Accelerating DevOps Performance Using Lean Metrics
Agenda

1. Compare and contrast lean and agile values
2. Accelerate DevOps performance using lean and agile metrics
3. Use the CUE model to solve organizational, departmental, and team performance issues
4. Discuss the importance of a DevOps culture
Embrace Lean-Agile values

House of Lean

VALUE

- Respect for people and culture
- Flow
- Innovation
- Relentless improvement

LEADERSHIP

Agile Manifesto

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan
Lean vs Agile

Agile Decision Filter
- Make progress with imperfect information
- Encourage a high trust culture
- Treat WIP as a liability rather than an asset

Lean Decision Filter
- Value trumps flow
- Flow trumps waste elimination
- Eliminate waste to improve efficiency, do not pursue economies of scale
Lean + Agile =

The Goal: Value

Sustainable shortest lead time. Best quality and value (to people and society). Most customer delight, lowest cost, high morale, safety.

Pillar 1: Respect for People
- don’t trouble your customer
- develop people-then build products
- no wasteful work
- teams and individuals evolve their own practices and improvements
- build partners with stable relationships, trust and coaching lean thinking
- develop teams

Development Practices
- long-term great engineers
- mentoring mgr-eng-teacher
- cadence
- cross-functional
- team room + visual mgmt
- entrepreneurial chief/product manager
- set based concurrent dev.
- create more knowledge

Pillar 2: Continuous Improvement
- Go See
- kaizen
- spread knowledge
- small, relentless
- retrospectives
- 5 whys
- eyes for waste
- variability, overburden, NVA, (handoff, WIP, info scatter delay, multitasking, defects, wishful thinking...)
- perfection challenge
- Work to flow (smaller batch sizes, low cycle time)

14 Lean Principles
Long-term philosophy, flow, pull, level workload, stop and fix, master norms, visual controls, tested tech, leaders-teachers from within, develop exceptional people, help partners be lean, go see, consensus and action, learning/reflection/kaizen

Foundation: Management Support
Management applies and teaches lean thinking, and bases decisions on this long-term philosophy

Lamman and Vedde (2000)

Adapted by Leffingwell, LLC (2009)
DevOps Tool Chain

**Build**
- CODE
  - Code Development and Review, Continuous Integration Tools
- BUILD
  - Version control tools, code merging, build status
- TEST
  - Test and results determine performance
- PACKAGE
  - Artifact repository, Application pre-deployment staging
- RELEASE
  - Change management, release approvals, release automation

**Release**
- CONFIGURE
  - Infrastructure configuration and management, Infrastructure as Code tools
- MONITOR
  - Applications monitoring performance, End user experience

**Operate**
<table>
<thead>
<tr>
<th>Lean</th>
<th>Agile</th>
<th>DevOps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce Waste</td>
<td>Increase Value</td>
<td>Speed to Market</td>
</tr>
<tr>
<td>Quality Products</td>
<td>Working Software</td>
<td>Lower failure rates</td>
</tr>
<tr>
<td>Shortened Lead Time</td>
<td>Improve Speed to Delivery</td>
<td>Increased Deployment Frequency</td>
</tr>
<tr>
<td>Continuous Improvement</td>
<td>Learning Organization</td>
<td>Aligned Organization</td>
</tr>
<tr>
<td>Build a Kaizen Culture</td>
<td>Collaborative Culture</td>
<td>DevOps Culture</td>
</tr>
</tbody>
</table>
House of DevOps

The Goal: Value

Revenue per user story

CSAT

Lead times

Efficiency and Effectiveness
- FTE to customer ratios
- Change / release cost burden
- Cost per transaction / app

Quality and Velocity
- MTTR
- Cycle Times
- Rollback rates
- Ops Costs
- Deployment Frequency

Culture, Collaboration and Sharing
- Staff retention
- Morale / Job satisfaction
- Open source contributions / Mentoring

CA Technologies, 2015
Organizational Goals and Alignment

Adapted from “BPM CBOK Version 3.0: Guide to the Business Process Management Common Body Of Knowledge”.

