

DevOps adoption

Are you razing the silos or raising them...?



- Mayuresh Deshpande,
Head Agile Coaching Practice, Fiserv India

Agenda

- Introduction
- Context Setting
- DevOps drivers
- The Problem Statement
- The solution space...!



Mayuresh Deshpande



- Industry Experience: 24+ years
- Agile Experience: 11+ years
- Designation : Head of Agile Coaching Practice
- Agile Certifications: CSM, ICAgile Coach, SA, etc.
- *Current Role : Continuous Integration Continuous Learning*
- *Hobbies: Reading, Motor Biking, Watching Movies, Golfing, Theatre activities, Weekend Binging on Agile related reading*

Session expectations

- My role here is a learner ...
- This is not a Technical Session...
- These are my musings from my experience ...
- Whatever I'm saying, question it, research it and see if that make sense...
- Expect a lot of interaction...



What are silos? why do they exist?



Silo

'sɪləʊ/

noun

plural noun: **silos**

1. a tall tower or pit on a farm used to store grain. a pit or other airtight structure in which green crops are compressed and stored as silage.
2. a system, process, department, etc. that operates in isolation from others.



© www.pagerduty.com/blog/break-down-silos/



© https

Silos have evolved because **they permit the specialization required** to deal with the immense, growing complexity IT deals with everyday, but they have now become the biggest impediment to agility and quality

A silo mentality can occur when a team or department shares common tasks but **derives their power and status from their group**. They are less likely to share resources or ideas with other groups or welcome suggestions as to how they might improve.

Your Organizational Readiness



- Is your Organization already doing DevOps ?
- Is your Organization Structured Horizontally and Vertically?
- Is your Organization a Product Organization or Services Organization?

Yes

No

Yes

No

Product

Service



Need for DevOps



- **Top Drivers** for adopting DevOps
 - Continuous software delivery
 - Less complexity to manage
 - Faster resolution of problems
 - Faster delivery of features
 - More stable operating environments
 - Improved communication and collaboration
- **Top Benefits** experienced by adopting DevOps
 - Improved deployment frequency
 - Faster time to market
 - Lower failure rate of new releases
 - Shortened lead time between fixes
 - Faster mean time to recovery

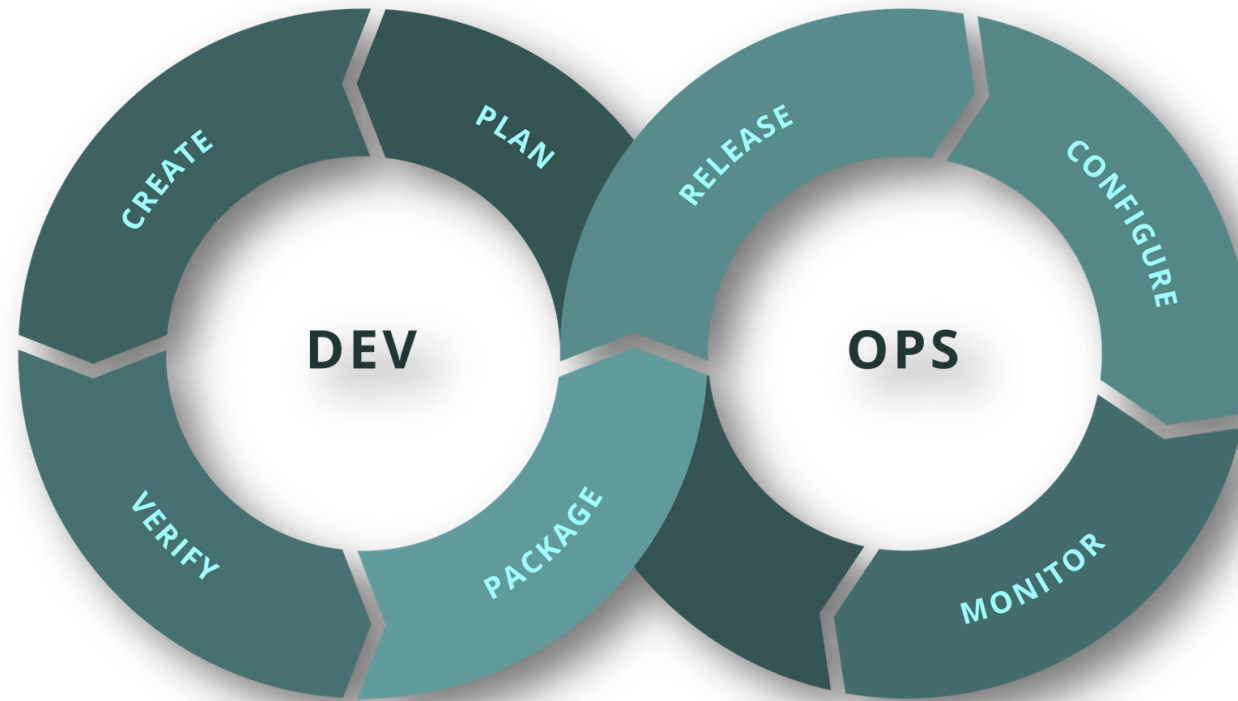


- Top Organizations using DevOps

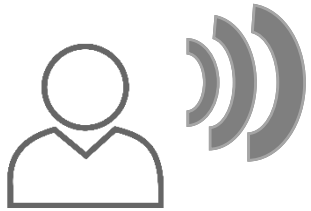


What is the most essential thing to achieve this...?

