catalyte

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
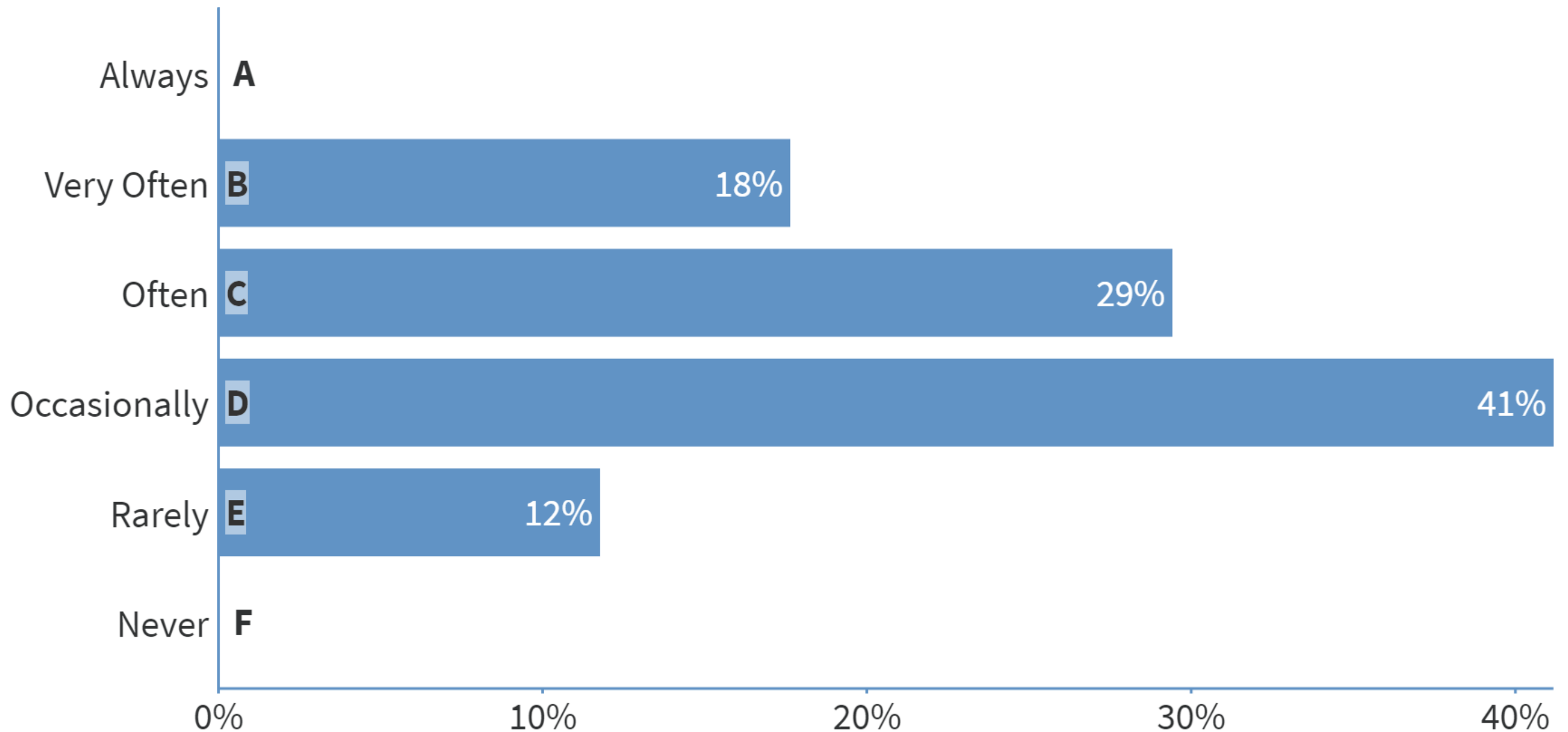Go to PollEV.com/lesgrove983

AGENDA

- Catalyte's "Road"

- Navigation

- Mapping
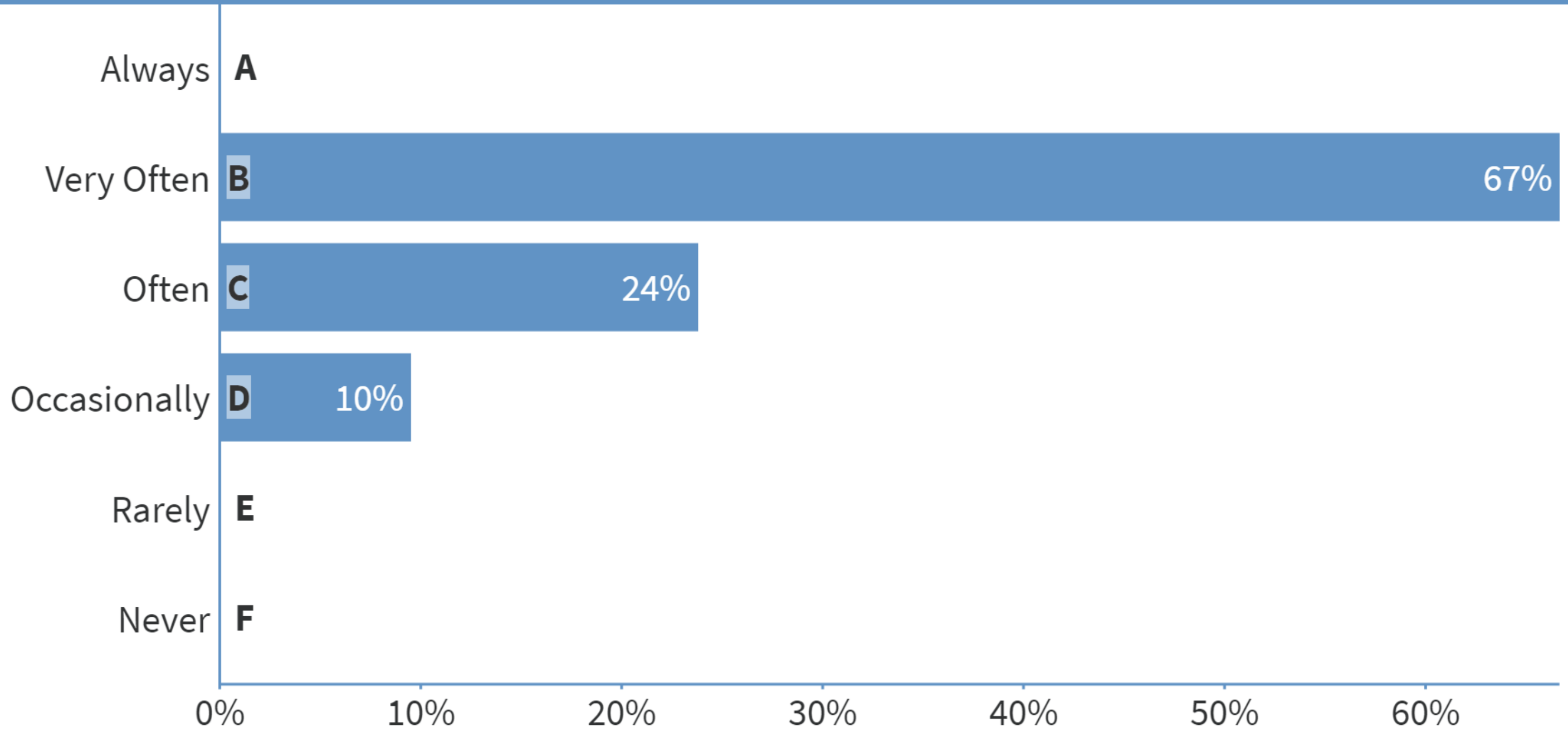
- Road Cones

- Next Stops

catalyte

**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**

0%   10%   20%   30%   40%

2. Critical defects are prevented from reaching Production

Always	A
Very Often	B 67%
Often	C 24%
Occasionally	D 10%
Rarely	E
Never	F

0%   10%   20%   30%   40%   50%   60%

catalyte

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
- NU SKIN. DISCOVER THE BEST YOU™
- Nike
- UNDER ARMOUR
- StubHub

HEALTHCARE
- CAMBIA HEALTH SOLUTIONS
- inovalon™
- Aetna®
- CareFirst
- DELTA DENTAL®
- THE EVANGELICAL LUTHERAN Good Samaritan Society® In Christ's Love, Everyone Is Someone.
- ZOOM+care®

TRAVEL & HOSPITALITY
- TravelClick®
- CHOICE HOTELS®
- AAA
- vistana™ SIGNATURE EXPERIENCES
- starwood Hotels and Resorts

TECHNOLOGY
- Microsoft
- AT&T
- redhat.
- Deloitte.

