Scaling a Rails Application from the Bottom Up

Jason Hoffman PhD, Founder and CTO, Joyent
Rails Conf Europe 2007
Monday, September 17, 2007

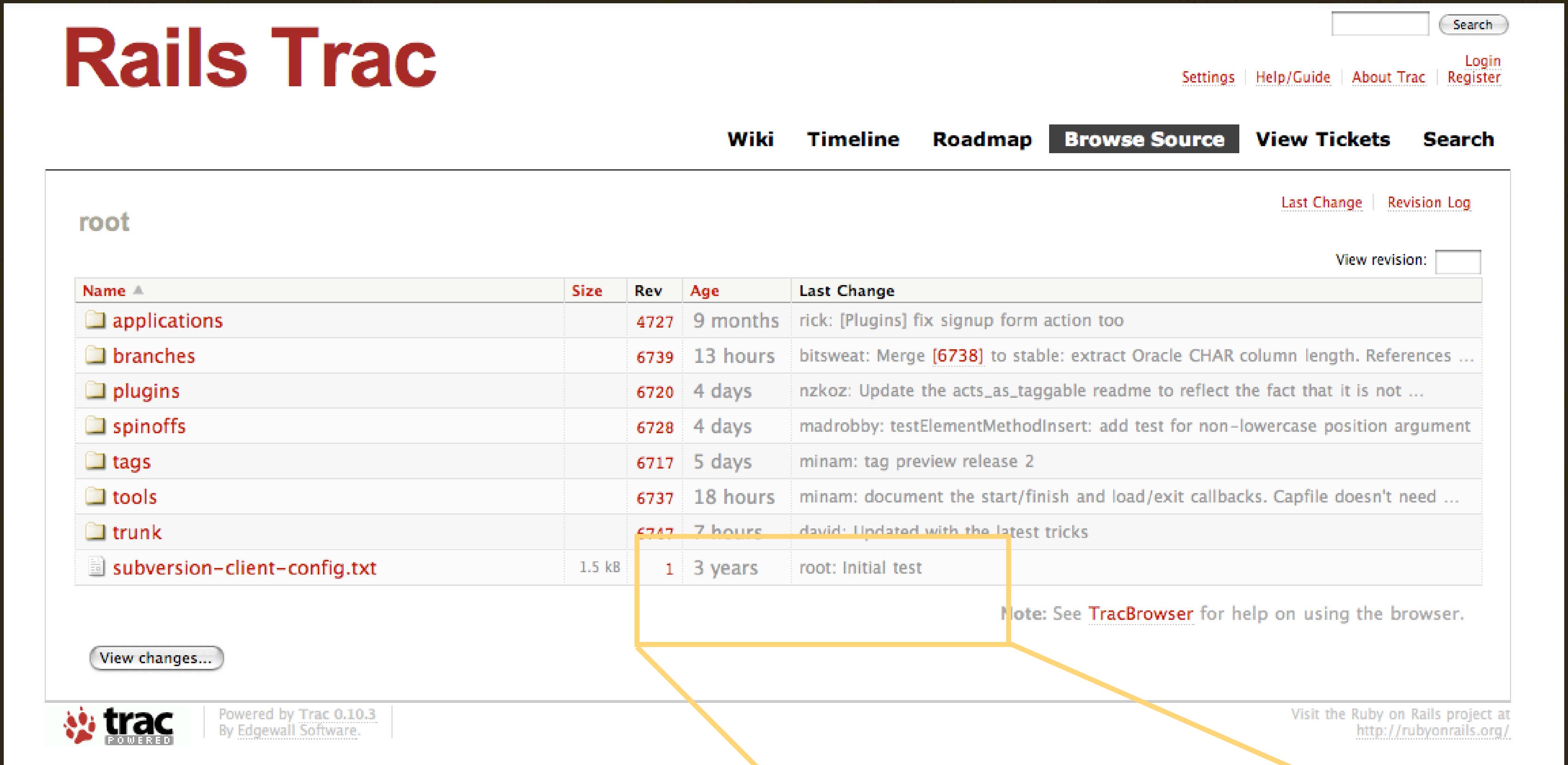
Joyent

The questions

- Can I write a [Rails] application where I can start in a datacenter in San Francisco, then drop copies in Tokyo, Hong Kong, Virginia, London, Amsterdam, and Frankfurt while maintaining a unified backend, differentially directing people based on their geography and having my costs only go from 1x to 7x?
- Can I write a [Rails] application where my user base grows from 10,000 to 10,000,000 and have my per user costs stay the same or less?
- Can I write a [Rails] application and have it run on an device like an iPhone?

Hi, I'm Jason

Joyent



Rails Trac

Search
Login
Settings | Help/Guide | About Trac | Register
Wiki | Timeline | Roadmap | Browse Source | View Tickets | Search
Last Change | Revision Log
View revision:

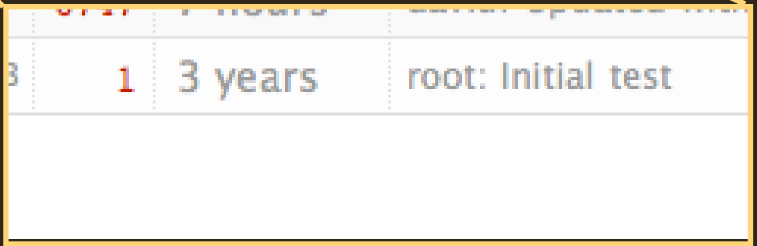
root

Name ▲	Size	Rev	Age	Last Change
applications		4727	9 months	rick: [Plugins] fix signup form action too
branches		6739	13 hours	bitsweat: Merge [6738] to stable: extract Oracle CHAR column length. References …
plugins		6720	4 days	nzkoz: Update the acts_as_taggable readme to reflect the fact that it is not …
spinoffs		6728	4 days	madrobby: testElementMethodInsert: add test for non-lowercase position argument
tags		6717	5 days	minam: tag preview release 2
tools		6737	18 hours	minam: document the start/finish and load/exit callbacks. Capfile doesn't need …
trunk		6747	7 hours	david: Updated with the latest tricks
subversion-client-config.txt	1.5 kB	1	3 years	root: Initial test

Note: See TracBrowser for help on using the browser.

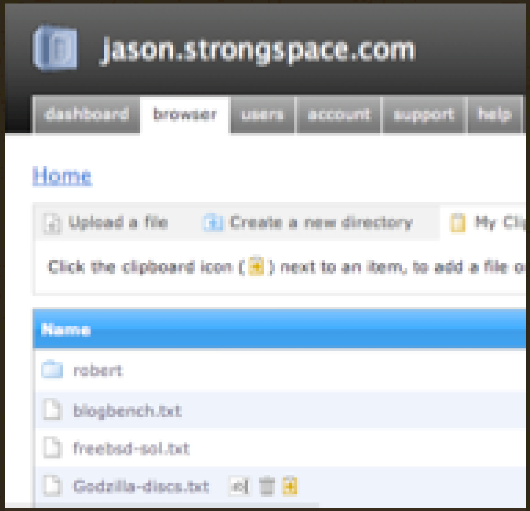
Powered by Trac 0.10.3
By Edgewall Software.

Visit the Ruby on Rails project at
http://rubyonrails.org/

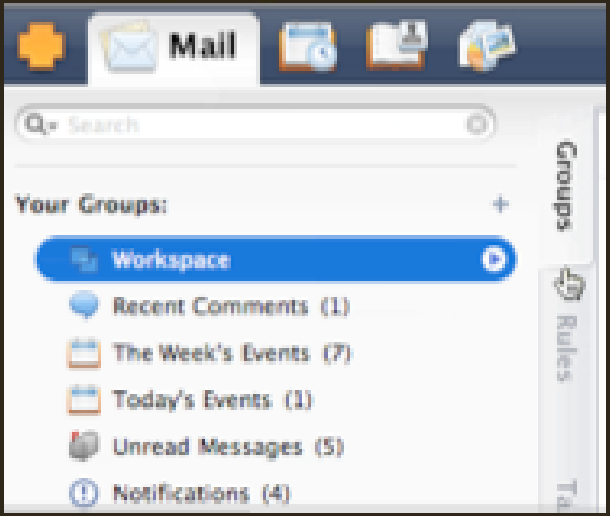


1 3 years root: Initial test

Some of Joyent's



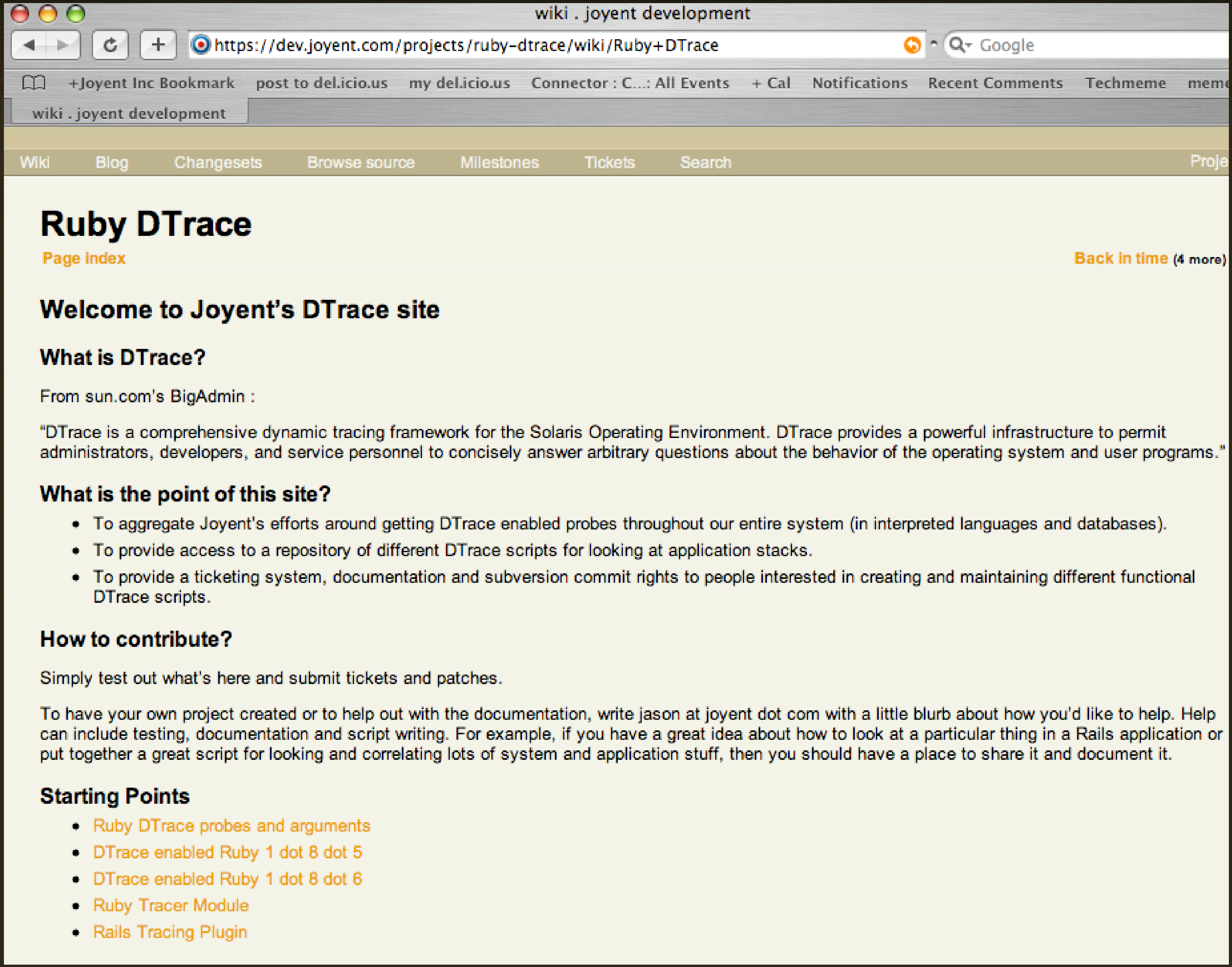
Click the clipboard icon ( ) next to an item, to add a file o



Bingo!

- 100 gigabytes o
- 50 gigabytes of
- 25 gigabytes of

Welcome to wide op
you 100, 50 or 25 g
platform with a gene
for a super-low pric
from a public folder
a complimentary 5-
collaboration produ



Ruby DTrace

Page Index

Back in time (4 more)

Welcome to Joyent's DTrace site

What is DTrace?

From sun.com's BigAdmin :

“DTrace is a comprehensive dynamic tracing framework for the Solaris Operating Environment. DTrace provides a powerful infrastructure to permit administrators, developers, and service personnel to concisely answer arbitrary questions about the behavior of the operating system and user programs.”

What is the point of this site?

- To aggregate Joyent's efforts around getting DTrace enabled probes throughout our entire system (in interpreted languages and databases).
- To provide access to a repository of different DTrace scripts for looking at application stacks.
- To provide a ticketing system, documentation and subversion commit rights to people interested in creating and maintaining different functional DTrace scripts.

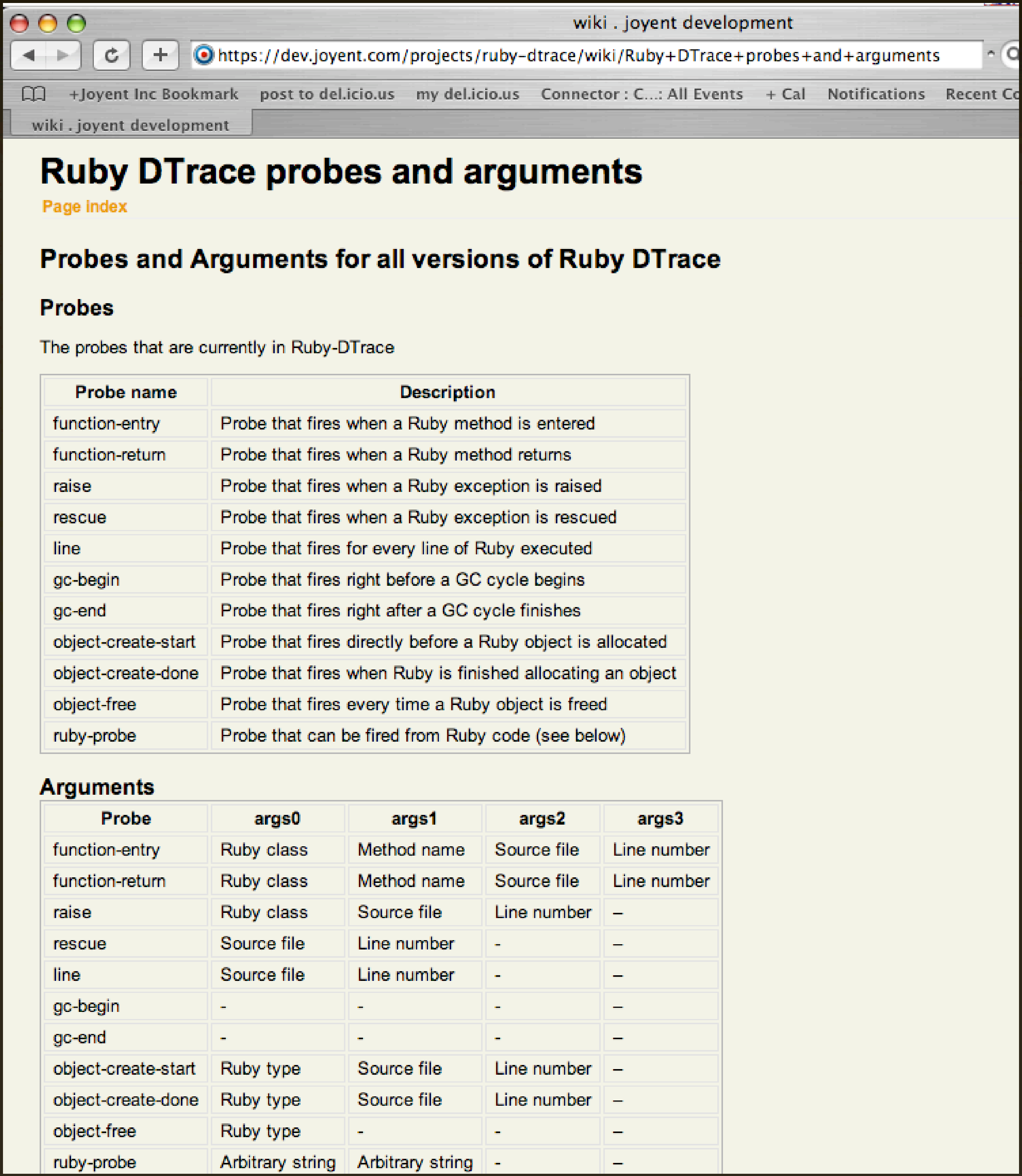
How to contribute?

Simply test out what's here and submit tickets and patches.

To have your own project created or to help out with the documentation, write jason at joyent dot com with a little blurb about how you'd like to help. Help can include testing, documentation and script writing. For example, if you have a great idea about how to look at a particular thing in a Rails application or put together a great script for looking and correlating lots of system and application stuff, then you should have a place to share it and document it.

Starting Points

- Ruby DTrace probes and arguments
- DTrace enabled Ruby 1 dot 8 dot 5
- DTrace enabled Ruby 1 dot 8 dot 6
- Ruby Tracer Module
- Rails Tracing Plugin


wiki . joyent development
https://dev.joyent.com/projects/ruby-dtrace/wiki/Ruby+DTrace+probes+and+arguments
+Joyent Inc Bookmark
post to del.icio.us
my del.icio.us
Connector : C...: All Events
+ Cal
Notifications
Recent Co
wiki . joyent development

Ruby DTrace probes and arguments

Page index

Probes and Arguments for all versions of Ruby DTrace

Probes

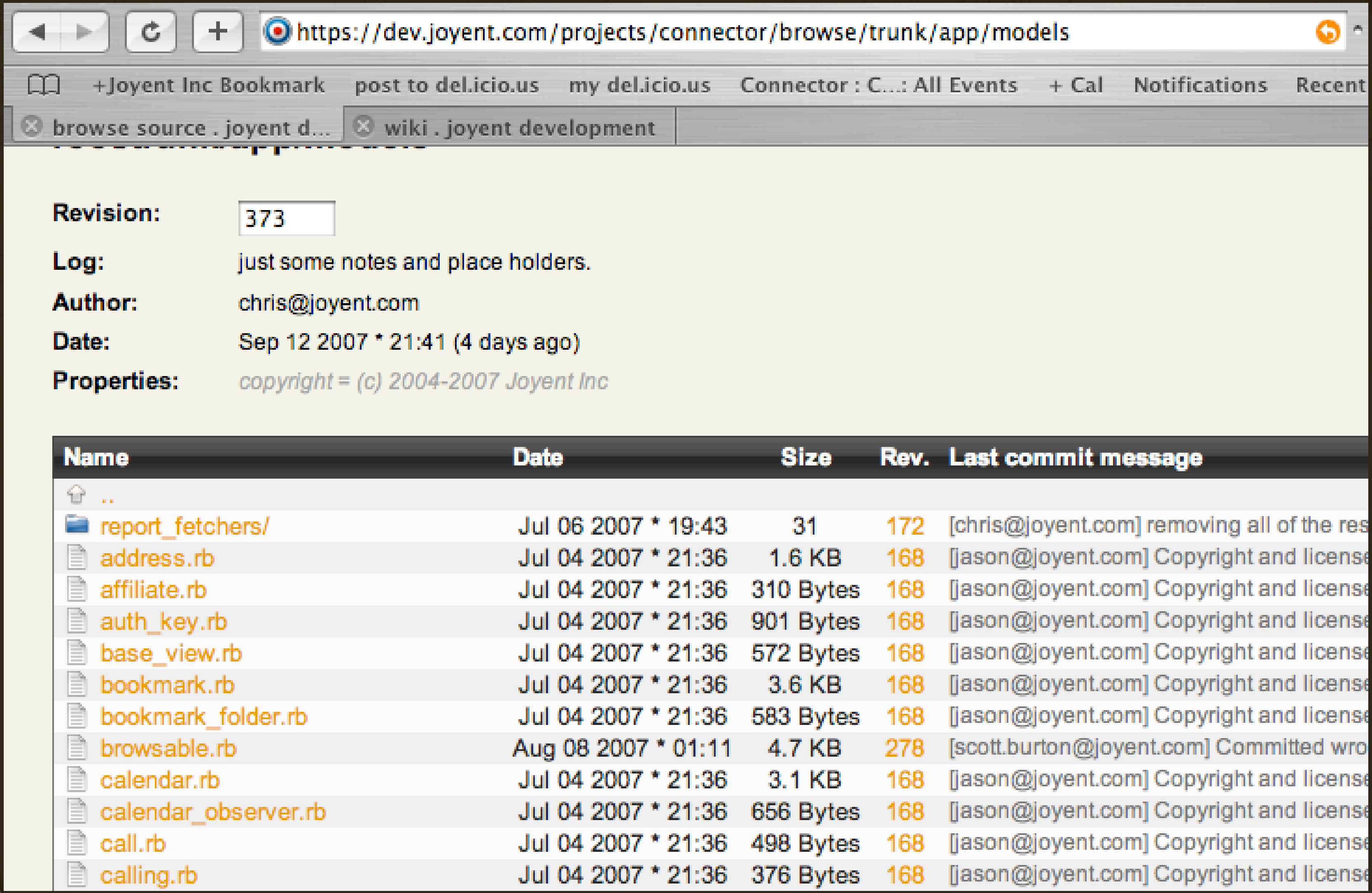
The probes that are currently in Ruby-DTrace

Probe name	Description
function-entry	Probe that fires when a Ruby method is entered
function-return	Probe that fires when a Ruby method returns
raise	Probe that fires when a Ruby exception is raised
rescue	Probe that fires when a Ruby exception is rescued
line	Probe that fires for every line of Ruby executed
gc-begin	Probe that fires right before a GC cycle begins
gc-end	Probe that fires right after a GC cycle finishes
object-create-start	Probe that fires directly before a Ruby object is allocated
object-create-done	Probe that fires when Ruby is finished allocating an object
object-free	Probe that fires every time a Ruby object is freed
ruby-probe	Probe that can be fired from Ruby code (see below)

Arguments

Probe	args0	args1	args2	args3
function-entry	Ruby class	Method name	Source file	Line number
function-return	Ruby class	Method name	Source file	Line number
raise	Ruby class	Source file	Line number	–
rescue	Source file	Line number	-	–
line	Source file	Line number	-	–
gc-begin	-	-	-	–
gc-end	-	-	-	–
object-create-start	Ruby type	Source file	Line number	–
object-create-done	Ruby type	Source file	Line number	–
object-free	Ruby type	-	-	–
ruby-probe	Arbitrary string	Arbitrary string	-	–

The connector


https://dev.joyent.com/projects/connector/browse/trunk/app/models
+Joyent Inc Bookmark
post to del.icio.us
my del.icio.us
Connector : C...: All Events
+ Cal
Notifications
Recent
browse source . joyent d...
wiki . joyent development

Revision: 373

Log: just some notes and place holders.

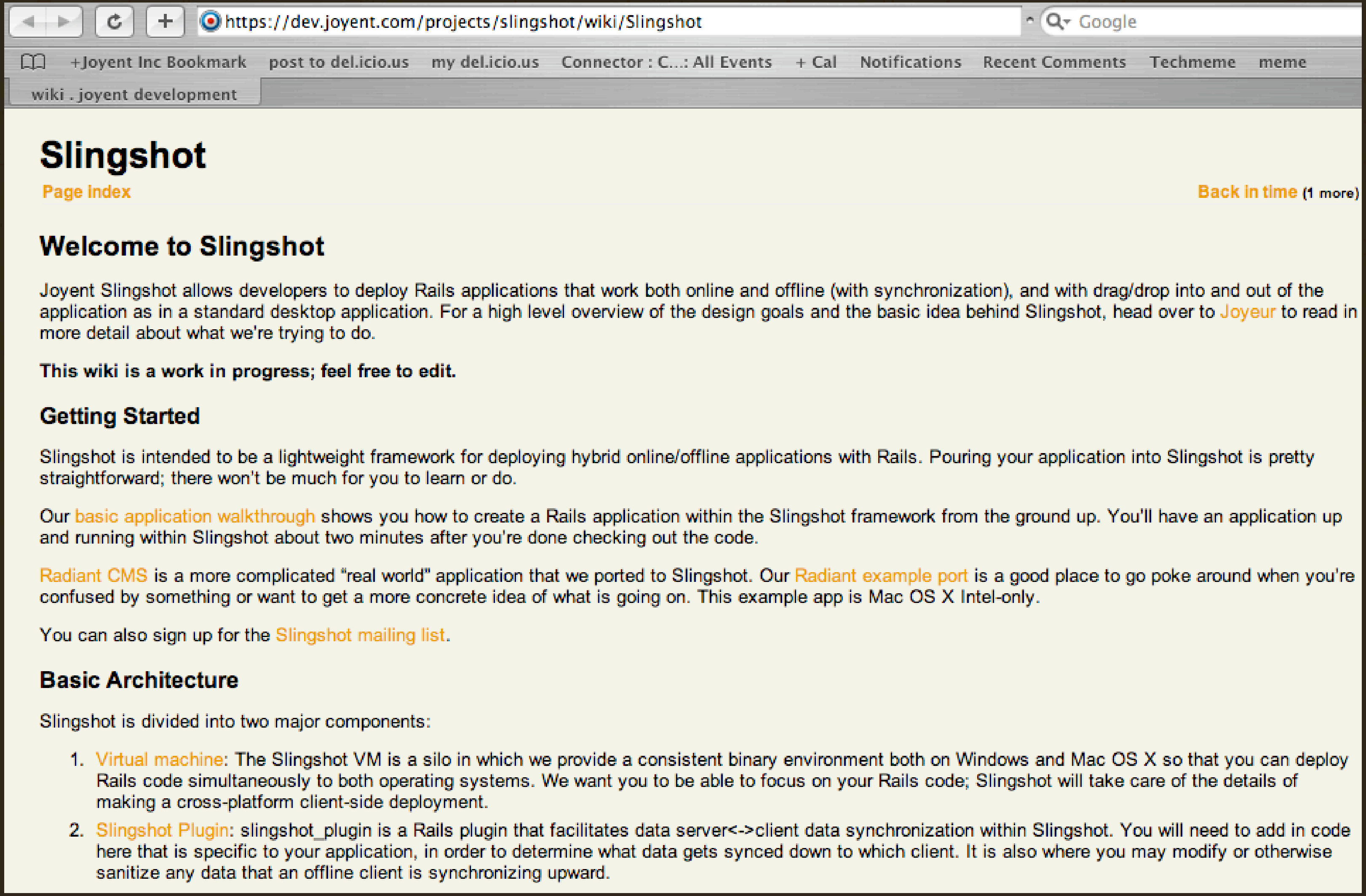
Author: chris@joyent.com

Date: Sep 12 2007 * 21:41 (4 days ago)

Properties: *copyright = (c) 2004-2007 Joyent Inc*

Name	Date	Size	Rev.	Last commit message
..				
report_fetchers/	Jul 06 2007 * 19:43	31	172	[chris@joyent.com] removing all of the res
address.rb	Jul 04 2007 * 21:36	1.6 KB	168	[jason@joyent.com] Copyright and license
affiliate.rb	Jul 04 2007 * 21:36	310 Bytes	168	[jason@joyent.com] Copyright and license
auth_key.rb	Jul 04 2007 * 21:36	901 Bytes	168	[jason@joyent.com] Copyright and license
base_view.rb	Jul 04 2007 * 21:36	572 Bytes	168	[jason@joyent.com] Copyright and license
bookmark.rb	Jul 04 2007 * 21:36	3.6 KB	168	[jason@joyent.com] Copyright and license
bookmark_folder.rb	Jul 04 2007 * 21:36	583 Bytes	168	[jason@joyent.com] Copyright and license
browsable.rb	Aug 08 2007 * 01:11	4.7 KB	278	[scott.burton@joyent.com] Committed wro
calendar.rb	Jul 04 2007 * 21:36	3.1 KB	168	[jason@joyent.com] Copyright and license
calendar_observer.rb	Jul 04 2007 * 21:36	656 Bytes	168	[jason@joyent.com] Copyright and license
call.rb	Jul 04 2007 * 21:36	498 Bytes	168	[jason@joyent.com] Copyright and license
calling.rb	Jul 04 2007 * 21:36	376 Bytes	168	[jason@joyent.com] Copyright and license

Slingshot


https://dev.joyent.com/projects/slingshot/wiki/Slingshot
Google
+Joyent Inc Bookmark
post to del.icio.us
my del.icio.us
Connector : C...: All Events
+ Cal
Notifications
Recent Comments
Techmeme
meme
wiki . joyent development

Slingshot

Page index

Back in time (1 more)

Welcome to Slingshot

Joyent Slingshot allows developers to deploy Rails applications that work both online and offline (with synchronization), and with drag/drop into and out of the application as in a standard desktop application. For a high level overview of the design goals and the basic idea behind Slingshot, head over to Joyeur to read in more detail about what we're trying to do.

