Implementing Continuous Iterative Development and Acquisition

Executive Summary

NDIA Systems Engineering Division
in partnership with INCOSE and PSM

22-Apr-2019
Background

Defense Science Board (DSB) released a report in Feb-2018 containing seven recommendations regarding software design and acquisition. Section 868 of NDAA 2019 mandates implementation of these recommendations within 18 months.

The Defense Innovation Board (DIB) Software Acquisition and Practices (SWAP) study group has also provided many insightful and largely compatible recommendations.

NDIA, INCOSE and PSM support the DSB and DIB concepts and the opportunities they offer to DoD and the defense industry.

- NDIA offers the recommendations herein to ASD(A&S) and ASD(R&E) representing an “industry perspective” on path forward.
- NDIA appreciates the opportunity to partner with DoD on implementation.
DSB SW Task Force 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.

The NDIA working group developed consensus recommendations responding to each of the 7 DSB findings:

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

This briefing is an executive summary of those recommendations. Detailed report provided separately.
Continuous iterative development (CID) methods have cross-functional implications. The scope includes not just SOFTWARE but also SYSTEMS ENGINEERING and supporting disciplines.

Software Factories include people, processes, and tools – not just a tool chain.

Funding and contracts must be aligned to support implementation and/or migration to SW factories with life cycle sustainment.

A collaborative approach to Intellectual Property (IP) across the entire acquisition life cycle will be developed that meets both Government and Supplier needs.

A business case can be made for the effective deployment and maintenance of integrated tool chains to build capability throughout the life of the system.

Traditional waterfall-based processes, tools, and measures are generally not well suited to CID.

A skilled SW-informed workforce cadre is available or can be developed across functions (e.g., software, acquisition, PMs, sustainment).

Cross-cutting assumptions. Refer to the separate detailed report with assumptions specific to each DSB recommendation area.
# DSB #1: Software Factory

## NDIA WG Recommendations

### Picture of Success (end state)

**People**
- Qualified factory workforce
- Continuous learning (relentless improvement, pipeline feedback)

**Process**
- Integrated PMB
- Metrics
- Predictability
- Digital Blueprint / Play Book
- Ontology, Nomenclature
- Secure Supply Chain
- Relentless Improvement

**Tools**
- Platform agnostic tool chain
- Adaptable to change
- Test automation at all levels
- Model-based SW validation vs. architecture
- Red team / Blue team factory

### Recommendations for Path Forward:

#### DSB #1: Software Factory (1 of 3)

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contract for software factory software</td>
<td>Create a blueprint of contracts and language to enable software factory delivery. Define use cases for different types of software (e.g., embedded, firmware, web, etc.), enabling business, critical, limiting.</td>
</tr>
<tr>
<td>Root value streams instead of projects</td>
<td>Identify a root value stream for a single, long-term project. Align project value streams with root value streams. Hold workshops with industry to identify root value streams. Align project plans with root value streams. Evaluate actual root value stream performance.</td>
</tr>
<tr>
<td>Roadmap software factory interfaces to facilitate data sharing</td>
<td>Develop a roadmap of interfaces to enable data sharing. Define standards for data exchange between different software factory interfaces. Ensure common language and tools are in place across vendors. Use existing frameworks, such as the Suite Agile Framework (SAFe).</td>
</tr>
</tbody>
</table>

#### DSB #1: Software Factory (2 of 3)

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Publish blueprints and playbooks</td>
<td>Publish a blueprint of key processes and playbooks for software development and delivery.</td>
</tr>
<tr>
<td>Truly integrated PMB</td>
<td>Ensure PMB is truly integrated across the software factory, from design through delivery, and includes all aspects of the software development lifecycle.</td>
</tr>
<tr>
<td>Secure software factory</td>
<td>Ensure a security-by-design approach is integrated into the software factory. Identify clear roles and responsibilities for ensuring security.</td>
</tr>
<tr>
<td>Biometric-based supply chain</td>
<td>Ensure a biometric-based supply chain is in place for供应链 integrity. Use biometric data to ensure authenticity and integrity of all components.</td>
</tr>
<tr>
<td>Define value streams for delivery and pack vendor-oriented locations through factory</td>
<td>Define clear value streams for delivery and integrate multiple vendors' solutions into the software factory. Ensure traceability and reliability.</td>
</tr>
</tbody>
</table>