PUBLIC SECTOR
- MSRB Municipal Securities Rulemaking Board
- OPENGOV
- JOHNS HOPKINS MEDICINE
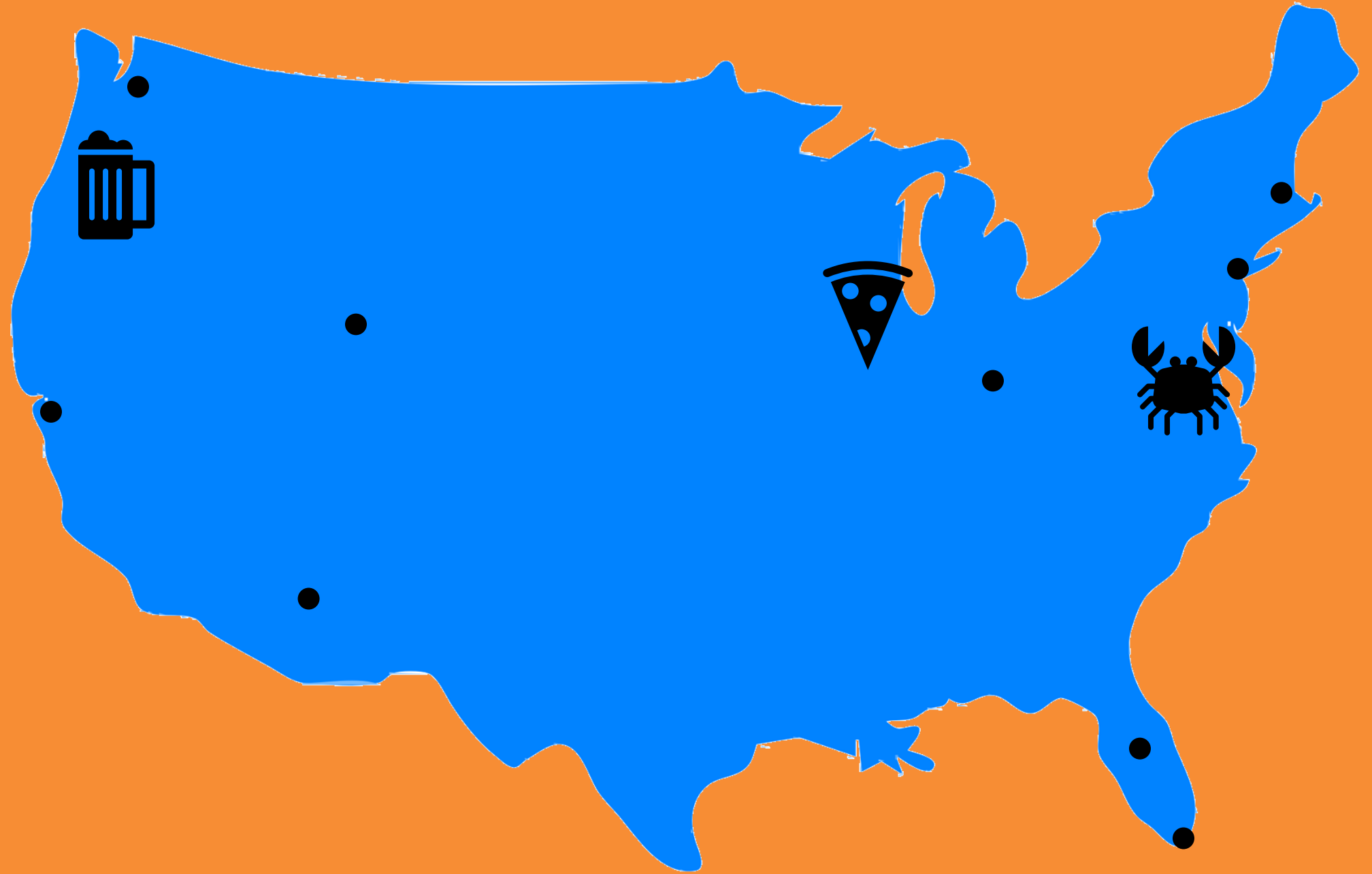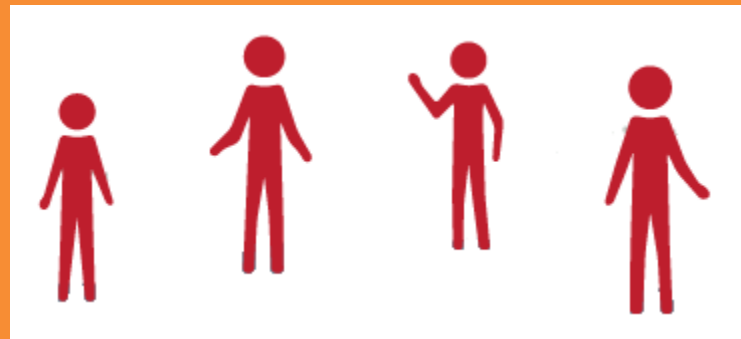- King County Library System

catalyte

DIFFERENT LOCATIONS

- At Catalyte
- At Client
- Combination

catalyte

DIFFERENT TEAM CONFIGURATIONS

Complete Catalyte Teams

Client-Catalyte Combination

Client-Catalyte-3rd Party Combo

catalyte

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

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**

- [ ] 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
  - [ ] Team **respects** DoD

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

- [ ] PO has a **product backlog (PBL)**
  - [ ] Top items are **prioritized** by business value
  - [ ] Top items are **estimated**
  - [ ] **Estimates written by the team**
  - [ ] Top items in **PBL small enough to fit** in a sprint
  - [ ] PO understands **purpose** of all backlog items

- [ ] Have **sprint planning meetings**
  - [ ] **PO participates**
  - [ ] PO brings **up-to-date PBL**
  - [ ] **Whole team** participates
  - [ ] Results in a **sprint plan**
  - [ ] Whole team believes plan is **achievable**
  - [ ] **PO satisfied with priorities**

- [ ] Timeboxed **iterations**
  - [ ] Iteration length **4 weeks or less**
  - [ ] Always **end on time**
  - [ ] Team **not disrupted or controlled** by outsiders
  - [ ] Team usually **delivers what they committed to**

- [ ] Team members **sit together**

- [ ] **Max 9 people** per team

---

the unofficial
Scrum Checklist
crisp
Henrik Kniberg

Recommended but not always necessary

Most of these will usually be needed, but not always all of them. Experiment!

- [ ] Team **has all skills** needed to bring backlog items to Done
- [ ] Team members **not locked into specific roles**
- [ ] Iterations that are **doomed to fail** are terminated early

- [ ] 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** when team is estimating
- [ ] Estimate **relative size** (story points) rather than time

- [ ] Whole team knows top 1-3 **impediments**
  - [ ] **SM has strategy** for how to fix top impediment
  - [ ] **SM focusing** on removing impediments
  - [ ] **Escalated to management** when team can't solve

- [ ] Team has a **Scrum Master (SM)**
  - [ ] SM **sits with the team**

- [ ] PBL items **are broken into tasks** within a sprint
  - [ ] Sprint tasks are **estimated**
  - [ ] Estimates for ongoing tasks are **updated daily**

- [ ] **Velocity** is measured
  - [ ] All items in sprint plan have an **estimate**
  - [ ] PO uses velocity for **release planning**
  - [ ] Velocity only includes items that are **Done**

- [ ] Team has a **sprint burndown chart**
  - [ ] Highly **visible**
  - [ ] **Updated** daily

- [ ] **Daily Scrum** is every day, same time & place
  - [ ] **PO participates** at least a few times per week
  - [ ] Max **15 minutes**
  - [ ] Each team member **knows what the others are doing**

Scaling

These are pretty fundamental to any Scrum scaling effort.

