

Agile Essentials

Track: Business Services

Presenter: Mark Thomas

Synopsis

Are you a victim of building the wrong solutions slowly? If so, you're not alone, and considering an Agile approach may be the right fit for your organization. The term Agile is used a lot today, and this is an opportunity to hear about what it is, as well as understand its applicability, success factors, and potential pitfalls. This presentation will also do an overview of the different 'flavors' of Agile and their characteristics.

Agenda

● Introduction & Overview

● *The Agile Manifesto*

● *Agile Elements and Team*

● *Agile Flow*

● *Closing and Questions*

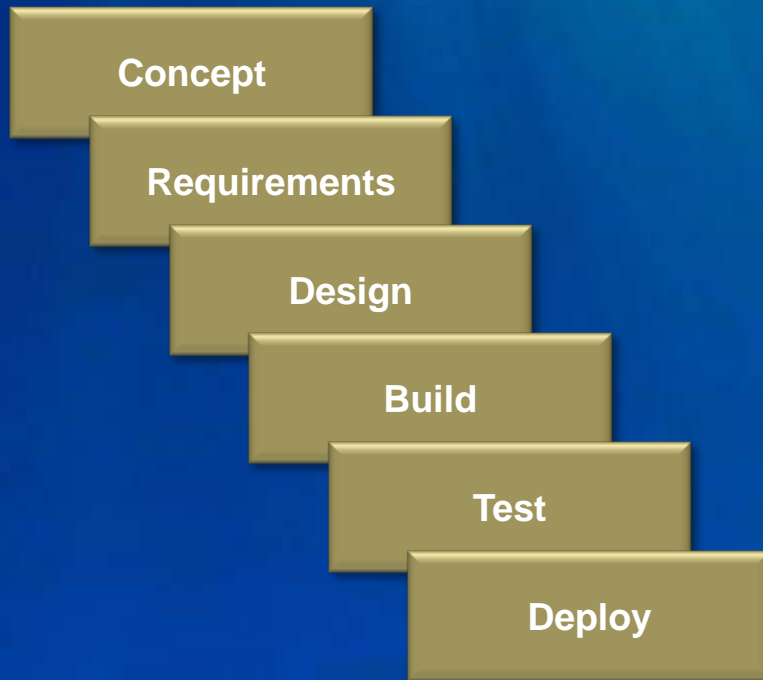
Why Projects Fail

When asked why projects fail, many organizations cite a wide range of issues. Essentially, these issues can be consolidated into the following:

- Inadequate, poor, or inconsistent requirements.
- Lack of customer or end user involvement in the process.
- Unrealistic or unattainable project schedules.
- Unmanageable scope creep due to the lack of change control.
- Insufficient testing.

How We Do It Today

Most organizations follow the tried and true “waterfall” approach, also known as SDLC (System Development LifeCycle).



- Assumes requirements are understood up front and are relatively stable.
- Assumes software can be “manufactured.”
- Emphasizes Big-Design Up Front (BDUF) with step-by-step execution.
- De-couple architecture and design from coding and testing.
- Different teams for different aspects.

SDLC Challenges

Of course this is not an issue with all users of SDLC; however, there are some inherent challenges that need to be overcome:

- Inherent focus on technical development challenges as opposed to user adoption and measureable benefits.
- Limited flexibility in change management.
- Customer involvement is limited to certain points along the lifecycle which causes potential scope issues.
- Testing is often a late phase which can create challenges with scope and schedule.

Agile as a Solution

Considering the failure rate of today's projects, Agile techniques can address key issues.

- Complete customer involvement by making the customer a member of the Agile team.
- Acceptance testing is defined as requirements are gathered and tests written before code is developed.
- Project schedules are negotiated using a collaborative process.
- Project Management is integrated into the process and not a separate activity.
- Change is expected and embraced.

Waterfall – Agile Comparison

Typical Waterfall

Agile

Following a strict Plan	----->	Plan as we go
Formal Checkpoints	----->	No Checkpoints
Big upfront design	----->	Agile modeling
“Big Bang” delivery	----->	Many small releases
Strict documentation	----->	Streamlined and adaptive documentation

Can You Make the Change?

