological systems are very much larger than anything (coherent) that people have built"

"When a cell is not healthy, an outside
cell that's part of the immune system can command the cell to destroy itself
without spreading toxins."

homeostasis

Homeostasis

Definition

noun

(Science: Biology)

(1) The tendency of an organism or a cell to regulate its internal conditions, usually by a system of feedback controls, so as to stabilize health and functioning, regardless of the outside changing conditions

(2) The ability of the body or a cell to seek and maintain a condition of equilibrium or stability within its internal environment when dealing with external changes

What is a cell?

What is a system?

When do you build a system vs. a cell?

Are you building the right one now?

If we get these right, can we build systems that outlive us?

MEANWHILE

BACK AT THE RANCH

“Please tell me I’m never going to hear you say you have to do this again.” - *a board member*

How do you CREATE Legacy software?

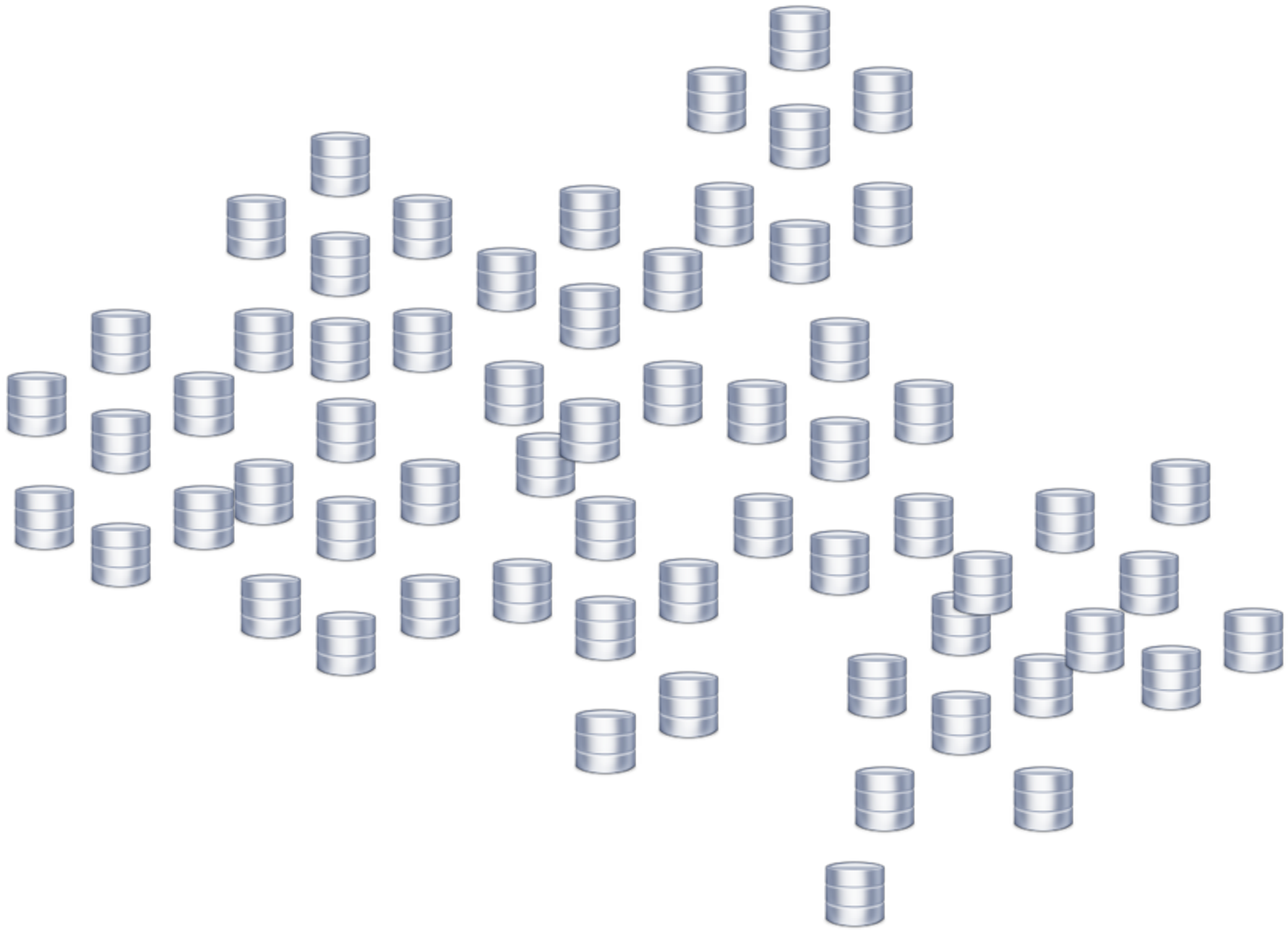
Heterogenous By Default

```
%w.rack tilt date INT TERM..map{|l|trap(l){$r.stop}rescue require l};$u=Date;$z=($u.new.year + 145).abs;puts "== Almost Sinatra/No Version has taken the stage on #$z for development with backup from Webrick
$n=Module.new{extend Rack;a,D,S,q=Rack::Builder.new,Object.method(:define_method),/@@ *([^\n]+)\n(((?!@@)[^\n]*\n)*)/m
  %w[get post put delete].map{|m|D.(m){|u,&b|a.map(u){run->(e){[200,{"Content-Type"=>"text/html"},[a.instance_eval(&b)]]}}}}
  Tilt.mappings.map{|k,v|D.(k){|n,*o|$t||=(h=$u._jisx0301("hash, please");File.read(caller[0][/^[^:]+/]).scan(S){|a,b|h[a]=b};h);v[0].new(*o){n=="#{n}"?n:$t[n.to_s]}.render(a,o[0].try(:[],:locals)||{})}}
  %w[set enable disable configure helpers use register].map{|m|D.(m){|*_,&b|b.try :[]}};END{Rack::Handler.get("webrick").run(a,Port:$z){|s|$r=s}}
  %w[params session].map{|m|D.(m){q.send m}};a.use Rack::Session::Cookie;a.use Rack::Lock;D.(:before){|&b|a.use Rack::Config,&b};before{|e|q=Rack::Request.new e;q.params.dup.map{|k,v|params[k.to_sym]=v}}}
```

