

Adopting Agile Methods

Agile Project Management

Agenda

- What is Agile
- Why do we use it?
 - 80/20 Rule
 - Acceptance of Change
 - Risk Management
- How do we manage it?
 - At the project level
 - At the business level
- How do we get our teams to use it and use it effectively?
 - Learning in minimal environments
 - Transition to distributed operations
 - Just in Time Training

What Is Agile ?

Agile is the ability to:

Have rapid response to customers and markets

Undertake real-time analysis and decision making

Have effective collaboration and coordination

Explore, simulate, experiment

Have rapid access to information

Reconfigure processes during operation

Have continuous communication

Reconfiguration of resources

Characteristics of Agility

- Capable of operating profitably in a competitive environment of continually, and unpredictably, changing customer opportunities
- Capable of contributing to the bottom line of a company that is constantly reorganizing its human and technological resources in response to unpredictably changing customer opportunities

The Necessary Paradigm Shifts

From

To

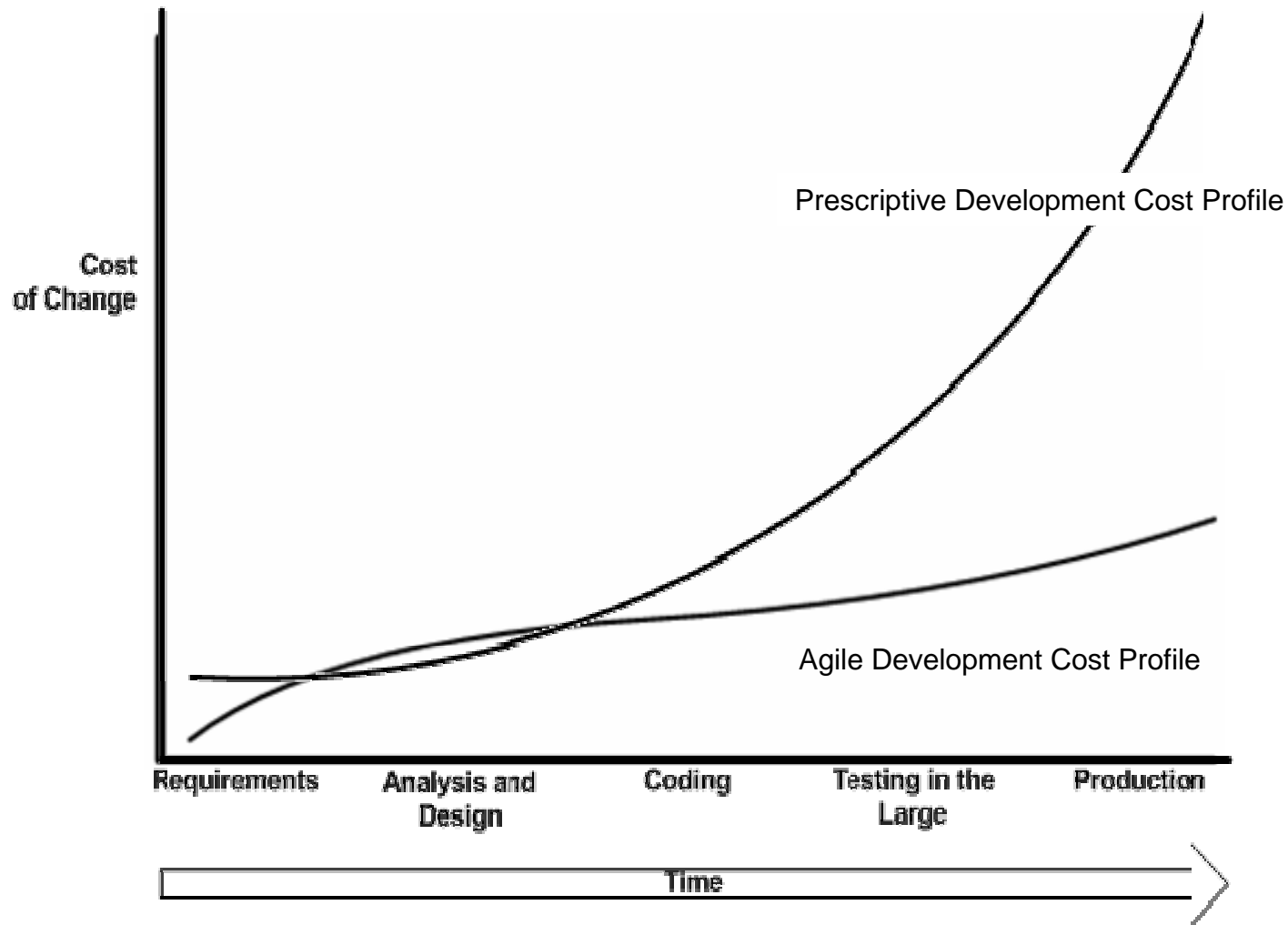
Implementing Requirements
Project Management
Agile Projects
Outsourcing
Compliance

Driving Competitiveness
Portfolio Management
Agile Systems
Strategic Sourcing
Improvement

Why do we use an agile approach?

