Adopting DevOps Practices

Adopting Devops practices can be bottom up or top down or some combination of both. However unless there is some buy in from management and alignment with the strategic goals of the business, the development and operations team will continue to work in a disjointed fashion.

This session describes a number of strategies and approaches to implementing DevOps practices aligned to your business goals.
Introduction

• Why the need?
• What is DevOps?
• Adoption Strategies
  – Principles & Practices
• Where to begin
• Summary
Why is DevOps Important Now?

- Enterprises have young, nimble start-up competitors
- Agile software development is increasing
- More organizations are migrating to the cloud
- Consumers have “app” mentalities and expectations
- There is more data available to the business
- Time to market must increase

Is DevOps realistic for Enterprise IT?
Yes.

While enterprise IT faces legacy challenges, DevOps practices are universal and adaptable to different environments.
Businesses are challenged to meet time pressures with quality software

41% experience delays in integration, configuration and testing of applications*

51% applications rolled back due to quality issues escaping into production*

45% experience delays due to troubleshooting and fine-tuning issues in production*

Up to **4-6 Weeks** to deliver a simple code change**

** Forrester “Five Ways To Streamline Release Management”, 2011
Walls of Confusion
Blamestorming

Application failures lead to blamestorming

75% Ops is a Roadblock
75% of those surveyed said that Dev perceives Ops as a roadblock or only somewhat supportive of agile development

72% Dev is not Supportive
72% of the respondents said that Ops sees Dev as either not supportive or only somewhat supportive of their goals

The End Result?
91% say their business counterparts don’t consider IT to be a true partner

Source: ITSM/Serena.com 2012 study of IT professionals
Dev-Ops Impedance Mismatch

• Software Development and Release processes have too much built-in wastage
  – Handovers
  – Silos
  – Sign-offs

• Dev and Ops have conflicting priorities
  – Dev wants Change
  – Ops want Stability
Agile Delivery
What is DevOps?

DevOps is a cultural and professional movement that stresses communication, collaboration and integration between software developers and IT operations professionals.

- Responds to accelerated demand for quality software products and services
- Acknowledges the interdependence of IT functions
- Recognizes the need for cultural improvements
- Supports and leverages agile, lean and IT service management (ITSM) practices
- Encourages the use of automation
DevOps’ Goals

• Better IT alignment and business responsiveness
• Faster, smaller, more frequent releases
• Improved
  – Time to market
  – Quality of code, products and services
  – Productivity
  – Customer satisfaction
  – Employee satisfaction
• Less waste and fewer defects
• Lower long-term costs
Adoption strategies

• Top Down
  – Senior management direction
  – Organisational set up and change
  – Align all aspects and a planned path
  – Experiment - Learn - retry

• Bottom up
  – Pilot project
  – Cross functional team cooperation
  – Create the new way
  – Bank the improvements – capture learnings
  – Repeat and communicate success as well as failures
DevOps cannot stand alone
Values

• CAMS
  – Culture
  – Automation
  – Measurement
  – Sharing
DevOps Principles
“The 3 ways”

• The first way - Flow of work (left to right)
  – Understanding and improving flow

• The second way
  – Amplifying feedback loops (Right to left)

• The third way
  – Continuous experimentation and learning (Kata)

• Ref – The Phoenix project (Gene Kim et al)
Practices – First way

• Continuous
  – Integration
  – Delivery
  – Deployment
• Value stream mapping
• Kanban
Practices – Second way

• Examples of feedback loops
  – Peer review of production changes
  – Dashboards
  – Production logs
  – Post mortems
  – Incident / problem / change mgt data sharing
Practices – third way

- Allocate time for improvement
- Create culture that fosters
  - Continual experimentation
  - Taking risks and learning from failure
Right Priorities

1. People
2. Process
3. Technology

Culture

Improve communications

Promote collaboration

DevOps
Where to begin

• Figure out the ‘why’
  • Management buy in – has to be around the Why – DevOps is not the Why in itself

• Simplification of complex processes
• Automation of repeatable tasks / processes

• Eight Steps to transforming your organisation
  – John P Cotter

• Tackle the cultural shift

• Pilot approach
  – Selection Criteria – aligned to business goals
  – Follow on – consolidate
  – Anchor results
  – If you cant change the culture - create a new one 😊
Summary

- Lock in the Why
- Agree an incremental approach
- Address the Culture and Values
- Experiment and learn from experience - always looking to improve all areas
What is the DevOps Foundation course?

- A sixteen (16) hour certification course that provides foundational knowledge of DevOps including
  - DevOps vocabulary
  - DevOps values and practices
  - The relationship between DevOps, Lean, Agile and ITSM
  - The principles of DevOps automation
  - Guidance for nurturing a DevOps culture

_The course and certification examination are accredited by the DevOps Institute._
DevOps Dimensions

The best tools to address each of the 4 dimensions of DevOps Practices.

1. Plan & Track
   Collaborative planning and tracking tools are a must if you want disparate groups to work together across the organization.

2. Dev & QA
   Dev and QA teams need to work together so a standard set of tools that encourage a single source of truth is a must.

3. Release & Deploy
   Frequent Release and Deploy requires automation of both infrastructure provisioning and software configuration and deployment.

4. Monitor & Optimize
   You need to be able to react quickly and fix production issues so that the business can have trust in the DevOps process and you constantly learn and improve.
Plan & Track
All types of work planned and tracked using a suite of enterprise collaboration tools

- JIRA and JIRA Agile
- Confluence & HipChat
- Wallboards
- JIRA Portfolio
Dev & QA
Tools to allow Dev and QA teams to work together

Productivity
Improve the speed and transparency of all development and QA activity

JIRA IDE Integration
Bamboo
Zephyr for JIRA
JIRA & JIRA Agile
Stash & Bitbucket
Release & Deploy

Frequent Releases Requires Automation

Release
- Release & Configuration Management
- Automated Deployment

Deploy
- Software Defined Infrastructure
- Automated Provisioning

inedo

CHEF
Monitor & Alert

Monitor application performance from the point of view of the end-user