27th annual INCOSE international symposium

Adelaide, Australia

July 15 - 20, 2017

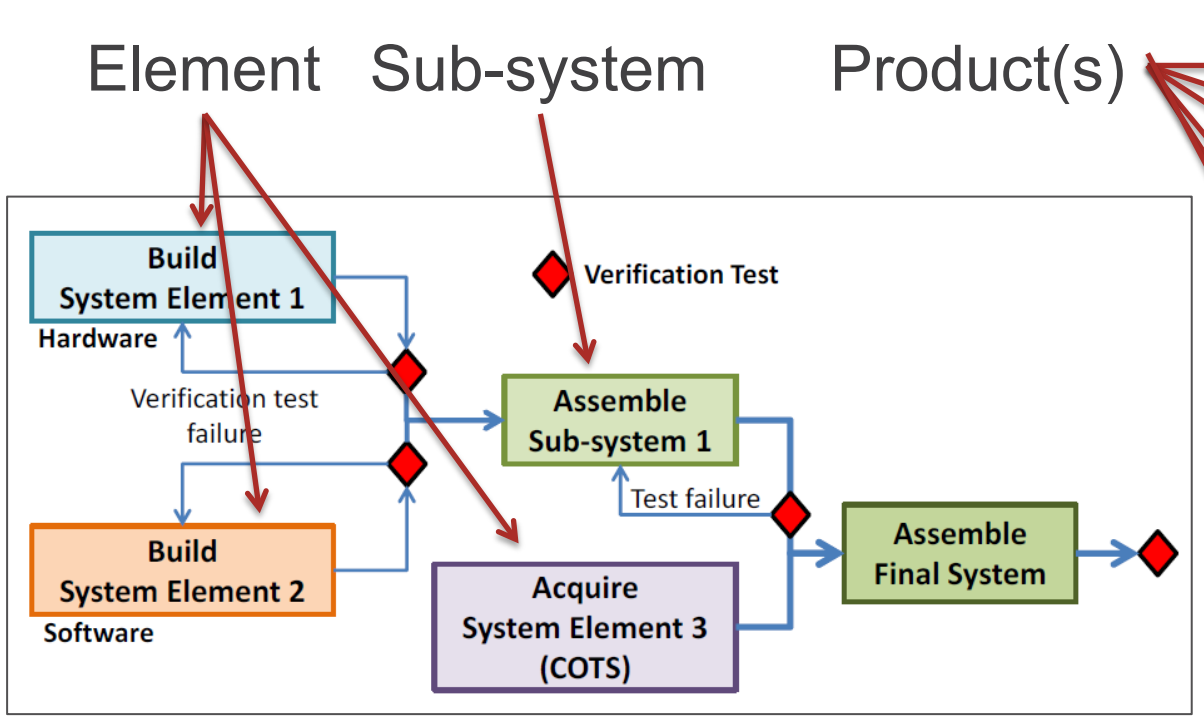
Improving Integration: Thinking Beyond the Physical Architecture

Jim Armstrong, Stevens Institute of Technology

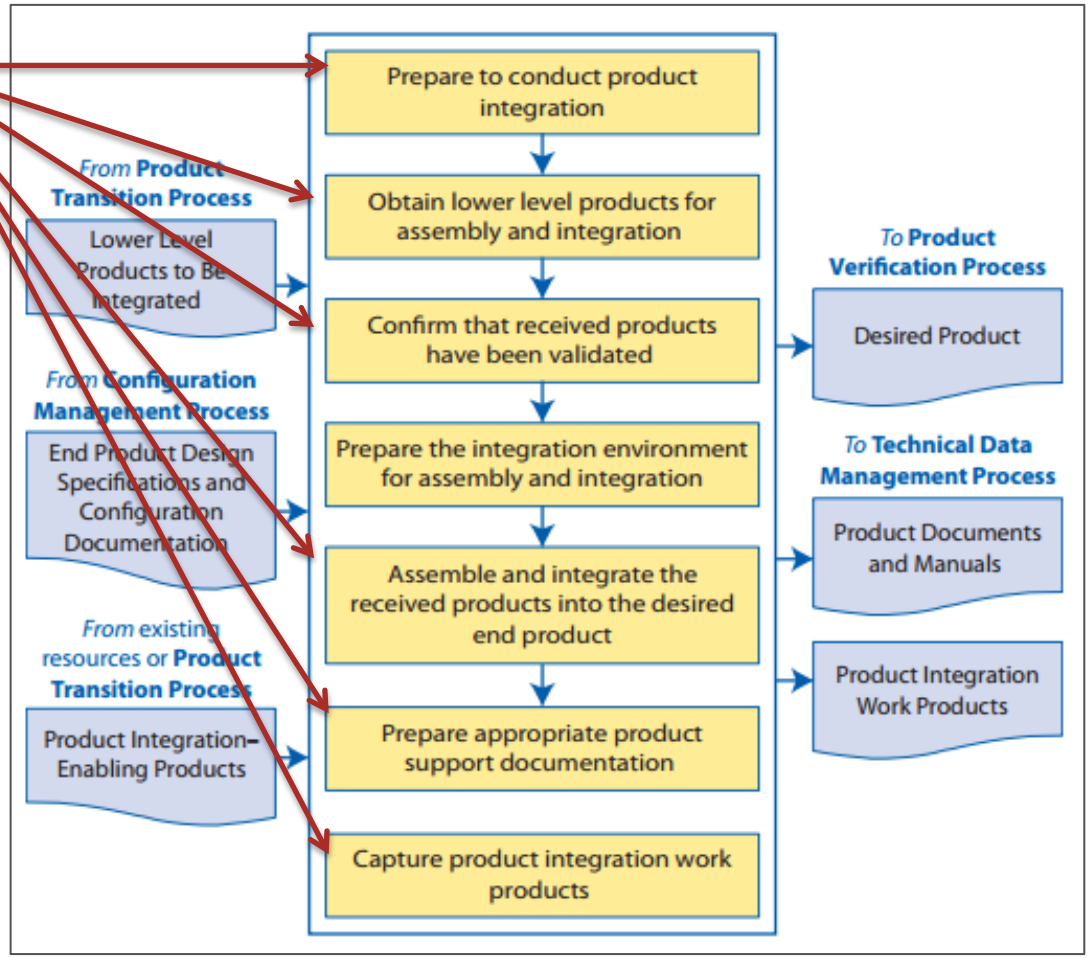
A Problem in the Definitions

- INCOSE Handbook: "…synthesize a set of system elements into a realized system (product or service)…"
  - System elements = *hardware, software, and operational resources*
- DoD Guidebook: "…assemble lower-level system elements…"
- NASA SE Handbook: "…assemble the higher level product from the lower level products or subsystems …"
- CMMI®: "…assemble the product from the product components…

Visual Bias to Physical



FAA SE Manual integration process diagram



NASA product integration process

Textbooks

- Kossiakoff and Sweet, Buede, Martin, Stevens: all limit to components
- Grady: Includes process components
- Larson, et al: Some mention of combining capabilities

