



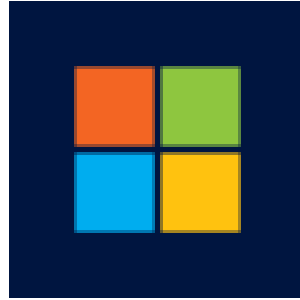
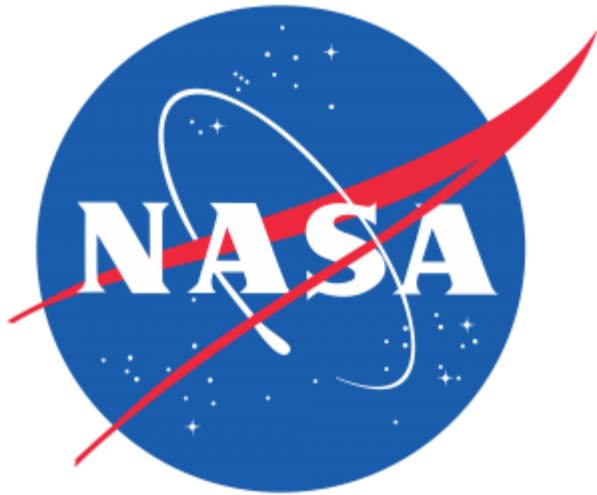
DevOps Best Practices

PEOPLE, CULTURE, TECHNOLOGY, AND PROCESS

Lesson Coverage & Key Objectives

- Who are the folks using the various solutions?
- DevOps implementation basics
- DevOps implementation checklist
- Patterns
- Process
- Culture
- Antipatterns
- Best practices

Who has been doing DevOps more than 5 years?



@WalmartLabs

DevOps Implementation Basics

- ❖ Automation may not be available for every step of a process
- ❖ Select projects and measurement metrics based on stakeholder input
- ❖ Assess current organization for SDLC, engineering and operations processes supporting technology-enabled business, create baseline metrics
- ❖ Initiate DevOps COE and/or organizational change management process
- ❖ Evaluation of current skillsets, and creation of educational roadmap
- ❖ Identify workforce strategy gaps and fill tactical gaps with vetted consultants
- ❖ Create educational awareness roadmap and identify patterns for custom courses aligned to the organizational DevOps strategy
- ❖ Create a checklist to ensure getting to the right finish line and to document any complicated process
 - ❖ First documented checklist was from Boeing after a plane crash
 - ❖ Your implementation checklist may vary and should be reviewed periodically
- ❖ Initiate tool selection POC/POV/Evaluation processes, align education roadmap

Lean Patterns

- ❖ Convergence of Process
- ❖ Efficiency is doing things right; effectiveness is doing the right things.”
Peter Drucker
- ❖ There have been many trends in the past 30 years
 - ❖ Kaizen
 - ❖ Agile
 - ❖ Process Theory
 - ❖ DevOpsMulti
 - ❖ Digital Transformation



Process Theory

- ❖ DevOps Builds on Process Theory
 - ❖ Don't do things that aren't required
 - ❖ Minimize waste (Reporting and Analytics)
 - ❖ Use appropriate tools and processes
 - ❖ Root cause analysis (Metrics)
 - ❖ Transparency (X-functional team)
 - ❖ Necessary Communication (ChatOps)
 - ❖ Mistake-proofing (Embedded QA)



