Continuous Iterative Development and Acquisition Working Group (CIDAWG)

Progress Report

NDIA Systems Engineering Division

06-Feb-2019

CIDAWG Overview

Purpose	Assist DoD in adoption of recommendations from 2018 DSB Report "Design and Acquisition of Software for Defense Systems"
Sponsor	Jeff Boleng, Special Assistant for Software Acquisition, ODASD (A&S)
Objectives	Provide input to DoD on the adoption of the DSB recommendations via a report detailing recommended guidance for programs to define, plan, develop, acquire, test, and support of software
Partnerships	INCOSE, PSM, DAU GAO, MDA, NRO, OSD, USAF, USA, USN
Membership	44 CIDAWG participants from industry, government, and academia.
Recent Activities	• Data gathering: first draft completed of briefing summarizing initial recommendations (Feb 2019) to ODASD (A&S) as input to Congressional briefing on DoD implementation of DSB recommendations as required by GFY19 NDAA. Under internal team peer review. See summary. • Held face-to-face CIDAWG meeting and engaged with INCOSE WGs at IW 2019. Seeking broad industry perspective and consensus on NDIA/INCOSE/PSM recommendations.

Draft Briefing Structure

DSB Recommendations:

1. **Software Factory** – A key evaluation criteria in the source selection process should be efficacy of the offeror's software factory.
2. **Continuous Iterative Development** – DoD and defense industrial base partners should adopt continuous iterative development best practices for software, including through sustainment.
3. **Risk Reduction and Metrics for New Programs** – For all new programs, starting immediately, implement best practices in formal program acquisition strategies (multiple vendors and down-selects, modernized cost and schedule measures, status estimation framework)
4. **Current and Legacy Programs in Development, Production, and Sustainment** – for ongoing development programs, PMs/PEOs should plan transition to a software factory and continuous iterative development.
5. **Workforce** – The U.S. Government does not have modern software development expertise in its program offices or the broader functional acquisition workforce. This requires Congressional engagement and significant investment immediately.
6. **Software is Immortal: Software Sustainment** – RFPs should specify the basic elements of the software framework supporting the software factory… reflected in source selection criteria
7. **IV&V for Machine Learning** – Machine learning is an increasingly important component of a broad range of defense systems, including autonomous systems, and will further complicate the challenges of software acquisition.

For responses to each of the 7 DSB recommendations:

- **Assumptions**
- **Picture of Success (End State)**
- **Current State**
- **Description**
- **Obstacles**
- **Path Forward**

Current working draft: 104 pages

Draft Recommended Actions - Summary Excerpts

Action Plan - #1: Software Factory
Establish common understanding: nomenclature, training
Contracts: establish contract language for enabling SW factory delivery.
Incentives: industry incentives for delivery of desired business outcomes
Standardization: Define standards at the data layer for software factory to facilitate data sharing Define nomenclature standards across vendors; use an existing framework such as Scaled Agile Framework (SAFe)
Software factory dojo: pilot on customer site, view system telemetry, validate O&M
Define delivery value stream: push vendor baselines thru factory.
Integrated performance measurement baseline: pilot transparency for deliveries
Automated metrics collection: Predictability, Quality, Performance
Relentless continuous improvement

Action Plan - #2: Continuous Iterative Development
Train key government and contractor personnel
Establish an initial approach to defining programs for CID implementation
Conduct pilot programs for CID, employing a set based design approach to explore options and refine approach
Iterate until a small set of effective approaches and techniques emerge and standardize on them
Develop an approach to integrate feedback into the standard process for continuous improvement

Action Plan - #3: Risk Reduction and Metrics for New Programs
Establish measurement framework: survey industry for best practices, consensus recommendations for typical measures; pilot measures; contract language
WBS-based estimating: expand approach for collecting historical DB measures across programs, expand to include DevOps measures. Partner with stakeholders on historical data estimating initiatives and development of appropriate size-based methods.

Draft Recommended Actions - Summary Excerpts

Action Plan - #4: Current and Legacy Programs in Development, Production, and Sustainment
Define patterns: for incremental transition of architecture and capabilities with minimal risk
Define standards: at data layer for SW factory to enable flexibility
Define nomenclature standards: across vendors, using existing framework such as SAFe
Conduct transition program pilots: capture and leverage lessons learned
Automate metrics collection: Predictability, Quality, Performance

Action Plan - #5:
Develop modern SW competency framework: create ability to identify and code SWEs in current and future SW skillsets. Create new SW Eng 0800 Occupational Series.
Workforce competency and training gap analysis: Identify workforce gaps; quantity/quality. Workforce shaping. Career path development.

Action Plan - #6: Software is Immortal – Software Sustainment
Contracting language: convene government/industry workshop to develop contract language supporting basic elements of SW framework supporting the SW factory. Develop policies and guidance. Conduct program pilots.
Develop mechanisms for sustainment organizational ecosystems: convene government/industry workshop for effective transfer of SW factories. Socialize effective transfer mechanisms. Establish policies, guidance. Conduct program pilots.

Draft Recommended Actions - Summary Excerpts

Action Plan - #7: IV&V for Machine Learning
Adopt a risk-based framework: for IV&V needs associated with machine learning in the system, using mitigation of associated risks.
Research and experimentation: to mitigate risks. Data quality techniques to validate real-world distributions. Run Time Assurance (RTA) approaches. Formal methods to prove correctness. Enhance trust in machine learning systems (see DARPA XAI).
Address ML risks/concerns within CONOPS and Architecture: risks dependent on impact of ML model on overall system behavior. Manage risks levels through CONOPS and system architecture decisions.
Ensure data availability and traceability across industry: DIB commandment: all data should be stored, mined, made available for ML. Establish secure data repository/exchange to foster sharing and innovation. Include requirements for maintaining history, provenance and pedigree of data sets and ML models, maintain data/model traceability.
SW factory considerations for ML systems: consider ML system needs so models can be rapidly re-trained, re-tested, re-deployed, etc. and included in evaluation criteria for SW factories. Provide abundant storage for training/validation data, ample compute power to support training runs.