

# CPSC 3720

## Lesson 39



Connie Taylor  
Professor of Practice



# CUSports – Review Schedule



REVIEW DATE	TEAMS
NOV 18	9 – Gibby Gang – Notifications 
	5 – Rick Roll – User Accounts 
NOV 20	1 – Da Bagel – Inventory 
	6 – Constant Tailors – Promotions 
NOV 23	4 – image(4).png – Pricing 
	3 – Chaos- Payment 
	2 – Dynamic Devs- Search 

# Before DevOps

## Development

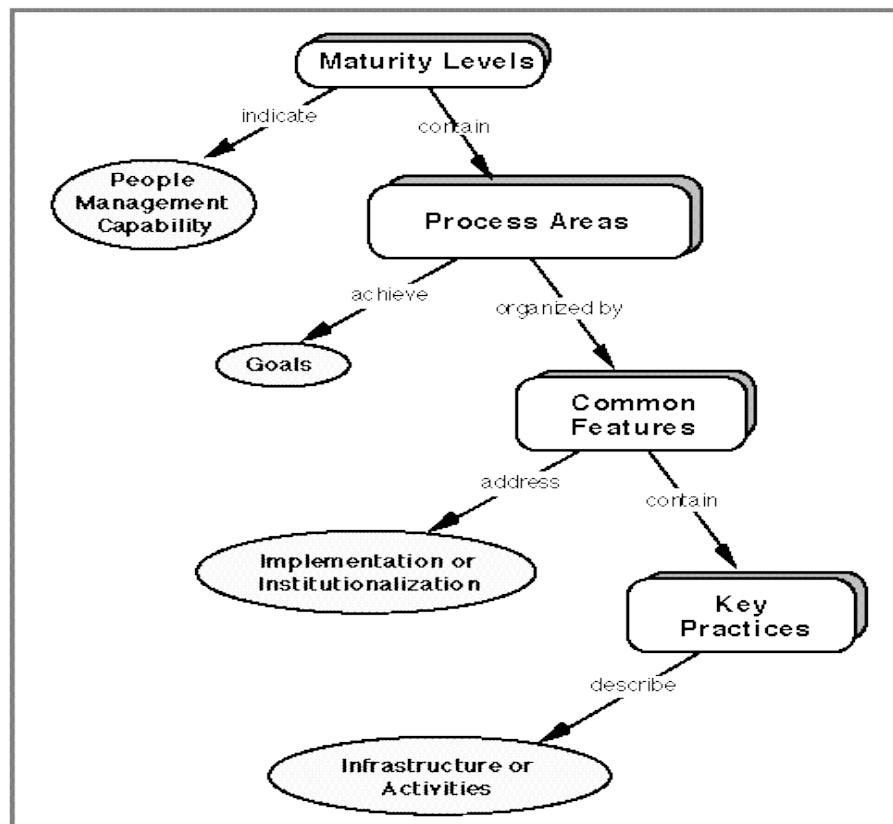
- SDLC: Waterfall, Agile, etc.
- Little/No ability to touch production environments
- Testing focus pre-GA exclusively
- Build and test environments owned by development teams
- **CMM** focus for tracking improvement

## IT/OPS

- Separate IT/Ops organization
- Get to steady state – “don’t touch anything”
- Mostly internal owned environments
- Heavy focus on governance
- **ITIL** as a process focus with SLAs