This wiki is a work in progress; feel free to edit.

Getting Started

Slingshot is intended to be a lightweight framework for deploying hybrid online/offline applications with Rails. Pouring your application into Slingshot is pretty straightforward; there won't be much for you to learn or do.

Our basic application walkthrough shows you how to create a Rails application within the Slingshot framework from the ground up. You'll have an application up and running within Slingshot about two minutes after you're done checking out the code.

Radiant CMS is a more complicated "real world" application that we ported to Slingshot. Our Radiant example port is a good place to go poke around when you're confused by something or want to get a more concrete idea of what is going on. This example app is Mac OS X Intel-only.

You can also sign up for the Slingshot mailing list.

Basic Architecture

Slingshot is divided into two major components:

1. Virtual machine: The Slingshot VM is a silo in which we provide a consistent binary environment both on Windows and Mac OS X so that you can deploy Rails code simultaneously to both operating systems. We want you to be able to focus on your Rails code; Slingshot will take care of the details of making a cross-platform client-side deployment.
2. Slingshot Plugin: slingshot_plugin is a Rails plugin that facilitates data server<->client data synchronization within Slingshot. You will need to add in code here that is specific to your application, in order to determine what data gets synced down to which client. It is also where you may modify or otherwise sanitize any data that an offline client is synchronizing upward.

Joyent

Home Connector Accelerator Developers Partners Support News & Events Weblog Login

Accelerator™ for Applications

Joyent's Accelerator provides on-demand computing, storage, applications and services to craft the exact solution you need for your application.

Accelerate your Development

» Overview

Technical Specs

On-demand Computing and Storage

- Typically putting about 10 servers (16GB) and 2x 4TBs of SAS in place every 2-4 days

Joyent

Today's schedule

130pm

230pm

240pm

340pm

350pm

QA

500pm

50

50

50

I

II

III

IV

V

VI

6 Acts

I. Introduction and foundational items

II. Where do I put stuff?

III. What stuff?

IV. What do I run on this stuff?

V. What are the patterns of deployment?

VI. Lessons learned

Joyent

What I’ll tell you about

- What we’ve done
- Why we’ve done it
- How we’re doing it
- Our way of thinking

Thinking

- Scalability, Throughput and Performance
- Limits (practical and theoretical)
- Rules of Ten
- Web applications are stateless

Scalability

- Scalability
- Throughput
- Performance

Rules of Ten

- Tiers
  - Tiers are different functionally
  - Tiers should be 10x different in throughput
- Costs
  - Infrastructure costs ≤10% of “revenue”

Web applications are stateless

- HTTP is a stateless protocol
- http://tools.ietf.org/html/rfc2616
  - The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems. It is a generic, stateless, protocol which can be used for many tasks beyond its use for hypertext, such as name servers and distributed object management systems, through extension of its request methods, error codes and headers [47]. A feature of HTTP is the typing and negotiation of data representation, allowing systems to be built independently of the data being transferred.

Because they're stateless

- Inherently "scalable"
- Front
  - How to scale when you have to connect it to an IP address?
- Application
- Back (Data store)
  - How to scale files, databases etc?

Simple
Standard
Open

Fundamental Limits

- Money
- Time
- People
- Experience
- Power (which limits memory and CPU)
- Bandwidth

Introduction and Foundational Items

I get asked lots of questions

- “I have yet to find any examples of websites that have heavy traffic and stream media that run from a Ruby on Rails platform, can you suggest any sites that will demonstrate that the ruby platform is stable and reliable enough to use on a commercial level?”

- “We are concerned about the long-term viability of Ruby on Rails as a development language and environment.”

- “How easily can a ruby site be converted to another language? (If for any reason we were forced to abandon ruby at some point in the future or I can’t find someone to work with our code?).”

- “My company has some concerns on whether or not Ruby on Rails is the right platform to deploy on if we have a very large scale app.”

- What is a “scalable” application?
- What are some hardware layouts?
- Where do you get the hardware?
- How do you pay for it?
- Where do you put it?
- Who runs it?
- How do you watch it?
- What do you need relative to an application?
- What are the commonalities of scalable web architectures?
- What are the unique bottlenecks for Ruby on Rails applications?
- What's the best way to start so you can make sure everything scales?
- What are the common mistakes?

But are these really Ruby or Rails specific ?

They have to do with designing and then running scalable "internet" applications

But the road to a top site on the internet is not from one interation

Let’s break that down

- Designing
- Running
- Scalable
- “Internet” applications

Scalable means?

Types of scalability

- Load
- Geographic
- Administrative

Joyent

A Sysadmin's view

- Ruby on Rails is simply one part
- Developers have to understand Rails horizontally (of course, otherwise they couldn't write the application)
- Developers ideally understand the vertical stack
- It can get complicated fast and it's easy to overengineer

What do you do with
1000s of physical machines?
100s of TB of storage?
In 4 facilities on 2 continents?

Is this a “Rails” issue?

No, I’m afraid not.

This has been done before.

The same big questions.

Let’s take the “connector”

“Logical” servers for the connector

1) Jumpstart/PXE Boot
2) Monitoring
3) Auditing
4) Logging
5) Provisioning and configuration management
6) DHCP/LDAP for server identification/authentication and control (at dual for failover)
7) DNS: DNS cache and resolver, and a (private) DNS system (4x + 2; 2+ sites)
8) DNS MySQL (4x + 2, dual masters with slaves per DNS node, innodb tables)
9) SPAM filtering servers (files to NFS store and tracking to postgresql)
10) SPAM database setup (postgresql)
11) SPAM NFS store
12) SMTP proxies and gateways out
13) SMTP proxies and gateways in (delivery to clusters to Maildir over NFS)
14) Mail stores
15) IMAP proxy servers
16) IMAP servers
17) User LDAP servers
18) User long running processes
19) User postgresql DB servers
20) User web servers
21) User application servers
22) User File Storage (NFS)
23) Joyent Organization Provisioning/Customer panel servers (web, app, database)
24) iSCSI storage systems
25) Chat servers
26) Load balancer/proxies/static caches
…

Guess which is “Rails”?

1) Jumpstart/PXE Boot &
2) Monitoring
3) Auditing
4) Logging
5) Provisioning and configuration management
6) DHCP/LDAP for server identification/authentication and control (dual for failover)
7) DNS: DNS cache and resolver, and a (private) DNS system (4x + 2)
8) DNS MySQL (x2, master/slave or dual master, innodb tables)
9) SPAM filtering servers (files to NFS store and tracking to postgresql)
10) SPAM database setup (postgresql)
11) SPAM NFS store (dual heads clustered)
12) SMTP proxy and gateway out
13) SMTP proxy and gateway in
14) Mail NFS store (dual heads)
15) IMAP proxy servers
16) IMAP servers
17) User LDAP servers
18) User long running processes
19) User postgresql DB servers
20) User web servers
21) User application servers
22) User File Storage (NFS)
23) Joyent Organization Provisioning/Customer panel servers (web, app, database)
24) iSCSI storage systems
25) Chat servers
26) Load balancer/proxies/static caches
...

Rails

Ease of management is on a log scale

- 10
- 100
- 1,000
- 10,000
- 100,000
- 1,000,000

↑computational power

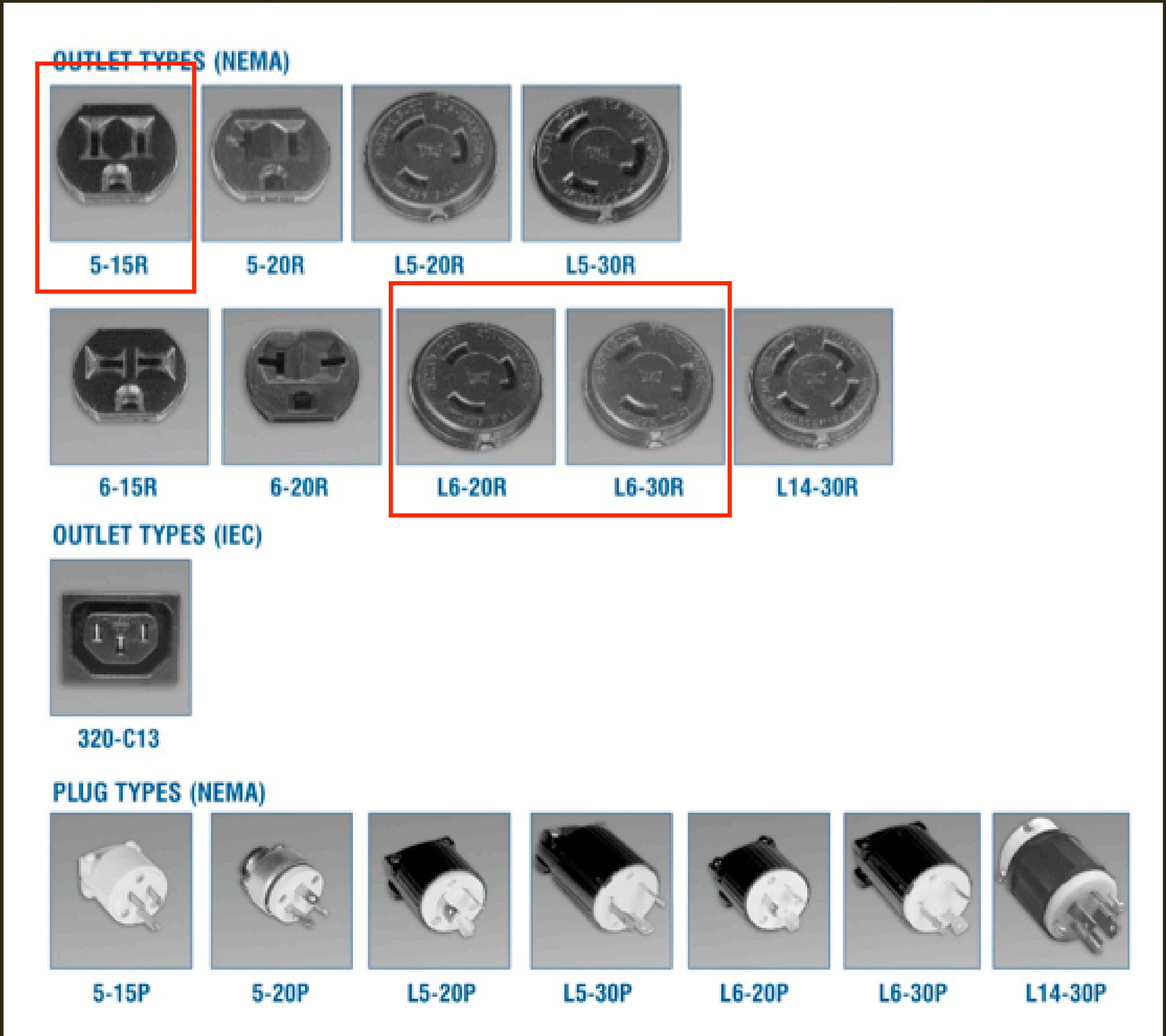
↓space

↓power

Power
A common limiting factor

Amps, Volts and Watts

- 110V, 208V, 240V
- 10, 15, 20, 30, 60 amp
- Standard baselines: 10 amp, 240V; 15 amp, 110V
- $25 per amp for 208V power
- 20 amps X 208V = 4160 watts
- 80% safely usable = 3328 watts



OUTLET TYPES (NEMA)

5-15R 5-20R L5-20R L5-30R

6-15R 6-20R L6-20R L6-30R L14-30R

OUTLET TYPES (IEC)

320-C13

PLUG TYPES (NEMA)

5-15P 5-20P L5-20P L5-30P L6-20P L6-30P L14-30P

A $5000 Dell 1850 costs $1850 to power over a 3 year lifespan

- 440watts x 24 hours/day x 1 kw/1000 watts = 10.56 kwh/day
- 10.56 kwh/day x $0.16/kwh = $1.69/day
- $1.69/day x 365 days/year = 616.85/year

Joyent

How many servers fit in a 100kw?

- 100 kilowatts to power and 100 kilowatts to cool
- At 250-400 watts each
- 250 - 400 servers

Other common limiting factors

Your Network

- The fast ethernet or gigabit network port (assuming there's more than one drive)
- Transactions of something/second
- 1 Gbps = 125 MB/sec (100 Mbps = 12.5 MB/sec)
- And let's say you can, just how much is a Gbps in some kind of other thing?

What’s a 100 Mbps in normal web traffic?

- Say a 122 KB page and has 20 objects (~125KB for an uncached page view).
- 125KB page => 100-1000 unique visitors per second
- 20 objects per page, that is 2000 requests per second that could pump out of that system.
- Maximum.

- What is the ability to do 2000 requests/second then?

- ((2000 requests/sec)*(20 requests per page)) * 0.125MB per page = 12.5 MB/sec (100Mbps constant).

- 86400 seconds/day on 100Mbps => 8,640,000 uniques in a day with 172,800,00 hits.

Joyent

How much does a 100 Mbps commit cost?

- Depends => One provider? Eight? Your own stuff?
- $5000-$8000/month for something “good”

Can you do 2000 requests/second?

- Sure an Apache, Lighttpd or Litespeed can do 1000-15,000 static or proxy requests/second
- A hardware load balancer like a Big-IP can do 20,000-100,000 fine.

Where do I put stuff?
What stuff?

Joyent

Physical considerations

- Space
- Power
- Network connection
- Cables cables cables
- Routers and switches
- Servers
- Storage

The 10% rule

- Google’s earning release:
- "Other cost of revenues, which is comprised primarily of data center operational expenses, as well as credit card processing charges, increased to $307 million, or 10% of revenues, in the fourth quarter of 2006, compared to $223 million, or 8% of revenues, in the third quarter."

The 10% rule

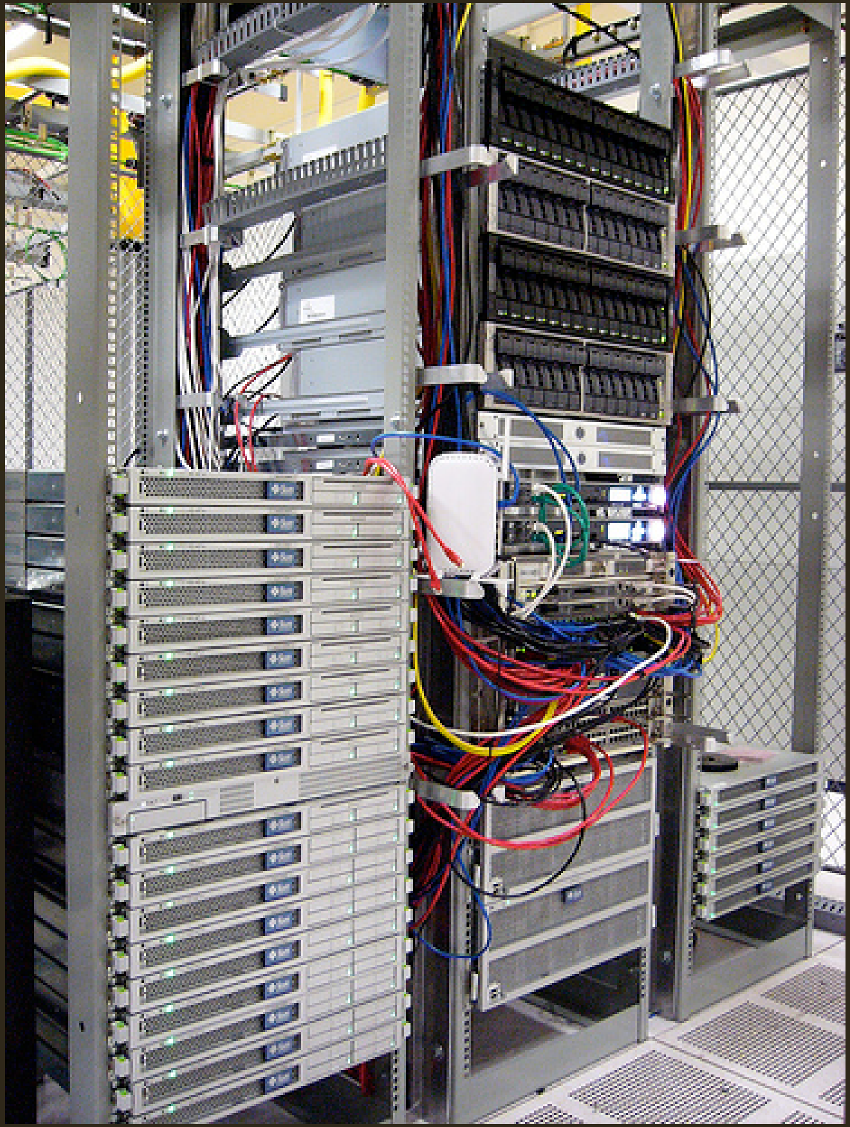
- A common rule of thumb I tell people is to target their performance goals in application design and coding so that their infrastructure (not including people) is ≤10% of an application’s revenue.

The 10% rule

- Meaning if you’re making $1.2 million dollars a year off of an online application, then you should be in area of spending $120,000/year or $10,000/month on servers, storage and bandwidth.
- And from the other way around, if you’re spending $10,000 a month on these same things, then you know where to push your revenue to.

Or maybe this is just a cost.
It used to be for me.

A joyent.net node (-ish)



Whatever you do

- Keep it simple
- Standardize, Standardize, Standardize
- Try and use open technologies

Some of my rules

- Virtualization, virtualization, virtualization
- Separating hardware components
- Keep the hardware setup simple
- Things should add up
- Configuration management and distributed control
- Pool and split
- Understand what each component can do as a maximum and a minimum

Pairing physical resources with logical needs while keeping the smallest footprint

You either build or you buy it

Or you buy all of it from someone else

Then You Build

“Buying”
(by the way)
means
“using Rails”
too

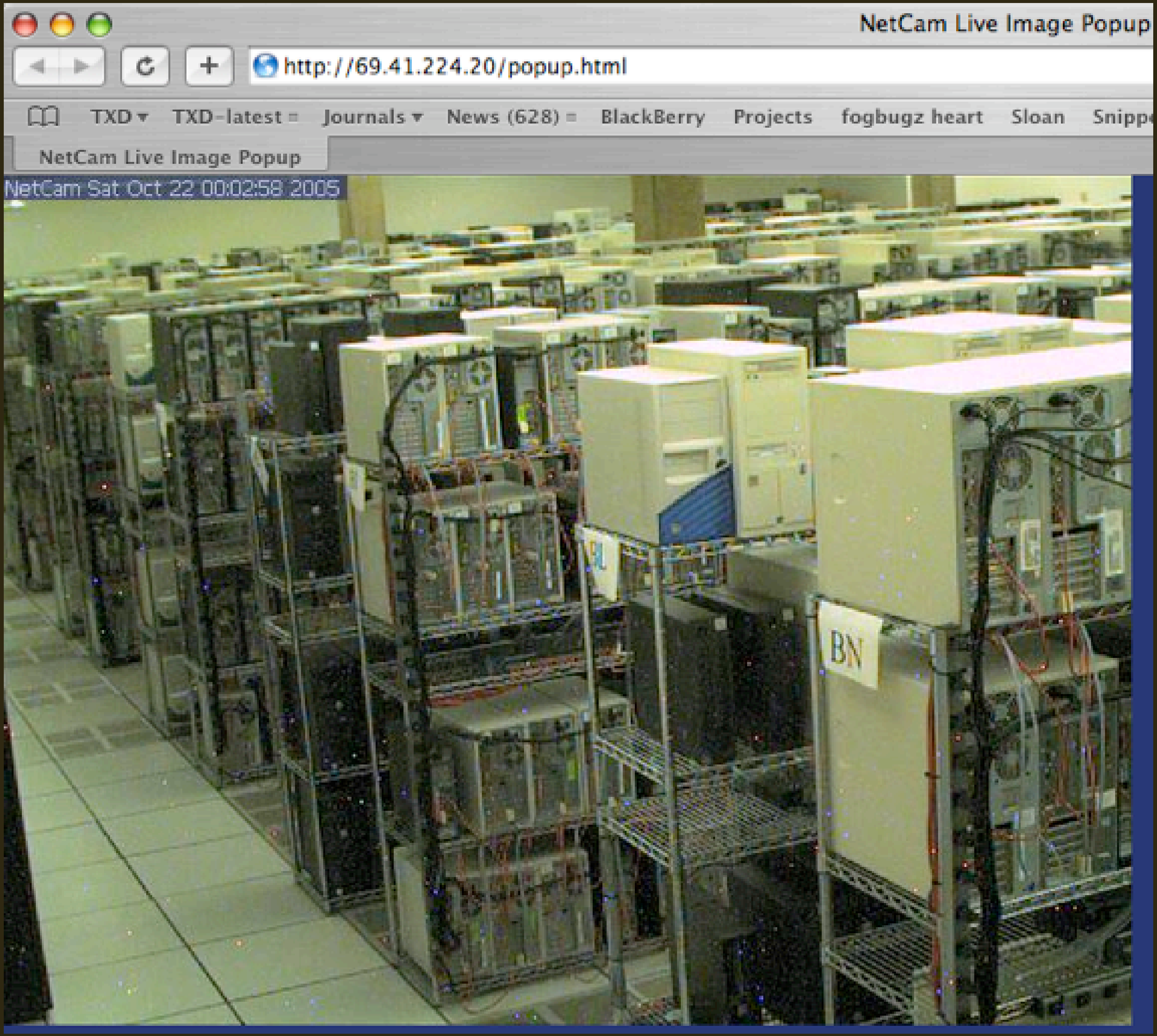
What's the cost breakpoints?

- Including people costs
- It's generally cheaper at the $20,000-$30,000/ month spending to do it in-house. *Assuming you or at least one of your guys knows what they're doing.

But what if I buy all my stuff from someone else?

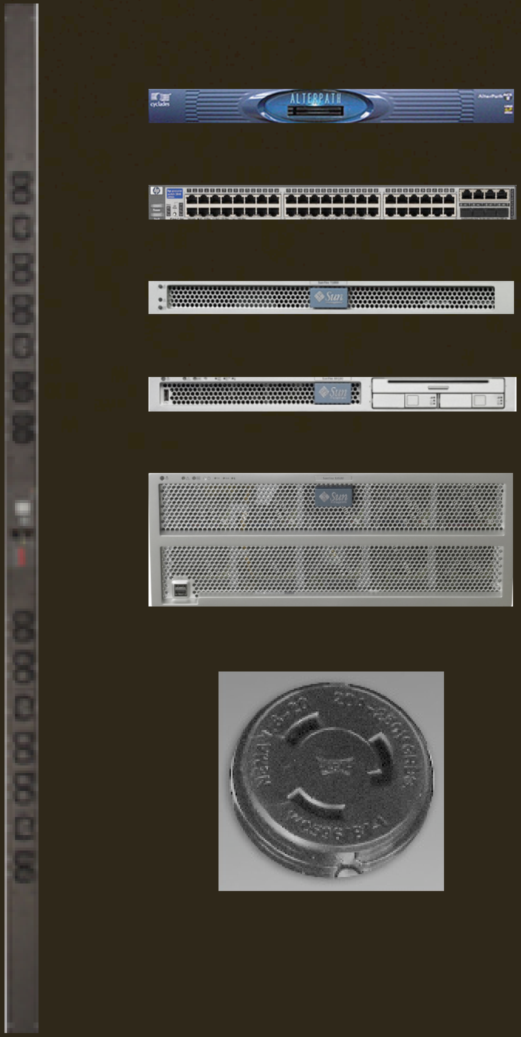
Our story

The Planet


NetCam Live Image Popup
http://69.41.224.20/popup.html
TXD
TXD-latest
Journals
News (628)
BlackBerry
Projects
fogbugz heart
Sloan
Snipp
NetCam Live Image Popup
NetCam Sat Oct 22 00:02:58 2005
BN

Cee-Kay

KISS



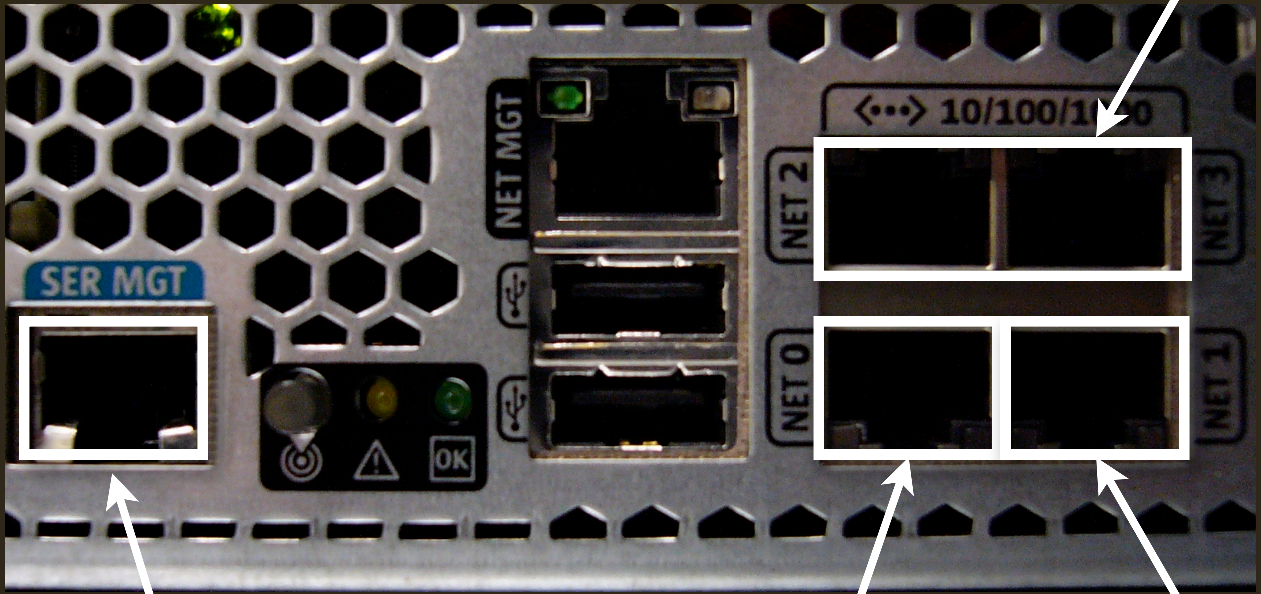
One kind of
Console server
Switch
Server
CPU
RAM
Storage
Disc
Operating system
Interconnect
Power plug
Power strip

Our Choices

- Console server => Lantronix
- Switch => 48 port all gigabit (Force10 E300s)
- Server => Sun Fire AMDs (X4100,X4600), T1000s
- CPU => Opteron 285 and T1 SPARC
- RAM => 2GB DIMMS
- Storage => Sun Fire X4500 and NetApp FAS filters
- Disc => 500GB SATA and 73GB/146GB SAS
- Operating system => Solaris Nevada ("11")
- Interconnect => gigabit with cat6 cables
- Power plug => 208V, L6-20R (the "wall"), IEC320-C14 to IEC320-C13 (server to PDU)
- Power strip => APC 208V, 20x

