



Test Early with Agile ATDD

*Combining Agile Processes with T&E
Acceptance Test Driven Development*

Joshua Strain, Systems Engineer

Agenda

- What is Agile ATDD?
- Bridging the “V” Gap
- Collaborative Acceptance Criteria
- Instilling Minimalism into Design
- Learning Through Example



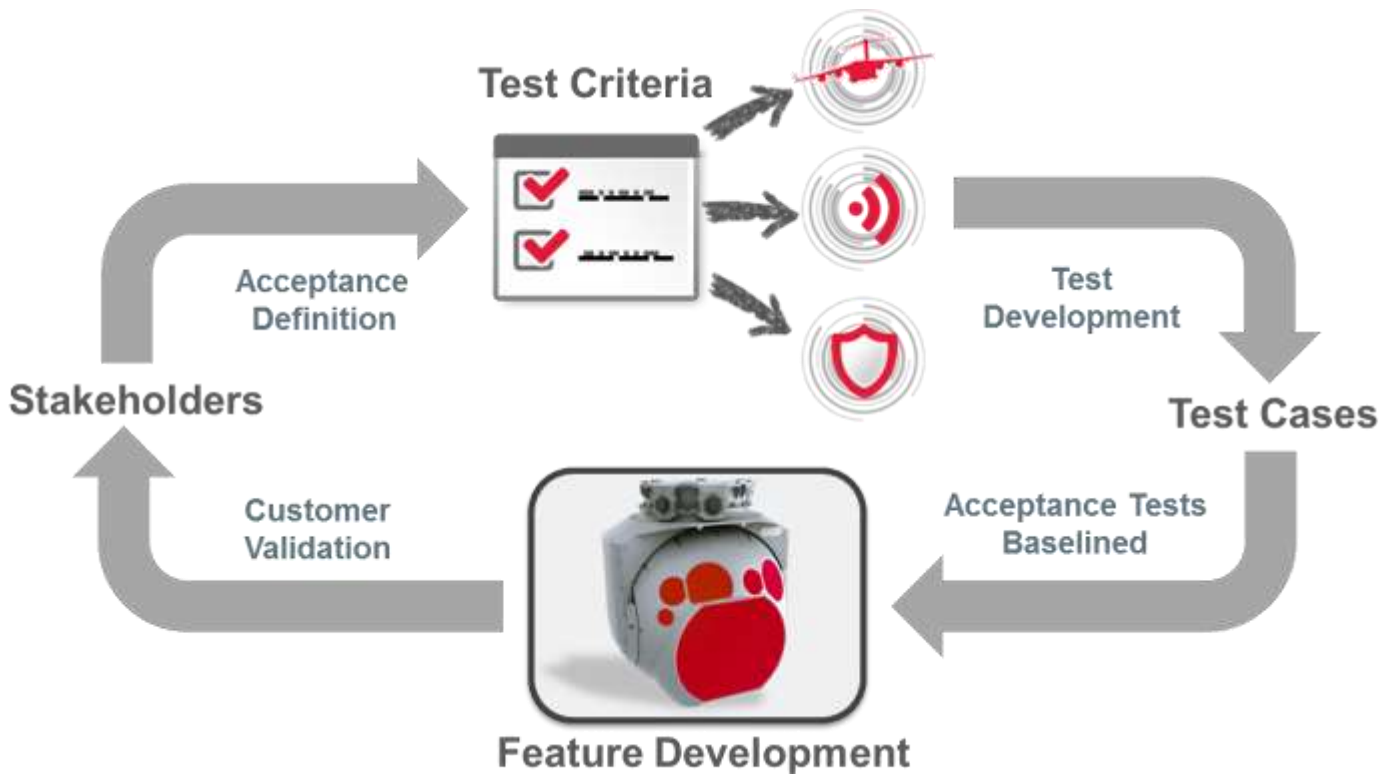
Josh Strain has been with Raytheon SAS for 7 years. He holds a B.S. in Computer Engineering from the University of Oklahoma and a M.S. in Systems Engineering from Johns Hopkins University. He is a Certified Agile Product Owner and currently works across the Multi-spectral Targeting System (MTS) product lines in McKinney, TX within the Systems Development Center.