- 80/20 Rule
 - 20% of the features are used 80% of the time
 - Finish that 20% first and put it into use, providing quick returns
 - Re-evaluate remaining requirements
- Accommodate Change
 - Business can add new requirements and reprioritize existing ones
 - Short list of organizational limitations to changes
 - Allows Business to drive the projects
- Risk Management
 - Impact of changes is minimized
- Improvement of Qualities

Cost of Change



Agile Benefits over Plan Driven

	Plan Driven	Agile	Benefits
Project Management	Produce Deliverables	Produce Software	More accurate estimates More accurate status
Design	All up front Approve documentation	Part of smaller increments Approve software	Reduced missed or false features
Development	After design Staged Progress measured by task % complete Individuals	Includes design Continuous Progress measured by passed tests Co-ownership	Increased visibility Reduce impact of staff changes Increased productivity
QA	Follows development	Throughout the project lifecycle	Improved quality
Change Management	Change is a problem	Change is expected Part of the process	Lower cost Faster time-to-market Lower project risk

Agile as risk management

- Every Agile technique addresses some project risk
 - What if some team doesn't deliver on time?
 - Always focus on the highest value features and deliver fully-tested software every two weeks
 - This way, we will always have a working version of the most important functionality.
 - What if a mandatory change is introduced at the eleventh-hour?
 - Implement the simplest possible design and develop a comprehensive automated test suite so that we can minimize the cost of introducing changes.
 - What if we lose a key person half-way through the project?
 - Adopt a collective code ownership approach, program in pairs and develop a comprehensive automated test suite
 - This fosters a much higher breadth of knowledge transfer amongst the team members

Improvement of Qualities

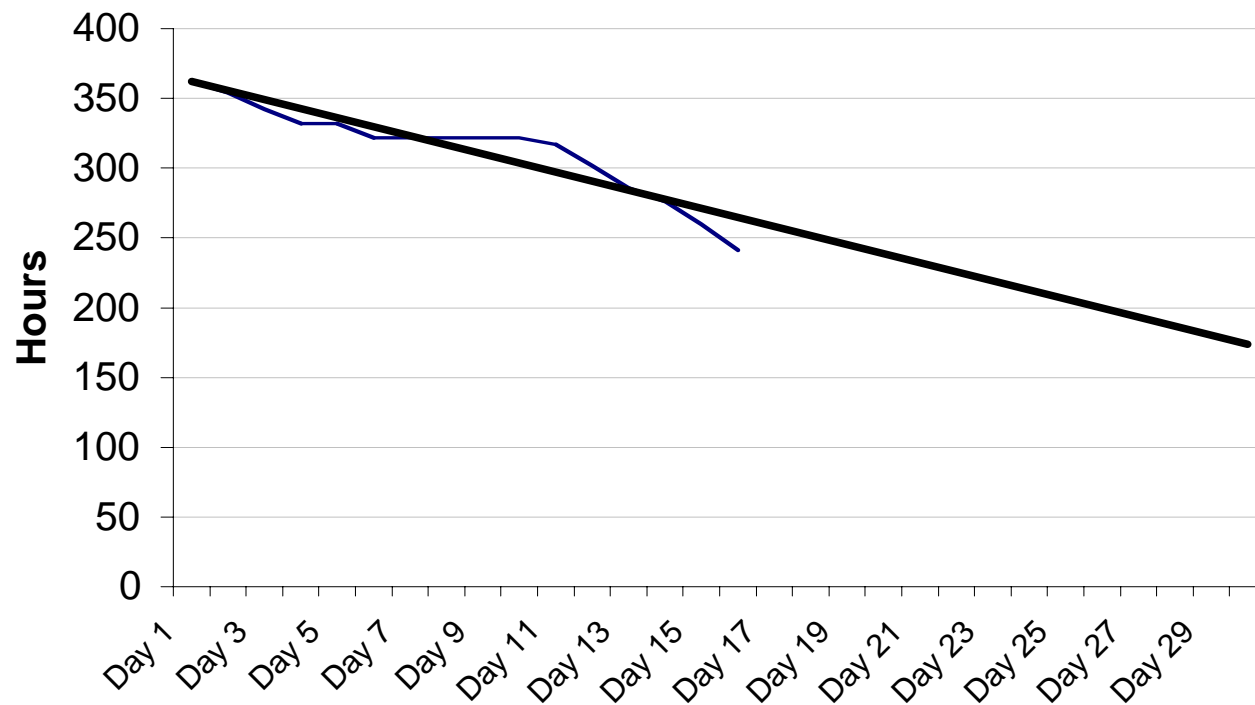
- Meet customer expectations
- Reduce missed and false features
- Lower released defect rates
- Increased maintainability
- Increased quality of life