I/O

- Flat network
- Physically segregated



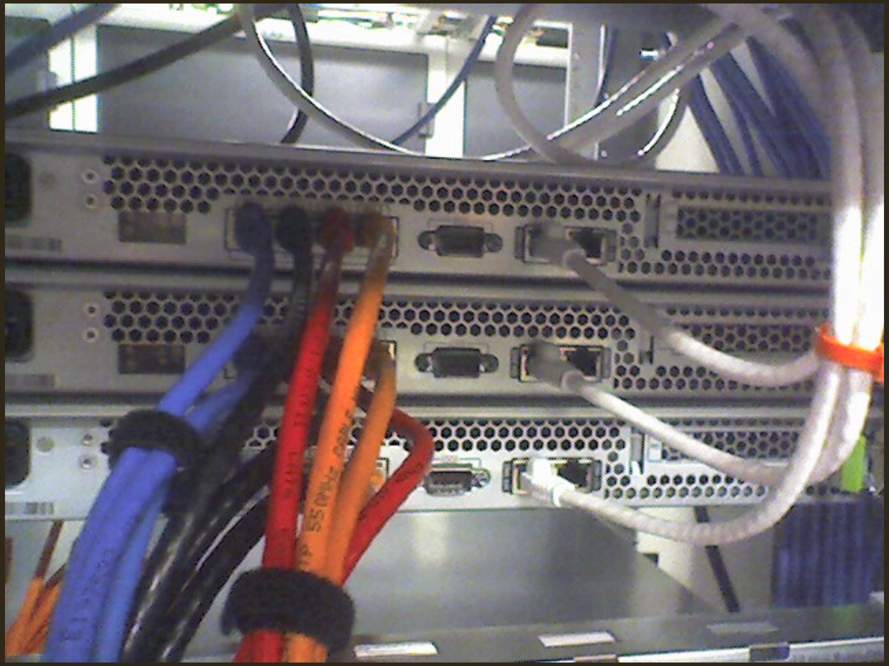
Storage

Remote

Public

Interconnects

Cabling standard



3’ cat6

Public network

Private interconnects

Storage

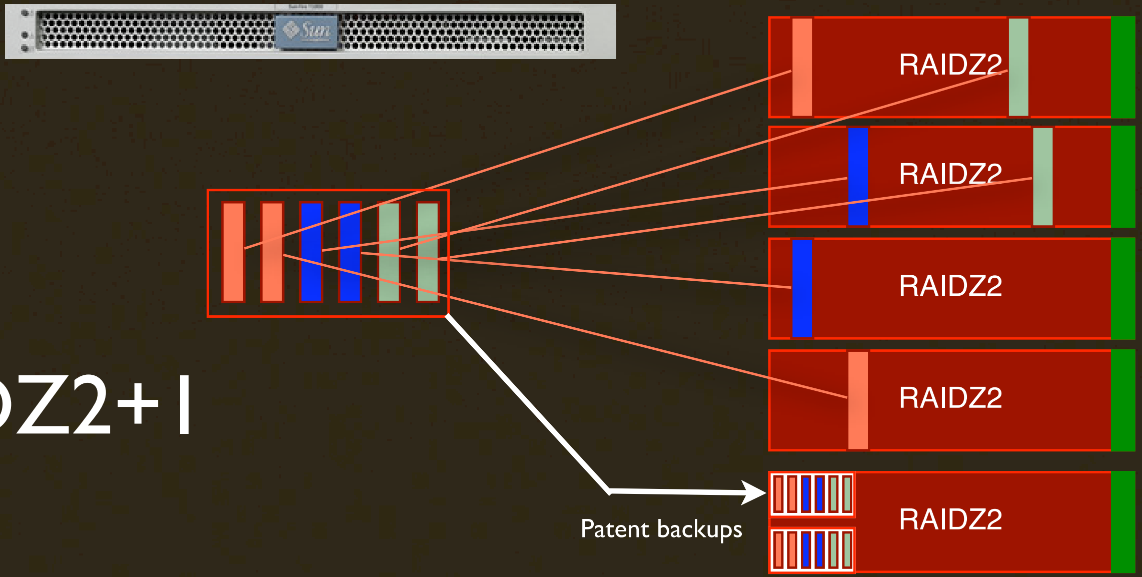
Storage

ALOM => console servers

Switch interconnects and mesh

Joyent

Our "SAN"

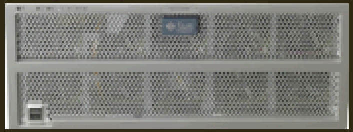


RAIDZ2+1

- Think of it as a distributed RAID6+1
- Dual switched
- Able to turn off half of your storage units
- You end up having data stripped across 44, 88, 132, 196 drives

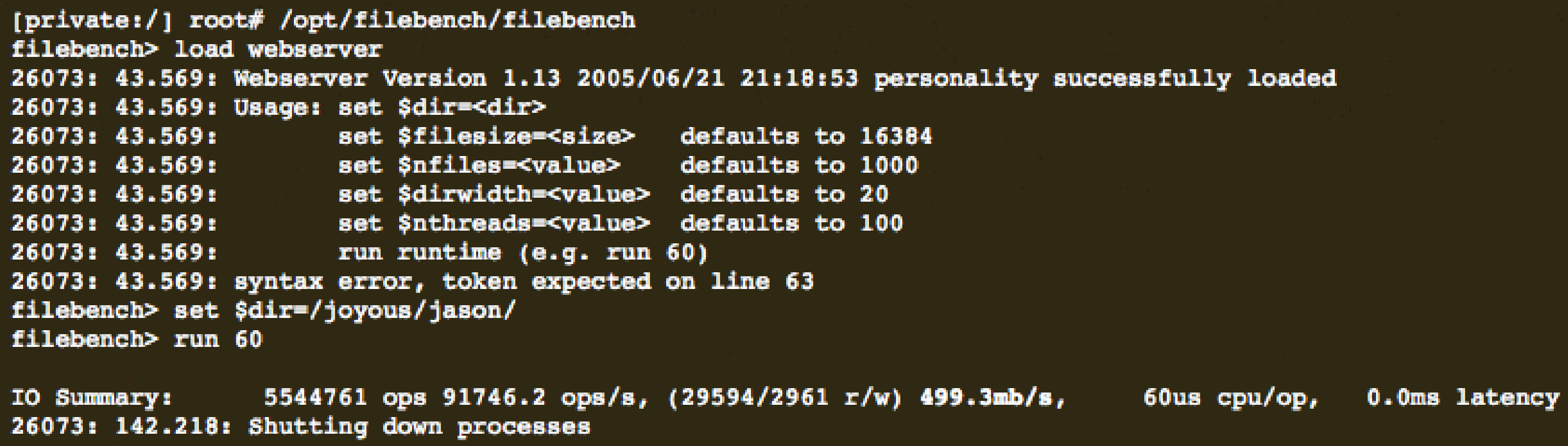
Our storage buckets

- 48 drive X4500



- Some do operations and storage on box
- Some export storage out to servers
- Spreading across this many drives is fast

How fast?



```
[private:/] root# /opt/filebench/filebench
filebench> load webserver
26073: 43.569: Webserver Version 1.13 2005/06/21 21:18:53 personality successfully loaded
26073: 43.569: Usage: set $dir=<dir>
26073: 43.569:        set $filesize=<size>   defaults to 16384
26073: 43.569:        set $nfiles=<value>    defaults to 1000
26073: 43.569:        set $dirwidth=<value>  defaults to 20
26073: 43.569:        set $nthreads=<value>  defaults to 100
26073: 43.569:        run runtime (e.g. run 60)
26073: 43.569: syntax error, token expected on line 63
filebench> set $dir=/joyous/jason/
filebench> run 60

IO Summary:      5544761 ops 91746.2 ops/s, (29594/2961 r/w) 499.3mb/s,     60us cpu/op,   0.0ms latency
26073: 142.218: Shutting down processes
```

- A simple “web server” benchmark
- Nearly 100,000 operations/sec
- 500 MB/sec is 4 Gbps (gigabits per second)

Vendors

- Networking: Dell, HP, Force10, Cisco, Foundry
- Servers: Dell, HP, Sun
- Storage: Dell, HP, Sun, NetApp, Nexsan

Joyent

Servers

- Dell: 1850 and 2850 models
- HP: DL320s
- Sun: X2100 and X4100

Storage?

- Lots of local drives
- DAS trays (trays that do their own RAID)
- iSCSI (it’s now possible to stay away from fiber)
- RAID6 and RAID10

Leverage

- **Pick a vendor** and get as much as you can from that single vendor

Some comments

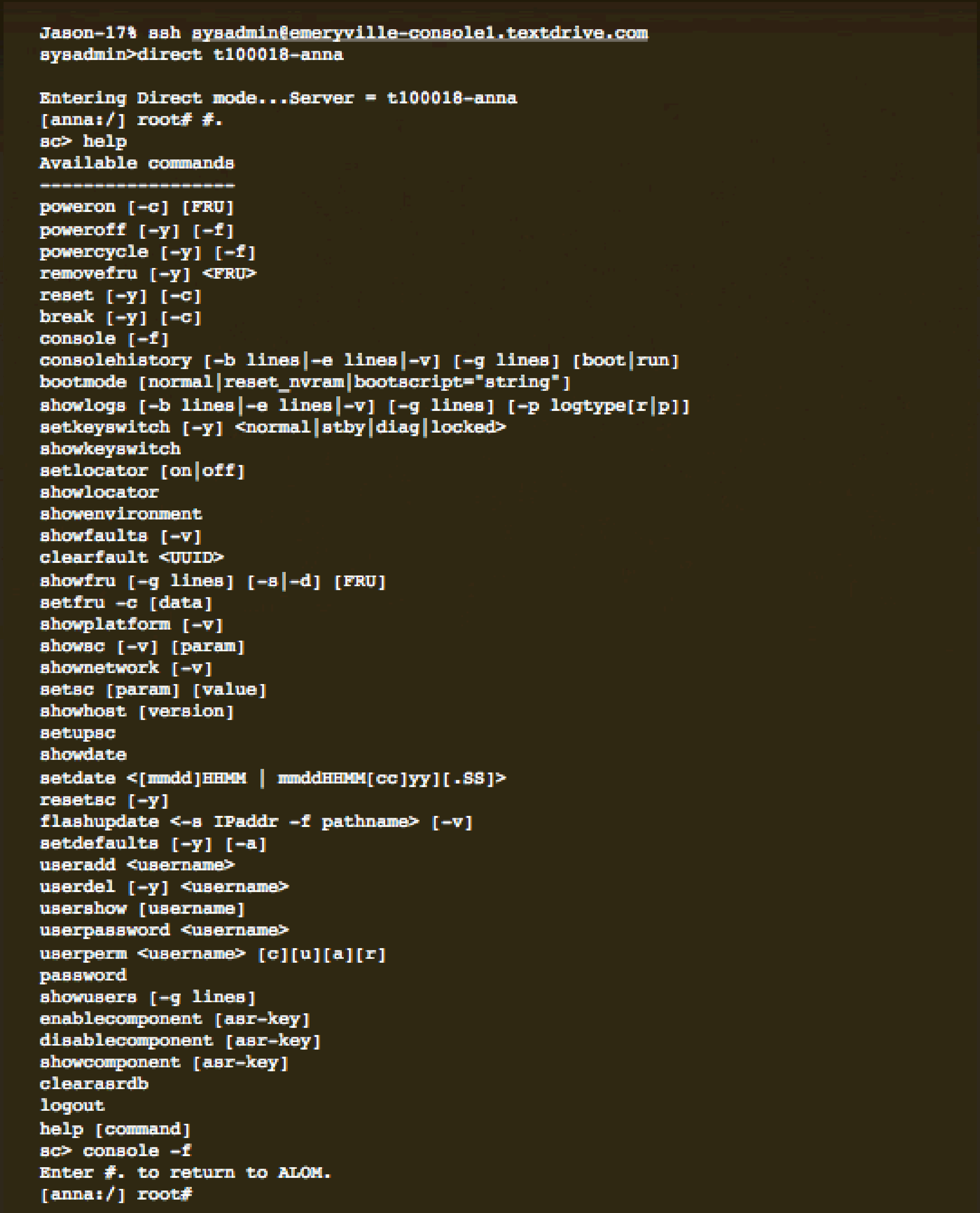
- Dell => direct, aggressive, helpful and they resell a lot of stuff
- HP => might be direct, likely reseller
- Sun => you go through a reseller

Why we started with Dells

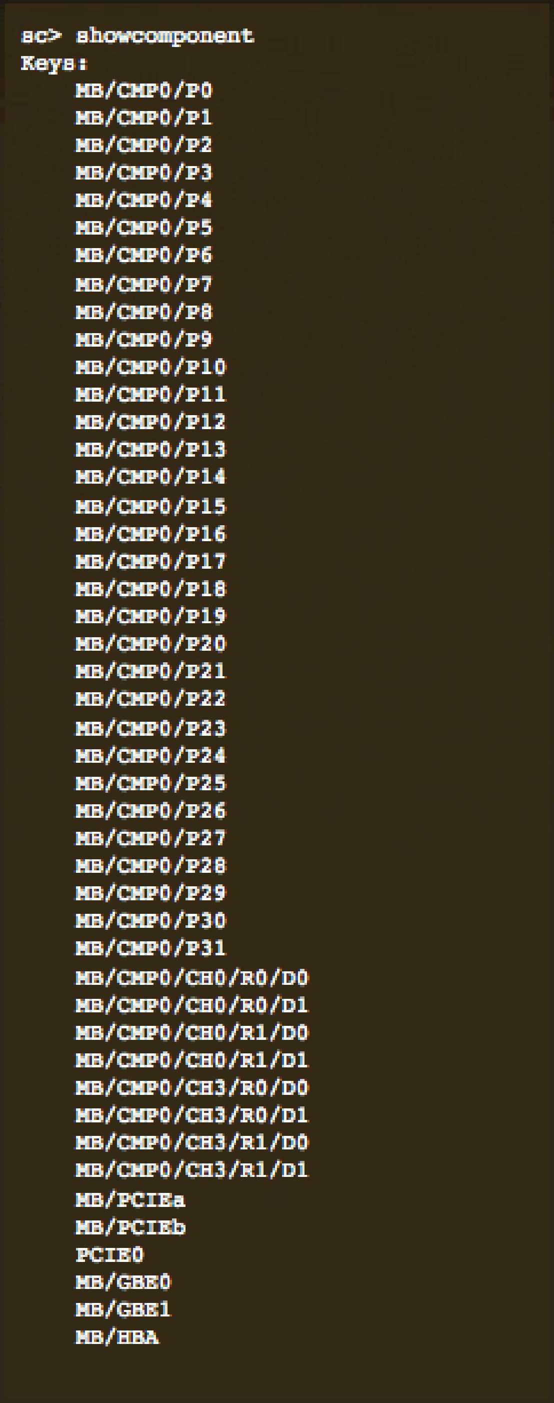
- Responsive
- They put us in touch with different leasing companies and arrangements
- They shipped
- We were a Dell/EMC shop (even with Solaris running on them)

Why we ended up using Sun

- The rails (literally the rack rails)
- RAS
- Hot-swappable components
- Energy efficient
- True ALOM/iLOM that works with console
- Often cheapest per CPU, per GB RAM
- Often cheapest in TCO
- We’re on Solaris (there’s some assurances there)



```
Jason-17% ssh sysadmin@emeryville-console1.textdrive.com
sysadmin>direct t100018-anna

Entering Direct mode...Server = t100018-anna
[anna:/] root# #.
sc> help
Available commands
------------------
poweron [-c] [FRU]
poweroff [-y] [-f]
powercycle [-y] [-f]
removefru [-y] <FRU>
reset [-y] [-c]
break [-y] [-c]
console [-f]
consolehistory [-b lines|-e lines|-v] [-g lines] [boot|run]
bootmode [normal|reset_nvram|bootscript="string"]
showlogs [-b lines|-e lines|-v] [-g lines] [-p logtype[r|p]]
setkeyswitch [-y] <normal|stby|diag|locked>
showkeyswitch
setlocator [on|off]
showlocator
showenvironment
showfaults [-v]
clearfault <UUID>
showfru [-g lines] [-s|-d] [FRU]
setfru -c [data]
showplatform [-v]
showsc [-v] [param]
shownetwork [-v]
setsc [param] [value]
showhost [version]
setupsc
showdate
setdate <[mmdd]HHMM | mmddHHMM[cc]yy][.SS]>
resetsc [-y]
flashupdate <-s IPaddr -f pathname> [-v]
setdefaults [-y] [-a]
useradd <username>
userdel [-y] <username>
usershow [username]
userpassword <username>
userperm <username> [c][u][a][r]
password
showusers [-g lines]
enablecomponent [asr-key]
disablecomponent [asr-key]
showcomponent [asr-key]
clearasrdb
logout
help [command]
sc> console -f
Enter #. to return to ALOM.
[anna:/] root#
```



```
sc> showcomponent
Keys:
        MB/CMP0/P0
        MB/CMP0/P1
        MB/CMP0/P2
        MB/CMP0/P3
        MB/CMP0/P4
        MB/CMP0/P5
        MB/CMP0/P6
        MB/CMP0/P7
        MB/CMP0/P8
        MB/CMP0/P9
        MB/CMP0/P10
        MB/CMP0/P11
        MB/CMP0/P12
        MB/CMP0/P13
        MB/CMP0/P14
        MB/CMP0/P15
        MB/CMP0/P16
        MB/CMP0/P17
        MB/CMP0/P18
        MB/CMP0/P19
        MB/CMP0/P20
        MB/CMP0/P21
        MB/CMP0/P22
        MB/CMP0/P23
        MB/CMP0/P24
        MB/CMP0/P25
        MB/CMP0/P26
        MB/CMP0/P27
        MB/CMP0/P28
        MB/CMP0/P29
        MB/CMP0/P30
        MB/CMP0/P31
        MB/CMP0/CH0/R0/D0
        MB/CMP0/CH0/R0/D1
        MB/CMP0/CH0/R1/D0
        MB/CMP0/CH0/R1/D1
        MB/CMP0/CH3/R0/D0
        MB/CMP0/CH3/R0/D1
        MB/CMP0/CH3/R1/D0
        MB/CMP0/CH3/R1/D1
        MB/PCIEa
        MB/PCIEb
        PCIE0
        MB/GBE0
        MB/GBE1
        MB/HBA
```

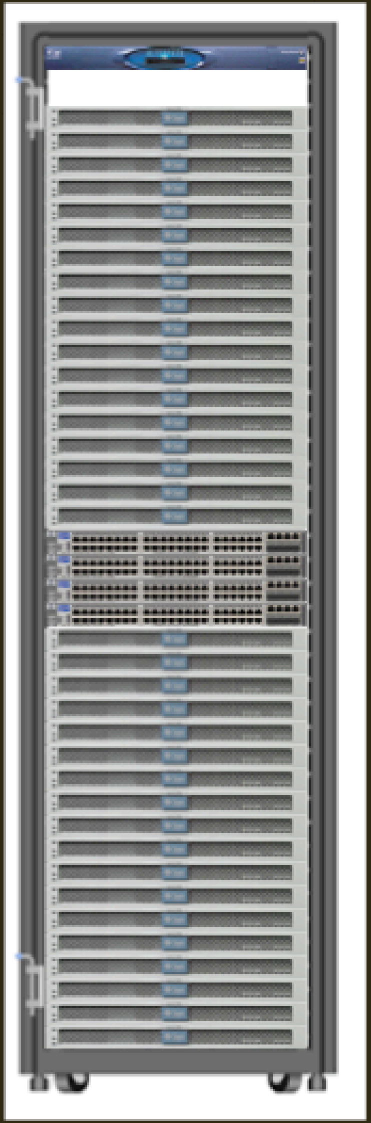
Lease if you can

- Generally it’s about 10-50% down
- And can be “ok” interest rate wise: 8-18%
- Do FMV where you turn over the systems at year 2
- How do you do it? Demonstrate that you have the cash overwise and push your vendor.

So what's a typical lease payment?

- $10,000 system
- $1000 down, $400/month (10% down, 4%/mo)

Designing around power



1152 logical 1Ghz CPUs
576 GB RAM
6,480 watts

31 amps @ 208V

2 x 20amp @ 208V L6-20R
2 x 24 plug 20amp/208V

----

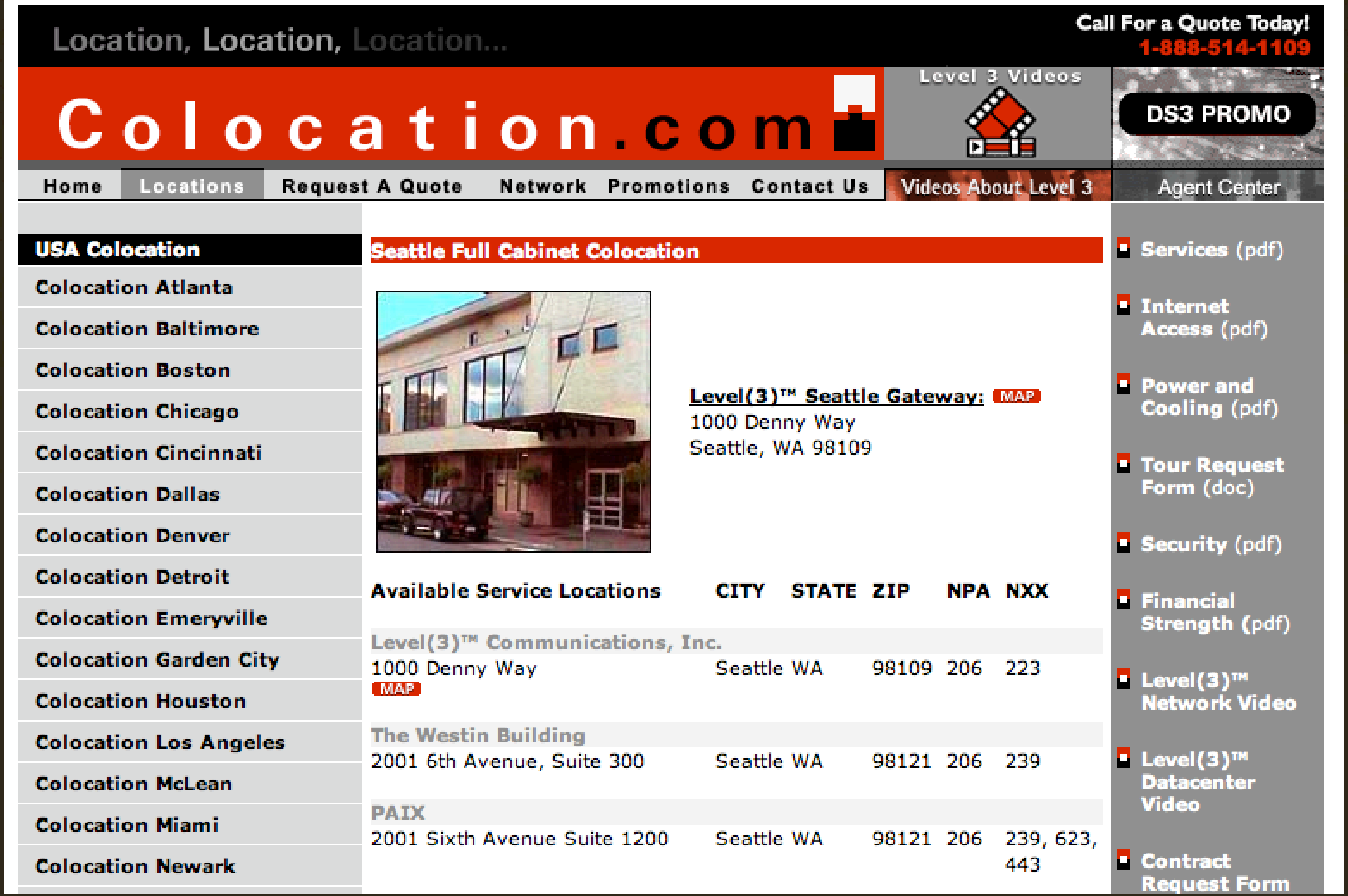
12 Enclosures
36 controllers
180 drives
90 TBs raw storage
77 TBs clustered storage

29.4 amps @ 208V

2 x 20amp @ 208V L6-20R
2 x 24 plug 20amp/208V

Joyent

So you want to colo?



Location, Location, Location...

Call For a Quote Today!
1-888-514-1109

USA Colocation

- Colocation Atlanta
- Colocation Baltimore
- Colocation Boston
- Colocation Chicago
- Colocation Cincinnati
- Colocation Dallas
- Colocation Denver
- Colocation Detroit
- Colocation Emeryville
- Colocation Garden City
- Colocation Houston
- Colocation Los Angeles
- Colocation McLean
- Colocation Miami
- Colocation Newark

Seattle Full Cabinet Colocation

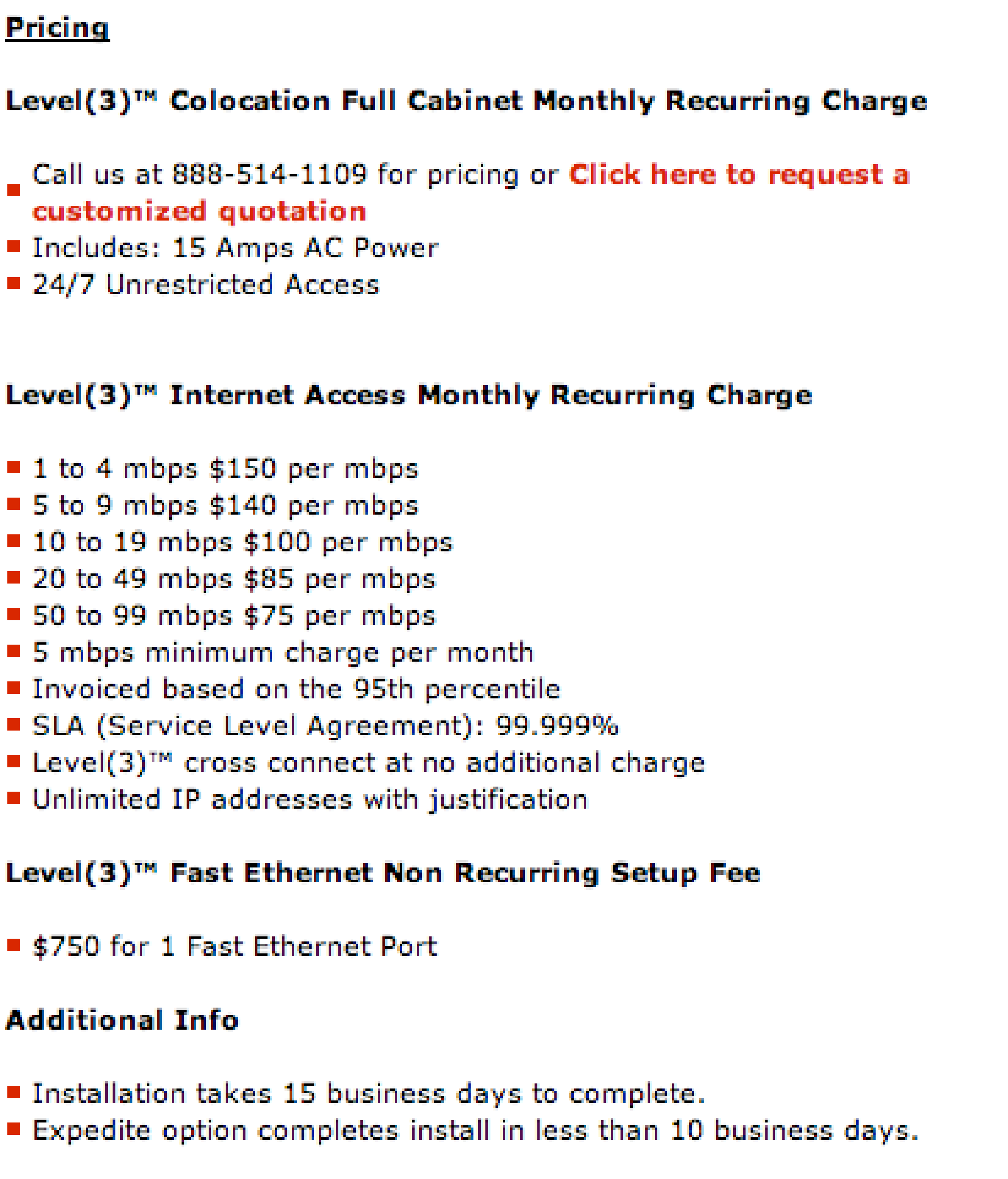
Level(3)™ Seattle Gateway: MAP
1000 Denny Way
Seattle, WA 98109

Available Service Locations	CITY	STATE	ZIP	NPA	NXX
Level(3)™ Communications, Inc.					
1000 Denny Way MAP	Seattle	WA	98109	206	223
The Westin Building					
2001 6th Avenue, Suite 300	Seattle	WA	98121	206	239
PAIX					
2001 Sixth Avenue Suite 1200	Seattle	WA	98121	206	239, 623, 443

- Services (pdf)
- Internet Access (pdf)
- Power and Cooling (pdf)
- Tour Request Form (doc)
- Security (pdf)
- Financial Strength (pdf)
- Level(3)™ Network Video
- Level(3)™ Datacenter Video
- Contract Request Form

You’re going to be small potatoes

Try and go local to start



Pricing

Level(3)™ Colocation Full Cabinet Monthly Recurring Charge

- Call us at 888-514-1109 for pricing or **Click here to request a customized quotation**
- Includes: 15 Amps AC Power
- 24/7 Unrestricted Access

Level(3)™ Internet Access Monthly Recurring Charge

- 1 to 4 mbps $150 per mbps
- 5 to 9 mbps $140 per mbps
- 10 to 19 mbps $100 per mbps
- 20 to 49 mbps $85 per mbps
- 50 to 99 mbps $75 per mbps
- 5 mbps minimum charge per month
- Invoiced based on the 95th percentile
- SLA (Service Level Agreement): 99.999%
- Level(3)™ cross connect at no additional charge
- Unlimited IP addresses with justification

Level(3)™ Fast Ethernet Non Recurring Setup Fee

- $750 for 1 Fast Ethernet Port

Additional Info

- Installation takes 15 business days to complete.
- Expedite option completes install in less than 10 business days.