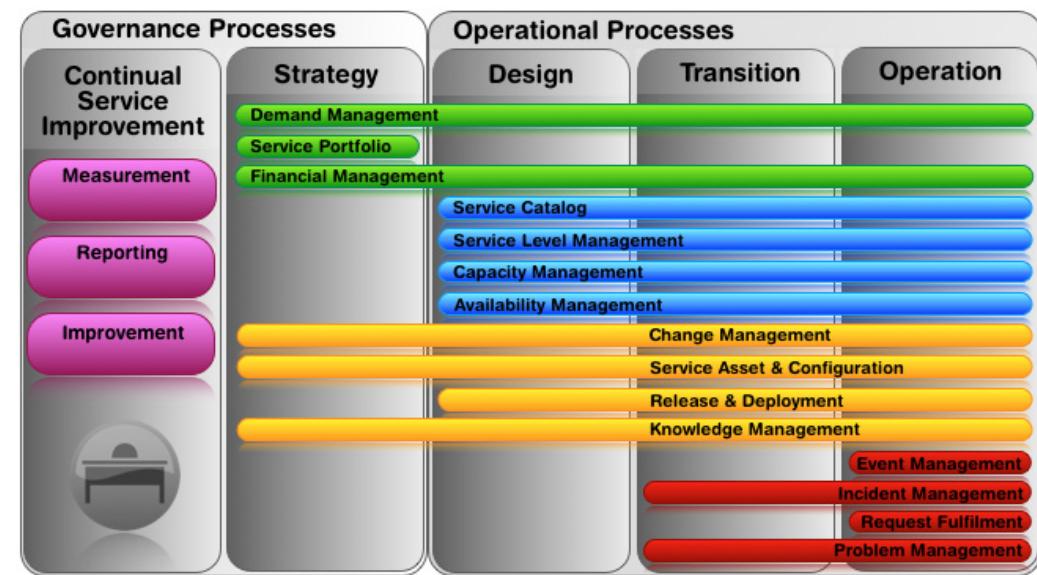
# What is CMM?

- CMM ≡ Capability Maturity Model (developed by Carnegie Mellon SEI in 1986)
- A framework that describes the key elements of an effective software process. It's an evolutionary improvement path from an immature process to a mature, disciplined process.
- Maturity Levels
  1. **Initial Level**- Adhoc, Unpredictable , Lack of Management.
  2. **Repeatable Level**- Encourages repetition of successful practices
  3. **Defined Level**- Identifies best practices
  4. **Managed Level**- Sets quantitative and quality goals
  5. **Optimizing Level**- Removal of waste, continuous improvement.



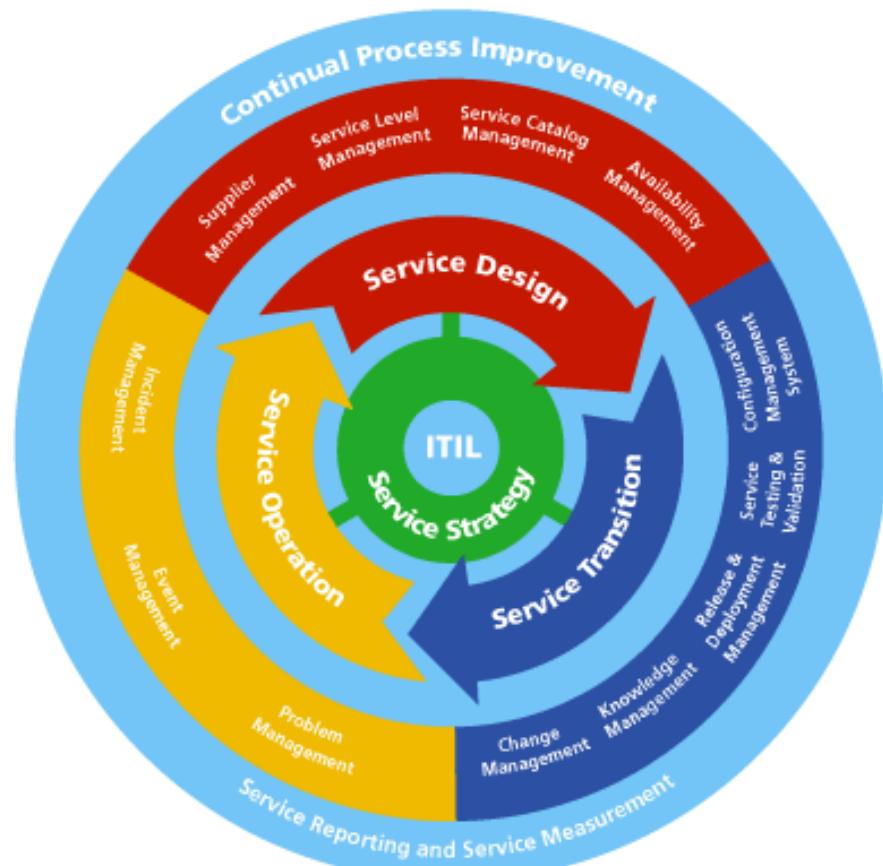
# What is ITIL?

