BETTER BUYING POWER THROUGH THE USE OF AGILE ACQUISITION STRATEGIES

Dr. Julie A. DeSot
2016 Project Management Symposium
What is Agile?

- Time-boxed, iterative approach to software delivery

  Increment → Increment → Increment

  Instead of all at once

- Seeks alternatives to traditional project management
After a release we go to release planning for another release or if it is the final release we close out the project.
Agile Manifesto (2001)

• Purpose:
  – Promote set of compatible values and principles in software development
  – Focus on providing customer with best possible product
  – Promoting open communication, trust, respect, and flexibility among development teams

• Based upon an empirical approach
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 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.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.
Agile Values
Why Agile?

• Allows for more flexibility in requirements and development
• Includes built-in increments and iterations
• Provides opportunity to encounter and address errors sooner in development cycle
• Increases organizational team efficiency
• Decreases unnecessary documentation and meetings
• Provides value-based approach to development
Agile Project Success

2009 CHAOS Report

- Successful: 32%
- Challenged: 24%
- Failed: 44%

2010 Agile Project Survey

- Successful: 55%
- Challenged: 35%
- Failed: 10%

The Standish Group (Levinson, 2009)
Dr. Dobb's: The World of Software Development, 2010
Perceived Project Success Rates

- **Iterative**: 12% Failed, 31% Challenged, 67% Successful
- **Agile**: 12% Failed, 29% Challenged, 61% Successful
- **Ad-Hoc**: 15% Failed, 40% Challenged, 53% Successful
- **Traditional**: 18% Failed, 41% Challenged, 52% Successful

Note: Accurate to within +/-7%
Figures don’t add to 100% due to use of “ranged options”
Development Approaches

Predictive    Adaptive
Project Constraints & Drivers

Traditional:
Scope drives Time & Cost

Agile (Adaptive):
Time & Cost drive Scope
AGILE ADOPTION
Project Failures

Reasons for Failure

- Development method
- Shortened testing schedule
- Contracting process
- Project Complexity
- Inability of innovative firms to bid on major / other portions
- Stove-piped applications
- Non-integrated, sub-optimized IT solutions

Lessons Learned

- Take incremental approach to development
- Avoid risk by using betas, early testing, and regular delivery of a completely tested system, in tandem with flexible scope
- One budget authorization for agency’s IT spending and give CIO sole budget authority
Project Successes

Challenges

• Prior to adopting Agile:
  – Long delivery cycles
  – Low adoption rates
  – Mismatches between delivered functionality and end user expectations

• After adopting Agile:
  – Needed full commitment to ensure the success of transitioning to Agile
  – Training
  – Team processes

Outcomes

• Reduced cycles times
• Focus on meeting users’ requirements
• User forum
• SME hired to work alongside developers
• Continuous introspection and improvement
Doing Agile vs. Being Agile

• Following Agile process vs. Fully embodying Agile culture
• Agile is a mindset
• Culture of agility as a strategy
  – Nimbleness; embrace change
Agile in Federal Environment

- GAO
  - Identified 32 best practices
    - Align with 5 key software development project management activities
- OMB
  - OMB Contracting Guidance to Support Modular Development
- DoD
- FAR
  - 39: Modular Development / Contracting
Modular Development

*Modular contracting is the use of one or more contracts to acquire IT systems in successive, interoperable increments (FAR 39.002)*

- Intended to:
  - Reduce program risk
  - Incentivize contractor performance
  - Meet government’s need for timely access to rapidly changing technology

- Acquire major IT systems
  - Consistent with agency’s IT architecture

- Acquire non-major IT systems (FAR 39.103)
ACQUISITION STRATEGIES
Contracting Approaches

- IDIQ
- BPA
- FFP
- T&M / LH
- CP
IDIQ

• Indefinite-Delivery Indefinite Quantity
• FAR 16.5
• Single contract with options
• Successive contracts
• Performance-based work statements
BPA

• Blanket Purchase Agreements
• One-time BPA competition
• Recurring needs for supplies & services
• Quantity discounts
• Admin time saving
FFP

