

Agile-Waterfall Hybrid

Jessica LaGoy, MS, PMP



About Jess

- **BS – Applied Physics, WPI / MS – Cybersecurity, UMUC**
 - **PMP, ITIL, Data Scientist, Tableau, Alteryx**
 - **Project Experience**
 - **Data and technology**
 - **Construction**
 - **Education**
 - **Non-Profit**
 - **Down time**
 - **Dogs and chaos**
 - **Muay Thai**
 - **Non-fiction addict**
- 

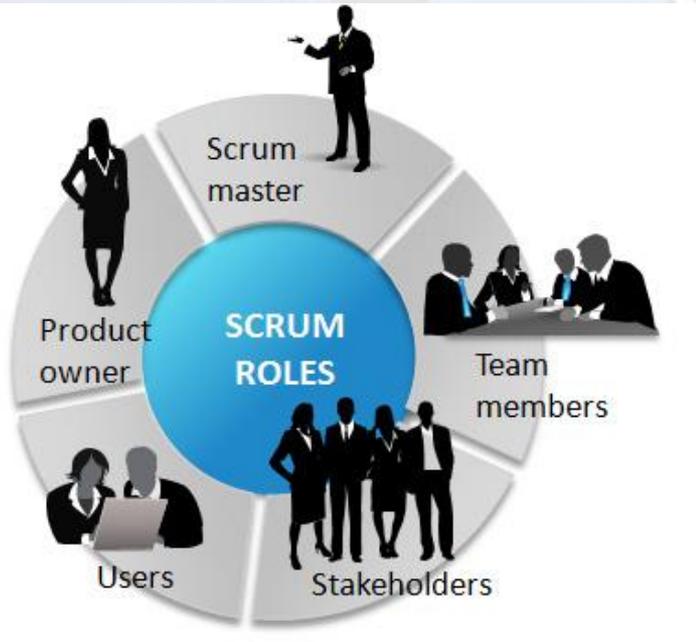
Working Double Duty

Senior Program Manager turned Senior Data Intelligence Engineer @ EMC

- Technical resource for data intelligence
- Program manager for global analytics projects
 - Global Services
 - Customer Services
 - Collaboration across multiple teams
 - Matrix management environment
 - Projects include development of tools, processes, reporting structures, and the actual data solutions

