Managing Massive Datacenters

Luke Kanies

Founder and CEO, Puppet Labs

luke@puppetlabs.com

@puppetmasterd

Question

Authority

puppet labs

Puppet: Open Source Systems Management

- You explain to Puppet how you want your infrastructure to look, and it makes it so
- As you change that explanation over time, Puppet updates all machines to match

puppet labs

Puppet Users

Deploy 1,800 machines in 2 hours
vs. 25 machines per day with HP Opsware

Scaled from 0 to over 10,000 servers
in 2 months without training

287 servers per SysAdmin
vs. 19 for BMC BladeLogic

Over 50,000 systems
managed by Puppet

Financial

TransUnion
BARCLAYS CAPITAL
JPMorgan
D E Shaw & Co
NYSE Euronext

Entertainment

Disney
VIACOM
Turner
PlayStation
FOX

Technology

redhat
DELL
MOTOROLA
Juniper NETWORKS
SIEMENS
rackspace

Defense

LOCKHEED MARTIN
NORTHROP GRUMMAN
Los Alamos NATIONAL LABORATORY

Web

Google
ebay
twitter
digg
zynga

puppet labs

Managing Datacenters

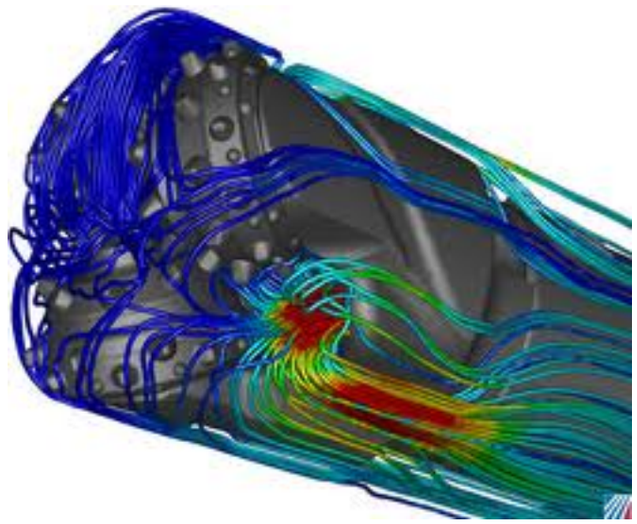
There's no pattern

- Enterprise vs. growth company
  - E.g., Nokia vs. Zynga
- Technology company vs. startup
  - Twitter today vs. Twitter 2 years ago
- Yours vs. Mine
  - Rackspace vs. Amazon vs. Google
- Big vs. really really big
  - Wal Mart vs. JPMorgan
- Corporate vs. Customer-facing
  - Google Production vs. Google Corporate

puppet labs

Web ops are cooler, corporate is harder

Web operations looks a lot like HPC



http://www.cfdesign.com/image/legacy/news/2009/tricone-bit-HPC.jpg?w=350&h=288&as=1

puppet labs

Lots of apps != lots of users

It can feel like Silicon Valley vs. The World

Over time, web ops companies become corporate

How are datacenters managed?