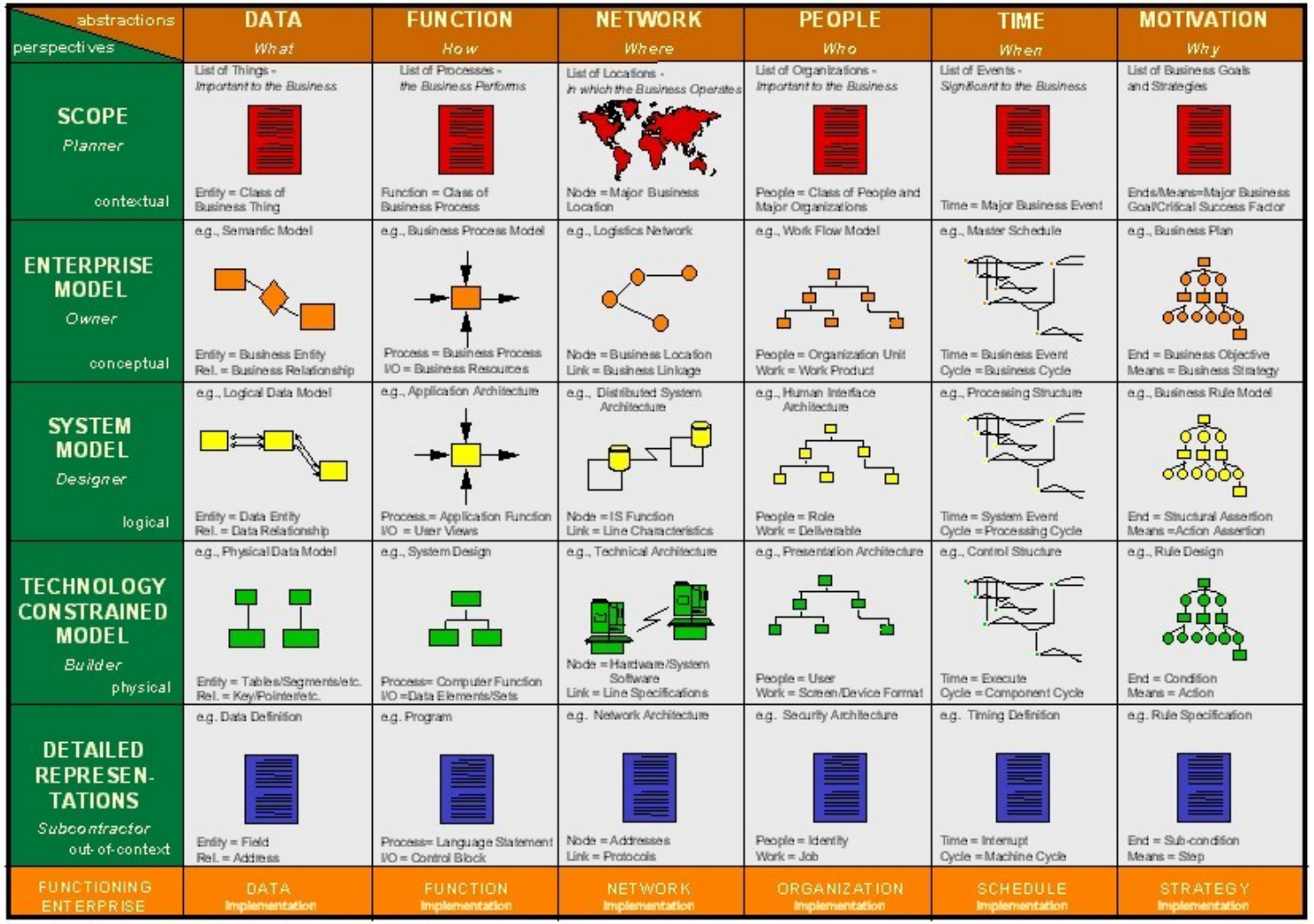
Architecture Frameworks

- Add multiple additional considerations to define a system
- Include Zachman, DoDAF, MODAF, TOGAF and others

Zachman Framework

Includes:

- Data
- Function
- Network
- People
- Time
- Motivation



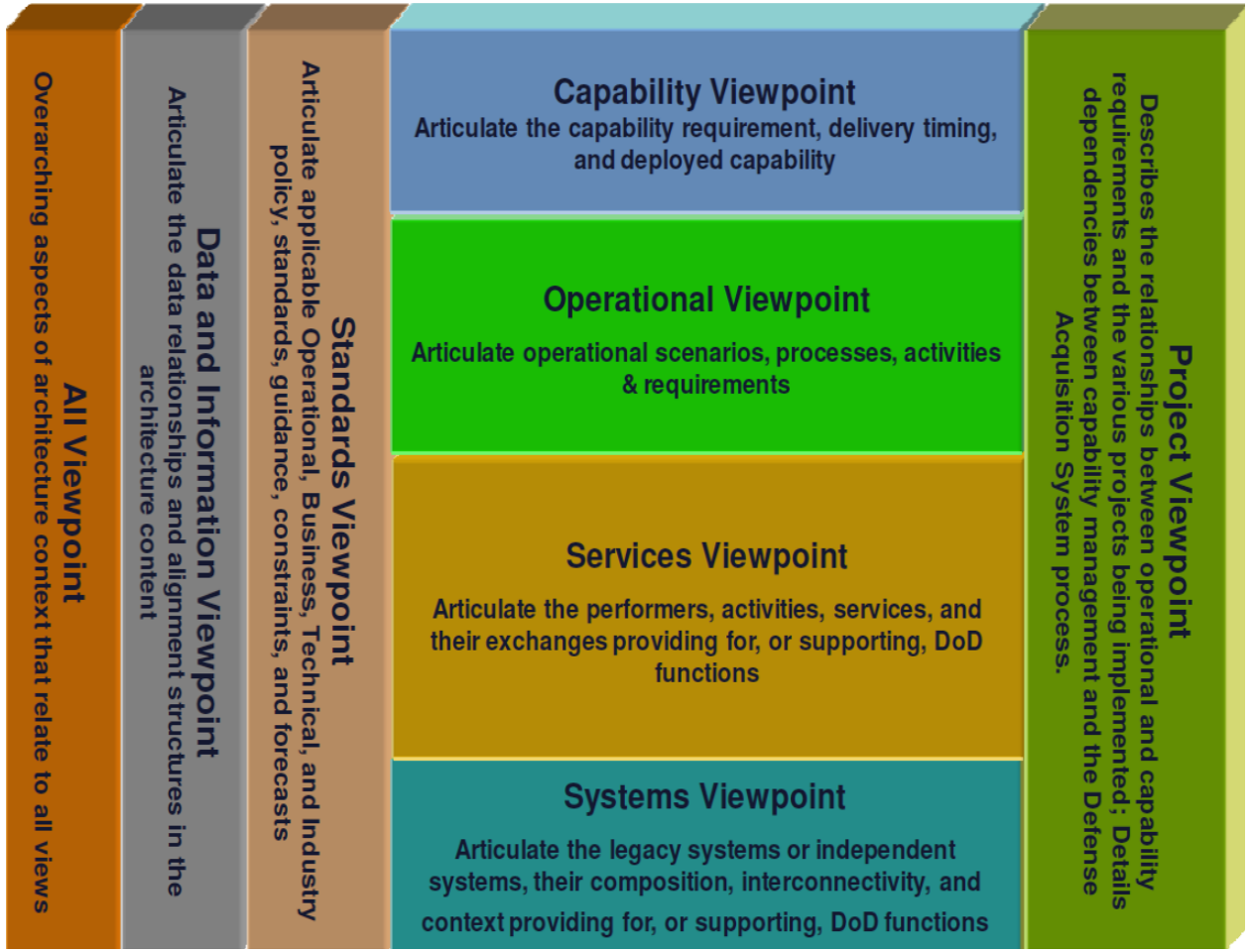
abstractions / perspectives	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*
SCOPE *Planner* contextual	List of Things - *Important to the Business*	List of Processes - *the Business Performs*	List of Locations - *in which the Business Operates*	List of Organizations - *Important to the Business*	List of Events - *Significant to the Business*	List of Business Goals and Strategies
	Entity = Class of Business Thing	Function = Class of Business Process	Node = Major Business Location	People = Class of People and Major Organizations	Time = Major Business Event	Ends/Means=Major Business Goal/Critical Success Factor
ENTERPRISE MODEL *Owner* conceptual	e.g., Semantic Model	e.g., Business Process Model	e.g., Logistics Network	e.g., Work Flow Model	e.g., Master Schedule	e.g., Business Plan
	Entity = Business Entity Rel. = Business Relationship	Process = Business Process I/O = Business Resources	Node = Business Location Link = Business Linkage	People = Organization Unit Work = Work Product	Time = Business Event Cycle = Business Cycle	End = Business Objective Means = Business Strategy
SYSTEM MODEL *Designer* logical	e.g., Logical Data Model	e.g., Application Architecture	e.g., Distributed System Architecture	e.g., Human Interface Architecture	e.g., Processing Structure	e.g., Business Rule Model
	Entity = Data Entity Rel. = Data Relationship	Process.= Application Function I/O = User Views	Node = IS Function Link = Line Characteristics	People = Role Work = Deliverable	Time = System Event Cycle = Processing Cycle	End = Structural Assertion Means = Action Assertion
TECHNOLOGY CONSTRAINED MODEL *Builder* physical	e.g., Physical Data Model	e.g., System Design	e.g., Technical Architecture	e.g., Presentation Architecture	e.g., Control Structure	e.g., Rule Design
	Entity = Tables/Segments/etc. Rel. = Key/Pointer/etc.	Process= Computer Function I/O =Data Elements/Sets	Node = Hardware/System Software Link = Line Specifications	People = User Work = Screen/Device Format	Time = Execute Cycle = Component Cycle	End = Condition Means = Action
DETAILED REPRESEN-TATIONS *Subcontractor* out-of-context	e.g. Data Definition	e.g. Program	e.g. Network Architecture	e.g. Security Architecture	e.g. Timing Definition	e.g. Rule Specification
	Entity = Field Rel. = Address	Process= Language Statement I/O = Control Block	Node = Addresses Link = Protocols	People = Identity Work = Job	Time = Interrupt Cycle = Machine Cycle	End = Sub-condition Means = Step
FUNCTIONING ENTERPRISE	DATA Implementation	FUNCTION Implementation	NETWORK Implementation	ORGANIZATION Implementation	SCHEDULE Implementation	STRATEGY Implementation

