Aligning the Language of Systems Engineering and Agile

Kelly Weyrauch

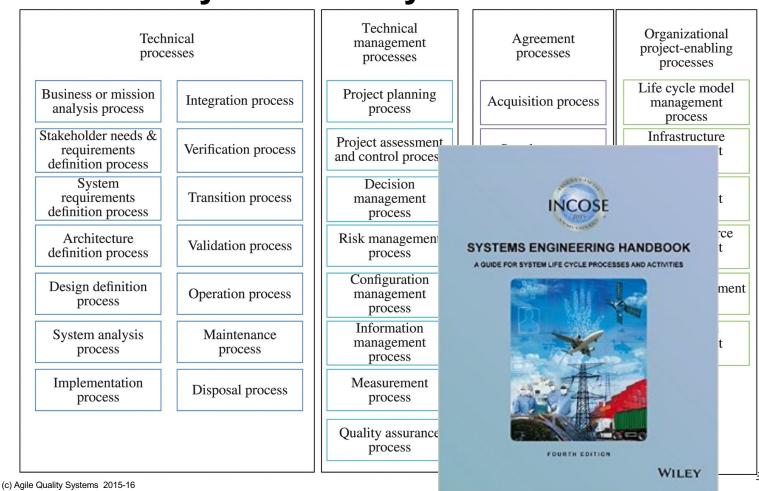
Kelly@AgileQualitySystems.com

Copyright © 2018 by Agile Quality Systems. Permission granted to INCOSE to publish and use.

NASA SE Handbook Requirements flow down Realized products from level above to level above TECHNICAL MANAGEMENT **PROCESSES** SYSTEM **PRODUCT** DESIGN Technical Planning REALIZATION **PROCESSES** Process **PROCESSES** 10. Technical Planning **Product Transition Process** Requirements Definition Technical Control 9. Product Transition Processes Processes 1. Stakeholder Expectations 11. Requirements Management Definition **Evaluation Processes** 2. Technical Requirements 12. Interface Management 7. Product Verification Definition 13. Technical Risk Management 8. Product Validation 14. Configuration Management 15. Technical Data Management Technical Solution Design Realization Technical Assessment **Definition Processes** Processes Process 3. Logical Decomposition 5. Product Implementation 16. Technical Assessment 4. Design Solution Definition 6. Product Integration Technical Decision Analysis Process 17. Decision Analysis Requirements flow down Realized products to level below from level below System design processes Product realization processes applied to each work breakdown applied to each product structure model down and up and across across system structure system structure