Our providers

- Level 3
- Equinix (IXEurope)

- Why?

Typical power they'll allow

- Dual 15 amp, 110V
- Dual 20 amp, 110V
- Dual 20 amp, 208V (rare in non-cage setups)
- $250/month for each 15/20 amp, 110V plug
- $500/month for 20amp, 208V plug

Typical Costs

- $500-750 for the rack
- $500-$1000 for power per rack
- $1000 bandwidth commit (10 Mbps x $100, applies to all racks)
- $4000 for systems (20x $200/mo)/rack

Comparison

- Total $6500 for 20 systems in a rack on a lease.
- “2850s” at The Planet or Rackspace: $900-1200 each ($18,000 - $24,000/month)
- DIY: does require a more involved human or two of them (that could use up the difference; a great sysadmin/racker is $100K+)

What do you run on it?

You have to use Virtualization

- Xen
- VMware
- Solaris zones

- **Seriously.** You have to pick and use a virtualization method.

The common issue with some virtualization methods

- Do you get linear performance?
- Are four “virtual servers” on a 4 CPU, 16GB of RAM server = four separate 1CPU, 4GB of RAM servers?
- Are the differences significant enough when balanced with differences in power and space?

Joyent

Why Solaris?

- Great PXE and remote boot support
- It doesn’t crash when stressed
- When it does crash it reboots itself, restarts everything and leaves a *complete* crash dump
- Proven on large systems => 512 GB of RAM with 1.0 TB of swap
- Observability tools are unmatched => process level and DTrace
- Fault management
- SMF
- ZFS filesystem => snapshots, zfs send -r | zfs receive -r, large filesystem support, compression
- Network aggregation and virtualization
- Lightweight virtualization
- FREE

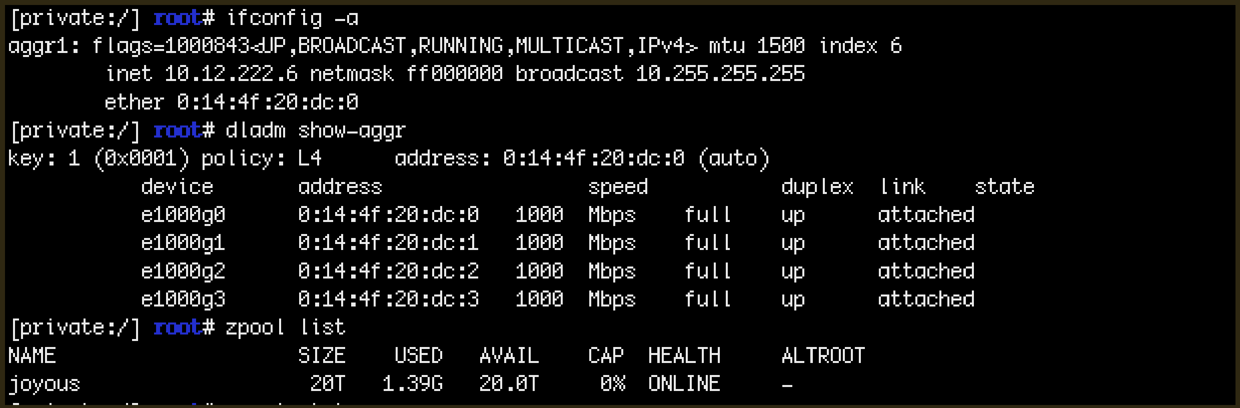
ZFS gives storage back

- Very easy CLI
- Same OS on both sides (servers and storage)
- NFS and iSCSI exports
- Compression
- Remote replication (real time; “slave” anything)
- Fully transactional
- Thin provisioning
- Dynamic striping
- Pool expansion
- Double parity RAID
- Hot spares

It’s easy to create a pool

```
# zpool create joyous \
        raidz2 c{5,4,7,6,1,0}t4d0 c{4,7,6,1,0}t0d0 \ {can lose 2}
        raidz2 c{5,4,7,6,1,0}t5d0 c{4,7,6,1,0}t1d0 \ {can lose 2}
        raidz2 c{5,4,7,6,1,0}t6d0 c{4,7,6,1,0}t2d0 \ {can lose 2}
        raidz2 c{5,4,7,6,1,0}t7d0 c{4,7,6,1,0}t3d0 \ {can lose 2}
        spare c5t{1,2,3}d0                                        {three hot spares}
```

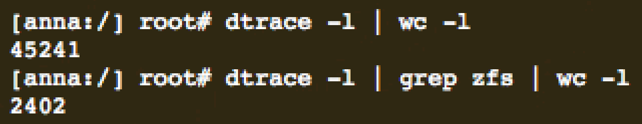
And to have a trunked network



```
[private:/] root# ifconfig -a
aggr1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 6
        inet 10.12.222.6 netmask ff000000 broadcast 10.255.255.255
        ether 0:14:4f:20:dc:0
[private:/] root# dladm show-aggr
key: 1 (0x0001) policy: L4      address: 0:14:4f:20:dc:0 (auto)
           device       address                 speed           duplex  link    state
           e1000g0      0:14:4f:20:dc:0   1000  Mbps    full    up      attached
           e1000g1      0:14:4f:20:dc:1   1000  Mbps    full    up      attached
           e1000g2      0:14:4f:20:dc:2   1000  Mbps    full    up      attached
           e1000g3      0:14:4f:20:dc:3   1000  Mbps    full    up      attached
[private:/] root# zpool list
NAME                    SIZE    USED   AVAIL    CAP  HEALTH     ALTROOT
joyous                   20T   1.39G   20.0T     0%  ONLINE     -
```

DTrace

- Subsystem that cuts throughput everything (via “probes”)



```
[anna:/] root# dtrace -l | wc -l
45241
[anna:/] root# dtrace -l | grep zfs | wc -l
2402
```

- Can reproduce anything like “vmstat” in D
- Probes in apache, ruby (we’re releasing soon), postgresql ... for a full drilldown

Zones/Containers

- Lightweight
- CLI/programmable interface
- Attach/detach
- Zone + Resource Controls = Container

```
zone1>create -b

global-zone # zonecfg -z 1container

zonecfg:1container>set pool=ZONE_NAME

zonecfg:1container>set zonepath=ZONE_PATH

zonecfg:1container>set autoboot=true

zonecfg:1container>add inherit-pkg-dir

zonecfg:1container:add>set dir=/lib

zonecfg:1container:add>end

zonecfg:1container>add net

zonecfg:1container:add>set address=ZONE_IPADDRESS

zonecfg:1container:add>set physical=bge0

zonecfg:1container:add>end

zonecfg:1container:rctl> set name=zone.cpu-shares

zonecfg:1container:rctl> add value (priv=privileged,limit=512,action=none)

zonecfg:1container:rctl> end

zonecfg:1container>commit
```

Joyent

The Key

What are the patterns of deployment?
Lessons learned

Ruby

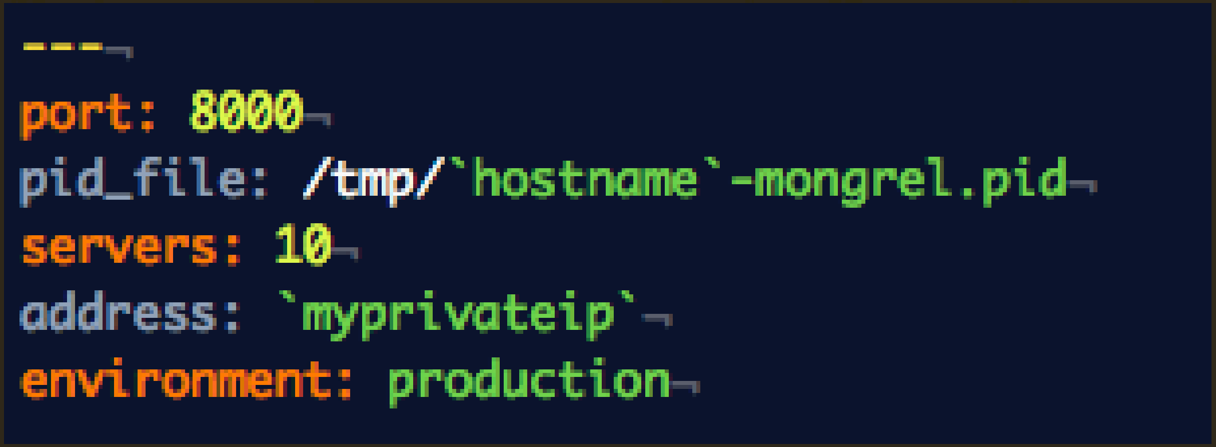
- I like that Ruby is process-based
- I actually don’t think it should ever be threaded
- I think it should focus on being as asynchronous and event-based on a per process basis
- I think it should be loosely coupled
- What does a “VM” do then: it manages LWPs
- This is erlang versus java

So how do you run a rails process?

- FCGI
- Mongrel (event-driven)
- JRuby in Glassfish

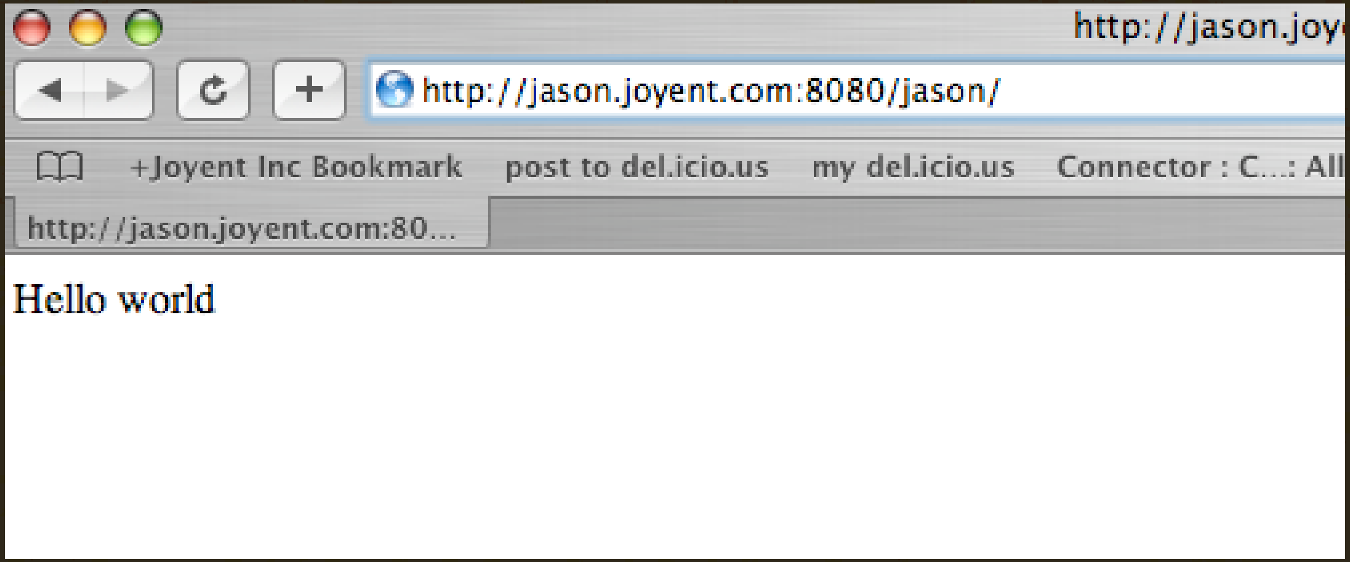
How we do Mongrel

- 16GB RAM, 4 AMD CPU machines
- 4 virtual “containers” on them
- Each container: 10 mongrels (so 10 per CPU)



```
---¬
port: 8000¬
pid_file: /tmp/`hostname`-mongrel.pid¬
servers: 10¬
address: `myprivateip`¬
environment: production¬
```

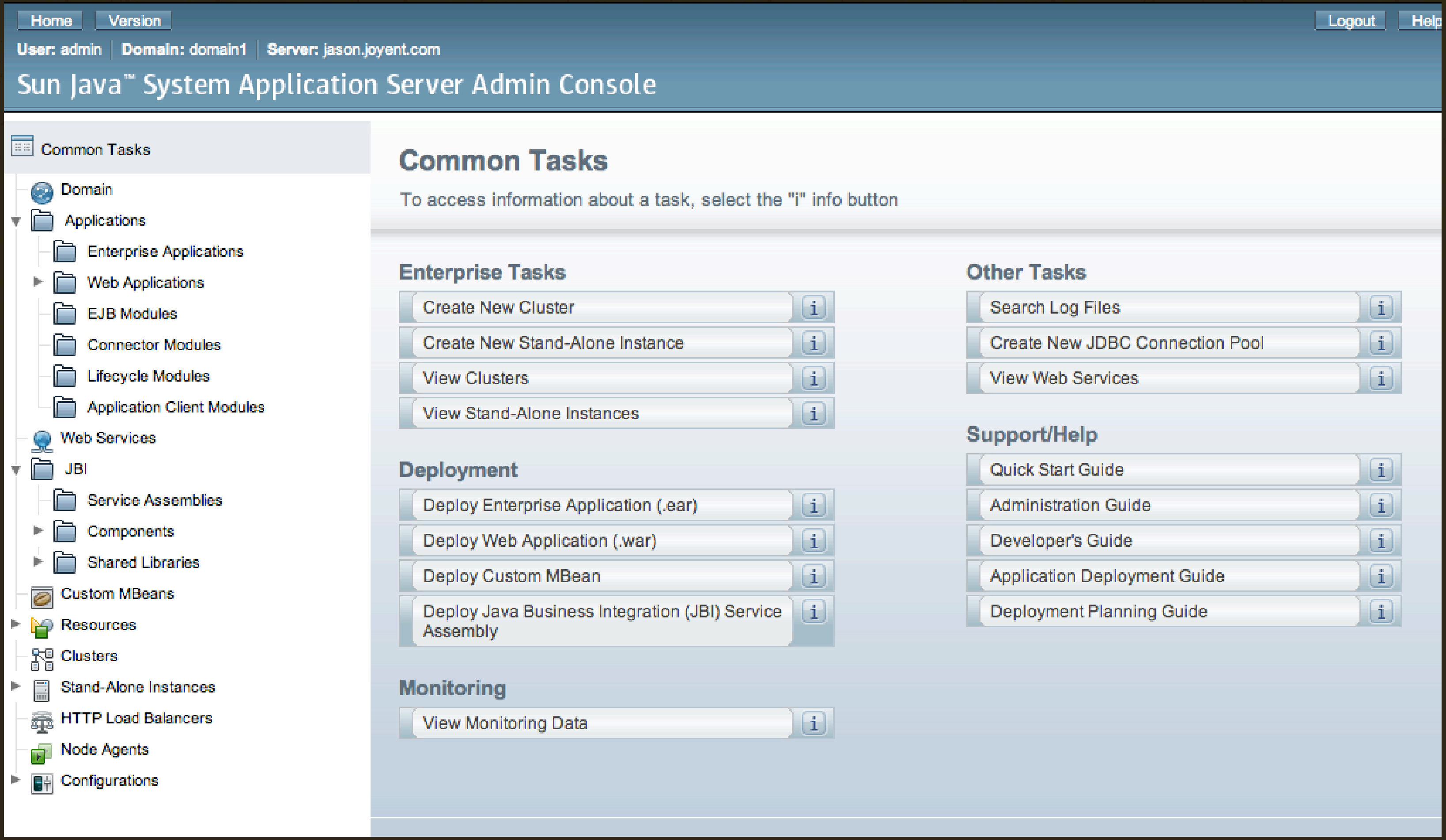
Glassfish



Hello world

```
$ rake war:standalone:create

(in /home/jason)

Assembling web application

        Adding web application

        Adding Ruby gem ActiveRecord-JDBC version 0.4

    Creating web archive

$ cp railstest.war /a/1/app/glassfish/domains/domain1/autodeploy/
railstest.war
```

Glassfish admin



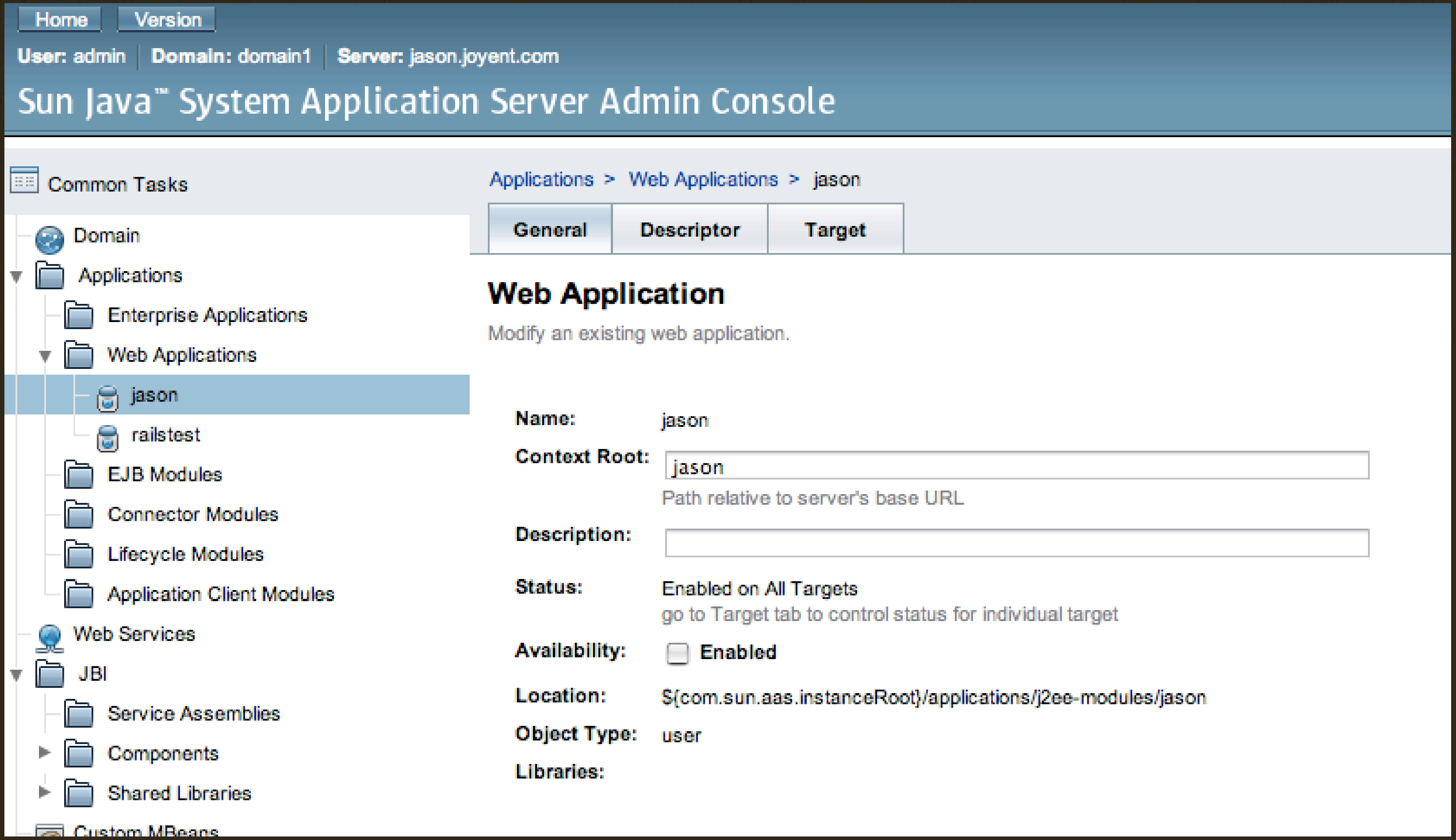
User: admin | Domain: domain1 | Server: jason.joyent.com

Sun Java™ System Application Server Admin Console

Common Tasks

To access information about a task, select the "i" info button

A controller or an application?



User: admin | Domain: domain1 | Server: jason.joyent.com

Sun Java™ System Application Server Admin Console

Web Application

Modify an existing web application.

Name:	jason
Context Root:	jason Path relative to server's base URL
Description:	
Status:	Enabled on All Targets go to Target tab to control status for individual target
Availability:	□ Enabled
Location:	${com.sun.aas.instanceRoot}/applications/j2ee-modules/jason
Object Type:	user
Libraries:	

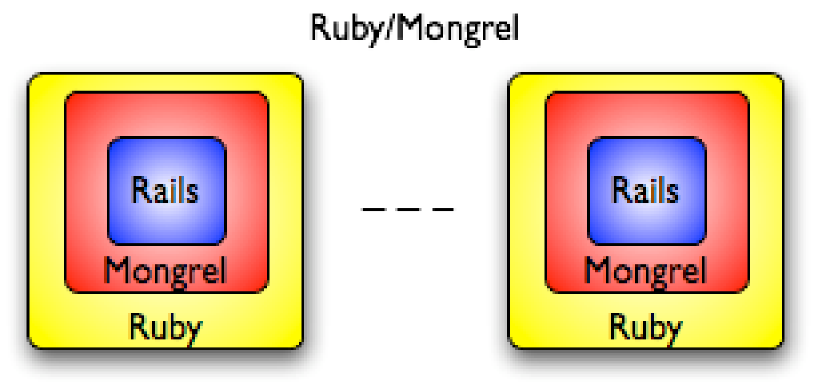
Quick note on relative performance

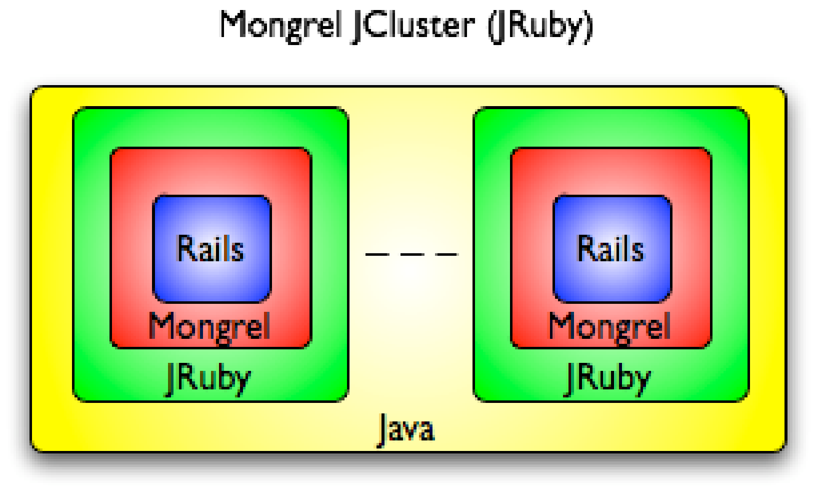
- A “hello world” application

- Mongrel: 150 req/sec
- Event mongrel: 400 req/sec
- Glassfish: 1000 req/sec

Base difference

- http://weblogs.java.net/blog/arungupta/archive/2007/08/rubyjruby_proce.html





Affects “administrative” scalability
Affects concept of process scalability

How do you scale processes?

- Run more and more of them
- They should add up

Add up how and where?

- Add up in the front
- Add up in the back
- Add up linearly

Horizontally scaling across processes

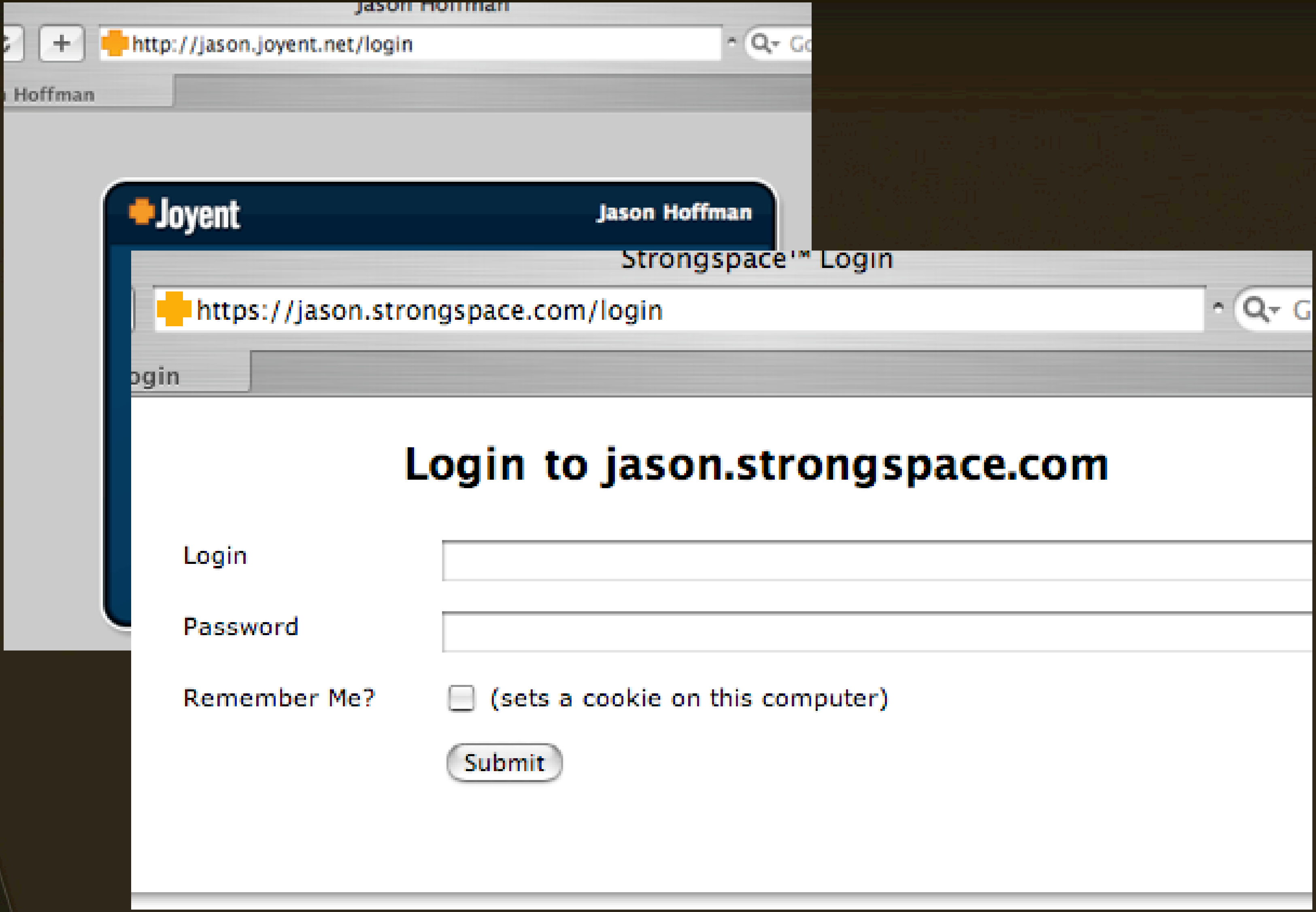
- In the front: load balancers capable of it
- In the back: database middleware and message buses

The front

- DNS
- Load-balancing
- The “front” cannot be built into the application

DNS

- Don’t forget about it.
- Always surprising how little people know about DNS servers
- Federation by DNS is an easy way to split your customers into pods.