How do we Manage the Process

- At the project level
 - Transparency
 - Standups
 - Burndowns
 - Feedback
 - Maturity
- At the business level
 - Transparency
 - Burnups
 - Balanced Score Card

Means of Visibility

- BurnDown Chart



Feedback through Retrospection

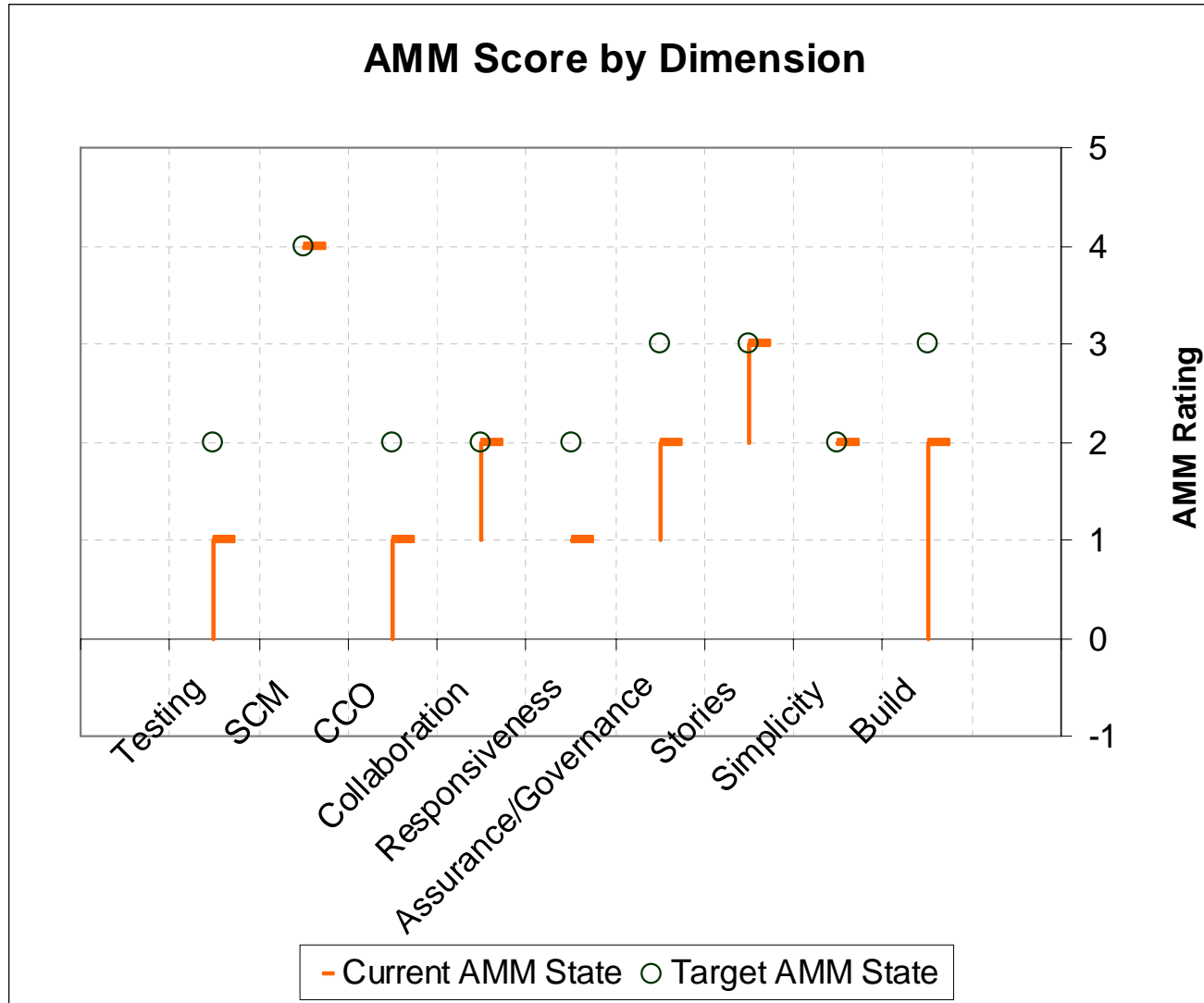
