Disclaimer

• The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

• References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring.

• This material is furnished on an "as-is" basis. The author makes no warranties of any kind, either expressed or implied, as to any matter including, but not limited to, warranty of fitness for purpose or merchantability, exclusivity, or results obtained from use of the material.
Overview

• Agile Ecosystem
• Contractual Flexibility
• Contractual Flexibility Applied (FFP Examples)
• Measuring Value
• Top 5 Questions
• Conclusion/Looking Forward
Agile Ecosystem
Benefits of Agile

Benefits of Adopting Agile

By implementing agile, respondents cited seeing improvements in the following areas:

- Ability to manage changing priorities: 71%
- Project visibility: 66%
- Business/IT alignment: 65%
- Delivery speed/time to market: 62%
- Increased team productivity: 61%
- Team morale: 61%
- Project predictability: 49%
- Software quality: 47%
- Project risk reduction: 47%
- Engineering discipline: 43%
- Managing distributed teams: 40%
- Software maintainability: 33%
- Project cost reduction: 22%

*Respondents were able to make multiple selections.

Source: VersionOne 2017
Agile Manifesto

• The foundational document for Agile software development

• Signed by 17 software developers in Feb 2001

• Core Values
  • Individuals and interactions over processes and tools
  • Working software over comprehensive documentation
  • Customer collaboration over contract negotiation
  • Responding to change over following a plan

http://agilemanifesto.org/
**Organization Culture**

**Definition:** a system of shared assumptions, values, and beliefs, which governs how people behave in organizations

**Traditional Example(s):** Primary measure of success is conformance to a plan, top-down control, detailed requirements, develops in stages (Design, Development, Test, etc.), siloed organization/programs

**Agile Example(s):** Light-weight requirements gathering, continual user involvement, integrated and self-organizing/managing teams, primary measure of success is the continuous delivery of value

---

“Culture eats strategy for breakfast” – Peter Drucker
The Path to Agility

“What gets measured gets done.” – UNKN

Organizational Culture

Principles behind the Agile Manifesto

We follow these principles:

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development, for more adaptive software.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build around motivated individuals. Trust them, and let them get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The team should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity – the art of maximizing the amount of work not done – is essential.

Metrics

Definition: a standard of measurement by which efficiency, performance, process, quality or progress can be assessed

Traditional Example(s): Earned Value Management (EVM), SLOC count, schedule focused measures

Agile Example(s): Sprint burndown, release burndown, velocity, cumulative flow diagram, lead time, cycle time

Benefits of Adopting Agile

By implementing agile, respondents cited seeing improvements in the following areas:

- Ability to manage changing priorities: 71%
- Project visibility: 66%
- Business/IT alignment: 65%
- Delivery speed/time to market: 62%
- Increased team productivity: 61%
- Team morale: 61%
- Project predictability: 49%
- Software quality: 47%
- Project risk reduction: 47%
- Engineering discipline: 43%
- Managing distributed teams: 40%
- Software maintainability: 33%
- Project cost reduction: 22%

*Respondents were able to make the following improvements.
"Change is the only constant in life" - Heraclitus

The Path to Agility

Principles behind the Agile Manifesto

We follow these principles:
- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes surpass software projects with customer satisfaction.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for shorter intervals.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to, and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity-the art of maximizing the amount of work not done-is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Contracts

Definition: an agreement with specific terms in which there is a promise to do something in return for a valuable benefit

Traditional Example(s): Long contract durations with detailed requirements, contract by phase

Agile Example(s): Short-duration contracts, contract for teams/skillsets

Benefits of Adopting Agile

By implementing agile, respondents cited seeing improvements in the following areas:

- Ability to manage changing priorities: 71%
- Project visibility: 66%
- Business/IT alignment: 65%
- Delivery speed: 62%
- Increased team productivity: 61%
- Team morale: 61%
- Project predictability: 49%
- Software quality: 47%
- Project risk reduction: 47%
- Engineering discipline: 43%
- Managing distributed teams: 40%
- Software maintainability: 33%
- Project cost reduction: 22%

*Respondents were able to make attempts to improve these areas.*
The Path to Agility

Principles behind the Agile Manifesto

We follow these principles:

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity–the art of maximizing the amount of work not done–is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Methods / Practices