#### DSB #1: Software Factory (3 of 3)

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantum resistance and process for results</td>
<td>Develop quantum-resistant processes and frameworks for secure software delivery. Ensure processes are robust against quantum attacks.</td>
</tr>
<tr>
<td>End-to-end retrospective for software of systems</td>
<td>Perform end-to-end retrospective analysis of the software development process. Identify opportunities for improvement and enhance the software factory.</td>
</tr>
<tr>
<td>Security in the cloud (OaaS)</td>
<td>Ensure security is a core aspect of cloud services. Implement secure cloud services and architectures.</td>
</tr>
</tbody>
</table>

---

**PMB: Performance Measurement Baseline**

22-Apr-2019

NDIA Continuous Iterative Development and Sustainment WG
DSB #2: Continuous Iterative Development (MVP)
NDIA WG Recommendations

**Picture of Success (end state)**

### Government / Contractor Interface

**Contracting**
- New programs defined by solution intent (CV-1)
- Contracts defined by evolutionary viability products (MVP/NVP)

**Funding**
- Contract funding structure supports seamless capability evolution

**Stakeholders**
- Active engagement in CID lifecycle

**Design**
- Guided by MOSA

**IP**
- Government access to source code with negotiated IP protections

### Program Execution

**People**
- Multi-discipline agile execution includes aligned milestones
- Direct user/developer interaction informs design (product owner)

**Process**
- Early SE > SW sequencing, refactoring, tools, environments

**Tools**
- Test automation accelerates delivery (rapid release, deployment)

---

Procurements based on iterative development of releases according to product capability thresholds

**Recommendations for Path Forward:**

- Establish CID pilot baseline
- Establish an initial high level CID approach
- Define key Government and Supplier personnel
- Define a design set for CID
- Develop a CID strategy for CID, emphasizing a risk-based design approach to express options and risks, approach
- Select a small set of effective approaches to mitigate risks and uncertainties
- Implement and evolve
- Develop an approach to integrate feedback into the standard process for continuous improvement
- Define CID, requirement phasing and impact and other on the roadmap
- Align the CID in standard approach
- Manage feedback and evolution
DSB #3a: Risk Reduction (Competitive Prototyping)
NDIA WG Recommendations

Picture of Success (end state)

| Competition       | • Business case: win-win partnership, common goals, acquisition/support strategy  
|                   | • Objective downselect evaluation criteria (RFP L&M) and feedback  
|                   | • Open architecture on critical components |
| Contracts         | • IP agreement negotiated, sustained across the life cycle  
|                   | • Funding and contracts aligned to support factory migration |
| Metrics           | • Continuous improvement, SMART measures against objectives  
|                   | • Risk-based decision making |
| Resources         | • Funding, staffing, tools, environments to support multiple teams |

Recommendations for Path Forward:

Competitive prototyping can help in many situations, but does not solve all acquisition problems.

IP: Intellectual Property  
RFP: Request for Proposal  
SMART: Specific, Measurable, Achievable, Relevant, Time bound
DSB #3b: Measures for CID
NDIA WG Recommendations

<table>
<thead>
<tr>
<th>Consensus frameworks</th>
<th>• Objectives first - measures aligned and tailored from information needs, goals and constraints, at program and enterprise levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modernized measures</td>
<td>• Migration toward consensus alternatives to traditional waterfall and phase-based SW measures (LOC, EVM, milestones, ...)</td>
</tr>
<tr>
<td></td>
<td>• Derived from SW factory processes, automated by toolchain</td>
</tr>
<tr>
<td></td>
<td>• Basis for measuring cost and schedule vs. plan</td>
</tr>
<tr>
<td>History-based estimates</td>
<td>• Repositories collect performance-based measures (e.g., WBS, staff, cost, productivity) supporting future comparisons, basis of estimates, proposals, and program monitoring</td>
</tr>
</tbody>
</table>

Measures for CID should be aligned with information needs and constraints, at program and enterprise levels.