- [ ] You have a **Chief Product Owner** (if many POs)
- [ ] Dependent teams do **Scrum of Scrums**
- [ ] 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, criticizing, 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

	Benefits to the team		Current Level	Impeded (0)	In transition (1)	Sustainable (2)	Agile (3)	Ideal (4)	Comments
Team Dynamics		Being Agile		No understanding of the spirit of Agile	Doing the mechanics	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	Working in an Agile manner	Actively pursuing new ways of working in an Agile manner	
	Productivity, Quality	Morale		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.	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	For the most part people are getting along and happy at work. There is very little if any talk about "going back" and it is generally believed that things are either better than before or at least not worse	The team generally believes that their work life is significantly better than before. They are happy, engaged, productive, and genuinely enjoy working together.	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.	
	Productivity, Quality	Tuckman Stage		Forming	Storming	Norming	Have been performing consistently for at least 8 weeks	Have been performing consistently for the past 6 months	
	Productivity, Quality	Working agreement		Non-existent	Some defacto team norms that are generally recognized, but haven't yet been written down and agreed on by the team.	Written down, agreed on by the team, clearly visible in a public area such as the team room.	Followed by the team	Followed naturally, very short list, highly visible, exceptions are quickly identified and addressed.	
		Team size		>20 people on team	It is recognized that a smaller team size is needed	< 20 people on the team	< 10 people on the team	7 +/- 2 people on the team

Organization Level Maturity Matrix

	Current Level	Target Level	Impeded (0)	In transition (1)	Sustainable (2)	Agile (3)	Ideal (4)	Comments
Organizational structure	0		Function and project based	There is understanding that structuring the organization around products, teams, and delivery is better. Some changes have already been made and more are underway	There is consistent effort applied to moving to a product, team, and delivery based organization	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	Product, team, and delivery based	
Projects in Progress	0		Unknown, or 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.	The number of projects in progress is known, there is some upper limit on how many can be in progress at a time, and that limit is usually enforced.	There is a projects in progress limit that insures there is no more than a 2-1 ratio between number of people needed to fully staff all projects in progress and the actual number of people	There is no more than a 3-2 ratio of people needed to people available.	There is no more than an 11-10 ratio of people needed to people available.	
Metrics	0		Function based	Delivery based metrics are being tried, planned, or discussed	50%+ of metrics are delivery based. Some of the overlapping metrics are still being tracked and used	50%+ of metrics are delivery based and none of the overlapping metrics are being tracked or used	Metrics are primarily delivery based	

BUILDING CATALYTE'S ASSESSMENT TOOL

ROAD CONSTRUCTION

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

catalyte

TOPICS

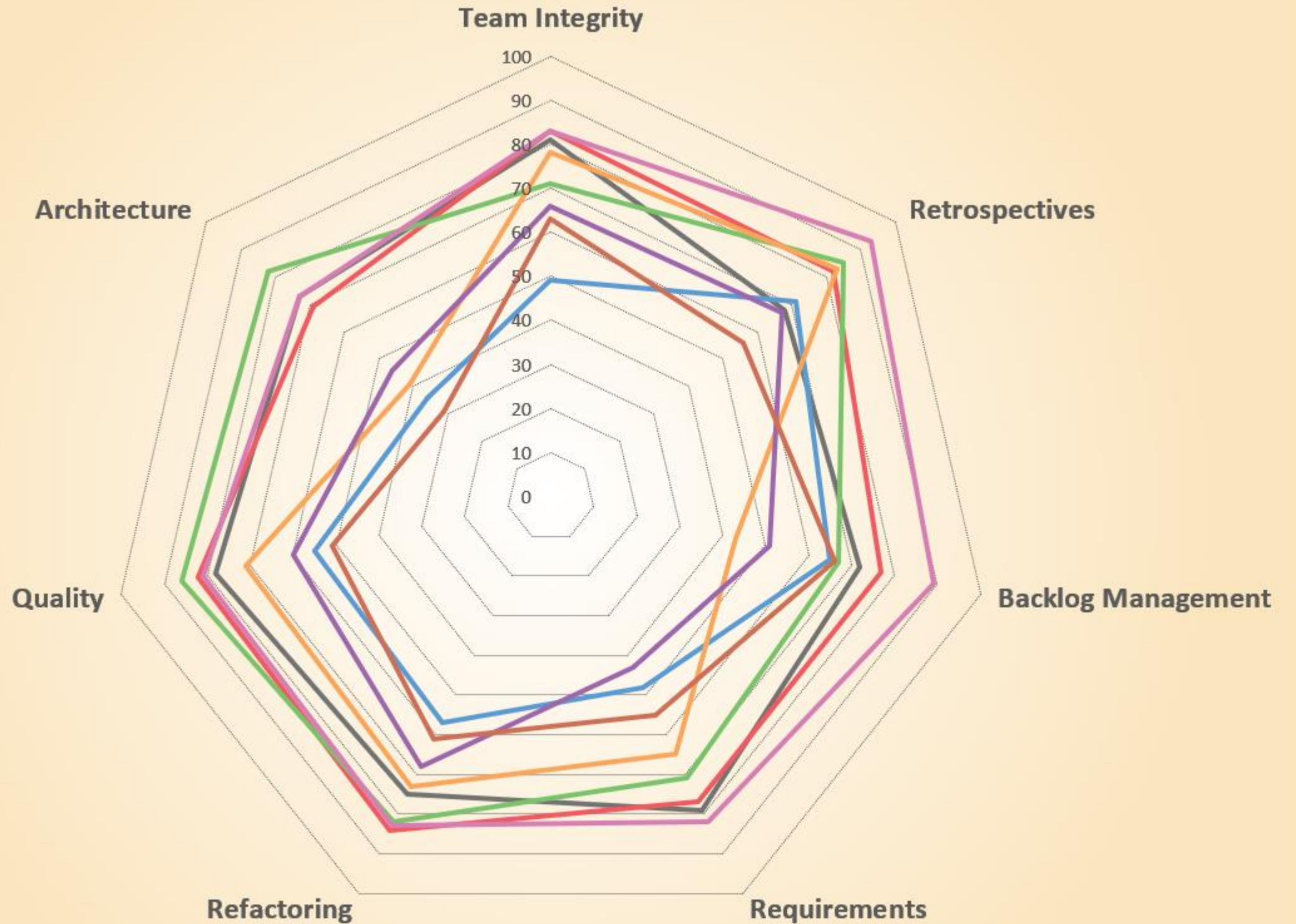Team Integrity

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) *

	0	1	2	3	4	5	
Never	○	○	○	○	○	○	Always

The team estimates the size of cards. *

	0	1	2	3	4	5	
Never	○	○	○	○	○	○	Always

catalyte

7- Topic Results (8 Teams)

