

Getting from ARTs to Large Solutions

Cindy VanEpps, SPCT



Topic Flow



Definition of Large Solution - SAFe®

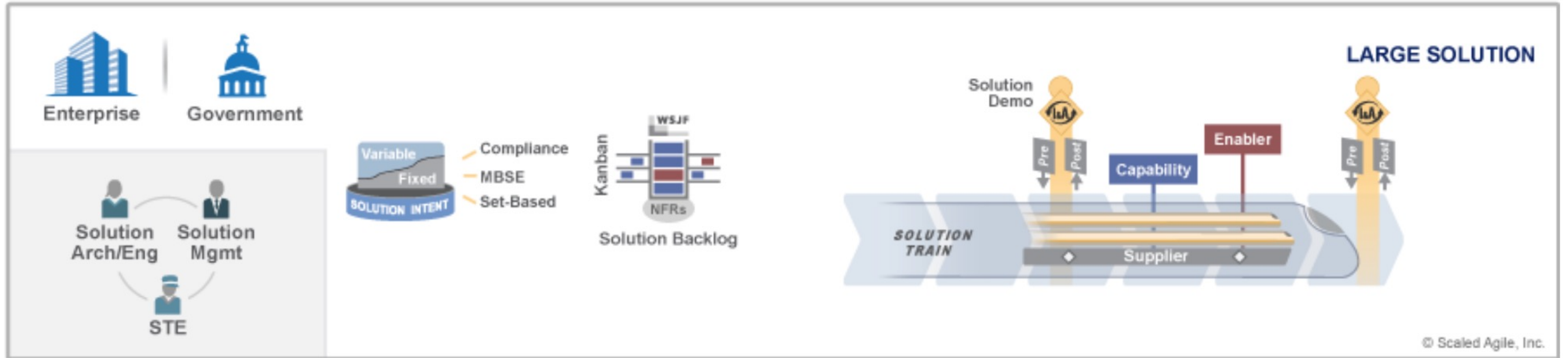


Figure 2. Large Solution Level

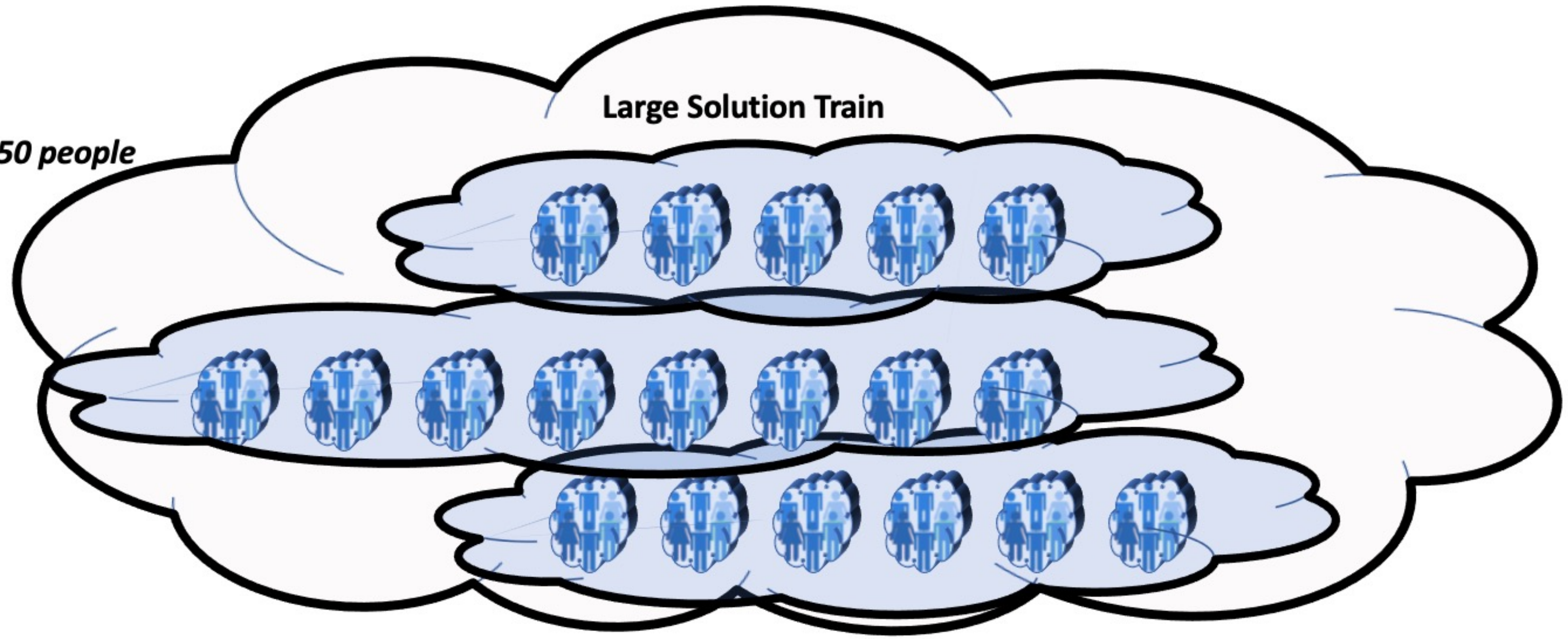
Large Solution SAFe is meant for enterprises that face the biggest challenges—building large-scale solutions that are beyond the scope of a single ART to develop. Building these solutions requires additional roles, artifacts, events, and coordination.