It may be difficult to 'make the switch' due to organizational policies, process and culture. Some reasons why you may have to still use a waterfall type approach can include:

- Existing projects, processes, and tools.
- Externally dependant groups using waterfall.
- Leadership's need to plan for annual project funding and resource allocation.
- Adequate training funding for Agile transition.
- Policy or regulatory concerns.
- Current investment in development maturity (CMM).

Agile Types

Various Agile methodologies share much of the same philosophy as well as many of the same characteristics and practices. The most popular including:

SCRUM

A lightweight framework with broad applicability for managing and controlling iterative and incremental projects of all types. Scrum's popularity is increasing in the software community due to its proven simplicity, productivity.

XP

Delivers high-quality software quickly and continuously by promoting high customer involvement, rapid feedback loops, continuous testing, continuous planning, and close teamwork to deliver working software at very frequent intervals.

CRYSTAL

Comprised of a family of methodologies whose that addresses the realization that each project may require a slightly tailored set of policies, practices, and processes in order to meet the project's unique characteristics.

DSDM

Specifically calls out "fitness for business purpose" as the primary criteria for delivery and acceptance of a system, focusing on the useful 80% of the system that can be deployed in 20% of the time.

FDD

A model-driven, short-iteration process that begins with establishing an overall model shape followed by a series of "design by feature - build by feature" iterations.

LEAN

Focuses the team on delivering value and efficiency to the customer. Lean eliminates waste by selecting only the truly valuable features for a system, prioritizing those selected, and delivering them in small batches.

Agenda



The Agile Manifesto

In 2001, seventeen software developers met to discuss lightweight development methods. From this meeting, they published the following Agile Manifesto:

- We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
 - *Individuals and interactions over processes and tools*
 - *Working software over comprehensive documentation*
 - *Customer collaboration over contract negotiation*
 - *Responding to change over following a plan*
- That is, while there is value in the items on the right, we value the items on the left more.

The 12 Principles

1. Customer satisfaction by rapid delivery of useful software
2. Welcome changing requirements, even late in development
3. Working software is delivered frequently (weeks rather than months)
4. Working software is the principal measure of progress
5. Sustainable development, able to maintain a constant pace
6. Close, daily co-operation between business people and developers
7. Face-to-face conversation is the best form of communication (co-location)
8. Projects are built around motivated individuals, who should be trusted
9. Continuous attention to technical excellence and good design
10. Simplicity
11. Self-organizing teams
12. Regular adaptation to changing circumstances

Agenda

- 
- Introduction & Overview*
 - The Agile Manifesto*
 - Agile Elements and Team**
 - Agile Flow*
 - Closing and Questions*

Key Terms

Iterative development

Dividing projects into sprints that have fixed duration (time boxing) and helps development teams to focus on a shippable product in the end of each sprint.

Sprint

Fixed-length iterations typically two weeks to 30 days long with the goal of building a potentially shippable product.

Ceremonies

The following ceremonies are typically planned with Scrum: sprint Planning, Review (with Retrospective), and Daily Scrum.

- Sprint Planning Meeting.
- Daily Scrum Meeting.
- Sprint Review.
- Sprint Retrospective.



Artifacts

An artifact is something created for a practical purpose. In Agile, there are three key artifacts:



- Product Backlog.
- Sprint Backlog.
- Burndown Charts.

Themes, Epics, User Stories

Themes

A top level objective that may span multiple projects and products. Themes may also be Epics.

Epics

Epics are groups of related User Stories. Epics are not introduced into a sprint without first breaking it down into its component User Stories to reduce uncertainty.

User Stories

Independent, negotiable, valuable, estimatable, small, testable requirements. They are used during sprints and broken down into tasks.

User Stories

As the customer develops the user stories, they are written down on a note card (3x5) with a name and description. User stories generally follow the following template:

“As a <role>, I want <goal/desire> so that <benefit>.”

