Engineering Velocity: Continuous Delivery at Netflix
engineering + velocity

applying science and technology to designing and building speed into a system
Availability vs. Rate of Change

![Graph showing the relationship between Availability (in 9's) and Rate of Change. The graph is a line that slopes downward from left to right, indicating that as the rate of change increases, the availability decreases. The x-axis represents the rate of change, with values ranging from 0 to 1000, and the y-axis represents availability (in 9's), with values ranging from 6 to 0.](image-url)
Shift the Curve

Availability (in 9's)
0
1
2
3
4
5
6

Rate of Change
0
10
100
1000
10000
Netflix Culture: Freedom & Responsibility

http://www.slideshare.net/reed2001/culture-1798664
Manager’s Role

Context, not Control

Loosely coupled, Tightly aligned

And hire well!
Get out of the Way
Freedom to Innovate
Support Experimentation

How We Built a Predictive Autoscaling Engine

Support Independent Paths of Exploration
Don’t Prematurely Optimize!
Blameless Culture
Developers Deploy Their Code

Run What You Wrote

• Rapid Innovation
• Rapid Detection
• Rapid Response

= Freedom + Responsibility
Support with Tools
Jenkins Job DSL

Configuration as Code

Groovy Script

Scripts go in Version Control

http://www.slideshare.net/quietryan/configuration-as-code
Aminator

Create AMI from Base AMI

Image contains service and everything needed to run it

Unit of Deployment for Test and Prod

Abstracts Cloud Details

http://techblog.netflix.com/2013/03/ami-creation-with-aminator.html
Asgard

Deploys Netflix to the Cloud

Red/Black push

Developed to address delays in rollback

http://www.infoq.com/presentations/asgard
Red/Black Push

- Scale up new instances
- Run canary analysis
- Turn on traffic to new ASG
- Turn off traffic to old ASG
- Wait ... analyze ... continue
Workflow

Continuous Delivery Engine

Judges between Stages

Represent Best Practices

One Click Deployment?
Regional Isolation

Limit Impact of Human Error

- Stagger Deployments?
- Canary Testing per Region?

Know your Service!
Multi-Region Consistency

Build Tooling to:

- Schedule Deployments
- Prefer Off-Peak
- Choose Next Available Region
- Provide Visibility by Region
Simian Army

- Chaos Monkey
- Latency Monkey
- Conformity Monkey
- Janitor Monkey
  (and more)

http://www.infoq.com/presentations/netflix-resiliency-failure-cloud
Chaos Monkey

Kills Running Instances

- Simulates failures inherent to running in the cloud
- In Production
Latency Monkey

Introduces Latency between services
Have Deployments Diverged?

- Balance Regional Consistency with Regional Isolation
- Build Best Practices into Tooling and Reporting
Janitor Monkey

Reduce Cognitive Load and Cost

• Remove unused instances
• Uniform way to clean up
Shifting the Curve with Tooling

• Value Self-Service

• Test Everywhere

• Awareness of Multiple Regions

• Best Practices Represented in Tooling

• Recover Quickly and Easily

• Be Cloud Native
Shifting the Curve with Culture

- Context not Control
- Freedom to Experiment
- Blameless Culture
“As the number of applications and the scale of the campaign's AWS infrastructure use climbed, the DevOps team shifted to using Asgard—an open-source tool developed by Netflix to manage cloud deployments.”

_ArsTechnica, November 2012_
Thanks!

Dianne Marsh (@dmarsh)
dmarsh@netflix.com