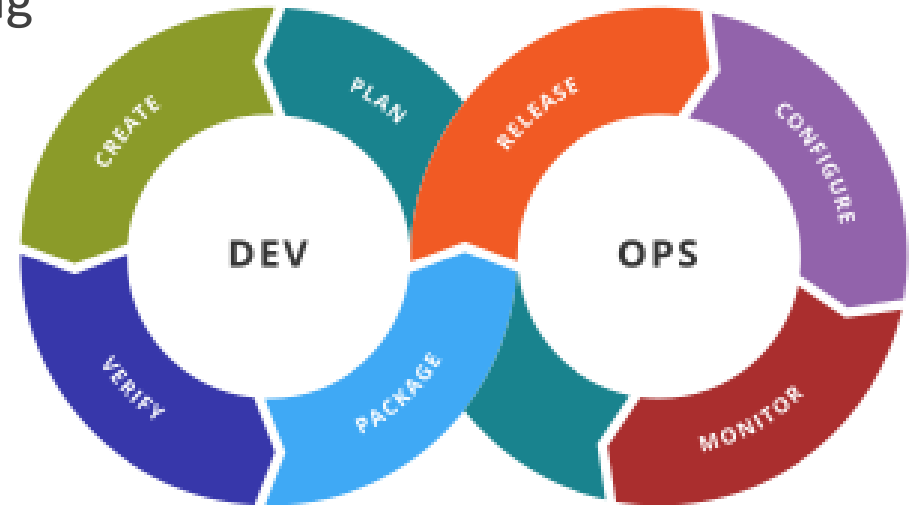
Tools Pattern

- ❖ DevOps Toolsets
 - ❖ Monitoring
 - ❖ Measurement
 - ❖ Agile
 - ❖ Scripting
 - ❖ Release Management
 - ❖ Software Configuration Management
 - ❖ Continuous Integration & Orchestration
 - ❖ Continuous & Automated Testing
 - ❖ Cloud, Containerization & Virtualization