Definition: an essential support structure or approach to build something

Traditional Example(s): Waterfall, rational process

Agile Example(s): Scrum, Kanban, SAFe, LeSS, crystal, feature driven development, timeboxing, iterations, increments, daily meeting, definition of done, user stories, epics, dedicated teams

Benefits of Adopting Agile

By implementing agile, respondents cited seeing improvements in the following areas:

- Ability to manage changing priorities: 71%
- Project visibility: 66%
- Business/IT alignment: 65%
- Delivery speed/time to market: 62%
- Increased team productivity: 61%
- Team morale: 61%
- Project predictability: 49%
- Software quality: 47%
- Project risk reduction: 47%
- Engineering discipline: 43%
- Managing distributed teams: 40%
- Software maintainability: 33%
- Project cost reduction: 22%

*Respondents were able to make it
“Sunlight, healthy soil, clean air, and water **all work together** to help a seed become a strong, healthy plant.”
Agile Ecosystem

Principles behind the Agile Manifesto

We follow these principles:

1. Our highest priority is to satisfy the customer through early and continuous delivery of usable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity—the art of maximizing the amount of work not done—is essential.

Agile Ecosystem

Methods/Practices

Organizational Culture

Metrics

Contracts

Benefits of Adopting Agile

By implementing agile, respondents cited seeing improvements in the following areas:

- Ability to manage changing priorities: 71%
- Project visibility: 66%
- Business/IT alignment: 65%
- Delivery speed/time to market: 62%
- Increased team productivity: 61%
- Team morale: 61%
- Project predictability: 49%
- Software quality: 47%
- Project risk reduction: 47%
- Engineering discipline: 43%
- Managing distributed teams: 40%
- Software maintainability: 33%
- Project cost reduction: 22%

*Respondents were able to make up to 10% improvement in each area.*
Scrum Overview

• SCRUM is a proven framework for delivering value using agile methods and practices
• Provides actionable, artifacts, roles, and activities which can be measured and analyzed

Methods/Practices

24 Hours
Daily Scrum
2-4 Weeks
Sprint
Backlog
Items

Government

Product Owner
• Key stakeholder
• Represents the business or user community
• Manages the backlog (Requirements)

The Team
• Cross-functional
• Self-organizing
• Self-managing

Scrum Master
• Removes impediments
• Coach for the team
• Process owner

Government / Vendor

Government / Vendor

Key stakeholder
Key stakeholder
• Represents the business or user community
• Manages the backlog (Requirements)
Contractual Flexibility
Contractual Flexibility

• Definition: The contractual freedom for a contract administrator to respond to changing requirements post contract award. Contractual freedom has the objective of reducing or eliminating contract modifications due to unanticipated scope changes, while not incurring unnecessary risk.
Traditional vs “Flexible” Contracts

Traditional Example
1) Detailed requirements with a long contract duration

“Flexible” Examples
1) Detailed requirements with a short contract duration
2) Requirements flexibility post contract award
A “New” Source Selection Approach

• A new approach using new same system

30 No. 5 Nash & Cibinic Rep. NL ¶ 23

Nash & Cibinic Report
May 2016
The Nash & Cibinic Report
Competition & Award

¶ 23. “HIGHEST TECHNICALLY RATED OFFERORS WITH FAIR AND REASONABLE PRICING”: A New Source Selection Technique

Ralph C. Nash & Vernon J. Edwards
Contractual Flexibility Applied
(FFP Examples)
Contracting Requirements

• Operate and maintain a system-of-systems, comprised of over 90 servers, geographically distributed databases, and 14* applications (desktop and web based) that support the agencies core mission
  • The systems are tightly coupled, in that an update to one system may require complimentary updates to 1+ other systems
  • Contain highly sensitive data including personally identifiable information (PII) and range from legacy (15+ years old) applications/architectures and new applications/architectures (< 5-years new)

* 4 Applications selected for pilot, plan to extend Team Concept all 14 application
Contract Background

• Executed through a series of iterations on a BPA over a 2-year period
  • Total Award $13M

• All development was completed by the vendor

• All contracts required the vendor to conform to the Scrum Process (Artifacts and Events)

• All orders were executed with the same:
  • Systems
  • Federal Employees
  • Vendor (4 of 5 Iterations)
Progression of Contracting Actions

• **Agile Façade (Not Agile):** Remove traditional terminology (PM, Requirements, etc.) and add agile terminology (Scrum Master, User Stories, etc.)