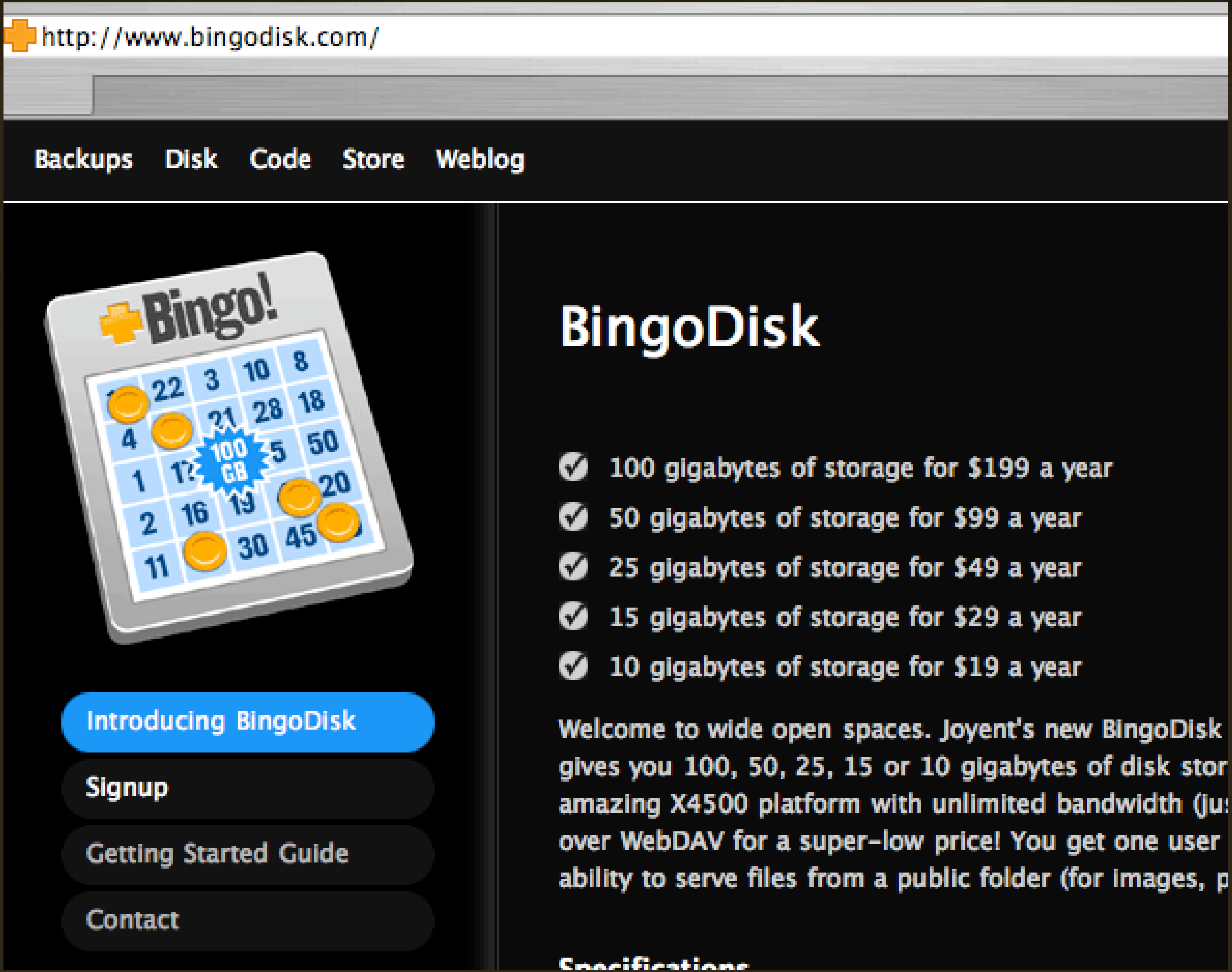
Jason Hoffman
http://jason.joyent.net/login
Hoffman
Joyent
Jason Hoffman
Strongspace™ Login
https://jason.strongspace.com/login
ogin

Login to jason.strongspace.com

Login

Password

Remember Me? □ (sets a cookie on this computer)



http://www.bingodisk.com/

Backups Disk Code Store Weblog

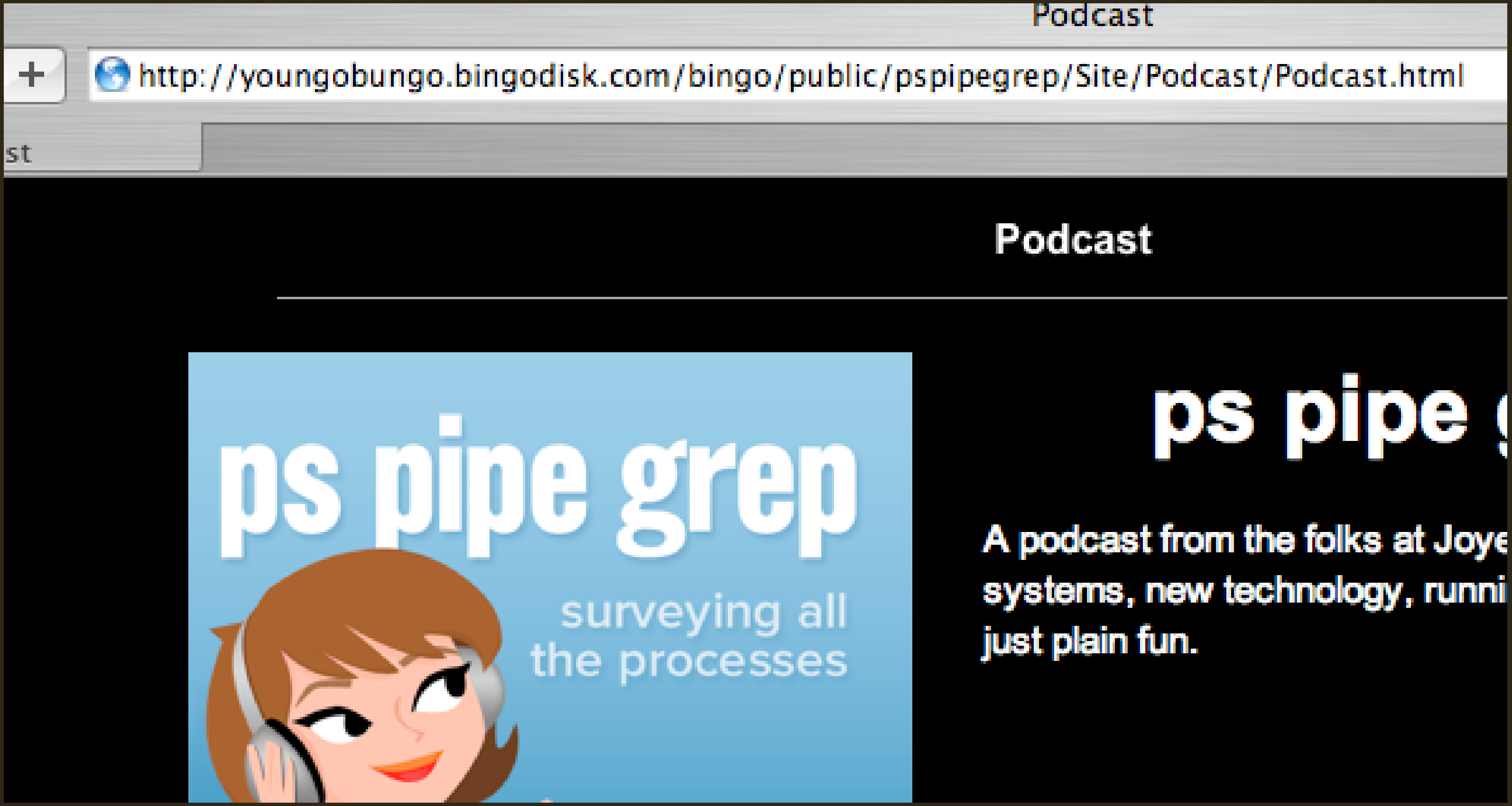
- Introducing BingoDisk
- Signup
- Getting Started Guide
- Contact

BingoDisk

- ✓ 100 gigabytes of storage for $199 a year
- ✓ 50 gigabytes of storage for $99 a year
- ✓ 25 gigabytes of storage for $49 a year
- ✓ 15 gigabytes of storage for $29 a year
- ✓ 10 gigabytes of storage for $19 a year

Welcome to wide open spaces. Joyent's new BingoDisk gives you 100, 50, 25, 15 or 10 gigabytes of disk stor amazing X4500 platform with unlimited bandwidth (ju over WebDAV for a super-low price! You get one user ability to serve files from a public folder (for images, p

Specifications

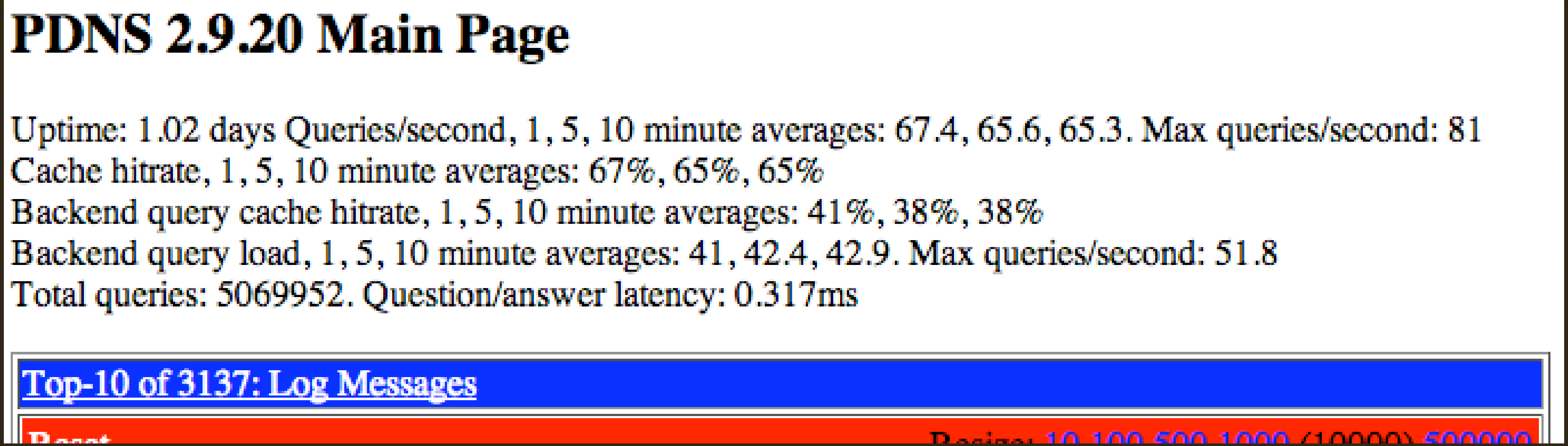


http://youngobungo.bingodisk.com/bingo/public/pspipegrep/Site/Podcast/Podcast.html

Podcast

ps pipe g

A podcast from the folks at Joye
systems, new technology, runni
just plain fun.

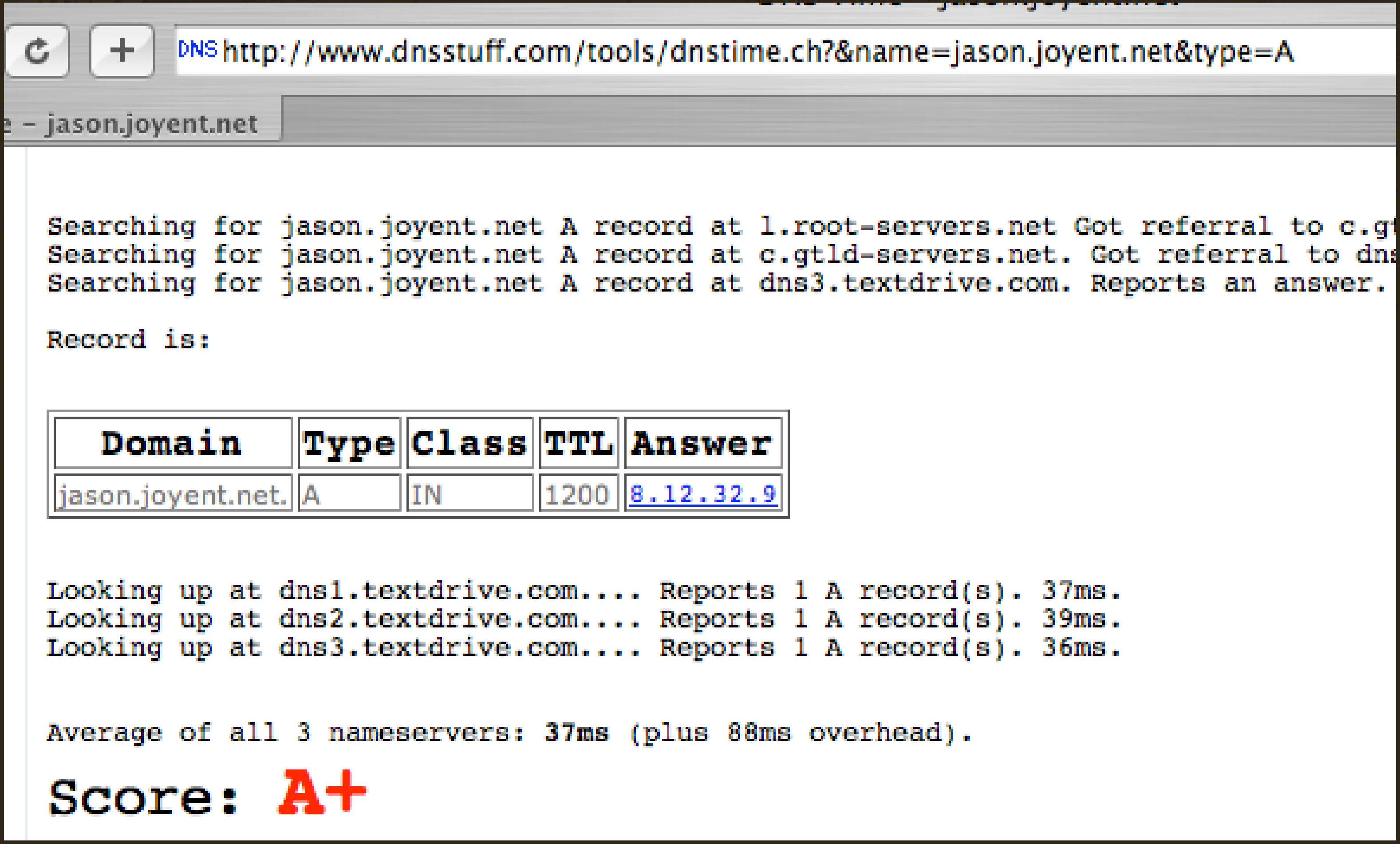


PDNS 2.9.20 Main Page

Uptime: 1.02 days Queries/second, 1, 5, 10 minute averages: 67.4, 65.6, 65.3. Max queries/second: 81
Cache hitrate, 1, 5, 10 minute averages: 67%, 65%, 65%
Backend query cache hitrate, 1, 5, 10 minute averages: 41%, 38%, 38%
Backend query load, 1, 5, 10 minute averages: 41, 42.4, 42.9. Max queries/second: 51.8
Total queries: 5069952. Question/answer latency: 0.317ms

Top-10 of 3137: Log Messages

Reset Resize: 10 100 500 1000 (10000) 500000

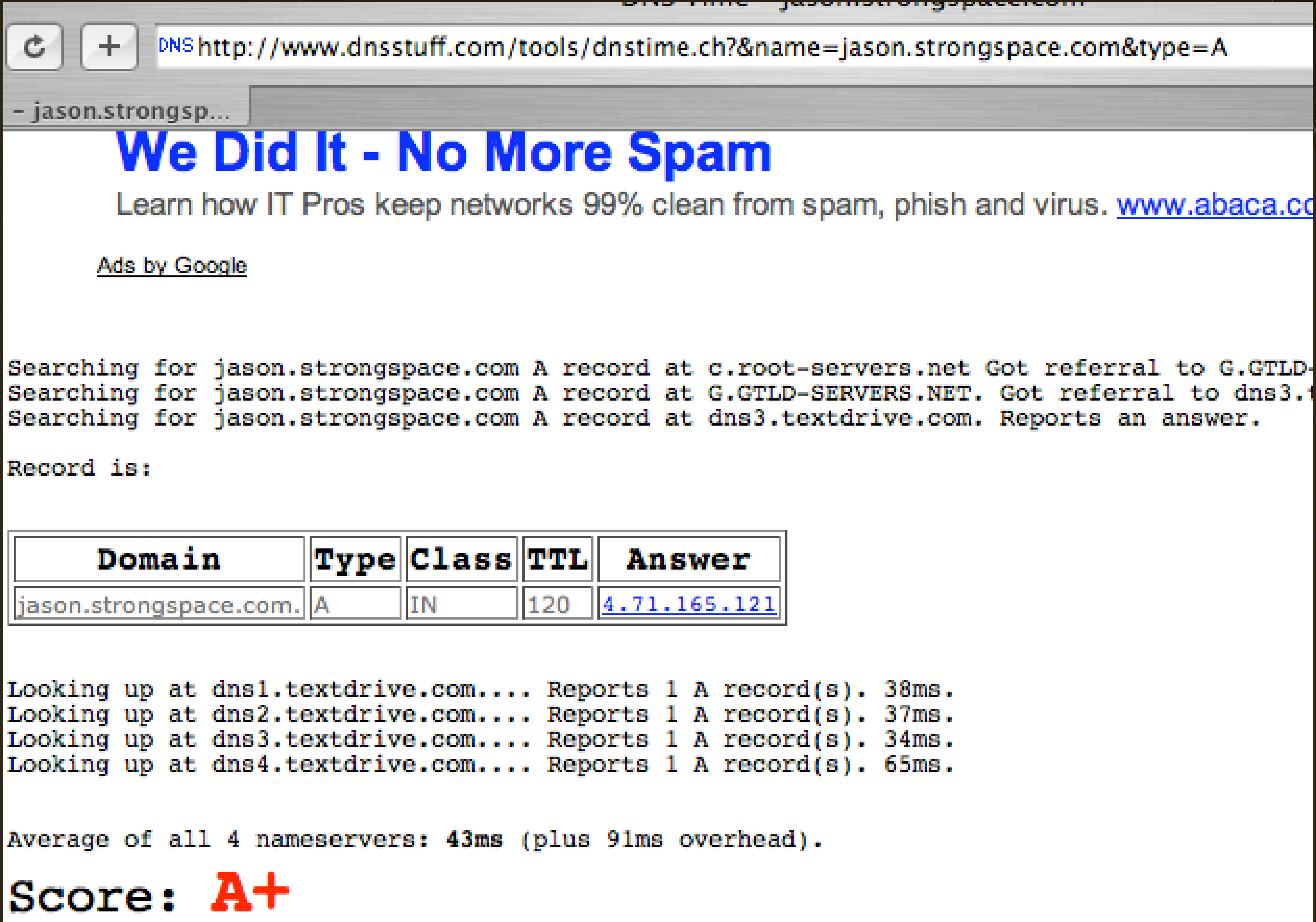

http://www.dnsstuff.com/tools/dnstime.ch?&name=jason.joyent.net&type=A
e – jason.joyent.net

```
Searching for jason.joyent.net A record at l.root-servers.net Got referral to c.gt
Searching for jason.joyent.net A record at c.gtld-servers.net. Got referral to dns
Searching for jason.joyent.net A record at dns3.textdrive.com. Reports an answer.

Record is:
```

Domain	Type	Class	TTL	Answer
jason.joyent.net.	A	IN	1200	8.12.32.9

```
Looking up at dns1.textdrive.com.... Reports 1 A record(s). 37ms.
Looking up at dns2.textdrive.com.... Reports 1 A record(s). 39ms.
Looking up at dns3.textdrive.com.... Reports 1 A record(s). 36ms.


Average of all 3 nameservers: 37ms (plus 88ms overhead).
```

Score: A+



http://www.dnsstuff.com/tools/dnstime.ch?&name=jason.strongspace.com&type=A

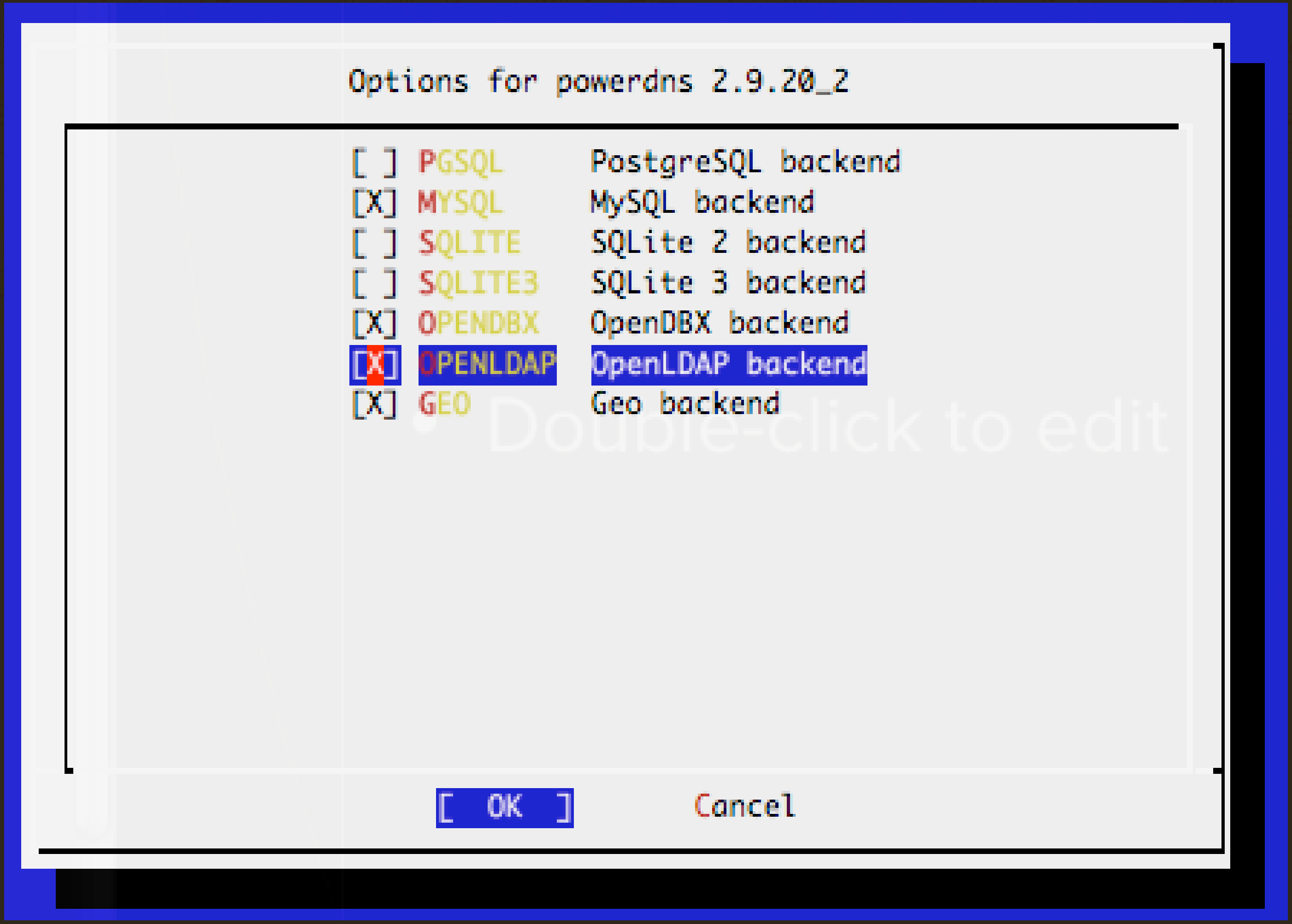
```
Searching for jason.strongspace.com A record at c.root-servers.net Got referral to G.GTLD-
Searching for jason.strongspace.com A record at G.GTLD-SERVERS.NET. Got referral to dns3.t
Searching for jason.strongspace.com A record at dns3.textdrive.com. Reports an answer.

Record is:
```

Domain	Type	Class	TTL	Answer
jason.strongspace.com.	A	IN	120	4.71.165.121

```
Looking up at dns1.textdrive.com.... Reports 1 A record(s). 38ms.
Looking up at dns2.textdrive.com.... Reports 1 A record(s). 37ms.
Looking up at dns3.textdrive.com.... Reports 1 A record(s). 34ms.
Looking up at dns4.textdrive.com.... Reports 1 A record(s). 65ms.

Average of all 4 nameservers: 43ms (plus 91ms overhead).
```

Score: **A+**

```
dns1# uname -a

FreeBSD dns1.textdrive.com 5.3-RELEASE FreeBSD 5.3-
RELEASE #0: Fri Nov  5 04:19:18 UTC 2004
root@harlow.cse.buffalo.edu:/usr/obj/usr/src/sys/
GENERIC  i386

dns1# cd /usr/ports/dns/powerdns

dns1# make config

dns1# make install
```



```
                 Options for powerdns 2.9.20_2

                 [ ] PGSQL     PostgreSQL backend
                 [X] MYSQL     MySQL backend
                 [ ] SQLITE    SQLite 2 backend
                 [ ] SQLITE3   SQLite 3 backend
                 [X] OPENDBX   OpenDBX backend
                 [X] OPENLDAP  OpenLDAP backend
                 [X] GEO       Geo backend

                    [  OK  ]         Cancel
```

- Double-click to edit

```
dns1# head /usr/local/etc/pdns.conf

# MySQL

launch=gmysql

gmysql-host=127.0.0.1

gmysql-dbname=dns

gmysql-user=dns

gmysql-password=blahblahblahboo
```

```
CREATE TABLE domains (
  id int(11) NOT NULL auto_increment,
  name varchar(255) NOT NULL default '',
  master varchar(20) default NULL,
  last_check int(11) default NULL,
  type varchar(6) NOT NULL default '',
  notified_serial int(11) default NULL,
  account varchar(40) default NULL,
  PRIMARY KEY  (id),
  UNIQUE KEY name_index (name)
) TYPE=InnoDB;
```

```
CREATE TABLE records (
  id int(11) NOT NULL auto_increment,
  domain_id int(11) default NULL,
  name varchar(255) default NULL,
  type varchar(6) default NULL,
  content varchar(255) default NULL,
  ttl int(11) default NULL,
  prio int(11) default NULL,
  change_date int(11) default NULL,
  PRIMARY KEY  (id),
  KEY rec_name_index (name),
  KEY nametype_index (name,type),
  KEY domain_id (domain_id)
) TYPE=InnoDB;
```

```
CREATE TABLE zones (
  id int(11) NOT NULL auto_increment,
  domain_id int(11) NOT NULL default '0',
  owner int(11) NOT NULL default '0',
  comment text,
  PRIMARY KEY  (id)
) TYPE=MyISAM;
```

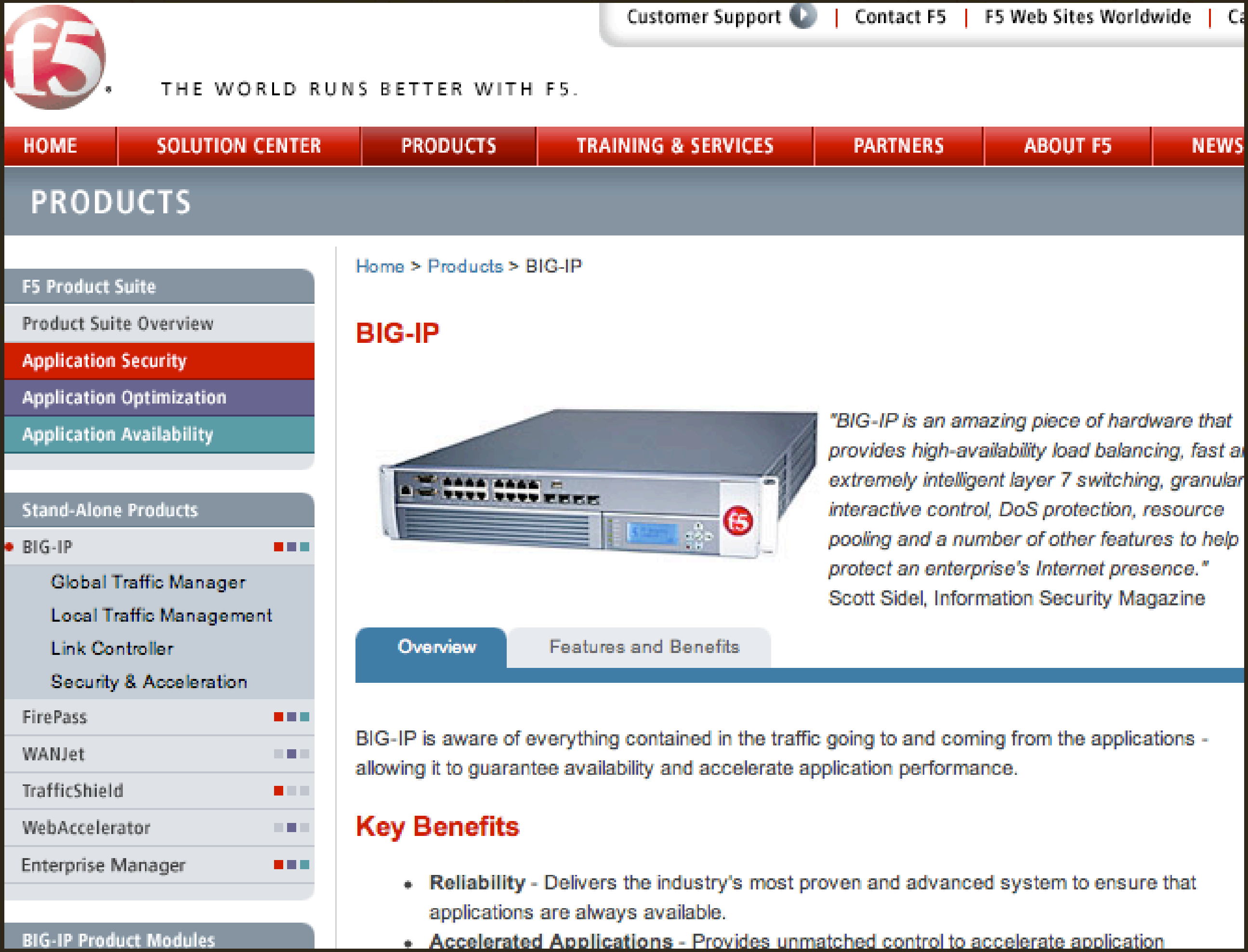
```
mysql> use dna;
ERROR 1044 (42000): Access denied for user 'dns'@'localhost' to database 'dna'
mysql> use dns;
Database changed
mysql> show tables;
+----------------+
| Tables_in_dns |
+----------------+
| domains        |
| records        |
| zones          |
+----------------+
3 rows in set (0.00 sec)
```

```
insert into domains (name,type) values ('joyent.com','NATIVE');

insert into records (domain_id, name,type,content,ttl,prio) select id ,'joyent.com', 'SOA',
'dns1.textdrive.com dns.textdrive.com 1086328940 10800 1800 10800 1800', 1800, 0 from domains where
name='joyent.com';

insert into records (domain_id, name,type,content,ttl,prio) select id ,'joyent.com', 'NS',
'dns1.textdrive.com', 120, 0 from domains where name='joyent.com';

insert into records (domain_id, name,type,content,ttl,prio) select id ,'*.joyent.com', 'A', '207.7.108.165
120, 0 from domains where name='joyent.com';
```

```
mysql> SELECT * FROM domains WHERE name = 'joyent.com';
+-------+------------+--------+------------+--------+-----------------+---------+
| id    | name       | master | last_check | type   | notified_serial | account |
+-------+------------+--------+------------+--------+-----------------+---------+
| 15811 | joyent.com | NULL   |       NULL | NATIVE |            NULL | NULL    |
+-------+------------+--------+------------+--------+-----------------+---------+
1 row in set (0.02 sec)

mysql> SELECT * FROM records WHERE domain_id = 15811 \G
*************************** 1. row ***************************
         id: 532305
  domain_id: 15811
       name: joyent.com
       type: A
    content: 4.71.165.93
        ttl: 180
       prio: 0
change_date: 1172471659
*************************** 2. row ***************************
         id: 532306
  domain_id: 15811
       name: _xmpp-server._tcp.joyent.com
       type: SRV
    content: 5 5269 jabber.joyent.com
        ttl: 180
       prio: 0
change_date: NULL
*************************** 3. row ***************************
         id: 532307
  domain_id: 15811
       name: _xmpp-client._tcp.joyent.com
       type: SRV
    content: 5 5222 jabber.joyent.com
        ttl: 180
       prio: 0
change_date: NULL
```

BIG-IPs

- http://f5.com/



Customer Support | Contact F5 | F5 Web Sites Worldwide | Ca

THE WORLD RUNS BETTER WITH F5.