Figure 2.1-1 The systems engineering engine

ISO 15288 System Life Cycle Processes



The Systems **Engineering Engine**



System Development Processes System Design Product Realization

Define Stakeholder Expectations

Validate

Define Requirements

Verify

Architect

Integrate

Design

Implement

Technical Management Processes

Technical Planning

Requirements Management

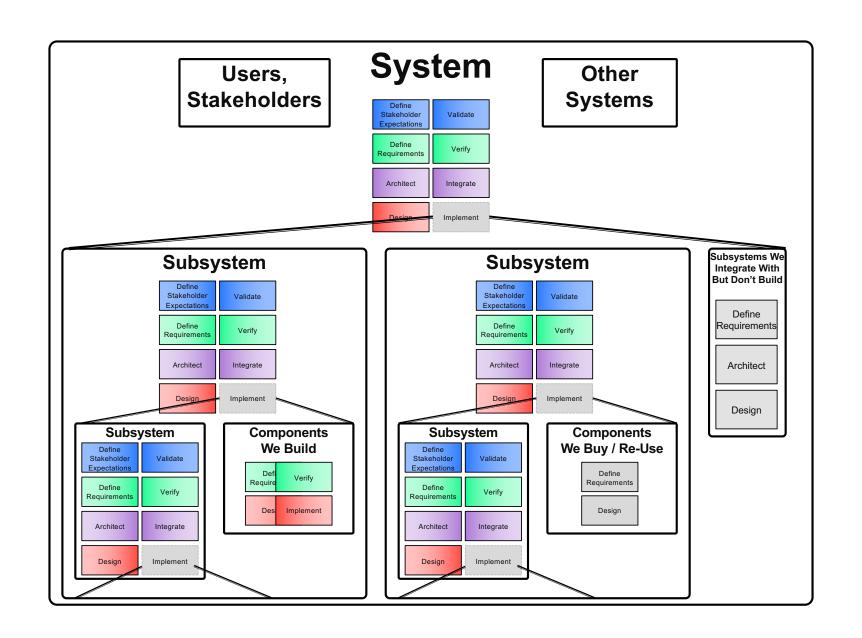
Interface Management

Risk Management

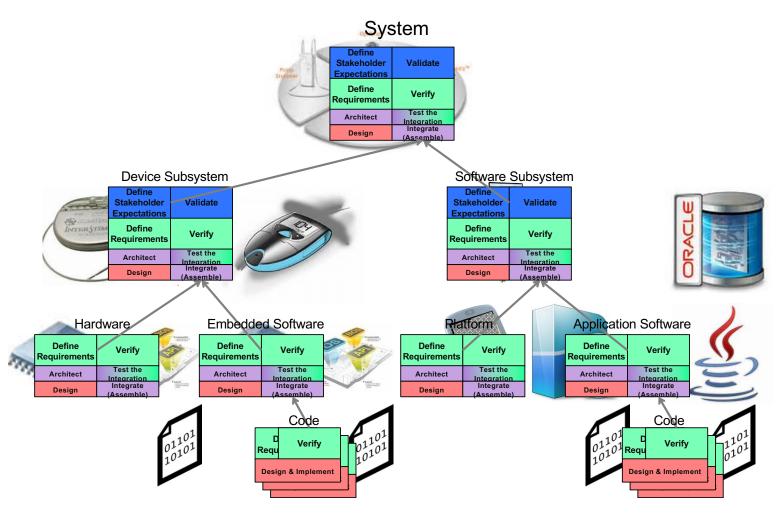
Configuration Management

Data Management

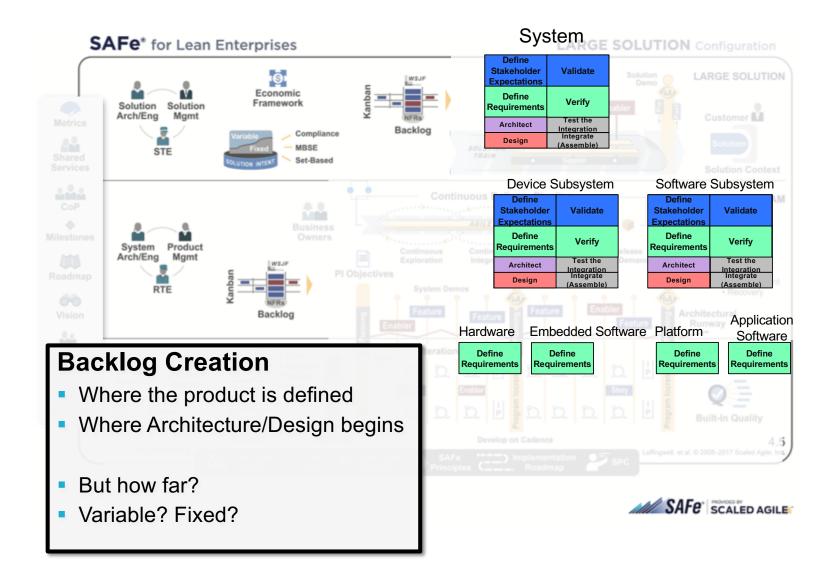
Adapted from: NASA Systems Engineering Handbook, NASA/SP-2007-6105

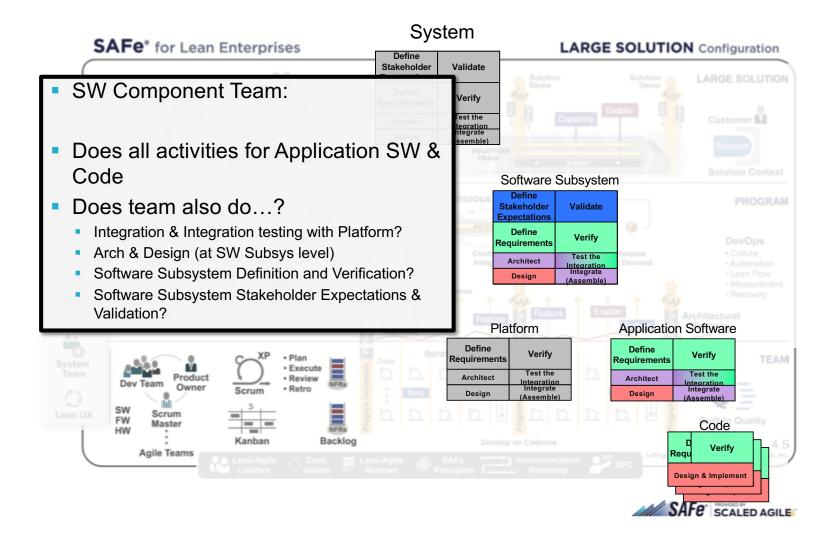


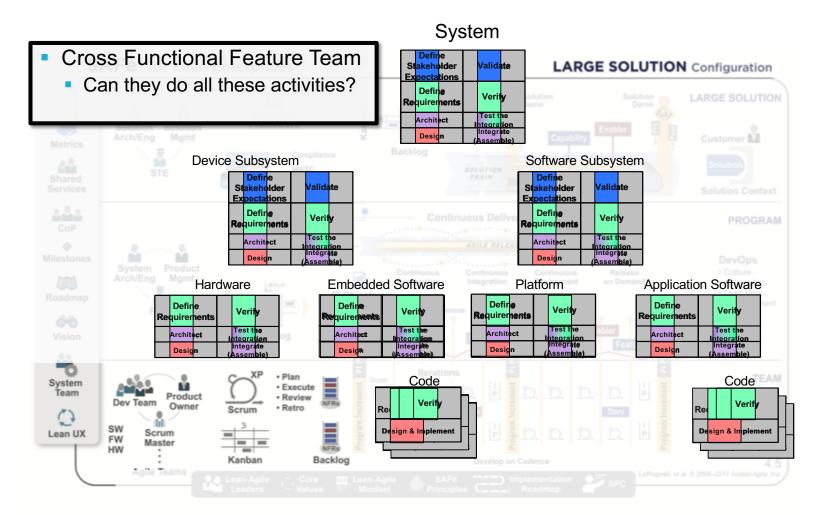
What is the System? And What are the Activities & Deliverables?