- Firm-Fixed Price
Needs Identification

- In Agile, needs are identified and requirements are developed through Progressive Elaboration. Each Iteration or Sprint allows business users/stakeholders to better define their needs to ensure the most effective development of solutions, in addition to allowing developers to ensure the criteria for completion and to ensure requested capability through the testing iterations:

- The testing iterations or sprints rely on the user stories that the business user/stakeholder communicates as important features to concentrate on; these are features users value and interact with directly

- "These short scenarios of user expectations are just part of the user story process. User stories include two additional elements:
  - Notes from further discussions about the story that help to clarify the expectations (Conversation)
  - Intent of the story and validation tests that will confirm to the user that the story, when delivered, does what it is expected to do (Confirmation)"
Stakeholder Analysis

• Just like with any project, the IPT has to balance the interests of many different project stakeholders. Some of these stakeholders may have competing goals and interests, and many may have limited resources. An effective stakeholder analysis can help the IPT, and specifically the program manager, allot appropriate attention to each stakeholder, and decide how best to mitigate conflicting stakeholder interests when they happen.

• The stakeholder analysis process will document important details about different project stakeholders, and summarizes the needs of each. Ultimately, the IPT needs to capture each stakeholder's success criteria for the project, as well as their impact on the work, strategies for keeping them up-to-date at the appropriate level, and the relative priorities.
Stakeholder Analysis, continued

• Identify the project's stakeholders
  – Stakeholder requirements elicitation
  – Document and Communicate
Multiple, Rapid Delivery of Capability

- Agile development provides opportunities to assess the direction of a project throughout the development life cycle.
- This is achieved through regular cadences of work, known as sprints or iterations, at the end of which teams must present a potentially shippable product increment.
Multiple, Rapid Delivery of Capability, continued

• Agile architecture practices that enable speed and stability:
  – Release planning with architecture considerations
  – Prototyping with a quality attribute focus
  – Roadmap/vision with external dependency management
  – Test-driven development with quality attribute focus
  – Technical debt monitoring with quality attribute focus
Delivering Capabilities Quickly to the User

- Organizations must make problems more visible to developers, management, and stakeholders. When considering whether to combine Agile and architecture practices, organizations should consider the following questions:
  - Are we delivering software to our customer at an expected pace?
  - Are we aware of problems that are cropping up as a result of losing focus on designing when Agile adoption activities become the primary focus?
  - Does our technical roadmap address short-term and long-term issues?
  - Does the team of software developers have skills that would enable them to successfully implement Agile and architecture?
  - Do we have the visibility into both the project management of the system and the quality expected from the system?
Delivering Capabilities Quickly to the User, continued

- The focus on deconstructing the requirements into smaller and smaller chunks has a beneficial effect on the ability to keep to the contracted needs in order through result of stakeholder analysis and needs identification. It becomes very easy to track the delivery of the contracted deliverables through each development cycle, lowers resistance, and any detailed change orders that result also have a fine granularity of time and cost impact.

- Agile can only work in an environment that eschews the boundaries of process, procedure, policy, practices, and politics and focuses on what makes the most sense to get the job done for the end user. For federal managers of Agile teams, this is very hard to accept. The loss of direct visibility and the ability of the team to shift priorities and deliverables are sometimes too much for project management offices.
Delivering Capabilities Quickly to the User, continued

• With traditional Waterfall development, information and data is not readily available because the team only prepares data documents based on limited knowledge. By closely collaborating with all participants and stakeholders, Agile methods allow the IPT to acquire ample information and data and get many more questions answered quickly and accurately. During Scrums, team members can discuss the information each team member was able to gather and discuss a path forward for completion of the task at hand.

