Addressing the Barriers to Agile Development in DoD

Su Chang
sjchang@mitre.org

Pete Modigliani
pmodigliani@mitre.org

May 2015

MITRE Defense Agile Acquisition Guide

Naval Postgraduate School’s 2015 Defense Acquisition Research Symposium

The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed by the author.
**Addressing the Barriers to Agile Development in DoD**

*Presented at the 12th Annual Acquisition Research Symposium held May 13-14, 2015 in Monterey, CA.*

**ABSTRACT**

Presented at the 12th Annual Acquisition Research Symposium held May 13-14, 2015 in Monterey, CA.
How DoD Acquisition professionals can apply Agile concepts within the unique and complex Defense Acquisition Environment

- DoD IT Acquisition Challenges
- Agile Overview
- Program Structure
- Requirements
- Contracting
DoD IT Acquisition Challenges

- Change in IT technology and operations is outpacing DoD IT acquisition development
- IT programs are subject to extensive documentation, reviews, and oversight that inhibits speed and agility needed for IT
- Major DoD systems average 38% cost, 27 month schedule overrun with >$1B/year spent on programs that are cancelled*
- Congress is demanding DoD to reform IT acquisition
  - Early and continual user involvement
  - Multiple, rapidly executed capability releases
  - Early, successful prototypes; evolutionary approach
  - Modular open systems approach

* Assessments of Selected Weapon Programs, GAO-14-340SP: Published: Mar 31, 2014

© 2015 The MITRE Corporation. *Approved for Public Release; Distribution Unlimited. Case Number 15-1457*
Agile Acquisition

Small, dynamic, collaborative Gov’t/Industry teams focused on:

- **Small, Frequent Releases**
- **Iteratively Developed**
- **Responsive to Changes**
- **Operations, Technology, Budgets**

- **Review Working Software**
- **Vice Extensive Docs**
- **Active User Involvement**
- **To Ensure High Ops Value**
DoD Barriers to Agile Acquisition

- Heavily regulated environment of acquisition policies and laws
- Bureaucratic, laborious, and slow processes
- Command-and-control governance structure and authorities

Agile Runs Counter to DoD’s Acquisition Environment

- Iterative releases vs big bang waterfall
- Working software vs extensive docs
- Respond to changes vs upfront plans of budgets, requirements, designs
Programs Should Consider Agile When…

- Requirements can be decomposed into small tasks
- Ops environment supports small, frequent capability deliveries
- Users can engage in development on CONOPS and feedback
- Programs can use existing infrastructure, focus on applications
- Industry has relevant domain expertise in Agile practices
- Decision authority supports Agile and tailored processes
Structuring an Agile Program

Time Boxed Release

- **Notional: 6 Month Release with 4-Week Sprints**
  - Continual development, integration, and testing
  - Monthly demonstration of capabilities to users
- **Gov’t testers, certifiers, and users involved early and often**
  - Minimizes work and surprises at the end of the release

Release Length Based on Program, Ops, and Technical Risk
Potential Agile Structure

1. **Materiel Development Decision**
   - Analyze trades of cost, performance risks and schedule.

2. **Materiel Solution Analysis Phase**
   - Requirements Definition
   - Develop Backlog

3. **Technology Maturation and Risk Reduction Phase**
   - Develop Technical Baseline
   - Competitive Prototypes

4. **Initial Operational Capability**
   - IOC

5. **Full Operational Capability**
   - FOC

6. **Engineering and Manufacturing Development Phase**
   - Develop, build, and test system to verify all requirements are met and ready for production or deployment.

### Release Phases
- **Release 1**
  - Deployment Decision
  - Plan
  - Release 1

- **Release 2**
  - Deployment Decision
  - Plan
  - Release 2

- **Release n**
  - Deployment Decision
  - Plan
  - Release n

### Operations and Sustainment
- Manage Contract(s)
Agile Requirements Backlog

- An evolving, prioritized queue of requirements
- Integrates operational and technical requirements
- Actively managed with user inputs and reviews
- Development team commits to scope of work for a sprint
- Sprint scope is locked, while release scope may change
- Sprint demos may identify new features or defects which would be added to the release or program backlogs
JCIDS IT Box Model

- Streamlined requirements process for software $>15M
- JROC approves IS-ICD – delegates approvals of follow-on docs
  - Follow-on docs tailored scope and content

ICD: Initial Capabilities Document
RDP: Requirements Definition Package
CD: Capability Drop
Consider a PEO, portfolio, or enterprise-level contract vehicle
- Streamlined contracting processes result in faster awards, deliveries
- Standardized, effective, and efficient contract management
# Contracting for Agile – Service vs Product