Recommendations for Path Forward:

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action/Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software measurement framework for CID</td>
<td>• Re-base current framework ([objects, categories, measures] with [Measurement and industry guidelines e.g., ITA, INCOSE, PSM, 1990] )</td>
</tr>
<tr>
<td></td>
<td>• Further evolve metrics for software CID</td>
</tr>
<tr>
<td></td>
<td>• Pull and validate measurements on integrated CID/CMTEC programs.</td>
</tr>
<tr>
<td></td>
<td>• Develop contracting language enabling measurement set for future programs</td>
</tr>
<tr>
<td>WBS-based benchmarking of historical competitiveness for staff, cost, productivity</td>
<td>• Recommend DF rep standard based approach and standard of key measures to additional programs (e.g., program level and not just DF to continuous software baselines required for future programs)</td>
</tr>
<tr>
<td></td>
<td>• Partner with companies to estimate the cost (E to E and to mitigate cost of changes to the system)</td>
</tr>
<tr>
<td></td>
<td>• Establish partnerships with industry for new methods (e.g., DF-8)</td>
</tr>
</tbody>
</table>

CID: Continuous Iterative Development
EVM: Earned Value Management System
LOC: Lines of Code
WBS: Work Breakdown Structure
DSB #4: Transition for Current and Legacy Programs
NDIA WG Recommendations

<table>
<thead>
<tr>
<th>People</th>
<th>Process</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Skill assessment for gap analysis</td>
<td>• Business case for transition</td>
<td>• Tools to generate legacy ‘as-built’ documentation and models for legacy code base</td>
</tr>
<tr>
<td>• Skilled capable workforce for transition on legacy programs</td>
<td>• Playbooks and Blue Prints for legacy code transition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Assessment of supply chain and SW pedigree (FOSS, COTS, GOTS)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Risk adjusted product backlog</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Strategies for incrementally building up test automation</td>
<td></td>
</tr>
</tbody>
</table>

**Picture of Success (end state)**

**People**
- Skill assessment for gap analysis
- Skilled capable workforce for transition on legacy programs

**Process**
- Business case for transition
- Playbooks and Blue Prints for legacy code transition
- Assessment of supply chain and SW pedigree (FOSS, COTS, GOTS)
- Risk adjusted product backlog
- Strategies for incrementally building up test automation

**Tools**
- Tools to generate legacy ‘as-built’ documentation and models for legacy code base

See also: Defense Innovation Board SWAP Study Report: Supplementary Documents, Appendix B.6 Sustainment / Modernization Subgroup Report

Recommendations for Path Forward:

Defense Science Board, Design and Acquisition of Software for Defense Systems, Feb 2018

22-Apr-2019 NDIA Continuous Iterative Development and Sustainment WG

COTS: Commercial Off the Shelf
FOSS: Free Open Source Software
GOTS: Government Off the Shelf
DSB #5: Workforce
NDIA WG Recommendations

Picture of Success (end state)

**Education and Training**
- DAU curriculum for DevSecOps and modern SW-centric systems
- Community of practice platforms
- Training across career fields (PM, sustainment, acquisition)
- Aligned with current/future development and recruiting needs

**Stakeholder Engagement**
- Collaborative government / industry partnerships
- Trained experienced industry partners and supply chain
- Consensus measurement framework
- Multi-discipline CID support teams (CDRLs, events, milestones)

**Staffing**
- Increased hiring, retention, training for acquisition experts
- Recruiting pipeline for SW experts
- Dedicated workforce funding and coaches across services
- PMO IPTs for modern SW practices

Recommendations for Path Forward:

- Continuous Iterative Software Development, Acquisition and Sustainment
- DevSecOps
- Competency Models
- Practical Experience (Boot Camp)
DSB #6: Sustainment (Software Is Immortal)
NDIA WG Recommendations

Picture of Success (end state)