- ITIL ≡ “Information Technology Infrastructure Library”; from UK and started in 1980
- Think of it as an ‘instruction manual’ for effective IT Service
- IT organization must document Process, Roles, Ownership
- Measuring/Metrics driven with SLAs for customer and users
- 5 stages of an IT “service” with multiple processes per stage
- Some key processes:
  - Change Management
  - Asset Management
  - Incident and Problem Management
- Help Desk guidance/measures through ITIL



# IT Service Lifecycle Stages

- **Service strategy:** Service gets approved
- **Service transition:** Service is tested
- **Service design:** Service is designed
- **Service operation:** Coordinate and carry out activities and processes required to deliver the service and manage them at agreed levels
- **Continual Service Improvement (CSI):** Continually align and re-align IT services to changing needs



# Emergence of DevOps

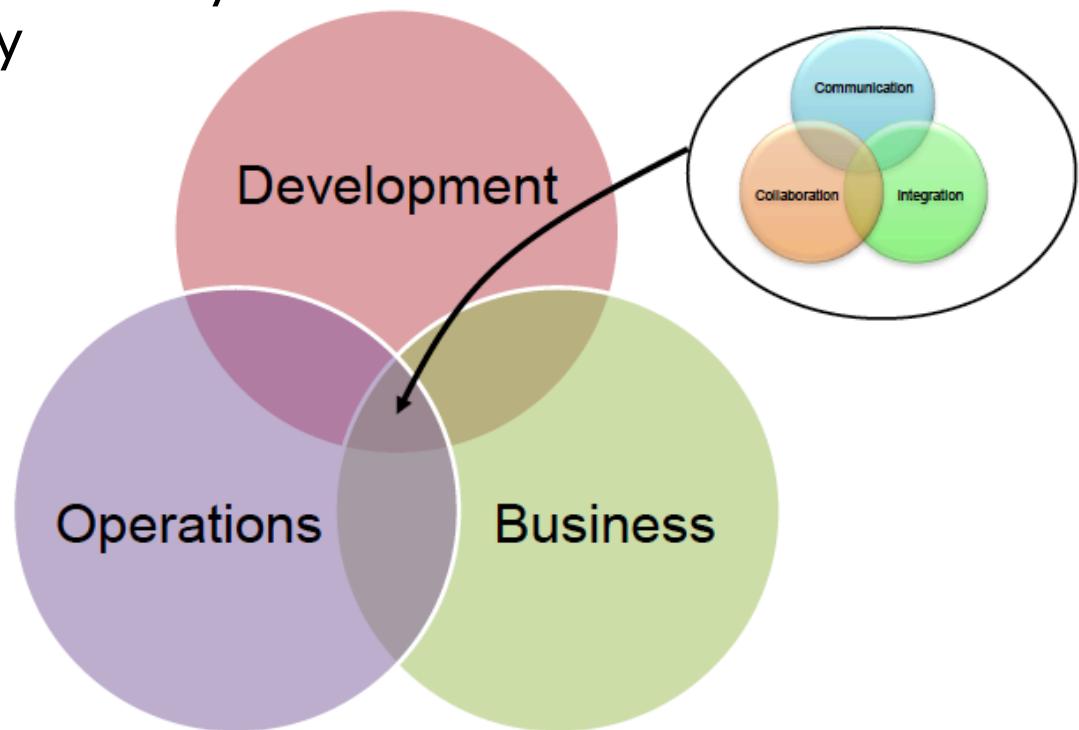
- A new bar has been set: Amazon, Facebook, Uber, etc.
- Cloud infrastructure and movement to SaaS solutions
- Customers expect that Dev/IT will:
  - Rapidly deliver new features that delight our customers
  - React to market changes quickly
  - Deliver a deep, direct engagement with the system and data
  - Ensure meeting regulatory and security requirements as needed

*“Keeping up with the rate of technological change is essential for [retail] organizations in these competitive environments who must keep demanding customers happy and satisfied while delivering consistent revenues to keep stakeholders satisfied. Retail may be the perfect example of technology delivering value...”*

