Embedded Software Design, Development processes & Standards

L&T Technology Services

Mr. Bhargav N.

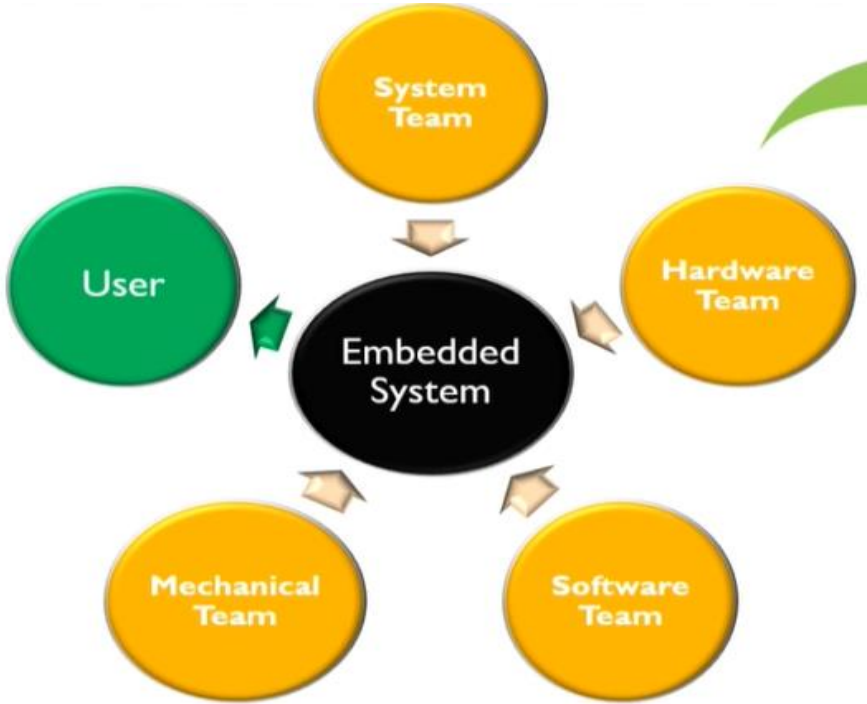
LTTS GLOBAL ENGINEERING ACADEMY

Agenda

- **Process**
- Embedded Product Development Phases
- Various Models for Embedded Product Development
- System Requirements
- System Design
- System Implementation
- System Testing

Process

Why Process ? What is Process ?