• **AGILE contracting:** Let multiple short contracts/actions to fulfil a given capability

• **Agile CONTRACT:** Provide freedom within a given contract to change requirements
Agile Contracting Journey

**Agile Facade**

**Expect:**
Low Requirements Flexibility
Delays Due to Contracting Actions

**Use When:**
Immature Agile Processes
Low Government Involvement
Teams are New to Agile

---

**AGILE Contracting**

**Expect:**
High Requirements Flexibility
Little/No Contract Related Delays

**Use When:**
Mature/Managed Agile Processes
High Government Involvement (Daily)
Teams are Skilled Agile Practitioners

---

**Agile CONTRACT**
FFP - Release

- Requirement: Upgrade a large legacy application to be 508 compliant in one release. The deliverables were fixed at contract start so there was no flexibility after contract award.

- Key Facts:
  - Cost: $1.2M
  - Duration: 9-Months
  - Contract Modifications: 1
FFP – User Story

• Requirement: Develop a fixed set of User Stories for a software release. The government could determine NOT to execute a user story (provided work had not started) but could not add user stories

• Key Facts:
  • Total Cost: $234,419.84
  • Low User Story: $850.82
  • High User Story: $71,314.08
  • Contract Modifications: 1
• Requirement: Develop a fixed number of user stories over the course of the contract
• This was rolled into a larger CLIN so accurately calculating the cost is not possible
• Key Facts:
  • Cost: $N/A
  • Low User Story: N/A
  • High User Story: N/A
  • Contract Modifications: N/A
FFP – Complexity Levels

• Requirement: Develop user stories based on contractually defined complexity levels until the “not to exceed” (NTE) amount was reached or additional funds were applied

• The PWS contained complexity levels using “Representative User Stories” and when new backlog items (requirements) arose, the new requirement(s) were assigned a complexity level at the contracted cost

• Key Facts:
  • Complexity Levels
    • Extra Low: $2,205
    • Low: $3,383
    • Medium: $9,532
    • High: $14,863
    • Very High: $22,550
FFP – Teams

• Requirement: Supply agile teams to deliver value within contractually defined technical constraints

• Initial contract cost was 10% less than the previous approach for the same capacity

• Teams are required to be cross-functional
  • Each team member was required to have a minimum of two skill sets (Developer and tester, Technical writer and business analyst, etc.)
  • Specific skillsets were required per team but the vendor proposed the overall team structure(s)
Responding to Change (Team Approach)

Option #1: Repurpose Existing Team(s)

<table>
<thead>
<tr>
<th>PRO</th>
<th>CON</th>
</tr>
</thead>
<tbody>
<tr>
<td>No additional cost</td>
<td></td>
</tr>
<tr>
<td>Quickest to implement</td>
<td></td>
</tr>
<tr>
<td>Most efficient</td>
<td>Decreased O+M work</td>
</tr>
</tbody>
</table>

Operations and Maintenance Team

Enhancement Team

Option #2: Stand Up New Agile Team(s)

<table>
<thead>
<tr>
<th>PRO</th>
<th>CON</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimal impact to O+M</td>
<td>Additional cost</td>
</tr>
<tr>
<td>Minimal contract overhead</td>
<td>Ramp-up time</td>
</tr>
</tbody>
</table>

Enhancement Team (Optional CLIN)
Agile Team Considerations

• You have to account for part-time (architects, security personnel, etc..) team support

• Plan for a team ramp-up time to allow for staffing
  • 30-60 days of LH
  • Provide long lead-time when starting a new team

• Success relies heavily on:
  • Product Owner
  • An organization with mature agile processes
Measuring Value
Measuring Performance

User Story: As an Analyst, I want to enter the planned follow-up date, regardless of the commitment date.