**2019 Accelerate State of DevOps**

# Emergence of DevOps

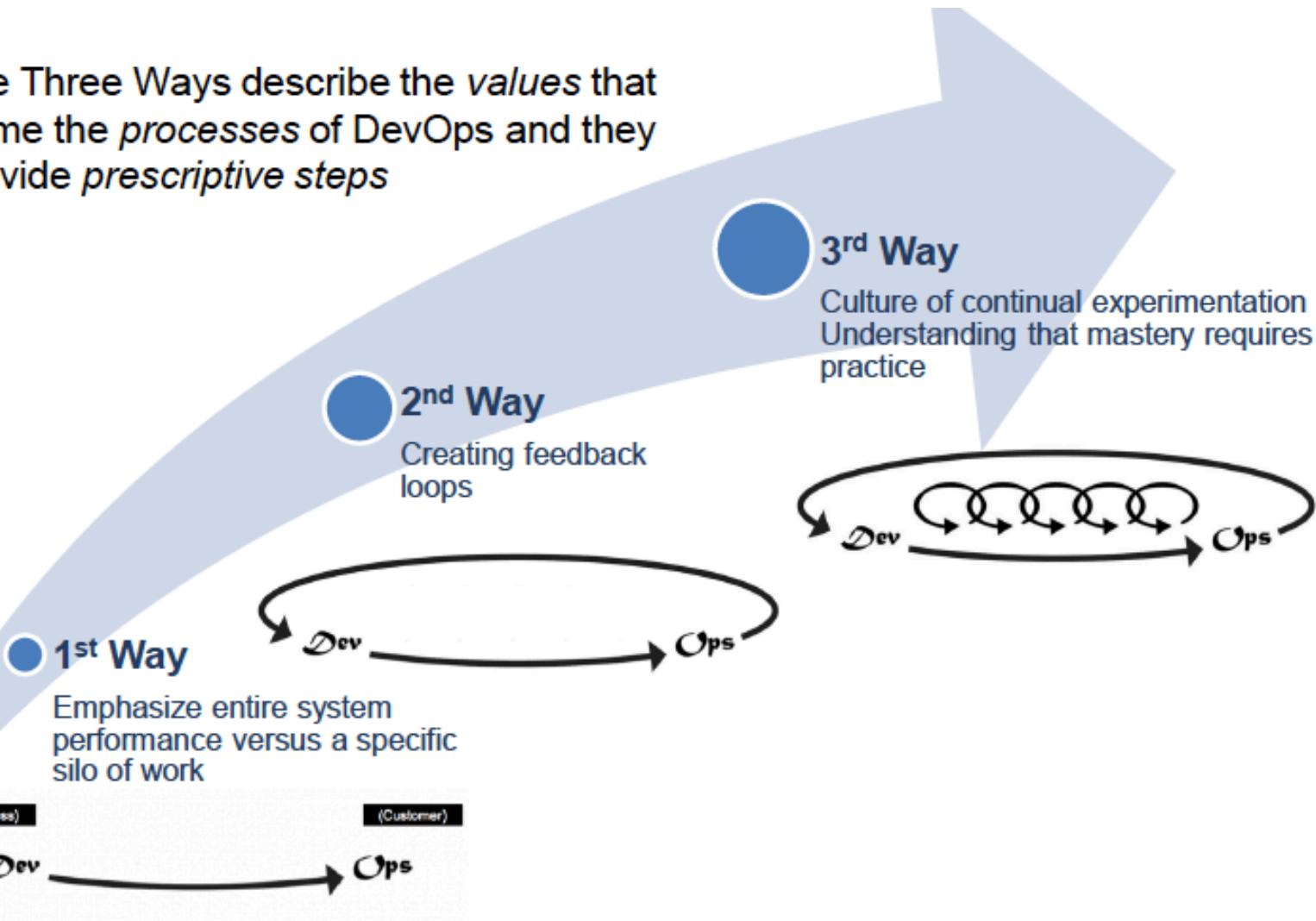
- DevOps is the practice of operations and development engineers participating together in the entire service lifecycle, from design through the development process to production support that enable:
  - Shorter development life cycles
  - Continuous delivery
  - High quality