- Collaboration among Teams
- Requirements Sharing and Interpretation

PROCESS

Step by Step activity for

- **Collaboration**
- **Requirements sharing**
- **Interpretation**

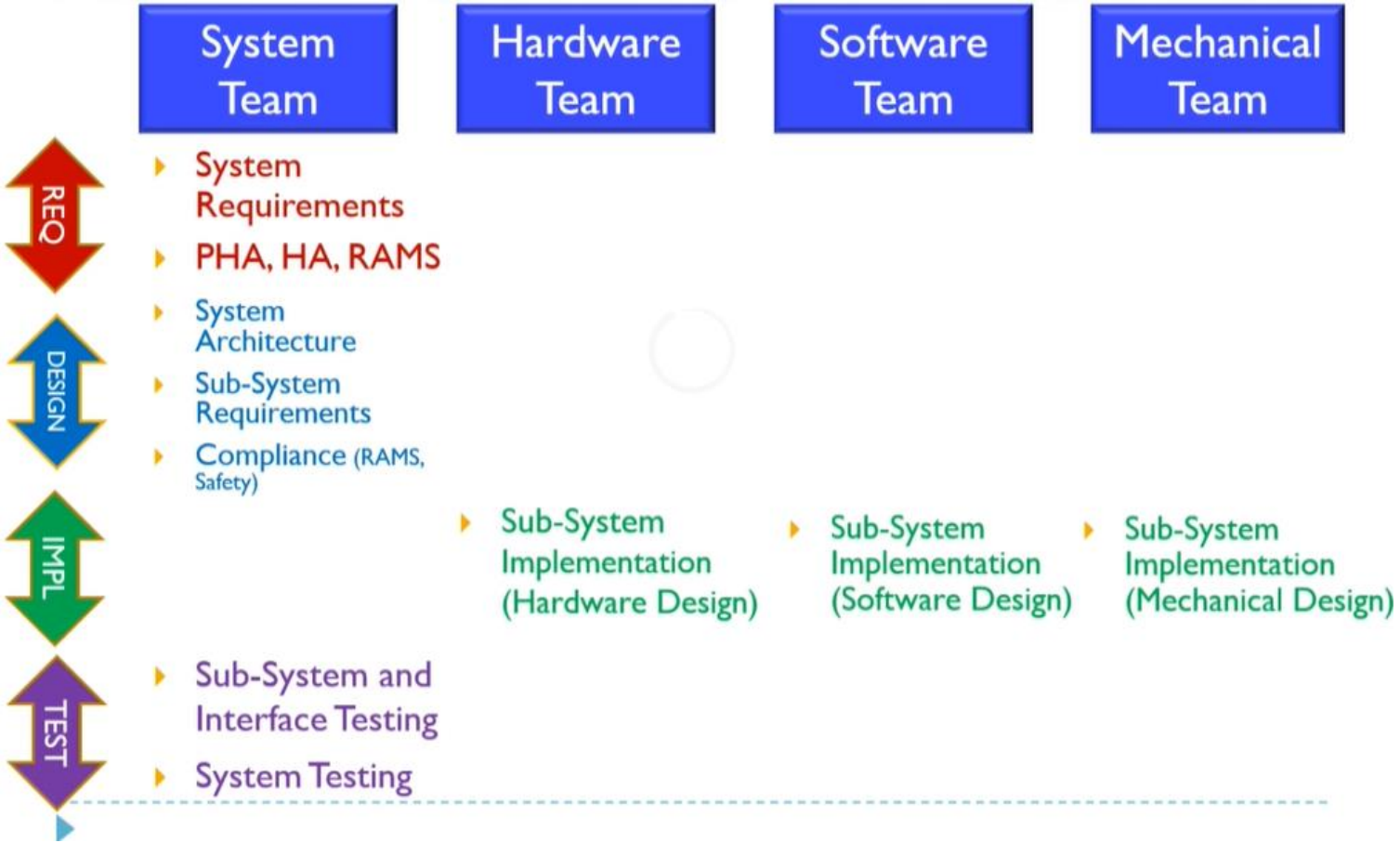
Project Success

Vocabulary

- Glossary of terms
  - SS - System Specification
  - TS - Technical Specification
  - SyRS - System Requirement Specification
  - SyDS - System Design Specification
  - SAD - System Architecture Document
  - SSS - Sub-System Specification
  - STSD - System Test Specification Document
  - PD - Preliminary Design
  - DD - Detailed Design
  - HRS - Hardware Requirement Specification
  - SRS - Software Requirement Specification
  - HDD - Hardware Design Document
  - SDD - Software Design Document
  - ATP - Acceptance Test Procedure

