Devops Workshop
(Section 1)

John Willis
@botchagalupe
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

This is a human-readable summary of (and not a substitute for) the license. Disclaimer.

You are free to:

**Share** — copy and redistribute the material in any medium or format

**Adapt** — remix, transform, and build upon the material

The licensor cannot revoke these freedoms as long as you follow the license terms.
Section 1 - Introduction to Devops
Why DevOps?
Why Devops?

▪ There is no single prescriptive definition/model of Devops

  ▪ Arguing/Debating the definition wastes time
  ▪ Instead learn to understanding anti-patterns
  ▪ Instead learn best practices and patterns from other practitioners
  ▪ Strive to continually learn and understand improvement
Part the role of DevOps is help other people do DevOps. It’s not just for you, it’s for other fellow travelers.

Dr Richard Cook
Understanding Improvement
Understanding Improvement
Understanding Improvement

The consequences of failure have never been greater…
Understanding Improvement

- High Performance Organizations
  - Amazon, Google, Facebook, Etsy, and Netflix routinely deploy 100’s a times a day to production.
  - Some organizations have engineers deploy on their first day of work.
  - Most organizations struggle deploying more than twice a year.
We assert that the Three Ways describe the values and philosophies that frame the processes, procedures, practices of DevOps, as well as the prescriptive steps.

Gene Kim
Devops Taxonomies

- CAMS
  - Culture
  - Automation
  - Measurement
  - Sharing

- The Three Ways
  - The First Way
  - The Second Way
  - The Third Way
Devops Taxonomies

• CAMS
  • Culture
  • Automation
  • Measurement
  • Sharing

• The Three Ways
  • The First Way
  • The Second Way
  • The Third Way
Devops Taxonomies

• CAMS
  • Culture
  • Automation
  • Measurement
  • Sharing

• The Three Ways
  • The First Way
  • The Second Way
  • The Third Way
Devops Taxonomies

• CAMS
  • Culture
  • Automation
  • Measurement
  • Sharing

• The Three Ways
  • The First Way
  • The Second Way
  • The Third Way
The Three Ways of DevOps

- The Three Ways of DevOps
  - The First Way - Accelerate Flow
  - The Second Way - Amplify Feedback
  - The Third Way - Continuous Learning
The Three Ways of Devops

Source: Damon Edwards - DTO Solutions
The Three Ways of Devops

- The First Way
  - Left to Right
  - System Thinking
  - Increased Visibility
  - Just in Time
  - Shorten Lead Time
The Three Ways of DevOps

- The Second Way
  - Right to Left
  - Shorten Feedback Loops
  - Learn Faster
  - Fix Defects Faster
  - Embedding Knowledge
The Three Ways of Devops

- The Third Way
  - Full Cycle
  - Continual Experimentation
  - Learning from Failure
  - Repetition and Practice
  - Increase Resilience
Empirically, Academically and Industrialized Evidence

- Better Efficiency (Faster)
- Better Quality (Resilience)
- Better Security (Safer)

2018 State of DevOps
Empirically, Academically and Industrialized Evidence

- Better Efficiency (Faster)
- Better Quality (Resilience)
- Better Security (Safer)

2018 State of DevOps
Empirically, Academically and Industrialized Evidence

- Better Efficiency (Faster)
- Better Quality (Resilience)
- Better Security (Safer)

2018 State of DevOps
Conventional Wisdom

Fast

“Pick Two!”

Good ——— Cheap
Conventional Wisdom

"Pick two!"

Fast

Good

Cheap
The Possible

Generative Behavior

“Must Have All Three!”

Ron Westrum Topology Model

<table>
<thead>
<tr>
<th>Pathological (power-oriented)</th>
<th>Bureaucratic (rule-oriented)</th>
<th>Generative (performance-oriented)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low cooperation</td>
<td>Modest cooperation</td>
<td>High cooperation</td>
</tr>
<tr>
<td>Messengers shot</td>
<td>Messengers neglected</td>
<td>Messengers trained</td>
</tr>
<tr>
<td>Responsibilities shirked</td>
<td>Narrow responsibilities</td>
<td>Risks are shared</td>
</tr>
<tr>
<td>Bridging discouraged</td>
<td>Bridging tolerated</td>
<td>Bridging encouraged</td>
</tr>
<tr>
<td>Failure leads to scapegoating</td>
<td>Failure leads to justice</td>
<td>Failure leads to enquiry</td>
</tr>
<tr>
<td>Novelty crushed</td>
<td>Novelty leads to problems</td>
<td>Novelty implemented</td>
</tr>
</tbody>
</table>
The Possible

Ron Westrum Topology Model

Generative Behavior

“Must Have All Three!”