PRODUCTS

Home > Products > BIG-IP

BIG-IP

"BIG-IP is an amazing piece of hardware that provides high-availability load balancing, fast a extremely intelligent layer 7 switching, granular interactive control, DoS protection, resource pooling and a number of other features to help protect an enterprise's Internet presence."
Scott Sidel, Information Security Magazine

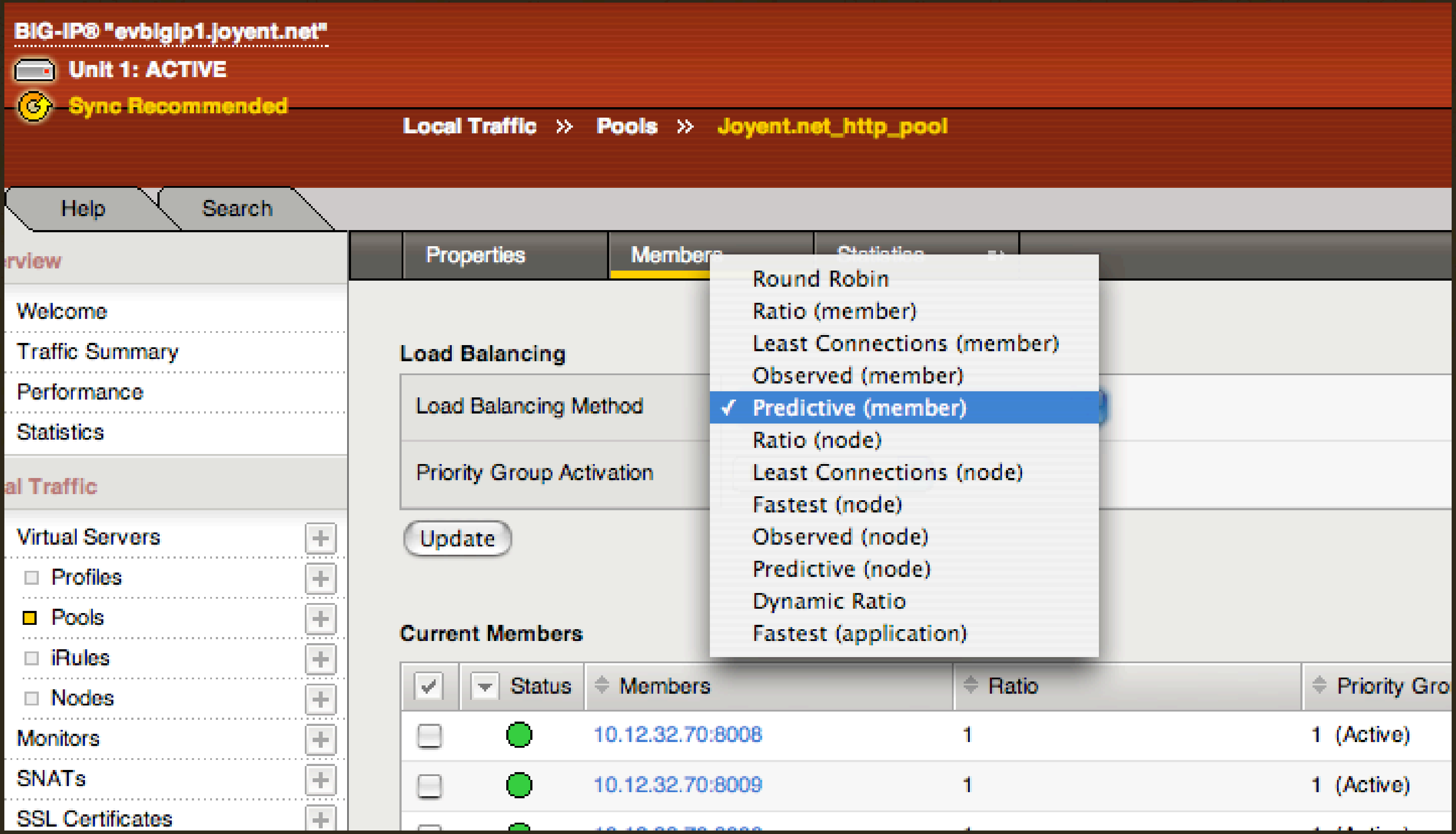
BIG-IP is aware of everything contained in the traffic going to and coming from the applications - allowing it to guarantee availability and accelerate application performance.

Key Benefits

- **Reliability** - Delivers the industry's most proven and advanced system to ensure that applications are always available.
- **Accelerated Applications** - Provides unmatched control to accelerate application

The real wins with BIG-IPs

- The only thing I’ve seen horizontally scale across a couple thousand mongrels
- Layer 7 and iRules (separate controllers)
- Full packet inspection



Load Balancing

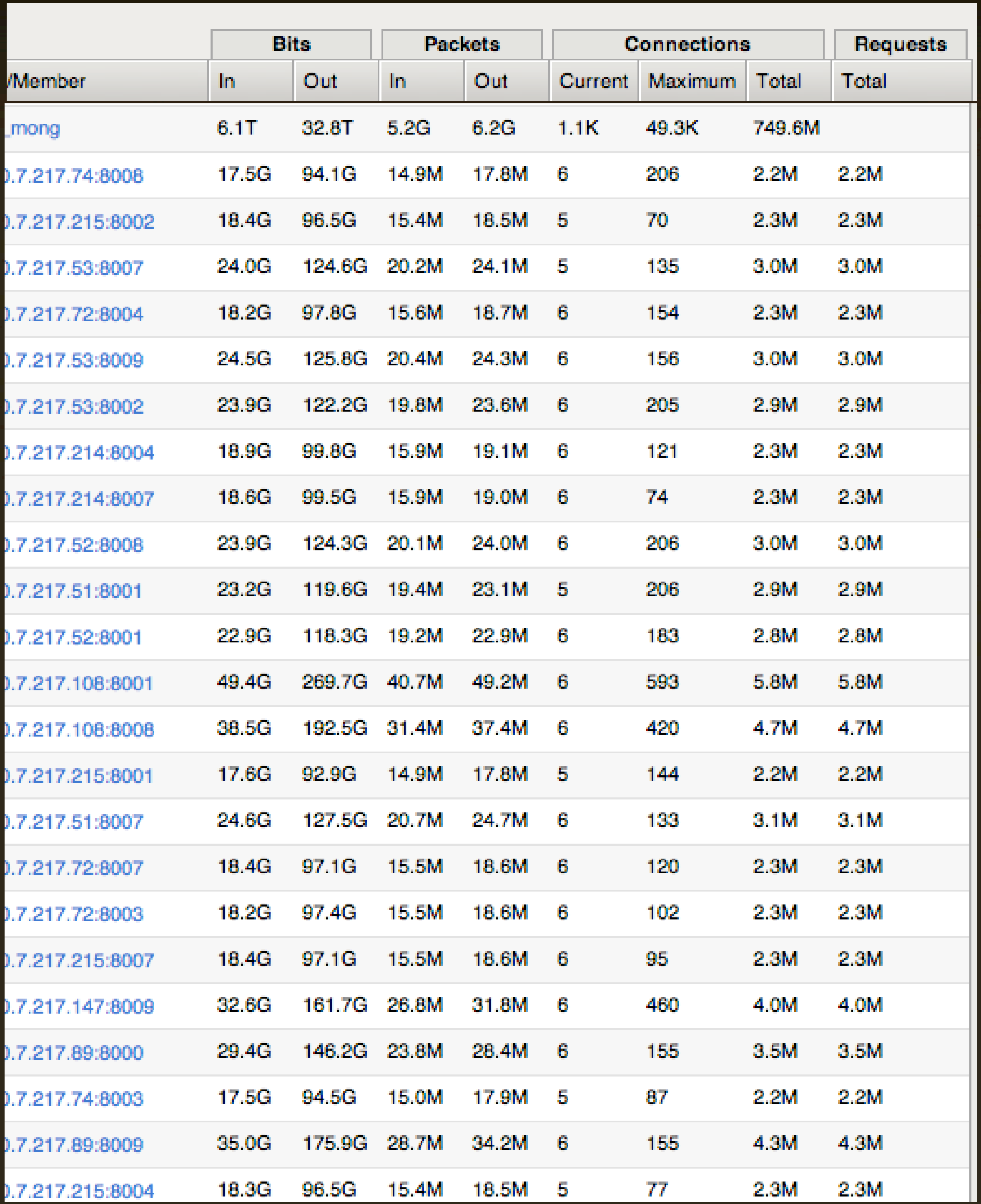
Load Balancing Method: Predictive (member)

Priority Group Activation

Current Members

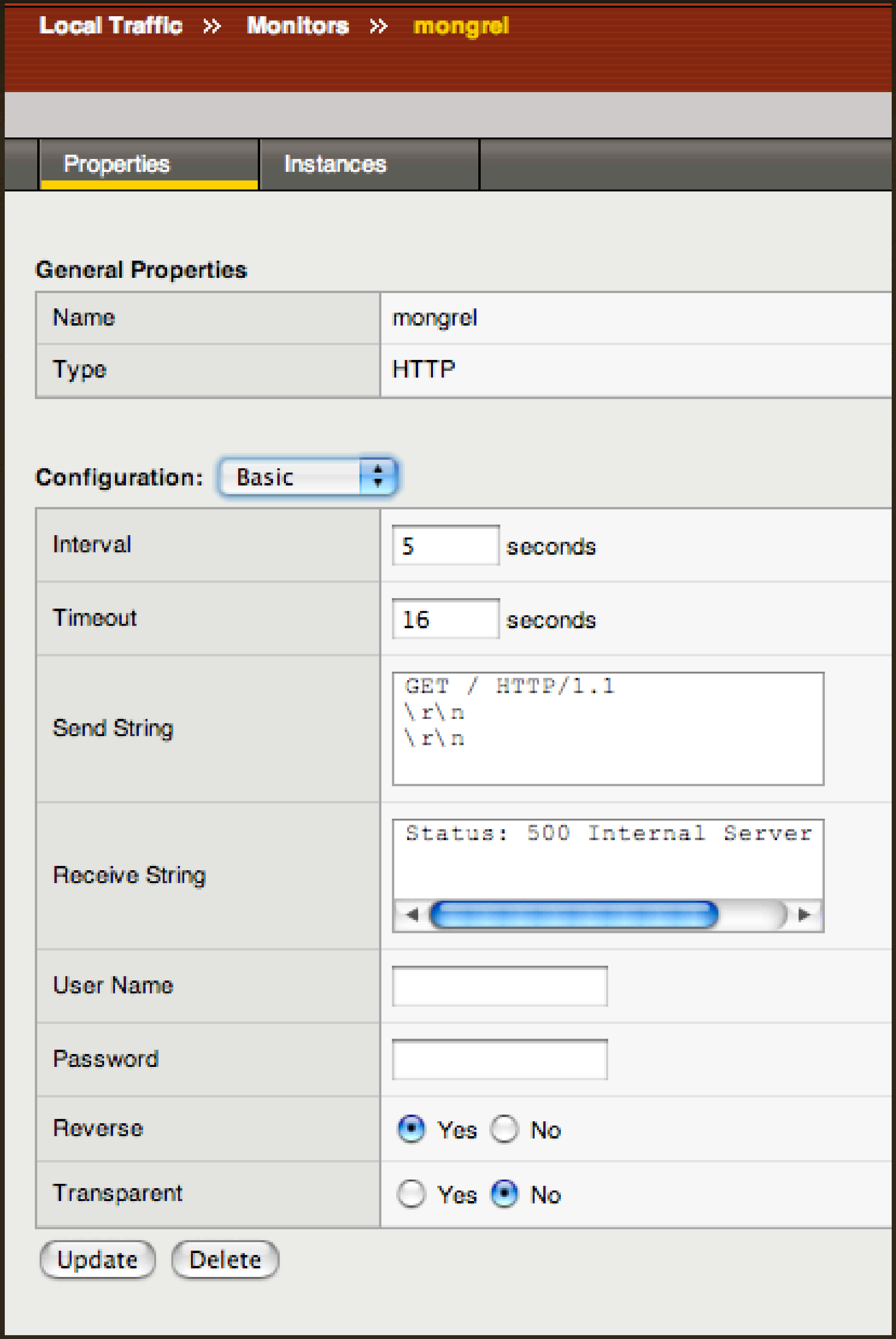
☑	Status	Members	Ratio	Priority Gro
□	●	10.12.32.70:8008	1	1 (Active)
□	●	10.12.32.70:8009	1	1 (Active)

Joyent



	Bits		Packets		Connections			Requests
/Member	In	Out	In	Out	Current	Maximum	Total	Total
_mong	6.1T	32.8T	5.2G	6.2G	1.1K	49.3K	749.6M	
).7.217.74:8008	17.5G	94.1G	14.9M	17.8M	6	206	2.2M	2.2M
).7.217.215:8002	18.4G	96.5G	15.4M	18.5M	5	70	2.3M	2.3M
).7.217.53:8007	24.0G	124.6G	20.2M	24.1M	5	135	3.0M	3.0M
).7.217.72:8004	18.2G	97.8G	15.6M	18.7M	6	154	2.3M	2.3M
).7.217.53:8009	24.5G	125.8G	20.4M	24.3M	6	156	3.0M	3.0M
).7.217.53:8002	23.9G	122.2G	19.8M	23.6M	6	205	2.9M	2.9M
).7.217.214:8004	18.9G	99.8G	15.9M	19.1M	6	121	2.3M	2.3M
).7.217.214:8007	18.6G	99.5G	15.9M	19.0M	6	74	2.3M	2.3M
).7.217.52:8008	23.9G	124.3G	20.1M	24.0M	6	206	3.0M	3.0M
).7.217.51:8001	23.2G	119.6G	19.4M	23.1M	5	206	2.9M	2.9M
).7.217.52:8001	22.9G	118.3G	19.2M	22.9M	6	183	2.8M	2.8M
).7.217.108:8001	49.4G	269.7G	40.7M	49.2M	6	593	5.8M	5.8M
).7.217.108:8008	38.5G	192.5G	31.4M	37.4M	6	420	4.7M	4.7M
).7.217.215:8001	17.6G	92.9G	14.9M	17.8M	5	144	2.2M	2.2M
).7.217.51:8007	24.6G	127.5G	20.7M	24.7M	6	133	3.1M	3.1M
).7.217.72:8007	18.4G	97.1G	15.5M	18.6M	6	120	2.3M	2.3M
).7.217.72:8003	18.2G	97.4G	15.5M	18.6M	6	102	2.3M	2.3M
).7.217.215:8007	18.4G	97.1G	15.5M	18.6M	6	95	2.3M	2.3M
).7.217.147:8009	32.6G	161.7G	26.8M	31.8M	6	460	4.0M	4.0M
).7.217.89:8000	29.4G	146.2G	23.8M	28.4M	6	155	3.5M	3.5M
).7.217.74:8003	17.5G	94.5G	15.0M	17.9M	5	87	2.2M	2.2M
).7.217.89:8009	35.0G	175.9G	28.7M	34.2M	6	155	4.3M	4.3M
).7.217.215:8004	18.3G	96.5G	15.4M	18.5M	5	77	2.3M	2.3M

Joyent



Local Traffic » Monitors » mongrel

General Properties

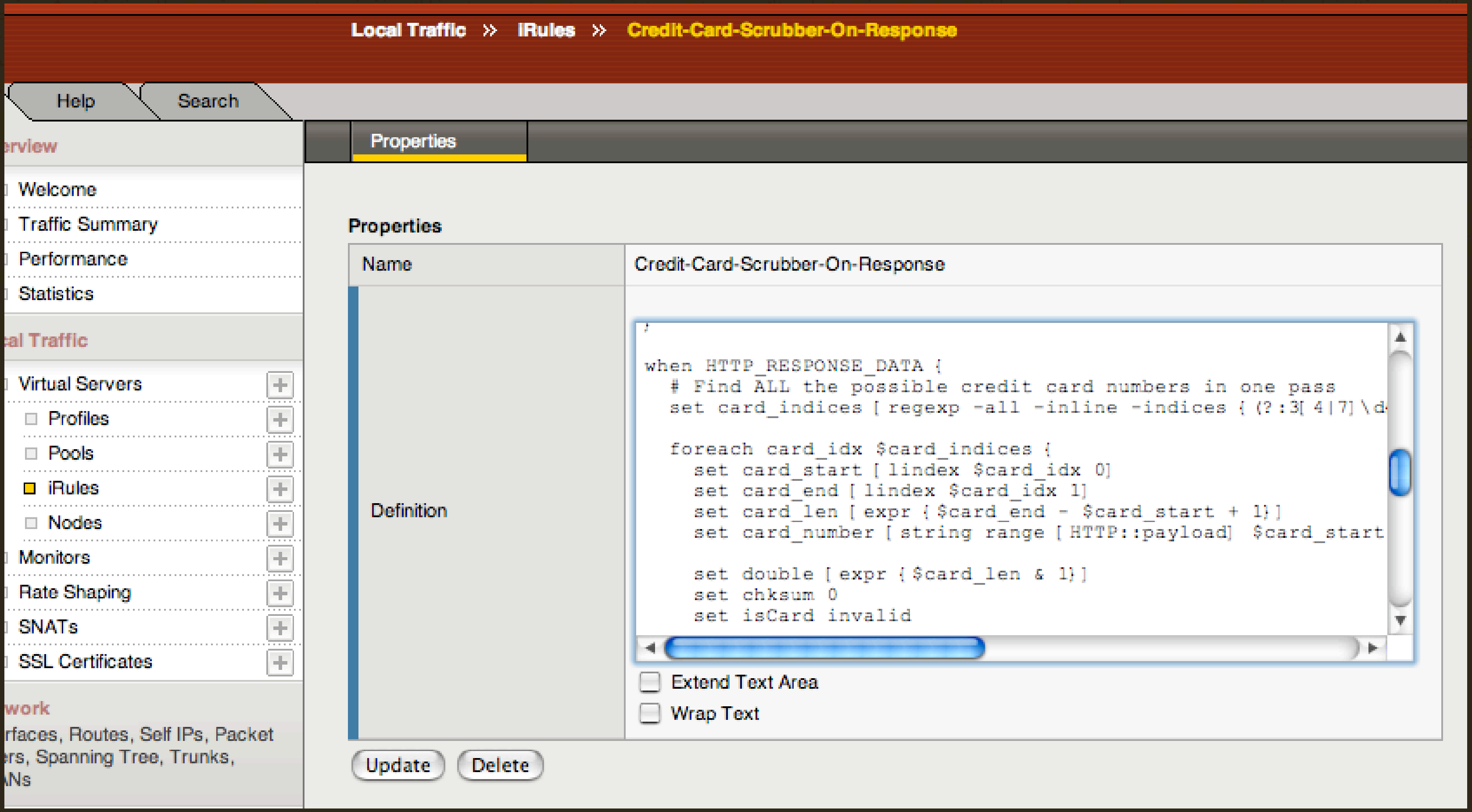
Name	mongrel
Type	HTTP

Configuration: Basic

Interval	5 seconds
Timeout	16 seconds
Send String	GET / HTTP/1.1 \r\n \r\n
Receive String	Status: 500 Internal Server
User Name	
Password	
Reverse	● Yes ○ No
Transparent	○ Yes ● No

```
when HTTP_REQUEST {
  if { [HTTP::uri] contains "svn" } {
     pool devror_svn
  } else {
     pool devror_trac
 }
}
```

```
when HTTP_REQUEST {
  if { [HTTP::host] contains "www"} {

  if { [HTTP::uri] contains "?" } {
        HTTP::redirect "http://twitter.com[HTTP::path]?[HTTP::query]"
     } else {
        HTTP::redirect "http://twitter.com[HTTP::path]"
     }
}
}
```



Properties

Name: Credit-Card-Scrubber-On-Response

Definition:

```
}

when HTTP_RESPONSE_DATA {
  # Find ALL the possible credit card numbers in one pass
  set card_indices [regexp -all -inline -indices {(?:3[4|7]\d
  
  foreach card_idx $card_indices {
    set card_start [lindex $card_idx 0]
    set card_end [lindex $card_idx 1]
    set card_len [expr {$card_end - $card_start + 1}]
    set card_number [string range [HTTP::payload] $card_start
    
    set double [expr {$card_len & 1}]
    set chksum 0
    set isCard invalid
```

□ Extend Text Area

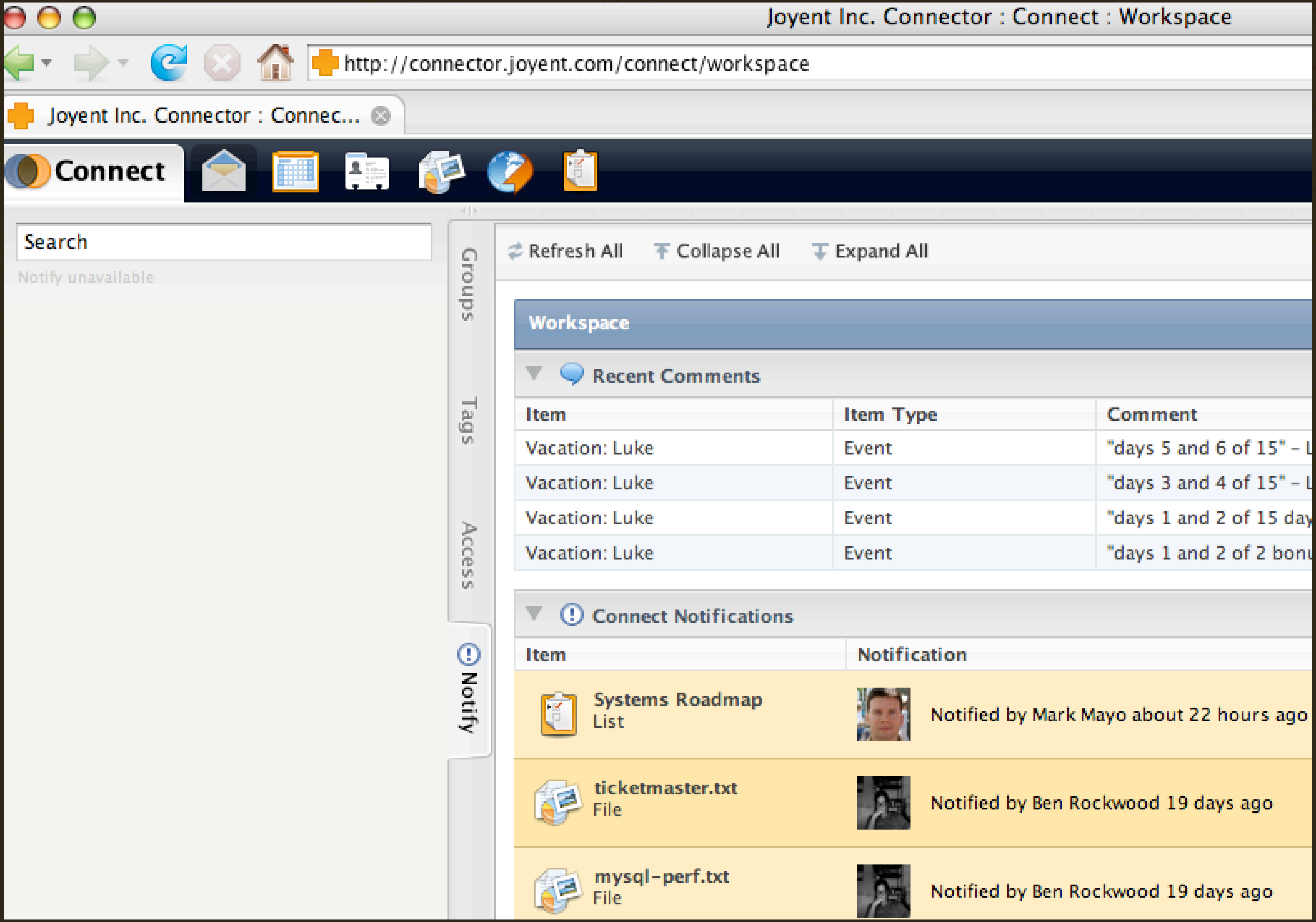
□ Wrap Text

```
when HTTP_REQUEST {
  # Don't allow data to be chunked
  if { [HTTP::version] eq "1.1" } {
      if { [HTTP::header is_keepalive] } {
         HTTP::header replace "Connection" "Keep-Alive"
      }
      HTTP::version "1.0"
   }
}

when HTTP_RESPONSE {
  # Only check responses that are a text content type
  # (text/html, text/xml, text/plain, etc).
  if { [HTTP::header "Content-Type"] starts_with "text/" } {
    # Get the content length so we can request the data to be
    # processed in the HTTP_RESPONSE_DATA event.
    if { [HTTP::header exists "Content-Length"] } {
      set content_length [HTTP::header "Content-Length"]
    } else {
      set content_length 4294967295
    }
    if { $content_length > 0 } {
       HTTP::collect $content_length
    }
  }
}

when HTTP_RESPONSE_DATA {
  # Find ALL the possible credit card numbers in one pass
  set card_indices [regexp -all -inline -indices {(?:3[4|7]\d{13})|(?:4\d{15})|(?:
5[1-5]\d{14})|(?:6011\d{12})} [HTTP::payload]]
```