Team Integrity

Retrospectives

Backlog Management

Requirements

Refactoring

Quality

Architecture

100
90
80
70
60
50
40
30
20
10
0
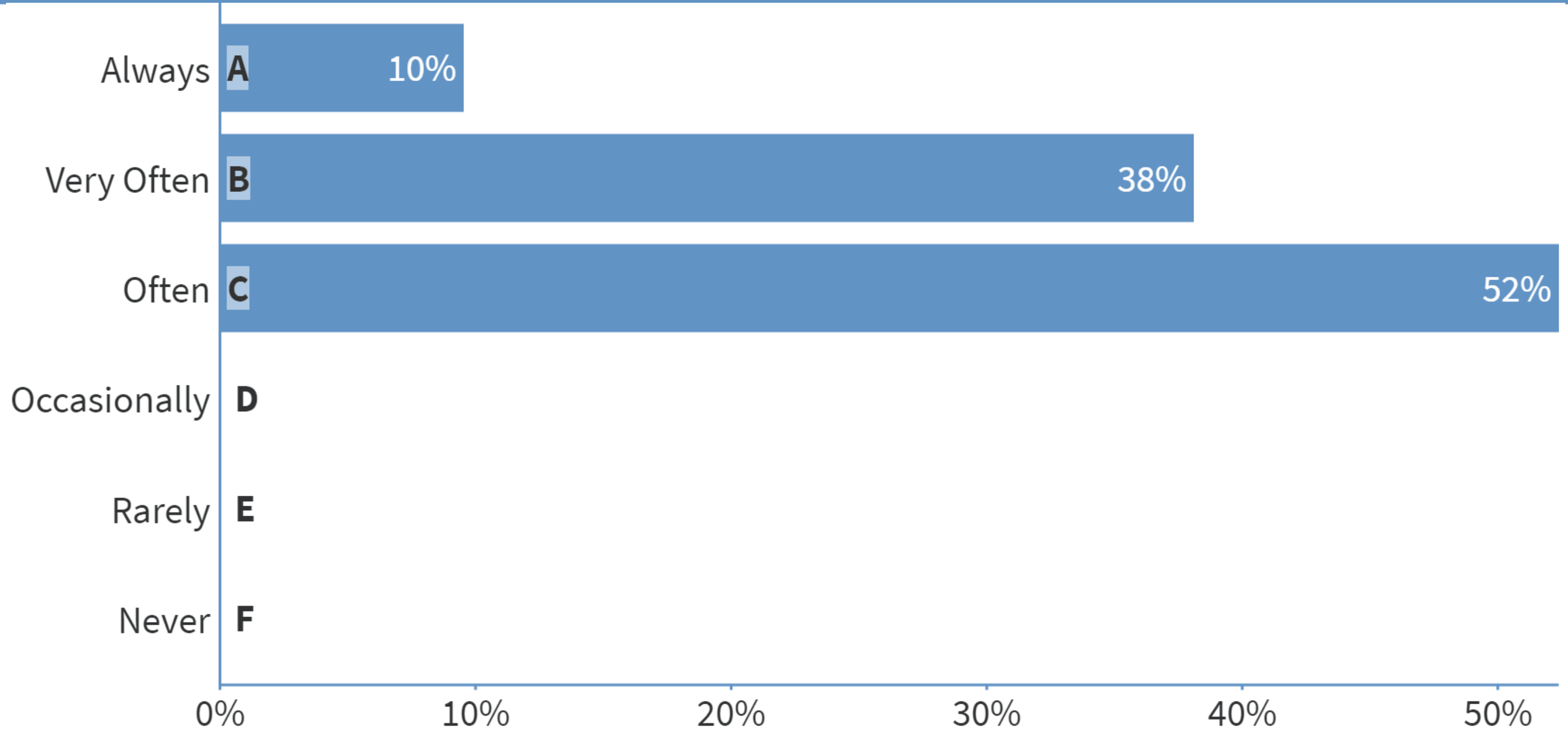
7- Topic Results (8 Teams)
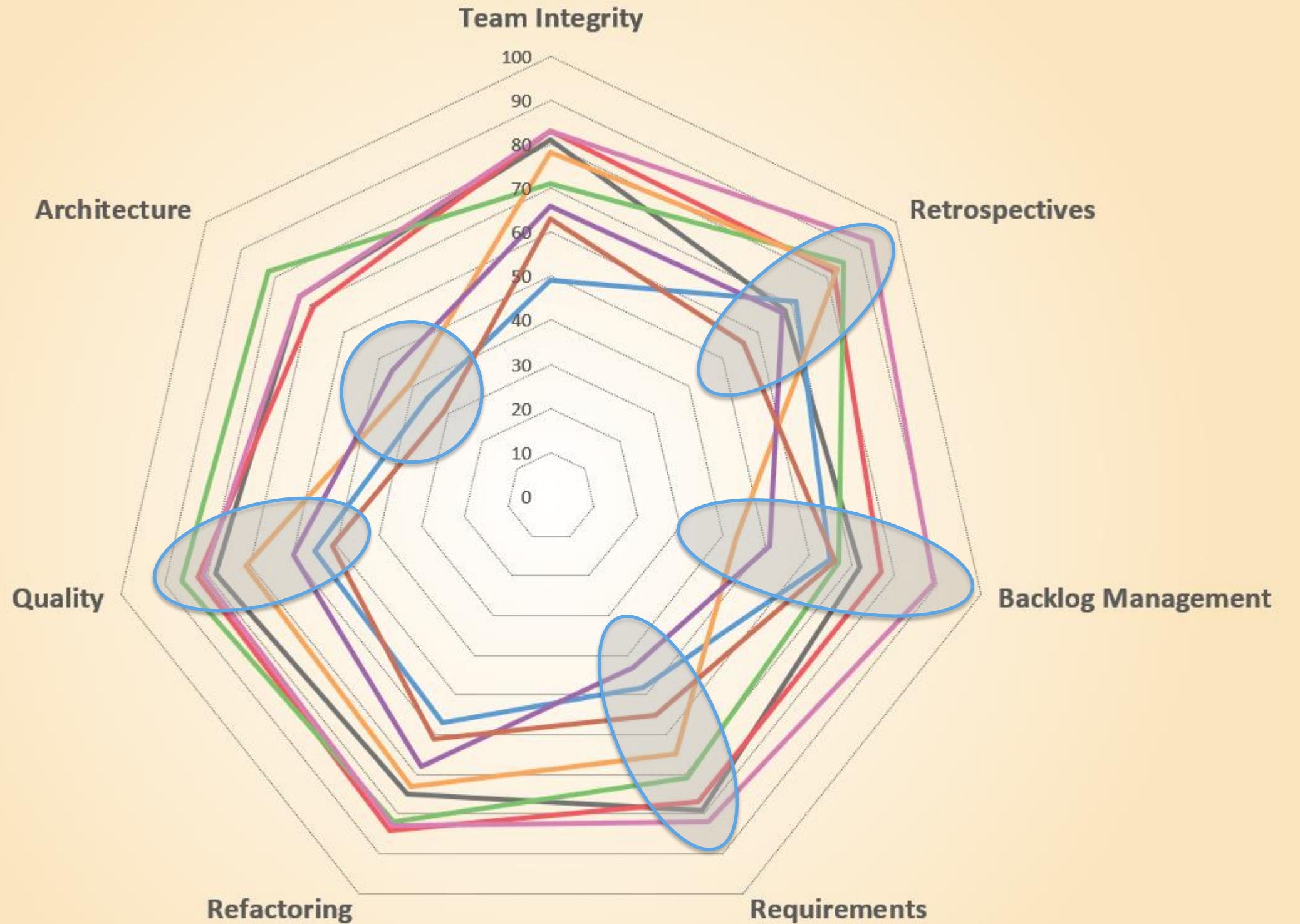
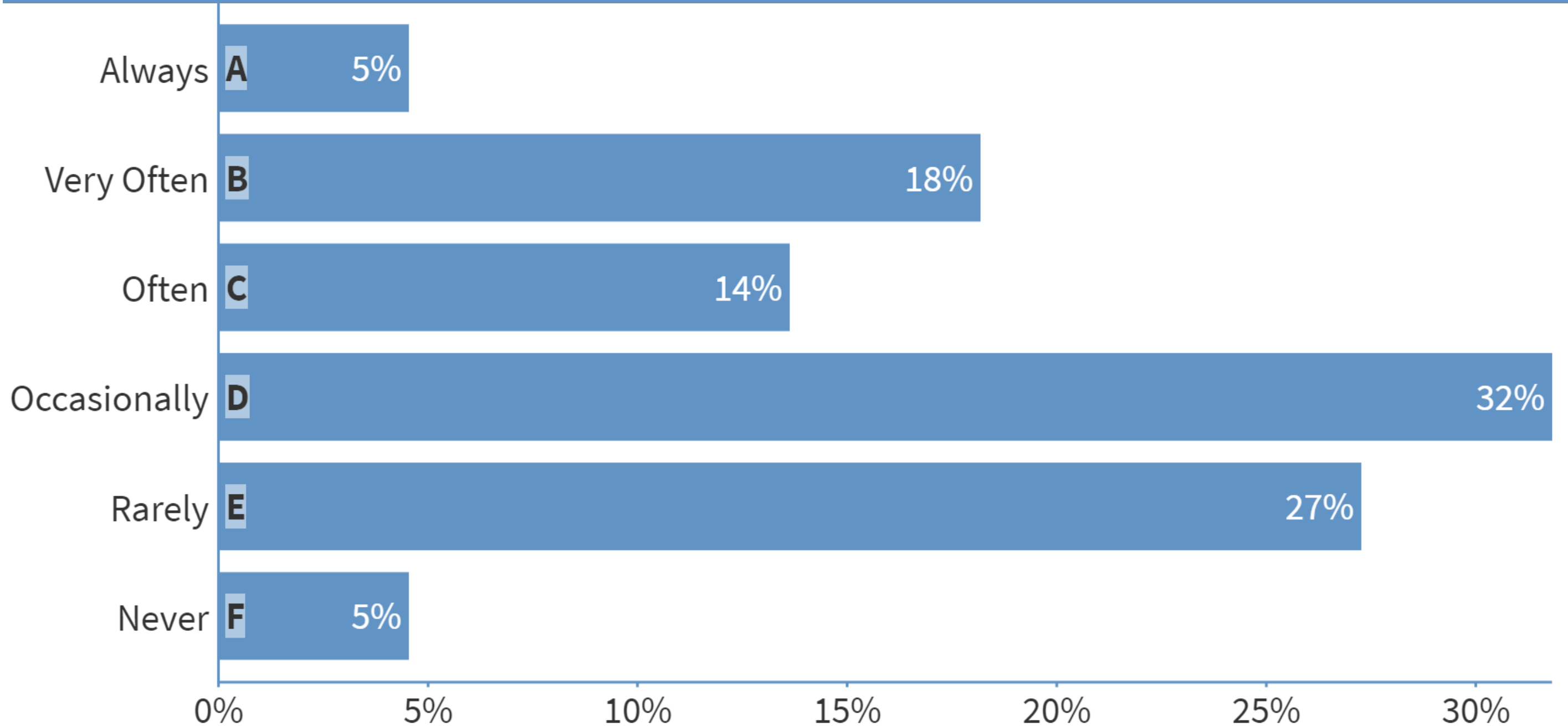7- Topic Results (2 Lowest)
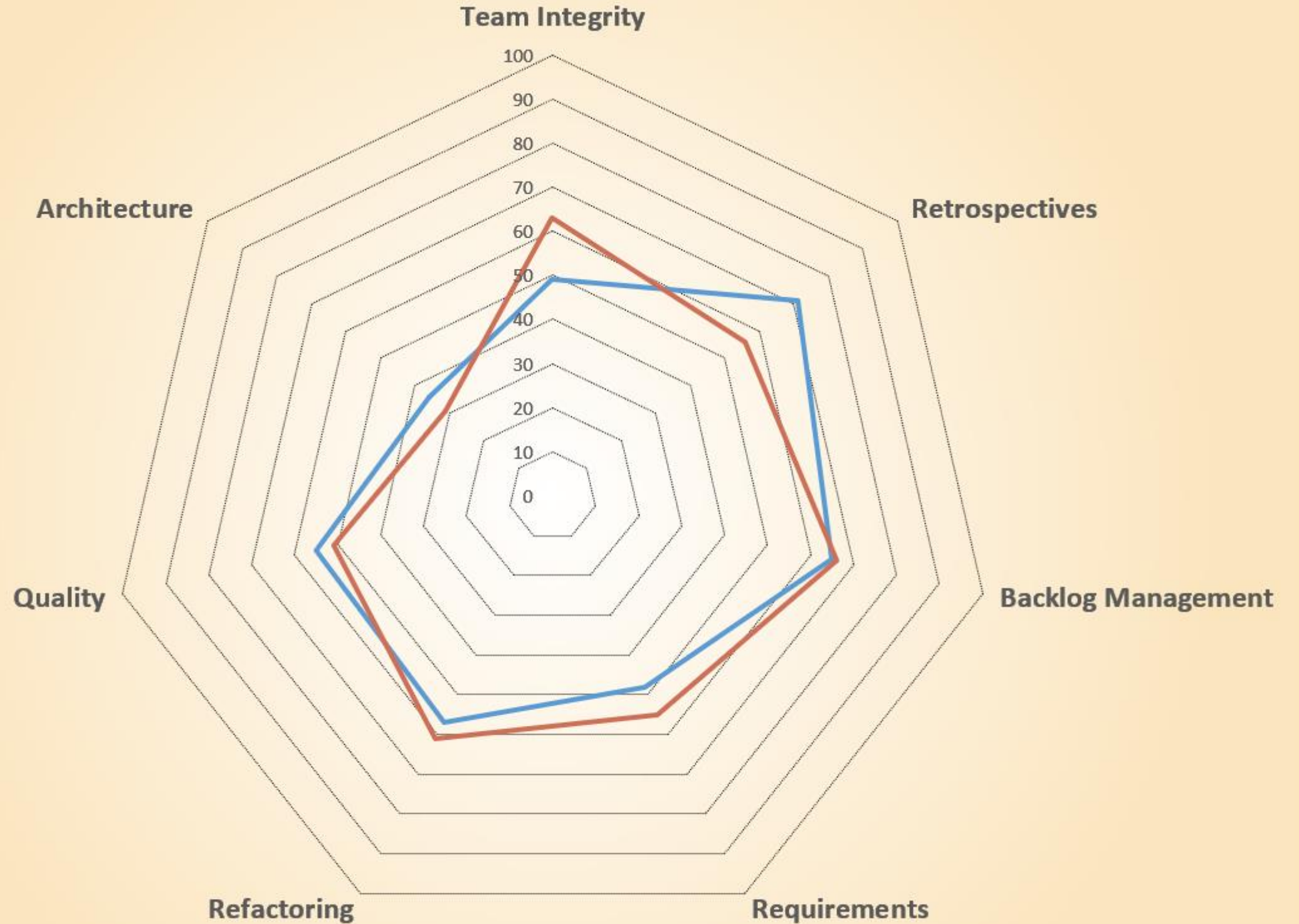
catalyte