# The Three ways of DevOps

## by Gene Kim

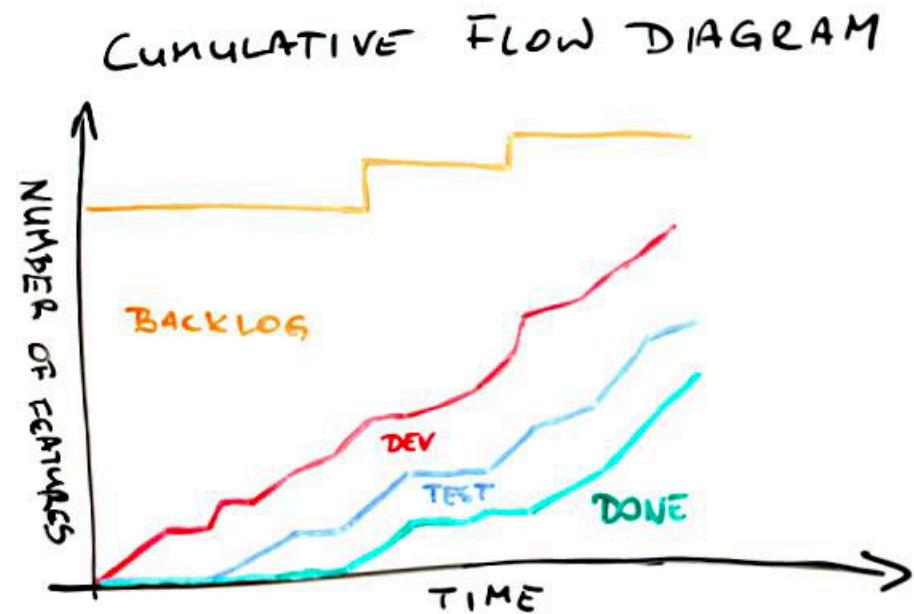
The Three Ways describe the *values* that frame the *processes* of DevOps and they provide *prescriptive steps*



# The First Way: Always Seek to Increase Flow

- Reduce work in progress (WIP)
- Reduce batch size of deliveries
- Reduce variation in size of work items

Deliver often –  
and get really good at it



“Work moving backwards, or standing still, is almost always indicative of problems that need to be solved, and will span people, process and technology.” –Gene Kim

# DevOps Practices – First Way

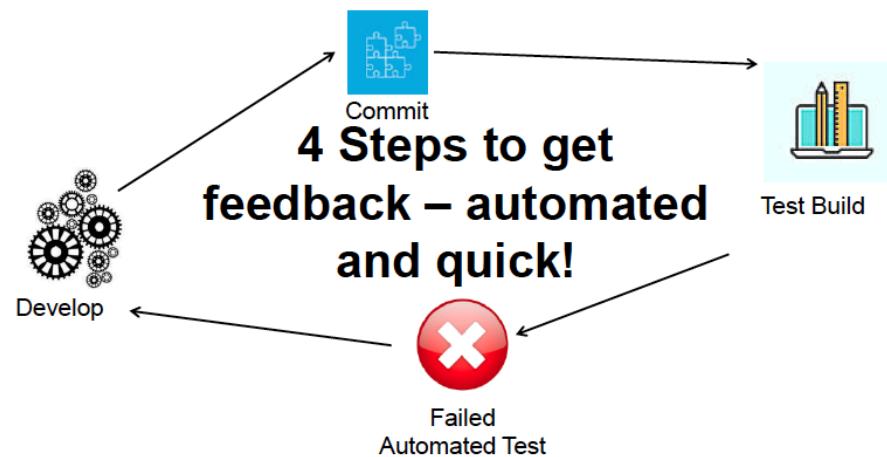
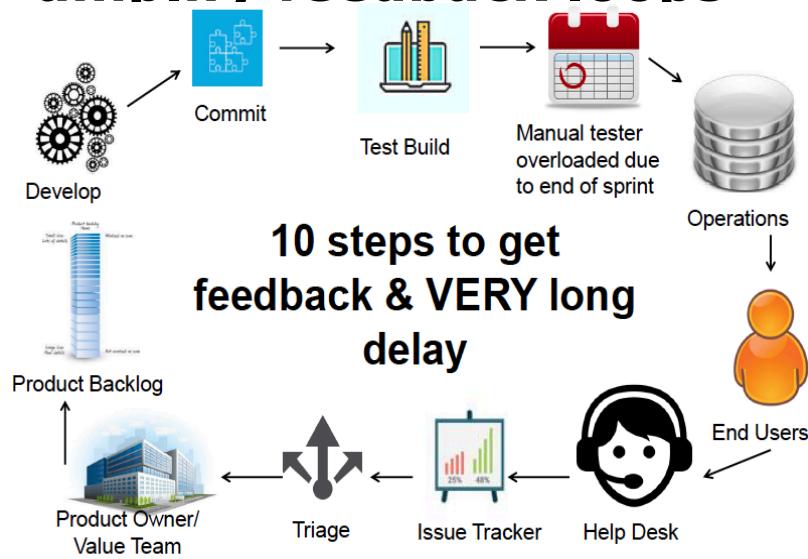
- **Infrastructure as Code (IaC)**
- **Continuous Integration and Deployment**
- **Automated Testing**
- **Self Service Environments**
- **Automated Environment Provisioning and De-Provisioning**