John A. Zachman, Zachman International

DoD Architecture Framework

Multiple additional things to consider

- Overall capability and mission
- Organizations
- People
- Services
- Functions
- Physical architecture above the physical component level



Functional Architecture

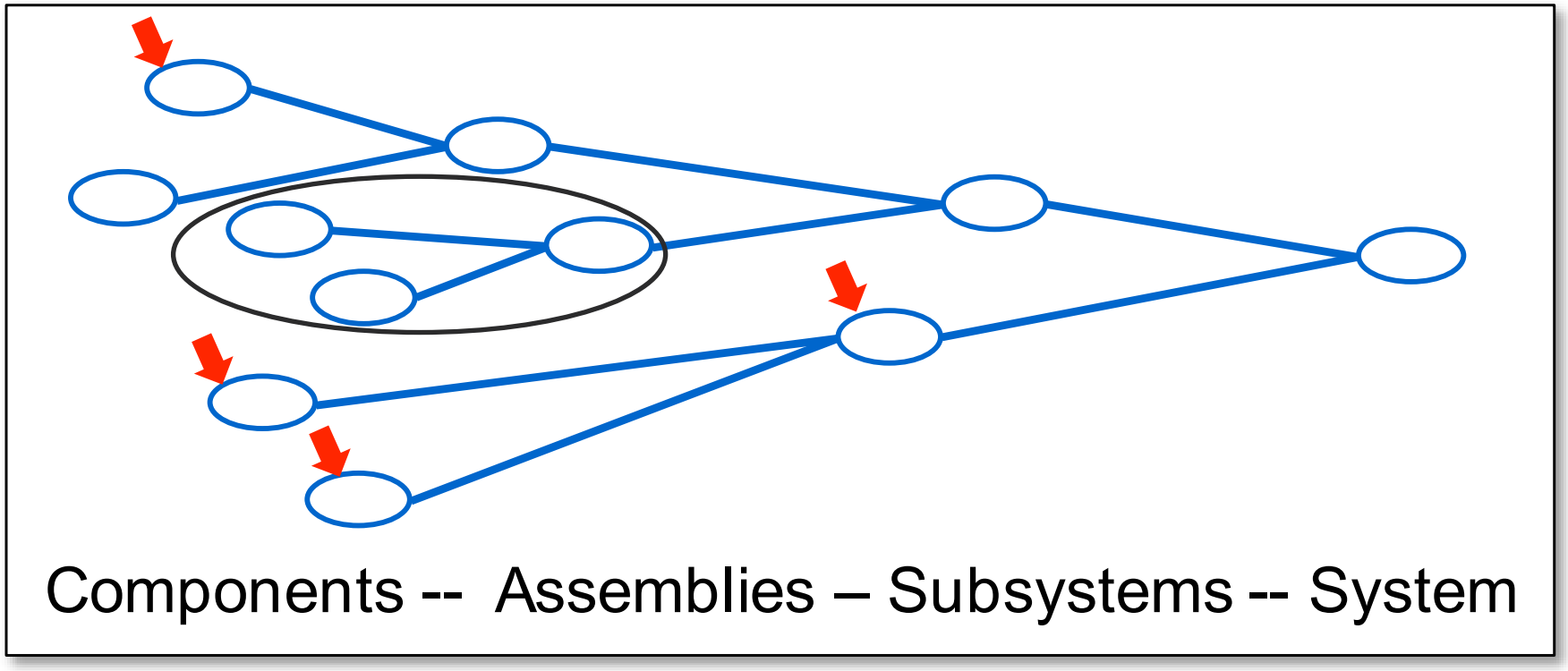
- Functionality – what the system does
- Data/material flow – what gets used/produced
- Control Flow – precedence, parallelism, loops...
- States and transitions – conditions and changes
- Functional interfaces – how functions interact

Integration Approaches

- Global (Big Bang)
- Bottom Up
- Subset
- With the Stream
- Incremental (adding on)
- Top Down
- Reorganization of coupling matrices
- Criterion Driven