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
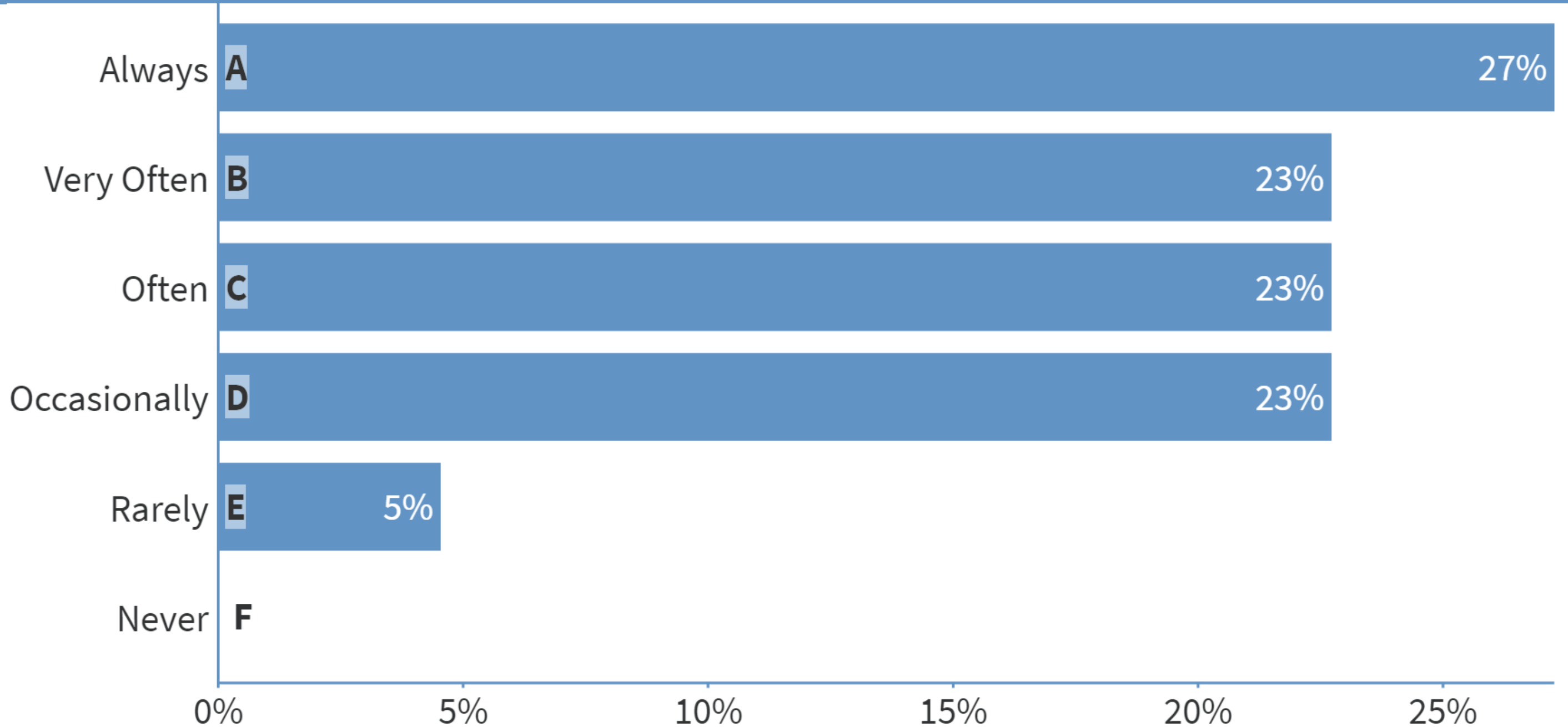5. Test results are used as the basis of code integration and deployment