SAFe* for Lean Enterprises **LARGE SOLUTION** Configuration LARGE SOLUTION Economic Framework Solution Solution Arch/Eng Mgmt Customer M Metrics Compliance Stakeholder Validate Expectations MBSE TRAIN STE Shared Define Set-Based Verify Services Solution Context Requirements Test the Architect Integration Integrate us Delivery Pipeline **PROGRAM** Design Å CoP (Assemble) GILE RELEASE TRAIN 0 Business Milestones Owners DevOps Syst Device Subsystem Software Subsystem Culture Continuous Release Define Exploration on Demand Automation Define Lean Flow Stakeholder **Validate** PI Objectives Stakeholder **Validate** Roadmap Measurement Expectations Expectations System Demo · Recovery 00 Define **Define** Verify Verify Requirements Requirements Feature Backlog Architectural Vision Feature Runway Test the Test the Architect Architect Integration Integrate Design Design (Assemble) (Assemble) Iterations • Plan TEAM System Dev Team Execute Team Product Owner · Review Embedded Software Platform Application Software Hardware Define Define Define Define It-In Quality Verify Verify Verify Verify Requirements Requirements Requirements Requirements Test the Test the Test the Test the 4.5 Architect Architect Architect Architect Integration Integration Integration Integrate Integrate Integrate Design Design Design Design (Assemble) (Assemble) (Assemble) Code Code ED AGILES Verify Requ Requ **Design & Implement Design & Implement**









Layers / Kinds of Backlog Items

(as defined by the Scaled Agile Framework, SAFe®)

- Four Levels
 - Portfolio-Level Epics
 - Large-Solution-Level Capabilities
 - Program-Level Features
 - Team-Level Stories
- Two Perspectives
 - Customer-Facing, Business Value
 - Solution-Facing, Enablers

	Epics (Portfolio)	Capabilities (Large Solution)	Features (Program)	Stories (Team)
Responsible	Portfolio Management Enterprise Architect (+PM, +SE)	Solution Management, Solution Architect (+PM, +SE)	Product Management (+PO, +SE)	Product Owner (+Team)
Provides Value to	Customers, Business	"Who" of the Capability, Customers, Enterprise Architect, The Epic	"Who" of the Feature, Business Owners, Solution Architect, The Capability/Epic	"Who" of the Story, System Architect, The Feature
Delivered by	N/A, delivered through Capabilities & Features	Agile Teams, System Team, Specialty Team?	Agile Teams, System Team	Agile Teams
Delivered when	Depends - When all Capabilities/Feat ures complete? Only some?	When all Features complete (1 or more Increments)	Each Program Increment	Each Sprint
Demoed	N/A, Demo done with lower layers	System Demo, Validation Studies	System Demo	Team's Sprint Demo
Content	Lightweight Business Case	Description & Benefit, Acceptance Criteria, Definition Of Done	Description & Benefit, Acceptance Criteria, Definition Of Done	Story Pattern, Acceptance Criteria, Definition Of Done

(c) Agile Quality Systems 2015-17

Solution-Facing Backlog Items

- SAFe® term: "Enablers"
 - Enabling the "Architectural Runway" upon which customer-facing value can be delivered.
- Infrastructure (Development, Product)
- Debt
- Spikes (Definition, Technical, Decision)
- System Integration
- Quality System Satisfaction

• . . .

13

SAFe* for Lean Enterprises **LARGE SOLUTION** Configuration System Product System Engineering expertise needed for Arch/Eng Mgmt **Decomposition and Allocation** Of Program-Level Features into Team-Level Stories Backlog Identification and Sequencing of Solution-Facing Infrastructure Enablers Identification of Spikes And execution of them? Backlog Managing Technical Debt

SAFe* for Lean Enterprises LARGE SOLUTION Configuration A.A ůů **System Team** Integration and Test activities Backlog * Lean UX System Execute Team Provide input to the Backlogs Guide teams during execution Lean UX Backlog Validation activities

Agile & Systems Engineering

- Agile Development depends on
 - Solid Product Architecture (Physical Architecture)
 - Solid Backlog Architecture (Functional Architecture)
 - Solid Development Team Architecture
- Systems Engineers speak the language of Architecture and System Development Activities
- SAFe speaks the language of Lean/Agile
- Align them to realize the benefits of both