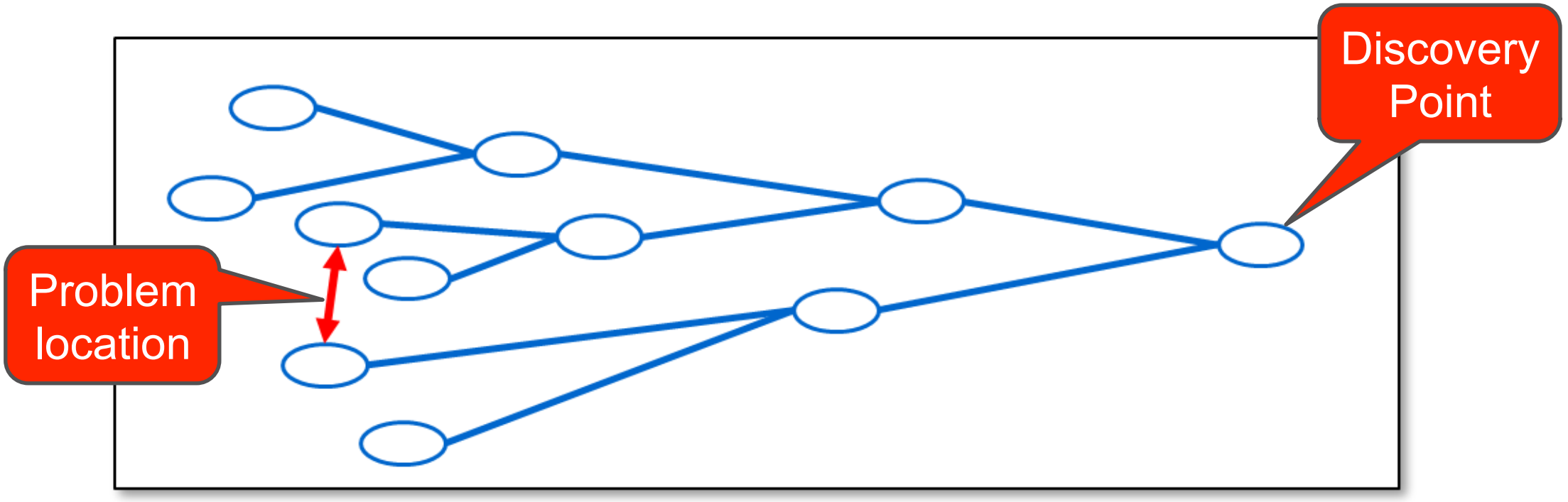
Big Bang

Bottom Up, Subset, With the Stream

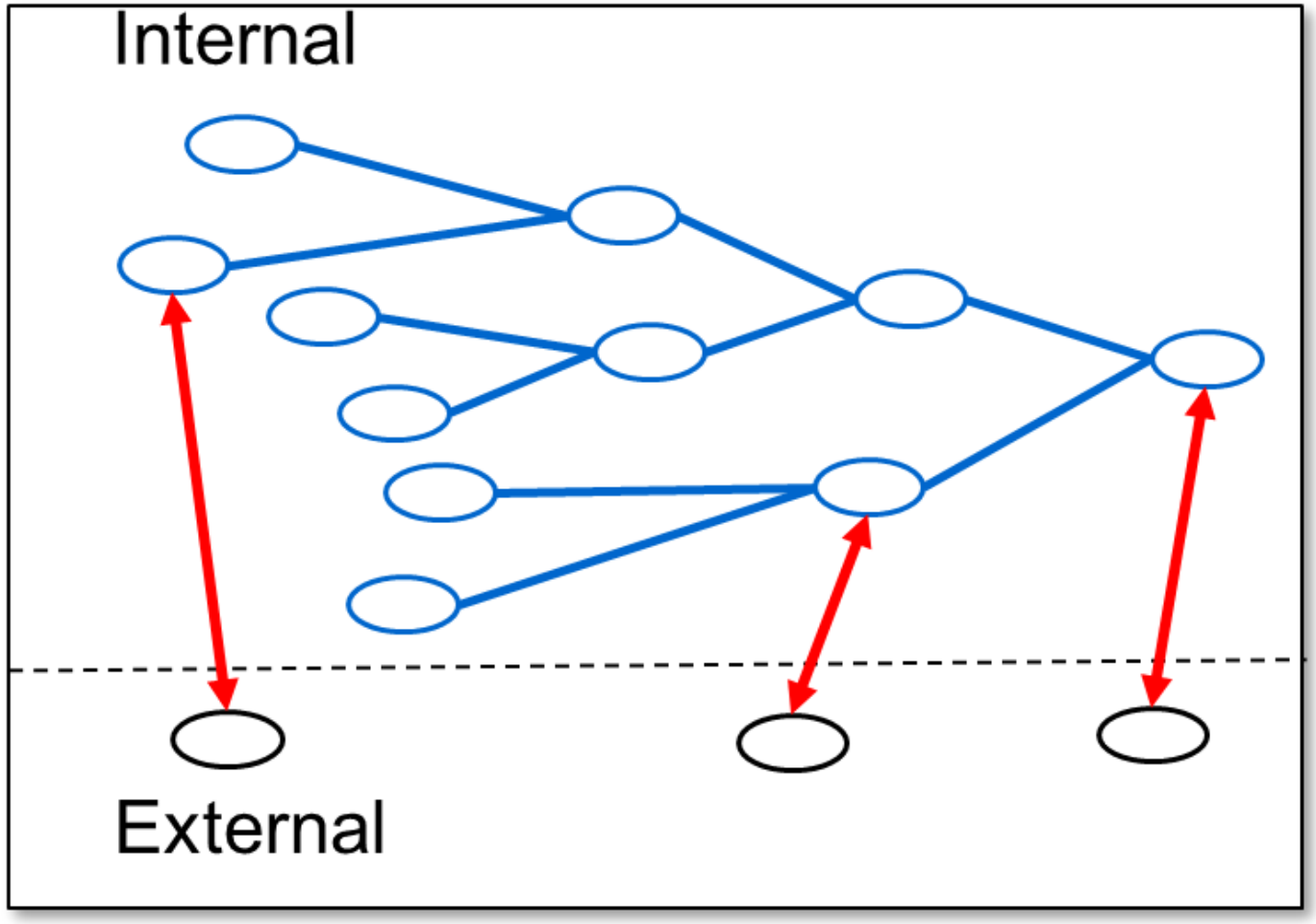


Incremental (adding on)

