INDUSTRY CASE STUDY: ENTERPRISE APPROACH TO DEVOPS

An Industry Executive View of Changes in Defense Software Acquisition

November 14, 2019
Esma Elmazaj, Senior Director, Corporate Engineering, L3Harris Technologies
Delivering Software Faster - An Enterprise Mission

“Certainly, as you deploy tools like **DevOps**, it's going to help you develop products which have significant **software content faster**. The concept is basically to be able to **continuously integrate and test** software builds so that you always have a software feature that you can field and market, and that is something that we have not developed. A lot of the defense companies have not, and that is going to compress the cycle time for software pretty substantially. We've seen that in multiple cases where we deployed **DevOps**. As we go out the next 2 to 3 years, by fiscal '21, we think **85%** or **90%** of our new starts will be on **DevOps**. I think it's going to be a key thing, the **compressing overall cycle time** and developing and launching new products.”

*William M. Brown, Chairman & CEO L3Harris Corporation*

*Earnings Call - July 31, 2018*

DevOps is driven from the highest executive levels at L3Harris
L3Harris Business Context

4 Mission Aligned segments

- Integrated Mission Systems: $4.9B
- Space & Airborne Systems: $4.0B
- Communication Systems: $3.8B
- Aviation Systems: $3.8B

L3Harris merger created opportunity to leverage best of breed from diverse software capabilities in a rapidly changing environment

* 2018 revenue
Challenges

Changing the culture
• Legacy to DevOps
• Automation (speed, speed, speed)
• Skill mix, support disciplines
• Industry and government, engagement

Enterprise metrics in DevOps era
• Adapting traditional measures to current needs
• Building new historical databases for estimates

Acquisition and contracting
• Streamlining policies, constraints
• Source code availability vs. IP

Integrating cybersecurity throughout the software process and toolchain

How did we get here?
Shift from Waterfall to Agile, from Silos to Collaboration

Figure 3. Theories of Software Development
Defense Science Board (DSB) Task Force on the Design and Acquisition of Software for Defense Systems

New practices driving changes to traditional A&D development
DevOps Program Transition Strategy

Enterprise Support

People
- Communities of practice
- Rewards and recognition

Process
- DevOps transition scorecards
- Baseline metrics
- Evaluate existing tools

Tools
- Provision software tools
- Assign DevOps support team
- Identify training needs

Training
- Develop DevOps pipeline
- Collect metrics, improvements
- Certify DevOps pipeline (QA)
- Monitor performance
- Collect lessons learned
- Evaluate Continuous Deployment opportunity

Continuous Integration

Assessment
- Identify candidate programs
- Baseline metrics
- Evaluate existing tools

Infrastructure
- Provision software tools
- Assign DevOps support team
- Identify training needs

Transition
- Develop DevOps pipeline
- Collect metrics, improvements
- Certify DevOps pipeline (QA)
- Monitor performance
- Collect lessons learned
- Evaluate Continuous Deployment opportunity

Continuous Deployment

Assessment
- Identify candidate programs
- Baseline metrics
- Identify training gaps

Artifact Management
- Componentize to multiple repos
- Triggers for new versions
- Versioning

Automated Deployment
- Install – dependencies
- Monitoring
- Provisioning
- Smoke testing

Automated Testing
- Test strategy (data, artifacts)
- Set automation goals
- Define feedback loop
Business Benefits of DevOps

Improved Collaboration
- Team ownership, accountability
- Integrated development and operations workflows

Higher Reliability
- Improved software quality
- Find defects earlier
- Predictable deployments

Return-on-Investment
- High value stream
- Minimal waste
- Cost savings

Increased Speed
- Higher velocity, throughput
- Shorter development cycles
- Adapt to change quickly

Resource Utilization
- Framework processes, tools, IT
- Stand up new projects quickly
- Automation frees staff for other work

Rapid Delivery
- Release frequency
- Innovate new products faster
- Responsive to customer needs

Software Factory Examples
- >50% Reduced build & release time
- >50% Decreased cycle time
- >90% Increased release frequency
- >50% Improved efficiencies

Program Benefits Examples
- 33% Reduced onboarding time
- 58% Reduced project defects
- 30% Decreased I&T effort
- 70% Reduced defect rework
DevOps Common Pipeline Framework

Continuous Integration / Delivery

- Certified & Secured Pipeline
- Rapidly stand up program environment
- Managed upgrades
- Support plug-ins for non-standard tool stack
- Savings through automation

Development

Operations

Commit

Build

Test

Stage

Deploy

Standardized Pipeline Template Libraries Provisioned with Program-Specific Plug-Ins

**Build**
- setuptools
- MSBuild
- Maven
- Make
- VxWorks

**Unit Test**
- pytest
- unittest
- VSTest
- JUnit
- googletest

**Static Code Analysis**
- SonarQube
- Klocwork
- Findbugs
- Spotbugs

**Security Test (Static)**
- Fortify

**Code Coverage**
- coverage
- pytest-Cov
- Cppcheck
- OpenCover
- Cobertura

**Artifact Management**
- Artifactory
- Nexus

**Configuration**
- Ansible
- Chef
- Puppet

**Security Test (Dynamic)**
- OWASP ZAP

**Functional Test**
- Selenium
- Test Execute
Achieving benefits from DevOps transition

Migration savings realized from the DevOps Common Pipeline Framework (DOCPF)

<table>
<thead>
<tr>
<th>Example Programs</th>
<th>Relative DevOps CI Implementation Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DOCPF</td>
</tr>
</tbody>
</table>

Cost Summary

-76%  -61%

DOCPF  DOCPF (tailored)  Without DOCPF*

Prioritize targets of opportunity

- Custom Hardware
- COTS Hardware
- Software as a Solution

- Highest Potential Value
- Lower Potential Value
- New
- Legacy
- O&M

Significant business impacts from strategic investment in DevOps pipeline frameworks and automation
Changing how we measure and improve software performance

Traditional enterprise measures (based on LOC, EV, defects)

- **Productivity**
  - SW productivity
  - Estimate accuracy
- **Quality**
  - Defect density
  - Rework

Enhanced enterprise measures (measurement framework categories)

- Throughput
- Predictability
- Quality
- Security

Program execution measures must support business objectives at the enterprise level

Aligned to incentivize rapid delivery of value for customer mission needs

Collaborating with industry and government partners to develop a consensus measurement framework, adapted to our business needs
Concept – Executive level oversight of DevOps performance

Predictability

Throughput

Enterprise vision – cohesive dashboards of software performance across all levels of the business generated on demand from integrated software tools
Executive Role in DevOps Transition

Success Factors

Establish vision, strategy and objectives
Visible commitment and engaged sponsorship
Investment in infrastructure
Enable software factories
Establish goals and measures (KPIs)

Implementation Strategy

DevOps advisory board
• Industry & functional collaboration
• Common metrics
• Governance

Enterprise DevOps Strategy
• Common pipelines
• Communities of practice
• Standard process & objectives

Program Execution
• Strengthen product quality
• Quickly meet customer demands
• Shift execution risk left

Successful execution requires a culture change and full buy-in at all levels
Contact Information

Esma Elmazaj
Senior Director, Corporate Engineering
L3Harris Technologies
Esma.Elmazaj@L3Harris.com
http://www.l3harris.com/