Definition of Done:
- Help Text updated including 508 Help
- Focused Regression Test
- 508 Test
- System Test
- Updated Requirements
- Updated Test Case / Scripts
- Update EA Design Document
- Coding/Unit Testing
- Print Crystal Report
- Release Notes
- Update EA Data Dictionary

Acceptance Criteria:
- Change the label from 'Est. Follow-up Date' to 'Planned Follow-Up'
- Make Planned Follow-Up an editable date field. Use standard date format with calendar
  - 'Planned Follow-Up' must not default to any value.
- Display error message and prevent save if there is a Completed Date but the Planned Follow-Up is missing.
  - Error message: 'Planned Follow-Up is required if the Completed Date is entered.'
  - After user responds to error message attach tooltip icon with associated tooltip text to Planned Follow-Up.
  - Tool tip text: 'Planned Follow-Up is required if the Completed Date is entered.'
Measuring Performance per Sprint

• Step 1: The Product Owner (PO) determines the highest priority user stories from the Backlog and defines Acceptance Criteria for the Sprint.

Three typical outcomes at the Sprint Review:
1) The user story is accepted by the PO as it meets the acceptance criteria and Definition of Done (DoD)
2) The user story is rejected by the PO because it DOES NOT MEET the acceptance criteria and DoD specified at the start of the sprint and this may be noted as a vendor performance issue
3) The user story is rejected by the PO but MEETS the acceptance criteria and DoD which does NOT reflect on the vendor performance since this was a requirements specification issue (Rare).
Increased Transparency

- The delivered value is measured every Sprint (2-weeks)
- Complete transparency into the release status

| Application Name |

| Release Summary |

<table>
<thead>
<tr>
<th>SPRINT</th>
<th>DATES</th>
<th>PLANNED STORY POINTS</th>
<th>ACCEPTED STORY POINTS*</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPRINT 1</td>
<td>7.5.18 – 7.18.18</td>
<td>50</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>SPRINT 2</td>
<td>7.19.18 – 8.1.18</td>
<td>55</td>
<td></td>
<td>[Comments Removed]</td>
</tr>
<tr>
<td>SPRINT 3</td>
<td>8.2.18 – 8.15.18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPRINT 11</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Accepted Story Points refer to stories that have been accepted by Product Owner

| Snapshot of Current Sprint Items |

- Sprint Planning (Sprint 2) Committed User Stories (Summary)

| Release Comments |

[Comments Removed]
Top 5 Questions
Do you have a Template?

• No ... BUT we can provide examples of what worked in our unique ecosystem.
  • Every program has multiple aspects that are unique to them, simply filling-in-the-blanks without analyzing your environment will likely cause undesired outcomes. Things to consider:
    • Organization culture
    • Maturity of government oversight personnel
    • Structure of government program oversight
    • Maturity of the vendor pool
What do the Examples Contain

• Task Summary
• Instructions to Offerors
• CLIN Structure
• Evaluation Criteria

Technical Landscape

The Vendor is expected to be proficient in the following ... As technology evolves, so must the technical expertise and the tools utilized by the Vendor. It’s expected the Vendor will evolve as technology evolves throughout the period of performance of the Contract...

<table>
<thead>
<tr>
<th>Name</th>
<th>Manufacturer</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>.NET Framework</td>
<td>Microsoft</td>
<td>Development Framework</td>
</tr>
<tr>
<td>Adobe Acrobat Pro</td>
<td>Adobe</td>
<td>Standard OCC-approved PDF software</td>
</tr>
<tr>
<td>AjaxControlToolkit</td>
<td>Open source</td>
<td>Third party add-in</td>
</tr>
<tr>
<td>Angular</td>
<td>Open source</td>
<td>A structural framework for dynamic web apps</td>
</tr>
<tr>
<td>Apache log4net</td>
<td>Apache</td>
<td>Logging service</td>
</tr>
<tr>
<td>Aspose cells</td>
<td>Aspose</td>
<td>A third-party tool used to manipulate MS Excel files</td>
</tr>
</tbody>
</table>
Why did you use FFP vs LH?

• Enables stable teams
  • 2 x as Productive

• Enables multidisciplinary team members
  • When compared to labor categories

• Focuses more on team outcomes versus individual contributions

• Low contract monitoring overhead, the focus is shifted to helping deliver value

• Provides the same requirements (Scope) flexibility
What are your Deliverables (CDRLS/DIDs)?

• Very few contractually specified deliverables
  • Most deliverables are included in the User Stories Acceptance Criteria and Definition of Done post contract award
    • This allows us to pay for what we need in-real-time versus what we think we need in the future

• Example Deliverables
  • Rejected user stories
    • If they don’t meet the Acceptance Criteria or Definition of Done
  • Team Metrics
    • Velocity
    • Sprint Burndown
    • Release Burnup
    • Committed Story Points vs. Accepted Story Points per Sprint
What is the CLIN structure?