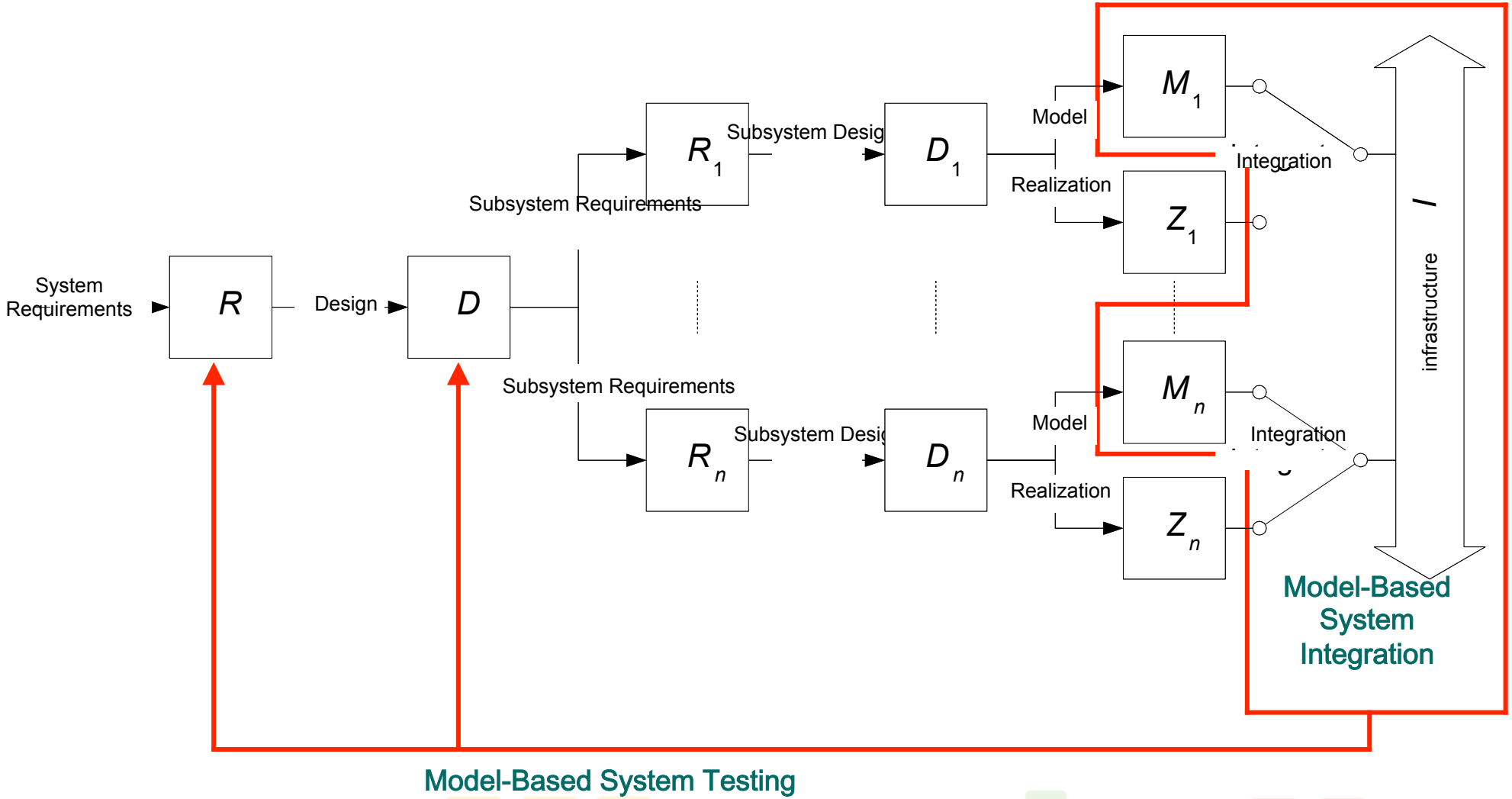
Cross-Branch Issues



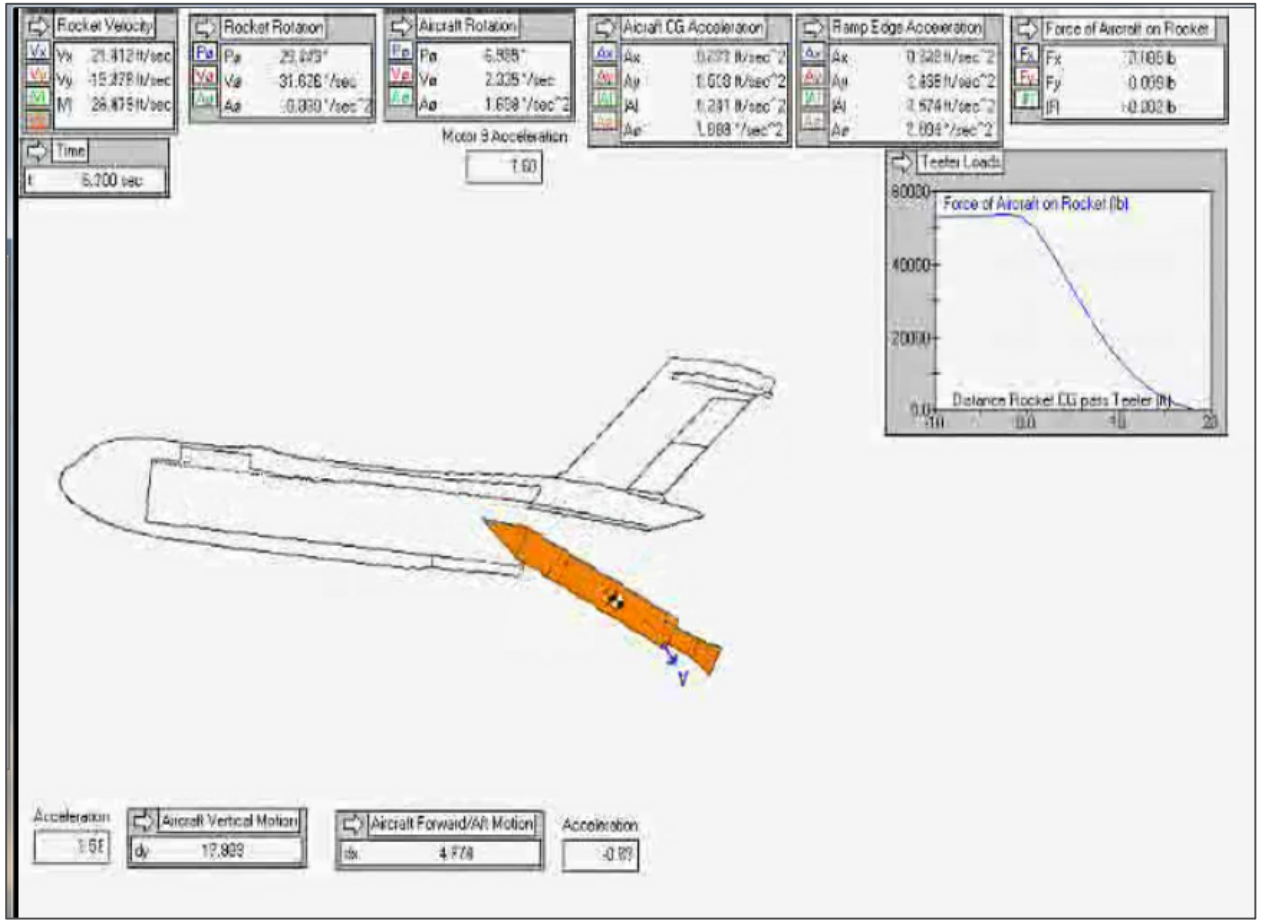
Cross-Branch Issues - Generalized



Model Based Integration

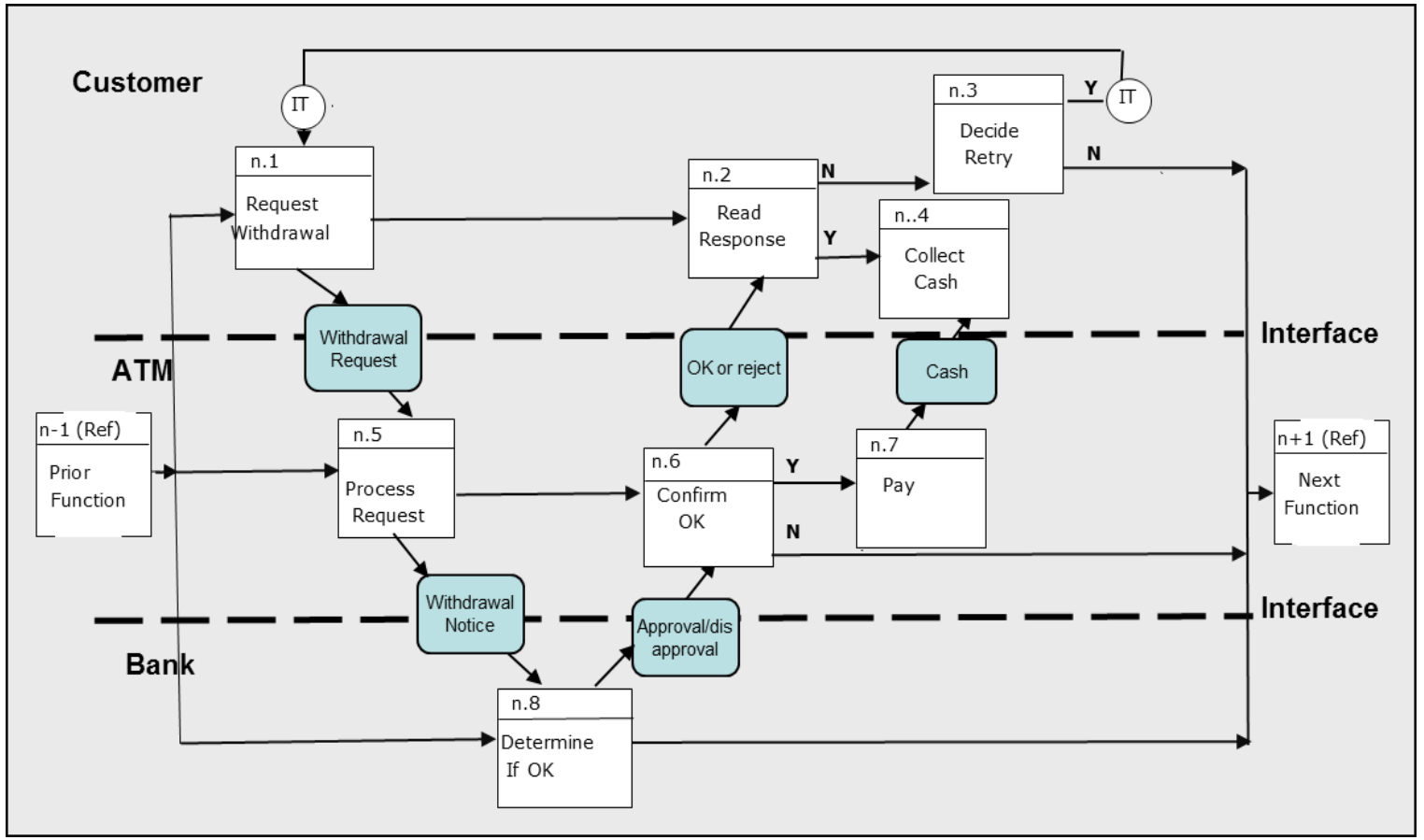


Physical Models