Code is “this big”

Disposable Software

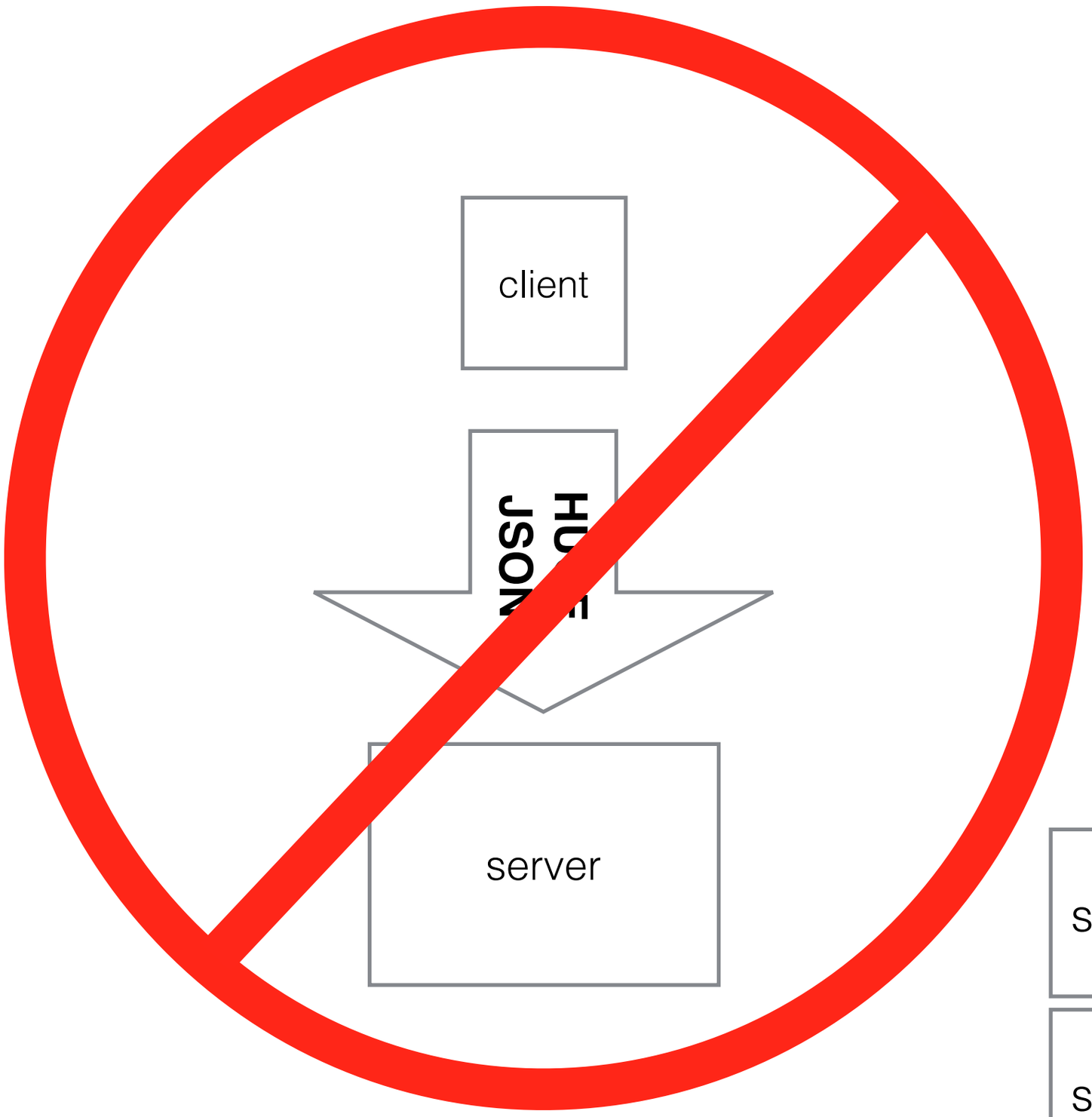
tiny databases

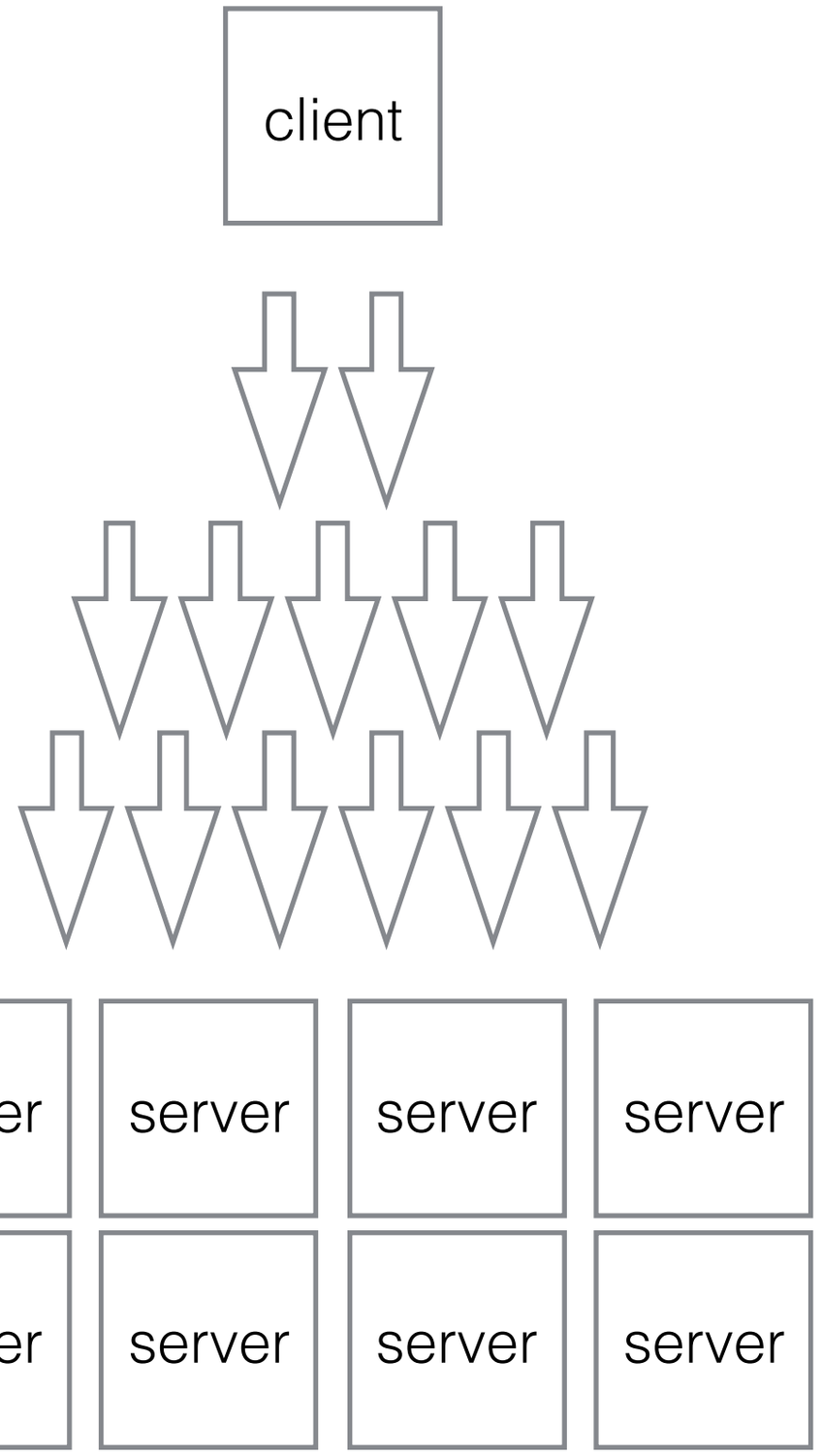


Many tiny services

```
➜  wake-repo git:(master) find config -name manifest.json |wc -l
151
```

tiny requests





re-usable services >

re-usable components

Nodes are Disposable

Immutable Deployments

Never Upgrade Software on an Existing Node

Threw away Chef

“wake” = “wunder-make”

3. chad@Chads-MBP-2: ~/Documents/monolith (zsh)

```
➜ monolith  wake info as_files
0)  AS_files-pgbouncer_production  ec2-54-75-141-98.eu-west-1.compute.amazonaws.com    10.85.20.104    i-145e1557  2014-09-27T03:07:27.000Z  b  ami-a2e43fd5
 m3.medium
1)  AS_files_production            ec2-54-78-168-3.eu-west-1.compute.amazonaws.com    10.193.130.101  i-99e5d27c  2014-11-05T09:34:17.000Z  a  ami-0c58fa7b
 c3.xlarge
2)  AS_files_production            ec2-54-216-126-12.eu-west-1.compute.amazonaws.com  10.97.19.136    i-9be5d27e  2014-11-05T09:34:17.000Z  a  ami-0c58fa7b
 c3.xlarge
3)  AS_files_production            ec2-54-74-167-255.eu-west-1.compute.amazonaws.com  10.33.40.43     i-9ae5d27f  2014-11-05T09:34:17.000Z  a  ami-0c58fa7b