<table>
<thead>
<tr>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability and support of a trained proficient workforce</td>
</tr>
<tr>
<td>Organic DoD software infrastructure, incentives, funding</td>
</tr>
<tr>
<td>Collaborative IP strategy throughout the life cycle, using a “work shared sustainment” approach</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Contracting Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contracts specify elements of framework supporting SW factory</td>
</tr>
<tr>
<td>Policies and guidance validated by workshops, pilots</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sustainment Ecosystems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understanding of current and future organizational ecosystems to ensure effective transfer of SW factories</td>
</tr>
</tbody>
</table>

Recommendations for Path Forward:

- Develop contracting language that contains the basic elements of the software framework supporting the software factory
  - NDIA workshop with government and supplier personnel
  - Generation and standardization of proposed contracting language
  - Conduct a set of pilot programs
  - Develop policies and guidance

- Develop an understanding of the current and future sustainment organizational customs to ensure the effective transfer of the software factories
  - NDIA workshop with government and contractor personnel
  - Generation and standardization of effective transfer mechanisms
  - Conduct a set of pilot programs
  - Develop policies and guidance

Click thumbnails to zoom
### Picture of Success (end state)

**Consensus**

- Model-based inference engine considering full system context
- Risk-based methodology supporting T&E needs, linked to ML model failures early in system development process
- Mitigation throughout system design, development, sustainment

**Open Data Sets**

- High data quality, quantity, availability, and traceability
- Data repository accessible to government and industry
- Governance model for availability, level playing field, innovation
- New repository data continuously collected and published

**Perpetual Updates**

- Continuous ML model updates – evolution at speed of relevance
- Continuous V&V methods sensing changes from models, environment
- Performance/accuracy aligned with changing environment, threats

---

### Recommendations for Path Forward:

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML IV&amp;V Framework</td>
<td>Develop a risk management framework for ML risk in the same way that cyber risk is managed. For all NDIA needs associated with ML in the system, use the mitigation of associated risks as a component of the test and evaluation process.</td>
</tr>
<tr>
<td>Research and Experimentation Programs</td>
<td>Provide guidance on approaches such as: Data assimilation techniques to assess if training data sufficiently represent real-world data. Risk-Based Assurance (RBA) approaches. Manual methods and other approaches to prove correctness of ML models. Enhancing trust in ML systems (see NIST/Aleatory v. Stochastic: An Introduction to Aleatoric and Epistemic Uncertainty)</td>
</tr>
<tr>
<td>Address ML risks/uncertainties with CNDPM and architecture</td>
<td>Standardize approaches to evaluating risk in the system, and develop practice of CNDPM, functional frameworks, and design patterns to mitigate three types of risk. The risks associated with ML in a system depend on how that ML model impacts overall system behavior. We can manage risk through CNDPM and system architecture decisions.</td>
</tr>
</tbody>
</table>

---

### T&E is a full lifecycle activity focused on mitigating risk of failing to meet operational needs

---

### Perpetual Upgrades

---

**IV&V**: Independent Verification & Validation  
**ML**: Machine Learning  
**T&E**: Test and Evaluation
Summary

The NDIA WG provides an industry perspective on picture of success, current state, obstacles and path forward for each DSB recommendation

<table>
<thead>
<tr>
<th>DSB Recommendation</th>
<th>NDIA “Path Forward” recommendations</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1 – Software Factory</td>
<td>14  Contracting, funding, incentives, methods, security, supply chain, and measures</td>
</tr>
<tr>
<td>#2 – Continuous Iterative Development</td>
<td>3  Pilots and continuous improvement</td>
</tr>
<tr>
<td>#3 – Risk Reduction &amp; Metrics</td>
<td>10 Acquisition strategy, competitive prototyping, culture, workforce, IP, and measures</td>
</tr>
<tr>
<td>#4 – Legacy Systems</td>
<td>5  Assessments, supply chain, methods, tools, and modeling</td>
</tr>
<tr>
<td>#5 – Workforce Development</td>
<td>3  Competency models, workforce assessment, workforce management, and training</td>
</tr>
<tr>
<td>#6 – Sustainment</td>
<td>2  Contracting and industry-government transfer of sustainment responsibilities</td>
</tr>
<tr>
<td>#7 – Machine Learning</td>
<td>5  Risk, research, CONOPs, ML data, and Software Factory interactions</td>
</tr>
</tbody>
</table>

Details of each topic and recommendation are provided in the separate report.
The NDIA Systems Engineering Division and its partners, INCOSE and PSM, appreciate the opportunity to provide an industry perspective for advancing the use of iterative methods in defense software acquisition.