© Scaled Agile, Inc.

<https://www.scaledagileframework.com/large-solution-safe/>

150 - 450 people

Large Solution Train



50 -125 people

Agile Release Train



5-12 people



Agile Team



Paths to Large Solution: 1

Multiple ARTs needed to support one Value Stream – usually clear from the start

1. Launch ARTs with independent backlogs first for 1+ PIs
2. Create large solution roles and backlogs
3. Add Pre and Post Planning, Large Solution sync meetings, demos, metrics . . .

Examples of Path 1

Delivering a complex product, e.g. an airplane, a specific year & model of an automobile, a medical ultrasound system





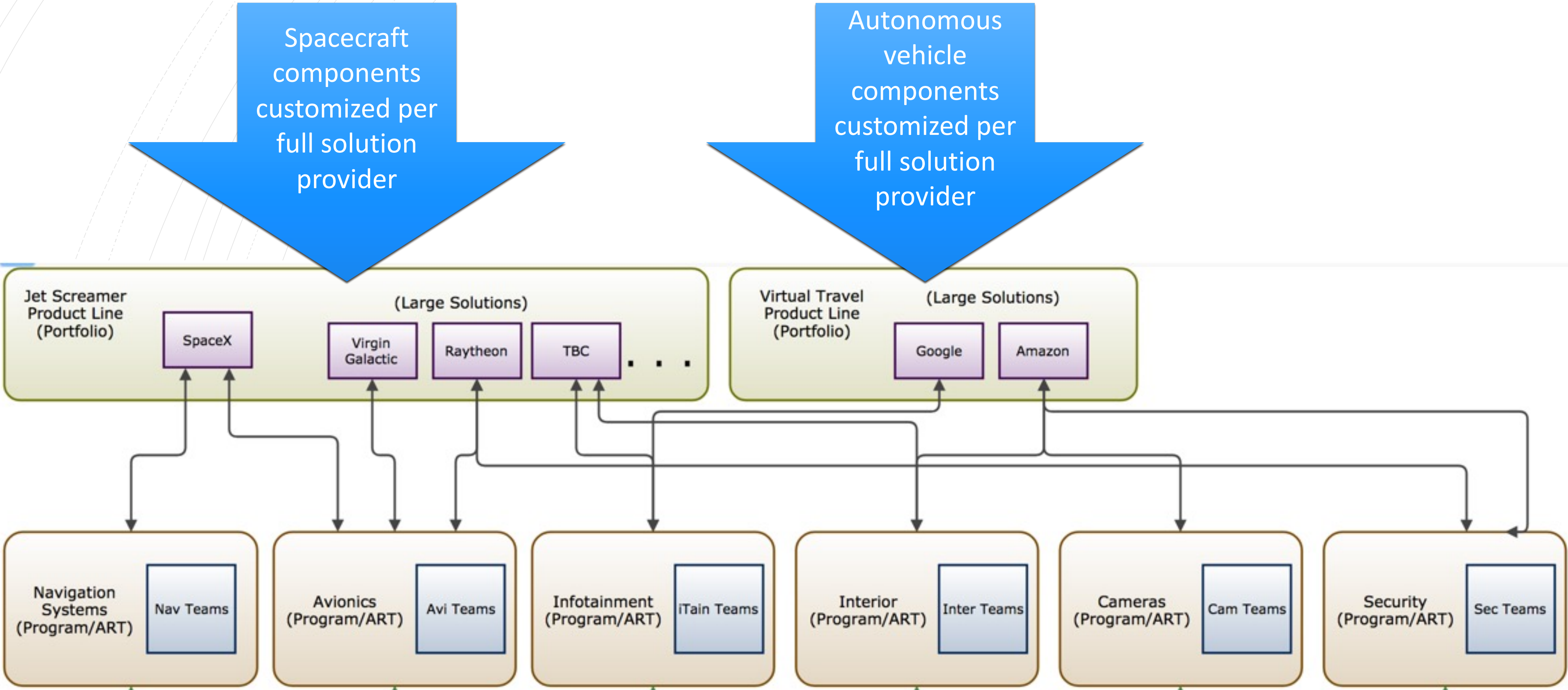
Paths to Large Solution: 2

- Multiple ARTs in an organization, formed one-at-a-time
- ART cross-dependencies managed ad hoc as external dependencies
- Initiatives from higher in the Enterprise are disruptive to multiple ARTs

Examples of Path 2

- Automotive components that are supplied standalone to an automaker or can be integrated into a comprehensive solution for an automaker
- IT Infrastructure solutions that may be provided standalone or integrated, e.g. WAN, LAN, wifi, security, desktops . . .
- Enterprise support systems that may be updated independently and are also integrated, e.g insurance application, policy provisioning and claims processing.

Example of Large Solution Path 2



Challenges

- What exactly is the Portfolio and who owns it?
- How to leverage and manage reuse across ARTs?
- How to structure Solution Intent?

How to Build an Effective Large Solution

Fundamentally its an architecture problem & solution.

Identify the overarching value provided by the Solution

“We provide customized and integrated infotainment systems to auto manufacturers”














Portfolio Canvas

Portfolio Canvas

Portfolio Name:

Date:

Version:

Value Propositions 						
Value Streams 	Solutions 	Customers 	Channels 	Customer Relationships 	Budget 	KPIs / Revenue 
Key Partners 		Key Activities 			Key Resources 	
Cost Structure 				Revenue Streams 		

