FINDING AND FIXING YOUR ORGANIZATION’S AGILE POTHOLES

Les Grove
October 10, 2018

We will do real time polling during this session. Text lesgrove983 to 22333 or Go to PollEV.com/lesgrove983
AGENDA

• Catalyte’s “Road”
• Navigation
• Mapping
• Road Cones
• Next Stops
First:
A Couple of Polls
1. Our test coverage satisfies our test quality goals

- Always (A)
- Very Often (B) - 18%
- Often (C) - 29%
- Occasionally (D) - 41%
- Rarely (E) - 12%
- Never (F)
2. Critical defects are prevented from reaching Production

- Always: 0% (Option A)
- Very Often: 67% (Option B)
- Often: 24% (Option C)
- Occasionally: 10% (Option D)
- Rarely: 0% (Option E)
- Never: 0% (Option F)
GIVEN JUST TWO DATA POINTS

WHICH POTHOLE SHOULD WE TRY TO FIX FIRST?

1. TEST COVERAGE SATISFYING QUALITY GOALS

2. CRITICAL DEFECTS PREVENTED FROM PROD
CATALYTE HAS TEAMS IN DIVERSE SITUATIONS
DIFFERENT DOMAINS

- RETAIL & E-COMMERCE
- HEALTHCARE
- TRAVEL & HOSPITALITY
- TECHNOLOGY
- PUBLIC SECTOR
DIFFERENT LOCATIONS

- At Catalyte
- At Client
- Combination
DIFFERENT TEAM CONFIGURATIONS

- Complete Catalyte Teams
- Client-Catalyte Combination
- Client-Catalyte-3rd Party Combo
PROBLEM STATEMENT

How to find the impediments caused by the organization?
AGILE COACHING

Delivering value across multiple teams and the organization by:

- Working with teams, stakeholders, and management
- Facilitating
- Training
- Mentoring
- Providing leadership
- Supporting collaboration
- Developing consistency
- Managing impediments (aka potholes)
HOW TO IDENTIFY IMPEDIMENTS AT ORGANIZATIONAL LEVEL?

Choice #1
• Attend ceremonies for all teams
• Interview many team members
• Look for patterns

Choice #2
• Agility assessments
AGILITY ASSESSMENTS

Team-Level Assessments
Checklists
- The ScrumMaster Checklist
- The Unofficial Scrum Checklist
- How Agile Are You? Checklist

Organization-Level Assessments
Frameworks
- Agile Fluency Model
- Agile Adoption Framework
- Agile Maturity Matrix

79 tools and checklists at BenLinders.com/tools/agile-self-assessments
The bottom line
If you achieve these you can ignore the rest of the checklist. Your process is fine.
- Delivering working, tested software every 4 weeks or less
- Delivering what the business needs most
- Process is continuously improving

Core Scrum
These are central to Scrum. Without these you probably shouldn’t call it Scrum.
- Retrospective happens after every sprint
- Results in concrete improvement proposals
- Some proposals actually get implemented
- Whole team + PO participates

- Clearly defined product owner (PO)
- PO is empowered to prioritize
- PO has knowledge to prioritize
- PO has direct contact with team
- PO has direct contact with stakeholders
- PO speaks with one voice (in case PO is a team)

- Team has a sprint backlog
- Highly visible
- Updated daily
- Owned exclusively by the team

- Daily Scrum happens
- Whole team participates
- Problems & impediments are surfaced

- Demo happens after every sprint
- Shows working, tested software
- Feedback received from stakeholders & PO
- Have Definition of Done (DoD)
- DoD achievable within each iteration

- Iteration length 4 weeks or less
- Always on time
- Team not distracted or controlled by outsiders
- Team usually delivers what they committed to
- Team members sit together
- Max 9 people per team

Recommended but not always necessary
Most of these will usually be needed, but not always all of them. Experiment!
- PBL items are broken into tasks within a sprint
- Team members not locked into specific roles
- Iterations that are doomed to fail are terminated early
- PO has a product backlog (PBL)
- Top items are prioritized by business value
- Top items are estimated
- Estimates written by the team
- PO has product vision that is in sync with PBL
- PBL and product vision is highly visible
- Everyone on the team participates in estimating
- PO available whose team is estimating
- Estimate relative size (story points) rather than time
- Whole team knows top 1-2 impediments
- SM has strategy for how to fix top impediments
- SM focusing on removing impediments
- Escalated to management when teams can’t solve
- Team has a Scrum Master (SM)
- SM sits with the team

Scaling
These are pretty fundamental to any Scrum scaling effort.
- You have a Chief Product Owner (if many POs)
- Dependent teams do Scrum of Scrum
- Dependent teams integrate within each sprint

Positive indicators
Leading indicators of a good Scrum implementation.
- Having fun! High energy level.
- Overtime work is rare and happens voluntarily
- Discussing, estimating, and experimenting with the process

PO = Product owner  SM = Scrum Master  PBL = Product Backlog  DoD = Definition of Done
http://www.crisp.se/scrum/checklist  |  Version 2.2 (2010-10-04)
### Team Level Maturity Matrix