Joshua.Strain@Raytheon.com - 972.952.7173



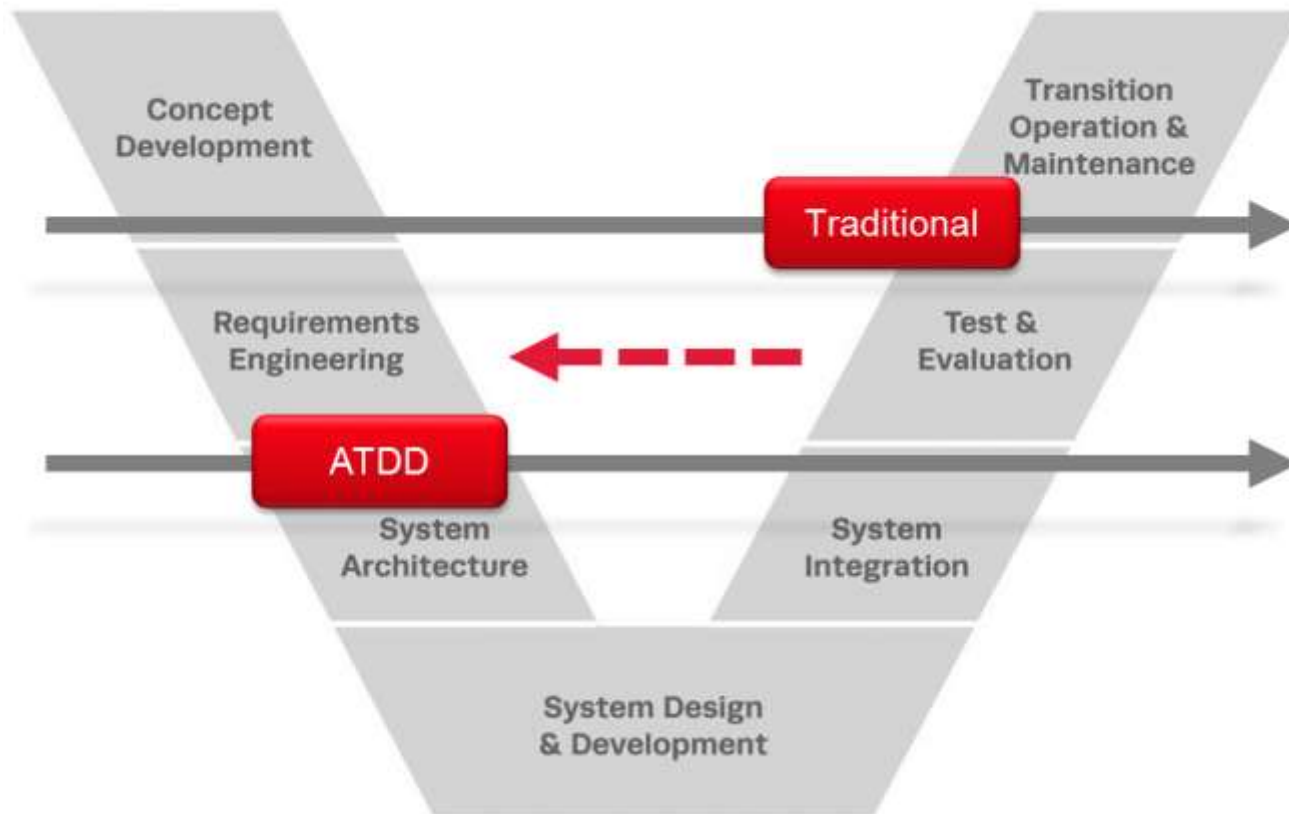
What is Agile ATDD?

- Acceptance criteria defined ***prior*** to design
- Agile process promotes incremental feature testing
- Customer focused collaboration drives test development



ATDD Bridges the “V” Gap

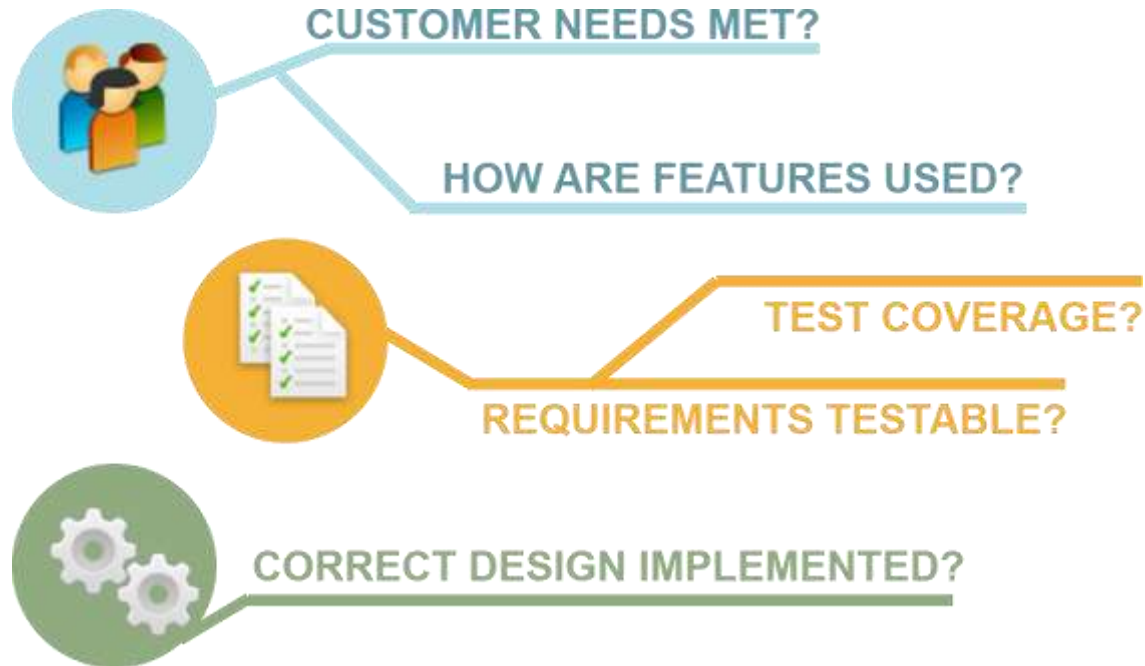
- Early test development identifies requirement issues
- Architecture, concept, and test gaps realized early in life cycle