What Worked Well

Stuff To Try

What Still Puzzles Us

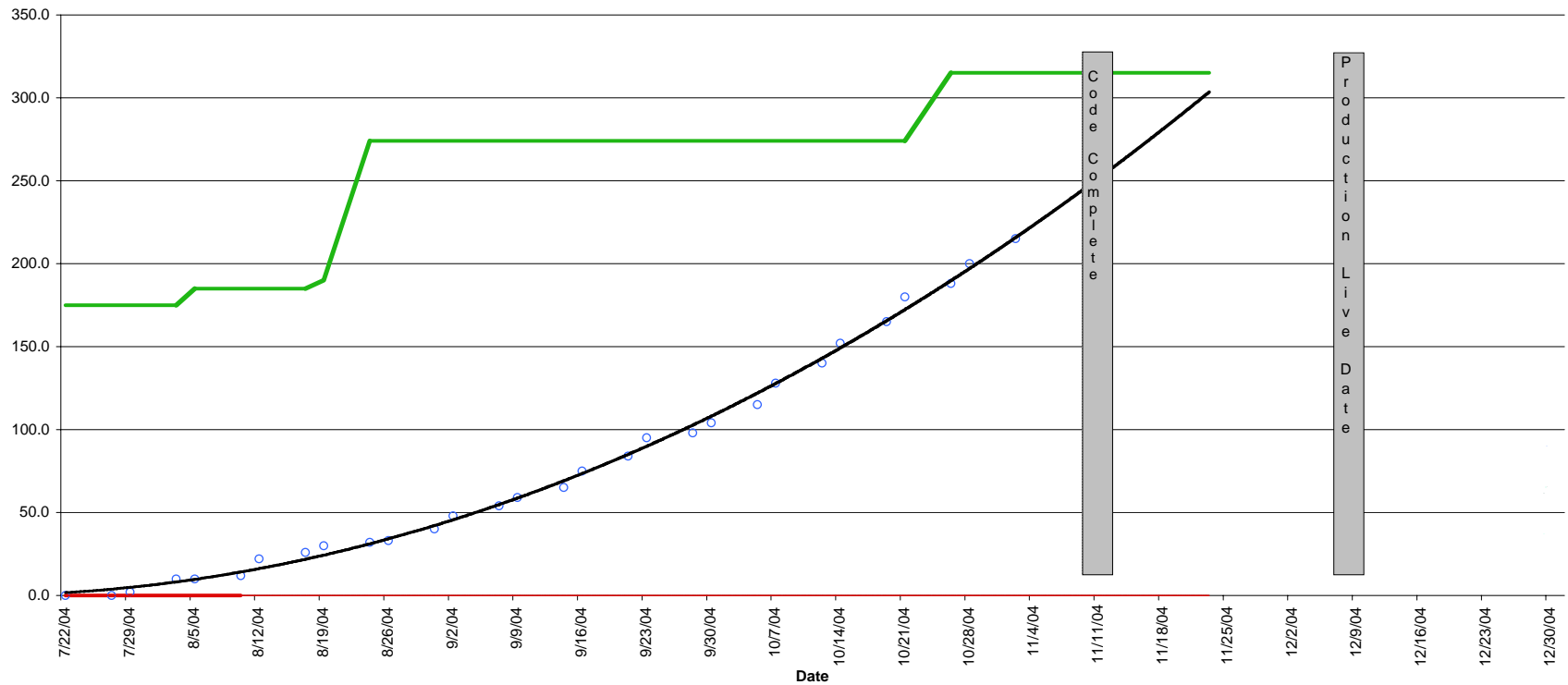
What Slowed Us Down

Team Maturity



Means of Visibility

- BurnUp Chart



Alignment by Balanced Scorecard

Value to Company

Efficiency of Delivery

Quality of Product

Agility in Process

So, how do we do Agile?