Some examples include:

- “As a customer representative, I want to search for my customers by their first and last names.”
- “As a user closing the application, I want to be prompted to save if I have made any changes in the data.”
- As a team member, I want to modify my schedule but not the schedules of the other team members.”

Estimation

There are three main concepts in estimation:

Estimation of size



Estimation of size gives a high-level estimate for the work item, typically measured using a neutral unit such as points.

Velocity



Velocity tells us how many points this project team can deliver within an iteration.

Estimation of effort



Estimation of effort translates the size (measured in points) to a detailed estimate of effort typically using the units of Actual Days or Actual Hours. The estimation of effort indicates how long it will take the team members to complete the assigned work items.

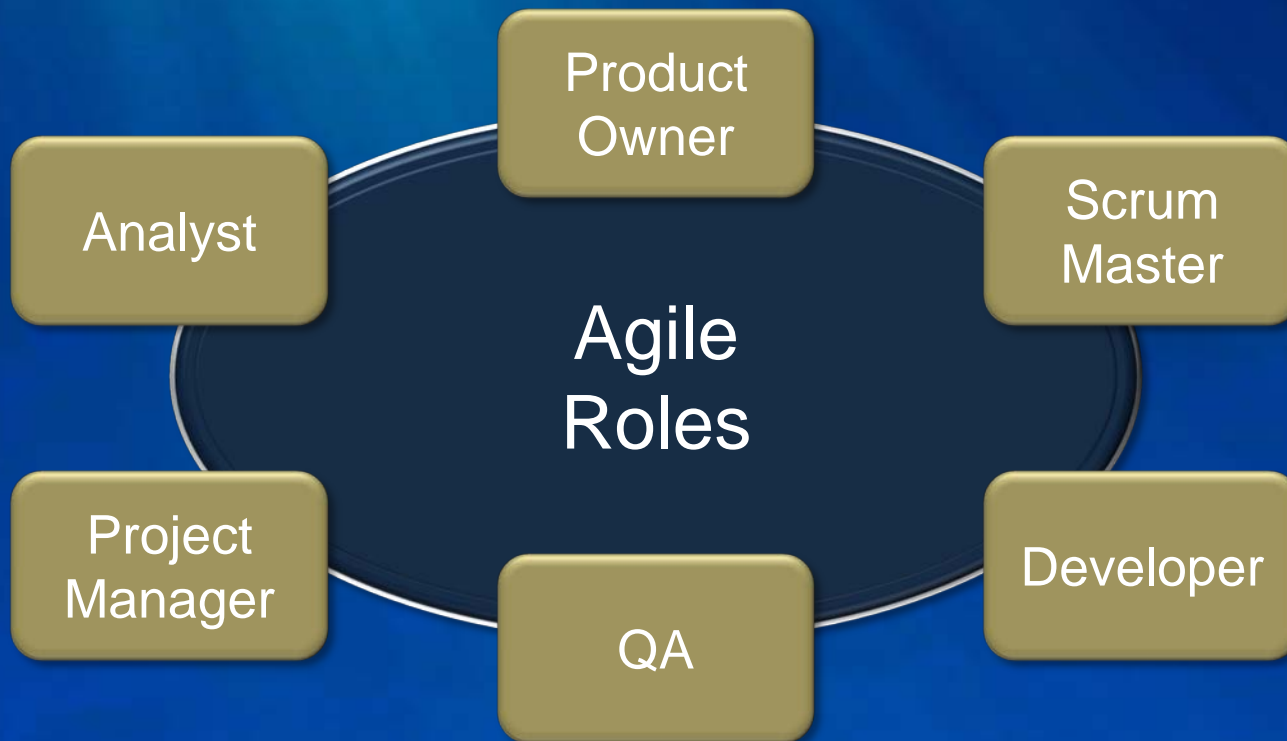
Agile Team Characteristics

Moving to Agile requires a paradigm shift, and part of that shift is the acceptance that some project roles have changed. Some key characteristics of the Agile team include:

- Collaborative, self-organizing and disciplined.
- Empowered by each other and by Management.
- Committed to delivering Sprint Goals.
- Trusting of one another and reliable.
- Responsible and accountable as a Team.
- Constructive in their criticism of themselves and each other.
- Creative, innovative, and multi-skilled.
- Business-value oriented.