It’s worse than you think

Worse than that

No no - not you

But probably the person next to you

puppet labs

Technology still can't solve political problems

$25m application with 100s of trivial root exploits

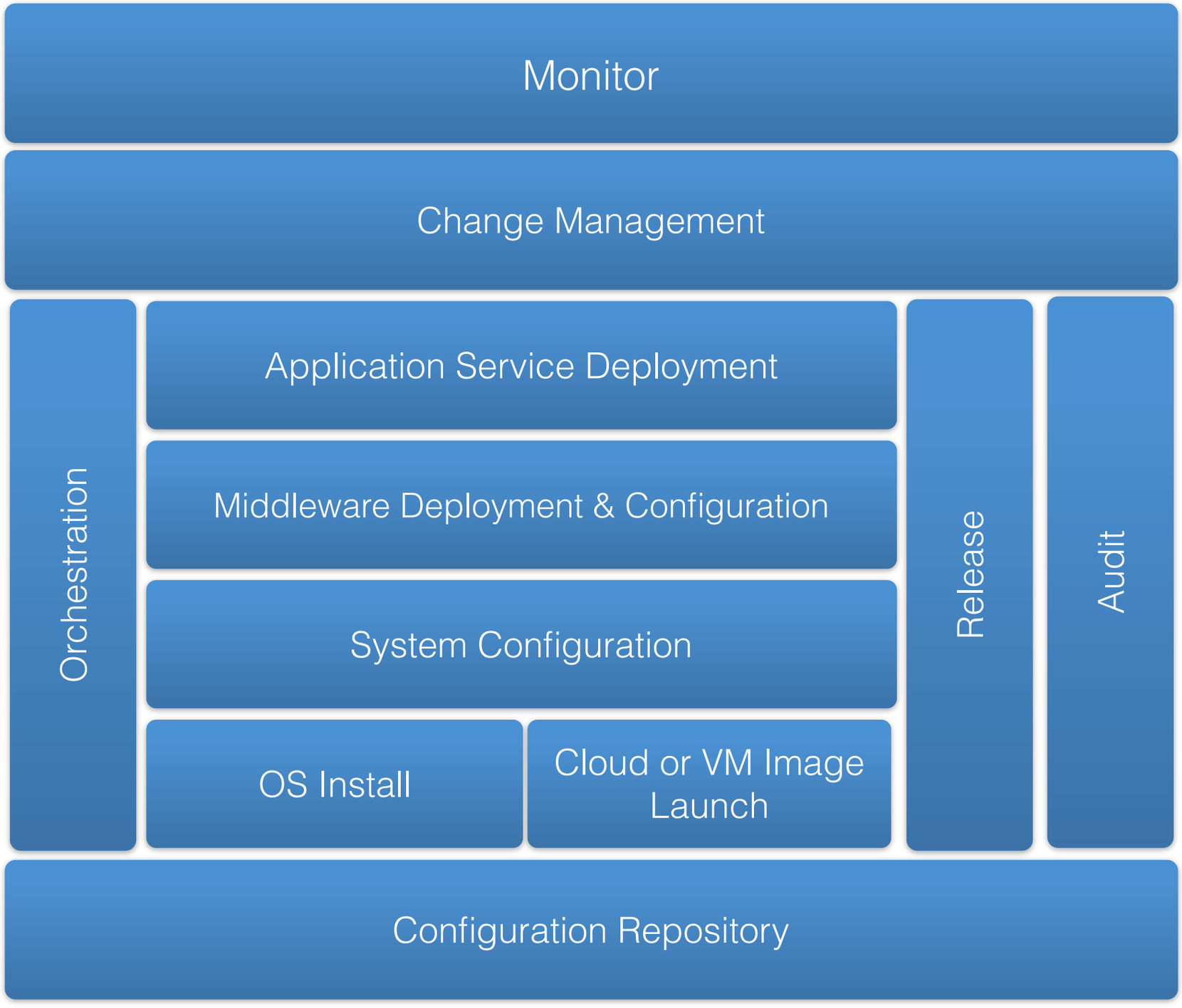
Deployment costing more than development