The defense industrial base embraces the opportunities offered by the DSB and DIB recommendations and looks forward to supporting the Department of Defense with implementation.

NDIA Continuous Iterative Development and Sustainment Working Group:

<table>
<thead>
<tr>
<th>Acknowledgments</th>
</tr>
</thead>
<tbody>
<tr>
<td>The NDIA Systems Engineering Division and its partners, INCOSE and PSM, appreciate the opportunity to provide an industry perspective for advancing the use of iterative methods in defense software acquisition. The defense industrial base embraces the opportunities offered by the DSB and DIB recommendations and looks forward to supporting the Department of Defense with implementation.</td>
</tr>
</tbody>
</table>
Backup

Supporting Content (Hidden Slides)

Excerpts of NDIA Recommendations by DSB Finding

(see separate briefing package for full details)
DSB #1: Securing the Factory

With security integrated

Product Backlog

Release Backlog

Automated Security tests
Run daily

Definition Of Done (sprint)
Security checks

Security Stories

Business Drivers

Enablers

RMF

Fuzzing

Static Analysis

Pen Test

Dynamic Analysis

Definition Of Done (Release)
Security checks

Plan Design Build Test

Definition Of Done (sprint)
Security checks

Security Story

Threat Modeling

Secure Design Principles

Defensive Coding

Self Service Security scanning

Continuous Monitoring

Shippable Product

22-Apr-2019 NDIA Continuous Iterative Development and Sustainment WG
## Initiative Action Plan

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
</table>
| **Contract for software factory delivery** | • Create a blueprint of contracts and language to enable software factory delivery  
• Define approaches for different types of software (e.g., embedded, firmware, web); (life-critical, business-critical, low risk) |
| **Fund value streams instead of projects** | • Pilot funding a value stream for a single vendor award program  
• Pilot funding value streams on multi-vendor award program |
| **Incentivize Suppliers to build interoperable software factories that are continuously exercised** | • Hold workshop with Industry to identify incentives  
• Pilot options on some small short term modular contracts |
| **Standardize software factory interfaces to facilitate data sharing** | • Common data architecture  
• Define standards at the data layer for software factory to enable flexibility  
• Define common nomenclature standards across vendors; use an existing framework such as the Scaled Agile Framework (SAFe) |
<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Publish blueprints and playbooks</td>
<td>• Collaborate with Industry to obtain software factory blueprints and playbooks and publish for use across programs to increase success</td>
</tr>
<tr>
<td>Transparent integrated PMB</td>
<td>• Publish blueprint of Integrated PMB (may differ across domains)</td>
</tr>
<tr>
<td></td>
<td>• Educate Government PMs on how to review PMB</td>
</tr>
<tr>
<td>Securing software factory</td>
<td>• Define a defense-in-depth approach to secure factory</td>
</tr>
<tr>
<td></td>
<td>• Identify a required cadence of Red Team / Blue team to ensure factory safe.</td>
</tr>
<tr>
<td>Standards-based supply chain</td>
<td>• Define supply chain standards</td>
</tr>
<tr>
<td></td>
<td>• Define interoperability for supply chain with multiple factories</td>
</tr>
<tr>
<td>Define value stream for delivery and push varied vendor baselines through factory</td>
<td>• Define value stream for delivery and enable multiple vendor baselines to deliver into the factory.</td>
</tr>
<tr>
<td></td>
<td>• Ensure interoperability</td>
</tr>
</tbody>
</table>
**Initiative** | **Action Plan**
--- | ---
**Measure practices and process for results**  
- Document program practices and processes being used  
- Measure success of programs by practice and environment to analyze which practices are demonstrating the best results based on customer criteria of value. (not methodology, but individual practice)

**DoD-run retrospectives for a sampling of programs**  
- Select a sampling of programs once a quarter and run a retrospective jointly between Industry and Government to identify root causes and improvements  
- Publish best practices identified in retrospectives for all vendors

**Open source**  
- Research approach to instantiate Government-based open-sourced ways of working to leverage common modules across vendors and programs

**Teams as a service (CID Cells)**  
- Research approach to leverage cross-functional teams as a service in work areas where there is higher availability of workforce.