 c3.xlarge
4)  AS_files_production            ec2-54-78-199-87.eu-west-1.compute.amazonaws.com   10.83.139.59    i-1d3f3b5e  2014-11-05T09:34:18.000Z  b  ami-0c58fa7b
 c3.xlarge
5)  AS_files_production            ec2-54-220-210-198.eu-west-1.compute.amazonaws.com 10.101.2.226    i-1e3f3b5d  2014-11-05T09:34:18.000Z  b  ami-0c58fa7b
 c3.xlarge
6)  AS_files_production            ec2-54-73-148-53.eu-west-1.compute.amazonaws.com   10.86.7.5       i-1f3f3b5c  2014-11-05T09:34:18.000Z  b  ami-0c58fa7b
 c3.xlarge
7)  AS_files_production            ec2-54-170-187-191.eu-west-1.compute.amazonaws.com 10.91.128.160   i-87bba7c5  2014-11-05T09:34:19.000Z  c  ami-0c58fa7b
 c3.xlarge
8)  AS_files_production            ec2-54-216-255-152.eu-west-1.compute.amazonaws.com 10.126.137.96   i-a3b7abe1  2014-11-05T09:34:19.000Z  c  ami-0c58fa7b
 c3.xlarge
9)  AS_files_production            ec2-54-195-205-215.eu-west-1.compute.amazonaws.com 10.89.144.222   i-a2b7abe0  2014-11-05T09:34:19.000Z  c  ami-0c58fa7b
 c3.xlarge
➜ monolith
```