- Strategic Change Management (Business)
- IT Change Management
- Organization Change Management (People Side)
- Organization Level
- Group Level
- Individual Level
Remove Constraints (Bottlenecks)

devops - optimize the whole

Business

Theory of Constraints

Customer

HR  MGMT  FIN

SALES  DEV  OPS
Accelerating the DevOps Process

Speed to Market

Increased Deployment Frequency

Reduced Failure Rates
Accelerating the DevOps Process
CUE Model

1. Consolidate activities
   • Follow tenets of LeSS, which provides a balance between process control improvements and value delivery
   • Use strategic planning tools to align Business and IT goals

2. Use metrics that matter
   • Agile (Value, Time to Value, Cost of Value)
   • Lean (Throughput, inventory, operating time (cycle time))

3. Eliminate waste
   • Remove unnecessary and/or nonproductive items and/or steps
Consolidate Activities

Use Metrics that Matter

Eliminate Waste
Consolidate Activities
CUE Model Exercise
Step 1: Consolidate

In teams of 3 people discuss and write down the following:

1. What activities will you consolidate?
2. What activities will you not consolidate?
3. How will you consolidate activities?
Use Metrics That Matter

One Metric that Matters

“You can’t manage what you can’t measure”

• Answers the most important question you have
• Forces you to draw a line in the sand
• Focuses the entire company
• Inspires a culture of experimentation
CUE Model Exercise
Step 2: Use Metrics That Matter

In teams of 3 people discuss and write down the following:

1. What is the one metric that matters most?

2. What other metrics are important to your organization?
Eliminate Waste

DevOps and Lean: Removing Waste

*DevOps shares ideas with Lean such as the removal of waste from business processes*

- Waste of overproduction
- Waste of time and resources waiting
- Waste of processes themselves (overhead)
- Waste of poor quality products
Eliminate Waste

DevOps focuses on removing waste from the process.

Remove waste first, then automate. But don’t automate processes you don’t fully understand.
CUE Model Exercise
Step 3: Eliminate Waste

In teams of 3 people discuss and write down the following:

1. What are your areas of waste?
2. Where are your areas of waste?
3. Where are your constraints (bottlenecks)?
Lean Management Builds Morale

- Use Metrics that Matter
- Consolidate Activities
- Eliminate Waste

DevOps Culture

- Decreased Burnout
- Technical Excellence
- Kaizen Culture
Summary

1. Lean + Agile practices encourage flow and improves throughput while creating valuable products.
2. Using the CUE model aligns goals to optimize processes and reduce waste.
A Few Good References
Appendices / Handouts
"Lean" is a systematic, continuous improvement approach that focuses on eliminating waste from your processes.

<table>
<thead>
<tr>
<th>Lean Principles</th>
<th>Agile/Scrum/XP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lean Tools</td>
<td></td>
</tr>
<tr>
<td>Customer Value</td>
<td>Standard dev practices, framework usage, communications model</td>
</tr>
<tr>
<td>Value Stream</td>
<td>Story Mapping from Roadmap to Release to MMF to Epic to Story</td>
</tr>
<tr>
<td>Flow</td>
<td>Automated testing, Continuous Integration, Code Health, Easy Deploy</td>
</tr>
<tr>
<td>Customer Pull</td>
<td>Refactor, Reuse, Big Visuals, Iterate, Incremental</td>
</tr>
<tr>
<td>Perfection</td>
<td>Iterate, Inspect and Adapt, Retrospectives</td>
</tr>
<tr>
<td>Reduce Waste</td>
<td>No defects, Definition of Done, Test Driven Development, Refactor, Shippable code</td>
</tr>
<tr>
<td>Cycle Time</td>
<td>Go See Yourself Customer close by, Prototypes, Iterative and Incremental, Early Deploy</td>
</tr>
<tr>
<td>Qualiry</td>
<td>Remove Waste Customer close, decisions at lowest level, good estimation, actionable user stories, no gold-plating, team instead of cowboy, remove needless metrics, automation of almost all testing, no over-the-wall, biz/IT combined teams, simplify governance, daily stand-ups</td>
</tr>
<tr>
<td>Efficiency &amp; Capability, Speed</td>
<td>Workload Leveling, Concurrent Eng. Small stories, slicing and swarming, cross-functional team, daily stand-ups</td>
</tr>
<tr>
<td>Optimal use of Resources</td>
<td>Quick Changeover Cross-functional team, simple and repeatable practices, no individual Code ownership, consistent build and deploy, team rotations, daily stand-ups</td>
</tr>
<tr>
<td>Cost</td>
<td>Automation Automated test, integration, build, deploy tools, communication</td>
</tr>
<tr>
<td>Orthogonal Arrays</td>
<td>Orthogonal Arrays OATS Pairing with Dev and QA, Automation</td>
</tr>
<tr>
<td>Lean Tools</td>
<td>Pull (Kanban) Prioritized backlog, Kanban (Scrum board) tracking/status, Feature focused, User Scenarios to Stories, Personas, Iterative and Incremental, User Acceptance Testing, Sprint Goals, Sprint Planning (only 1-2 iterations),</td>
</tr>
<tr>
<td>Capacity Planning</td>
<td>Takt Time (customer to customer) Backlog to Production measurement</td>
</tr>
<tr>
<td>Flow</td>
<td>Single Piece Flow Small stories, feature based/slicing and swarming, Iterative and Incremental</td>
</tr>
<tr>
<td>Customer Pull</td>
<td>Waste Elimination Only develop stories that are actionable, only do what’s absolutely necessary, simplicity in all things, automation, excellence in engineering,</td>
</tr>
</tbody>
</table>
Lean + Agile =