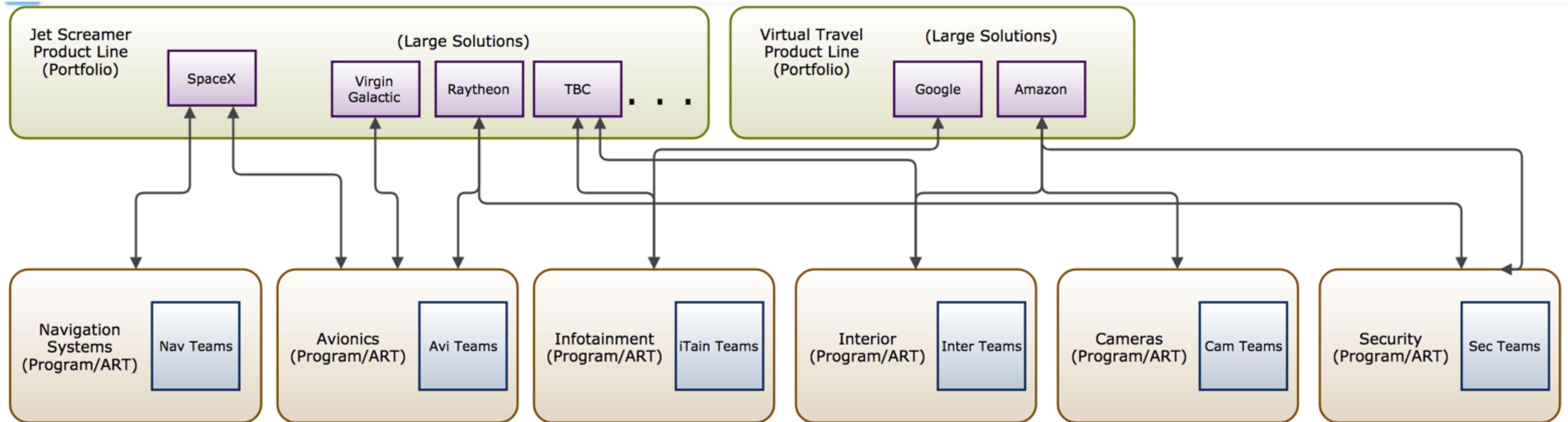
The Portfolio Canvas is adapted from The Business Model Canvas (<http://www.businessmodelgeneration.com>).
 This work is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported License.
 To view a copy of this license visit: <http://creativecommons.org/licenses/by-sa/3.0>.

