

Forge.mil: A Case Study for Utilizing Open Source Methodologies Inside of Government

Guy Martin and Aaron Lippold

Forge.mil Community Management Team
gmartin@collab.net, aaron.lippold@disa.mil
<http://www.forge.mil>

Abstract. In late 2008, DISA (Defense Information Systems Agency), the global IT arm of the US Department of Defense, embarked upon a project to create an internal collaboration and software application lifecycle management system. Beyond simply fielding yet another tool, the Forge.mil effort was designed to fundamentally change the way the DoD developed and acquired software technology and systems. The method of this change was the application of Open Source principles **inside** of the larger DoD community, including ideas such as meritocracy and code sharing, as well as Agile and collaborative software development. This lightning talk will explain the rationale behind Forge.mil, how it was developed using Open Source principles, and how it continues to influence technology acquisition within the DoD in both practice and policy changes.

1 Introduction

The US Department of Defense has had a long (and at times challenging) relationship with the Open Source world. While there have been successes along the way (including work on projects such as SE Linux), by and large, the use of Open Source within the DoD has been limited to certain server side applications.

There have been groups within the DoD community who recognized that there was value not only in working within the Open Source world where appropriate, but also in applying the same principles to internal development that made projects such as Linux, Apache, and Subversion successful.

This goes beyond simple ‘code sharing’ or ‘reuse repositories,’ which, while a laudable goal, are rarely successful without a large upfront effort at cultural upheaval. The DISA team (civilian, military, and contractors) that came together in 2008 to start Forge.mil realized early on that they were not going to change years of entrenched cultural resistance to ‘the Open Source way.’ What they hoped to provide, however, was not only a simple set of integrated tools to make life easier for all project stakeholders (developers, program managers, testers, senior executives), but a ‘safe place’ to begin the cultural change process necessary to make technology acquisition in the DoD much more streamlined.

Forge.mil is the third incarnation of this effort and arguably, the most successful, with 10,000 registered users and a new ‘Community layer’ built upon one of the most successful Open Source projects to date – the Drupal content management engine.

However, there are still entrenched cultural issues that need to be addressed to make the Forge.mil effort more successful and pervasive.

1.1 Vision

Forge.mil's official mission statement is: *“Improve DoD's ability to rapidly deliver dependable software, services and systems”*

Unofficially, the team works very hard at breaking down cultural barriers and ‘silos of excellence’ through the application of Open Source principles such as: meritocracy, transparency, reuse, trust, and community. While there has to be a base of operations, per se, with a unified tool set, a lot of the work done by the Forge.mil community management team revolves around coaching projects on collaboration techniques and software reuse strategies. Additionally, the team has shown leadership by helping lawmakers craft new guidelines on DoD acquisition strategy¹. These new guidelines direct the US Secretary of Defense to establish *“a new acquisition process for information technology systems, designed to include:*

- *Early & continual involvement of users*
- *Multiple, rapidly executed increments of capability*
- *Early, successive prototyping to support an evolutionary approach*
- *A modular, open-systems approach”*

Forge.mil helps provide a place to continue defining this improved style of ‘innersourcing’ for DoD technology acquisition.

1.2 Implementation Specifics

Forge.mil serves two specific types of audiences:

- Projects and developers looking to share and collaborate on internal code/projects
- Teams looking to collaborate with a limited set of government/contract workers

Because of these diverse audiences, and to make administration simpler, Forge.mil is primarily composed of two main capabilities:

- **Software.forge.mil**
 - Freely available for project hosting for ‘internal open’ projects
- **Project.forge.mil**
 - Fee-for-service offering allowing for limited participation

Additionally, there is a Content Management System (utilizing Drupal) called **Community.forge.mil** that provides ‘social development’ tools such as voting, reputation management, and project activity streams to provide visibility across multiple projects on each project site that may be part of a larger community of interest.

¹ US Congress: HR 2647, National Defense Authorization Act (2010), Section 804.

It is important to note again at this point that these systems are not freely available to the outside Open Source community – they are behind DoD firewalls and protected by Public Key Infrastructure systems. These systems are designed to replicate the Open Source ‘dynamic’ present in successful projects such as Linux, Apache, Drupal, etc.

The primary human interface between the community of users and the system is the Forge.mil Community Management team, composed of two half-time community managers (contractors), and one quarter-time government employee. This team’s roles include:

- Project onboarding and adjudication
- Coaching for Open Source and Agile best practices
- Detailed consulting/support to remove barriers to entry

One of the primary roles of this team is to help in determining what projects should be allowed on the ‘Internal Open Source’ system (Software.forge.mil). Projects that are forks of existing Open Source projects are usually rejected and their requesters sent to the proper external Open Source community, so that unmaintainable DoD-specific forks do not occur.

1.3 Challenges

Attempting culture change of this magnitude inside of an organization like the US Department of Defense is not without significant challenges. Among those are:

- Contractor resistance to ‘loss of intellectual property’
- Fear of sharing and/or showing substandard code/systems
- Perceived increases in cost to develop systems (easily countered – see Metrics/Outcomes below)
- Cultural resistance to change

There is no single ‘silver bullet’ to address all of these issues, but increasing economic pressures have provided a lever to help most teams ‘cross the chasm.’ Additionally, the government can invoke ‘Government Purpose Rights’ on software they have paid a contractor to develop, though there are sometimes additional costs for that.

1.4 Metrics/Outcomes

Forge.mil closely tracks several key metrics and ROI figures. In three years of operation, the program has had the following successes:

- 10,000 registered users
- 500 projects
- 57,000 software commits
- 51,000 software downloads
- 4,000 discussion posts
- 15,000 shared documents
- 1,000 software repositories created
- \$175M in ROI savings (cost avoidance and software asset reuse)

1.5 Conclusion

The implementation of Forge.mil provides a useful case study in how to apply Open Source methodologies to internal development problems within corporations or governments for maximum benefit. There are many challenges along the way, but a collaborative approach can be utilized to overcome most stakeholder concerns and issues.