• Various options

<table>
<thead>
<tr>
<th>CLIN</th>
<th>Description</th>
<th>QTY</th>
<th>Unit of Measure</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001 (Required)</td>
<td>Agile Teams (5x9-members, 4x7-members, etc.)</td>
<td>1</td>
<td>EA</td>
<td>$0.0</td>
</tr>
<tr>
<td>0002 (Optional)</td>
<td>Agile Team (5-members)</td>
<td>n</td>
<td>EA</td>
<td>$0.0</td>
</tr>
<tr>
<td>0003 (Optional)</td>
<td>Agile Team (7-members)</td>
<td>n</td>
<td>EA</td>
<td>$0.0</td>
</tr>
<tr>
<td>0004 (Optional)</td>
<td>Agile Team (9-members)</td>
<td>n</td>
<td>EA</td>
<td>$0.0</td>
</tr>
</tbody>
</table>
Is this performance-based contracting?

- Modification from the traditional performance based contracting model
- Give the vendors freedom to propose solutions within certain guideposts
- Could you make this approach truly performance based?
Conclusion
Future Applications of Agile Teams

• Anywhere you require teams that:
  • Are Multidisciplinary
  • Have a fixed capacity
  • Have requirements that continually change
  • Have a continual backlog of requirements
The Path to Agility

Principles behind the Agile Manifesto

We follow these principles:

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter time scales.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity—the art of maximizing the amount of work not done—is essential.

Agile Ecosystem

Methods/Practices

Organizational Culture

Metrics

Contracts

Benefits of Adopting Agile

By implementing agile, respondents cited seeing improvements in the following areas:

- Ability to manage changing priorities: 71%
- Project visibility: 66%
- Business/IT alignment: 65%
- Delivery speed/time to market: 62%
- Increased team productivity: 61%
- Team morale: 61%
- Project predictability: 49%
- Software quality: 47%
- Project risk reduction: 47%
- Engineering discipline: 43%
- Managing distributed teams: 40%
- Software maintainability: 33%
- Project cost reduction: 22%

*Respondents were able to make improvements in these areas.*
Makoto P. Braxton

- Makoto P. Braxton is the Director of the Contracts and Grants Division within the Office of Resource Management at the Assistant Secretary for Preparedness and Response (ASPR). ASPR is a Staff Division within the Department of Health and Human Services. The Contracts and Grants Division provides acquisition and assistance support across the entire ASPR organization. The Division supports major Program Offices within the Division of the Strategic National Stockpile, the Office of Emergency Management and Medical Operations, as well as the Office of the Biomedical Advanced Research and Development Authority.

- Makoto has been working in the contracting field since 2005. He started with the Federal Government in 2008. Previously, Makoto has worked for the Department of the Air Force, the Department of the Army, and the Office of the Comptroller of the Currency (within the Department of the Treasury). Makoto holds a Bachelor of Science in Business Administration from the University of Arizona and is Level III Certified in Contracting in accordance with the Defense Acquisition Workforce Improvement Act, as well as Federal Acquisition Certification in Contracting Level III Certified.
Matthew R. Kennedy, PhD

- Matthew R. Kennedy is a Senior IT Program Manager and Contracting Officer Representative (COR) at the Office of the Comptroller of the Currency (OCC). Formerly, Matt was a Program Manager at the Army's Program Executive Office - Enterprise Information Systems (PEO-EIS) and was a Professor of Software Engineering at Defense Acquisition University (DAU) where he specialized in agile acquisition. Matt served as the Associate Director of Engineering at the National Cancer Institute’s Center for Biomedical Informatics and Information Technology and served in the U.S. Air Force as a network intelligence analyst. He has worked both inside and outside of the government on various IT projects over the last 18 years.

- Matthew holds a Bachelors in Computer Science, and a masters and PhD in Computer Science and Software Engineering from Auburn University. He is Defense Acquisition Workforce Improvement Act (DAWIA) Level III certified in Program Management, Systems Engineering, and Information Technology (IT).

Contact: MatthewRKennedyEmail@gmail.com