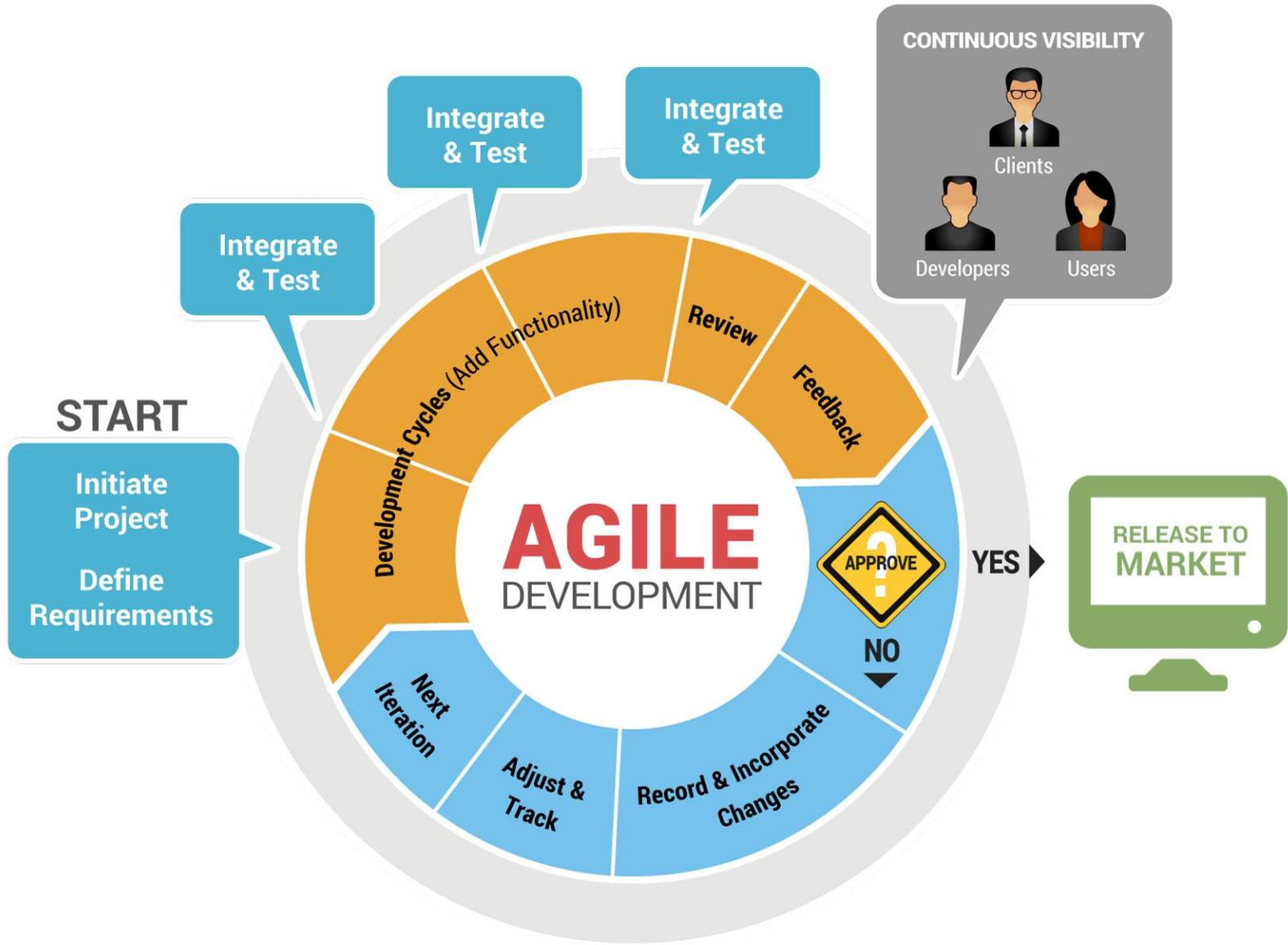
Agile Basics

Project work that follows a cyclical path over multiple iterations. Less rigid and able to evolve during the project execution.



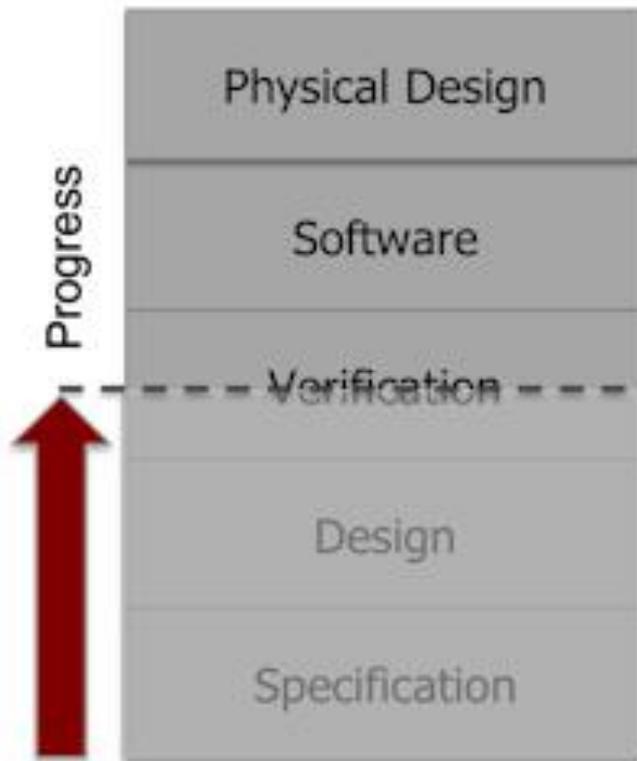
- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Agile Basics



Agile vs Waterfall

- Waterfall Model



- Agile model



WATERFALL



Waterfall requires detailed planning at the beginning of a project.



All the steps are laid out, dependencies mapped, and you move to the next stage only after completing the previous one.