3. chad@Chads-MBP-2: ~/Documents/monolith (zsh)

```
➜ monolith  wake describe-elb ical
total running: 6
revision: 394d1a8af207e3923345f5de4112c984f5c4e836
  image_id: ami-a0a609d7
    a 2
        in service: i-54e2bab1 i-53e2bab6
    b 2
        in service: i-5abe9319 i-5bbe9318
    c 2
        in service: i-cefdfb8c i-cffdfb8d
➜ monolith  wake scale -a ical -n 3
Removing 3 instances...
wake contract -a ical -e production -n 3 -t AS
Terminating 3 instances...
wake terminate i-54e2bab1 i-53e2bab6 i-5bbe9318
Deregistered i-54e2bab1 i-53e2bab6 i-5bbe9318 from ELB
Waiting for 10 secs..
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-53e2bab6",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
```

url	https://github.com/6wunderkinder/activities
manifest	show manifest
travis	build unknown

production Servers (calculate cost)

AS	count	3
	current ami	ami-a2e653d5 - 63a166bc8 (22 days ago)
	instances	show instances

experimental Servers (calculate cost)

AS	count	0
	current ami	nothing deployed
	instances	show instances

Last Commits (upto 10)

To force a rebuild an ami, you can `git commit --allow-empty -m "Rebuild ami"`.

commit	author	commit time	last ami	last ami's env
63a166bc89 – change default logo to https	Chad Fowler <chad@chadfowler.com>	22 days ago	ami-a2e653d5	production
64059e7453 – Grey border	Timothy Achumba <tim@6wunderkinder.com>	about a month ago	ami-12c76765	production

multi-tiered architecture

- Socket
- Proxy
- English
- German
- “Write layer”

Message Queuing

- Standardized Mutation flow
- Feeds real-time synchronization to clients
- Feeds anything else, by convention

“real time”