**IATO for infrastructure**  
- Research opportunity to obtain IATO on Infrastructure of software Factory.  
  - bare metal / cloud / database (DB) are the longest lead-time items to approve  
  - If we could secure a common architecture, the application layer would be cheaper and faster to approve, reducing cycle time for capabilities
DSB #2: Continuous Iterative Development
Picture of Success (End State)

**Government/Contractor Interface:**

- New Programs Defined by Solution Intent
- Contracts Defined by Minimal Viable Product (MVP)
- Funding Supports Capability Evolution
- Stakeholders Actively Engaged in Continuous Iterative Development Lifecycle
- Design Guided by MOSA
- Government Access to Source Code with IP Protections

**Program Execution:**

- Multi-discipline agile execution includes milestones
- Direct user interaction informs design
- Test automation accelerates delivery
## DSB #2: Continuous Iterative Development
### NDIA WG Recommendations

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
</table>
| Establish CID pilot baseline      | • Establish & communicate an initial high level CID approach  
• Establish an initial approach to defining programs for CID implementation  
• Train key Government and Supplier personnel                                                                                       |
| Pilot, learn and refine           | • Define a design set for CID  
• 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 it                                                                 |
| Implement and evolve              | • Develop an approach to integrate feedback into the standard process for continuous improvement  
• Define CID requirement phasing and Inspect and Adapt workshop timing  
• Roll out CID as standard approach  
• Manage feedback and evolution                                                                                                        |
DSB Recommendation #3 – Risk Reduction

Competitive Prototyping Survey, 2008 (USC CSSE)

Study indicates that CP can help in many situations, but has a number of pitfalls. CP does not solve all acquisition problems.
### Initiative Action Plan

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Acquisition strategy</strong></td>
<td>Acquisition strategies that provide a fair opportunity to compete, retain competition throughout the lifecycle for critical components to enable rapid evolution of the product.</td>
</tr>
</tbody>
</table>
| **Competitive prototyping** | • Review analyses/reports from prior DoD competitive prototyping initiatives, and integrate lessons learned into action plan for DSB recommendations.  
  • Competitive prototyping risk reduction strategy should account for both functional and non-functional requirements. |
| **Cultural shift**       | Migrate from subjective qualitative assessment to objective quantitative assessment of risk that support business decisions                             |
| **Resources**            | DoD investment to acquire, deploy, integrate, and maintain evaluation tools and test beds                                                   |
| **Workforce development**| Recommend DoD initiate a development plan to provide workforce with skills and knowledge needed to plan, perform and execute the risk reduction strategies during competitive prototyping. |
### Initiative: Program measurements

- Define a minimum core set of metrics and ownership for measures needed to do the job at the Program, Functional, and Integrated Product Team (IPT) levels
- Develop and track metrics to control factory processes, measure against goals and objectives, assess/measure risk, and make decisions
- Enable real-time insight into measures and program status
- Ensure measures provide a comprehensive view of risk reduction strategy, including: functional and non-functional requirements; reliability, security, ...
- Develop consensus Government/Industry measurement framework and common measures applied across defense software acquisition programs.

### Initiative: IP strategy

- Develop contracting approaches that protect Supplier IP while providing the Government access to source code for analysis, deployment, support, and evolution.
- Sustain IP required for maintenance of the following:
  - Renewable capital – patents, license, IP, ...
  - Human capital – People, skills, experience, surge/slack...
  - Structural capital – data bases, tools, processes, test scripts, ...
  - Relationship capital – customers, supplier agreements, business relationships, personal relationships, ...
DSB Recommendation #3 - Metrics
Frameworks for aligning measures with objectives

Summary of DIB Metrics Categories

**Deployment Rate**
- Initial launch to deployment of simplest useful functionality (MVP)
- Time to field high priority fn (spec>ops) or security hole (find>ops)
- Time from code committed to code in use

