Software is Never Done:
Refactoring the Acquisition Code for Competitive Advantage

Defense Innovation Board, 3 May 2019

Current state – the problem:

- **Software is ubiquitous and U.S. national security relies on software.** The ability to acquire and deploy software is central to national defense and integrating with allies.
- **The threats the U.S. faces change rapidly**, and DoD's ability to adapt and respond is now determined by its ability to develop and deploy software to the field.
- **The current approach to software development is a leading source of risk to DoD:** it takes too long, is too expensive, and exposes warfighters to unacceptable risk.
- **Software is not being used to enable a more effective force,** strengthen our ability to work with allies, and improve the business processes of the Department.
- **Nothing is changing:** most of this has been said before and the 1987 DSB report on military software pretty much says it all. What is it going to take to actually do something?

Moving forward – fundamental themes:

- **Speed and cycle time are the most important metrics for managing software.** DoD needs to deploy and update software that works for its users at the speed of (mission) need, and execute inside the OODA loop of our adversaries to maintain advantage.
- **Software is made by people and for people, so digital talent matters.** DoD's current personnel processes and culture will not allow its military and civilian software capabilities to grow nearly enough to meet its needs. New mechanisms are required.
- **Software is different than hardware (and not all software is the same).** Hardware can be developed, procured, and maintained. Software is an enduring and evolving capability that must be supported and continuously improved throughout its lifecycle.

Main lines of effort:

- **Congress and OSD: Refactor statutes, regulations, and processes for software**, providing increased insight to reduce the risk of slow, costly, and overgrown programs, and enabling rapid deployment and continuous improvement of software to the field.
- **OSD and the Services: Create and maintain cross-program/cross-Service digital infrastructure** that enables rapid deployment, scaling, testing, and optimization of software as an enduring capability; manage them using modern development methods; and eliminate the existing hardware-centric regulations and other barriers.
- **Services and OSD: Create new paths for digital talent (especially *internal* talent)** by establishing software development as a high-visibility, high-priority career track and increasing the level of understanding of modern software within the acquisition workforce.
- **DoD and industry: Change the practice of how software is procured and developed** by adopting modern software development approaches.

OSD should lead the efforts for the enterprise (OSD, Services, Congress, contractors, and vendors) to rapidly and forcefully implement the recommendations in this study (and the multitude of previous software acquisition studies that have said similar things).

The Ten Most Important Things to Do (Starting Now!)

Line of Effort A (Congress and OSD): Refactor statutes, regulations, and processes for software	
A1	Establish one or more new acquisition pathways for software that prioritize continuous integration and delivery of working software in a secure manner, with continuous oversight from automated analytics
A2	Create a new appropriation category for software capability delivery that allows (relevant types of) software to be funded as a single budget item, with no separation between RDT&E, production, and sustainment
Line of Effort B (OSD and Services): Create and maintain cross-program/cross-Service digital infrastructure	
B1	Establish and maintain digital infrastructure within each Service or Agency that enables rapid deployment of secure software to the field and incentivize its use by contractors
B2	Create, implement, support, and use fully automatable approaches to testing and evaluation (T&E), including security, that allow high-confidence distribution of software to the field on an iterative basis
B3	Create a mechanism for Authority to Operate (ATO) reciprocity within and between programs, Services, and other DoD agencies to enable sharing of software platforms, components and infrastructure and rapid integration of capabilities across (hardware) platforms, (weapons) systems, and Services
Line of Effort C (Services): Create new paths for digital talent (especially *internal* talent)	
C1	Create software development units in each Service consisting of military and civilian personnel who develop and deploy software to the field using DevSecOps practices
C2	Expand the use of (specialized) training programs for CIOs, SAEs, PEOs, and PMs that provide (hands-on) insight into modern software development (e.g., Agile, DevOps, DevSecOps) and the authorities available to enable rapid acquisition of software
Line of Effort D (DoD and industry): Change the practice of how software is procured and developed	
D1	Require access to source code, software frameworks, and development toolchains – with appropriate IP rights – for DoD-specific code, enabling full security testing and rebuilding of binaries from source
D2	Make security a first-order consideration for all software-intensive systems, recognizing that security-at-the-perimeter is not enough
D3	Shift from the use of rigid lists of requirements for software programs to a list of desired features and required interfaces/characteristics, to avoid requirements creep, overly ambitious requirements, and program delays

Chapter 5 of the SWAP report provides context and the additional 16 recommendations. Appendix A contains draft implementations.

All DIB Software Acquisition and Practices (SWAP) documents can be found here: https://innovation.defense.gov/software/