<https://www.scaledagileframework.com/portfolio-vision/>

How to Build an Effective Large Solution

Identify the responsibilities of and interactions between the ARTs

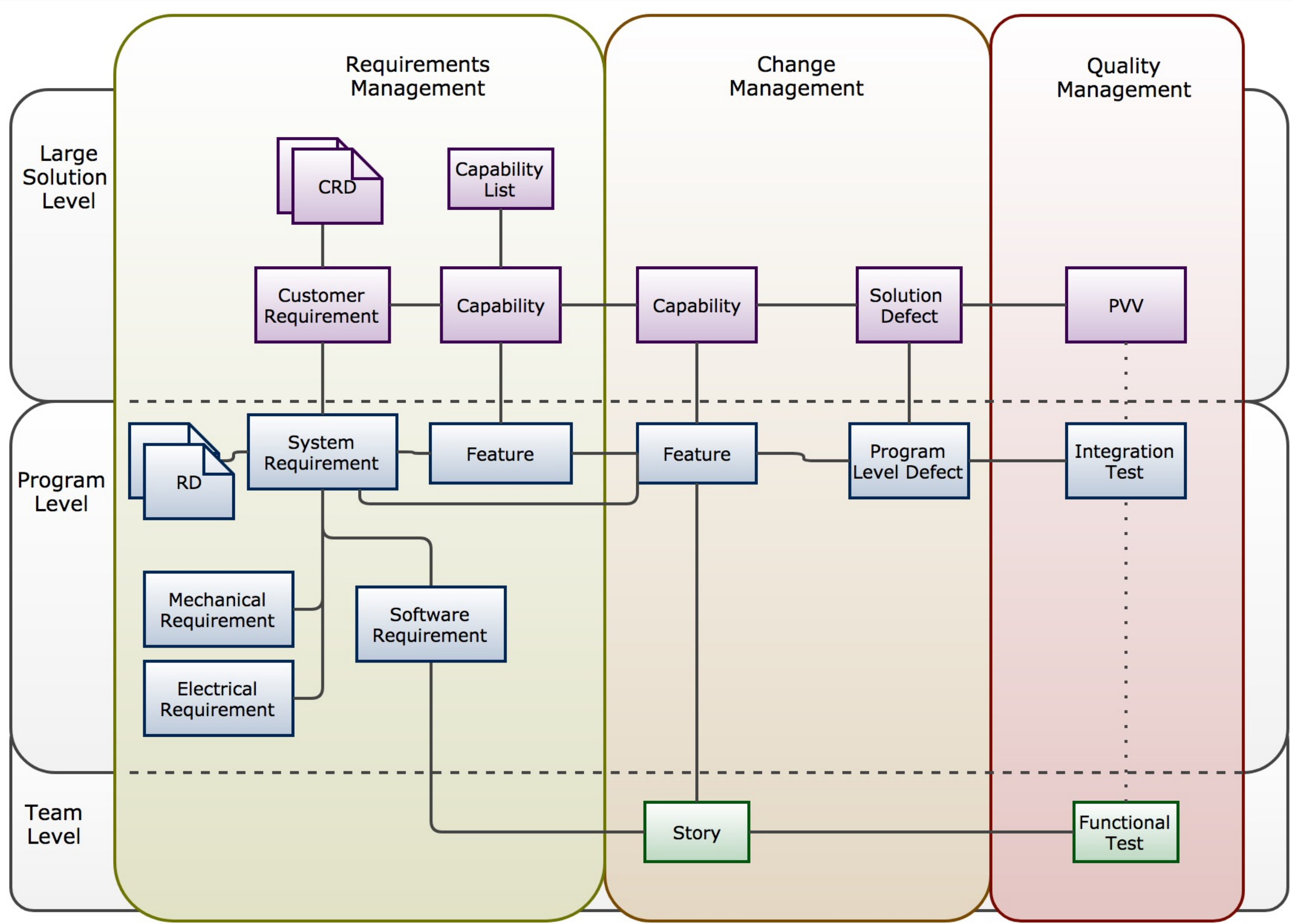
- How do customer-specific solutions tie into reuse?
- Who owns the overarching customer solution requirements, design, etc?