# The Second Way: Amplify Feedback Loops

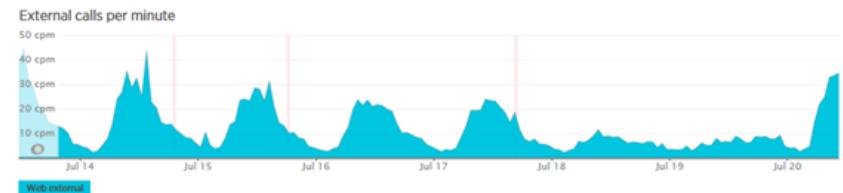
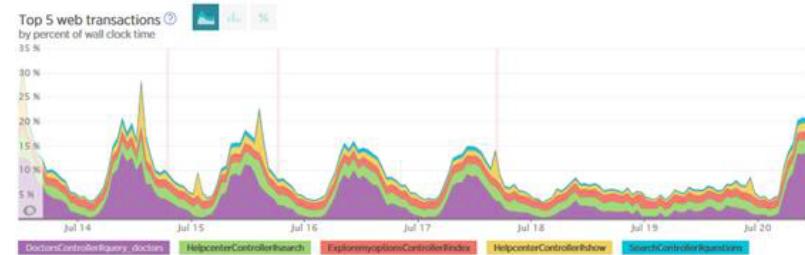
- Shorten and amplify “right to left” feedback loops
- Use feedback to create even higher quality at the Source
- Create and embed knowledge where it is needed to provide immediate feedback
- Understand needs of all customers, internal and external, and respond to their feedback

**The goal of any process improvement is to shorten and amplify feedback loops**



# DevOps Practices – Second Way

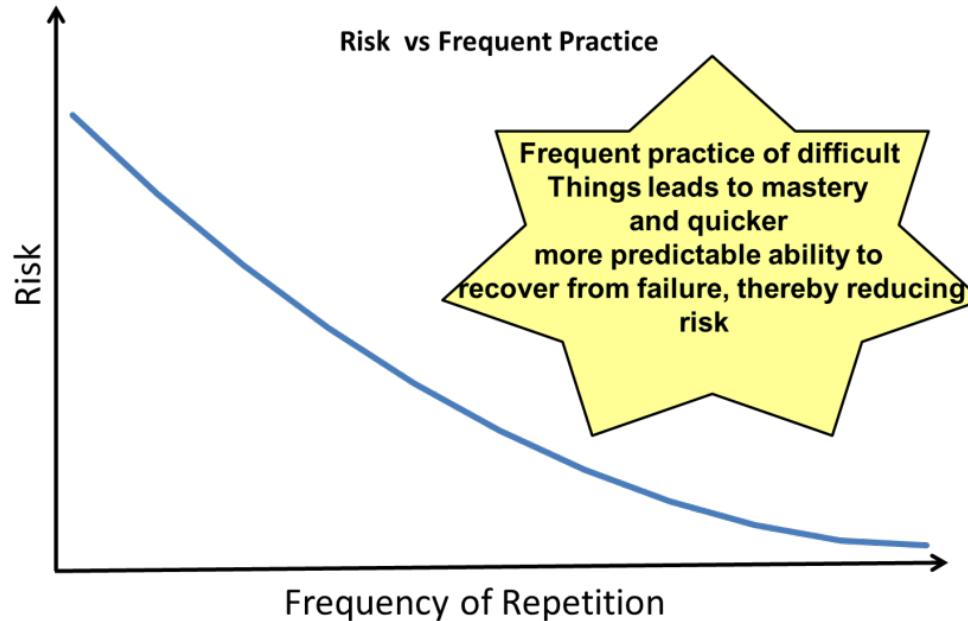
- Feature Flags
- App Performance Monitoring
- Load Testing & Auto-Scale
- Availability Monitoring
- Hypothesis Driven Development
  - Testing in Production
  - Fault Injection
  - Usage Monitoring/User Telemetry