Pros.



Best for projects that deal with physical objects – from a construction project to a hardware installation project.



Best for projects with defined tasks and phases that must be completed in a specific sequence (e.g., build the first floor of a building before the second floor).



Project plans are repeatable for identical or similar projects in the future.



Cons.



Requires substantial scope and schedule planning before work begins.



Scope changes can be slow and require formal change control processes.



Less effective for software, design and other non-physical or services-based projects.

AGILE



This is a fast and flexible approach to project management based on principles of collaboration, adaptability and continuous improvement.



Unlike the orderly stages of a waterfall approach, agile project management is typically set up in quick, iterative project release cycles.



Pros.



Best for projects that deal with services-oriented and non-physical deliverables like code, copywriting and design projects.



Allows for quick course correction based on stakeholder feedback.



Empowers project teams to work creatively and efficiently.



Includes engagement and collaboration from all team members.



Cons.



Not suited for projects with strictly defined requirements and scope.



Uncertainty around scope and schedules can make stakeholders and executives nervous (at first).



Requires vigilant backlog and documentation maintenance, and tech debt management.

The project management method you choose will vary based on the project, your team and goals.

Once you select a planning style, make sure you use project management software that lets you and your team set up your projects the way you want.

Looking at Hybrid Solutions

Existing Use Cases

- HW/SW combined solutions/products

Resource for Determining Fit

- <http://www.bpminstitute.org/resources/articles/hybrid-sdlc>

My Experience Deploying

- Data/Business intelligence projects

Deploying a Hybrid Solution

- Team not co-located
- Project not a SW only solution
- Need faster development, testing, deployment
- Need immediate relief from current situation
- Want more input & involvement from users
- Not enough resources to have dedicated to this project only / sharing resources for other functions and responsibilities
- Not possible to have all information up front due to nature of deliverables or solution desired

Selling the Idea of Stepping Away from Waterfall

Points Highlighted

- Slow progress on project/program from previous team
- Resource availability/constraints
- Nature of deliverables/requirements
- Time constraints/need for urgency
- Benefits of allowing additional user input/involvement
- Benefits of ongoing development / testing for quality and improved business value
- Reduced risk of failure
- Earlier usability/replacement of non-functional existing options

Selling the Idea of Stepping Away from Waterfall

Keeping Everyone Happy

- Organization of processes, requirements, and deliverables in phases
- Review of contents of each phase near end of previous phase
- Create phased rollout of documentation required for business to align with priorities
- Increased communications to users about process and progress
- 1-2 calls per week for updates by group/function instead of daily scrums for larger group

Success!

Success due to ability to move further faster with limited resources and not having to experience delay from requirements gathering and testing

1st Quarter – storming and struggle, but large steps forward while accumulating requirements

2nd Quarter – significant functionality improvements, high levels of user satisfaction, flushed out hidden requirements

3rd Quarter – 6 months ahead of schedule from original plan prior to takeover/transition to hybrid methodology

Cheat Sheet

The 12 Agile Principles

The 12 Agile Principles are a set of guiding concepts that support project teams in implementing agile projects. Use these concepts to implement agile methodologies in your projects.

- 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.**

Cheat Sheet

- THE ROADMAP TO VALUE -

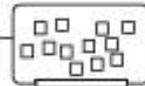
Stage 1: VISION

Description: The goals for the product and it's alignment with the company's strategy
Owner: Product Owner
Frequency: At least annually

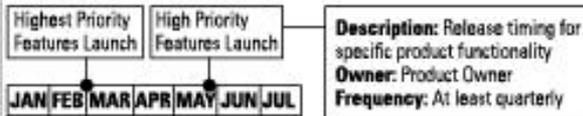


Stage 2: PRODUCT ROADMAP

Description: Holistic view of product features that create the product vision
Owner: Product Owner
Frequency: At least biannually



Stage 3: RELEASE PLANNING



(Stages 1-3 are best practices outside of core Scrum)

Stage 4: SPRINT PLANNING



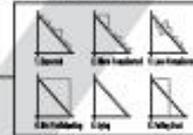
Description: Establish specific iteration goals and tasks
Owner: Product Owner and Development Team
Frequency: At the start of each sprint

