

Requirements Management in an Agile and Lean Environment

Systems and Software Technology Conference

Salt Lake City, UT

May 2011



Mr. George Haley

Director of Architecture, Design &
Analysis

Northrop Grumman Corp
El Segundo, CA

Dr. Suzette S. Johnson

Agile Engineering
Northrop Grumman Corp
Columbia, MD

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

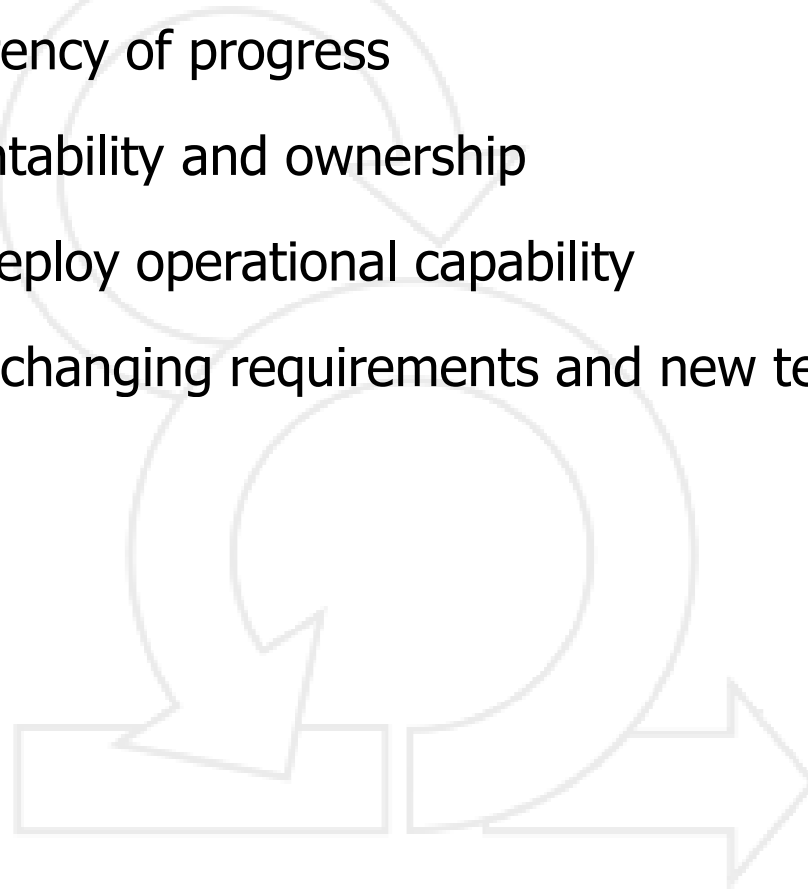
1. REPORT DATE MAY 2011	2. REPORT TYPE	3. DATES COVERED 00-00-2011 to 00-00-2011			
4. TITLE AND SUBTITLE Requirements Management in an Agile and Lean Environment		5a. CONTRACT NUMBER			
		5b. GRANT NUMBER			
		5c. PROGRAM ELEMENT NUMBER			
6. AUTHOR(S)		5d. PROJECT NUMBER			
		5e. TASK NUMBER			
		5f. WORK UNIT NUMBER			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Northrop Grumman Corp,9160 Guilford Road,Columbia,MD,21046-1803		8. PERFORMING ORGANIZATION REPORT NUMBER			
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)			
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)			
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES Presented at the 23rd Systems and Software Technology Conference (SSTC), 16-19 May 2011, Salt Lake City, UT. Sponsored in part by the USAF. U.S. Government or Federal Rights License					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 25	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

Discussion Outline

- Defining an Agile Environment
- Requirements, Use Cases, User Stories
- Levels Planning
- User Story Verification and Validation
- Summary
- References

An Agile Environment

- Adaptive, Responsive, Continuous Improvement, Evolving
- Improved transparency of progress
- End-to-end accountability and ownership
- Reduces time-to-deploy operational capability
- Ability to adapt to changing requirements and new technological advancements