3. Our product is stable.

Always	**A**	10%
Very Often	**B**	38%
Often	**C**	52%
Occasionally	**D**	
Rarely	**E**	
Never	**F**	

0%   10%   20%   30%   40%   50%

4. Tests are automated where possible

Always **A**	5%
Very Often **B**	18%
Often **C**	14%
Occasionally **D**	32%
Rarely **E**	27%
Never **F**	5%

0%   5%   10%   15%   20%   25%   30%

5. Test Results are used as a basis of code integration and deployment.

| Response		Percentage
Always	A	27%
Very Often	B	23%
Often	C	23%
Occasionally	D	23%
Rarely	E	5%
Never	F	

0%   5%   10%   15%   20%   25%

catalyte

QUALITY STATEMENTS

1. Our test <u>coverage</u> satisfies our test quality goals

2. Critical defects are <u>prevented</u> from reaching Production

3. Our product is <u>stable</u>

4. The tests are <u>automated</u> where possible

5. Test <u>results</u> are used as the basis of code integration and deployment
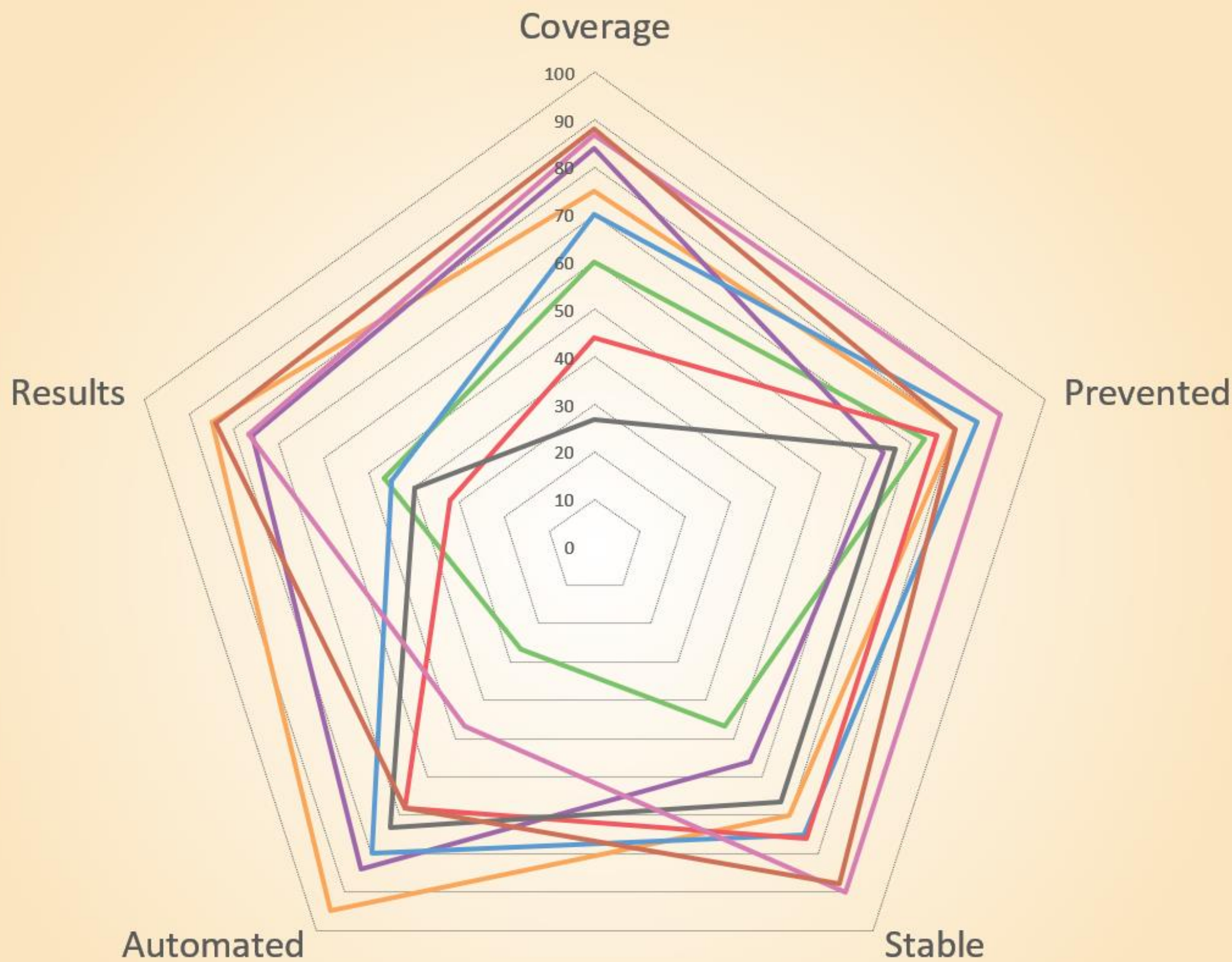
Quality Results
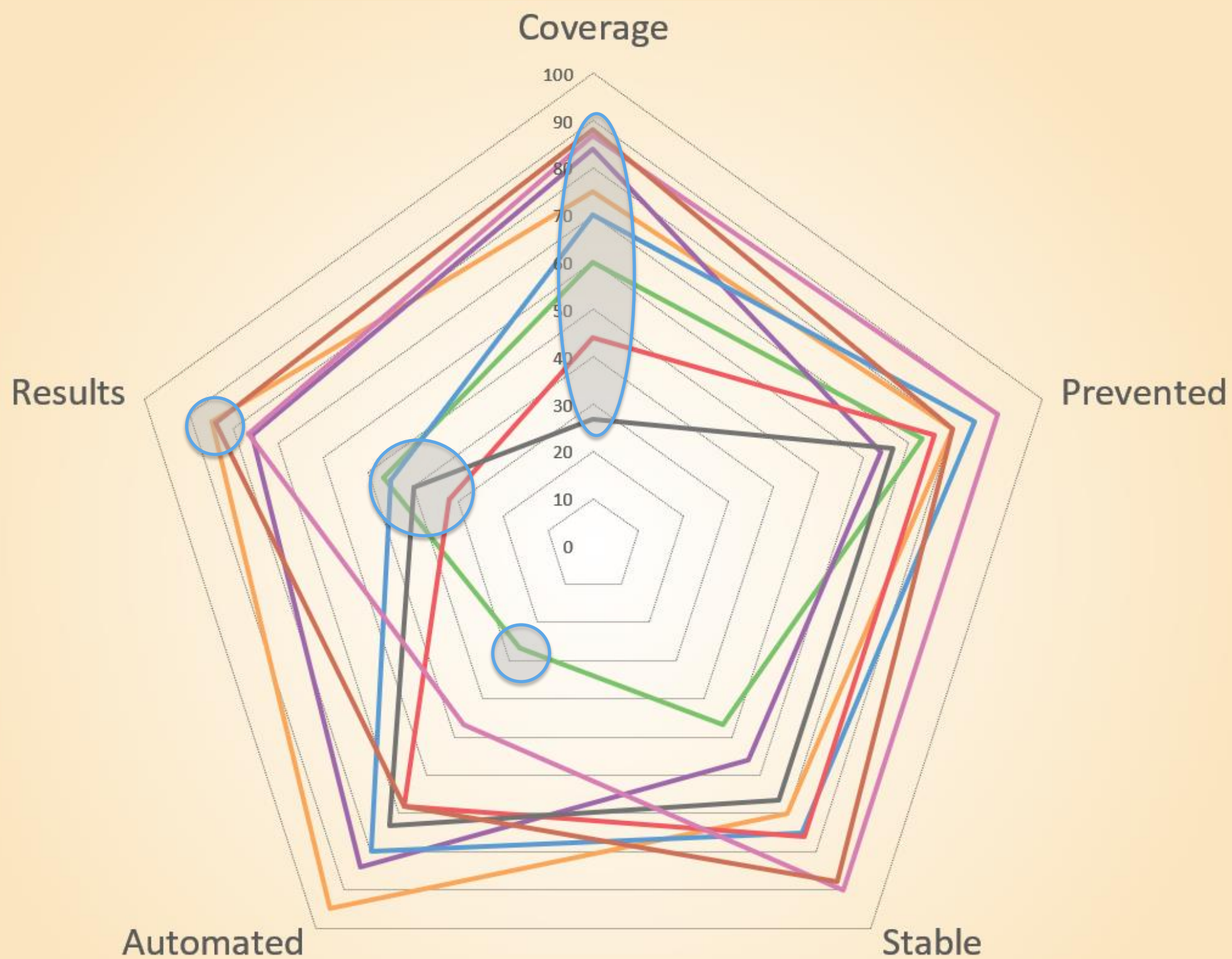
DRAWBACKS OF THIS ASSESSMENT TOOL