Embedded Product Development Phases

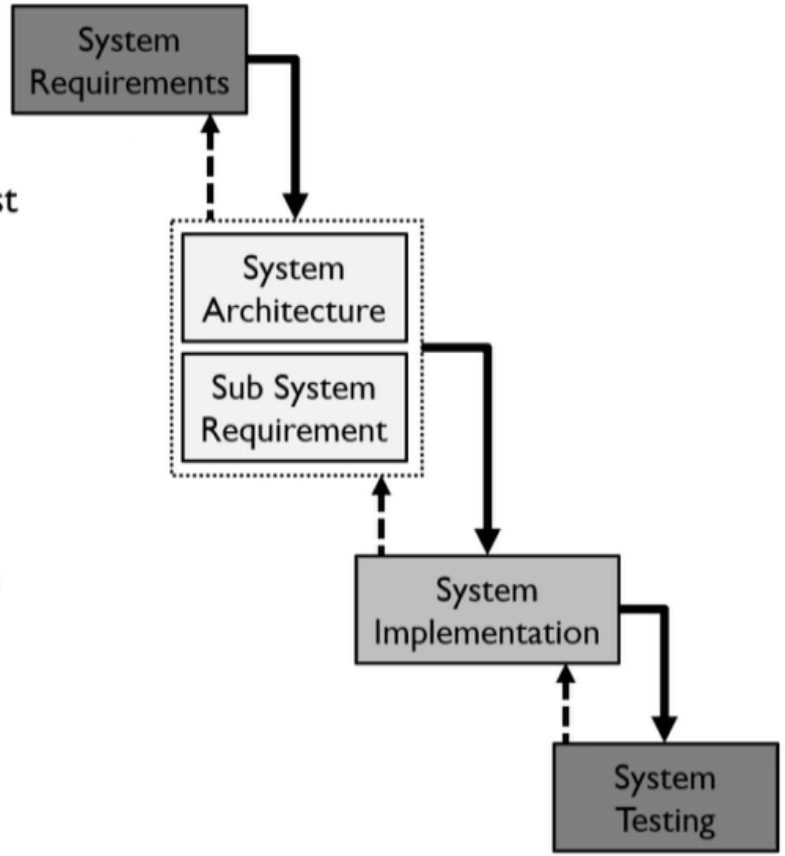
Embedded Product Development Phases



Various Models for Embedded Product Development

Waterfall Model

- Flow looks like Waterfall
- Requirements are clear and frozen
- Disadvantage is System Testing is done at the end and if any defects found, it will cost more ☹



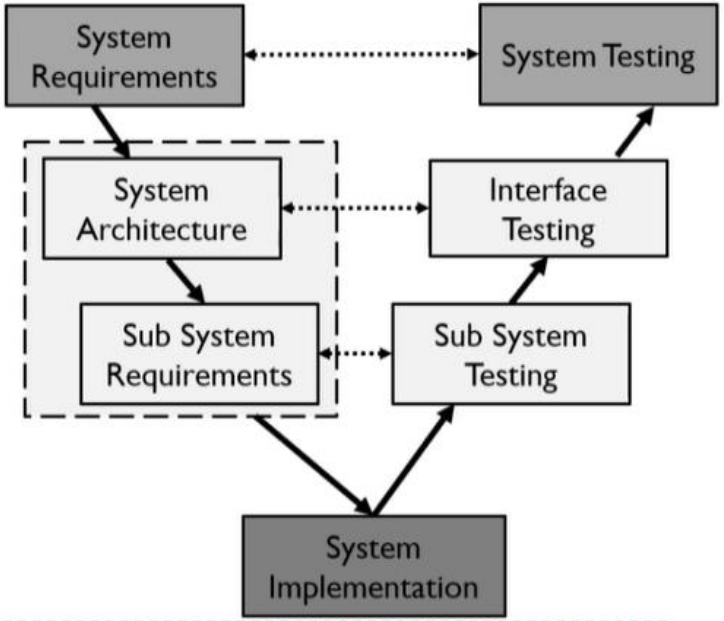
Activities:

- System Requirements
  - Functional, Interface, Mechanical requirements.
- System Design
  - Architecture into Subsystems
  - Allocation of System to Sub System Requirements
- System Implementation
  - Each Subsystem Design
  - Hardware, Software, Mechanical, Electrical
- System Testing
  - Subsystem Testing, Interface Testing, Functional Testing