Building Practice on Principles



Eliminate Waste

Build Quality In

Create Knowledge

Defer Commitment

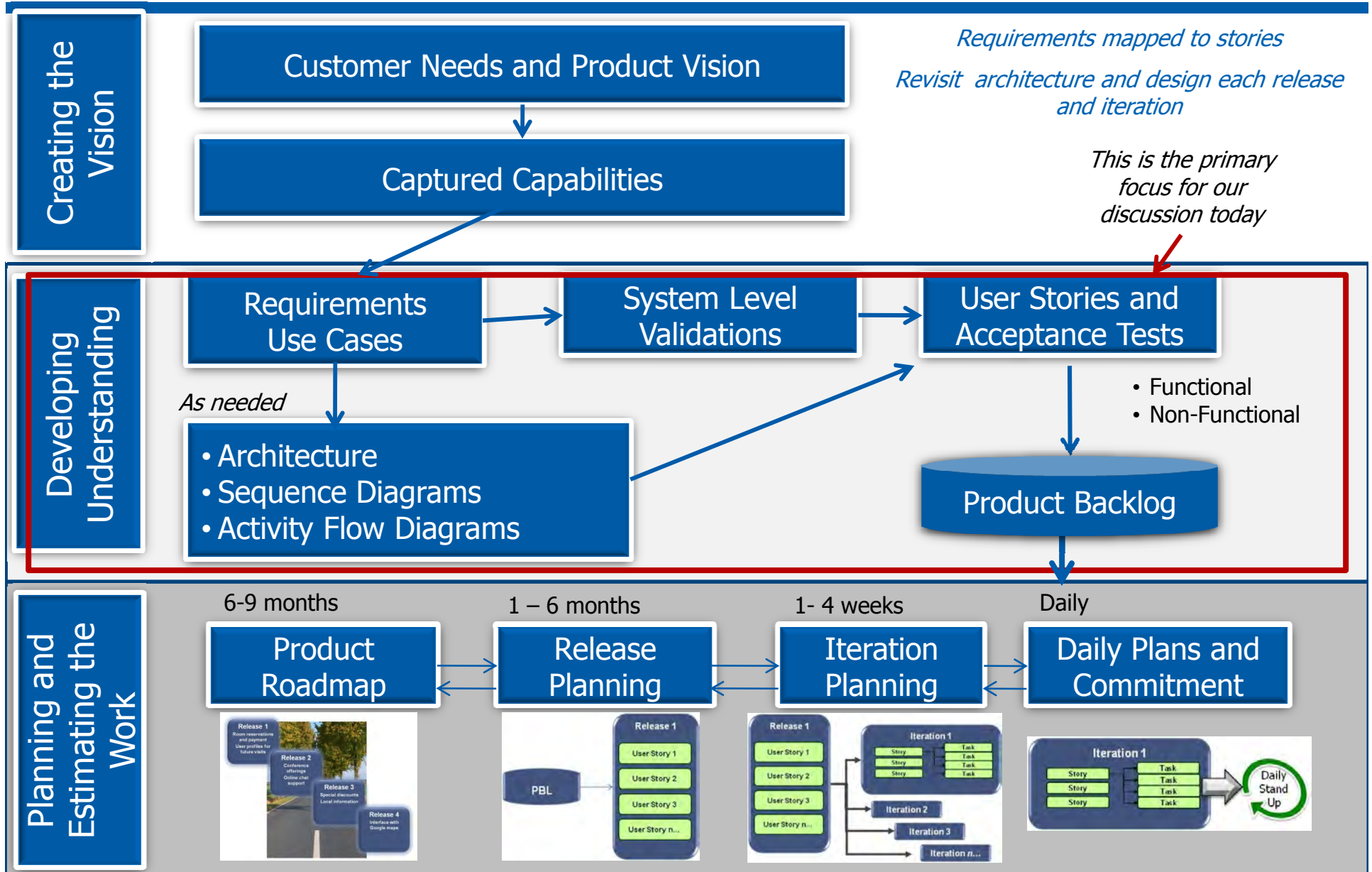
Deliver Fast

Respect People

Optimize the Whole

Adapted from: Implementing Lean Software Development: From Concept to Cash by Mary and Tom Poppendieck

Deliver Value



Use cases

- “A **use case** is the specification of a set of actions performed by a system, which yields an observable result that is, typically, of value for one or more actors or other stakeholders of the system.... Use cases are a means for specifying required usages of a system. Typically, they are used to capture the requirements of a system, that is, what a system is supposed to do”¹.
- Agile methods emerged with a focus on user stories. User stories are similar to use cases but are:
 - Typically more fine-grained & smaller;
 - Not intended to specify requirements;
 - More closely related to schedulable work.

Do use cases have a place in Agile environments?

¹OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2

Use Cases and User Stories

Why?

