DevOps and TFS

A Framework for Enterprise Agility, Experimentation and Innovation

Joe Altovar  David Votaw
What you get from this Briefing…

- DevOps Recap
- DevOps and TFS
- Demo

PRESENTATION GOAL: To Provide You More Awareness of Emerging IT Trends and What Our Organization is Doing
Who are these guys?

- Joe has 20 years in Business and IT with a broad range of architecture, design, development and process improvement experience. He has held various positions supporting Northrop Grumman’s internal and external lines of business for the past 10 years. He is currently an IT Program Manager. Prior to joining Northrop Grumman, Joe worked as a software developer, chief systems engineer/integrator to management positions for companies such as Accenture, Deloitte Consulting and various tech start-ups. He is a certified Scrum practitioner and has a BA in History from UCSD and a Master's degree from UCSD’s School of Global Policy and Strategy.

- David is a Developer with Northrop Grumman Corporation (NGC) and has been with the company for one year. Prior to working for Northrop Grumman he worked at HP for ten years with a variety of roles including, Operations Support, Team Lead, Engineer, and a Developer. David has a Bachelor of Science (BS) in Information System Security (ISS), a Master of Science (MS) in Software Engineering, and is in process of obtaining a Doctorate of Computer Science (DCS) in Information Assurance.
DevOps Recap
DevOps Definitions…

DevOps is a software development method that stresses communication, collaboration and integration between software developers and Information Technology (IT) professionals. DevOps is a response to the interdependence of software development and IT operations. It aims to help an organization rapidly produce software products and services.

“DevOps is, in many ways, an umbrella concept [introduced in 2009] that refers to anything that smoothes out the interaction between development and operations”

Problem Statement Bottom-Line

IT Before DevOps...

I am Dev!
Let’s change everything

I am Ops!
Don’t change anything

Source: W. Pullen, The Value of Application Delivery & DevOps
Solution: A Different Way of Thinking and Operating…

- Collaborative, Dev and Ops teams combine or working closely together
- Continuous improvement across the deployment pipeline targeted at producing something of value to a user or organization
- Feedback-driven, shared insight into architectural health and performance measures at each stage and end-to-end
The Pay-off

Based on a 2016 Puppet Labs study Highly effective DevOps organizations have the following impact on the business:

- **200X** Frequent Deployments
- **24X** Return to Normal Operations
- **3X** Change Failure
- **22X** Unplanned Work & Rework
- **50%** Security Remediation Time

Source: 2016 State of DevOps Report, Puppet Labs

High performing teams offer secure and reliable products, services and capabilities quicker, faster
DevOps Lifecycle

Global Product Data Interoperability Summit | 2016

Continuous Planning

Agile/Continuous Integration Process

Continuous Integration

Continuous Deployment

Continuous Delivery
DevOps Lifecycle Activities Example

1. Defects – Existing Releases
   - (1a) Defects
   - (1b) Enhancements & Defects – New Release

2. Dev Team
   - (2) Dev Team

3. Source Code
   - (3) Source Code

4. Version Control
   - (4) Version Control

5. Source Code Label/Build
   - (5) Source Code Label/Build

6. Static Code Analysis
   - (6) Static Code Analysis

7. Setup Test Databases
   - (7) Setup Test Databases

8. Run Automated Tests
   - (8) Run Automated Tests

9. Code Coverage Analysis
   - (9) Code Coverage Analysis

10. Create SDK Documentation
    - (10) Create SDK Documentation

11. Create CD Installation
    - (11) Create CD Installation

12. Deploy to Test Cluster
    - (12) Deploy to Test Cluster

13. Update Build Results/Artifacts
    - (13) Update Build Results/Artifacts

14. Update Build Dashboard
    - (14) Update Build Dashboard

15. Build Notification Emailed
    - (15) Build Notification Emailed

Continuous Planning

Continuous Integration

Agile/Continuous Integration Process

Continuous Deployment

Continuous Delivery
DevOps Systems Thinking: Patterns and Practices

PEOPLE
- Autonomous Teams
- Integrated Teams
- Joint Meetings
- Job Rotations
- End-to-End SDLC Engineers

PROCESS
- Minimum Viable Product
- Automated Builds
- Small Batches
- Automated Testing/Test Everything
- Continuous Delivery/Release Automation
- Canary Roll-Outs
- Chaos Monkey

CULTURE
- Collective Ownership
- Collaborative
- Continuous Experimentation
- Hackathons

TECHNOLOGY
- Developer Self-Service
- Infrastructure as Code
- Platform as a Service
- Continuous Monitoring/Monitor Everything
- Integrated ecosystems

Multi-dimensional Perspective Required To Be Effective

Netflix: Randomly terminates a running service in a group to see how well the system tolerates the failure
DevOps Technical Reference Model

Global Product Data Interoperability Summit | 2016

**Configuration Management**
- TFS
- GIT
- SVN
- Mercurial
- Sonatype Nexus

**Continuous Integration & Delivery**
- Visual Studio/TFS
- Jenkins
- Maven
- Ant

**Static V&V – Continuous Delivery**
- TFS
- Sonar
- Coverity
- PMD, Findbug
- HP Fortify

**Behavioral V&V – Continuous Delivery**
- TFS, Eggplant
- JUnit, CPPUnit
- JMeter
- LoadRunner
- Selenium
- Valgrind

**Infrastructure Management**
- Microsoft System Center
- Red Hat Satellite Server
- Vagrant, Docker

**Continuous Deployment**
- TFS
- Chef
- Puppet
- Ansible
- Docker

**Monitoring**
- Tivoli
- OpenView
- ArcSight
- ManageEngine
- Nagios

**Continuous Planning/ Lifecycle Mgmt**
- TFS
- Jira
- Rally
- e.POWER

**Beware: Tools rapidly emerging and evolving as DevOps matures**
DevOps and TFS
Team Foundation Server (TFS)

A platform for integrating, collaborating, experimenting and deploying software products Better, Cheaper, Faster
Team Foundation Server (TFS) Capabilities

Project Start-up
Collaboration Site
Requirements Gathering

Source: DevOps and Application Lifecycle Management, Microsoft Corporation,

Incubated organically and growing within Northrop Grumman

Code Check-in
Microsoft Test Manager /
3rd Party Test Software

Source Control
Continuous Integration
Release Management
Configuration Management

Reporting
MS Office Integration
Demo
Summary
On the path to DevOps

Global Product Data Interoperability Summit | 2016

- **Continuous Planning**
  - Agile Project/Portfolio Management Tools

- **Continuous Integration**
  - Development activity
  - Integrate and test software often using automated version control & management tools
  - Early feedback to developers

- **Continuous Delivery**
  - Continuous Integration+
  - Deliver working software to next phase
    - QA and V&V
    - Security Testing

- **Continuous Deployment**
  - Continuous Delivery+
  - Deploy integrated and tested product to production
  - Monitoring and Incident Management Tools

- **TFS as an enabler for future work within Northrop Grumman**
Graphics and References

- **DevOps Considerations with a Focus on Large Enterprise, Feb 27th, 2014** by Stephany Bellomo, Software Engineering Institute, Carnegie Mellon
- **The Value of Application Delivery & DevOps**, 12/10/15 by Wesley Pullen, Electric Cloud