DevOps – an Agile Perspective (at scale)

by Brad Appleton
About the Presenter

Brad Appleton
Enterprise Agile & DevOps Engineering Manager
<brad@bradapp.net>
linkedin.com/in/bradappleton

 Agile/DevOps/Lean Leader & Coach
 DevOps/ADLM/SCM Solution Architect
 Leading Agile adoption & scaling for teams and organizations since 2000.
 Publications
  » Software Configuration Management Patterns (Addison-Wesley, 2002)
  » Agile CM Environments article series & blog (AgileConnection & CMCrossRoads, 2003-2014)
Why DevOps?

No idea what the fuss is all about. We've been working with Deaf OPS for years!

hey!
State of DevOps 2018

Elite performing IT organizations report:

- 46 TIMES MORE frequent code deployments
- 2,555 TIMES FASTER lead time from commit to deploy
- 7 TIMES LOWER change failure rate (changes are 1/7 as likely to fail)
- 2,604 TIMES FASTER time to recover from incidents
Highly evolved IT orgs are:

- More likely to use CM tools
- More likely to make monitoring & alerting configurable by teams (24x)
- More likely to reuse deployment patterns (23x)
- More likely to reuse testing patterns (27x)
- More likely to contribute to other teams tooling (44x)
- More likely to reuse testing patterns (44x)

Highly evolved IT orgs are:

- Normalize the technology stack
- Standardize and reduce variability
- Expand DevOps practices
- Automate infrastructure delivery
- Provide self-service capabilities
WHAT IS DEVOPS?

Culture & Mindset

Agile CM

Continuous Delivery

Dev & Ops Collaboration

Pipeline Automation Tools

Infrastructure as Code
What is DevOps?

A cross-disciplinary community of practice dedicated to the study of building, evolving and operating rapidly-changing resilient systems at scale. — Jez Humble

An IT mindset encouraging communication, collaboration, integration and automation among software developers and IT operations to improve the speed and quality of delivering software. — VersionOne

A set of practices and cultural changes — supported by the right tools — that creates an automated software delivery pipeline, enabling organizations to win, serve, and retain customers. — Forrester

A professional movement advocating a collaborative working relationship between Development & IT Operations, resulting in the fast flow of planned work, while simultaneously increasing the reliability, stability, resilience & security of the production environment. — Gene Kim
DevOps is ...

A professional cultural movement/philosophy/mindset emphasizing ...

- Continuous collaboration between development & operations
- Automated CI/CD pipelines, working in small-batches, with shorter lead-times (frequent deployment), and low failure-rates.
- Agile (coding & automation) practices applied to infrastructure, configuration, deploying/releasing, and monitoring.
Relationship between Agile & DevOps

Agile Development
- Iterative Development
- Scrum, Sprint, Stories
- Velocity

DevOps
- Continuous Integration
- Continuous Deployment
- IT Automation
- Application Management

Source: http://www.effectivepmc.com/devops
Relationship between Agile & DevOps

DevOps Perspective of Agile Development

Agile Developer’s Perspective of DevOps
DevOps is a direct result of applying Agile (values, principles & practices) into operations & infrastructure

- **Agile** Systems Administration & **Agile** (Web) Operations
- Transforming CD from **Agile** Principle into Practice
- **Agile** Testing Practices (TDD, BDD, ATDD, SBE)
- **Agile CM** Practices (Sw CM & Infrastructure CM)
- **Agile ALM** Tooling comprises most of a DevOps toolchain

As **Agile has expanded beyond software** and has become almost exclusively associated with Scrum, **DevOps has effectively “assimilated” all the technical Agile practices!**
The Three Ways of DevOps (Gene Kim)

- **The First Way** – System Flow
  - Understand and increase the flow of work (left to right)

- **The Second Way** – Amplify Feedback
  - Create short feedback loops that enable continuous improvement (right to left)

- **The Third Way** – Continuous Experimentation & Learning
  - Create a culture that fosters experimentation, risk-taking & learning from failure (repetition & practice is a prerequisite for mastery)
CALMS Model of DevOps