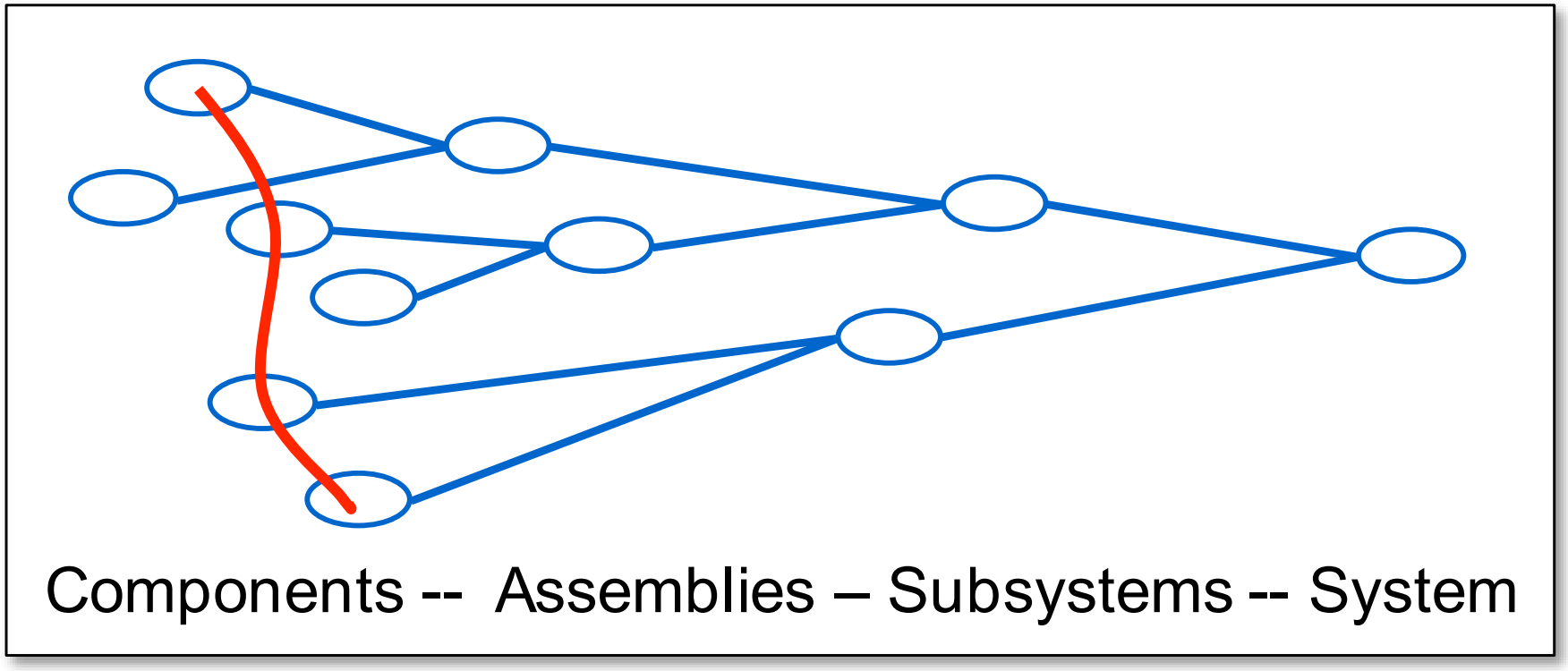


Functional Models

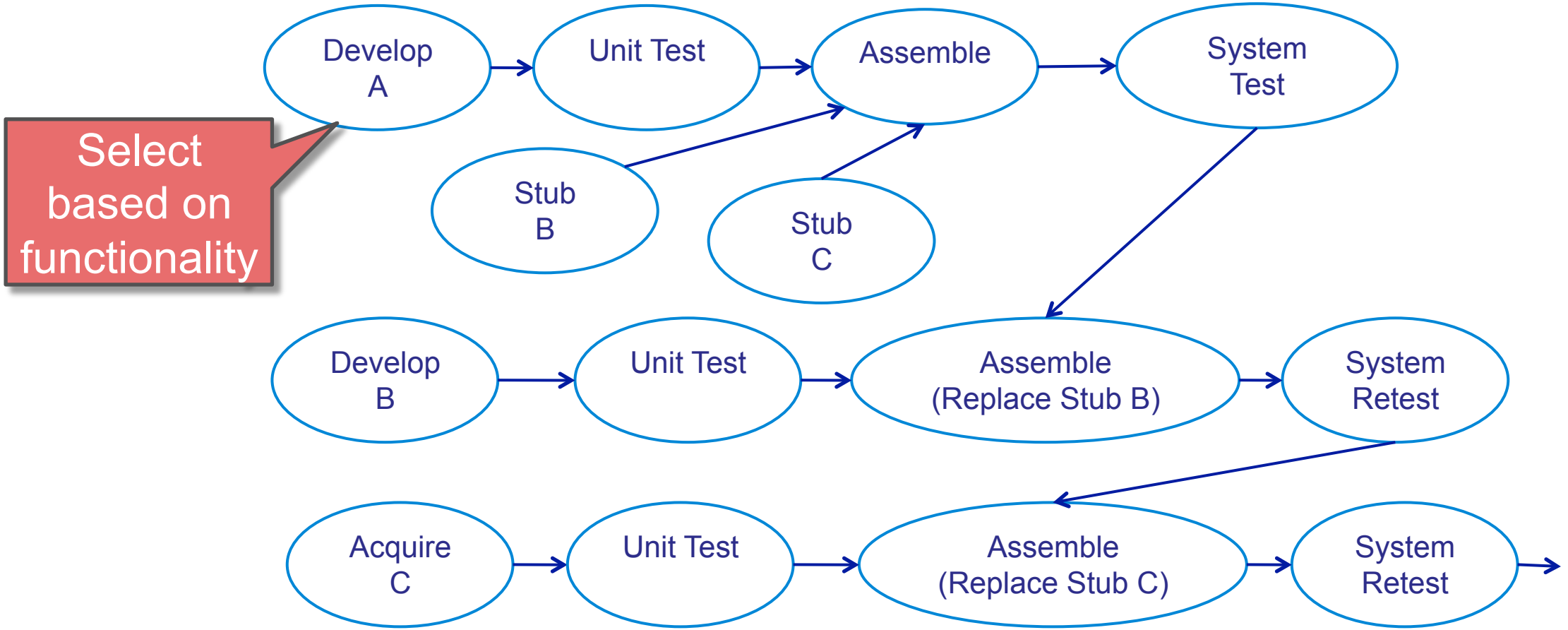
- Interactions across interfaces
- Can address threads – related actions independent of physical component structure
- Key to cross-branch issue