Six months with no upgrades because they *forgot how*

50-96% of outages caused by human error

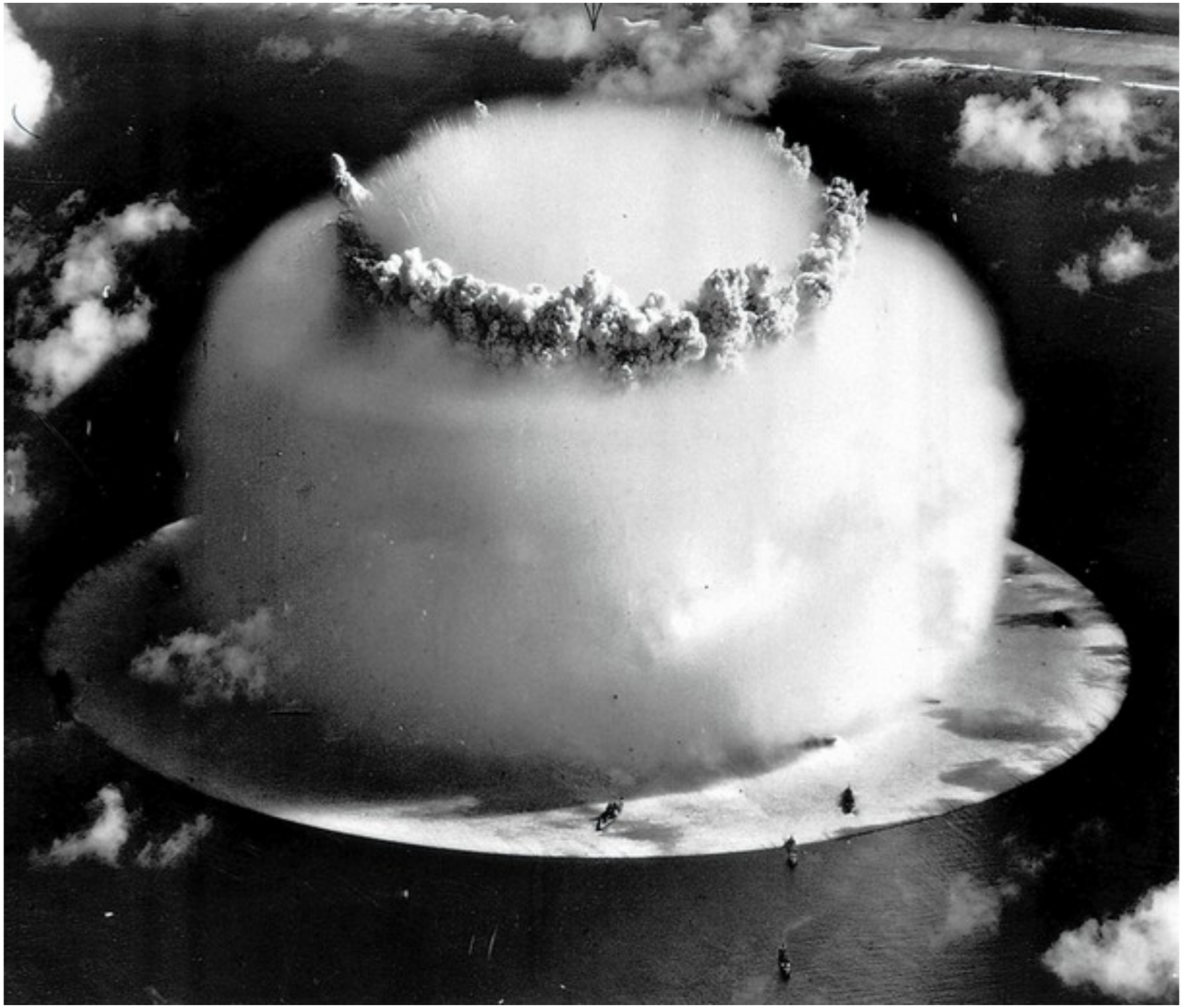
puppet labs

The problem actually is complex



puppet labs

Disaster is constant



Solution: Enterprise-grade management solution

They suck

http://t1.gstatic.com/images?q=tbn:ANd9GcSGbCu2jBwJBMo7iO4IP3CiwSTHkIIVi_ppxWrK2cq5Wjmi2rt24A&t=1

puppet labs

Built for protection, not productivity

http://www.deviantart.com/download/97345411/Straight_jacket_Joker_by_MasterDrawer.jpg