An orange callout box with a white pointer pointing downwards, containing the text "Have Clarity on Solution Intent".

Have Clarity on Solution
Intent

Solution Intent Design



Context

Transient

Persistent

Level	Work Management	Requirements Management	Quality Management	Design Management
Portfolio	Backlog, Epics, Funding			
Large Solution	Backlog, Epics, Capabilities, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	Solution integration tests, plans and results. Solution regression tests.	Solution components and interfaces
Program	Backlog, Epics, Features, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	System integration tests, plans and results. System regression tests.	System components and interfaces
Team	Backlog, Stories, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	Component tests, plans and results. Component regression tests.	Component context

Traceability Up and Down Layers

Level	Work Management	Requirements Management	Quality Management	Design Management
Portfolio	Backlog, Epics, Funding			
Large Solution	Backlog, Epics, Capabilities, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	Solution integration tests, plans and results. Solution regression tests.	Solution components and interfaces
Program	Backlog, Epics, Features, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	System integration tests, plans and results. System regression tests.	System components and interfaces
Team	Backlog, Stories, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	Component tests, plans and results. Component regression tests.	Component context

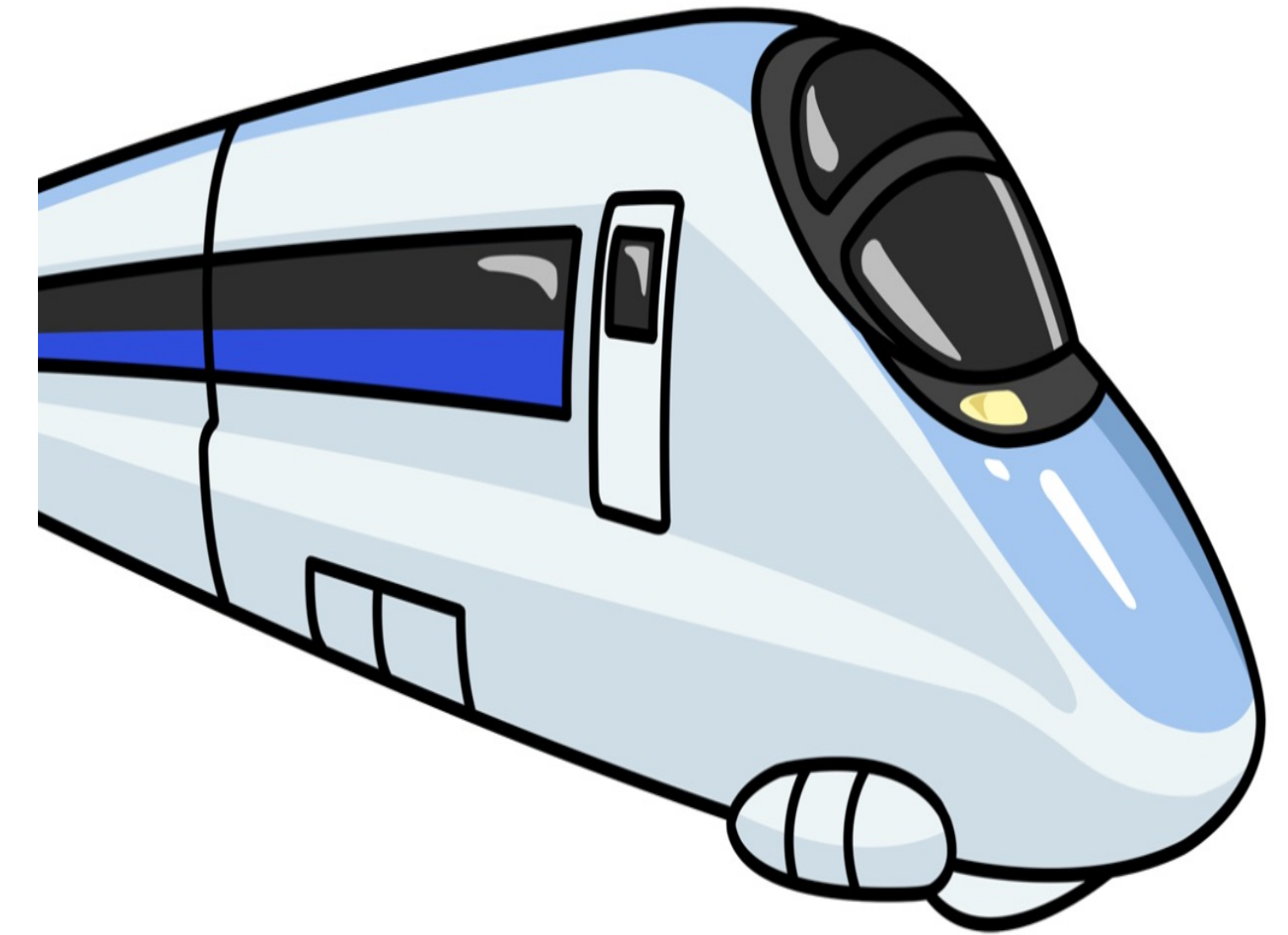
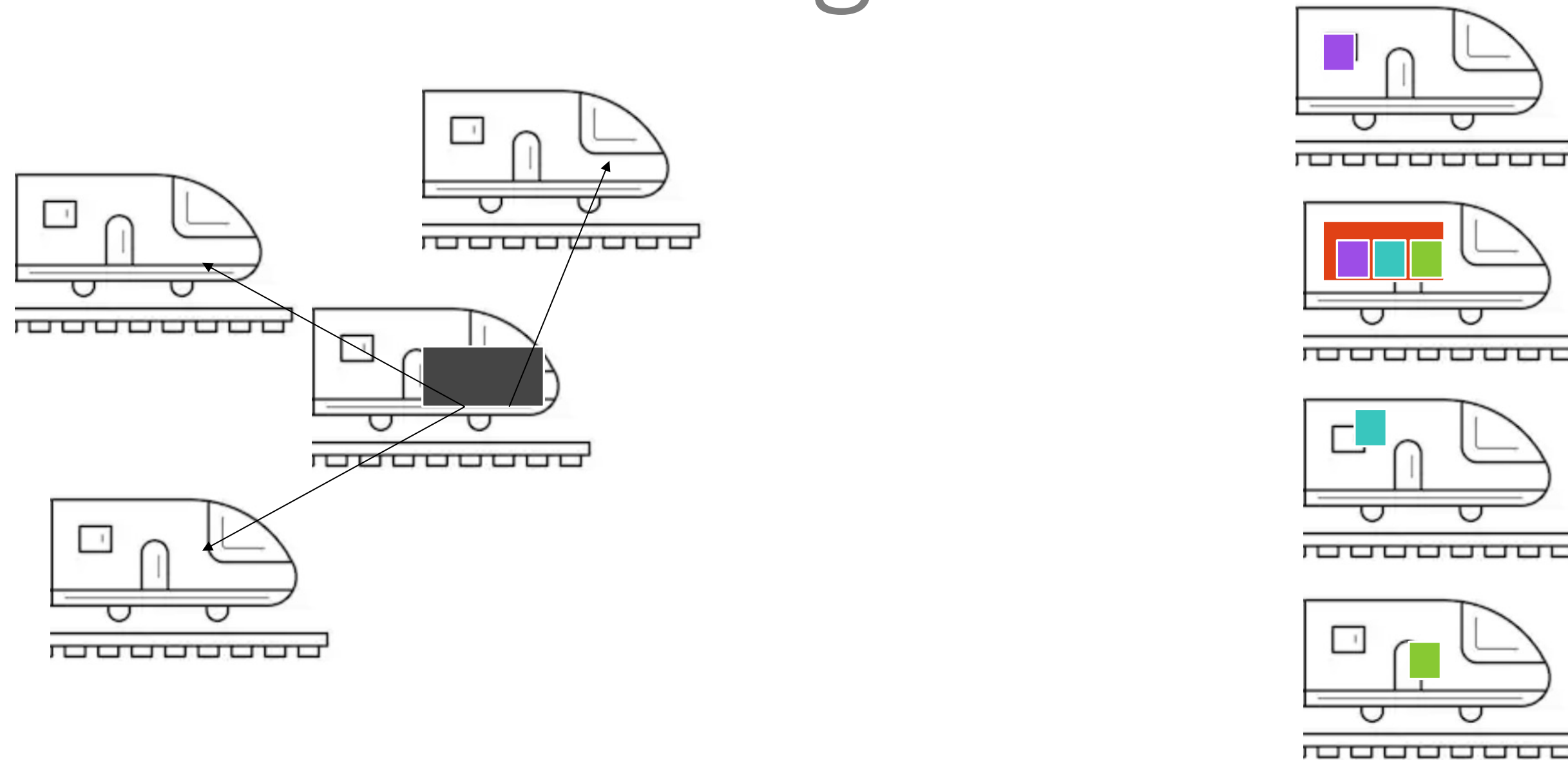
Traceability Across Domains