- System behavior is described operationally from users' perspectives
 - Greatly reduces validation issues
- Drives verification to focus on operationally-relevant cases

How?

- In agile environments use cases are written “just-in-time” (for release planning) versus all up front
- Using use cases in an Agile environment. Ask:
 - How much do we need to write at this time?
 - When do we need to write more?
 - What is the fastest way to write/convey them?
 - Who benefits from more information or more detail?¹

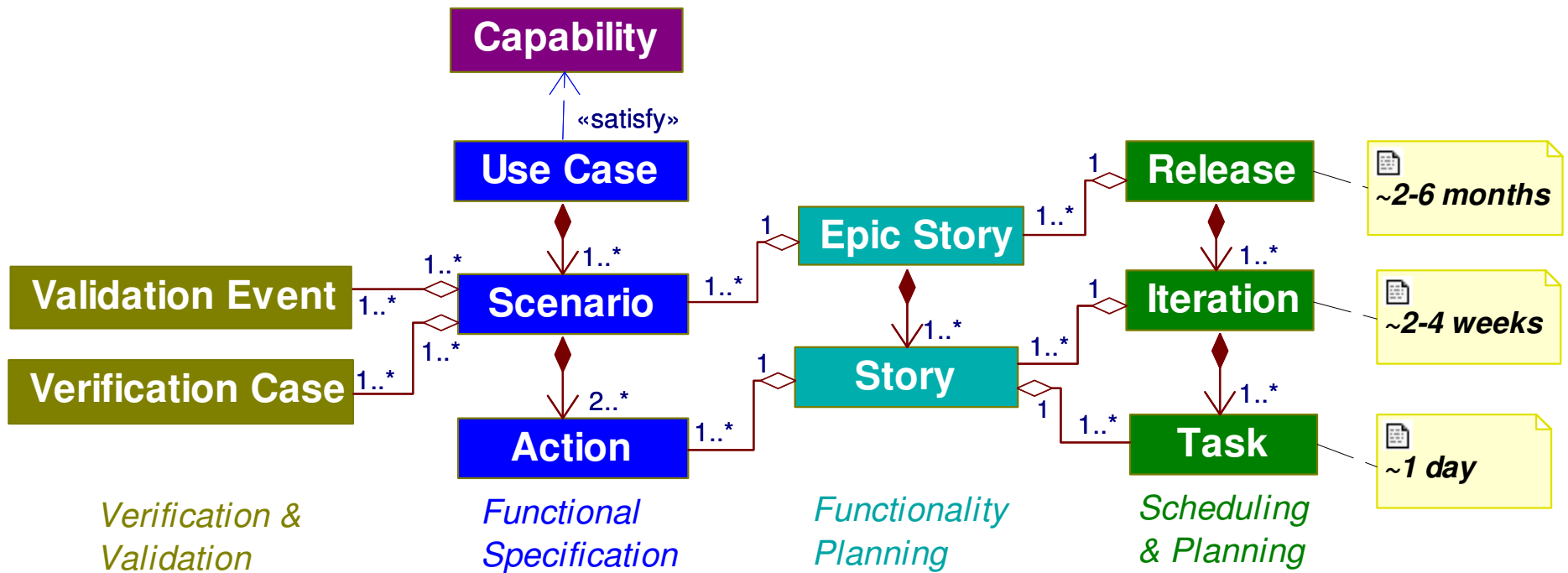
Both approaches can coexist

(2003). Allistair Cockburn. Agile Use Cases (Presentation).