“smart” proxy

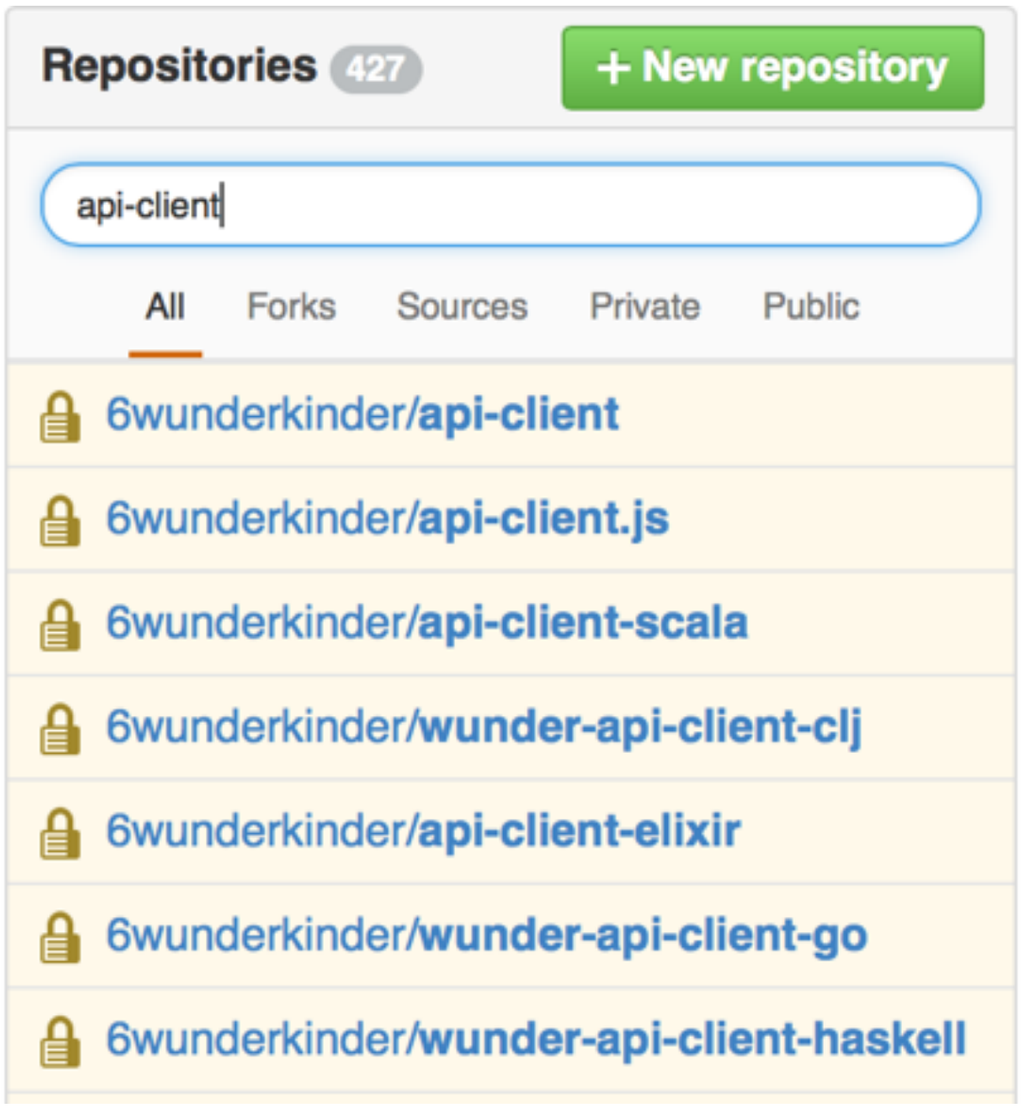
authorization

authentication

Convention Over Configuration

```
for service in $SERVICE_NAMES; do
  rails new $service -m service_template.rb
  cd $service
  git init
  git add . && git commit -m "Initial commit"
done
```

Convention Over Configuration



Service Resolution

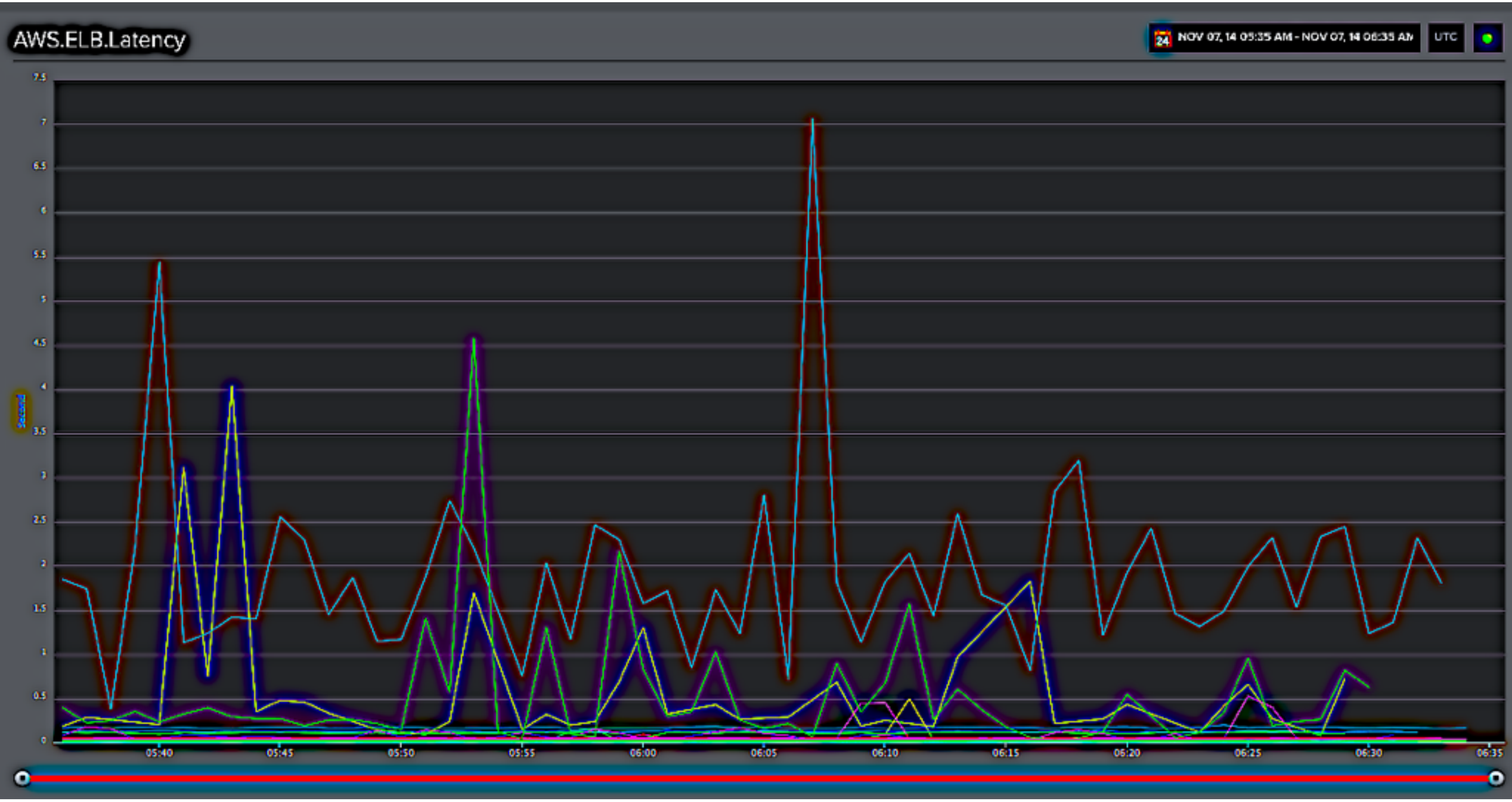
YAML -> JSON -> Auto-generated JSON -> Consul

