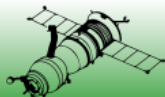# Embedding Agile Practices Within a Plan-driven Hierarchical Project Lifecycle

Dan Johnson

ScrumMaster, PMP, CSM

Nick Lombardo

Principal Project Manager, CSEP

**Pacific Northwest National Laboratory**

Presentation Outline

- Situation
  - Project description
- Need
  - Flexible development framework
- Solution
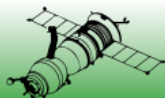 - Agile engineering
- In retrospect

Situation: Project Description

- Plan in place
  - Sensor-based correlation software architecture
- Project objective
  - Integrate disparate sensors
  - Perform real time risk assessment
- PNNL-developed product
  - Part of a larger system-of-systems
  - Plan-driven, requirements-based methodology
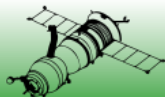 - Demonstrate end-to-end capabilities
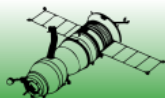
Situation: Requirements

- Defined user needs and capability gaps
- Developed operational requirements document
- Completed an analysis of alternatives
- Determined initial architectural solution
- Developed performance specifications
- Established requirements traceability

Situation: Project Risks

- First of a kind system
  - Elements developed concurrently
- Uncertain sensor performance
- Complex interfaces
  - Humans-in-the-loop
- Large SoS integrated with another SoS

- Multiple stakeholders/ developers
- Third party verification and validation
- Hierarchical engineering framework
- Cost and schedule

Need: Flexible Framework

- Different system integration process

- Inspect and adapt

- Evolve requirements
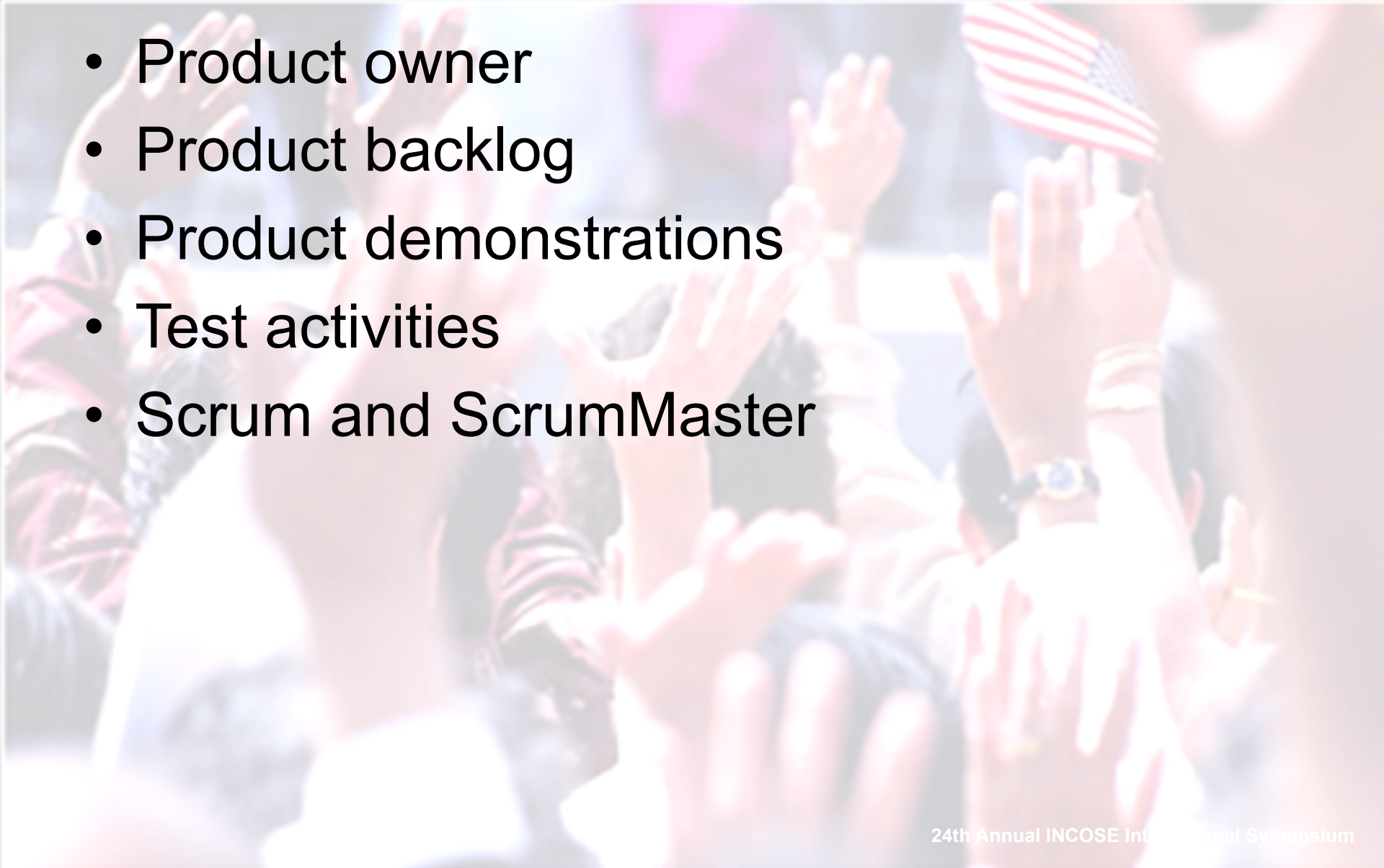
- Embed Agile practices
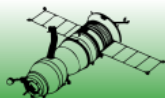
Solution: Agile Engineering