Joyent

```
    # Calculate MOD10
    for { set i 0 } { $i < $card_len } { incr i } {
       set c [string index $card_number $i]
       if {($i & 1) == $double} {
          if {[incr c $c] >= 10} {incr c -9}
       }
       incr chksum $c
    }

    # Determine Card Type
    switch [string index $card_number 0] {
       3 { set type AmericanExpress }
       4 { set type Visa }
       5 { set type MasterCard }
       6 { set type Discover }
       default { set type Unknown }
    }

    # If valid card number, then mask out numbers with X's
    if { ($chksum % 10) == 0 } {
       set isCard valid
       HTTP::payload replace $card_start $card_len [string repeat "X" $card_len]
    }

    # Log Results
    log local0. "Found $isCard $type CC# $card_number"
  }
}
```

Layer7


Joyent Inc. Connector : Connect : Workspace
http://connector.joyent.com/connect/workspace
Joyent Inc. Connector : Connec...
Connect
Search
Notify unavailable
Groups
Tags
Access
Notify
Refresh All
Collapse All
Expand All

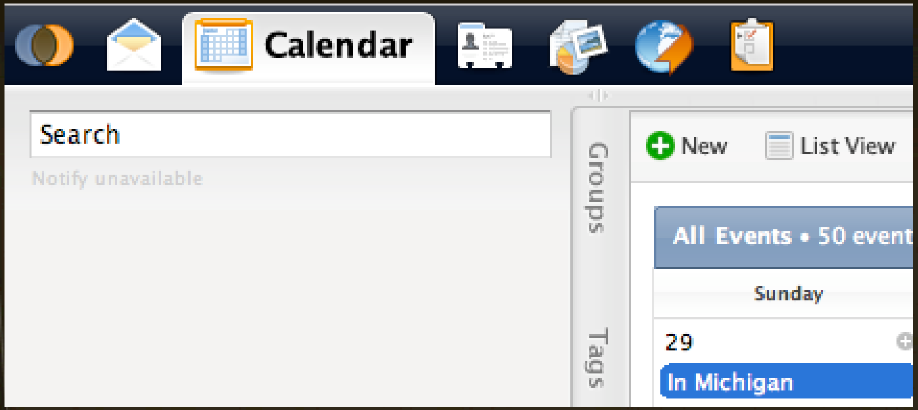
Workspace

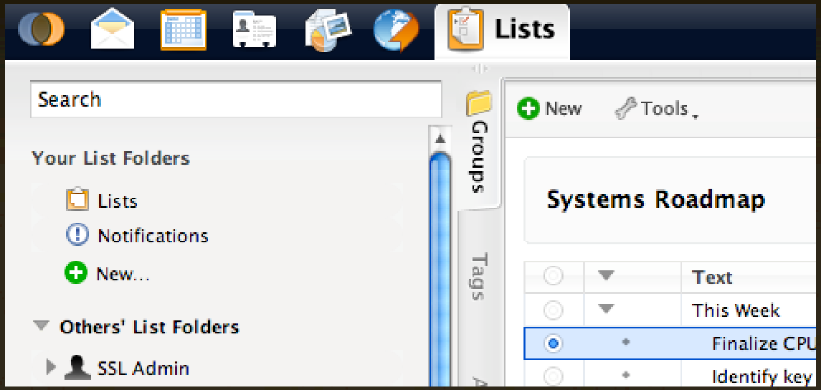
Recent Comments

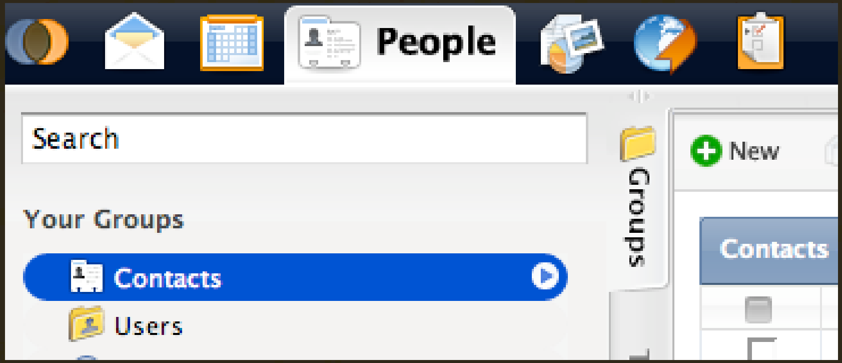
Item	Item Type	Comment
Vacation: Luke	Event	"days 5 and 6 of 15" – L
Vacation: Luke	Event	"days 3 and 4 of 15" – L
Vacation: Luke	Event	"days 1 and 2 of 15 day
Vacation: Luke	Event	"days 1 and 2 of 2 bonu

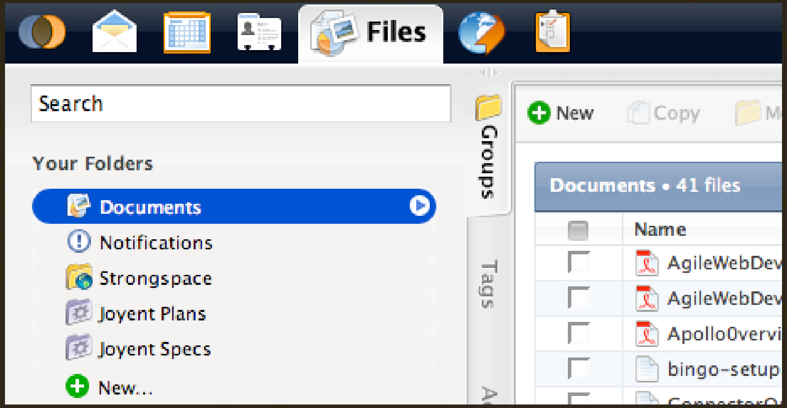
Connect Notifications

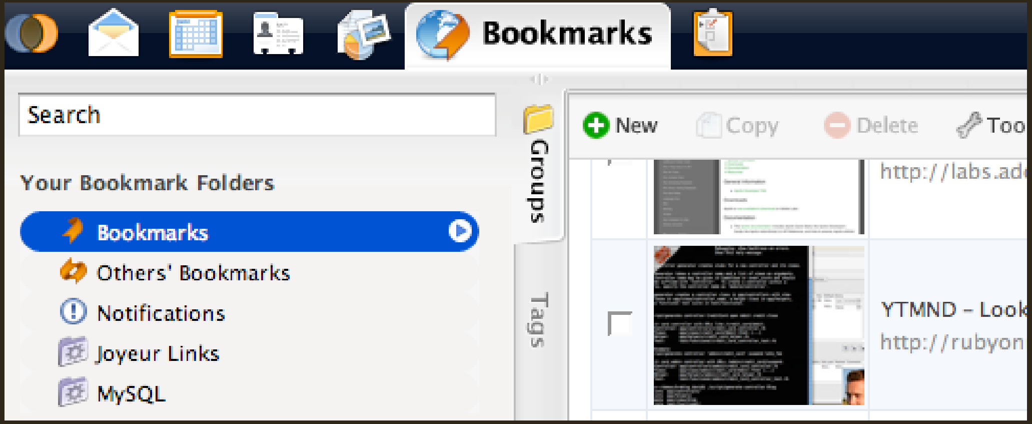
Item	Notification
Systems Roadmap List	Notified by Mark Mayo about 22 hours ago
ticketmaster.txt File	Notified by Ben Rockwood 19 days ago
mysql-perf.txt File	Notified by Ben Rockwood 19 days ago


Calendar
Search
Notify unavailable
Groups
Tags
New
List View
All Events • 50 event
Sunday
29
In Michigan


Lists
Search
Your List Folders
Lists
Notifications
New...
Others' List Folders
SSL Admin
Groups
Tags
New
Tools
Systems Roadmap
Text
This Week
Finalize CPU
Identify key


People
Search
Your Groups
Contacts
Users
Groups
New
Contacts


Files
Search
Your Folders
Documents
Notifications
Strongspace
Joyent Plans
Joyent Specs
New...
Groups
Tags
New
Copy
Documents • 41 files
Name
AgileWebDev
AgileWebDev
ApolloOvervi
bingo-setup


Bookmarks
Search
Your Bookmark Folders
Bookmarks
Others' Bookmarks
Notifications
Joyeur Links
MySQL
Groups
Tags
New
Copy
Delete
Too
http://labs.ad
YTMND – Look
http://rubyon

Joyent

Each controller has their own app servers

- http://jason.joyent.net/mail
- http://jason.joyent.net/lists
- http://jason.joyent.net/calendar
- http://jason.joyent.net/login

The partitioning and federation then possible ...

Free software LB alternatives

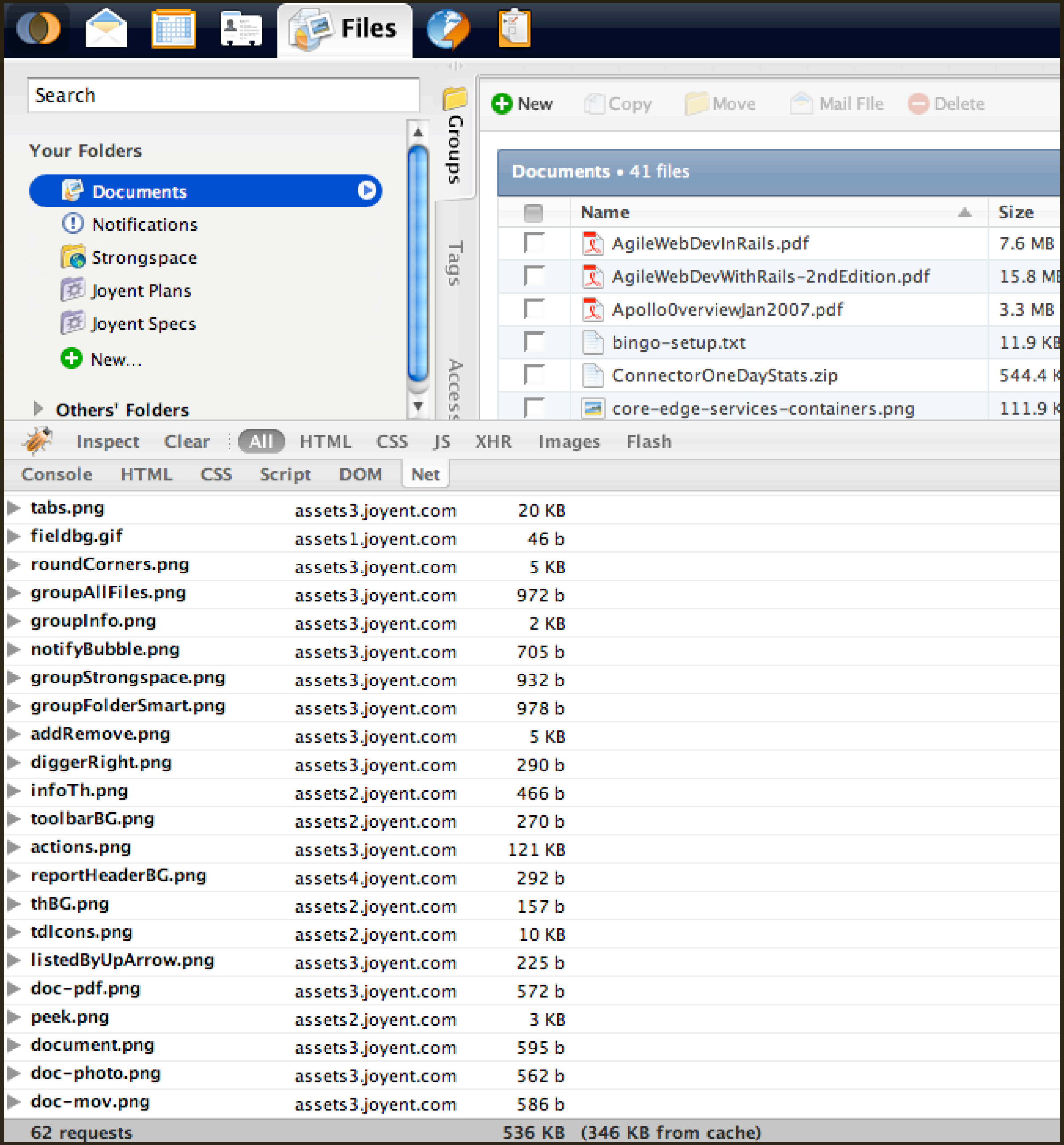
- That I also like and think will get you far
- Varnish
- HA-Proxy

The appearance of a rule of 10

- Apache, Nginx, Lighttpd, Litespeed: ≤1000 req/sec
- Varnish: 10,000 req/sec
- BIG-IP: 100,000 req/sec

My preferred web server + LB proxy

- Nginx
- Static assets with solaris event ports as the engine



Documents • 41 files

□	Name	Size
□	AgileWebDevInRails.pdf	7.6 MB
□	AgileWebDevWithRails-2ndEdition.pdf	15.8 MB
□	ApolloOverviewJan2007.pdf	3.3 MB
□	bingo-setup.txt	11.9 KB
□	ConnectorOneDayStats.zip	544.4 K
□	core-edge-services-containers.png	111.9 K

```
▶ tabs.png                assets3.joyent.com    20 KB
▶ fieldbg.gif             assets1.joyent.com    46 b
▶ roundCorners.png        assets3.joyent.com     5 KB
▶ groupAllFiles.png       assets3.joyent.com   972 b
▶ groupInfo.png           assets3.joyent.com     2 KB
▶ notifyBubble.png        assets3.joyent.com   705 b
▶ groupStrongspace.png    assets3.joyent.com   932 b
▶ groupFolderSmart.png    assets3.joyent.com   978 b
▶ addRemove.png           assets3.joyent.com     5 KB
▶ diggerRight.png         assets3.joyent.com   290 b
▶ infoTh.png              assets2.joyent.com   466 b
▶ toolbarBG.png           assets2.joyent.com   270 b
▶ actions.png             assets3.joyent.com   121 KB
▶ reportHeaderBG.png      assets4.joyent.com   292 b
▶ thBG.png                assets2.joyent.com   157 b
▶ tdIcons.png             assets2.joyent.com    10 KB
▶ listedByUpArrow.png     assets3.joyent.com   225 b
▶ doc-pdf.png             assets3.joyent.com   572 b
▶ peek.png                assets2.joyent.com     3 KB
▶ document.png            assets3.joyent.com   595 b
▶ doc-photo.png           assets3.joyent.com   562 b
▶ doc-mov.png             assets3.joyent.com   586 b
  62 requests                                  536 KB  (346 KB from cache)
```

The back

- Your datastore
- What is your data?
- Where does it live?
- How does it relate to hardware size?
- How do you interact with it?
- This can be built into the application

What do I mean by built-in?

- MySQL handling the replication?
- Sequoia (middleware) handling the replication?
- Application handling the replication?

Another advantage with JRuby

- Is the interaction with the backends

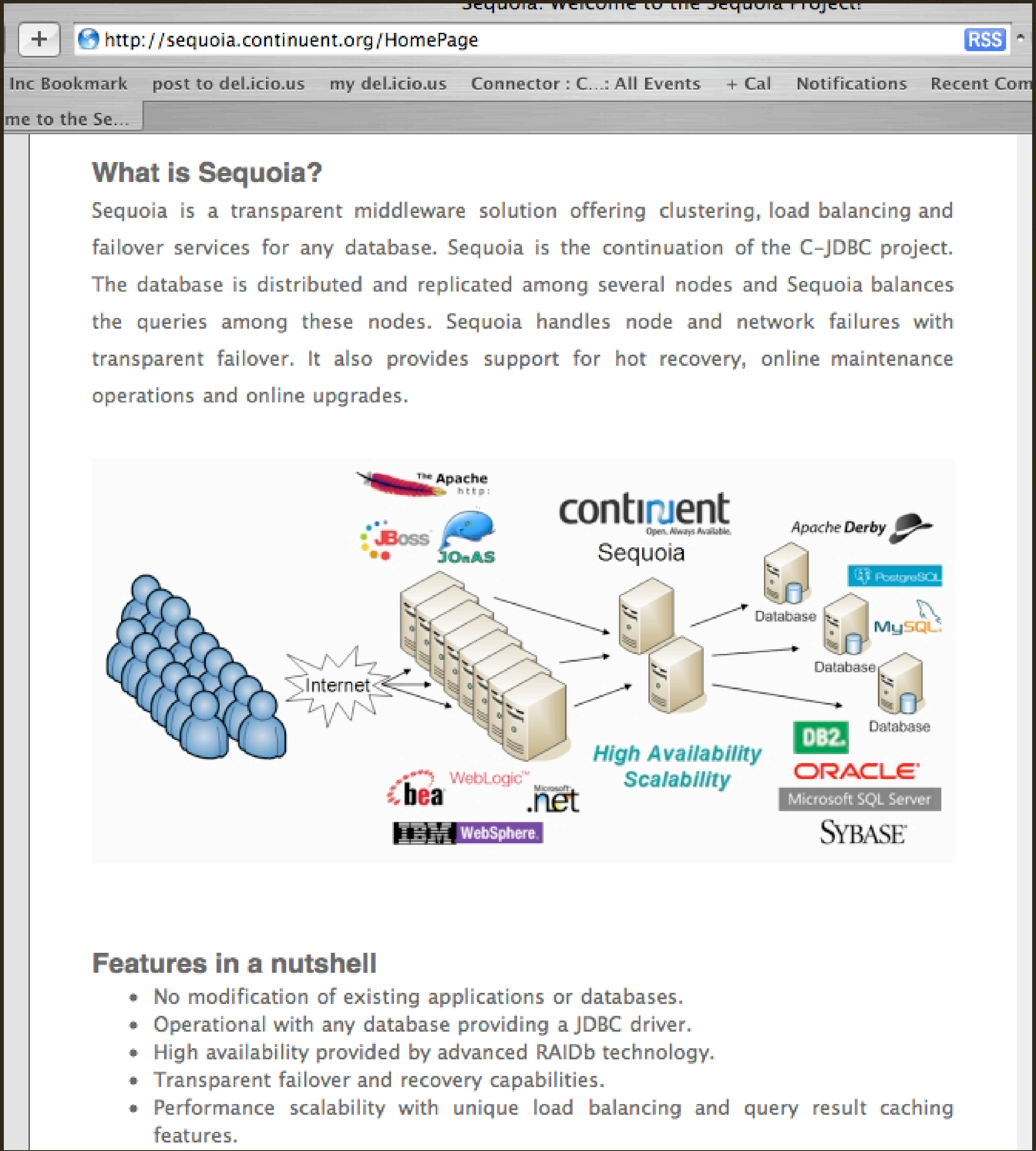
JDBC

```
$ /a/1/app/glassfish/bin/asadmin create-jdbc-connection-pool \
  --restype javax.sql.DataSource \
  --datasourceclassname
com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource \
  --property User=root:Password=root:URL=jdbc\\:mysql\\://localhost/
railstest_production \
  jdbc/railstestpool
/a/1/app/glassfish/bin/asadmin ping-connection-pool jdbc/railstestpool
```

```
$ /a/1/app/glassfish/bin/asadmin create-jdbc-resource \

  --connectionpoolid jdbc/railstestpool jdbc/railstest
```

You get database connection pooling and caching then

You get “traditional” middleware



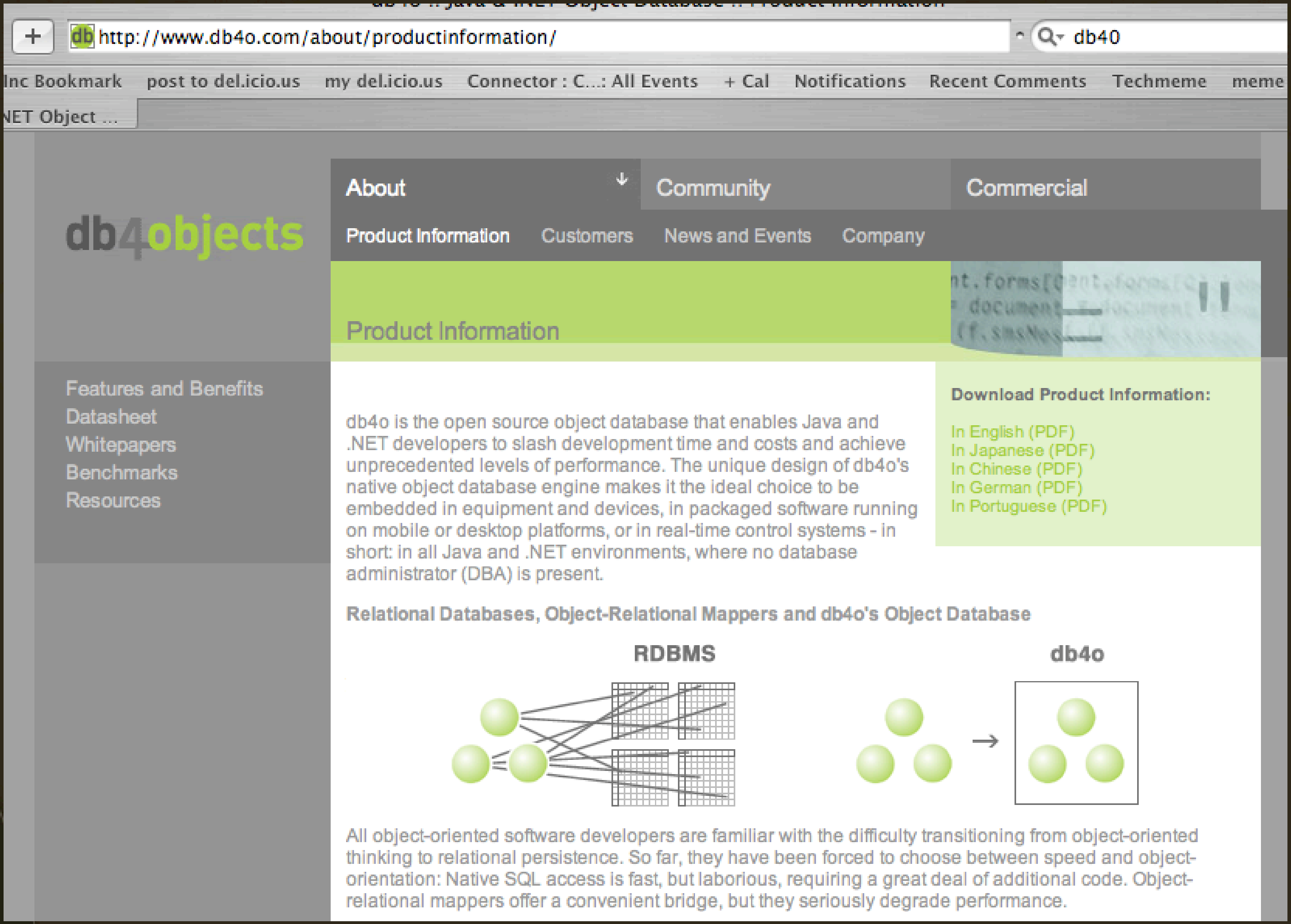
What is Sequoia?

Sequoia is a transparent middleware solution offering clustering, load balancing and failover services for any database. Sequoia is the continuation of the C-JDBC project. The database is distributed and replicated among several nodes and Sequoia balances the queries among these nodes. Sequoia handles node and network failures with transparent failover. It also provides support for hot recovery, online maintenance operations and online upgrades.

Features in a nutshell

- No modification of existing applications or databases.
- Operational with any database providing a JDBC driver.
- High availability provided by advanced RAIDb technology.
- Transparent failover and recovery capabilities.
- Performance scalability with unique load balancing and query result caching features.

You get increasing access to ODMBS



Product Information

db4o is the open source object database that enables Java and .NET developers to slash development time and costs and achieve unprecedented levels of performance. The unique design of db4o's native object database engine makes it the ideal choice to be embedded in equipment and devices, in packaged software running on mobile or desktop platforms, or in real-time control systems - in short: in all Java and .NET environments, where no database administrator (DBA) is present.

Download Product Information:

In English (PDF)
In Japanese (PDF)
In Chinese (PDF)
In German (PDF)
In Portuguese (PDF)

Relational Databases, Object-Relational Mappers and db4o's Object Database

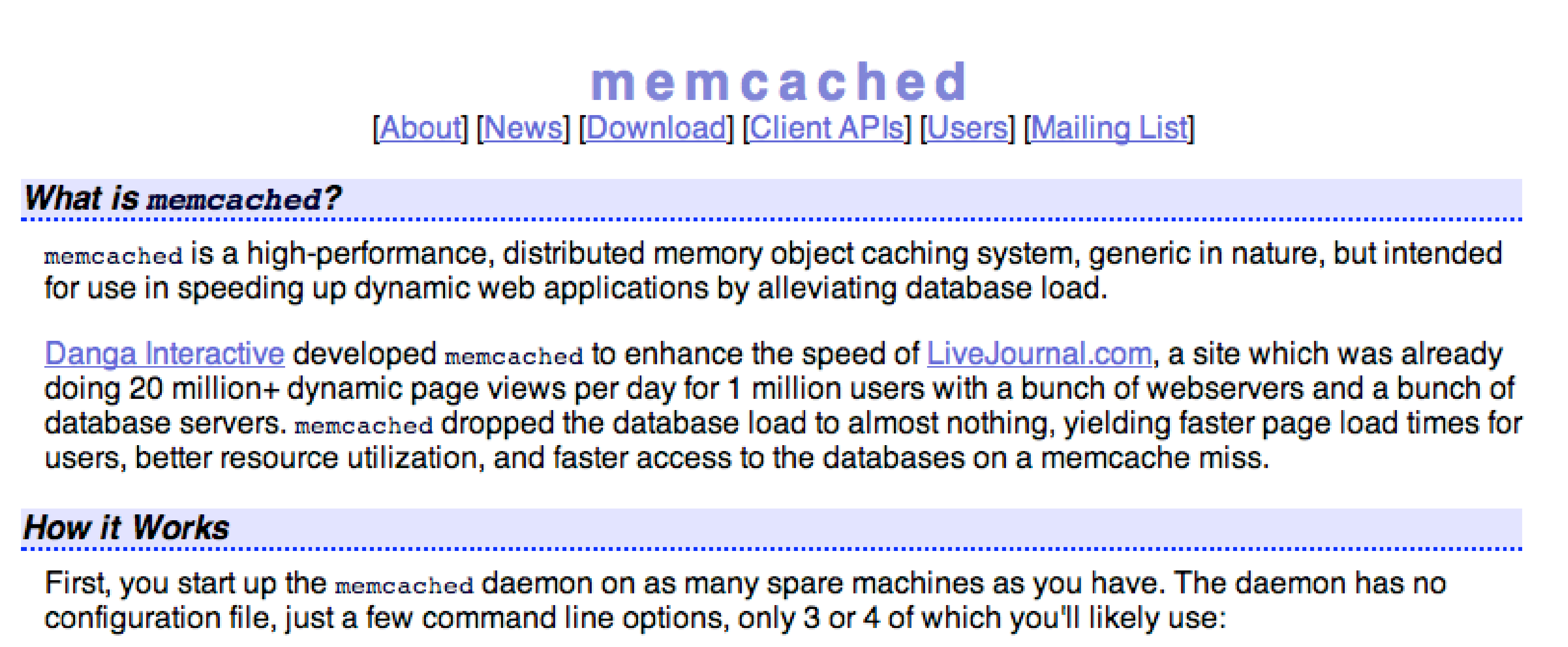
All object-oriented software developers are familiar with the difficulty transitioning from object-oriented thinking to relational persistence. So far, they have been forced to choose between speed and object-orientation: Native SQL access is fast, but laborious, requiring a great deal of additional code. Object-relational mappers offer a convenient bridge, but they seriously degrade performance.

Maybe a RDMS isn’t the only thing

- Memcache (in memory and easy)
- LDAP
- Message buses with an in-memory db (JEAI, ActiveMQ)
- Object databases
- File system

memcached

- http://www.danga.com/memcached/



memcached

[About] [News] [Download] [Client APIs] [Users] [Mailing List]

What is `memcached`?