Level	Work Management	Requirements Management	Quality Management	Design Management
Portfolio	Backlog, Epics, Funding			
Large Solution	Backlog, Epics, Capabilities, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	Solution integration tests, plans and results. Solution regression tests.	Solution components and interfaces
Program	Backlog, Epics, Features, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	System integration tests, plans and results. System regression tests.	System components and interfaces
Team	Backlog, Stories, Plans, Progress, Dependencies	As built and to-be acceptance criteria and scenarios, common requirements and variants	Component tests, plans and results. Component regression tests.	Component context

Do We Need Large Solution Constructs?

- What percentage of each ART's backlog comes from a higher-level Initiative/Epic?
- Who owns the priorities and trade-offs of the Initiatives/Epics that involve contributions across ARTs?
- Who owns the priority of a single ART's backlog when it contains both locally-provided value as well as from cross-ART initiative?

From ARTs to Large Solution

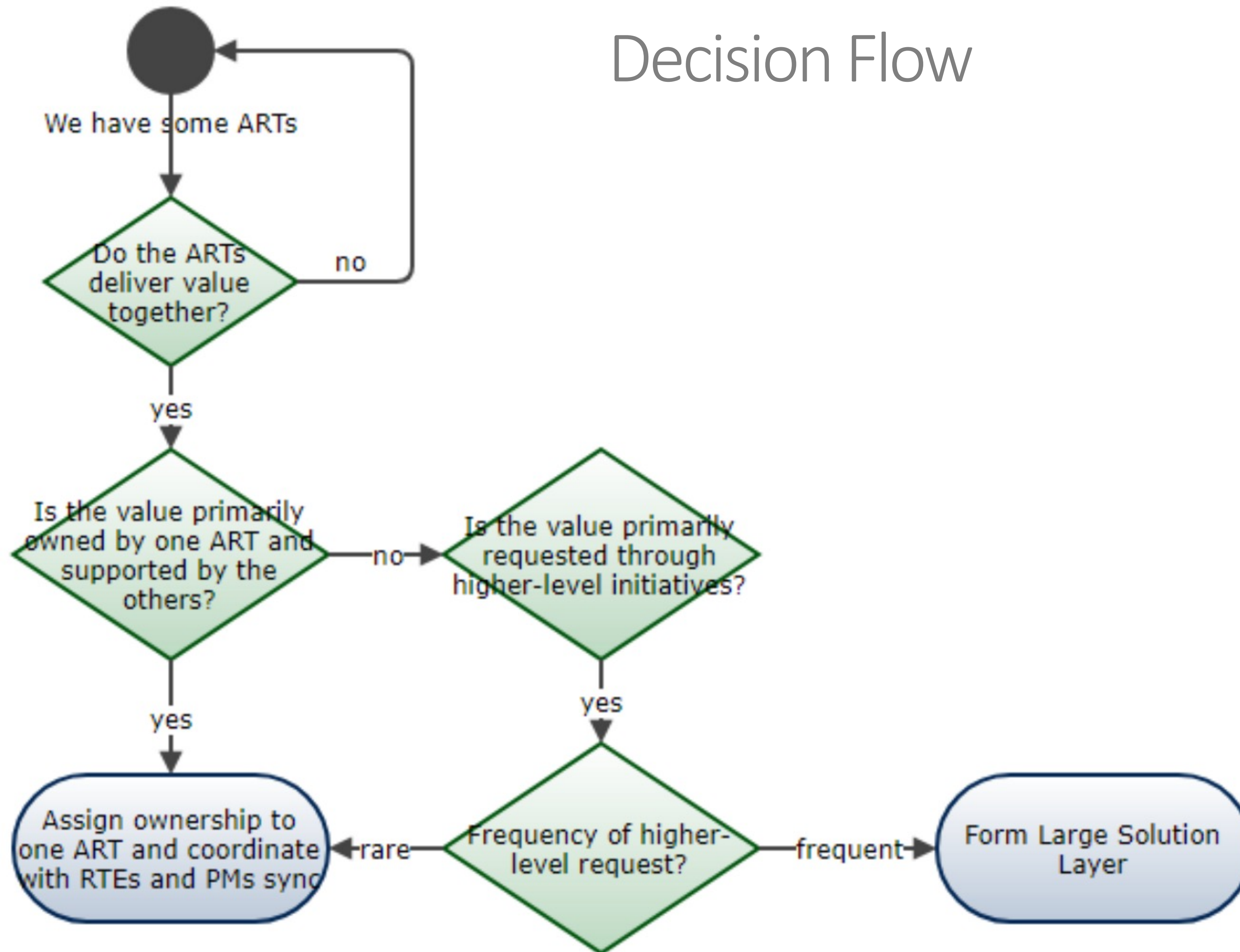


- **Multiple independent ARTs**
- **Ad hoc dependency mgmt**
- **Designated ART as owner of value delivery**

- **Cadence-aligned ARTs**
- **Coordination of RTEs and PMs for dependency mgmt**
- **One ART selected as owner of value delivery**

- **Large Solution with Backlog, Roles, Events, & Architecture to manage dependencies and value delivery**

Decision Flow





Project & Team, Inc.

Local: +1 415 84-AGILE

Toll Free: +1 833-431-0442

hello@projectandteam.com

projectandteam.com