<table>
<thead>
<tr>
<th>Benefits to the team</th>
<th>Current Level</th>
<th>Impeded (0)</th>
<th>In transition (1)</th>
<th>Sustainable (2)</th>
<th>Agile (3)</th>
<th>Ideal (4)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Being Agile</td>
<td></td>
<td>No understanding of the spirit of Agile</td>
<td>Doing the mechanics</td>
<td>80% of the team can explain the benefits of Agile, believe in the benefits of Agile, understand the spirit of Agile. The team is making improvements on a regular basis.</td>
<td>Working in an Agile manner</td>
<td>Actively pursuing new ways of working in an Agile manner</td>
<td></td>
</tr>
<tr>
<td>Productivity-Quality</td>
<td></td>
<td>Blame game, finger pointing, denial, anger, shouting, backstabbing, passive aggressive, turn-over and other behaviors on a regular basis. Desire to go back to the old ways, active resistance to change, scapegoating. There is churn or people are frequently making references to quitting or how much they dislike their work or work environment.</td>
<td>There are still elements of previous state, but there is steady progress away from those behaviors, problems are being actively addressed, and there is a general feeling that morale is improving</td>
<td>For the most part everyone is getting along and happy at work. There is very little if any talk about &quot;going back&quot; and there is generally believed that things are either better than before or at least not worse</td>
<td>The team generally believes that their work life is significantly better than before. They are happy, engaged, productive, and genuinely enjoy working together.</td>
<td>Most team members feel like this is one of the best teams they have ever worked on, they are excited to come in to work and are looking forward to the next day when they leave.</td>
<td></td>
</tr>
</tbody>
</table>

### Team Dynamics

<table>
<thead>
<tr>
<th>Team Stage</th>
<th>Current Level</th>
<th>Impeded (0)</th>
<th>In transition (1)</th>
<th>Sustainable (2)</th>
<th>Agile (3)</th>
<th>Ideal (4)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forming</td>
<td></td>
<td>Some defect team norms that are generally recognized, but haven't yet been written down and agreed on by the team.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storming</td>
<td></td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Norming</td>
<td></td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Have been performing consistently for at least 8 weeks</td>
<td></td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tuckman working agreement</td>
<td></td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td>Written down, agreed on by the team, clearly visible in a public area such as the team room.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Organization Level Maturity Matrix

<table>
<thead>
<tr>
<th>Organizational Structure</th>
<th>Current Level</th>
<th>Target Level</th>
<th>Impeded (0)</th>
<th>In transition (1)</th>
<th>Sustainable (2)</th>
<th>Agile (3)</th>
<th>Ideal (4)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function and project based</td>
<td>0</td>
<td>Unknown, no projects that are approved are immediately considered started, or there is no upper limit on how many can be in progress at a time.</td>
<td>Function and project based</td>
<td>There is understanding that structuring the organization around products, teams, and delivery is better. Some changes have already been made and more are underway.</td>
<td>There is consistent effort applied to moving to a product, team, and delivery based organization.</td>
<td>50%+ of the organizational changes required to move to a product, team, and delivery based organization have been made and the rest is actively in progress.</td>
<td>Product, team, and delivery based</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Projects in Progress</th>
<th>Current Level</th>
<th>Target Level</th>
<th>Impeded (0)</th>
<th>In transition (1)</th>
<th>Sustainable (2)</th>
<th>Agile (3)</th>
<th>Ideal (4)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delivery based metrics are being tried, planned, or discussed</td>
<td>0</td>
<td>Unknown, all projects that are approved are immediately considered started, or there is no upper limit on how many can be in progress at a time.</td>
<td>Function based</td>
<td>There is understanding that structuring the organization around products, teams, and delivery is better. Some changes have already been made and more are underway.</td>
<td>There is consistent effort applied to moving to a product, team, and delivery based organization.</td>
<td>50%+ of the organizational changes required to move to a product, team, and delivery based organization have been made and the rest is actively in progress.</td>
<td>Product, team, and delivery based</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Current Level</th>
<th>Target Level</th>
<th>Impeded (0)</th>
<th>In transition (1)</th>
<th>Sustainable (2)</th>
<th>Agile (3)</th>
<th>Ideal (4)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function based</td>
<td>0</td>
<td>Unknown, all projects that are approved are immediately considered started, or there is no upper limit on how many can be in progress at a time.</td>
<td>Function based</td>
<td>There is understanding that structuring the organization around products, teams, and delivery is better. Some changes have already been made and more are underway.</td>
<td>There is consistent effort applied to moving to a product, team, and delivery based organization.</td>
<td>50%+ of the organizational changes required to move to a product, team, and delivery based organization have been made and the rest is actively in progress.</td>
<td>Product, team, and delivery based</td>
<td></td>
</tr>
</tbody>
</table>
BUILDING CATALYTE’S ASSESSMENT TOOL
REQUIREMENTS FOR ASSESSMENT SOLUTION

• Quick and Easy
• Reflect the team
• No additional instructions
• Identify team improvements
• Measure over time
• Any flavor of agile

• Aggregate results to identify organizational impediments
Team Integrity
Requirements
Retrospectives
Backlog Management
Refactoring
Architecture
Quality
DELIVERY METHODS

Word Document

Excel Spreadsheet

Google Form

Requirements

Requirements describe the desired functionality or behavior of a product.