Use Case/User Story Definition

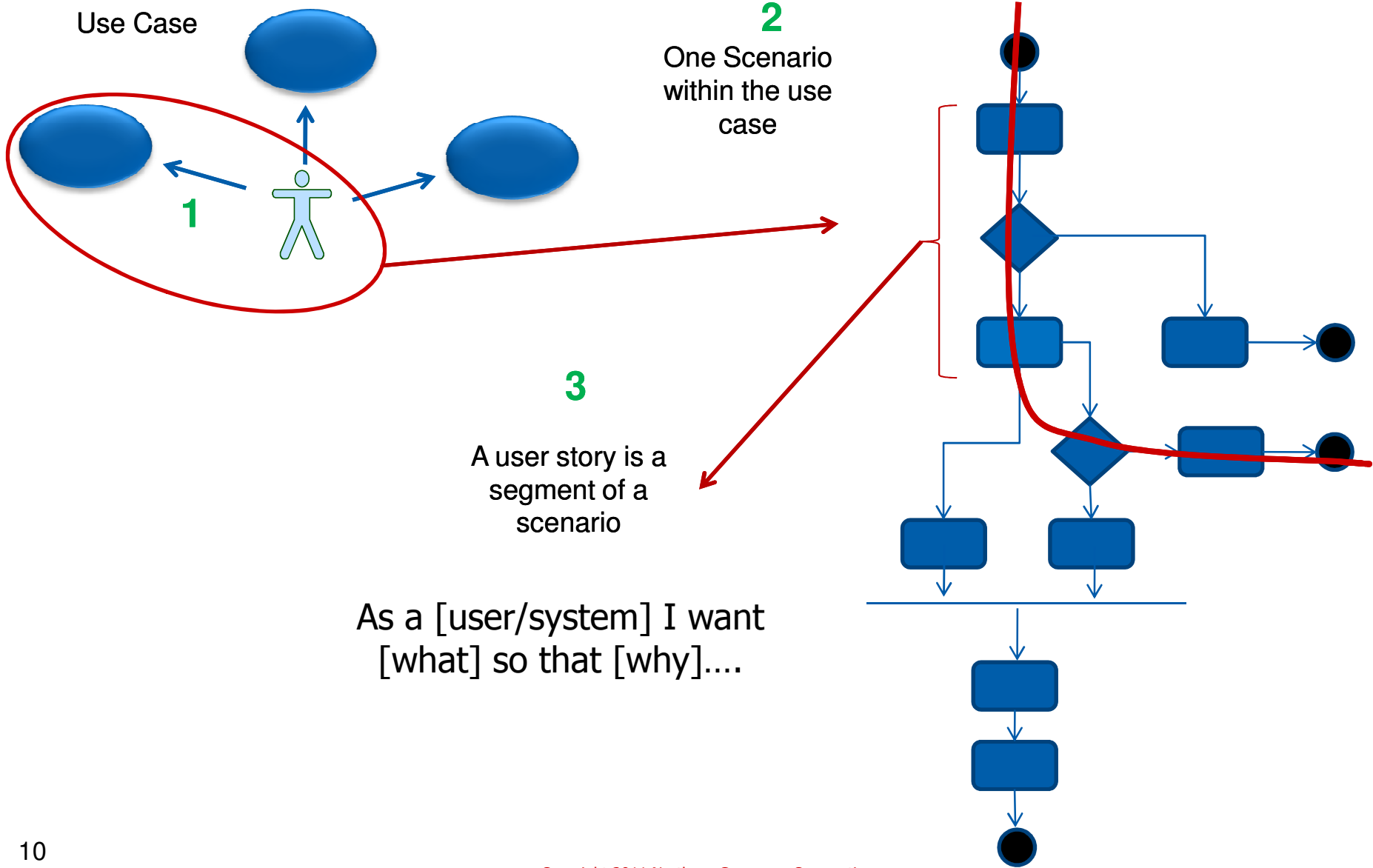
- Written from the user perspective
- Captures value to the customer/user (not the developer)
- Emphasizes verbal *communication and collaboration*
- ***The right size for estimating and planning*** (User Stories only)
- Testable
- Demonstrable

Agile Systems Engineering Ontology



- "Action" is a.k.a. "Step"
- "Scenario" is a.k.a. "Flow"

Use Case to Scenario to User Story



User Stories Convey Meaning

Example of traditional approach shortcomings

IEEE 830 Software Req. Spec

- 1.The product shall have a gas engine.
- 2.The product shall have four wheels.
The product shall have a rubber tire mounted to each wheel
- 3.The product shall have a steering wheel.
- 4.The product shall have a steel body.

Reference: Mike Cohn, mountaingoatsoftware.com

Source: Adapted from *The Inmates are Running the Asylum* by Alan Cooper. (1999)

User Stories Convey Meaning

As a <lawn service provider> I want to mow lawns quickly and easily.

As a <lawn service provider> I want to sit comfortably while mowing lawns.



Capabilities to User Stories

The system shall provide the capability for making hotel reservations.

As a premiere member, I want to search for available discounted rooms.

As vacationer, I want to search for available rooms.

As vacationer, I want to save my selections.

What About Non-Functional Requirements?

As a vacationer and user of the hotel website, I want the system to be available 99.99% of the time...

As vacationer, I want web pages to download in <4 seconds...

Stories for non-functional requirements

As the hotel website owner, I want 10,000 concurrent users to be able to access the site at the same time with no impact to performance...