puppet labs

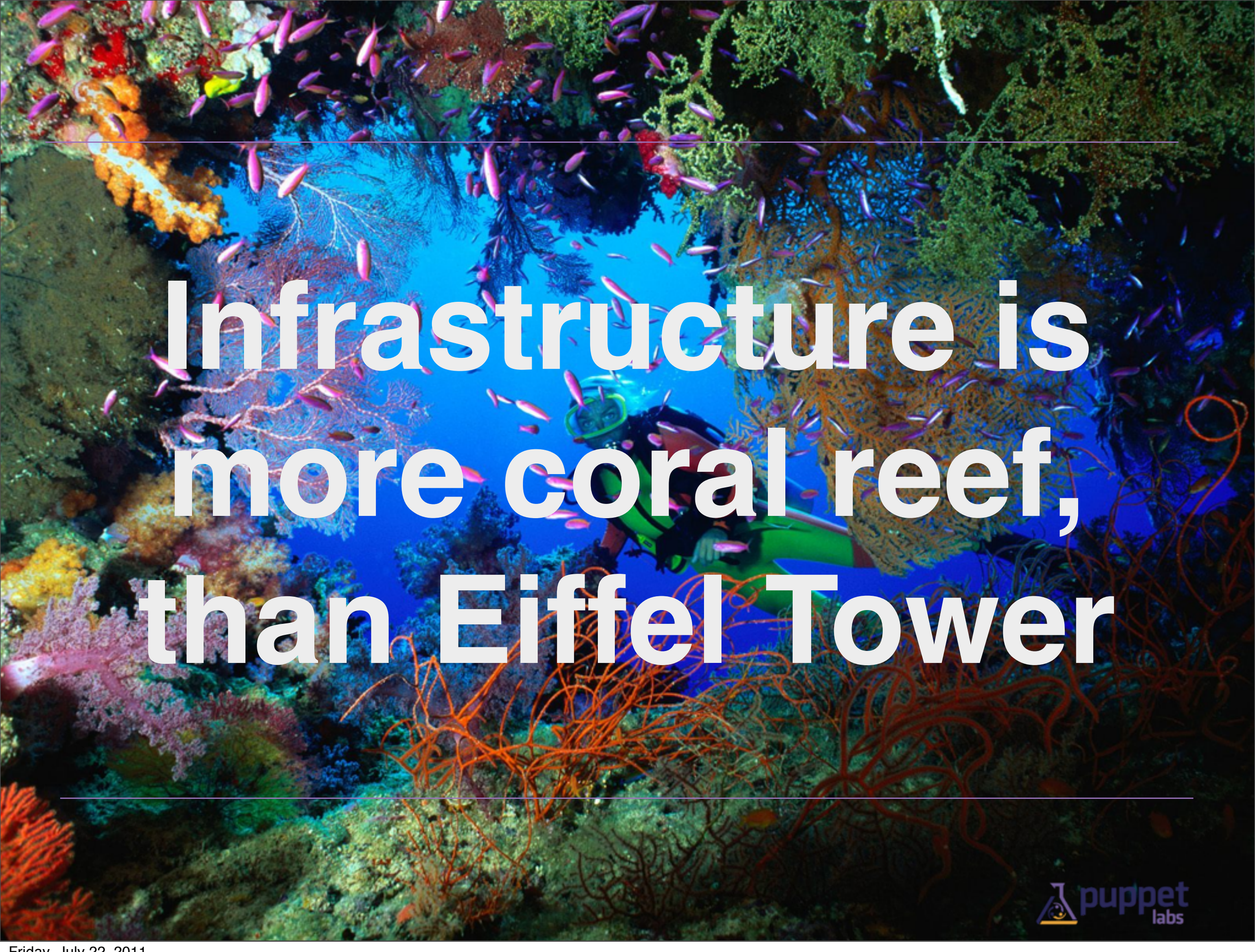
Claim to solve every problem

Reality: Shell scripts and cron jobs

...and you wasted

$15m

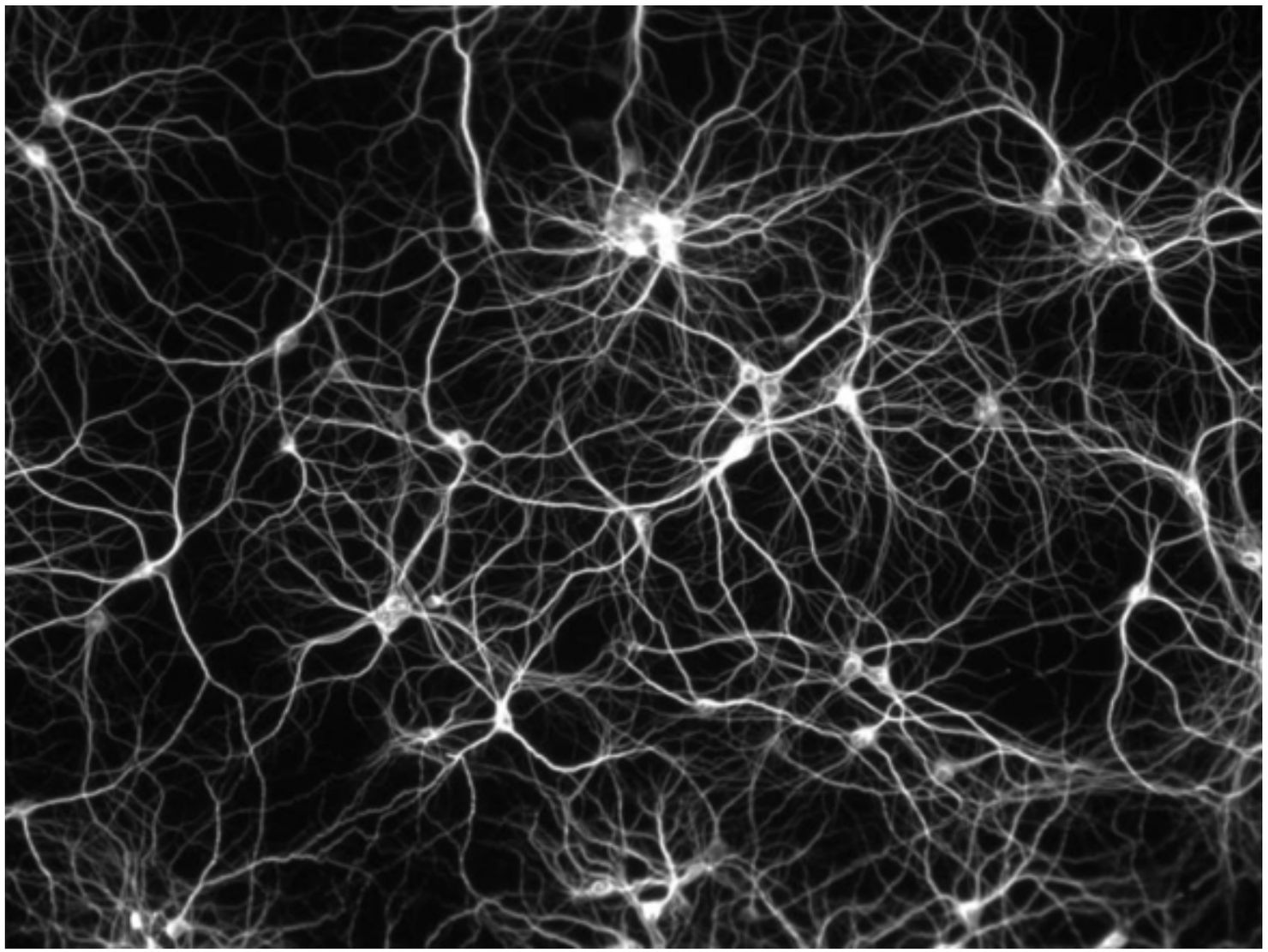
...and an exec who bet his career on it


Infrastructure is
more coral reef,
than Eiffel Tower
puppet
labs

Simplicity scales, complexity crumbles

puppet labs

Scalable complexity requires simplicity



http://freeassociationdesign.files.wordpress.com/2010/04/neuron-map1.jpg

puppet labs

It’s even more so the cloud

Simplicity provides frictionless scale

puppet labs

High cost, no agility, big resources kill you

How did we get here?

puppet labs

Jenga administration

http://www.thecolorcurve.com/blog/wp-content/uploads/2011/02/jenga_negativeskew.jpg

puppet labs

Theory: Deploy, use, decommision

Reality: Deploy, back away slowly

People fear change, rather than enable it

http://www.freefoto.com/images/04/28/04_28_57---Pile-of-Coins_web.jpg

puppet labs

Poor teamwork

Development produces the world's best app

Ops has no idea how to deploy it

Auditors won't sign off on it

No one's fault, everyone's problem

Consultants are happy to “help”

Devops : Ops ::

Agile : Development

Cultural change takes forever

The best have learned

It can’t go on like this forever

Image source: http://sloblogs.thetribunenews.com/slovault/files/2008/08/pacific-telephone.jpg