**Response Rate**
- Time req’d for full regression test (automated) and cyber testing
- Time required to restore service after outage (MTTD, MTTR, MTTA)

**Code Quality**
- Automated test coverage of specs/code
- Number of bugs caught in testing vs. field use (defect detection %)
- Change failure rate (rollback)

**Program Management**
- Complexity metrics. Devel plan/env metrics (specs, code, staff, ...)

Measures, goals, and priorities are tailored based on program objectives and information needs

The NDIA WG recommends a measurement framework that can be adapted to specifics of the program, domain, or acquisition

Measures for continuous iterative development should be aligned with information needs, objectives and constraints, at program and enterprise levels

Industry feedback (usefulness, effectiveness)
Available measures instrumented and automated by the toolchain
Selection of program measures tailored by information needs (with a few primary colors required by the enterprise)
Enterprise measures driven by business performance objectives

Success is measured at multiple levels:
- Mission capability
- Program execution
- Enterprise improvement
- Business results, competitiveness

NDIA Continuous Iterative Development and Sustainment WG
### DSB #3b: Metrics

**Path Forward**

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Software measurement framework for CID</strong></td>
<td>• Validate measurement framework (objectives, categories, measures) with Government and industry stakeholders (e.g., NDIA, INCOSE, PSM, SERC)</td>
</tr>
<tr>
<td></td>
<td>• Finalize initial consensus measures for software CID</td>
</tr>
<tr>
<td></td>
<td>• Pilot and validate measures/analysis on selected CID /DevSecOps programs.</td>
</tr>
<tr>
<td></td>
<td>• Develop contracting language requiring measurement set for future programs</td>
</tr>
<tr>
<td><strong>WBS-based estimating of historical comparables for staff, cost, productivity</strong></td>
<td>• Recommend DoD expand WBS-based approach and historical DB measures to additional programs but at program level and not specific to continuous software initiatives (doubtful consistent data yet exists).</td>
</tr>
<tr>
<td></td>
<td>• Engage Government stakeholders on historical data estimating initiatives</td>
</tr>
<tr>
<td></td>
<td>• Partner with independent cost estimate (ICE) groups to migrate away from SLOC-based methods (CAPE, PARCA, ICE, ...); establish partnerships with industry for new methods (DSB #3)</td>
</tr>
<tr>
<td><strong>Reach consensus on cost and schedule measures vs. plan for software CID</strong></td>
<td>• Consider alternatives to EVM for managing performance vs. plan.</td>
</tr>
<tr>
<td></td>
<td>• Review EVM agile studies, publications, and guidance. Hold workshops with Industry and Government to define framework and measures.</td>
</tr>
<tr>
<td></td>
<td>• Recommend consensus approach for DoD software acquisition</td>
</tr>
</tbody>
</table>
### DSB #4: Current and Legacy Programs
#### NDIA WG Recommendations