- Agile background

- Agile mindset
  - Emphasis on working systems
  - Interact frequently
  - Collaborate with customers
  - Respond to and welcome change

Beck K, M Beedle, A van Bennekum, A Cockburn, W Cunningham, M Fowler, J Grenning, J Highsmith, A Hunt, R Jeffries, J Kern, B Marick, RC Martin, S Mellor, K Schwaber, J Sutherland, and D Thomas. 2001. *Manifesto for Agile Software Development*. Accessed June 24, 2014 at www.agilemanifesto.org (last updated 2001).
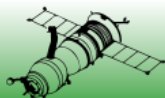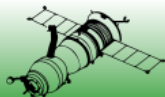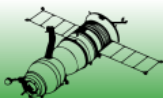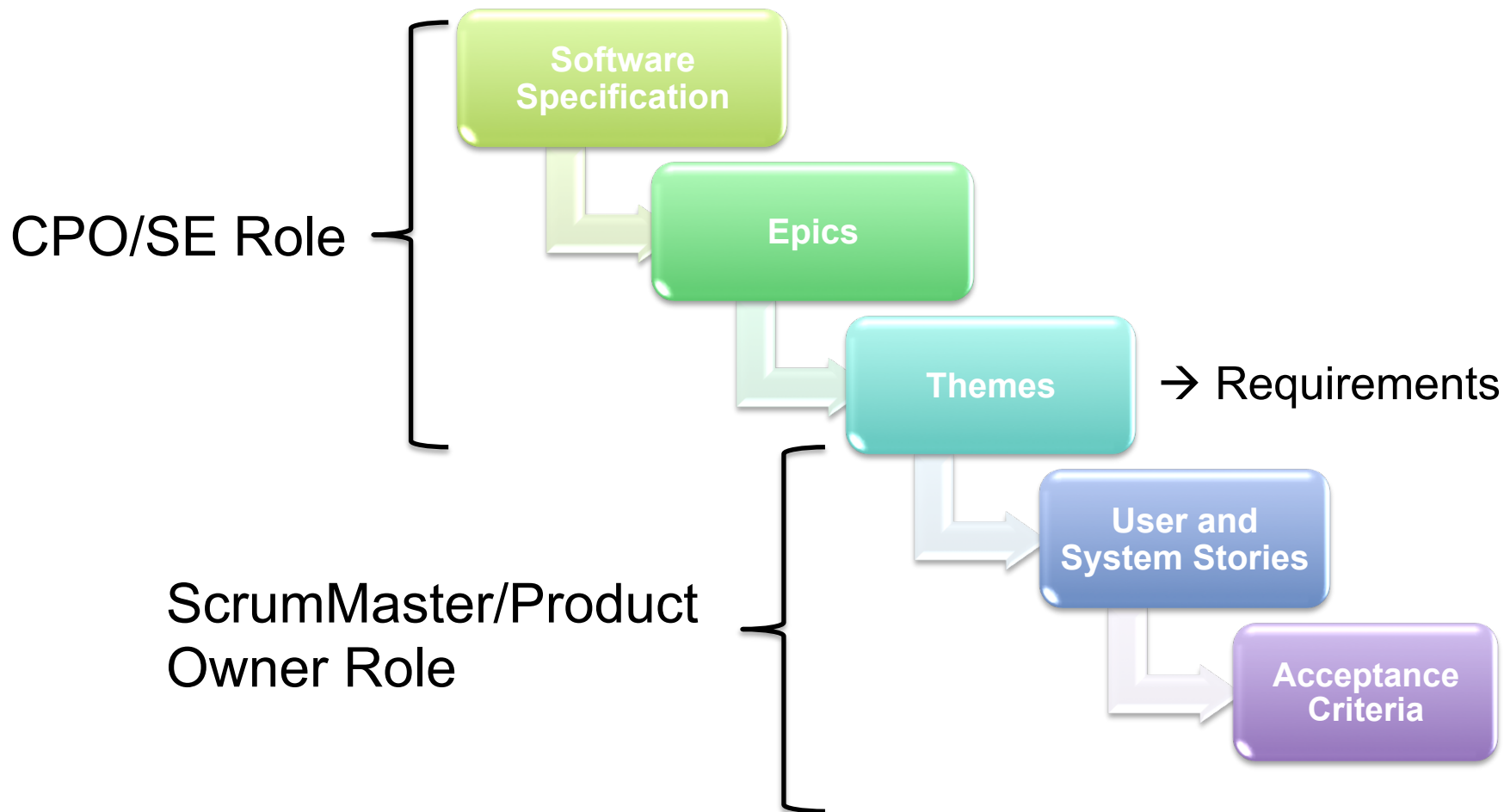
Key Roles and Agile Practices