puppet labs

They build their own

amazon.com

ebay™

Google

E*TRADE

Morgan Stanley

puppet labs

=> Lots of

duplicated effort

And thrown away code

puppet labs

They're still the exception

puppet labs

Normal: Purchased, hand-rolled, and broken

Operational Antipatterns

Direct from keyboard to production

Alerts delivered via email

Expecting tools to protect you from incompetence

puppet labs

Golden Images

Image from http://www.flickr.com/photos/fungep/2516767121/sizes/l_

What you should do instead

Step 1

puppet labs

Make everyone responsible for operational success

See the change, not the state

Think services, not hosts

Use the stupidest tool that can possibly work

Find where

people's time goes

Solve the simple problems first

puppet labs

Focus on value/ cost

Only monitor things you care about

puppet labs

Continuous integration, even in the infrastructure

Keep humans at the console

puppet labs

Automate tasks,

not decisions

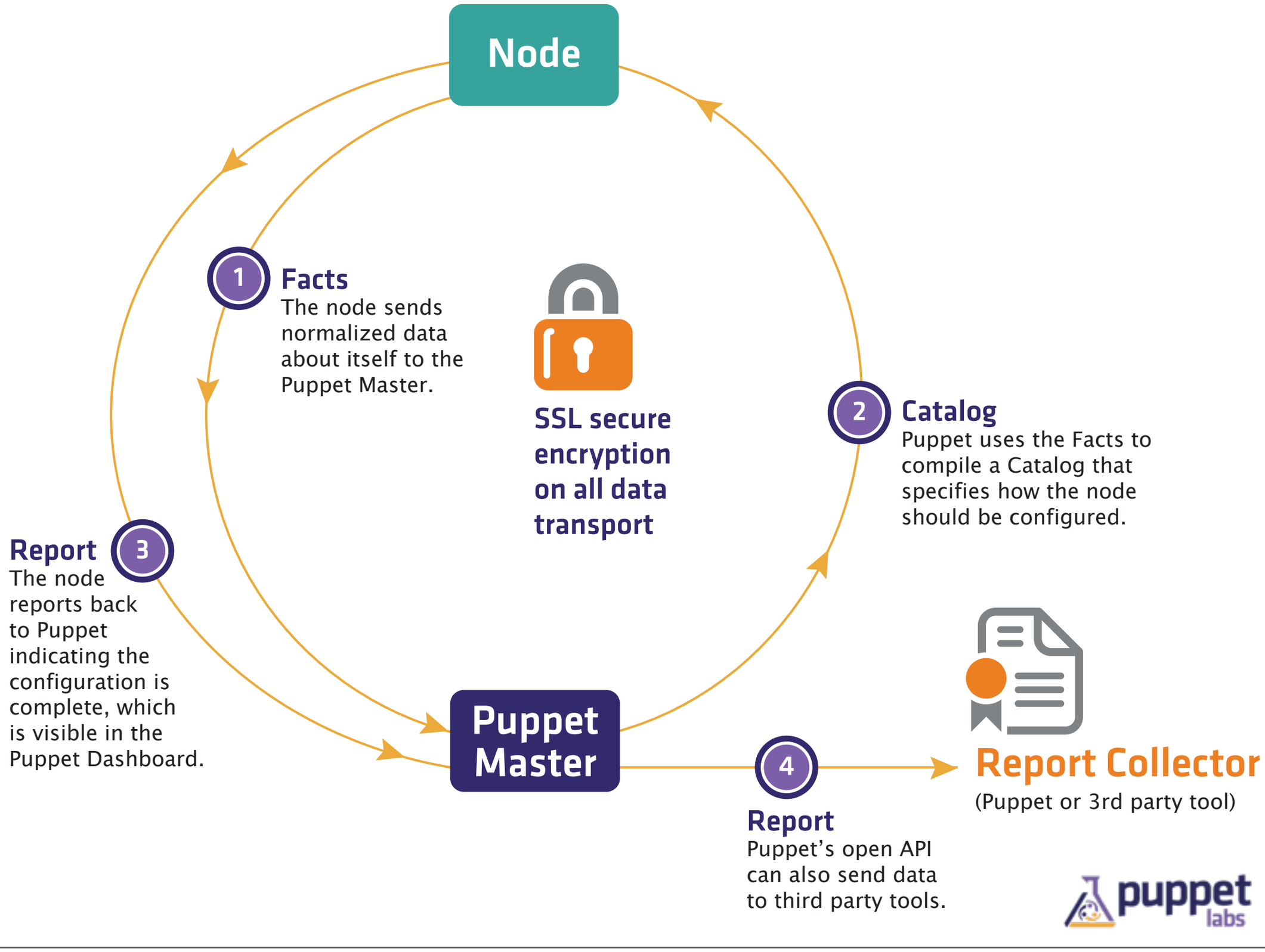
puppet labs

A little more about

Puppet

puppet labs

Constant Configuration



Simple Configuration Language

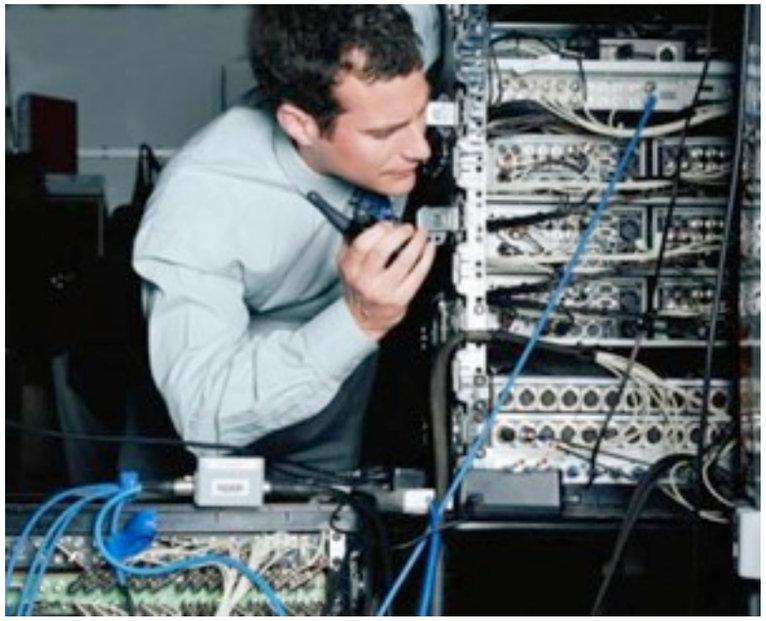
```
class ssh {
    package { ssh: ensure => installed }
    file { sshd_config:
        name => "/etc/ssh/sshd_config",
        owner => root,
        group => root,
        source => "puppet://server/apps/ssh/sshd_config",
        after => Package[ssh]
    }
    service { sshd:
        ensure => running,
        subscribe => [Package[ssh], File[sshd_config]]
    }
}
```