Speed  Resilience

Pathological (power-oriented)  Bureaucratic (rule-oriented)  Generative (performance-oriented)

- Low cooperation  Modest cooperation  High cooperation
- Messengers shot  Messengers neglected  Messengers trained
- Responsibilities shirked  Narrow responsibilities  Risks are shared
- Bridging discouraged  Bridging tolerated  Bridging encouraged
- Failure leads to scapegoating  Failure leads to justice  Failure leads to enquiry
- Novelty crushed  Novelty leads to problems  Novelty implemented
Organizational culture was one of the strongest predictors of both IT performance and the overall performance of the organization.
The Convergence, History, and Value of DevOps
The Convergence of DevOps
By John Willis

I’d like to describe how DevOps is the culmination of three amazing and significant movements. This is what we’re putting into the DevOps Cookbook.

So the saying goes “If I could have a nickel for every time I have heard the expression “Cloud? We’ve been doing that for 20 years!” I would be a …”

So goes DevOps. The funny thing is that most people are right in saying “we’ve been doing DevOps for years.” That’s mainly because there is not good canonical definition for DevOps. However, the purpose of this post is not to debate the definition of DevOps; moreover, it’s to help describe and understand the movement’s history.

MOST POPULAR POSTS
Where To Learn More About Concepts In “The Phoenix Project” (Part 1)
The Convergence of DevOps
Devops Convergence

- Why/How Devops?
  - Devops Direct Drivers
  - Devops Indirect Influencers
  - Current Influences
Devops Convergence

- Devops Direct Drivers
  - Open Source
    - Monitoring
    - Configuration Management
  - Continuous Delivery
    - Integration, Delivery and Deployment
Devops Convergence

- Original Devops “Indirect” Influencers
  - Lean
  - Agile
  - Webscale
Devops Convergence

- Current Influences
  - Resilience Engineering and Human Factors
    - Dr Dekker, Dr Cook and Dr Woods
  - Learning Organizations
    - Peter Senge (Fifth Discipline)
  - Organizational Change Management
    - John Kotter
  - Psychology
    - Dr Maslach (Burnout)
Lean
- Module 1: The Phoenix Project
- Module 2: Goldratt
- Module 3: Deming
- Module 4: Lean
- Module 5: Safety Culture
- Module 6: Learning Organizations
- Module 7: Lean, Safety and Learning
- Module 8: Case Studies
- Module 9: Conclusion
Module 1: The Phoenix Project
Module 2: Goldratt
Module 3: Deming
Module 4: Lean
Module 5: Safety Culture
Module 6: Learning Organizations
Module 7: Lean, Safety and Learning
Module 8: Case Studies
Module 9: Conclusion
Devops Convergence

- Lean Thread
  - TPS - Toyota Production Systems (1956)
  - Lean Manufacturing (1988)
  - Lean Software Development (2006)
  - Lean IT (2010)
  - Lean Startup (2011)
  - Lean Enterprise (2014)
Dr Steven Spear

Toyota Production Systems is a community of scientists performing continual experiments.
- The Machine That Changed The World
  - Womack, Jones and Roos - 1991
- Learning To See
  - John Shook, Mike Rother - 1999
- Decoding The DNA Of The Toyota (TPS)
  - Dr Spear - 1999
- Lean Thinking
  - James Womack and Daniel Jones - 2003
- The Toyota Way
  - Dr. Jeffrey Liker - 2003
A total of 1.5 million Prius family members have been sold in the US by 2013, representing a 50.1% market share of total hybrid sales. Versus Chevy Volt only 50,000 by 2013.

### Comparing the Prius and the Volt

<table>
<thead>
<tr>
<th></th>
<th>Toyota Advantage</th>
<th>Toyota Prius</th>
<th>Chevy Volt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Cost</td>
<td>61%</td>
<td>$24,200</td>
<td>$39,900</td>
</tr>
<tr>
<td>Units Sold</td>
<td>13x</td>
<td>23,294</td>
<td>1,788</td>
</tr>
<tr>
<td>In-House Production</td>
<td>50%</td>
<td>27%</td>
<td>54%</td>
</tr>
<tr>
<td>Plant Suppliers</td>
<td>16% (10x per)</td>
<td>125</td>
<td>800</td>
</tr>
<tr>
<td>Firm-Wide Suppliers</td>
<td>4%</td>
<td>224</td>
<td>5,500</td>
</tr>
</tbody>
</table>
Devops Convergence

Your Toyota is My Toyota

"If I suspect a problem I pull the Andon cord and the whole production line stops immediately, and it doesn’t start again until the issue has been resolved. We pulled it thousands of times and it’s good because it makes everyone personally responsible for producing the highest quality cars."