V Model

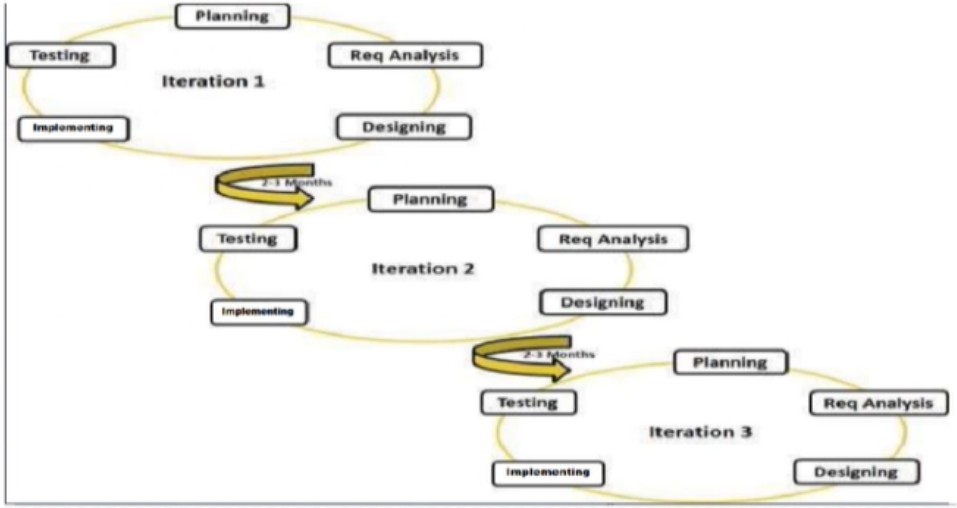
- Flow looks like english alphabet "V"
- Not all requirements of system are available and requirements are not matured
- Advantage: At every phase testability is checked and requirements are made clear in-terms of measurability, ambiguity and precise.

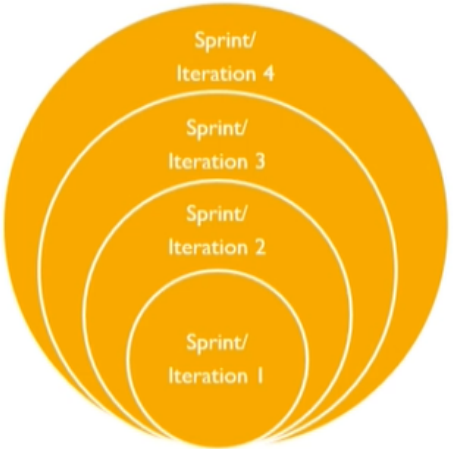
- This is one of the favorite Model used across industry
- Both Development and Testing Team was involved from the beginning
- Since defect is removed at every stage, it makes product very robust and reduces the rework effort.



AGILE Model

- Iterative and Incremental => AGILE
- Objective is "reduced cycle time to market"
- Not all requirements are matured, Not all requirements available
- Based on Business Value, requirements are prioritized and implemented as iterative
- Care has to be taken to make sure the designed platform or framework supports future requirements else this model will fail and everything need to be started afresh





Standards and Tools

- Aerospace Standards
  - ARP 4754, 4761, DO-178B, DO254
  - DOD 2167A
- Rail Standards
  - EN 50126, 50128, 50129
  - IEC 61508
- Automotive
  - AUTOSAR, ASIL-D
  - ASPICE, ISO26262, ISO9001
- Configuration Management Tools
  - Clear Case
  - SVN
- Requirements Management and Modelling Tools
  - DOORS
  - Rhapsody, ARTiSAN
  - MagicDraw
- PLM/PMP Tools
  - Teams center
  - MPP
  - JIRA