Roles

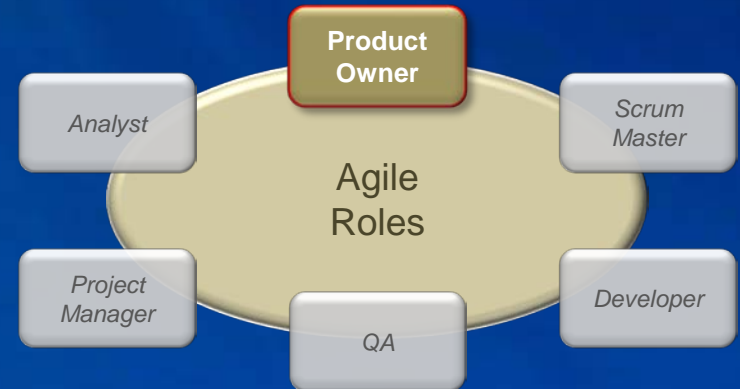
There are many roles associated with Agile development. A few key roles include the following:



Product Owner

The Product Owner is the customer representative who shapes the overall goals and vision of the project or initiative.

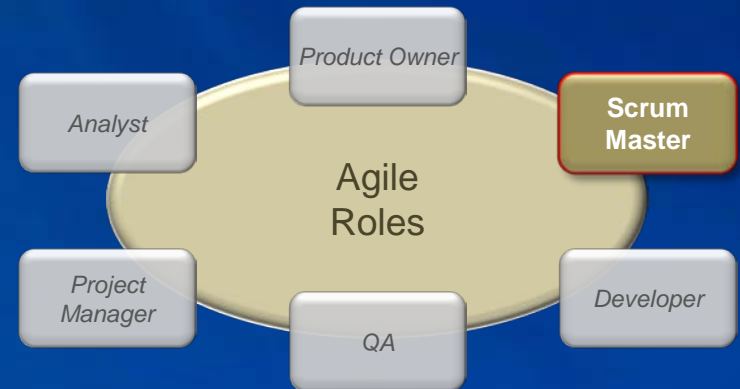
- Provides product vision.
- Ensures the team is moving toward its goal.
- Understand, identify, prioritize and communicate requirements and the product backlog.
- Ensures the team attains the appropriate return on investment.
- Provides realistic boundaries (time, cost, speed, etc.).



Scrum Master

The SCRUM Master guides teams rather than directs them.

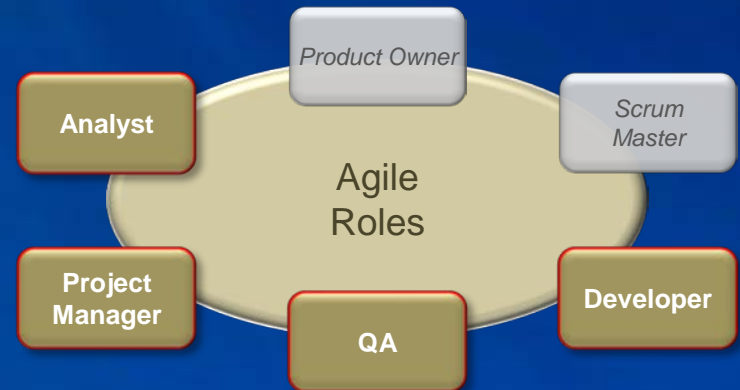
- Removes impediments to the team's progress.
- Has authority over the process (duration of sprints, etc.), but encourages the team to decide on their own.
- Solid technical knowledge does not necessarily equate to Scrum Master.