Describes system behavior or characteristics

Advantages and Practices

- Why User Stories are preferred over traditional methods
 - Emphasizes verbal communications
 - Comprehensible by both customer and developer
 - The right size for planning
 - Works well for iterative development
 - Encourages deferring detail until you have the best understanding you are going to have about what you really need.
 - Helps the Team understand to whom they are delivering certain functionality
 - Helps the Team understand when they are “done”
- User Stories can be used with traditional requirements
 - Best to keep the requirements high level
 - A mapping from requirements to users stories needs to be maintained, especially if requirements are part of the contract.

Reference: Mike Cohn, MountainGoatSoftware.com

User Stories

- A story is either “done” or “not done”.
- As stories are completed, the status of the high-level capability is updated (verified, partial...).
- A set of tests is linked to each story and the high level requirement.
- Each story has a set of test objectives and automated tests.
- Stories are for communication and to better understand the work.



**Focused on the
Conversation**

Release Plan, Iteration Plan, Daily Plan

Example: Hotel Website

Capability 1: Make Room Reservations

Release Plan (Stories)

Relative Value	User Stories	Points
80	As a vacationer, I want to search room availability...	Tests: <ul style="list-style-type: none"> • Test with search on 1 room • Test with search on executive suite....
75	As a vacationer, I want to save my request...	Test Objective
70	As a vacationer, I want to pay with a credit card...	Test Objective

Iteration Plan (Tasks)

Hours	
Design Review	4
Install Baseline	4
ICD Updates	8
Acquire Test Data	8
Code	24
Develop Tests	8
Run Tests	8