• Close collaboration with participants and stakeholders (as well as fellow teammates and the management team) also helps to identify current and potential stumbling blocks. The team has the opportunity to identify these stumbling blocks and discuss ways to resolve or prevent them. Also, risks can be identified and mitigated when teams collaborate regularly.
### Operating Models & Organizational Structure Changes

<table>
<thead>
<tr>
<th>Before</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Organizational structure</strong></td>
<td><strong>Product-based focus</strong> with stable teams and dedicated resources; end-to-end perspective</td>
</tr>
<tr>
<td>Application-oriented focus, with ever-changing teams and pooled resources; siloed perspective</td>
<td>Product-based focus with stable teams and dedicated resources; end-to-end perspective</td>
</tr>
<tr>
<td><strong>Interactions between business and IT</strong></td>
<td>Development process is managed by strong product owner from business, who works closely with IT at all stages</td>
</tr>
<tr>
<td>Development process is managed by proxy product owner from IT, with input as needed from business</td>
<td>Development process is managed by strong product owner from business, who works closely with IT at all stages</td>
</tr>
<tr>
<td><strong>Roles and responsibilities</strong></td>
<td>All roles are integrated within self-organizing scrum teams; project-manager role is minimized and line managers focus on capability building</td>
</tr>
<tr>
<td>Scrum teams comprise developers and testers; project-manager and line-manager roles remain unchanged from waterfall approach</td>
<td>All roles are integrated within self-organizing scrum teams; project-manager role is minimized and line managers focus on capability building</td>
</tr>
<tr>
<td><strong>Budgeting and planning</strong></td>
<td>Venture-capital-style budgeting, where minimally viable product is launched and future funding depends on</td>
</tr>
<tr>
<td>Traditional yearly budgeting, with fixed budget allocated to projects</td>
<td>Venture-capital-style budgeting, where minimally viable product is launched and future funding depends on</td>
</tr>
</tbody>
</table>
SUCCESSFUL AGILE PLANNING TEAM
Team Characteristics
Project Management Plan Integration

- “The Agile approach is best suited for projects of an experimental nature incorporating new or untried technology, in which change or refinement of the requirements will be a necessary aspect of the defined product release.”

- For example, building a software application may benefit from using Agile since the desired end-state system may be known, but the details of the technical solution will have to be determined using a sequence of tightly defined iterative loops, or possibly parallel project teams working in tandem on a variety of sub-systems.
Stakeholder / Customer Involvement

- Most customers are not very good at providing details about what they want, yet most stakeholders are fairly good at indicating what they think they want—then, when an option is presented, stakeholders provide feedback for what they do and do not want about the option.

- It is important to work with stakeholders to identify what they think they want, produce something which reflects that understanding, get feedback from stakeholders, and update the solution to reflect an improved understanding. Thus, working in a more evolutionary and collaborative manner helps to provide solutions which reflect stakeholders' actual needs.
Agile Culture / Training

• As the Agile team is built, the project manager needs to codify the characteristics that the team should embody. Capturing these characteristics in the project charter and sharing them with the team is necessary to make sure all team members have the proper focus on the values critical to success.

• “The key ingredient within any successful Agile team is absolute clarity about the goal and objectives. The project manager must ensure that all team members have a shared vision of success and that they all know what needs to be done. In essence, the PM will provide the 'what' while the team provides the 'how'. Part of the leader's role is to help the team not only understand the result or ultimate goal, but also to help them understand the process goals they need to achieve to guarantee success. A process goal is a leading metric, it is a guide that continually informs the team that what they are doing is right.”
Agile Advocate (AA)

• Ensure stakeholders' needs are understood
  – Empathize with stakeholders' perceptions, fears, and requirements
  – Create ways of managing and reporting risk, scope, cost, time, and quality—not from the technical in-team perspective, but from a senior management perspective
  – Differentiate between problems suitable for Agile solutions and those that need a different approach

• AAs focus on areas where real value is created by using Agile
Consistent Teams

- The most important aspect of a successful Agile team is the people. A software development team needs talented people who are committed to the project, the mission, and who produce consistent results. Skilled developers are required to build complex systems using new technology. One or two people will not suffice for building these complex systems—a team is necessary. Therefore, the developers that are needed must also be skilled in working on a team.

- Ultimately, successful project managers and sponsors develop consistency through meritocracy not governance. The key challenge to a meritocratic process, though, is communication.

- The organization must have open, consistent, frequent, and quality communication among teams to maintain engaged team members that have the freedom to deliver, and will not be looking for the exits. Turnover on Agile teams is a dangerous possibility.
Integrated Project Team (IPT) Responsibilities

• The responsibilities of the Agile Integrated Project Team (IPT) are pretty straight-forward: deliver results.