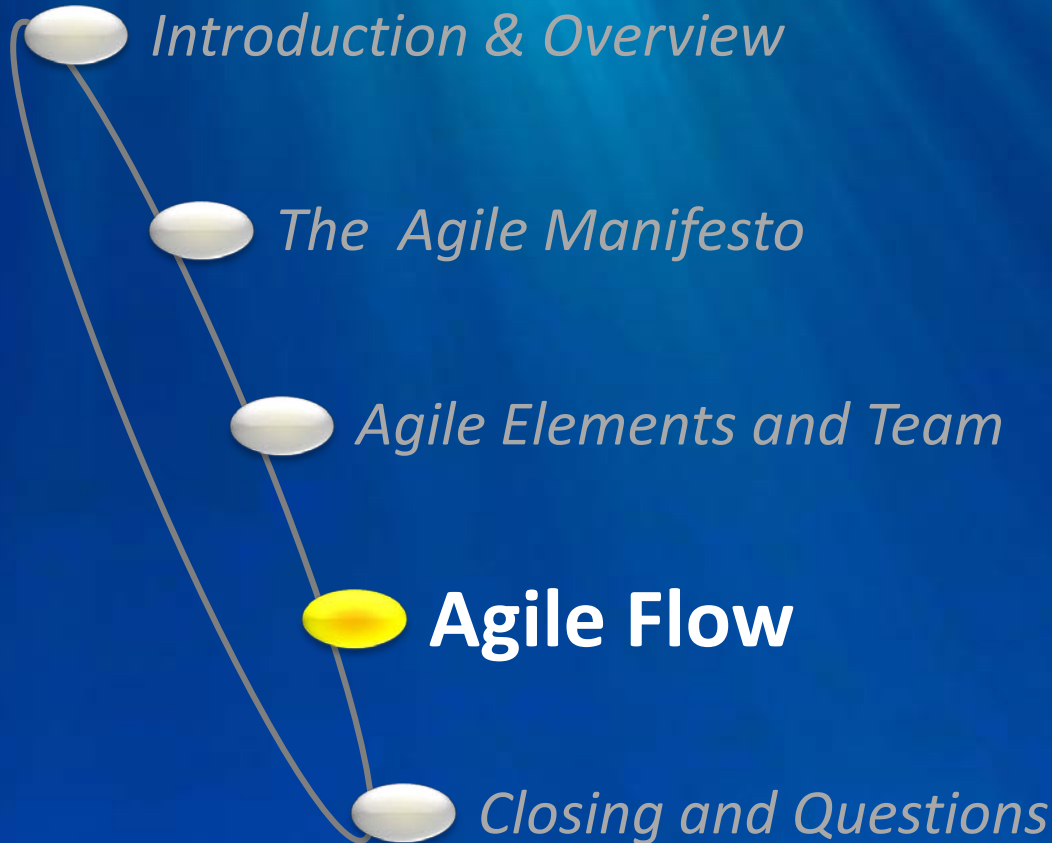
The Agile “Team”

This is a cross functional group who do the actual analysis, design, development, testing, and implementation.

- Multiple roles exist, including developers, QA, Project Management, Analyst (BA), etc.
- Others can include roles such as architect, technical managers, SMEs,
- The team is responsible for delivering the product.



Agenda

- 
- Introduction & Overview*
 - The Agile Manifesto*
 - Agile Elements and Team*
 - Agile Flow**
 - Closing and Questions*

Agile Flow



The customer puts together a prioritized list (feature backlog) of desired features for the upcoming product release.