- Product owner
- Product backlog
- Product demonstrations
- Test activities
- Scrum and ScrumMaster
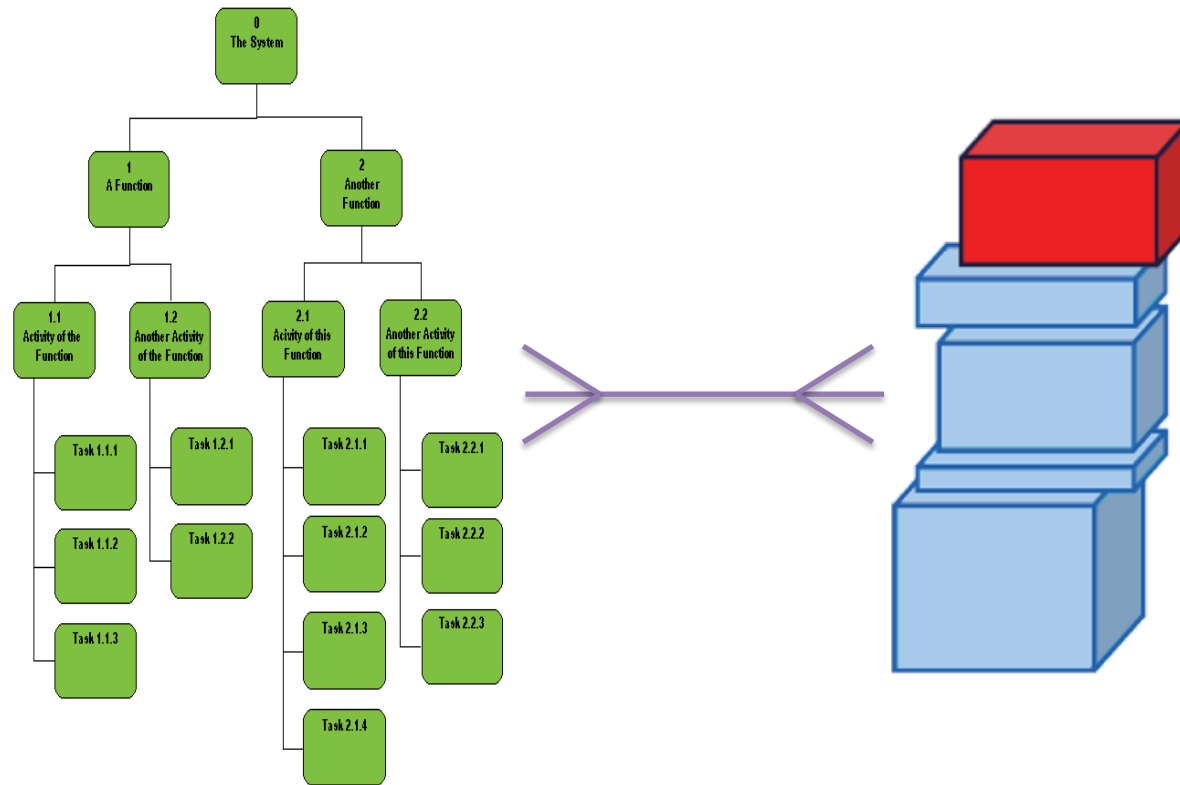
Product Ownership

- Establish a chief product owner
  - Has a vision of entire system
- Establish product owners for major subsystems
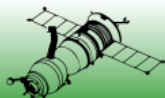 - Have a vision of the subsystems within the system

Product Backlog

CPO/SE Role

Software Specification

Epics

Themes → Requirements

ScrumMaster/Product Owner Role

User and System Stories

Acceptance Criteria

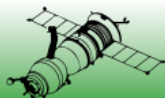# Functional Elements vs. Product Backlog Items

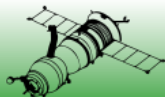**Functional Elements**   **Product Backlog Items**

Product Demonstrations

- Establish priorities during backlog grooming

- Develop acceptance criteria from requirements set

- Conduct demonstrations

- Accept and verify new features during demonstration

Test & Demonstration Activities

- Develop a test plan
- Create and maintain a requirements traceability matrix
- Create and maintain a requirements verification matrix
- Collaborate regularly with the client
 - Successfully transition to the IV&V team