**The Goal: Value**
Sustainable shortest lead time. Best quality and value (to people and society). Most customer delight, lowest cost, high morale, safety.

**Pillar 1: Respect for People**
- don’t trouble your customer
- develop people-then build products
- no wasteful work
- teams and individuals evolve their own practices and improvements
- build partners with stable relationships, trust and coaching lean thinking
- develop teams

**Development Practices**
- long-term great engineers
- mentoring mgr-eng-teacher
cadence
- cross-functional
- team room + visual mgmt
- entrepreneurial chief/ product manager
- set based concurrent dev.
- create more knowledge

**Pillar 2: Continuous Improvement**
- Go See
- kaizen
- spread knowledge
- small, relentless,
- retrospectives
- 5 whys
- eyes for waste variability, overburden, NVA, (handoff, WIP, info scatter delay, multitasking, defects, wishful thinking...)
- perfection challenge
- Work to flow (smaller batch sizes, low cycle time)

**14 Lean Principles**
Long-term philosophy, flow, pull, level workload, stop and fix, master norms, visual controls, tested tech, leaders-teachers from within, develop exceptional people, help partners be lean, go see, consensus and action, learning/reflection/kaizen

**Foundation: Management Support**
Management applies and teaches lean thinking, and bases decisions on this long-term philosophy.

Derived from: Toyota Production System (2004), Lamping and Vodde (2009)
Adapted by Leffingwell, LLC. (2009)
The Seven Wastes

The ultimate goal of LEAN processes and production is the continual elimination of waste.

Waste is defined as any factor that does not add value.
**KPI**

**Definition:**
A measurable expression for the achievement of a desired level of results in an area relevant to the entity’s activity.

---

**SMART Objectives**

Objective: Increase customers base
+ KPI: % Market share
+ Target: 20%
+ Timeframe: By Financial Year End
+ Responsible: Sales Director

*Increase customers base to reach 20% market share by FY end under Sales Director leadership*

---

**KPI naming standards**

KPIs – start with symbols

- Value of
- Number of
- Percentage of
  - $ Net profit
  - # Defects
  - % Budget variance

---

**Value added by KPIs**

- **Clarity**
  Paint a clear picture of strategy

- **Focus**
  Focus on what matters / requires attention

- **Improvement**
  Monitor progress towards the desired state

---

**Metrics - KPIs - KRI - Analytics**

- **KPIs Performance Assessment**
- **KRI Risk Assessment**
- **Analytics Decision Making**
Lean Metrics Definitions

• Lead Time (LT): The elapsed time from receiving a customer request to delivering on that request. Lead Time = Process Time plus Wait Time.

• Wait Time (WT): The time that work sits idle not being worked.

• Work-in-progress (WIP): The amount of work in a system that has been started but not finished.

• Process Time (PT): Process time begins when the work has been pulled into a doing state and ends when the work is delivered to the next downstream customer.
Flow Metrics

• Throughput
  • Rate which inventory generates money through sales, not just production.

• Inventory
  • Money invested in goods the firms intends to sell.

• Operating Expense
  • Includes all money the firm spends converting inventory into throughput.
Metrics

1. Number of bug fixes for top 20 systems
2. Number of critical issues, average time to close and longest time to close – looking just at the average may mask longer running issues
3. WIP: number of projects in each phase of the SDLC
4. Customer feedback for top 5 systems – preferably actual user feedback; failing that whatever you can infer from app usage. Top 5 systems may change over time as you launch new functionality or as usage changes according to the business calendar.
5. Percentage of projects who deliver 100% of their agreed scope
6. Core application availability – actual availability to users, not SLA adherence