- 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. ❓

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

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

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

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.

Response	Percentage
True (A)	4%
More True than False (B)	29%
Neither False nor True (C)	38%
More False than True (D)	13%
False (E)	17%
Not Applicable (F)	
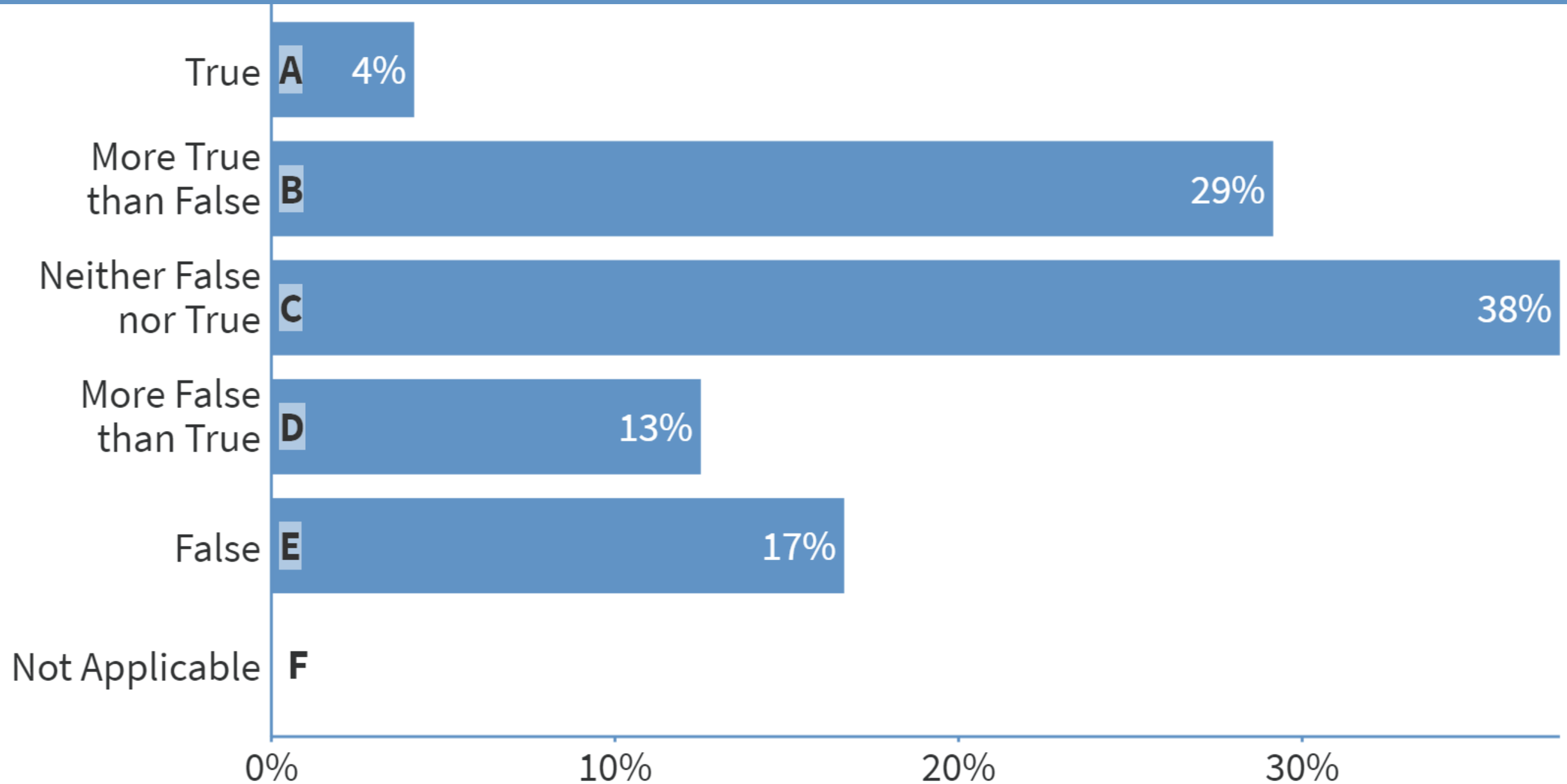
And We Still Get Spider Charts...

catalyte

Teamwork
Requirements
Planning
Technical Practices
Quality
Culture
Knowledge-Creating
Outcomes

And See Changes Over Time...

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.