Common DevOps visual



Agile teams over DevOps



“ Agile is the thing of past, we should move to DevOps...”

“We used to do Agile in the first decade, this decade we should do DevOps...”

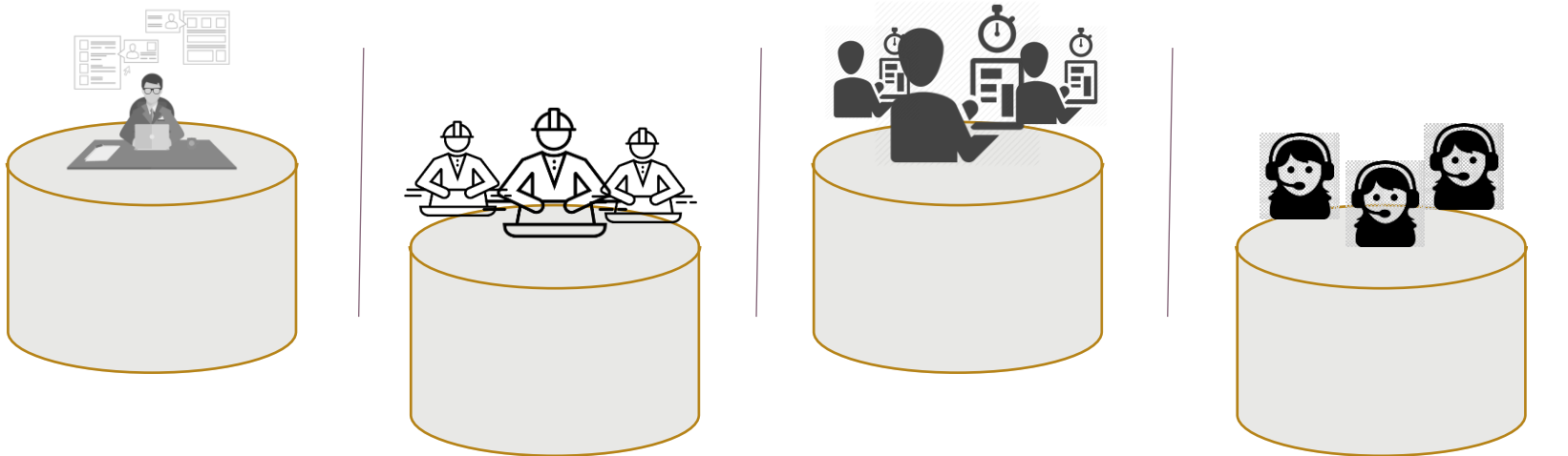
“... Agile hasn't given us the right improvement in productivity, DevOps would probably do that...”