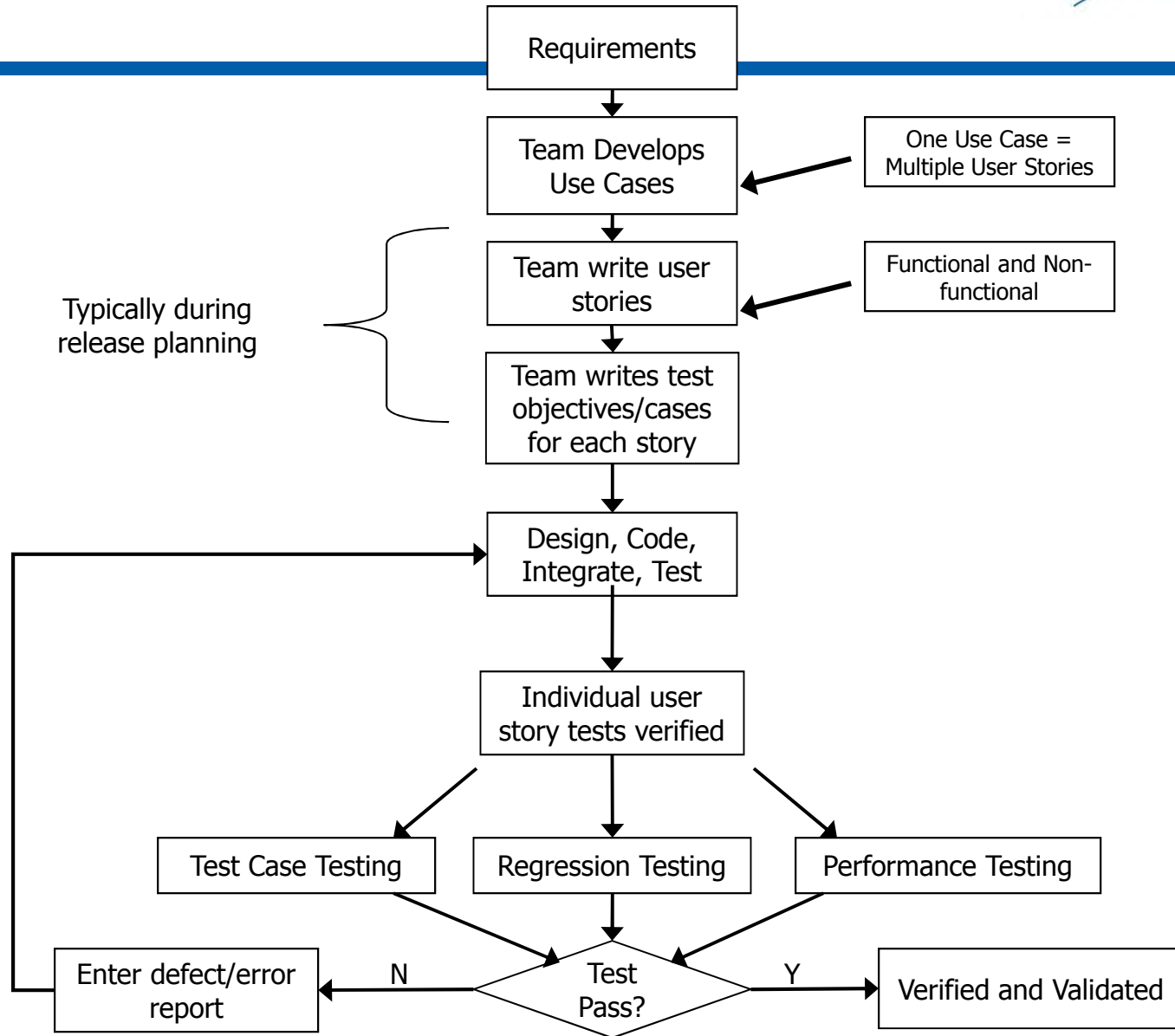
The Daily Plan

Yesterday I started on the interface....
 Today I plan to...
 The one thing standing in my way...

Product Backlog

- The high-level requirements with stories mapped
- Prioritized by the product owner in terms of business value and risk
- Reprioritized at the start of each iteration

User Stories and Testing



The Iteration Demonstration and Acceptance

- Transparency and information sharing
- Team presents what it accomplished during the iteration
- Typically takes the form of a demo of new features or underlying architecture
- Time-boxed
- Whole team participates
- Feedback from stakeholders and users
- User Stories validated and accepted



*User Story
Validation*

Requirements Mapping

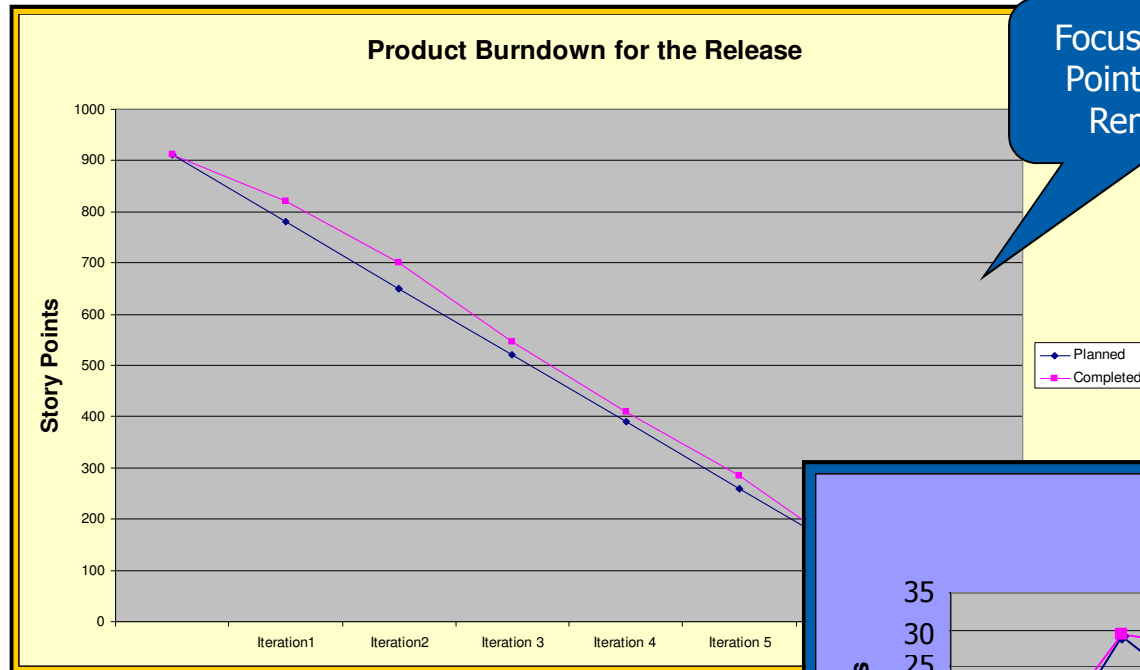
	A	B	C	D	E	F	G	N	O	Z	AA
1	Project Requirements			Stories							
	SS ID	Text	Verification Summary	Iteration 1	Iteration 2	FD 0032 (U) As a user I want to browse online help	FD 0077 (U) As a user I want to display Metadata Result on 2D Map	Iteration 3	FD 0005 (U) As an administrator I want to monitor system health and status and derive Availability	Iteration 4	(U) The site for location information
2	SS-19713	(U) The SYSTEM shall provide a detailed online help capability that is context sensitive.	Verified			X					
3	SS-19714	(U) The SYSTEM shall provide an environment to support the online help capability from content providers.	Verified			X					
4											