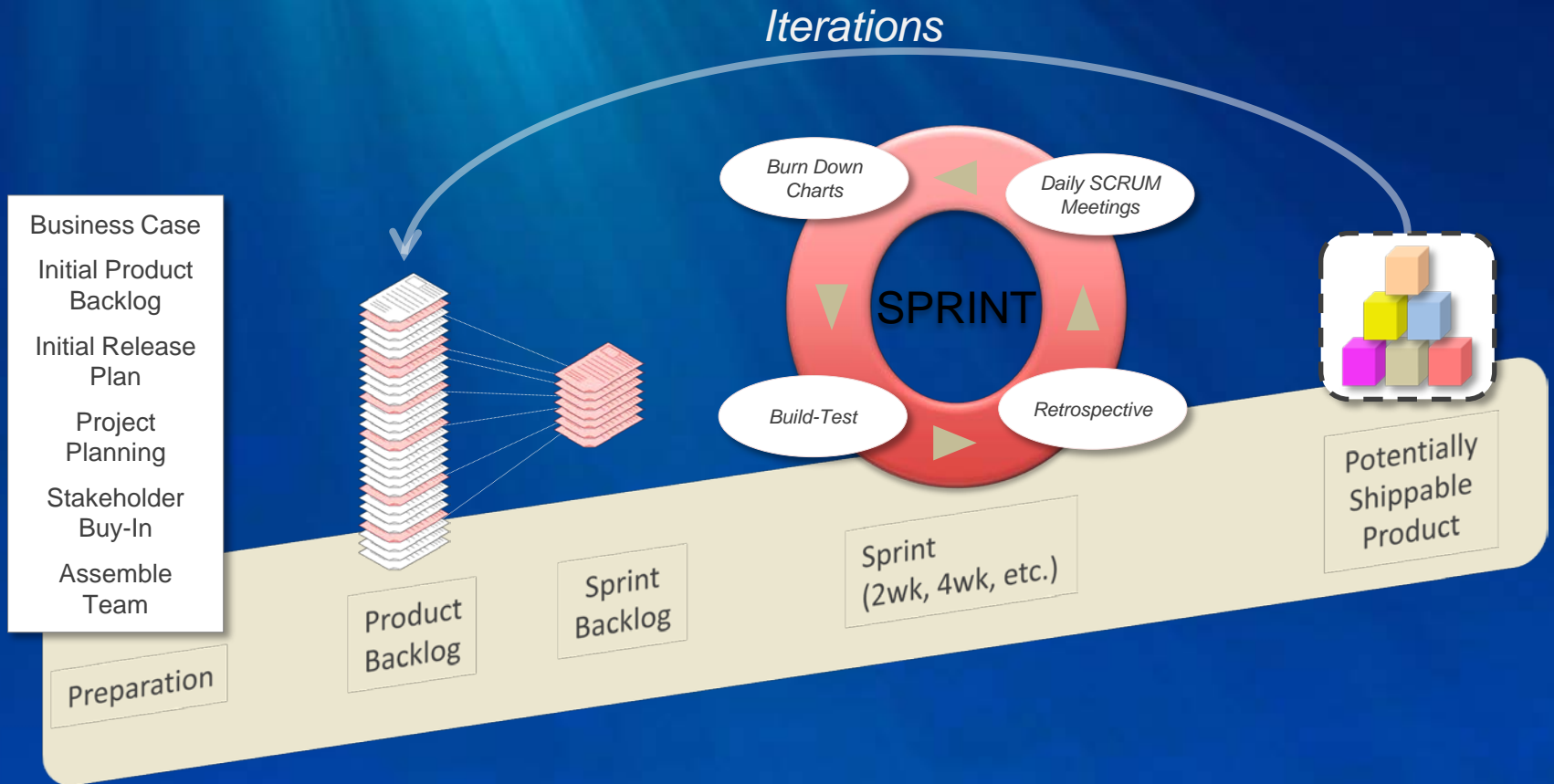
The release is broken into iterations, and the team and customer agree on what will be delivered at the start of each iteration.

The iteration is of fixed length. The team begins gathering data, which in turn they feed back into their estimates.