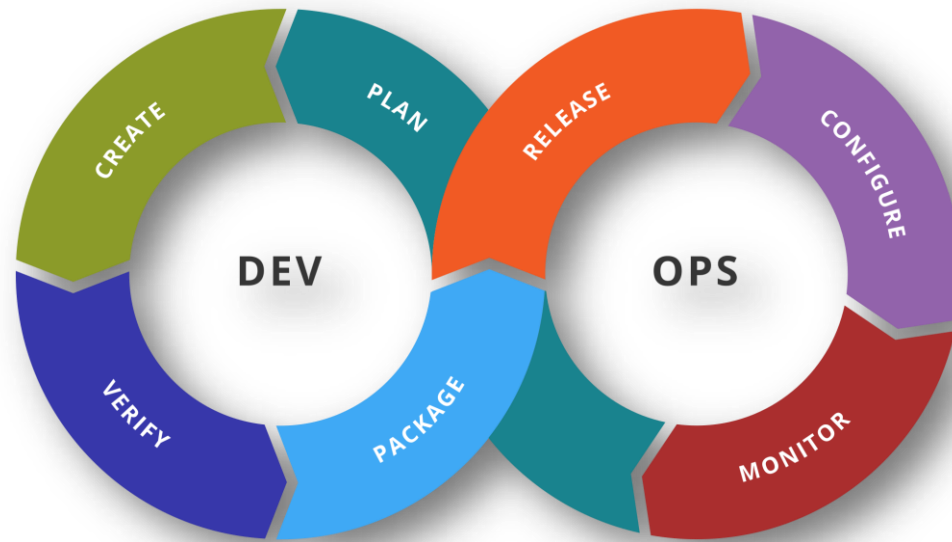
How do we raise Silos



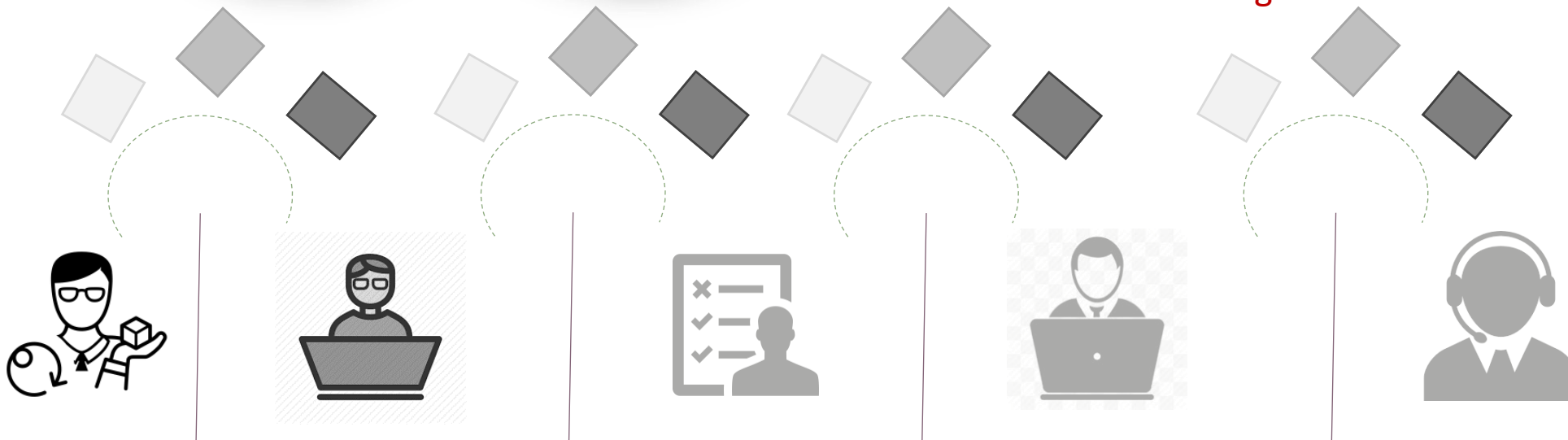
- Project Office –reporting for the “Senior Management” reporting to a Tower
- Multiple CoEs reporting to a Tower
 - Technology CoE
 - Testing CoE
 - User Experience CoE
 - ..
 - ..
- Product Organization reporting to a Tower
- Information Security associates reporting to a Tower (more often than not a corporate function)
- “Operational People...” where do they sit?? Not sure... But definitely reporting to a Tower



Continuing the anti-pattern



- Scrum Master + Manager + PMO (PMO)
- Product Owner / Business Analysts (Product)
- Developers (Tech CoE + Delivery)
 - Service Developers
 - UI Developers
 - DBAs
 - Framework developers
- Testers (Testing CoE)
 - Manual Testers
 - Automation Testers
 - “Non-Functional” Testers
- Once again... where do the operations folk sit?



Silos created by objectives

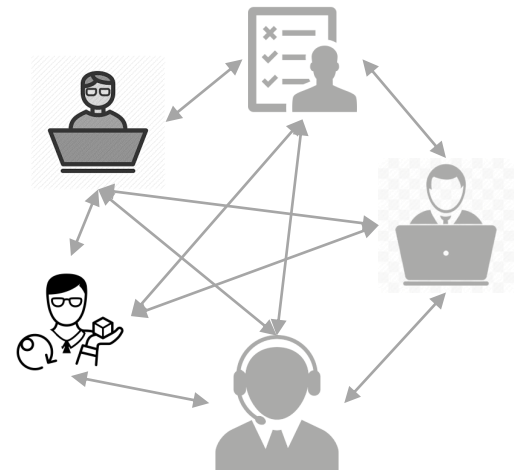
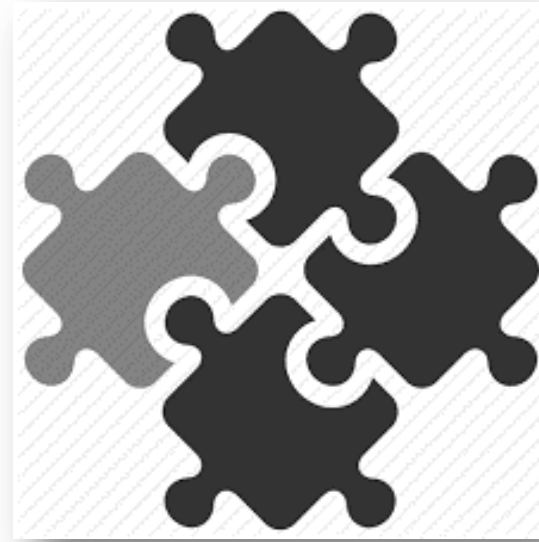


