Increased Code Quality With DevOps

David Votaw
NGC
Who is David Votaw?

• **Background**
  • Software Developer - 13 years
  • 86% complete with a Doctorate of Computer Science (DCS) in Information Assurance

• **Current Assignments**
  • Taking teams to the next level of DevOps
  • Research and innovation

• **Fun**
  • Family
  • Video games
  • Road trips
  • Heavy metal \m/
Who is David Votaw? Cont.

Global Product Data Interoperability Summit | 2018
• This presentation will give an overview of DevOps practices including:

  • Test Driven Development (TDD)
  • Continuous Integration (CI)
  • Static Code Analysis (SCA)
  • Functional Testing
  • Continuous Deployment (CD)
  • Infrastructure as Code (IaC)
  • Virtualization and Parallelization
  • Monitoring and Metrics
DevOps

Developer and Operations (DevOps)
What is DevOps?

According to DevoPedia, “DevOps is the coming together of both development and operations teams into a coordinated workflow such that collaboration and productivity are improved to meet shared business goals. Building on Agile and Lean, DevOps enables the business to respond to changes and meet customer needs faster. Tools and automation are necessary enablers. Practices such as Continuous Integration and Continuous Delivery are often followed.”
**DevOps Process**

*Delivering Value Through Collaboration*

- **Development**
- **Operations**
- **Feedback**

### Example Environment Setup

- **Development Environment**
  - Continuous Integration
    - Build and Unit Tests
- **Integration/Test Environment**
  - Continuous Testing
    - Functional Tests
- **Validation Environment**
  - Continuous Delivery
    - Acceptance Tests
      - End-to-End Demonstrations
  - Continuous Deployment
- **Production Environment**
  - Continuous Monitoring

**Continuous Improvements**

**Right-to-Left Feedback**
Test Driven Development (TDD)
TDD Cont.

- Works well with Agile teams and short development cycles
- Keep the unit or module small
- Steps
  - Write the test first
  - Write only the code that is needed to pass the test
  - Refactor to organize and eliminate duplication
  - Repeat
- Focus on software quality
- Avoid building dependencies between test cases
TDD using NUnit

- **Arrange**
- **Act**
- **Assert**
- **Cleanup**

Test Fixture -- Causes NUnit to recognize this as a test class.

Test -- Causes NUnit to recognize this as a test method.

Actual API call we are trying to test.

Assertions are used to check values and conditions.
Benefits and Limitations of TDD

Global Product Data Interoperability Summit | 2018

**Benefits**
- Develop code that reflects new business requirements
- Higher productivity by focusing on one test at a time
- Tidier and higher quality code
- Helps keep unused code out of the system
- Built-in regression testing as part of a Continuous Integration process
- Fewer bugs
- Living documentation

**Limitations**
- Requires time and effort up front
- Difficult to write good test cases
- Takes time maintain test suites
Continuous Integration (CI)
CI Process

Global Product Data Interoperability Summit | 2018

- **Continuous Planning**
  - (1a) Defects – Existing Releases
  - (1b) Enhancements & Defects – New Release

- **Continuous Integration**
  - (2) Dev Team
  - (3) Source Code and unit tests
  - (4) Version Control
  - (5) Source Code Label/Build
  - (6) Static Code Analysis
  - (7) Setup Unit Test Databases

- **Continuous Deployment**
  - (8) Run Automated Unit Tests
  - (9) Code Coverage Analysis

- **Continuous Delivery**
  - (10) Create SDK Documentation
  - (11) Create CD Installation
  - (12) Deploy to Test Cluster
  - (13) Update Build Dashboard
  - (14) Update Build Results/Artifacts
  - (15) Build Notification Emailed
CI – Source Control

Source code is structured so each release is branched and maintains its own history.
CCNet polls for changes every 5 minutes. If a change is found it will wait an additional 5 minutes of inactivity before starting a build.
CI – Label, Get, and Compile Code

NAnt script is run to label the “build” directory in source control with the build number (e.g. 9.1.0.538)

Compiled code results for all projects in the solution on the build machine.
Static Code Analysis (SCA)
SonarQube is used to scan the code to check for good coding practices.
Fortify is used to check code for good security practices.