- We charter the project
- We fill roles (not positions)
- We establish guidelines and timeboxes
- We iterate upon the work at hand
- We validate what we have completed was what was desired
- We reflect on what we have done and decide if we could do better

And we change as appropriate

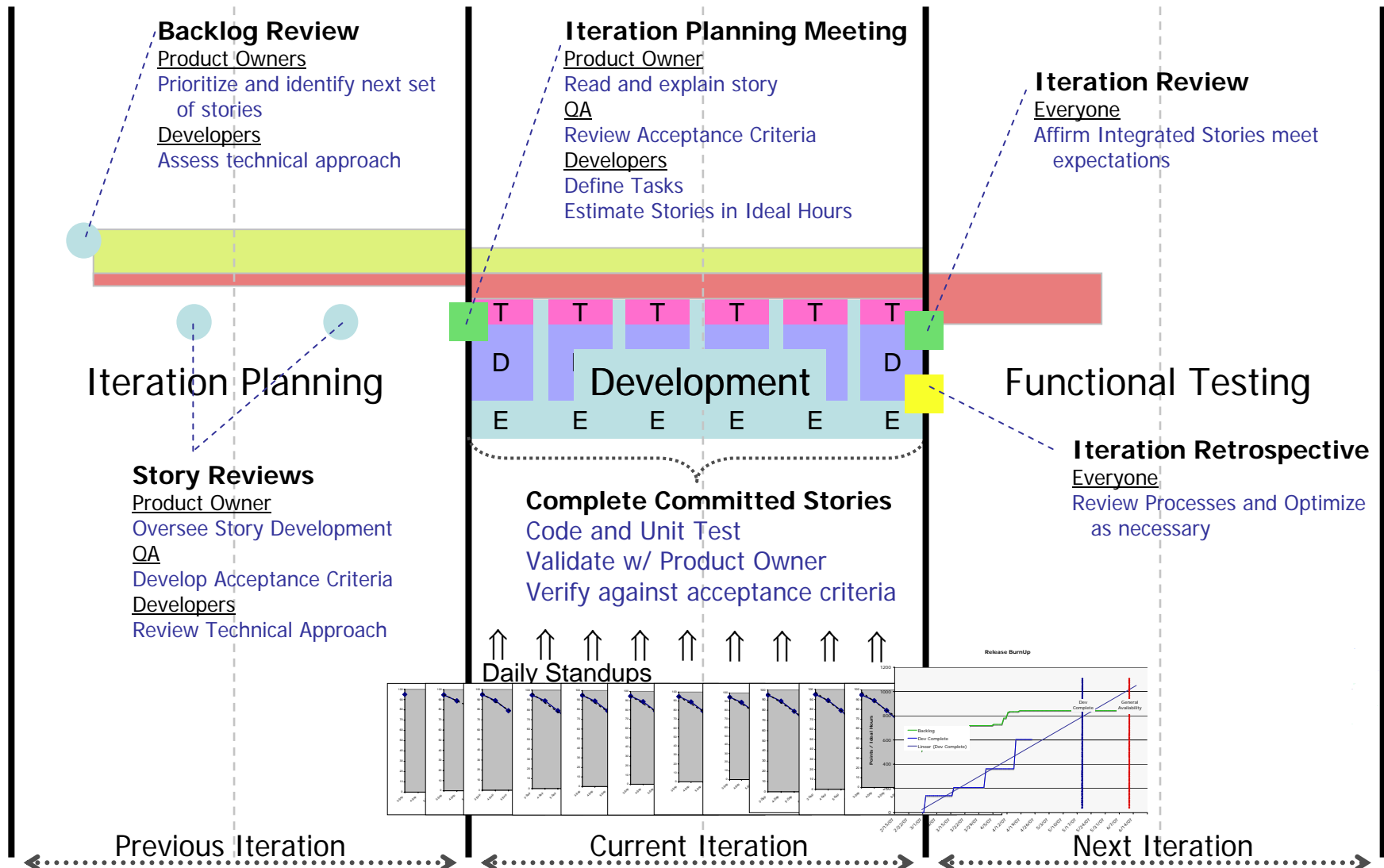
Rolling the Process out

- Agile is best understood experientially
 - Training is best through “games”
 - Learning by doing / Shadowing / Coaching and Mentoring
- Teams should learn together
- Training should be directly before project initiation
- First project should be a public example
 - Entire organization should be able to decide upon the methodology's expansion or contraction within the organization
 - Subjective and Objective Retrospective
 - Teams need to embrace the change or it will not work
 - Both top-down and bottom-up acceptance is necessary

Steps in an Agile Project

- Charter
 - Input is the Vision and the Value Proposition
 - Output is the Business Case including
 - High Level Backlog and Scope
 - Operational and Technical Vision
 - Risk Analysis and Plan
 - Release Schedule and Cost Model
 - Resource and Staffing Model
- Iteration 0
 - Write initial Stories
 - Setup Development Environment
 - Train on tools and practices

Inner Workings of an Iteration



Steps in an Agile Project

- Pre-Deployment Iteration
 - Culminating Rolling Regression Test
 - Quality Assessment
 - Business Decisions:
 - Prioritization of remaining defects
 - Assessment of Expectation Matching
 - Release / Release with Changes / Hold and Re-assess

Tailored Map of Agile Practices