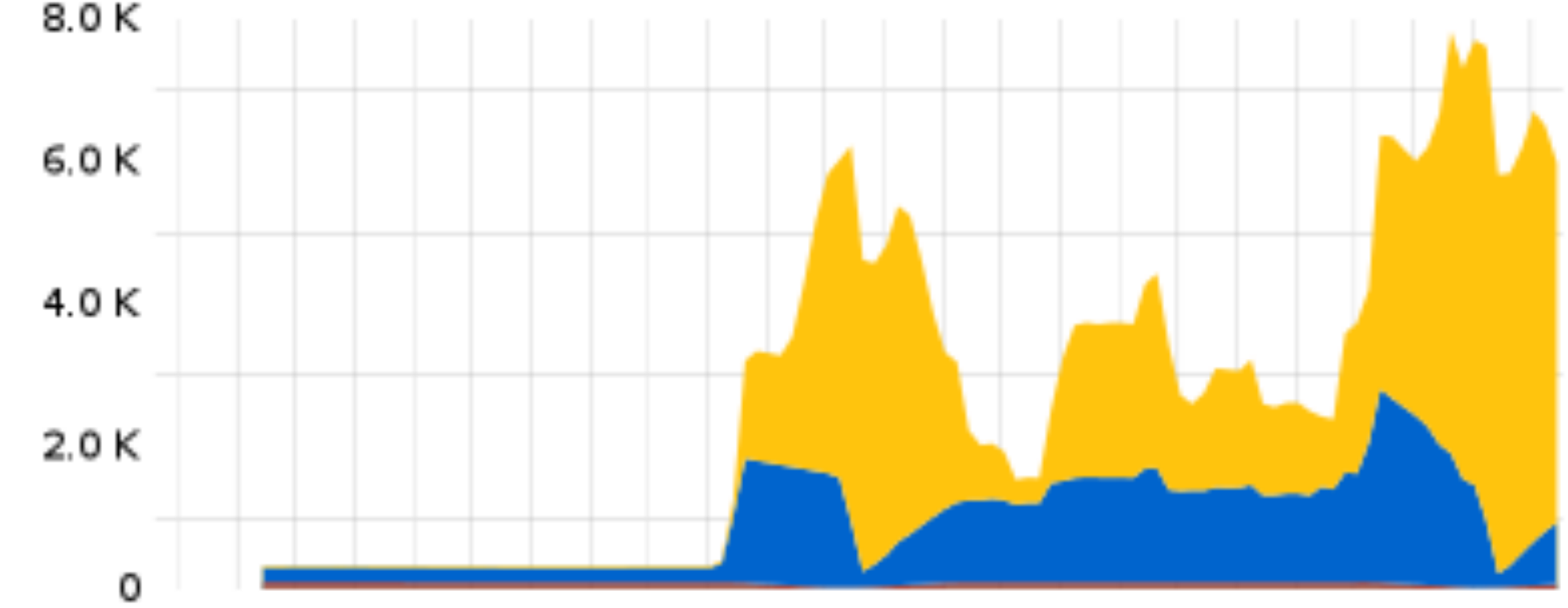
https://codeascraft.com/2011/02/15/measure-anything-measure-everything/

Measure Anything, Measure Everything

Posted by **Ian Malpass** on February 15, 2011

If Engineering at Etsy has a religion, it's the Church of Graphs. If it moves, we track it. Sometimes we'll draw a graph of something that isn't moving yet, just in case it decides to make a run for it. In general, we tend to measure at three levels: network, machine, and application. (You can read more about our graphs in Mike's Tracking Every Release post.)


AWS.ELB.Latency
NOV 07, 14 05:35 AM - NOV 07, 14 06:35 AM
UTC
Second
7.5
7
6.5
6
5.5
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
05:40
05:45
05:50
05:55
06:00
06:05
06:10
06:15
06:20
06:25
06:30
06:35


8.0 K
6.0 K
4.0 K
2.0 K
0

How We Migrated

- Removed joins
- Separated databases
- New features prototyped new service approach (comments, files)
- New features prototyped real-time/mutation approach (comments)
- Replace database connections in old API with new APIs (some ran as long as 7 months)
- Test under crazy load/behavior (next slide)