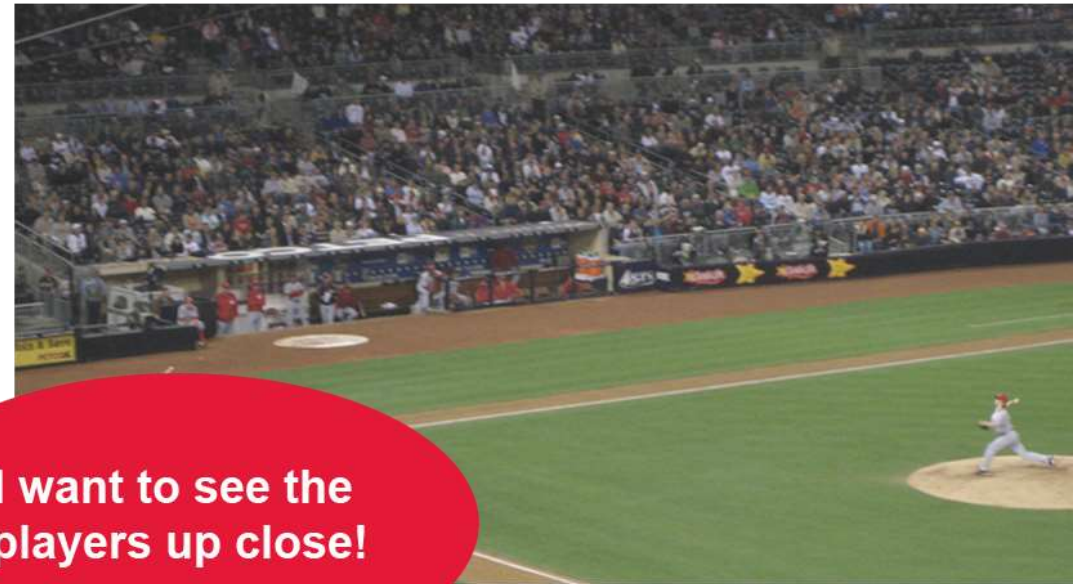
Collaborative Acceptance Criteria

- Improves product quality through better acceptance understanding
- Stakeholder input important to successful test validation

Engineers
Customer
Designers
T&E
Users
Systems
Personnel



Instilling Minimalism into Design



I want to see the
players up close!



Instilling Minimalism into Design



Instilling Minimalism into Design

ATDD GOALS:

- Develop just enough functionality to meet acceptance criteria
- Eliminate waste and unnecessary product features



Example of Agile ATDD

Customer Requirement

- The system shall track a minimum of 10 targets concurrently.

Understand the User Story

- As the system operator, I want to track multiple targets so that I can easily adapt the mission under stressful conditions such as being swarmed by boats at sea.

Define the Test Criteria Prior to Design

- Given a set of 10 moving boats when the system is over water, the system will maintain track on all targets within line of sight and field of view of the system



Example of Agile ATDD

Create Test Cases

- Simulated target movement patterns
- Target goes off line of sight
- Targets directly cross paths
- Operational sea testing with real swarm scenario

Define Acceptance Prior to Starting Design

- Collaboration reveals questions on target visibility with water spray from targets
- Customer feedback loop in place to define acceptable visibility requirement and additional test cases



Example of Agile ATDD

Design criteria now well understood!

What are some things designers might have missed from just looking at original spec?



Summary / Questions

- ATDD shifts test design prior to system design helping early validation of feature understanding
- ATDD collaboration lowers life cycle risk of rework
- ATDD minimizes design complexity by implementing just enough functionality to pass acceptance criteria

Questions?