Focus on Functional Threads

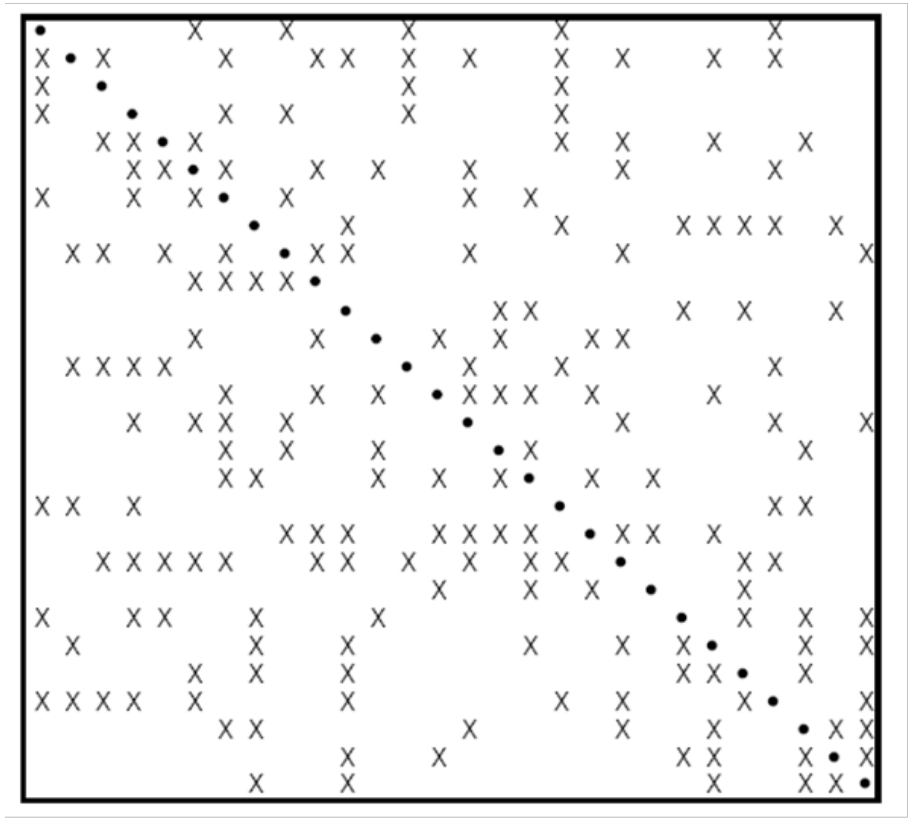


Top Down

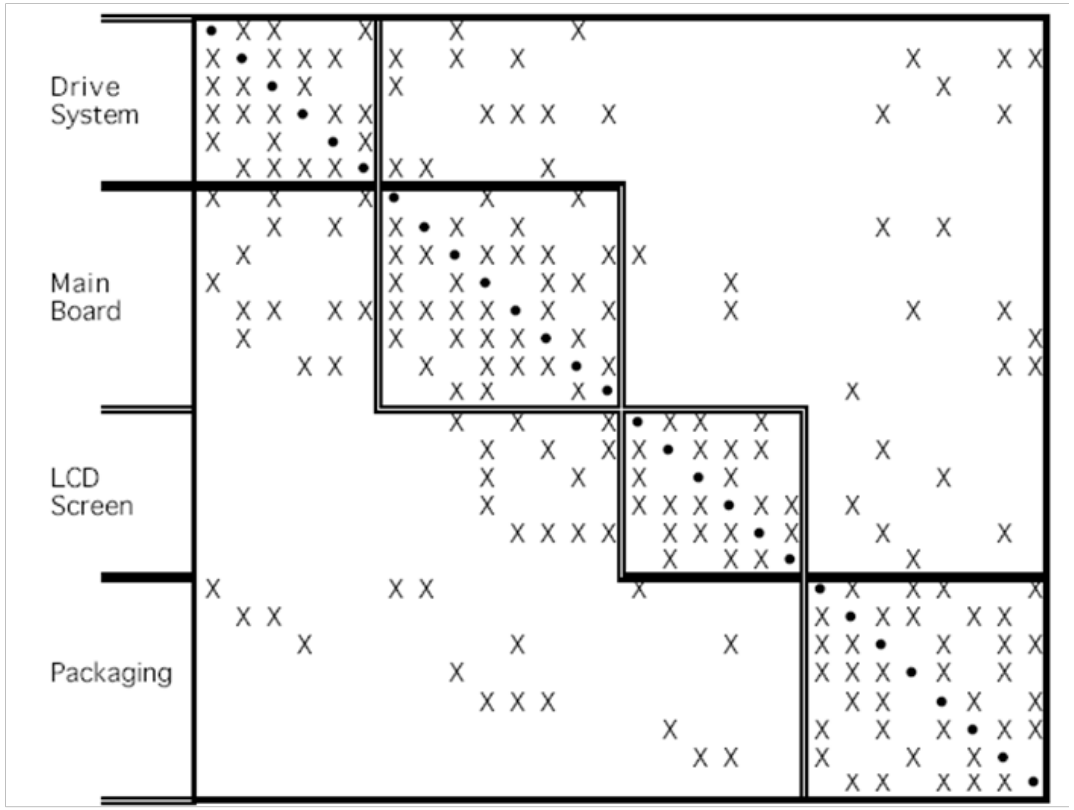

Develop A
Unit Test
Assemble
System Test
Select based on functionality
Stub B
Stub C
Develop B
Unit Test
Assemble (Replace Stub B)
System Retest
Acquire C
Unit Test
Assemble (Replace Stub C)
System Retest

Reorganization of Coupling Matrices

- Software modules
- x Interactions between modules



Laptop DSM



Laptop Reconfigured DSM

Criterion Driven

- Risk
- Schedule
- Other priorities

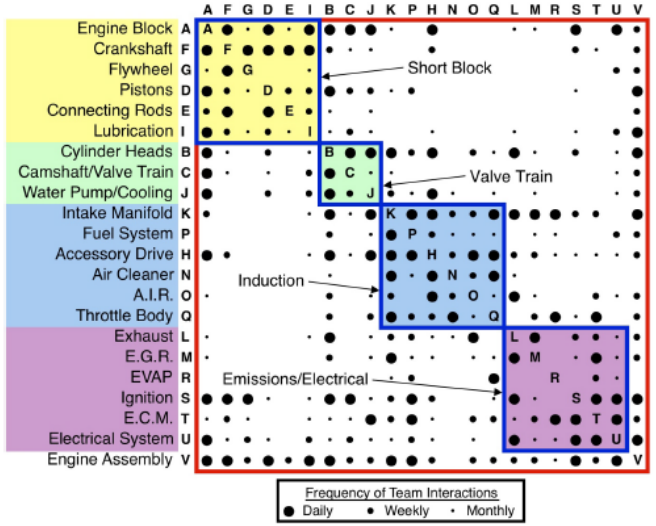
Application of Functional Integration

- Software – basically functional product
  - Agile and integration of features
- Services
- Processes
- Systems of Systems

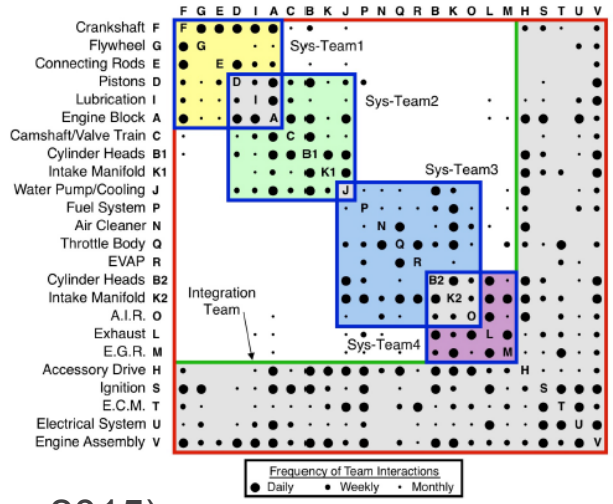
Organizational Architecture

Conway's law

"Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure."



DSM Analysis of Communications

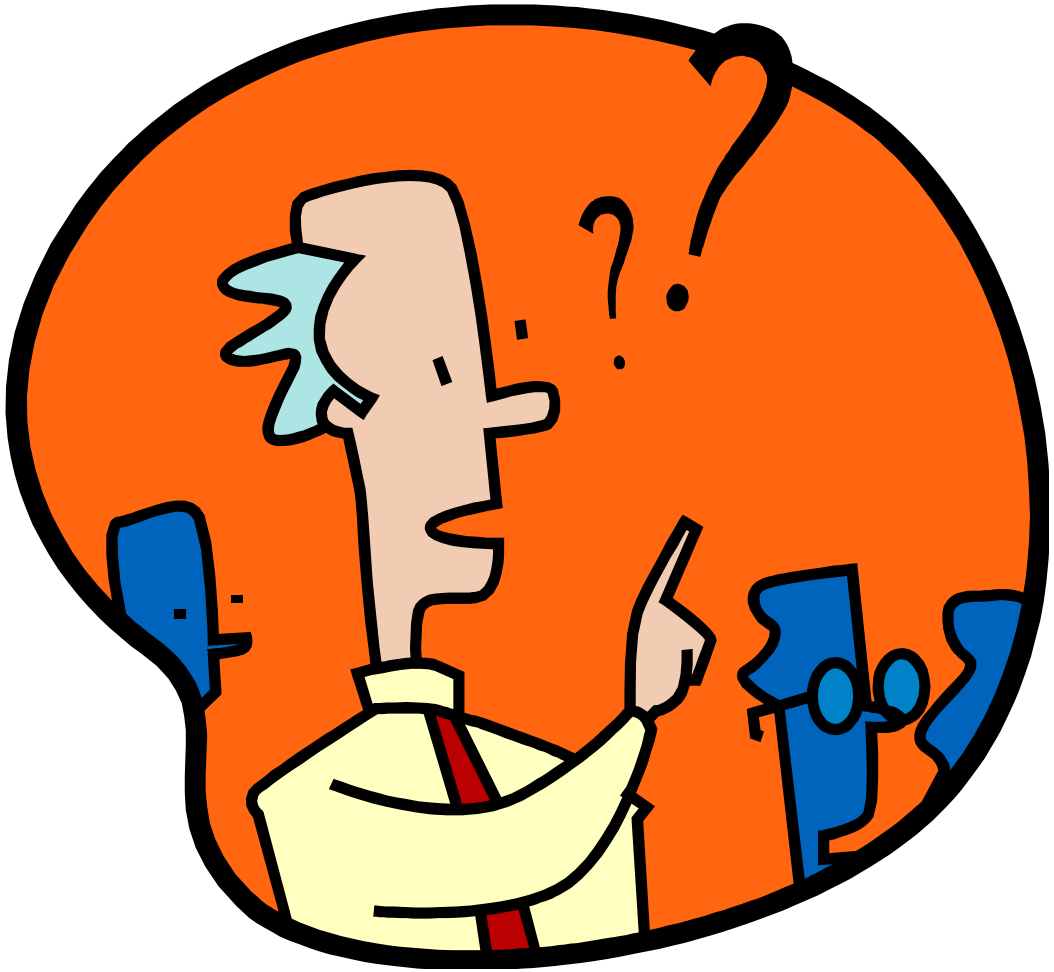


(Eppinger and Browning, 2015)

Summary

- Current thought focuses on the physical architecture and components
- Late discovery of cross-branch issues result
- Integration of other architectures can address these issues earlier in the project
  - Functional architecture
  - Organizational architecture
- Model-based integration implements this fix

Questions?