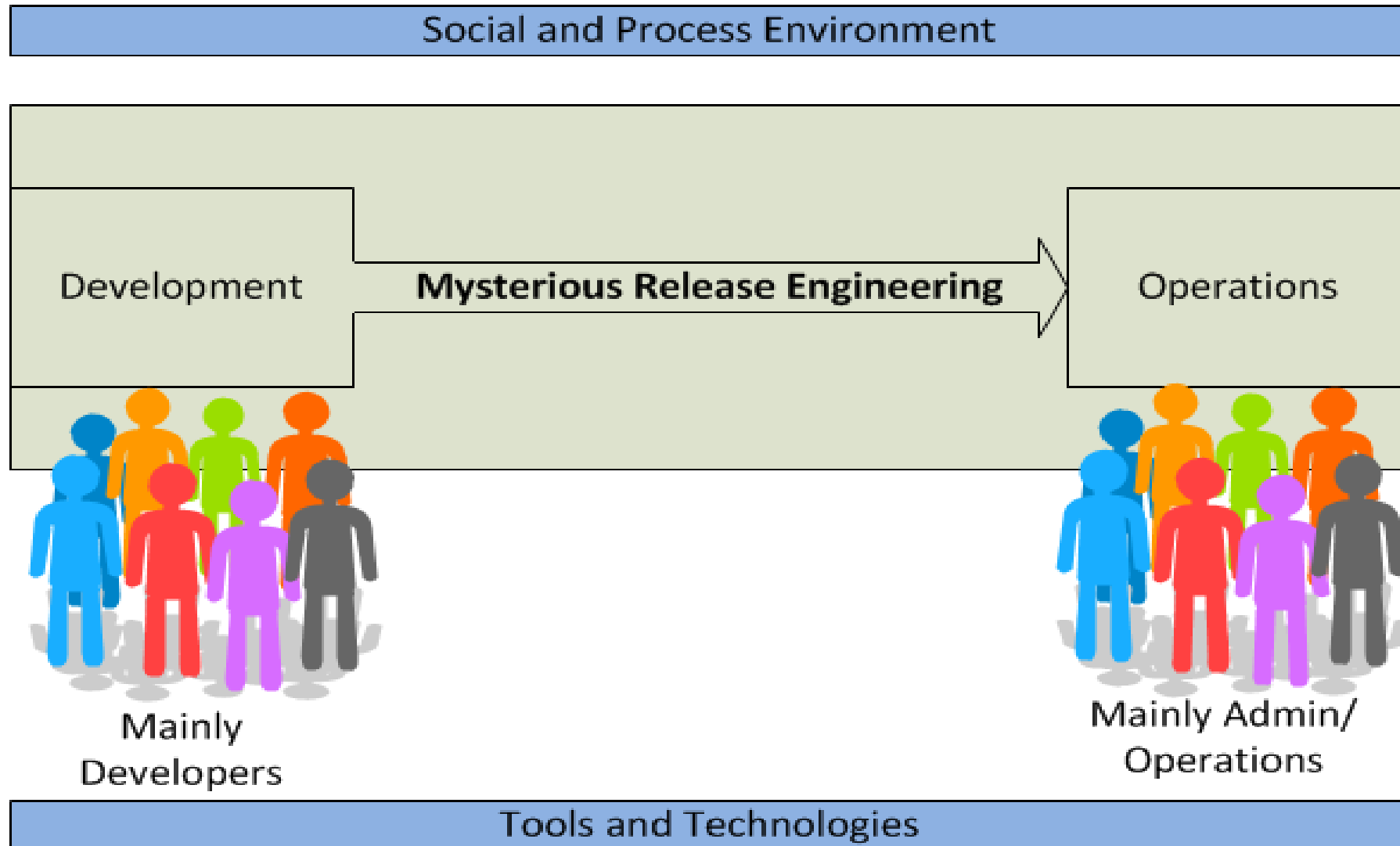


Tool checklist

- ❖ Automation
 - ❖ Configuration management
 - ❖ Cloud provisioning & management
 - ❖ Monitoring/Measurement/Analytics
 - ❖ Continuous quality/TDD/BDD/Continuous testing
 - ❖ Build/VCS/Binary Repositories
 - ❖ Orchestration & Pipelining
 - ❖ Continuous Integration & Delivery
 - ❖ Containerization
- ❖ Digital Data Supply Chain
 - ❖ BI, MDM, EAI, SOA, BigData, IoT
- ❖ Software Defined Network, Virtualized devices
- ❖ Cross functional team communication (ChatOps)
- ❖ Dashboard/Project management/Reporting/Notifications



Culture Alignment



Culture Antipatterns

- ❖ Silos
- ❖ Not involving the teams required
 - ❖ Executive
 - ❖ Development
 - ❖ Operations
 - ❖ Security
 - ❖ Business
- ❖ No leadership buy-in
- ❖ Lack of workforce strategy

Process Antipatterns

- ❖ Lack of understanding of current processes
- ❖ Failure to communicate
- ❖ Involving too large a team without coordination
- ❖ Heads down without sharing wins
- ❖ Boil the ocean
- ❖ Incorrectly loading the team
- ❖ Going it alone
- ❖ Environment that doesn't guard rail mistakes

Technology Antipatterns

- ❖ Wrong tools for the job or team
- ❖ Lack of knowledge sharing
- ❖ Overlapping tool functionality
- ❖ Choosing the tools before understanding the team skill level, technology acumen, and project priorities
- ❖ Lack of understanding of technology requirements across all involved SDLC, engineering, operations, and business teams

Tale of Two Audiences

❖ Developers

- ❖ Containerization – developing and architecting for...
- ❖ Micro-services architecture
- ❖ Service Oriented Architecture
- ❖ Scalability
- ❖ Unit test automation

❖ Operations

- ❖ Deployment
- ❖ Docker
- ❖ Virtualization
- ❖ Cloud
- ❖ Services infrastructure
- ❖ Monitoring/data collection
- ❖ Regression/Smoke test automation

Best Practices: DevOps

- ❖ Choose top 5
- ❖ Dashboard
- ❖ Evangelization
- ❖ ChatOps
- ❖ All brains in the game – “Jack Welch, CEO of GE”
- ❖ Small project teams
- ❖ Infrastructure self-service
- ❖ Automate the heck out of anything you could do more than once
- ❖ Workforce strategy

Best Practices: Process

- ❖ Capture knowledge gained along the way
- ❖ Standups that last 15 minutes
- ❖ Communicate scoreboard against metrics
- ❖ Document everything simply
- ❖ Kanban Boards – everyone should know the key dates and those should be easily visible.
- ❖ Peer reviews of processes, code, tools, dashboards, automation

Best Practices: Culture

- ❖ Publish what each team members knows and needs to know as part of the team as Kaizen
- ❖ Executive sponsorship
- ❖ Team should understand the business value of what the solution
- ❖ DevSecOps - everyone means business, quality, performance, dev, ops, security, dba, vendors, partners, consultants
- ❖ Celebrate small and large successes
- ❖ Reward people for supporting the changes in culture, process and technology
- ❖ Allow people to make guard railed mistakes

Best Practices: Metrics

- ❖ Set clear metrics
 - ❖ Reduce downtime by X
 - ❖ Find bugs Y faster
 - ❖ Team learns Z topics per quarter
 - ❖ Everyone attends A conferences or social learning events
 - ❖ Rollback occurs B faster
 - ❖ Defects reduced by C
 - ❖ Improve Cycle Time by V
 - ❖ Increase Velocity by W
- ❖ Refine based on metrics

Best Practices: Change Management

- ❖ Review end to end business processes currently enabled by technology and software engineering
 - ❖ Identify gates in approval, manual process, handoffs
 - ❖ Are they necessary?
 - ❖ Can they be
 - ❖ automated?
 - ❖ refined?
 - ❖ removed?
- ❖ Document
- ❖ Knowledge sharing - Kaizen

Summary

- Who are the folks using the various solutions?
- DevOps implementation basics
- DevOps implementation checklist
- Patterns
- Process
- Culture
- Antipatterns
- Best practices