- Mutually exclusive goals (Dev vs Test)
- Having different understanding of the operating model (Dev vs Product)
- Executives working with different mindset than the delivery teams (Value vs productivity)
- Specialization is needed not isolation (CD – CICD – CICTCD – DevTestOps)



How do we raze the silos

- Better Alignment across teams
 - Frequent Interactions on Vision and Goals
 - Shared common minimum goals
 - More Matrix set up
- Metrics those really drive the right behaviors
 - SLI, SLA & SLO (Courtesy : Google)
 - MTTR
 - Throughput of Business Value per release
 - User adaption (monitoring analytics)
- Encourage tower collapsing roles
 - SDETs
 - SREs
- Institutionalize Feedback loop
 - Pair Programming
 - Product Demo / Desk-checks
 - Support execs into product discussions
 - SRE review



It's the right side of your grey matter...



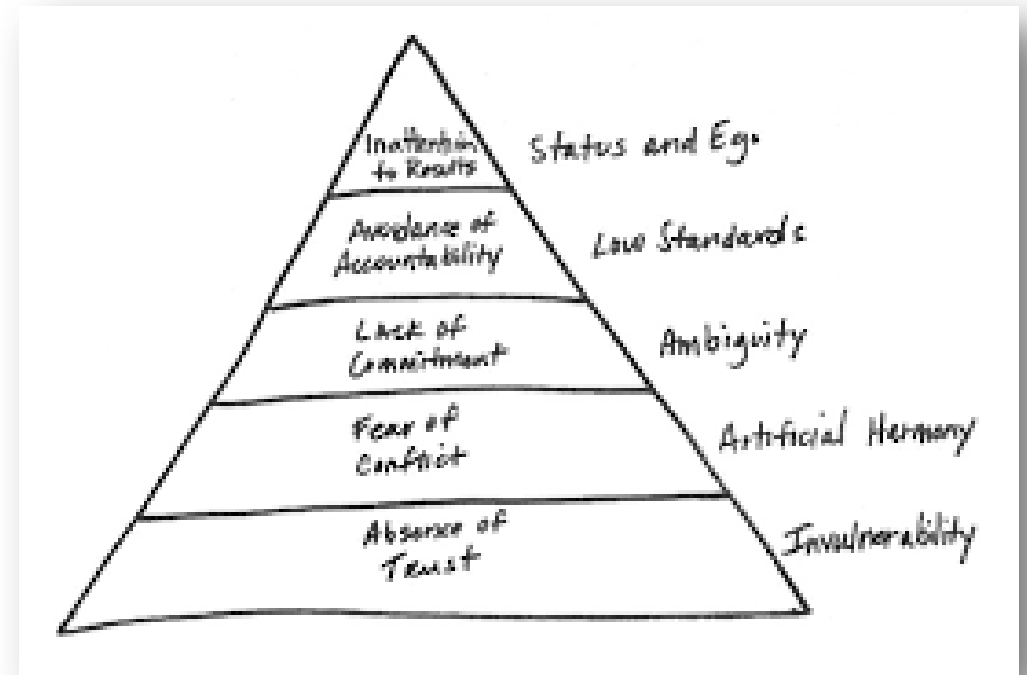
- NOT easy but Possible
 - It isn't about tools
 - It is about the culture
- Don't be slave of your tools choices
 - GitHub vs TFS
 - Angular JS vs Bootstraps
 - Selenium vs Tosca
 - ...
 - ...
- Get over the mental Silos
 - "Why should I test when I'm a developer..."
 - Functional testing over 'Non-Functional' testing
 - "Security is not my headache ..."
- Socialize Success stories - internally
 - Seeing is believing it
 - Nothing succeed like success



Be ready to reinvent



- Think Big – start small
 - Small batch size here too
 - Easier not to compromise when you are experimenting
 - Fail faster
- Be acutely aware of the 5 Dysfunctions (®Patrick Lencioni)
 - Every team *will* dysfunction because we are not Robots
 - Personal agenda vs team's success
 - Don't forget the context
- Reflect & Readjust
 - Get used to having Hard conversation (biggest Taboo)
 - Conflict driven Meetings



**“In the past, business success was all about size: The large eat the small.
Today, business success is all about speed:
The fast eat the slow.”**

Daniel Burrus
Futurist



Thank You