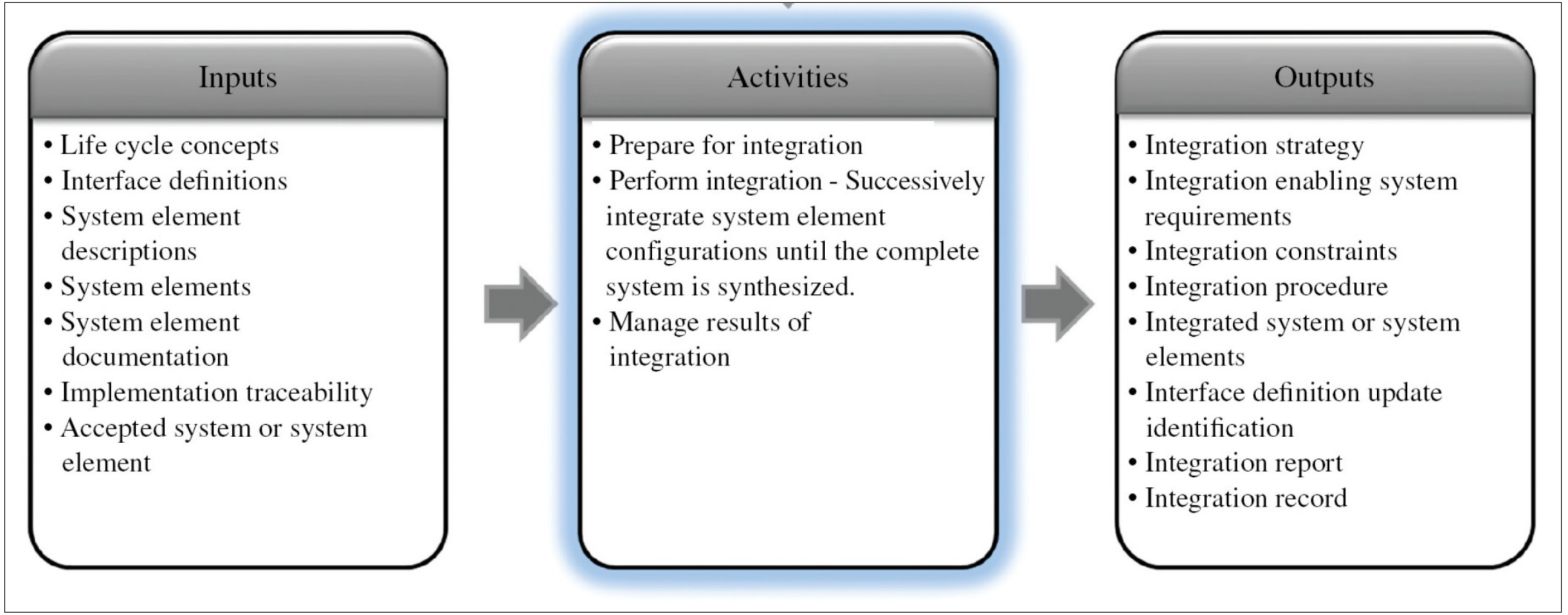
27th annual INCOSE international symposium

Adelaide, Australia

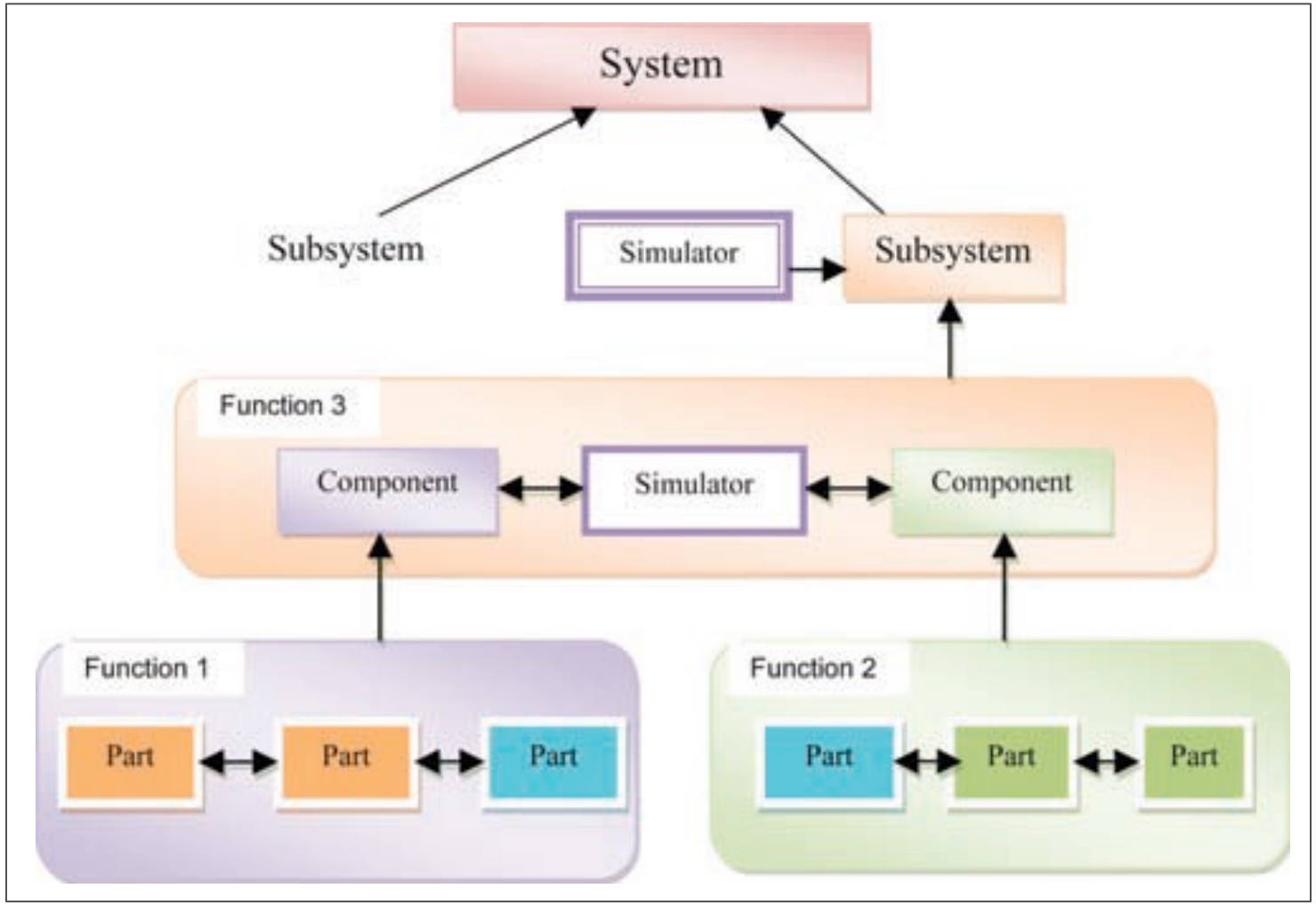
July 15 - 20, 2017

www.incose.org/symp2017

INCOSE Handbook Process View



Functional Integration



(Frank, Harel, and Orion, 2014)