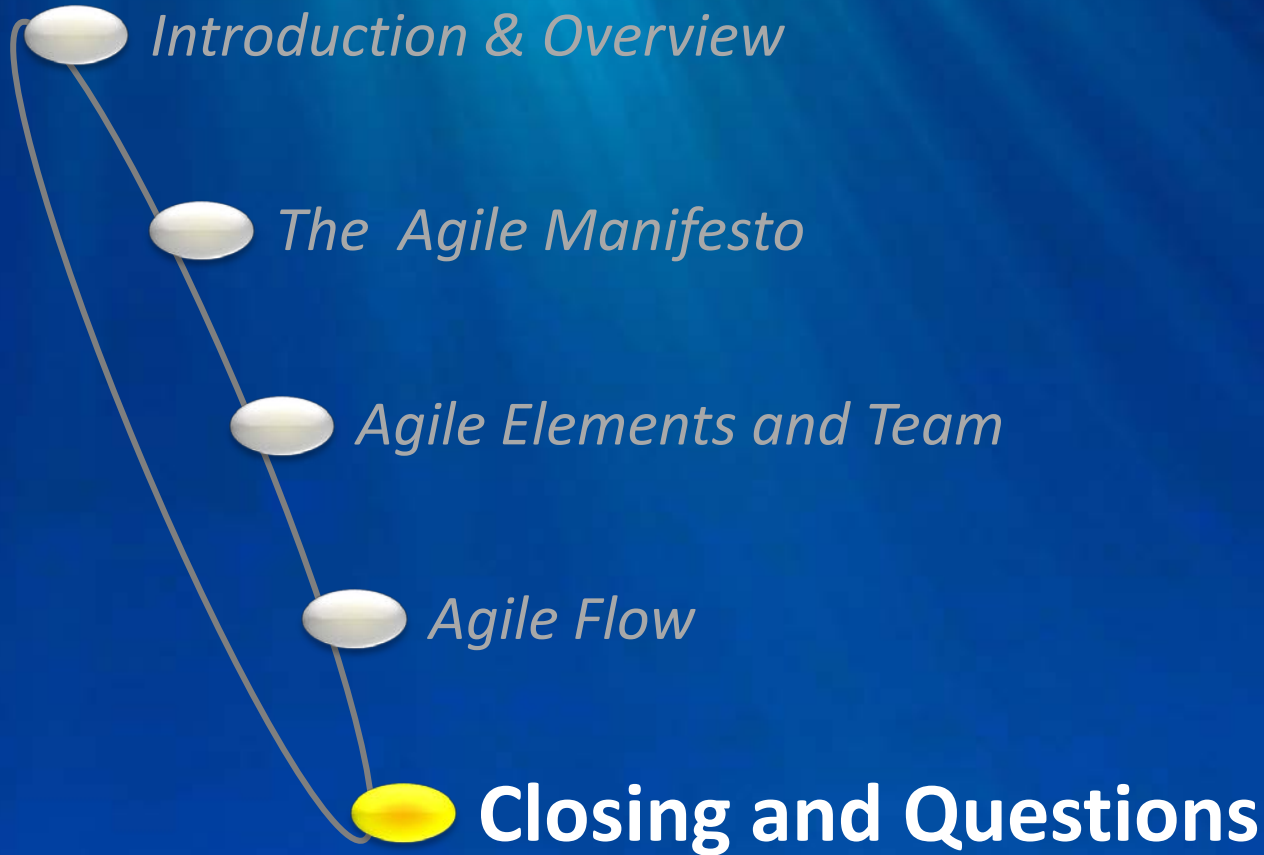
Upon incrementing and iterating the software, the software reaches a state that it may be released to the customer.

If the software passes all acceptance tests – then release to production.

Agile Flow



Agenda



Potential Challenges

Challenges	Strategies
Culture doesn't support change.	Communicate, communicate, communicate. Train the team, customer, and organization on Agile principles. Energize the team by rewarding and recognizing the team. Continue to collaborate internally as well as with the customer. Support change. Strive to automate as much as possible.
Ineffective or non-existent retrospectives.	
Lack of collaboration in planning.	
Product Owners not available or none exists (also, too many Product Owners).	
Ineffective Scrum Master.	
Individual heroics rewarded.	
Revert to traditional styles.	
Limited resources.	

The background of the slide is a deep blue underwater scene. Sunlight rays stream down from the surface, creating a shimmering, ethereal effect. The water's surface is visible at the top, with gentle ripples and light reflections. The overall mood is serene and professional.

Thank you.