**Culture**
- Focus on People
- Embrace Change & Experiment

**Automation**
- Continuous Delivery
- Infrastructure as Code

**Lean**
- Focus on Producing Value for the User
- Small Batch-sizes

**Measurement**
- Measure Everything
- Show the Improvement

**Sharing**
- Open Information Sharing
- Collaboration & Communication
http://www.slideshare.net/CarmenDeArdo/interconnect-2016-continuous-visibility-dta3268-59336071
DevOps Practices / Disciplines

1. Use version control for all artifacts.
2. Automate your deployment process.
3. Use trunk-based development.
4. Implement continuous integration.
5. Implement continuous delivery.
6. Use loosely coupled architecture.
7. Architect for empowered teams.
DevOps Practices / Disciplines

- Configuration Management
- Continuous Integration
- Continuous (Automated) Testing
- (Agile) Infrastructure as Code
- Continuous Delivery/Deployment
- Continuous Monitoring
DevOps Principles & Culture:

1. Treat operations as a first-class citizen
2. Developers act as first responders to issues with the production system
3. Shorten the time between identification of a production issue and its repair
4. Shorten the time between code commit and code deploy
5. Minimize coordination to deploy releases
6. Stop and fix potential defects identified by continuous flow and monitoring
7. Enforce standardized processes to ensure predictable outcomes
8. Become a learning organization through continual feedback and action
The “DevOps Team” Antipattern

We do not believe in the existence of DevOps standalone teams—DevOps responsibilities must exist within the Agile teams. Just like a software developer or tester, a DevOps engineer or an engineer with DevOps skills must take ownership of the feature on which the team is working and work to make the software better for the end user.
Organizational Structures Used

<table>
<thead>
<tr>
<th>Structure</th>
<th>C-Suite</th>
<th>Management</th>
<th>Team</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cross-functional teams responsible for specific services or applications</td>
<td>18%</td>
<td>39%</td>
<td>36%</td>
</tr>
<tr>
<td>Dedicated DevOps team</td>
<td>15%</td>
<td>41%</td>
<td>46%</td>
</tr>
<tr>
<td>Centralized IT team with multiple application development teams</td>
<td>10%</td>
<td>24%</td>
<td>23%</td>
</tr>
<tr>
<td>Site reliability engineering team</td>
<td>28%</td>
<td>68%</td>
<td>81%</td>
</tr>
<tr>
<td>Service team providing DevOps capabilities (e.g. building test environments, monitoring)</td>
<td>18%</td>
<td>45%</td>
<td>48%</td>
</tr>
</tbody>
</table>
Evolutionary Path for DevOps Operating Model

Source: *Thinking Environments – Evaluating Organizational Models for DevOps to Accelerate Business and Empower Workers*

Source: *DevOps Team Topologies: What Team Structure is Right for DevOps to Flourish? --devopstopologies.com*
“DevOps, in a sense, is about setting up a value delivery factory – a streamlined, waste-free pipeline through which value can be delivered to the business with a predictably fast cycle-time” – Mark Schwartz, *The Art of Business Value*
A **DevOps toolchain** is a set or combination of tools that aid in the delivery, development, and management of applications throughout the **software development lifecycle**, as coordinated by an organization that uses **DevOps** practices.
Continuous Delivery

- Short lead times require more deployments
  - DevOps Governance

- Infrastructure Configuration Management
  - Infrastructure is under version control
  - Deployments are orchestrated

- Immutable Infrastructure (no snowflakes)

- More Testing
  - Performance and Security
  - Integration testing
  - Service Virtualization
Continuous Delivery vs. Continuous Deployment

https://blog.crisp.se/2013/02/05/yassalsundman/continuous-delivery-vs-continuous-deployment
Summary
Keys to DevOps

- Plan small/ fail fast/ deliver quickly
- Everything is under SCM
- Test & Automation
- People (Kaizen/ Quality Culture)
- Infrastructure under CM
DevOps Outcomes