Preparation

Execution

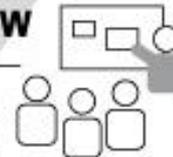
Stage 7: SPRINT RETROSPECTIVE

Description: Team refinement of environment and processes to optimize efficiency
Owner: Scrum Team
Frequency: At the end of each sprint



Stage 6: SPRINT REVIEW

Description: Demonstration of working product
Owner: Product Owner and Development Team
Frequency: At the end of each sprint



Stage 5: DAILY SCRUM



Description: To establish and coordinate priorities of the day
Owner: Development Team
Frequency: Daily

SPRINT



Release Product
 [Per the Release Plan]

Cheat Sheet

Agile Project Management Artifacts

Project progress needs to be measurable. Agile project teams often use six main artifacts, or deliverables, to develop products and track progress, as listed here:

- **Product vision statement:** An elevator pitch, or a quick summary, to communicate how your product supports the company's or organization's strategies. The vision statement must articulate the goals for the product.
- **Product backlog:** The full list of what is in the scope for your project, ordered by priority. Once you have your first requirement, you have a product backlog.
- **Product roadmap:** The product roadmap is a high-level view of the product requirements, with a loose time frame for when you will develop those requirements.
- **Release plan:** A high-level timetable for the release of working software.
- **Sprint backlog:** The goal, user stories, and tasks associated with the current sprint.
- **Increment:** The working product functionality at the end of each sprint.

Cheat Sheet

Agile Project Management Events

Most projects have stages. Agile projects include seven events for product development. These events are meetings and stages and are described in the following list:

- Project planning:** The initial planning for your project. Project planning includes creating a product vision statement and a product roadmap, and can take place in as little time as one day.
- Release planning:** Planning the next set of product features to release and identifying an imminent product launch date around which the team can mobilize. On agile projects, you plan one release at a time.
- Sprint:** A short cycle of development, in which the team creates potentially shippable product functionality. Sprints, sometimes called *iterations*, typically last between one and four weeks. Sprints can last as little as one day, but should not be longer than four weeks. Sprints should remain the same length throughout the entire projects.
- Sprint planning:** A meeting at the beginning of each sprint where the scrum team commits to a sprint goal. They also identify the requirements that support this goal and will be part of the sprint, and the individual tasks it will take to complete each requirement.
- Daily scrum:** A 15-minute meeting held each day in a sprint, where development team members state what they completed the day before, what they will complete on the current day, and whether they have any roadblocks.
- Sprint review:** A meeting at the end of each sprint, introduced by the product owner, where the development team demonstrates the working product functionality it completed during the sprint.
- Sprint retrospective:** A meeting at the end of each sprint where the scrum team discusses what went well, what could change, and how to make any changes.

Cheat Sheet

The Agile: Scrum Framework at a glance

Inputs from Executives,
Team, Stakeholders,
Customers, Users

PDT Lead
Engr Pgm Mgr
Engineering Mgr
Prod Release Team Mgrs



Burndown/up
Charts



Daily Scrum
Meeting



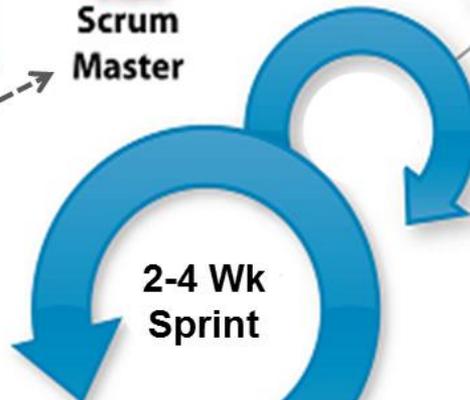
Product Owner



The Team



Scrum
Master



2-4 Wk
Sprint



Sprint Review



Product
Backlog



Sprint
Planning
Meeting



Sprint
Backlog

Sprint end date and
team deliverable
do not change
Task Completion



Finished Work



Sprint
Retrospective

Begin Next Sprint