- Requirement to story mapping

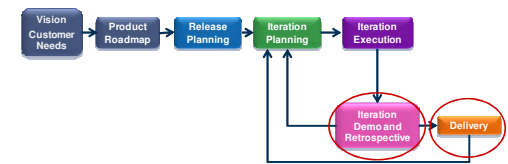
- Requirement to Story to Test to Verification

Requirement				
ID	Text	Spec Paragraph	Test	Verification Method
SS-19707	(U) The SYSTEM shall collect and maintain metrics on the number of users logged in on the system: Total number, average daily, max/min simultaneously logged on, daily totals of users logged in by organisation.	3.1.1.6, (U) Infrastructure	Test Objectives	Analysis
SS-19709	(U) The SYSTEM shall collect and maintain metrics on the number, size and type of queries successfully and non-successfully executed by the system: Number, size and type of queries successfully and non-successfully executed; totals by day/month; average daily numbers; max/min number, max number simultaneously run.	3.1.1.6, (U) Infrastructure	Test Objectives	Analysis

Monitoring Progress: Product Burndown

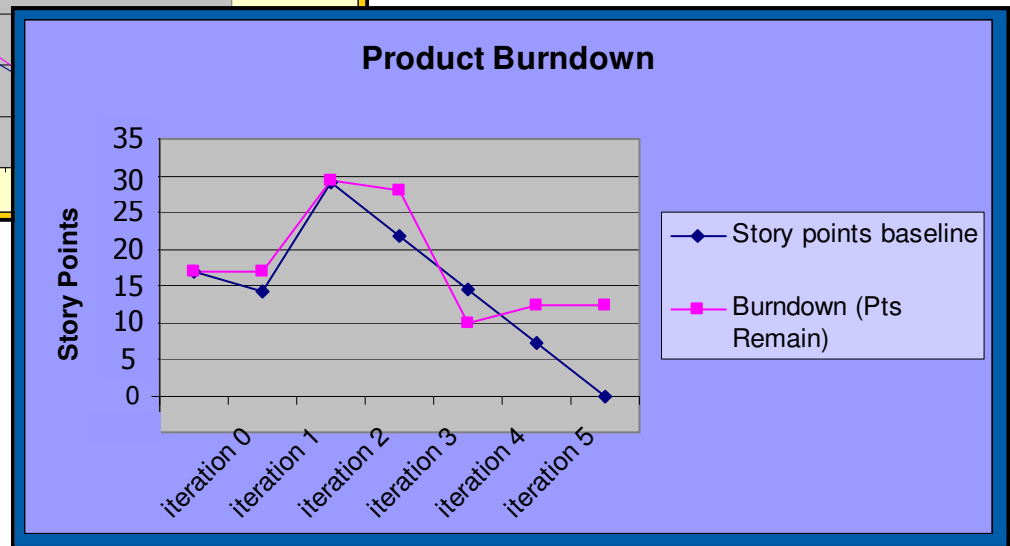


Focuses on the Points (Work) Remaining



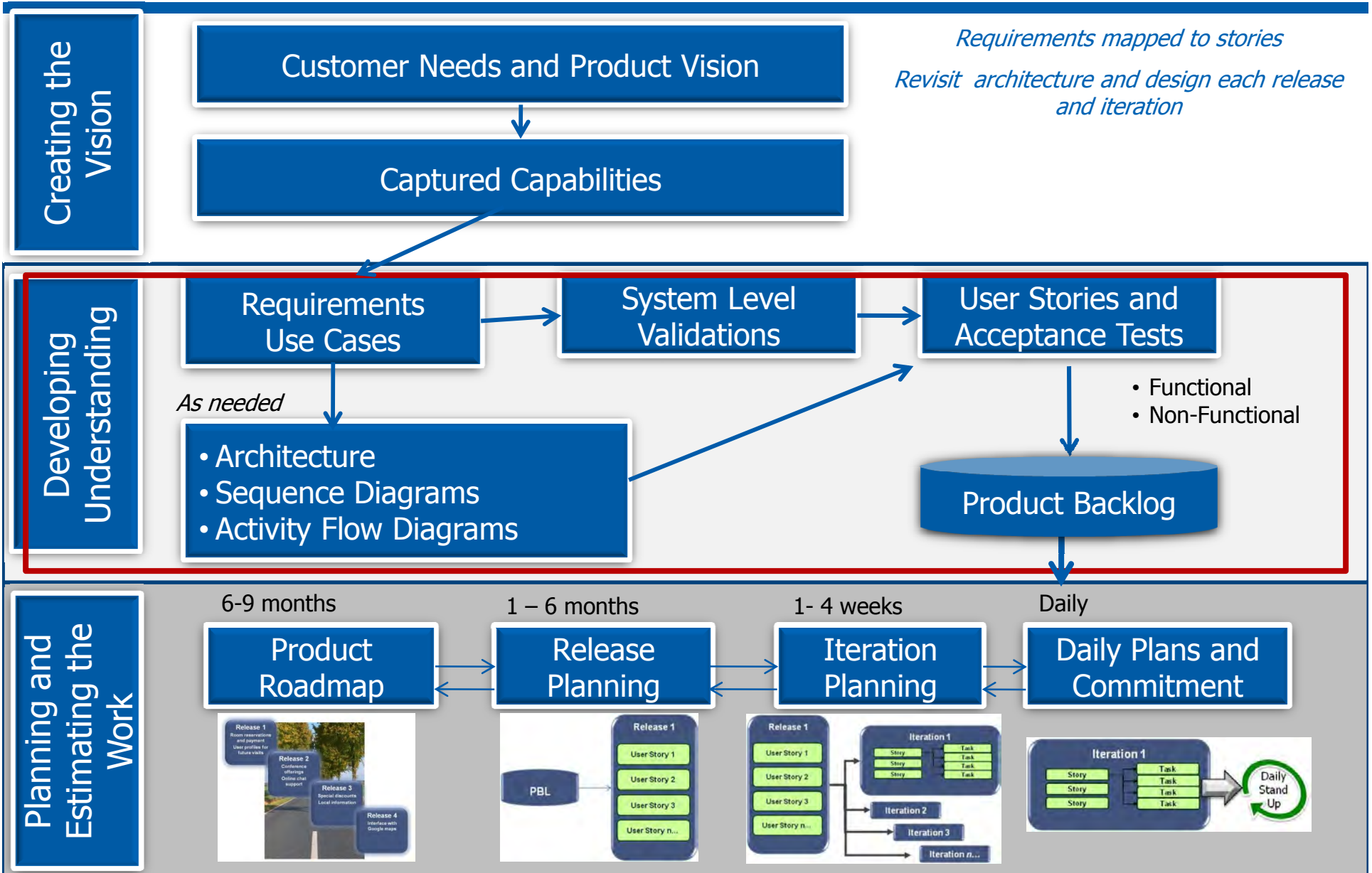
A project team's burndown (team of teams)

- Based on story points planned
- Updated and reviewed each iteration
- As stories are accepted and tests passed, requirement progress is updated



A team's product burndown

Summary: Bringing It Together



Final Notes

- Requirements and User Stories
 - High level requirements can be mapped to user stories
 - User stories convey understanding (user, need, why)
 - User stories create the Product Backlog
- Requirements Analysis and Design
 - Upfront requirement analysis is done during release planning for the capabilities being delivered in that release
 - Designs are developed/built upon each iteration
 - Design reviews for user stories are part of the story's tasks and are done as needed
- Requirements Priorities and Changes
 - Requirements and user stories may be reprioritized
 - Contract modifications may be needed, but would not be done every iteration
- Requirements and Tests
 - High level requirements (end-to-end capabilities) have tests and each user story has tests.
- People
 - Systems engineers part of the team
 - Those responsible for performing the end-to-end capabilities testing should be part of team planning and regular collaboration

References and Recommended Readings



<i>Agile Requirements and Collaboration</i>	
Requirements by Collaboration	Ellen Gottesdiener, EBG Consulting
Collaboration Explained	Jean Tabaka, Rally Software
User Stories Applied	Mike Cohn
<i>Agile Development Practices</i>	
Agile Software Requirements	Dean Leffingwell
Agile Software Development with Scrum	Ken Schwaber and Mike Beedle
Agile Testing	Lisa Crispin and Janet Gregory
Agile Estimating and Planning	Mike Cohn

NORTHROP GRUMMAN

