

Off the Shelf Capabilities

github.com/rkhatibi/devops2012



Ramin Khatibi

- Sysadmin for sixteen years
- Currently at SnappyTV
- Yahoo, Netzero
- Half a dozen startups

SnappyTV™

Our Network Looks Like Yours

- At least a bit (maybe?)
- Data stores, Provisioning, Config Mgmt, Monitoring, Applications

SnappyTV™

SnappyTV

- Cloud based video editing
- Real time social media data consumption and analysis
- Video stream ingestion

SnappyTV™

(re)Use Services

- Built scalable video ingest in 40 hours of dev time
- Integrate existing systems and write very little code
- Still no free lunch

SnappyTV™

If You Remember One Thing

- Service reuse > code reuse
- Use the services you're heavily invested in to not write code

SnappyTV™

Which Cases Are Simple(r)

- End of day processing
- Heavy resource jobs
- Planned Work

SnappyTV™

What Do We Really Need?

- The minimum we need to cause fewer problems than we solve
- Your env may need less|more

SnappyTV™

Application Logic

- We use Rails w/ CMS-ish code
- Decides it needs resources
- and when
- and for how long

SnappyTV™

Track State

- We use Redis and Mysql
- Where in the chain are we
- Provides coarse locking

SnappyTV™

Workers

- We use Resque
- Checks out a job
- Hands it back
- Distributed

SnappyTV™

Provisioning

- We use Ruby Fog, AWS, AMIs
- Reliable > Speed
- How provisioned is provisioned?
- close is often good enough

SnappyTV™

Configuration

- We use Puppet w/ vcs module
- Last 5% of config
- No new code to live machines

Error Handling

- Work in progress (still)
- Defense in Depth
- Break work into discrete chunks
- Majority of new code

SnappyTV™

Acceptance

- We generate status.html
- Can we manually tell it worked?
- What do we do if it didn't?
- May need to live in multiple places

Monitoring

- We use Nagios, Statsd, collectd
- Stats are hard to collect after you've recycled the machines
- Don't forget trending

SnappyTV™

Decommissioning

- Anything we brought up needs to go down
- Missing resources usually worse than unknown ones

SnappyTV™

That Was 9 Things

- Mix, match, combine
- Does it reliably accomplish work for your system?

SnappyTV™

Corner Cases

- ... Suck!
- Architect around them if you can
- Less flexibility is usually simpler

SnappyTV™

If You Remember One Thing (again)

- Reuse your existing services
- Arrange them in interesting ways
- Go home at 5:05pm

SnappyTV™