Are we measuring the right thing?

2017

Robin Yeman
LM Fellow
Introduction

Robin Yeman
LM Fellow ; Enterprise Agile Coach
Business Performance Program Operations

Lockheed Martin Business Unit
Email: robin.yeman@lmco.com
Phone: 571-535-5854

Career Highlights : 22 Years at Lockheed Martin, 15 Years of Agile

Roles: Software Engineer, Systems Engineer, Test Engineer, Capture Manager, Engineering Program Manager (EPM), Subcontracts Program Manager (SPM), Program Manager (PM)

Certifications: Certified Enterprise Coach (CEC); Scrum Master (CSM), Certified Scrum Practitioner (CSP), Professional Scrum Master (PSM), Scaled Agile Program Consultant (SPC), Certified Systems Engineer (CSEP), Program Management Professional (PMP), Program Management Agile Professional (PMI-ACP), ITIL Foundations v3

Education:
Syracuse University B.S. Management Information Systems
Rensselaer Polytechnic Institute M.S. Software Engineering
Agenda

1. What is DevOps
2. Measures
3. Results from DevOps
1.0 DevOps
DevOps is “a cross-disciplinary community of practice dedicated to the study of building, evolving and operating rapidly-changing resilient systems at scale.”

- Jez Humble
Why is it confusing?

https://blog.openshift.com/what-is-devops-really/
# DevOps Manifesto

We are uncovering better ways of running systems by doing it and helping others do it. Through this work we have come to value:

<table>
<thead>
<tr>
<th>DevOps Manifesto</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individuals and interactions</td>
</tr>
<tr>
<td>Working systems</td>
</tr>
<tr>
<td>Customer collaboration</td>
</tr>
<tr>
<td>Responding to change</td>
</tr>
</tbody>
</table>
DevOps Principles

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable functionality.
- Infrastructure is code and should be developed and managed as such.
- Welcome changing requirements even late in development, Agile processes harness change for competitive Advantage.
- Deliver functionality frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people, operations and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
DevOps Principles (continued)

- Working software successfully delivered by sound systems is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, operations, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity—the art of maximizing the amount of work not done is essential. (KISS Principal)
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

DevOps Principals
Why Measure

• Enable us to predict success against business goals
• Evaluate status against goal
• Establish baseline for future comparison
2.0 Measures
Biggest Problem with our measures

Measuring Output instead of Outcomes

**Inputs**
- Requirements
- Resources

**Outputs**
- Source Lines of Code (SLOC)
- Defects
- Velocity
- Utilization

**Outcomes**
- Lead time
- Deployment Frequency
- Mean Time To Restore
- Change fail percentage
Source Lines of Code (Sloc)

- Incentivizes team incorrectly
- Varies greatly outside of procedural languages
- Adversely affects estimation
Defects

- Incentivizes team to find bugs as opposed to minimizing defects in implementation in the first place.
- Not all code is created equal.
- Does not reflect severity.
Velocity

- Team Dependent
- Teams game velocity

Velocity - Average number of Story points a team can complete Per time box.
Utilization

- Once utilization gets high there is not any spare capacity to handle unplanned work or changes.
- Reduction in quality levels
- Burnout
Measure Outcomes not people

- Hours
- Utilization
- Speed

Outcome Achieved

Cycle Time of Capability
You will get what you measure
Measures will be gamed

- Choose Carefully
- Pick metrics that only improve when gamed
What should I measure

Things the business cares about
What do businesses care about

• Profit
• Value Delivered
• Quality
• Cost
• Schedule
## Good options

<table>
<thead>
<tr>
<th>Metric</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Deployment Frequency</td>
<td>How frequent we deploy</td>
</tr>
<tr>
<td>2. Lead Time</td>
<td>Time from a needed capability is identified to the time customer receives</td>
</tr>
<tr>
<td>3. Change Failure Rate</td>
<td>% of Failures for every baseline change</td>
</tr>
<tr>
<td>4. Mean Time To Recover (MTTR)</td>
<td>How quickly we recover from a failure</td>
</tr>
</tbody>
</table>
Deployment Frequency

Increase deployment frequency

- Drives value quicker
- Increases learning from feedback
- Drives down risk
Lead Time

Shorter lead times:

- Increase profit
- Increase Quality
- Reduce Cost
- Reduce Risk
Change Fail Percentage

Reducing the percentage of failures when making changes to the baseline

- Increase flexibility to make change
- Reduce risk of changes
- Increase Value to our customers
- Higher profit for the business
Mean Time to Restore

Shorter mean time to repair

• Increases stability
• Reduces the amount of time teams are working on unplanned work
• Reduce Life cycle cost for customers
• Increases profit for the business
3.0 Results from DevOps
Results from high performing organizations

46x
More frequent deployments

5x
lower Change failure rate

96x
Faster recovery From failures

440x
Shorter lead times

Puppet and DORA (DevOps Research and Assessment)
2017 State of DevOps Report
High performing organizations are twice as likely to achieve or exceed business goals.

<table>
<thead>
<tr>
<th>Commercial</th>
<th>Non-Commercial</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Productivity</td>
<td>• Quantity of Products</td>
</tr>
<tr>
<td>• Profitability</td>
<td>• Operating Efficiency</td>
</tr>
<tr>
<td>• Market Share</td>
<td>• Customer Satisfaction</td>
</tr>
<tr>
<td>• # of Customers</td>
<td>• Quality of Products</td>
</tr>
<tr>
<td>2x</td>
<td>• Mission Goals</td>
</tr>
</tbody>
</table>

Puppet and DORA (DevOps Research and Assessment)  
2017 State of DevOps Report
Who was measured

27,000 Responses

Puppet and DORA (DevOps Research and Assessment)
2017 State of DevOps Report
Performance Drivers to obtain these results

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

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

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

**IT performance**
- Organizational performance
- Non-commercial performance

**Deployment pain**

Puppet and DORA (DevOps Research and Assessment)
2017 State of DevOps Report
By 2020, half of the CIO’s who have not transformed their teams will be displaced from their organizations’ digital leadership teams.

Gartner
DevOps. Where Do We Start?

Figure 1. Gartner DevOps Model