- Improved deployment frequency;
- Faster time to market;
- Lower failure rate of new releases;
- Shortened lead time between fixes;
- Faster mean time to recovery
- Better employee engagement & leadership

Investment in DevOps and associated mindset is one of the top predictors of IT organizational performance!
**Related Terms & Trends**

- NoOps
- DevSecOps / SecDevOps
- TestOps
- ChatOps
- HugOps
- GitOps/DiffOps
- DataOps
- Antifragility
- Chaos Engineering
- Serverless
- Pipeline as Code
- Snowflake
- Configuration Drift
- Immutable / Phoenix Server
- Canary Release
- A/B Testing
- Blue-Green Deployment
- Feature Toggles / Dark Launch
- Service/Data Virtualization
- Containers/Containerization
- FaaS / Lambdas
- DevOps as a Service

[https://xebialabs.com/glossary/](https://xebialabs.com/glossary/)
Lean/Agile/DevOps is *a journey* (not a destination) that requires *real change* for most organizations!

- Don’t expect big changes to happen overnight!
- Your mileage will vary!
Recommended Books

- The Phoenix Project
- Continuous Delivery
- Infrastructure as Code
- Effective DevOps
- ACCELERATE

Recommended Books

- The DevOps Handbook
- LEAN ENTERPRISE
- Leading the Transformation
- Designing Delivery
- DevOps in the Modern Enterprise

Thank You!

ADDITIONAL INFORMATION / RESOURCES
DevOps – an Agile Perspective (at scale)

Appendix: Resources & Links
References

- puppet.com/resources/whitepaper/state-of-devops-report
- slideshare.net/dev2ops/the-history-of-devops-and-what-you-need-to-do-about-it
- slideshare.net/jallspaw/10-deploys-per-day-dev-and-ops-cooperation-at-flickr
- slideshare.net/littleidea/agile-infrastructure-velocity-09
- https://itrevolution.com/the-three-ways-principles-underpinning-devops/
- https://itrevolution.com/devops-culture-part-1/
- slideshare.net/devopsguys/dev-opsguys-devops-101-for-recruiters
- slideshare.net/CarmenDeArdo/interconnect-2016-continuous-visibility-dta3268-59336071
- accenture.github.io/adop-docker-compose/
DevOps Sites & Resources

- devops.com
- bestdevops.com
- devops.sys-con.com
- infoq.com/devops
- dzone.com/devops-tutorials-tools-news
- agileconnection.com/topics/devops
- devopscafe.org
- itrevolution.com
- techbeacon.com/devops
- continuousdelivery.com
DevOps Vendor Sites & Resources

- accenture.github.io/adop-docker-compose/
- developer.capitalone.com/opensource-projects/hygieia/
- aws.amazon.com/devops/
- ibm.com/developerworks/learn/devops/
- visualstudio.com/devops/
- atlassian.com/devops
- puppet.com/solutions/devops
- xebialabs.com/solutions/devops/
- ca.com/us/why-ca/devops.html
- software.microfocus.com/en-us/solutions/devops-solutions
High performers

- 22% less time spent on unplanned work and rework
- 29% more time to spend on new work
- 33% more of their CM
- 27% more of their testing
- 30% more of their deployments

High-performing IT organizations report:

- 46x more frequent deployments
- 96x faster recovery from failures
- 460x faster lead times
- 5x lower change failure rate

Have better employee loyalty (eNPS)

- 2.2x more likely to recommend company
- 1.8x more likely to recommend team

Source: State of DevOps Report Google/DORA
DevOps Team Topologies (Matthew Skelton) devopstopologies.com