Böse Gurken

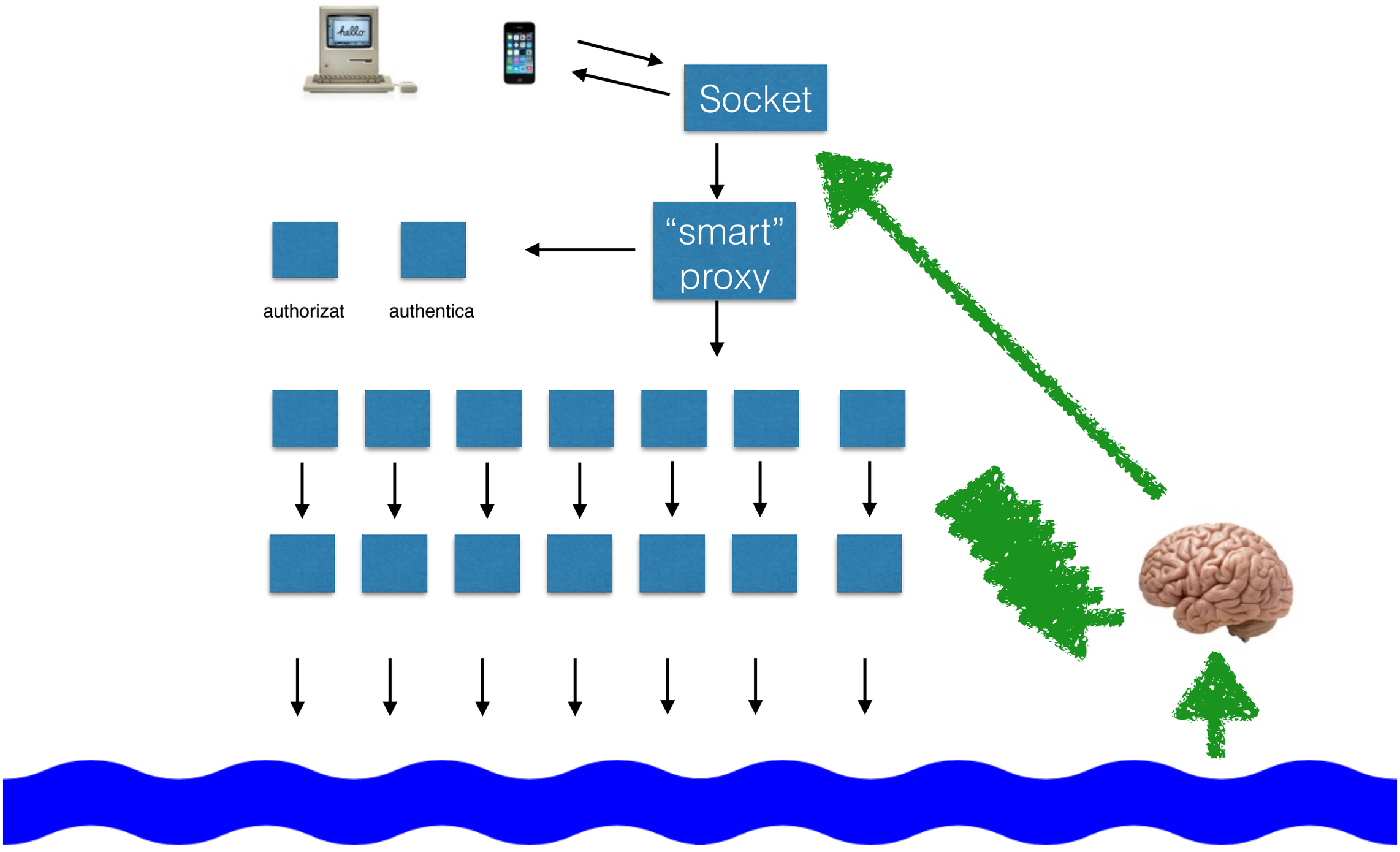
```
def receive = {
  case CreateUsers(userCount) ⇒ {
    val user = context.actorOf(Props(classOf[GurkeUser]))
    user ! SignUp
    user ! LogIn
    user ! CreateGurken
  }
}
```


```
def receive = {
  case Tick ⇒ {
    roughlyHalfTheTime { createAList }
    prettyOften { createATask }
    roughlyHalfTheTime { completeAtTask }
    // and so much many more actions...
  }
}
```

https://medium.com/tales-from-6w/saving-our-bacon-with-evil-cucumbers-2e3d26cebacc

Wunderlist 3 is Here

The ~~Big~~ *Little* Rewrites



The Future

homeostasis

Homeostasis

Definition

noun

(Science: Biology)

(1) The tendency of an organism or a cell to regulate its internal conditions, usually by a system of feedback controls, so as to stabilize health and functioning, regardless of the outside changing conditions

(2) The ability of the body or a cell to seek and maintain a condition of equilibrium or stability within its internal environment when dealing with external changes