Bride Tucker
Team Manager, Trim Assembly, Toyota Burnaston, UK

3
Devops Convergence
Devops Convergence
Devops Convergence

- Lean Startup Thread
  - MVP
  - Pivot
  - Continuous Deployment
  - Actionable Metrics
  - No Vanity Metrics
  - 5 Why’s
  - Split Testing
Agile
DevOps Convergence

- Agile Thread
  - Scrum (1995)
  - Agile Manifesto (2001)
  - Kanban (2008)
  - Agile Conference (2008)
Devops Convergence

- Agile 2008 - Agile Infrastructure BoF
WebScale
Devops Convergence

- Web Scale
  - CFEngine (1995)
  - Puppet (2006)
  - O’Reilly Radar (2007)
  - Opscode Chef (2009)
  - Velocity Conference (2009)
Operations is a competitive advantage... (Secret Sauce for Startups!)

by Jesse Robbins | @jesserobbins | October 23, 2007
Devops Convergence

- O’Reilly Velocity 2009 (Devops Milestones)
  - John Allspaw and Paul Hammond, "10+ Deploys Per Day"
  - Andrew Clay Shafer - Agile Infrastructure
  - Adam Jacob & Ezra Zygmuntowicz demo Chef & Nanite
Devops Convergence

- Devops Values...
  - No rock star mentality
  - Shared contributions
  - Healthy attitudes towards failure
  - Failures are leaning opportunities
  - The problem is the enemy
  - No blame games
  - No victims

- Devops Values...
  - Shared blame
  - 5 Why’s
  - Develop shared metrics
  - Focus on end goal
  - Alignment of Purpose
  - Shared goals/ slay the dragon
  - Aim, Goal, Why
Devops Convergence

- Organizations with Large Contributions to Devops
  - Netflix
    - Culture Deck
  - Etsy
    - Code as Craft
  - Google
    - Site Reliability Engineering
  - Spotify
    - Engineering Culture Videos
Devops Convergence

- Netflix Culture Deck Quotes

  The *actual* company values, as opposed to the *nice-sounding* values, are shown by who gets rewarded, promoted, or let go.

  Unlike many companies, we practice:

  *adequate performance gets a generous severance package*
Devops Convergence

- Devopsdays 2009 (Ghent)

The conference that brings development and operations together.
Devops Convergence

- Devopsdays 2009 (Ghent)

71 Devops Days in 2018
Patterns and Practices
Devops Practices and Patterns

- **Continuous Delivery**
  - Everything in version control
  - Small batch principle
  - Trunk based deployments
  - Manage flow (WIP)
  - Automate everything

- **Culture**
  - Everyone is responsible
  - Done means released
  - Stop the line when it breaks
  - Remove silos

itrevolution.com/devops-handbook
Devops Automated Deployment Pipeline

Source: Wikipedia - Continuous Delivery
KEEP CALM AND PULL THE ANDON CHORD
Devops Automated Deployment Pipeline

Google developer scale

- 30,000+ developers
- 45,000 commits per workday
- 20,000 code reviews per workday
- 1 billion files
- 9 million source files
- 2 billion lines of code
- 800,000 builds per day
- 150 million test cases run per day
- 2+ PB of build outputs per day

Melody Meckfessel, Senior Director of Engineering, Google Cloud Platform

@botchagalupe
Devops Automated Deployment Pipeline

Google developer scale

30,000+ developers
1 billion files
800,000 builds per day

45,000 commits per workday
9 million source files

20,000 code reviews per workday
2 billion lines of code

150 million test cases run per day

JFrog SwampUP Melody Meckfessel, Senior Director of Engineering, Google Cloud Platform

@botchagalupe
Amazon

- 11.6 second mean time between deploys.
- 1079 max deploys in a single hour.
- 10,000 mean number of hosts simultaneously receiving a deploy.
- 30,000 max number of hosts simultaneously receiving a deploy
Unicorns and Horses (Enterprises)

Shamelessly stolen and repurposed from: Pete Cheslock
Enterprise Organizations

- Ticketmaster - 98% reduction in MTTR
- Nordstrom - 20% shorter Lead Time
- Target - Full Stack Deploy 3 months to minutes
- USAA - Release from 28 days to 7 days
- ING - 500 applications teams doing devops
- CSG - From 200 incidents per release to 18
Patterns

- High Performance Organizations
  - Make work visible
  - Manage WIP
  - Manage Flow
  - Create high trust work environments
  - Learn and embrace failure
High Performance Organization Don’ts

- Don’t have planned outages
- Don’t have policy checklists
- Don’t have tool toil
- Don’t have knee jerk reactions
High Performance Organization Do’s

- Do create institutional vs tribal knowledge
- Do have psychological safety
- Do maniacally visualize
- Do systemically track outcomes
- Do have realistic technology roadmaps
Solutions

1. Make All Work Visible
2. Consolidate Work Management Systems
3. Eliminate Bottlenecks
4. Collaboration Hacks
5. Toyota Kata
6. Market Oriented Organized
7. Shift Left Auditors
CAMS as a Feedback Loop
Practices

- Continuous Delivery Principles
  - Build quality in
  - Work in small batches
  - Automate repeatable tasks
  - Pursue continuous improvement
  - Everyone is responsible
Practices

- Continuous Delivery Anti-Patterns
  - Incongruent testing and production environments
  - Testing takes too long
  - Manual regression and acceptance tests
  - Long lead times
  - High technical debt
  - Slow and hard to change
Practices

- Continuous Delivery Patterns
  - Everything starts in source control
  - Peer reviews (pull requests)
  - Automate everything
  - Trunk based deployment
  - Done means released
  - Stop the line