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
</table>
| Program assessment for categories of legacy software programs. | • Collaborate with industry building program categorization table for varied types of software and products being built  
  • Define common list of program readiness attributes  
  • Define metrics for how to measure transition success  
  • Develop common risk categories to evaluate  
  • Prototype process for iteratively and incrementally transitioning programs |
| Supply chain pedigree evaluation tool           | • Investigate methods for evaluating software pedigree  
  • Prototype process and tools to evaluate supply chain pedigree  
  • Validate pedigree on FOSS/COTS/GOTS/Supplier components |
| Blueprints and playbooks for low risk transition | Collaborate with Industry to build repository of blueprints , playbooks, and strategies for different types of programs. |
| Visualization tools for varied code bases.      | Investigate Visualization tools for different types of code bases |
| Auto generate “As-Built” and Models to evaluate system and develop transition plans | • Investigate standardized set of tools to auto-generate models and “As-Built” of the varied legacy systems  
  • Define a prioritization strategy for migrating program components to the software factory |
<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modern software-intensive-systems engineering competency model development</td>
<td>• DAU/INCOSE/NDIA/ISO collaboration to add software-centric systems engineering roles and proficiencies to INCOSE SE competency model and identify / develop workforce development content to improve proficiency</td>
</tr>
<tr>
<td></td>
<td>• Create ability to ID/code software-intensive-systems engineering in current/future software-centric systems skillsets</td>
</tr>
<tr>
<td>Informed PMs and software SMEs Training</td>
<td>• Development and deploy training at Defense Acquisition University on iterative software development for all acquisition communities (PM, Systems Engineering, Software, Financial Management, Cost Estimating, ...)</td>
</tr>
<tr>
<td></td>
<td>• Develop a consensus government/industry measurement framework and common measures applied across defense software acquisition programs</td>
</tr>
<tr>
<td></td>
<td>• Supply chain integration - Deploy supply chain pedigree evaluation tools and techniques</td>
</tr>
<tr>
<td></td>
<td>• Develop blueprints and playbooks for low risk transition</td>
</tr>
<tr>
<td></td>
<td>• Develop RFP guide for acquiring and transitioning to software factories</td>
</tr>
<tr>
<td>Initiative</td>
<td>Action Plan</td>
</tr>
<tr>
<td>-------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Workforce management    | • Baseline current software intensive capabilities and needs  
                           • Identify workforce gaps; quantity/quality  
                           • Update workforce needs to shape workforce recruitment and training  
                           • Create a new software-centric-systems Engineering 0800 Occupational Series to enable tracking, management and growth of software-centric-systems engineers, managers, and functional personnel  
                           • Fund software intensive develop training  
                           • Support continuous learning |
## DSB #6: Software Sustainment
### NDIA WG Recommendations

<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
</table>
| Develop contracting language that contains the basic elements of the software framework supporting the software factory | • NDIA workshop with government and Supplier personnel  
• Generation and socialization of proposed contracting language  
• Conduct a set of pilot programs  
• Develop policies and guidance |
| Develop an understanding of the current and future sustainment organizational ecosystems to ensure the effective transfer of the software factories. | • NDIA workshop with government and contractor personnel  
• Generation and socialization of effective transfer mechanism  
• Conduct a set of pilot programs  
• Develop policies and guidance |
<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
</table>
| **Adopt a risk-based framework** | Deploy a risk-based framework for managing ML risk in the same way that cyber risk is managed  
- For the IV&V needs associated with ML in the system, use the mitigation of associated risks as a core part of the test and evaluation process |
| **Research and experimentation programs should place a primary focus on approaches to mitigate risks** | Pilot R&D programs focused on approaches such as:  
- Data quality techniques to assess if training data sufficiently represent real-world distributions  
- Run Time Assurance (RTA) approaches  
- Formal methods and other approaches to prove correctness of ML models  
- Enhancing trust in ML systems (see DARPA Explainable AI (XAI)) |
| **Address ML risks/concerns within CONOPS and architecture** | Standardize approaches to evaluating ML risk in the system, and develop playbook of, CONOPS, architectural frameworks, and design patterns to mitigate these types of risk  
- The risks associated with ML in a system depends on how that ML model impacts overall system behavior  
- We can manage risk levels through CONOPS and system architecture decisions |
<table>
<thead>
<tr>
<th>Initiative</th>
<th>Action Plan</th>
</tr>
</thead>
</table>
| Ensure data availability and traceability across industry | Establish a data exchange that is not just a simple repository/dumping ground for data... Instead espousing a governance model and necessary security controls  
  • DIB: “All data generated by DoD systems - in development and deployment - should be stored, mined, and made available for machine learning (ML)”  
  • To allow for greater innovation, make all this data available to industry via a secure data repository/exchange  
  • Include requirements for maintaining history, provenance and pedigree of data sets and ML models, and maintain data/model traceability  
  • Continuous V&V methods tied to sensing of changes from models & environment |
| Software factory considerations for ML systems       | Ensure that evaluation criteria for a “Software Factory” considers the special needs of ML systems:  
  • Evaluation criteria for Software Factories must consider the special needs of development and deployment for ML (models need to be rapidly re-trained, re-tested, re-deployed) Software factory considerations include: abundant storage for training/validation data, ample compute (e.g., Graphics Processing Units (GPUs), Tensor Processing Units (TPUs)) to support training runs, etc. |