**Anti-Type A: Dev & Ops Silos**

- Dev
- Ops

**Anti-Type B: DevOps Team Silo**

- Dev
- DevOps
- Ops

**Anti-Type C: Dev Don't Need Ops**

- Dev
- DevOps
- Ops

**Anti-Type D: DevOps as Tools Team**

- Dev
- DevOps
- Ops

**Anti-Type E: Rebranded SysAdmin**

- Dev
- DevOps
- Ops

**Anti-Type F: Ops Embedded in Dev Team**

- Dev
- DevOps
- Ops

---

**Type 1: Dev+Ops Collaboration**

- Dev
- Ops

**Type 2: Fully Shared Ops**

- Dev
- Ops

**Type 3: Ops as IaaS**

- Dev
- DevOps
- Ops

**Type 4: DevOps-as-a-Service**

- Dev
- DevOps
- Ops

**Type 5: Temp DevOps Team**

- Dev
- DevOps
- Ops

**Type 6: DevOps Evangelists Team**

- Dev
- DevOps
- Ops

**Type 7: SRE Team (Google Model)**

- Dev
- DevOps
- SRE
- Ops

**Type 8: Container-Driven**

- Dev
- DevOps
- Ops
Software Development DevOps 2018 Q1 Graph

http://infoq.link/devops-trends
What is DevOps? [continued]

An agile relationship between Development & IT Operations to improve communication, collaboration, and fast flow between the various roles in the software value-delivery lifecycle. –XebiaLabs

Operations & development engineers participating together in the entire service lifecycle, from design through development process to production support. –The Agile Admin

A change in IT culture, focusing on rapid IT service delivery through adoption of agile/lean practices in the context of a system-oriented approach, that emphasizes people (and culture), seeks to improve collaboration between operations and development teams. Implementations utilize automation tools that leverage an increasingly programmable and dynamic infrastructure. –Gartner

A software engineering culture and practice that aims at unifying software development (Dev) & IT operations (Ops) ... shorter development cycles, increased deployment frequency, more dependable releases, in close alignment with business objectives. –Wikipedia
Comparison of Lifecycles

- **Waterfall (phases)**
  - Definition
  - Analysis
  - Design, Code & Unit-Test
  - Integration Build+Test
  - Test
  - Accept
  - Deploy

- **Incremental (integrations)**

- **Agile (sprints)**

- **Continuous Delivery**

- **Continuous Deployment**
Source: https://itnext.io/do-not-put-devops-in-a-cage-3604a83821e1
Source: [https://itnext.io/do-not-put-devops-in-a-cage-3604a83821e1](https://itnext.io/do-not-put-devops-in-a-cage-3604a83821e1)
## DevOps Tools Landscape

### PERIODIC TABLE OF DEVOPS TOOLS (V2)

<table>
<thead>
<tr>
<th>Open Source</th>
<th>CI</th>
<th>SCM</th>
<th>Build</th>
<th>Database Mgmt</th>
<th>Testing</th>
<th>Containerization</th>
<th>Collaboration</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fr</td>
<td>Free</td>
<td>Fr</td>
<td>Deployment</td>
<td>Release Mgmt</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pd</td>
<td>Paid</td>
<td>Bl</td>
<td>Cloud / Iaas / Paas</td>
<td>Logging</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>En</td>
<td>Enterprise</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DevOps Culture > Principles > Practices > Tools

Figure 1. Structured equation model showing relationships between constructs

Transformational Leadership
- Personal recognition
- Supportive leadership
- Intellectual stimulation
- Inspirational communication
- Vision

Test and deployment automation
- Continuous integration
- Trunk-based development
- Shifting left on security
- Loosely-coupled architecture
- Empowered teams

Lean Product Management
- Team experimentation
- Working in small batches
- Gathering and implementing customer feedback

Continuous Delivery

IT performance
- Organizational performance
- Non-commercial performance

Deployment pain

Source: puppet.com/resources/whitepaper/state-of-devops-report
Engage In Seven Habits To Achieve Highly Effective DevOps

1. Establish trust and transparency between Dev and Ops.
2. See everything through the eyes of the customer.
3. Streamline your application delivery pipeline.
4. Adopt a loosely coupled service-oriented architecture.
5. Reward solution simplicity and reliability.
6. Adapt and improve how you use customer experience data.
7. Dev and Ops need to walk in the other’s shoes.

Source: https://www.microfocus.com/media/analyst-paper/Forrester_The_Seven_Habits_Of_Highly_Successful_DevOps_10-2014.pdf