# The Third Way: *Culture of Improvement*

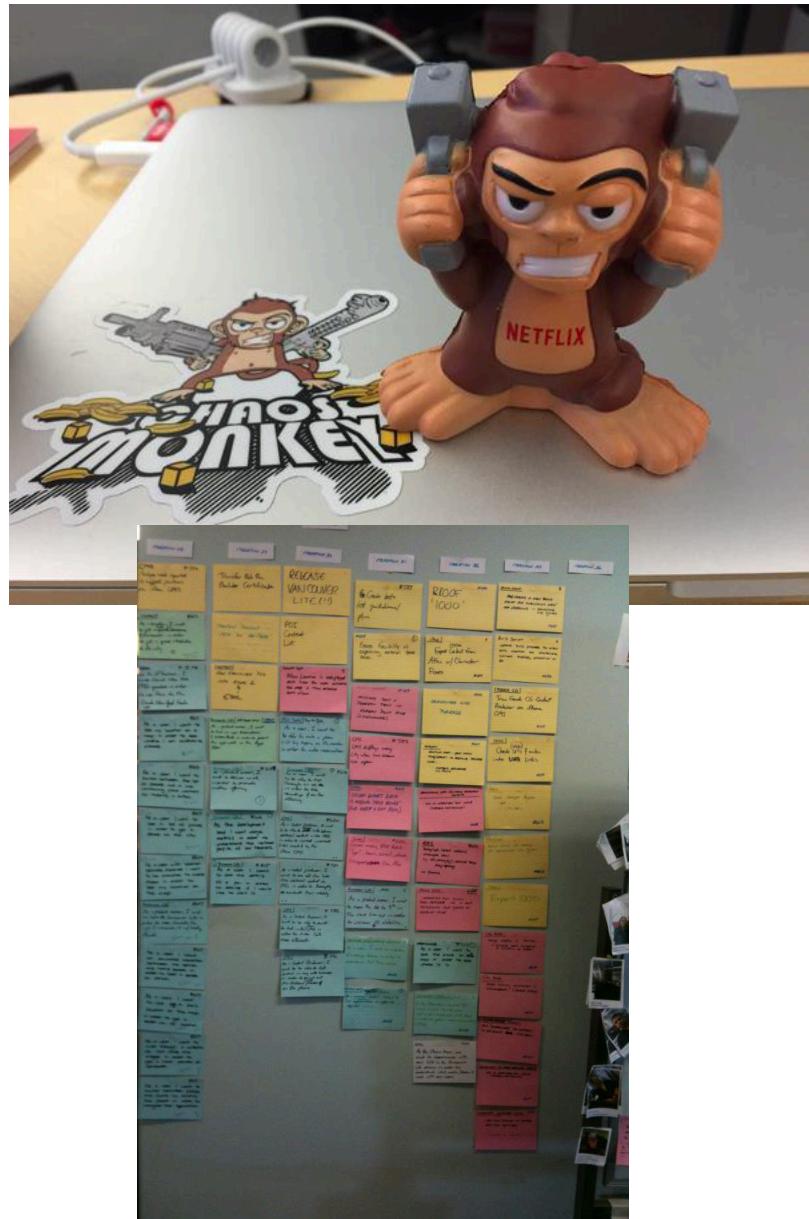
- Continual experimentation, taking risks, and learning from failure
- Understanding that repetition is the prerequisite to master
- DevOps is **Culture** and not – a tool, a role, or a team

**DevOps requires a culture of operations and development engineers participating together in the entire service lifecycle**



# DevOps Practices – Third Way

- **Inject Failures for Practice (Chaos Monkey)**
- **Allocate time for improvements (cruft!). Along with regular work plan for:**
  - Technical debt
  - Unplanned work
  - Experiments
  - Learning backlog
- **Invest in Automated Testing that continually improves!**
- **Invest in Tooling that continually improves!**



## Next Up: Measuring Success