`memcached` is a high-performance, distributed memory object caching system, generic in nature, but intended for use in speeding up dynamic web applications by alleviating database load.

Danga Interactive developed `memcached` to enhance the speed of LiveJournal.com, a site which was already doing 20 million+ dynamic page views per day for 1 million users with a bunch of webservers and a bunch of database servers. `memcached` dropped the database load to almost nothing, yielding faster page load times for users, better resource utilization, and faster access to the databases on a memcache miss.

How it Works

First, you start up the `memcached` daemon on as many spare machines as you have. The daemon has no configuration file, just a few command line options, only 3 or 4 of which you'll likely use:

J-EAI

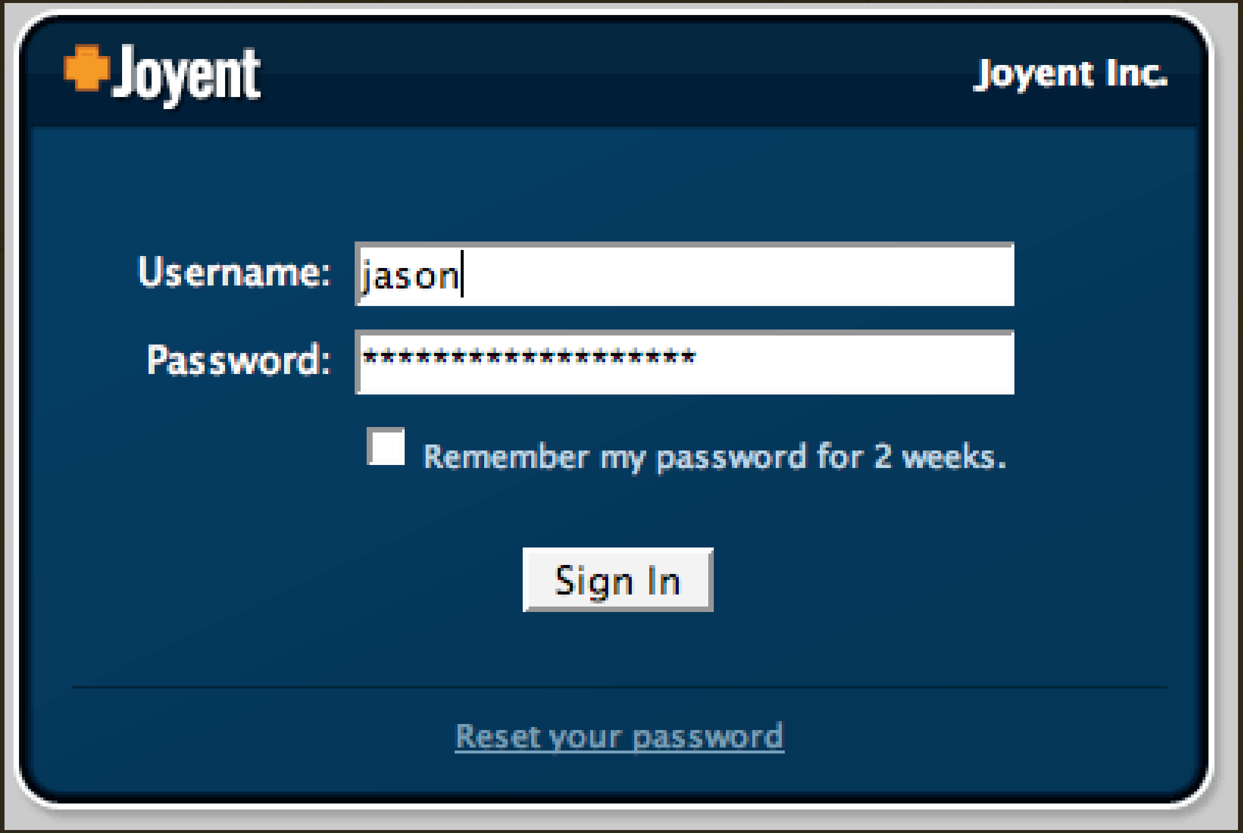
- XMPP-Jabber message bus for XML (atom)
- Erlang-based
- Cluster-ready and very scalable
- Lots of connectors: SMTP, JDBC
- App <-> Bus <-> Database

Asynchronous messaging

- http://code.google.com/p/activemessaging/
- ActiveMQ
- Stomp (http://stomp.codehaus.org/)

LDAP

- Hierarchical database
- Great for parent-child modeled data
- We use for all authentication, user databases, DNS ...
- Basically as much as we can

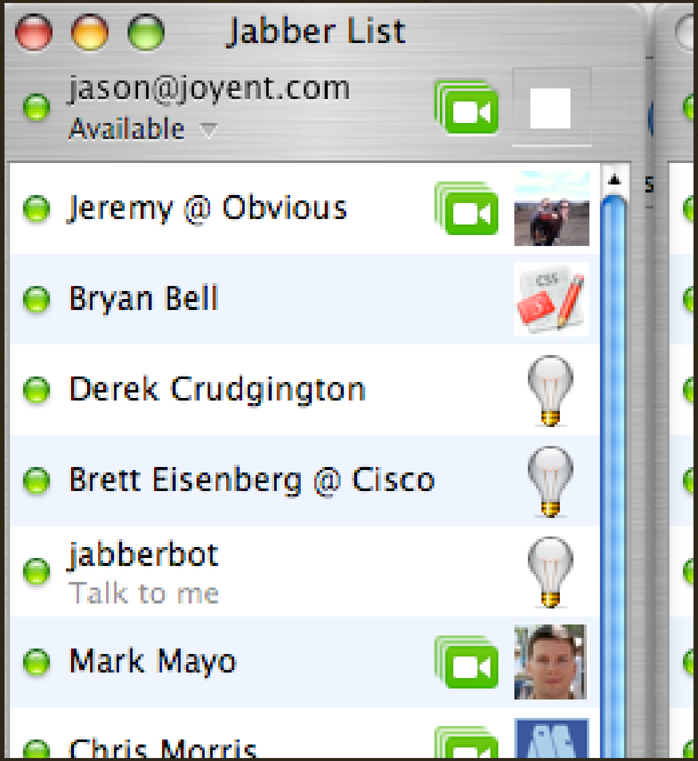


Joyent Inc.

Username: jason

Password: ********************

□ Remember my password for 2 weeks.



Jabber List

jason@joyent.com
Available

Jeremy @ Obvious

Bryan Bell

Derek Crudgington

Brett Eisenberg @ Cisco

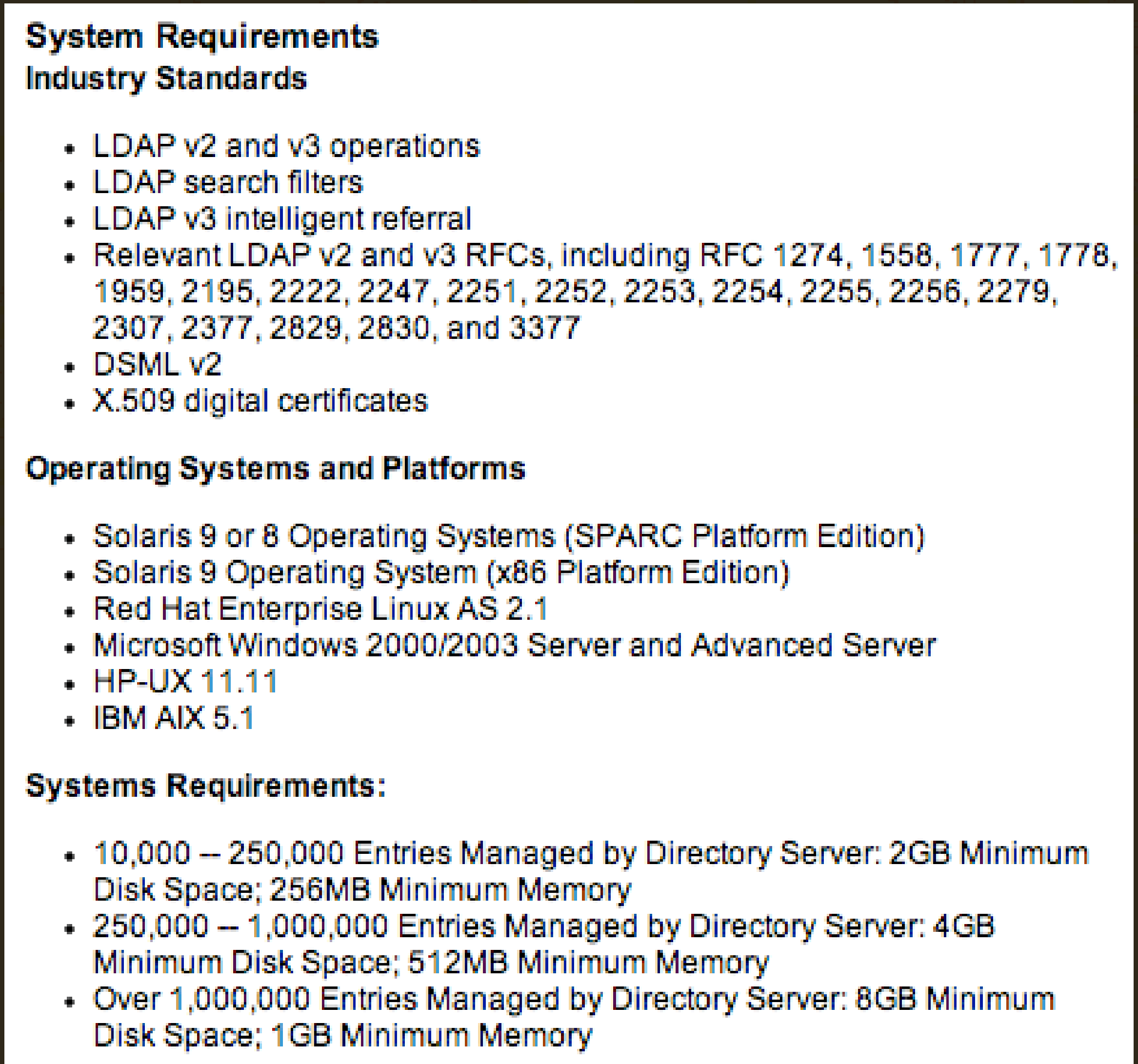
jabberbot
Talk to me

Mark Mayo

Chris Morris

Why?

The multi-master replication is amazing when you’ve been living in MySQL and PostgreSQL lands



System Requirements

Industry Standards

- LDAP v2 and v3 operations
- LDAP search filters
- LDAP v3 intelligent referral
- Relevant LDAP v2 and v3 RFCs, including RFC 1274, 1558, 1777, 1778, 1959, 2195, 2222, 2247, 2251, 2252, 2253, 2254, 2255, 2256, 2279, 2307, 2377, 2829, 2830, and 3377
- DSML v2
- X.509 digital certificates

Operating Systems and Platforms

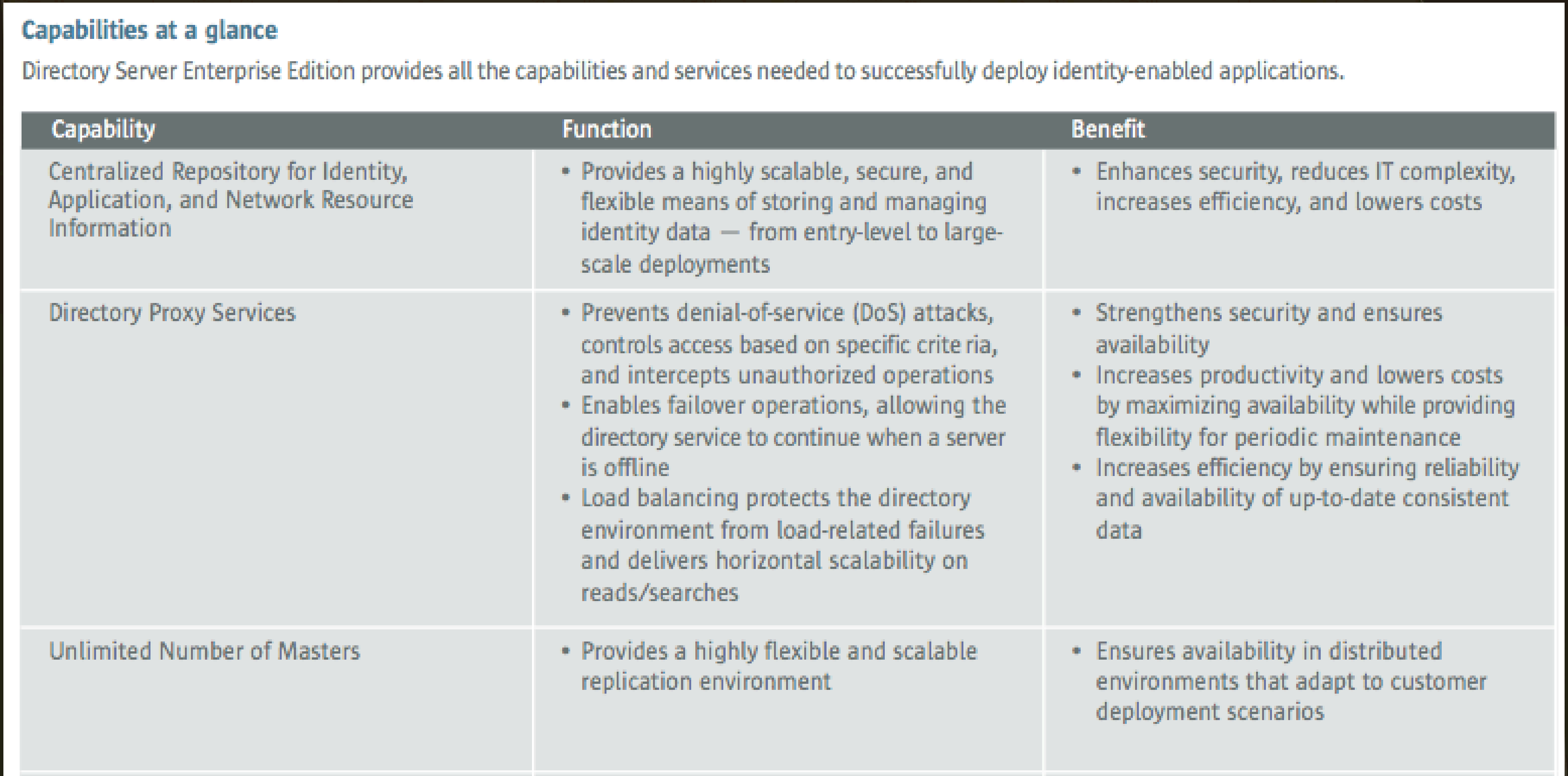
- Solaris 9 or 8 Operating Systems (SPARC Platform Edition)
- Solaris 9 Operating System (x86 Platform Edition)
- Red Hat Enterprise Linux AS 2.1
- Microsoft Windows 2000/2003 Server and Advanced Server
- HP-UX 11.11
- IBM AIX 5.1

Systems Requirements:

- 10,000 -- 250,000 Entries Managed by Directory Server: 2GB Minimum Disk Space; 256MB Minimum Memory
- 250,000 -- 1,000,000 Entries Managed by Directory Server: 4GB Minimum Disk Space; 512MB Minimum Memory
- Over 1,000,000 Entries Managed by Directory Server: 8GB Minimum Disk Space; 1GB Minimum Memory

Sina

- “With over 230 million registered users, over 42 million long-term paid users for special services, and over 450 million peak daily hits, Sina is one of the largest Web portals and a leading online media and value-added information service provider in China.”
- 12 Sun Fire T1000 servers running Solaris 10 and the Sun Java System Directory Server.



Capabilities at a glance

Directory Server Enterprise Edition provides all the capabilities and services needed to successfully deploy identity-enabled applications.

Capability	Function	Benefit
Centralized Repository for Identity, Application, and Network Resource Information	• Provides a highly scalable, secure, and flexible means of storing and managing identity data — from entry-level to large-scale deployments	• Enhances security, reduces IT complexity, increases efficiency, and lowers costs
Directory Proxy Services	• Prevents denial-of-service (DoS) attacks, controls access based on specific crite ria, and intercepts unauthorized operations • Enables failover operations, allowing the directory service to continue when a server is offline • Load balancing protects the directory environment from load-related failures and delivers horizontal scalability on reads/searches	• Strengthens security and ensures availability • Increases productivity and lowers costs by maximizing availability while providing flexibility for periodic maintenance • Increases efficiency by ensuring reliability and availability of up-to-date consistent data
Unlimited Number of Masters	• Provides a highly flexible and scalable replication environment	• Ensures availability in distributed environments that adapt to customer deployment scenarios

Pay attention to how you store your files

A story

Hashed directory structures

- Never more than 10K files / subdirs in a single directory (I aim for a max of 4K or so..)
- Keep it simple to implement / remember
- Don't get carried away and nest too deeply, that can hurt performance too

A couple of approaches

The 16x256

- Pre-create 16 top level dirs, 256 subdirs each which gives you 4096 "buckets".
- Keeping to the 10K per bucket rule, that's 4M "things" you can put into this structure. Go to 256 x 256 if you're big and/or want to keep the number of things in the buckets lower.

- How do you decide where to put stuff?
- Pick randomly from 1 to 16 and from 1 to 256. Store path in the profile. What's it look like:

```
userid=76340

fspath=/data/12/245/76340/file1,file2,etc..
```

- You get nice even distribution, but the downside is that you can't "compute" the directory path from the thing's ID.

The Hasher

- Idea is to compute the FS path from something you already know.
- Big plus is that anything you write that needs to access the FS doesn't need to look up the path in a database.
- Dubious value since you probably had to look the object/thing you're doing this for in the database anyways.. but you get the idea...

- Example: Use the userid to form the multi-level "hash" into the filesystem. Take for example the first two digits as your top level directory, the second two as the subdirectories. So sticking with our userid above we'd get a path like:

- /data/76/34/76340

- Downside is you can end up building stupid logic around the thing to handle low ids (where does user "46" go?) or end up padding stuff, all of which is ugly.

- A fancier alternative to this is using something like a MD5 hash (which you probably also already have for sessions) and that works well, is easy to implements, tends to give you better distribution "for free", and looks sexy to boot:

```
# echo "76340" | md5

e7ceb3e68b9095be49948d849b44181f

gives us:

/data/e7/ceb/76340
```

Downsides of the MD5-style

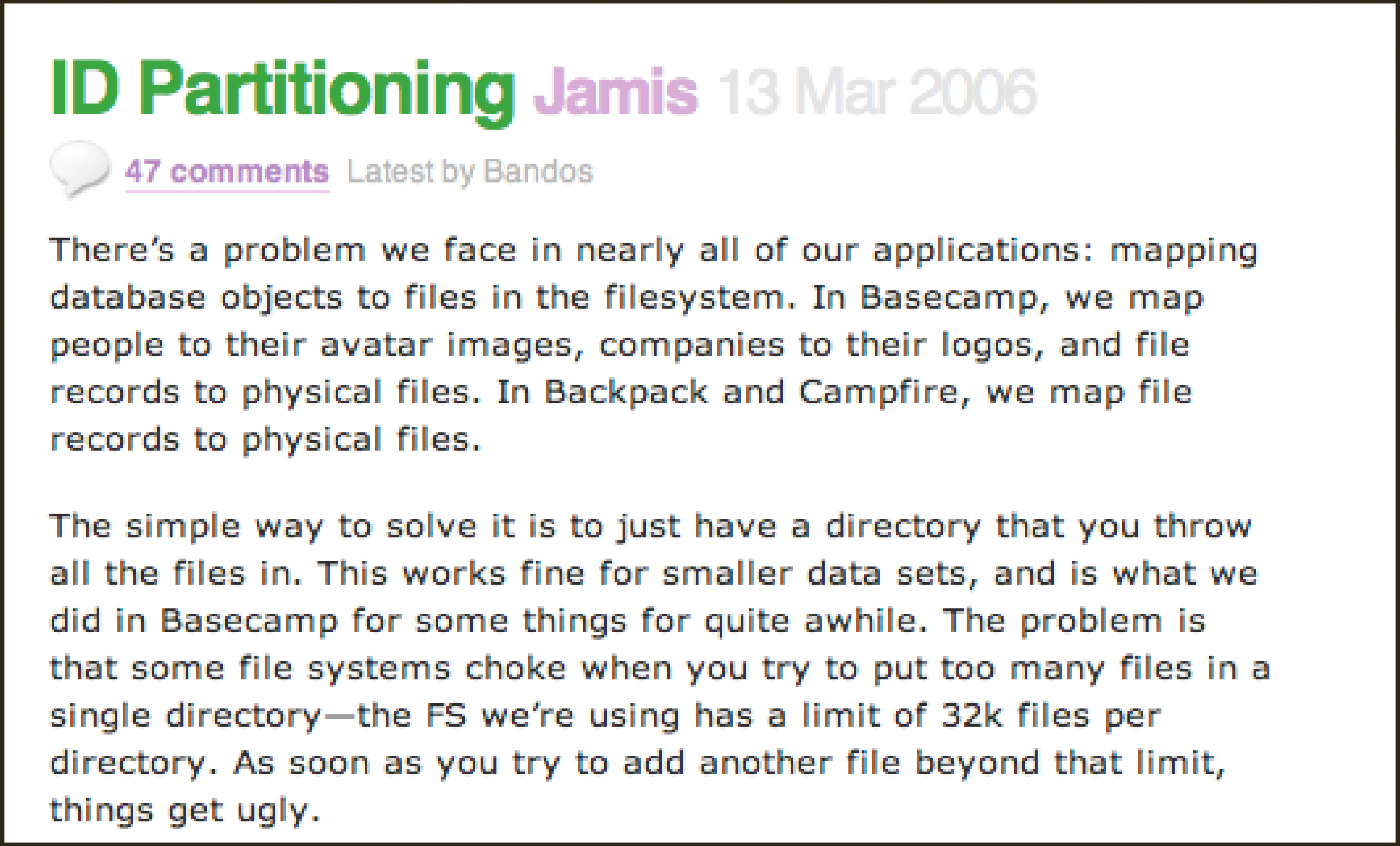
- Distribution is still unpredictable
- Watch your crypt()-style implementation cause it might output characters you need to escape!
- You can't compute it in your head

But

- The attractiveness of using some sort of computed hash will mostly depend on what sort of ID structure you already have, or or planning to use.
- Some are very friendly to simple hashing, some are not.
- So think “friendly”

Jamis does something like this

- http://www.37signals.com/svn/archives2/id_partitioning.php



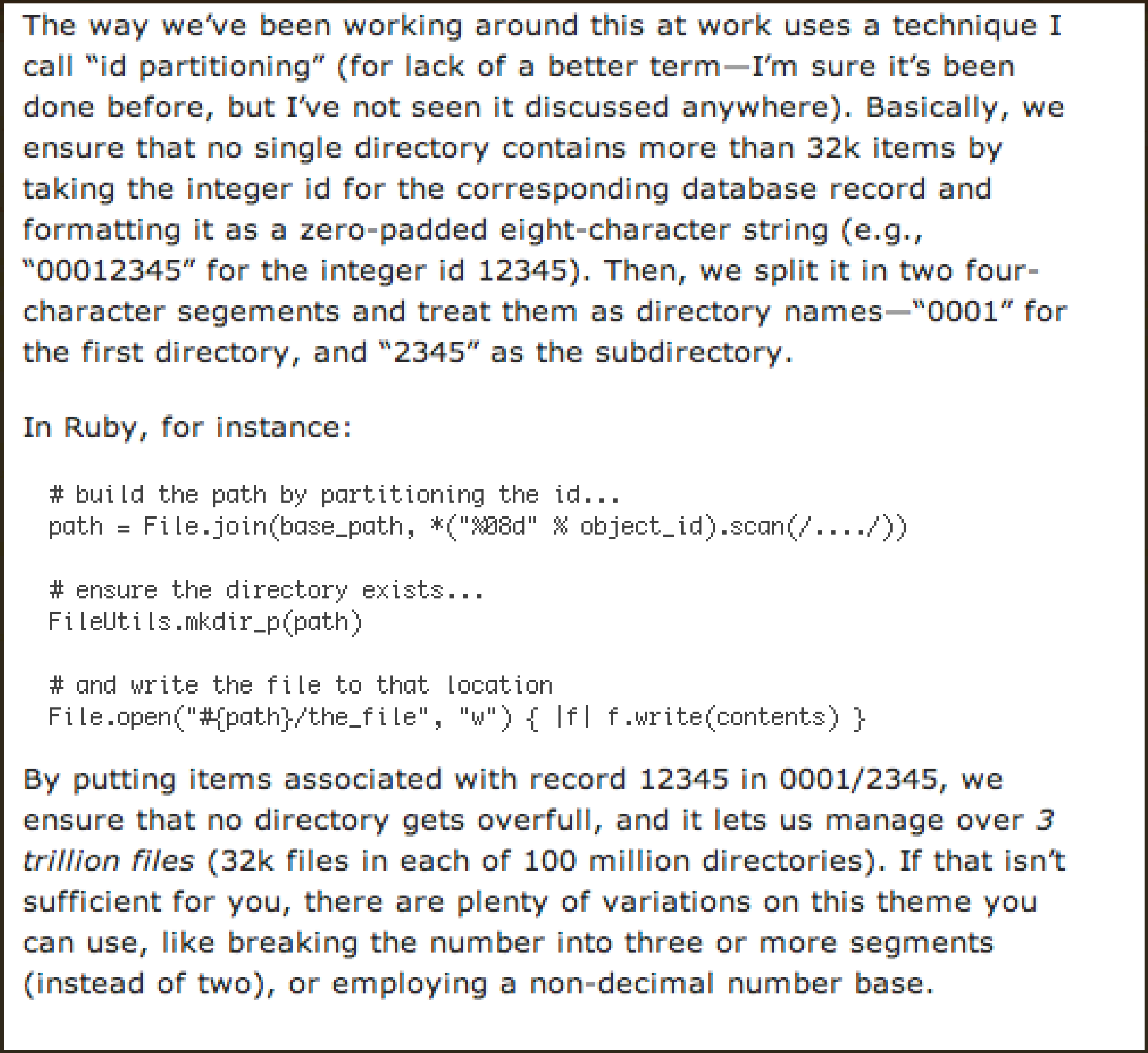
ID Partitioning Jamis 13 Mar 2006

47 comments Latest by Bandos

There's a problem we face in nearly all of our applications: mapping database objects to files in the filesystem. In Basecamp, we map people to their avatar images, companies to their logos, and file records to physical files. In Backpack and Campfire, we map file records to physical files.

The simple way to solve it is to just have a directory that you throw all the files in. This works fine for smaller data sets, and is what we did in Basecamp for some things for quite awhile. The problem is that some file systems choke when you try to put too many files in a single directory—the FS we're using has a limit of 32k files per directory. As soon as you try to add another file beyond that limit, things get ugly.

Joyent



The way we've been working around this at work uses a technique I call "id partitioning" (for lack of a better term—I'm sure it's been done before, but I've not seen it discussed anywhere). Basically, we ensure that no single directory contains more than 32k items by taking the integer id for the corresponding database record and formatting it as a zero-padded eight-character string (e.g., "00012345" for the integer id 12345). Then, we split it in two four-character segements and treat them as directory names—"0001" for the first directory, and "2345" as the subdirectory.

In Ruby, for instance:

```
# build the path by partitioning the id...
path = File.join(base_path, *("%08d" % object_id).scan(/..../))

# ensure the directory exists...
FileUtils.mkdir_p(path)

# and write the file to that location
File.open("#{path}/the_file", "w") { |f| f.write(contents) }
```

By putting items associated with record 12345 in 0001/2345, we ensure that no directory gets overfull, and it lets us manage over *3 trillion files* (32k files in each of 100 million directories). If that isn't sufficient for you, there are plenty of variations on this theme you can use, like breaking the number into three or more segments (instead of two), or employing a non-decimal number base.

Recap

- Use **DNS**
- Great load balancers
- Event-driven mongrels
- A relational database isn’t the only datastore: we use LDAP, J-EAI, file system too
- A Rails process should only be doing Rails
- Static assets should be coming from static servers and then a CDN
- Go layer7 where you can: a rails process should only be doing one controller
- Federate and separate as much as you can

Required separations

- All DNS based
- Dynamic (domain.com)
- Static (assets1-4, assets5-8)
- Uploads (authenticated; uploadsX.domain.com)
  - You build a separate application
- Downloads (authenticated vs unauthenticated)
  - downloadsX.domain.com
  - 60 second URLS
- Administrative (admin.domain.com