puppet labs

System Portability with the Resource Abstraction Layer

puppet labs

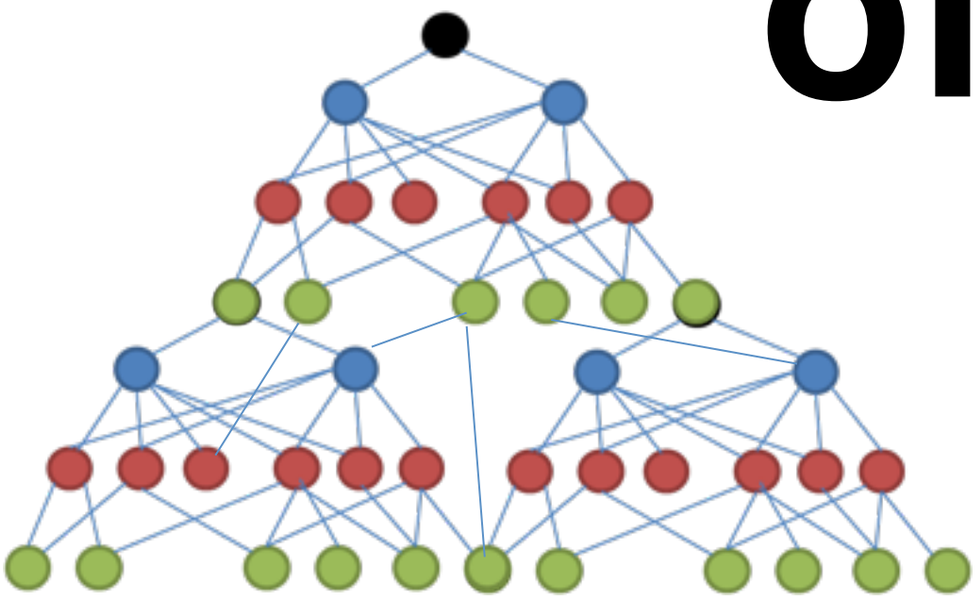
We're building Iron Man, not robots



Usability first, capability second

Separation of specification and execution

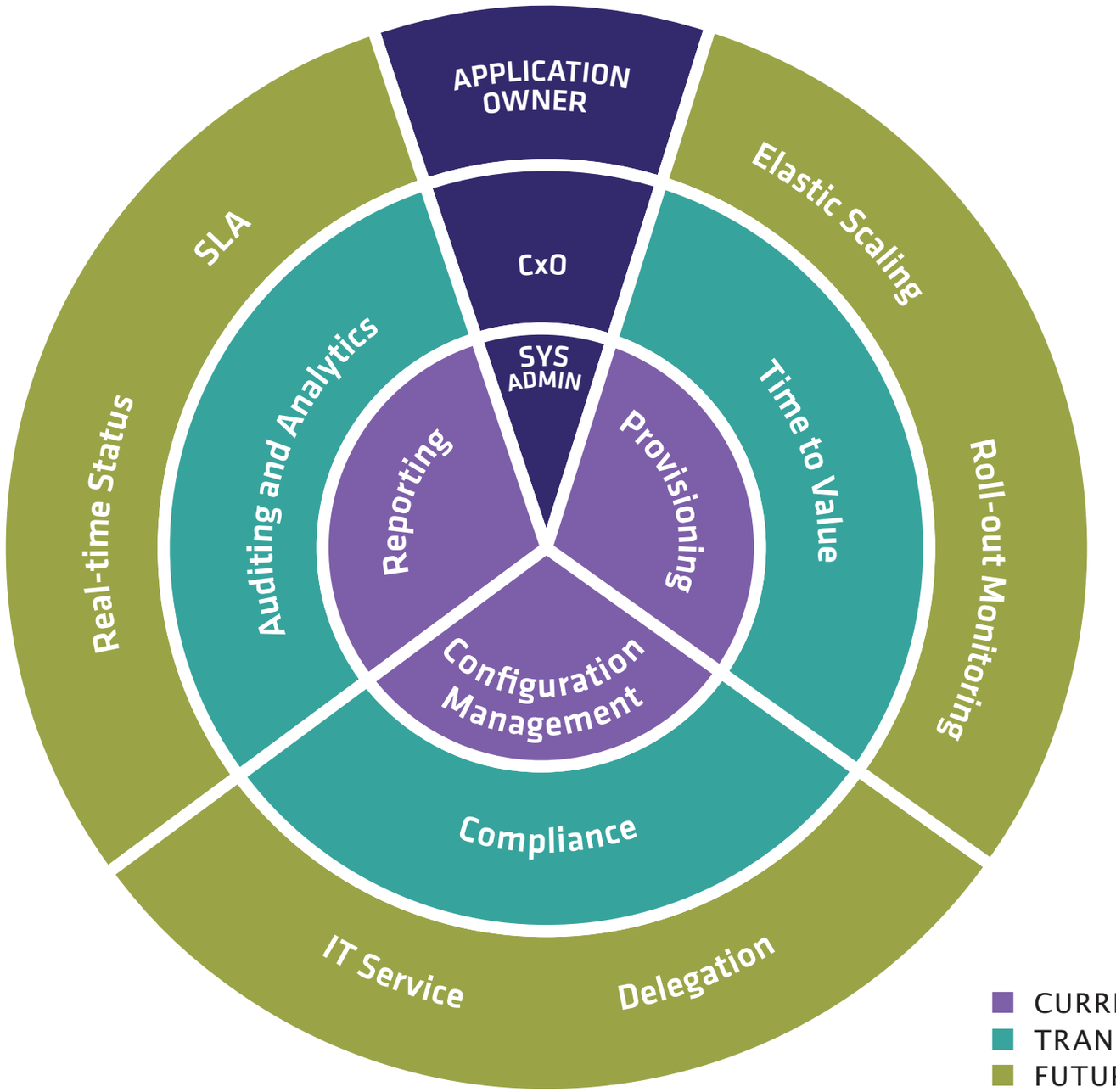
Massive amounts of data



puppet labs

Portability enables DRY Operations

The Future of the Puppet Platform



puppet labs

Summary

The state of operations is still really bad

puppet labs

The cloud is making it worse, or better, or something

puppet labs

Trying to solve it yourself makes you a software company

puppet labs

But you might have no choice

puppet labs

So build something simple and be ready to toss it

puppet labs

Or just use Puppet :)

Questions?