Scoring: 0 - Never, 1 - Rarely, 2 - Occasionally, 3 - Often, 4 - Very Often, 5 - Always

Cards are independent (meaning that there are no inherent dependencies on other cards)

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Never</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Always</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The team estimates the size of cards.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Never</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Always</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7- Topic Results (8 Teams)
7- Topic Results (8 Teams)
7- Topic Results (2 Lowest)
QUALITY STATEMENTS

1. Our test coverage satisfies our test quality goals
2. Critical defects are prevented from reaching Production
3. Our product is stable
4. The tests are automated where possible
5. Test results are used as the basis of code integration and deployment
3. Our product is stable.

- Always (A): 10%
- Very Often (B): 38%
- Often (C): 52%
- Occasionally (D)
- Rarely (E)
- Never (F)
5. Test Results are used as a basis of code integration and deployment.

- Always: 27%
- Very Often: 23%
- Often: 23%
- Occasionally: 23%
- Rarely: 5%
- Never: 0%
QUALITY STATEMENTS

1. Our test coverage satisfies our test quality goals
2. Critical defects are prevented from reaching Production
3. Our product is stable
4. The tests are automated where possible
5. Test results are used as the basis of code integration and deployment
Quality Results
Quality Results
DRAWBACKS OF THIS ASSESSMENT TOOL

• Time-consuming to compile the data
• Created by collection of mortals
• Not easy to compare teams and organizations over time
NEXT STOP: COMPARATIVE AGILITY TOOL

Tool’s Topics
• Teamwork
• Requirements
• Planning
• Technical Practices
• Quality
• Culture
• Knowledge Creating
• Outcomes

Previous Topics
• Team Integrity
• Requirements
• Retrospectives
• Backlog Management
• Refactoring
• Quality
• Architecture
### COMPARATIVE AGILITY – QUALITY ITEMS

**Product owners actively participate in the creation of the acceptance criteria for each feature.**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>More True than False</td>
<td>Neither False nor True</td>
<td>More False than True</td>
<td>False</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>

**All bugs are fixed during the iteration in which they are found.**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>More True than False</td>
<td>Neither False nor True</td>
<td>More False than True</td>
<td>False</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>

**At the end of each iteration there is little or no manual testing required.**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>More True than False</td>
<td>Neither False nor True</td>
<td>More False than True</td>
<td>False</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>
The team performs a variety of types of testing including functional, performance, integration, and scalability each iteration.

- True
- More True than False
- Neither False nor True
- More False than True
- False
- Not Applicable

Team members who perform testing are involved and productive right from the start of each iteration.

- True
- More True than False
- Neither False nor True
- More False than True
- False
- Not Applicable

At the end of each iteration, the team has high-quality working software that it is comfortable being tested by people outside of the team.

- True
- More True than False
- Neither False nor True
- More False than True
- False
- Not Applicable

The team has pre-defined and agreed-upon criteria for considering a feature done.

- True
- More True than False
- Neither False nor True
- More False than True
- False
- Not Applicable
All bugs are fixed during the iteration in which they are found.

- True: 4%
- More True than False: 29%
- Neither False nor True: 38%
- More False than True: 13%
- False: 17%
- Not Applicable: Non-interactive
And We Still Get Spider Charts...
Quality (+0.04 Difference from Comparative Benchmark)

Evaluates the degree to which Quality is built-in at the source. Clearly defined customer acceptance criteria, an end-to-end testing strategy, automation and a commitment to only delivering fully tested code are all best practices that typically lead to higher software quality, fewer defects in production and less rework.

And See Changes Over Time...

Product owners actively participate in the creation of the acceptance criteria for each feature (0.2)

All bugs are fixed during the iteration in which they are found (-0.09)

At the end of each iteration there is little or no manual testing required (-0.33)

The team performs a variety of types of testing including functional, performance, integration, and scalability each iteration (-0.17)

Team members who perform testing are involved and productive right from the start of each iteration (0.17)

At the end of each iteration, the team has high-quality working software that it is comfortable being tested by people outside of the team (0.26)

The team has pre-defined and agreed-upon criteria for considering a feature done (0.22)
And Better Deep-Dives into the Data...
And See the Response Distributions
SUMMARY

• Catalyte’s “Road”
• Navigation
• Mapping
• Road Cones
• Next Stops
CREDITS

Polling is done using PollEverywhere.com

All Photos are by Unknown Authors and are licensed
  • Slides 1, 3, 12, 13 and 14 are licensed under CC BY-SA
  • Slide 9 licensed under Openclipart.org
  • Slide 7 is licensed under CC BY
  • Slide 10 is modified and licensed under CC BY
  • Slide 11 is licensed under CC BY-NC-ND
  • Slides 2, 12, and 17 are licensed under CC BY-SA-NC