• An Agile team works as a cohesive unit, not a loose grouping of individuals. It is important that the team strives to become cross-functional, if they are not so already.

• Being cross-functional helps the IPT avoid being blocked and assists with the flow of work, ensuring the IPT delivers value to the customer consistently and frequently, thus enabling a positive feedback loop.
Overall Responsibilities

• Describe the problem (ensure agency mission focus)
  – Examine public/private sector solutions
  – Describe the work and define the strategy
  – Decide how to measure and monitor performance
  – Select the contractor
  – Manage performance
  – Define roles for government business owners as participants in the process delivery (demo review)
Best Practices for Managing an IPT Team

- Best practices for Agile software at the federal level normally entail use of the Agile Software Development Life Cycle (ASDLC) to describe an Agile life cycle customized to the needs, parameters, and culture of the organization.

- Using the ASDLC helps to:
  - Increase user involvement
  - Prioritize business requirements
  - Develop useable software code in working applications
  - Deliver frequent releases

- The ASDLC is anchored to the Project Management Accountability System (PMAS) to serve as a framework for positive transformation in current and future projects. The ASDLC seeks to create a common ground for project development and mitigate the issues between disparate project development practices.
Milestone Versus Process

• The traditional Waterfall view of projects:
  – Maintains an "on time/in budget/manage to the plan“ approach
  – Requires deterministic activities to be successful
  – Is used in many modern IT projects in the federal government

• Software development:
  – Is a non-deterministic activity
  – Is a process of change, invention, and design
  – Has "unknown unknowns“ at the start of the project, some of which are never fully understood until the completion of the project
Milestone with Time-based Economic Impact

- An Agile team needs to complete features in a software project for a demonstration at an upcoming annual agency convention.
- If the software is late, the organization will lose an opportunity to demonstrate new product features and corporate technology progress to a large audience.
- If the product is completed too early, there is value lost in terms of opportunity cost.
Shared Vision and Approach

- Agile teams and organizations using Agile produce their best work when their members work both individually and collectively toward a shared goal. The Agile principle of simplicity—the art of maximizing the amount of work not done—applies here. When all team members share a common vision, more work that matters to the ultimate goal gets done, and less work that is extraneous gets done.
Grounded Visioning

- Grounded Visioning is a group process that quickly, and enjoyably, helps teams define a shared vision in two hours or less. It overcomes the isolation many team members feel when disconnected from a clear and compelling understanding of the team’s shared goals. The method also scales up to work with large groups, and can be comfortably used to find common ground in groups of 250 or more, in just four hours or less.

- The Grounded Visioning process follows six simple steps:
  1. Assemble your stakeholders.
  2. Ignite your spark.
  3. Describe best practices.
  4. Share your dreams.
  5. Select the best.
  6. Plan next steps.
Team Collaboration

• If you look at the principles behind the Agile Manifesto, you will see that at least five of the 12 Agile Principles require that teams be highly collaborative:
  – Satisfy the customer through early and continuous delivery
  – Welcome changing requirements
  – Business people and developers work together daily
  – The best way to convey information is face-to-face
  – The team regularly tunes and adjusts its behavior
Team Collaboration, continued

• Becoming collaborative is a two-pronged task:
  1. The organization has to support collaboration and reward it.
  2. The individuals need to develop the habits and behaviors that yield collaboration.

• The organization needs to make the team responsible—to give the team authority as a whole, rather than having a single “boss” make all the important decisions. The team itself is measured by its success or failure, and the team members are measured by their ability to make the team work better, not by an individual idea that was promoted or criticized.
Agile Coach

- The Agile Coach role is part embedded trainer, part consultant—specifically an advisor. Even the best Agile training courses cannot cover every detail or roadblock a team will encounter, so the Coach is there to continue the training after the formal classes are over.
- The Agile Coach helps teams apply Agile and Lean thinking to the specific environment and impediments that they face.
- Working as an advisor, the Coach can help the team adapt the methodology to their environment, and help the team challenge the existing environment.
- Taken together, these two sides make the Coach an effective change agent—someone who is both motivating change and making it happen. The fact that an organization is prepared to spend money on a Coach demonstrates that they are serious about making the change happen.