Divergence from the Mean

	45	46	47	48	49	50	51
Value	0.2	-0.09	-0.33	-0.17	0.17	0.26	0.22

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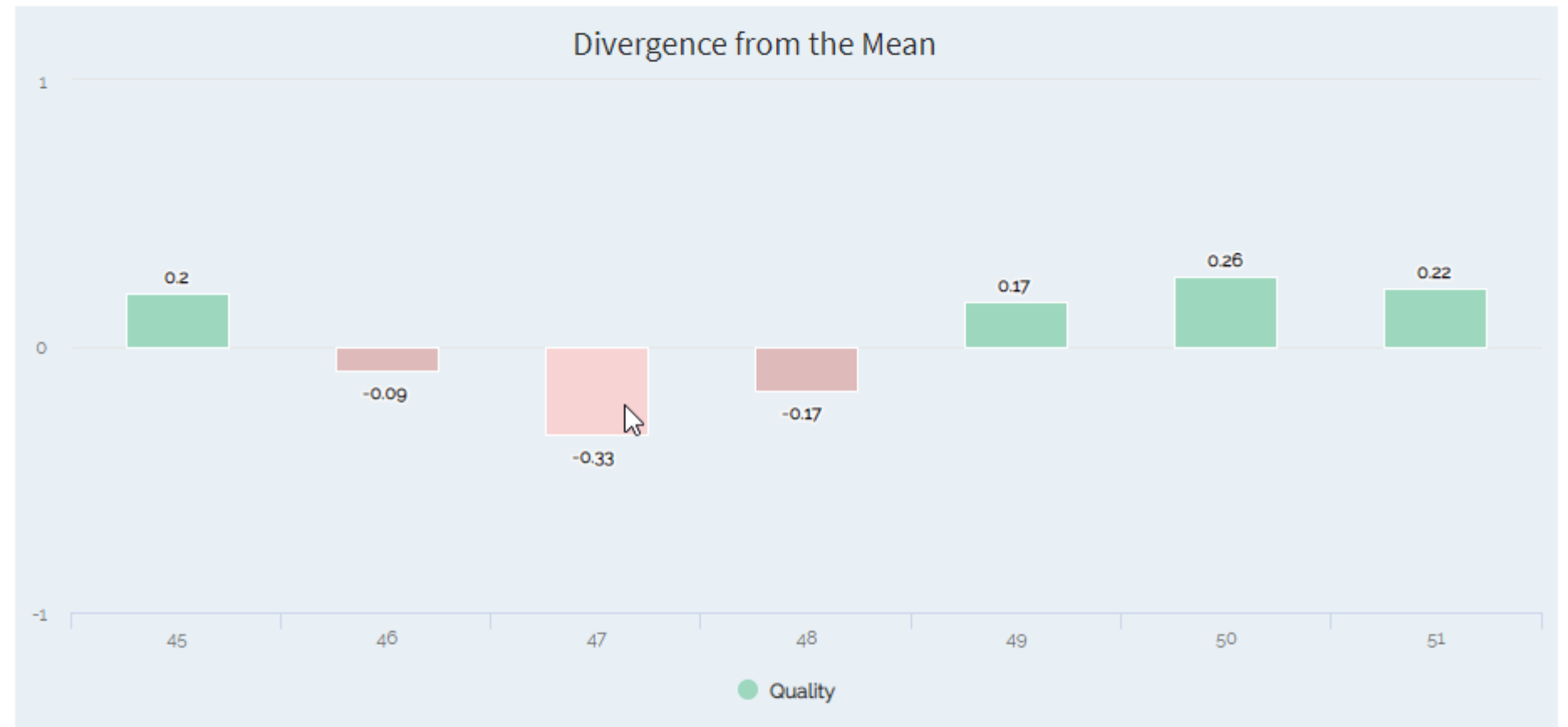
45 Product owners actively participate in the creation of the acceptance criteria for each feature.(0.2)

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

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

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

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

50 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)

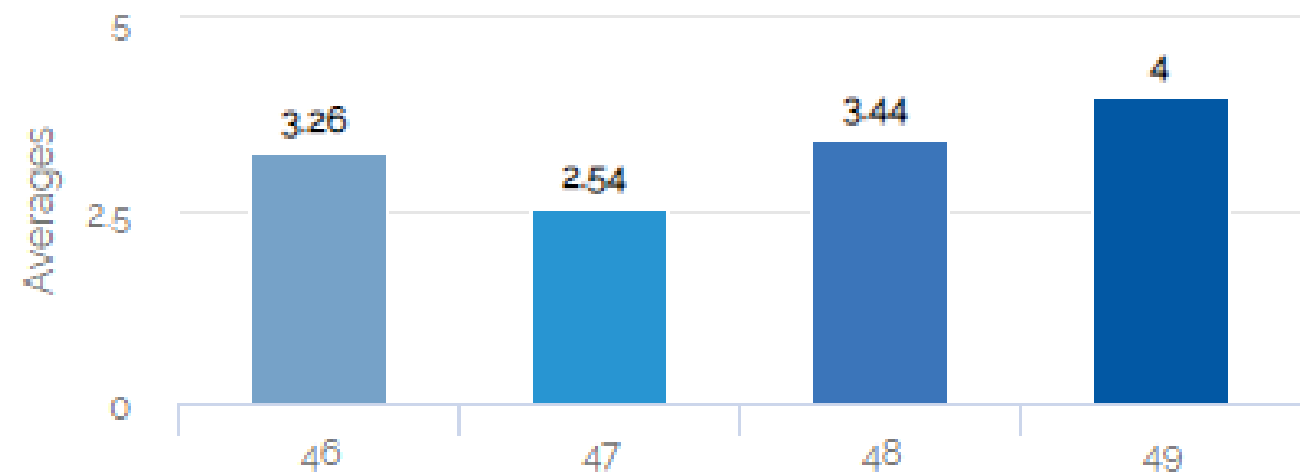# 51 The team has pre-defined and agreed-upon criteria for considering a feature done.(0.22)
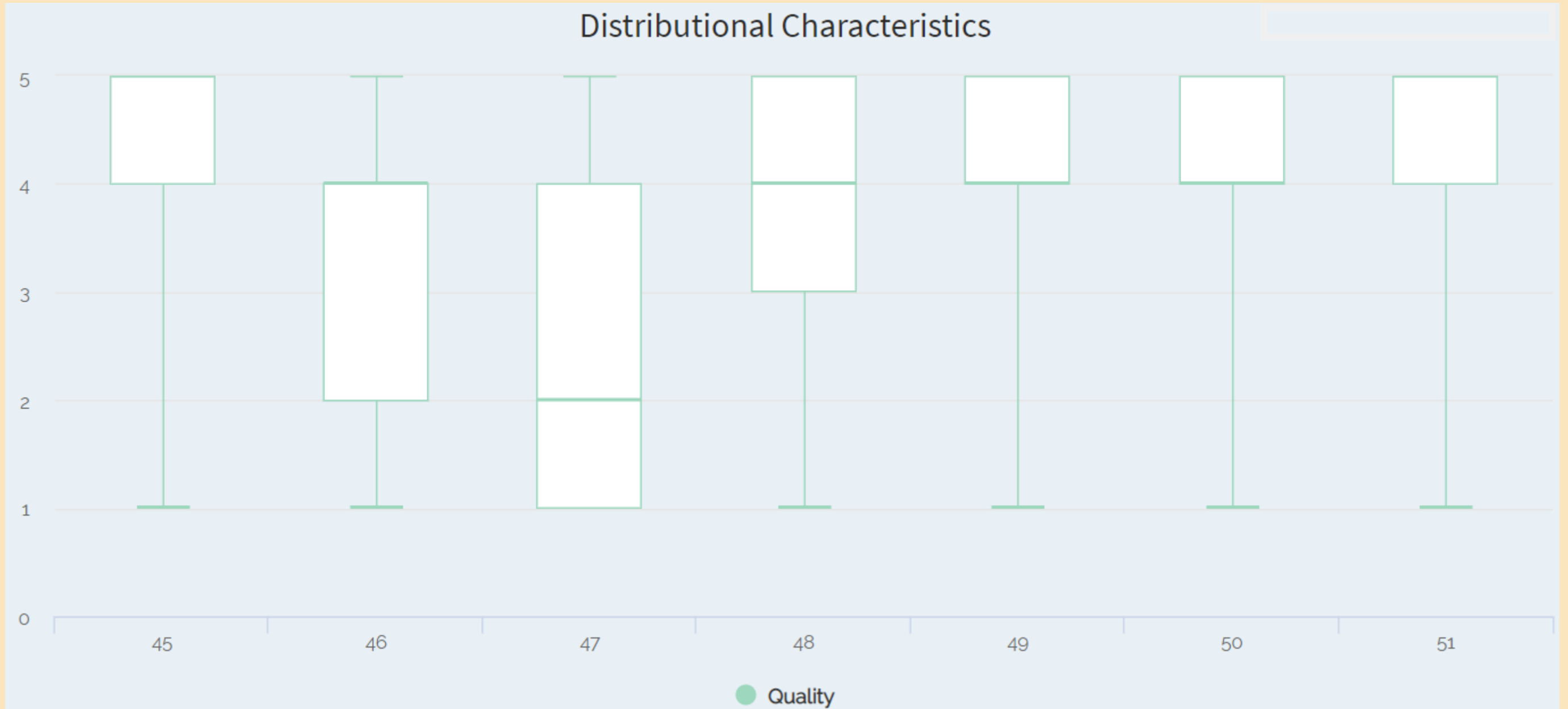
catalyte

SUMMARY

- Catalyte's "Road"
- Navigation
- Mapping
- Road Cones
- Next Stops

Questions?

catalyte

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