Scrum and ScrumMaster

- Agile project management framework
- Scrum activities
    - Develop product backlog
    - Groom product backlog
    - Plan sprints
    - Conduct daily planning session
    - Conduct sprint closeout and retrospective
- ScrumMasters for this project
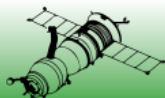
Embedded Agile Practices
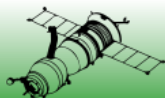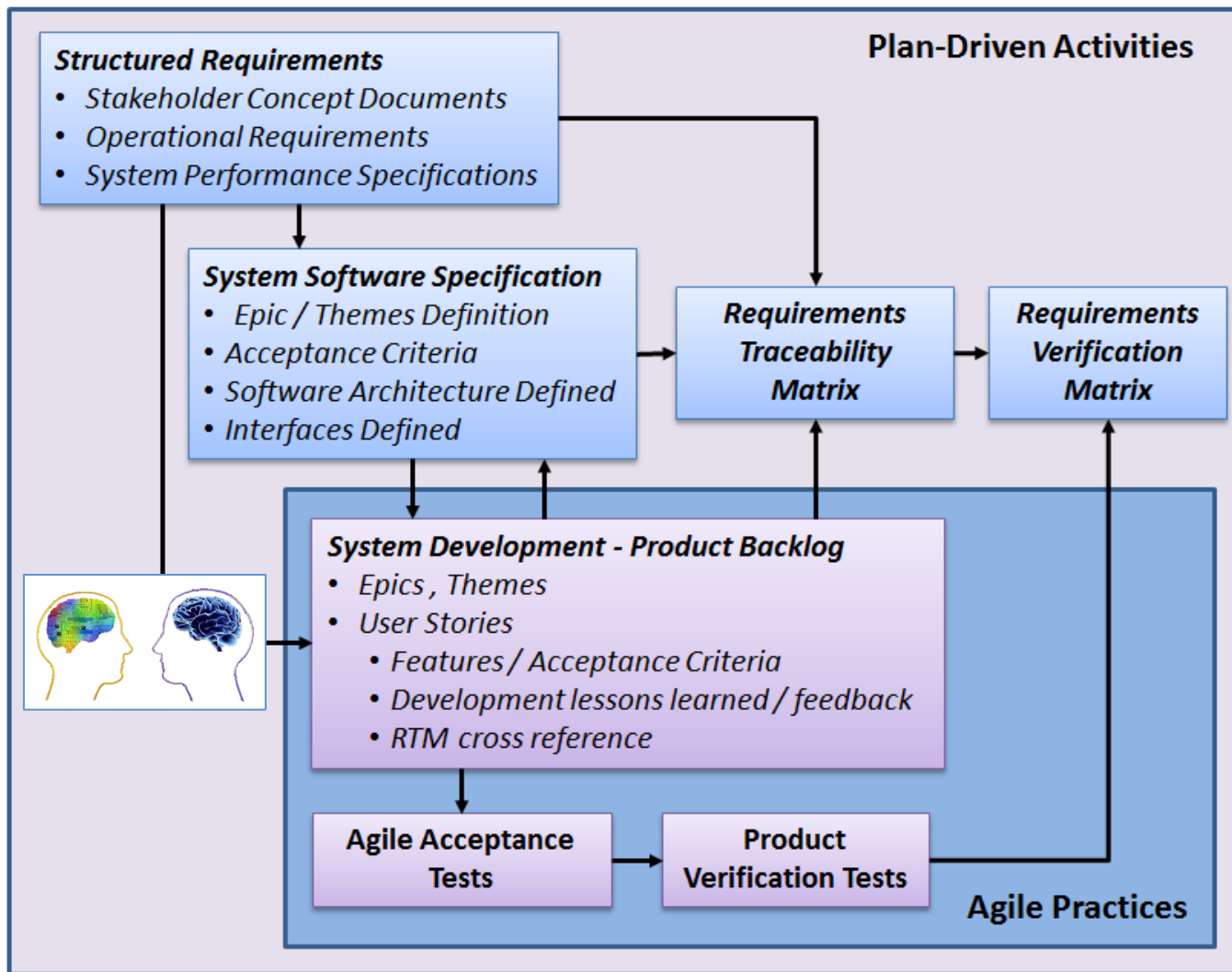
Embedded in Iterative Incremental Development Activities