### Executive Summary

Issues Overview

On Jul 20, 2018, a source code review was performed over the Client code base. 527 files, 36,761 LOC (Executable) were scanned and reviewed for defects that could lead to potential security vulnerabilities. A total of 293 reviewed findings were uncovered during the analysis.

<table>
<thead>
<tr>
<th>Issues by Folder</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>0</td>
</tr>
<tr>
<td>High</td>
<td>27</td>
</tr>
<tr>
<td>Medium</td>
<td>5</td>
</tr>
<tr>
<td>Low</td>
<td>254</td>
</tr>
<tr>
<td>False Positive - Access Control</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - Cross Site Scripting</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - Misc</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - Null Dereference</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - Path Manipulation</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - Privacy</td>
<td>3</td>
</tr>
<tr>
<td>False Positive - Random</td>
<td>1</td>
</tr>
<tr>
<td>False Positive - Resources</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - SQL Injection</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - Third Party</td>
<td>3</td>
</tr>
<tr>
<td>False Positive - UI</td>
<td>0</td>
</tr>
<tr>
<td>False Positive - XML Validation</td>
<td>0</td>
</tr>
<tr>
<td>Defect Created</td>
<td>0</td>
</tr>
</tbody>
</table>
Functional Testing
Functional Testing  Cont.

Scripts are automatically run to create Oracle and SQL Server unit test databases.

Tables and Indexes with bootstrap data are created along with procedures, views, sequences, etc.

Unit test startup, execute, and teardown is recorded in the build log.
Continuous Deployment/Delivery (CD)
Custom Automation

Intelligently transferring and deploying a solution to the desired environments
Infrastructure as Code (IaC)
IaC – Gold Disk Management

Windows 10 - **Template**

- Visual Studio 2017
- Eclipse Neon
- IntelliJ
- .NET
- Java
- Office
- NetBeans

Windows Server 2012 - **Blueprint**

- SQL
- Jenkins
- IIS
- Oracle
- JBoss
- TeamCity
- TeamCenter
- eQube

- Windows Server 2012 - Blueprint
- SQL
- Oracle
- Jenkins
- IIS
- JBoss
- TeamCity
- TeamCenter
- eQube

- Microsoft SQL Server
- Oracle Database
- SQL Server Management Studio
- Oracle Database Management Console

- Microsoft IIS
- Oracle Home
- JBoss Administration
- TeamCity

- VMware vRealize Automation
- VMware vRealize Orchestrator
- VMware vCenter Server

- API
- vRealize Orchestrator
- vCenter Server

- ELYSIUM
- Parker
- Northrop Grumman
- BOEING

BOEING is a trademark of Boeing Management Company. Copyright © 2018 Boeing. All rights reserved. Copyright © 2018 Northrop Grumman Corporation. All rights reserved.
Virtualization and Parallelization

Virtualization and Parallelization

HYPERVERSOR

1:1
Virtualization and Parallelization  Cont.

Before

5 hours in total (1 hour for each task)

After

1 hour in total

Automated regression test tend to take hours to complete, parallelization can decrease that time.
Metrics
Metrics Cont.
Metrics - Code

NCover drills down all the way to individual lines of code to show what was tested.
Questions???
References and Images

Global Product Data Interoperability Summit | 2018

- Continuous Delivery - http://electric-cloud.com/resources/continuous-delivery-101/continuous-deployment/
- Continuous Integration - https://visualstudio.microsoft.com/team-services/continuous-integration/
- DevoPedia - https://devopedia.org/devops
- IaC - https://docs.microsoft.com/en-us/azure/devops/what-is-infrastructure-as-code
- Metrics - https://www.monsterinsights.com/google-analytics-vanity-metrics-that-are-wasting-your-time/
- Questions - https://nosweatpublicspeaking.com/your-questions/my/
- TeamCity - https://www.slideshare.net/vomel/continuous-delivery-with-teamcity
- TDD - https://www.vectorcast.com/blog/2011/03/test-driven-development-vectorcast-part-1
- TDD2 - https://centricconsulting.com/case-studies/agile-test-driven-development/