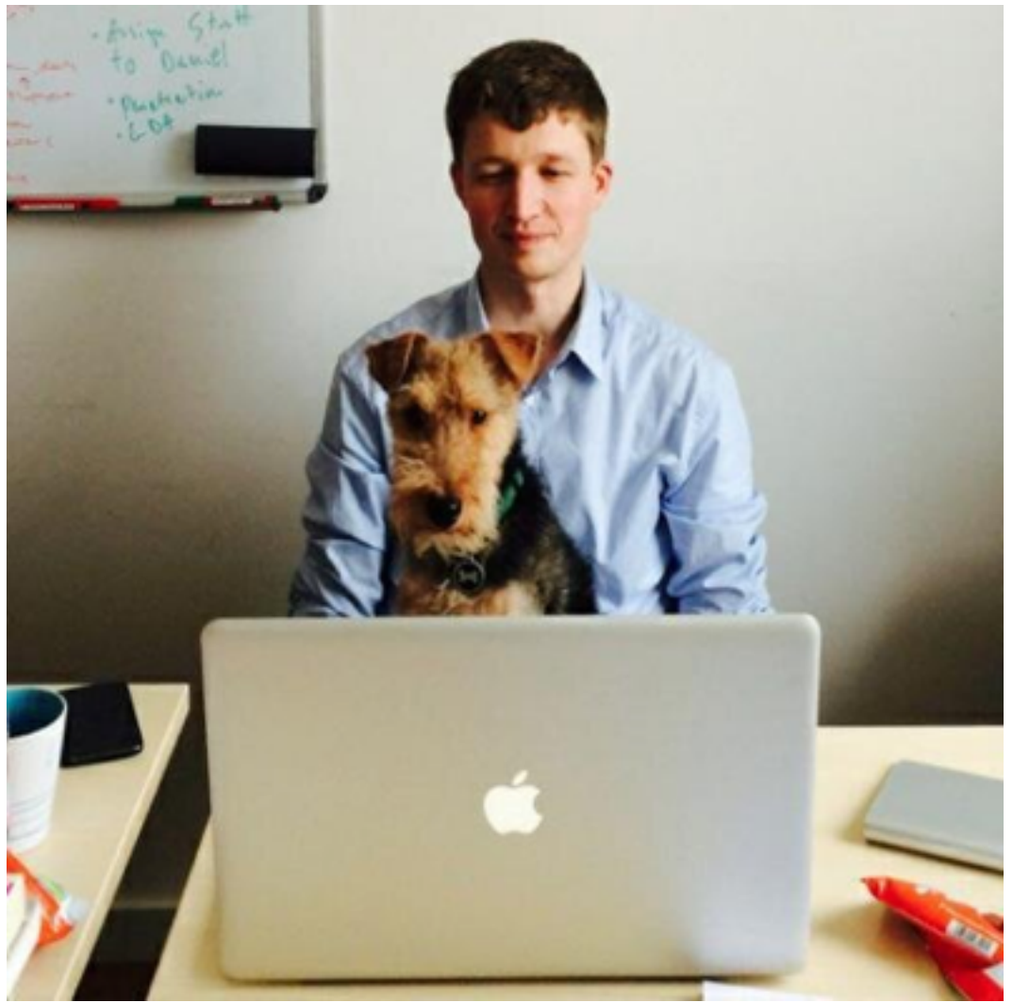
Global
async
validation
middleware

Cost Reduction

- Make it work
- Make it fast
- Make it cheap

http://hans.io/blog/2015/05/05/spot_instances/

Hans Hasselberg

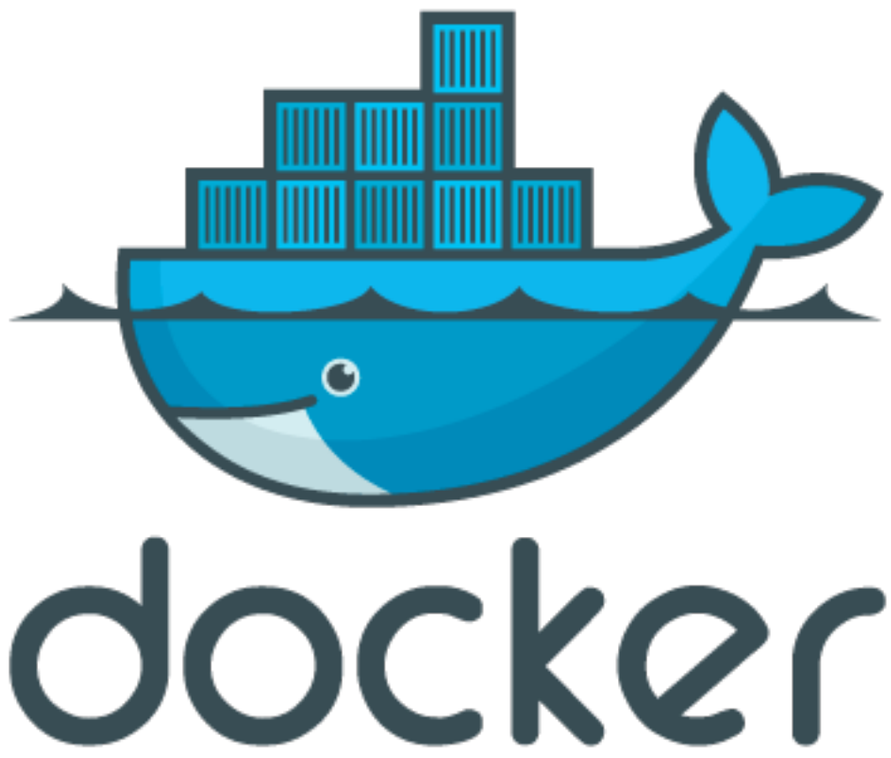


How we use AWS Spot Instances

05 May 2015

Spot instances are just like normal ondemand instances from Amazon Web Services(AWS). There are two differences: they can be way cheaper and they can go away any time. At 6Wunderkinder we want to use them because of the savings. We are planning to use spot instances just like we use ondemand now. This blog post will explain how we deal with spot instances and the risk of losing them.

Deployment Speed



Francesco Cesarini
@FrancescoC

Following

As we get more & more sophisticated microservice implementations, each one grows their own crappy version of Erlang: infoq.com/news/2015/04/n…

RETWEETS 31 FAVORITES 26

1:24 AM - 25 Apr 2015

Reply to @FrancescoC

Chad Fowler @chadfowler · 8h
@FrancescoC reflecting on whether we have created a crappy Erlang at @Wunderlist

Investigate: How does this work in client code?

Refactor to Monolith

CHICAGO
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE 2015

goto; conference

Questions?

Please remember to evaluate via the GOTO Guide App