Incremental/Linear and Evolutionary Development
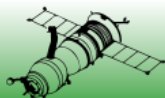Single or Multiple Deliveries
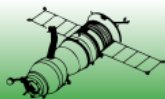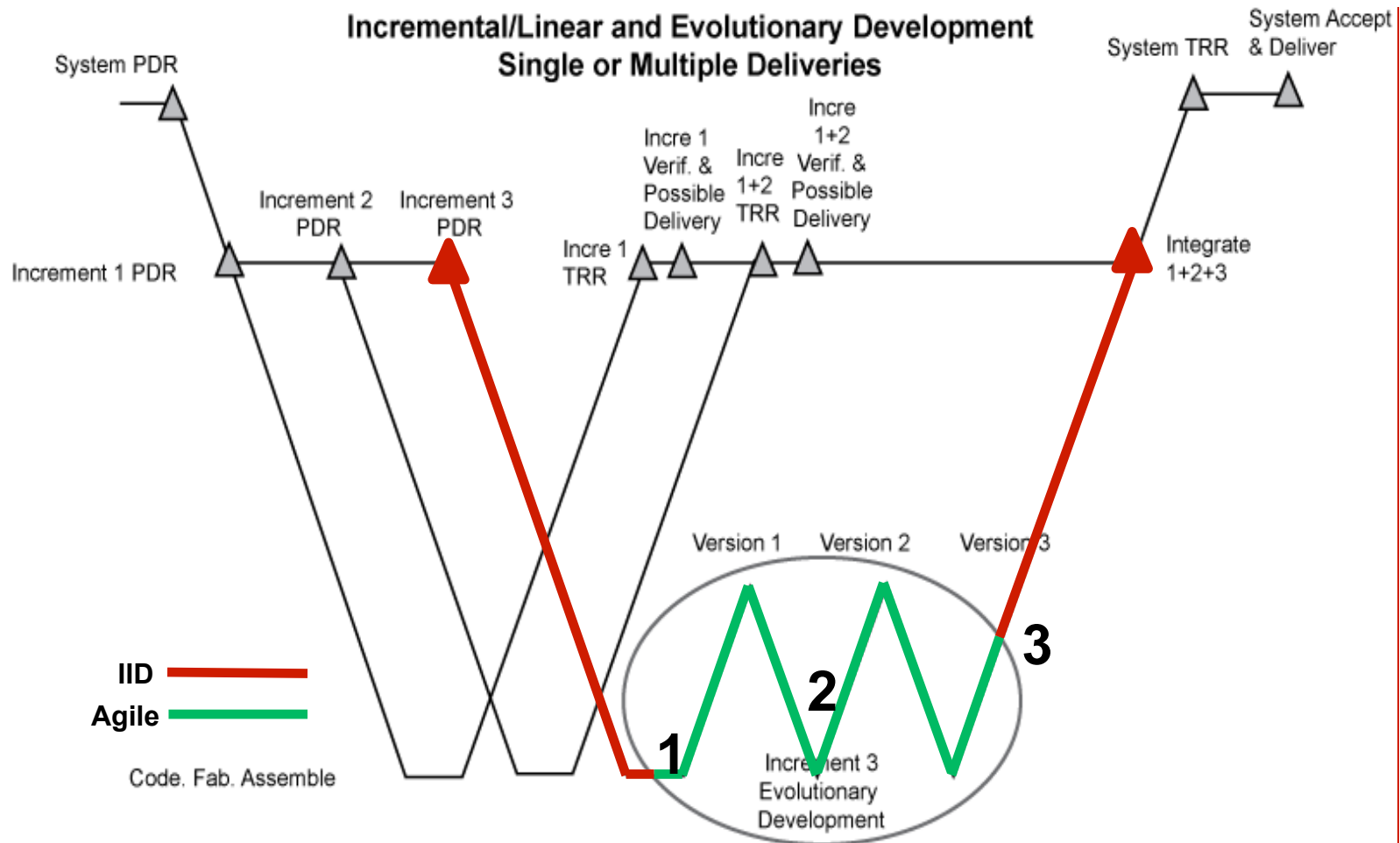
In Retrospect – Agile Adoption

- Adopting Agile
  - PM acceptance
  - Creating an Agile team
  - Translation between paradigms
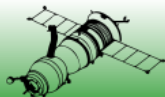 - Designing a repeatable (and successful) process

In Retrospect – Team Maturity

- Agile team maturity
  - ScrumMaster as coach
  - Scrum <> Agile
  - Owning Agile mindset

In Retrospect - Recipe for Success

- Chief Product Owner
- Requirements as acceptance criteria
- Requirements Traceability Matrix
  – resolves many-to-many relationships
- End-to-end product demonstrations
- Delivered on time

Questions and Conversations