<table>
<thead>
<tr>
<th>Services (FAR Part 37)</th>
<th>Product-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pay for the time and expertise of an Agile development contractor</td>
<td>Contract for a defined software delivery product</td>
</tr>
<tr>
<td>• Fixed priced</td>
<td>• Firm Fixed Price</td>
</tr>
<tr>
<td>• Cost-reimbursement term</td>
<td>• Cost-reimbursement completion</td>
</tr>
<tr>
<td>• T&amp;M</td>
<td></td>
</tr>
<tr>
<td>• Contractor is selected based on the strength of the development team</td>
<td>• Contractor selected on technical solution</td>
</tr>
<tr>
<td>• Enables a teaming environment between the Government and contractor</td>
<td>• Requires upfront requirements definition for contractor cost estimates</td>
</tr>
<tr>
<td>• Appropriate when the Government wants to drive the development strategy</td>
<td>• Difficult to hold contractor accountable for delivery by directing Agile methods</td>
</tr>
<tr>
<td>• Responsive to requirements changes</td>
<td>• Requirements changes requires contract negotiation, ECPs, and/or mods</td>
</tr>
<tr>
<td>• Close collaboration required to ensure an integrated solution is delivered</td>
<td>• Diminishes flexibility and negotiation power of the Government</td>
</tr>
</tbody>
</table>

| Best option for Agile                                                                | Very difficult for Agile                                       |
## Services Contract Type

<table>
<thead>
<tr>
<th>Contract Type</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFP Services</td>
<td>• Generally preferred contract type in DoD</td>
<td>• Requires deliverables for payment (e.g., monthly report) unless progress payments are authorized</td>
</tr>
<tr>
<td></td>
<td>• Easiest contract type to manage</td>
<td>• Contract amount cannot be changed without contract modification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cannot easily change labor mix and # of hours</td>
</tr>
<tr>
<td>Cost Reimbursement Term (Level of Effort)</td>
<td>• Flexibility to change labor mix and hours under contract ceiling</td>
<td>• Contract ceiling may be difficult to establish, which can affect upfront fee determination</td>
</tr>
<tr>
<td></td>
<td>• Does not require a deliverable for payment</td>
<td>• Requires closer Gov’t monitoring</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Requires a certified cost accounting system among other FAR requirements</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Less incentivize for contractor to control</td>
</tr>
<tr>
<td>Time-and-Material (T&amp;M) (Labor Hour)</td>
<td>• Flexibility to change labor mix and hours under contract ceiling</td>
<td>• Unpopular contract type across the Gov’t</td>
</tr>
<tr>
<td></td>
<td>• Does not require a deliverable for payment</td>
<td>• Requires close Gov’t monitoring</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Contractor is not incentivized to control costs</td>
</tr>
</tbody>
</table>
Summary

- Using Agile development is an attractive option for IT programs
  - Regular capability deliveries
  - Responsive to changes in operations, tech, and budgets
  - Active user involvement and empowered teams

- Structure 6-12 month releases and tailor processes
- Dynamic and iterative requirements management
- Portfolio services contracting for industry partnership

- Tailoring DoD acquisitions to enable Agile adoption, successful IT

- For additional info, see MITRE Defense Agile Acquisition Guide
# Potential Agile Structure

## Technology Development, Competitive Prototyping, Architecture Development and Evolution

<table>
<thead>
<tr>
<th>FY14</th>
<th>FY15</th>
<th>FY16</th>
<th>FY17</th>
<th>FY18</th>
<th>FY19</th>
</tr>
</thead>
<tbody>
<tr>
<td>Materiel Development Decision</td>
<td>Program Backlog Validation</td>
<td>Development RFP</td>
<td>PEO Reviews</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solution Analysis</td>
<td>Program Planning</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Program Backlog – Continual Grooming and Prioritization

- **RB1**: Release 1
- **RB2**: Release 2
- **RB3**: Release 3
- **RB4**: Release 4
- **RB5**: Release 5
- **RB6**: Release 6
- **RB7**: Release 7
- **RB8**: Release 8

### Release Details

**Release Planning**

- Design Review
- Release Design, Development, Integration, Test, and Certification
  - Sprint 1
  - Sprint 2
  - Sprint 3
  - Sprint 4
  - Sprint n

**Design Review**

- Delivery
- Feedback

**Government Testing, Operational Assessments**

- Final Release Test and Certification

**Deployment Decision**
Potential Contract Construct

- **Portfolio-level agile development contract**
  - Quick execution of orders for each release (e.g., 6 months)
  - Single award for quick orders and consistent contractor
  - T&M for max flexibility (transition to FFP or CR after initial period)
  - Scope/requirements can adjust over time

- **Services-based contract**
  - Contract for the services of the development team
  - Cost-boxed and time-boxed releases and sprints
  - Requirements in product backlog are flexible
  - Structure releases (e.g. 6 months) via separate task orders
Agile Overview

- Leading software methodology – begin in 2001

- Core Elements
  - Small, frequent capability releases
  - Valuing working software over comprehensive documentation
  - Responding rapidly to changes in ops, technology, and budgets
  - Actively involving users throughout development

- Small, empowered, collaborative teams
  - Follow disciplined process
  - Dynamic, tailored, and evolving
  - Continual process improvement
Five Prerequisites for Agile Acquisition

1. Small, frequent capability releases
2. Embrace change
3. Partnership: requirements, acquisition, contractor
4. Small, empowered, high-performing teams
5. Leverage a portfolio structure