Aerospace Standards

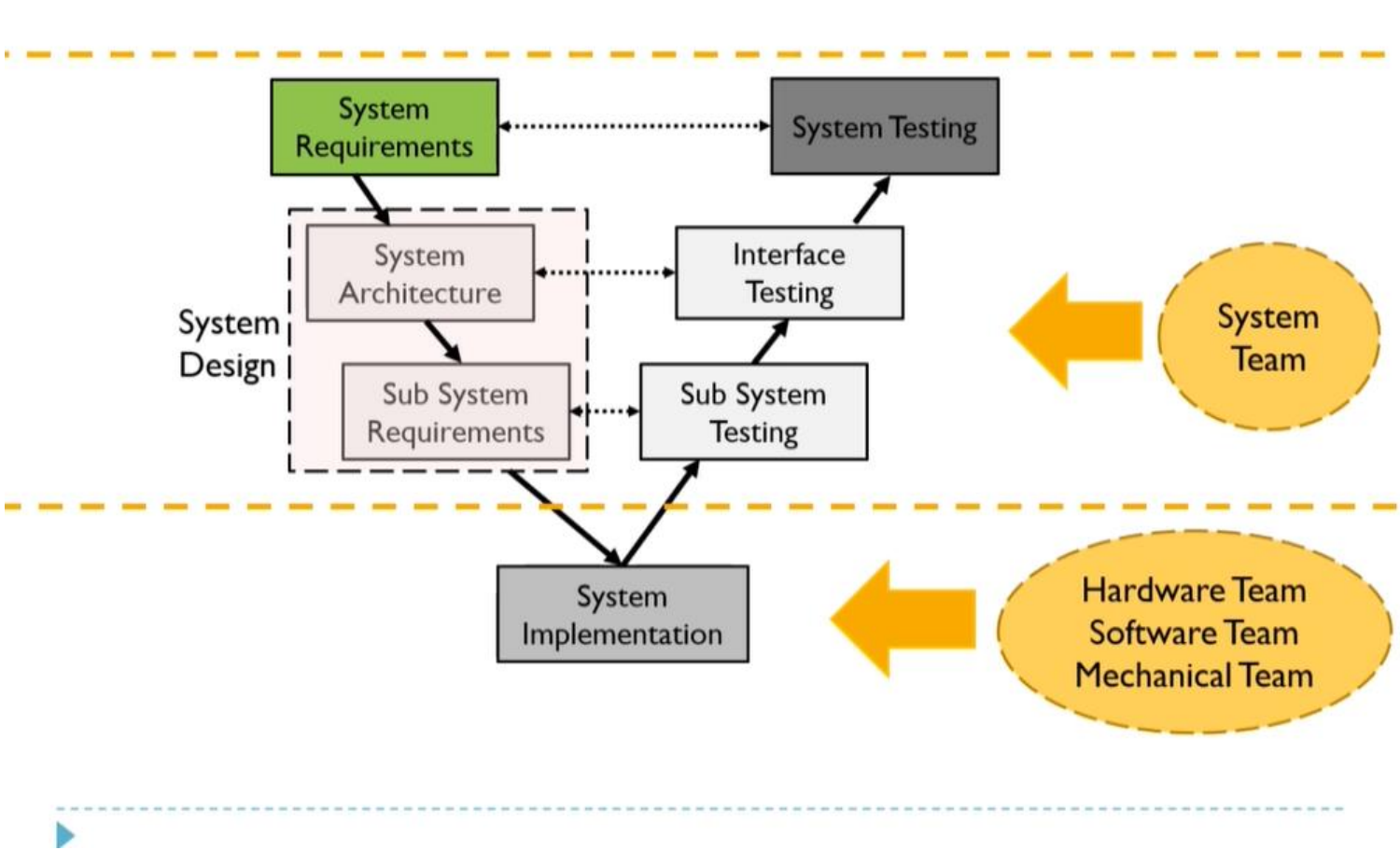
- Aerospace Standards
  - ARP 4754 - Certification Considerations for Highly-Integrated Or Complex Aircraft Systems
  - ARP 4761 - Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment
  - DO-178B - Software Considerations in Airborne Systems and Equipment Certification
  - RTCA/DO-297- Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations
  - DOD 2167A - Defense Systems Software Development

Rail Standards

- Rail Standards
  - EN 50126 (IEC 62278) – Railway Applications Specification and Demonstration of RAMS
  - EN 50128 (IEC 62279) – Software for railway control and protection systems
  - EN 50129 (IEC 62425) –Safety related electronic systems for signaling
  - IEC 61508 - Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems
  - EN 15380: Classification system for railway vehicles -Part5:System Breakdown Structure (SBS)
  - ISO/IEC 15288:2008 - System life cycle processes
  - 1220-2005 - IEEE Standard for Application and Management of the Systems Engineering Process
  - ISO/IEC 12207:2008 - Software life cycle processes

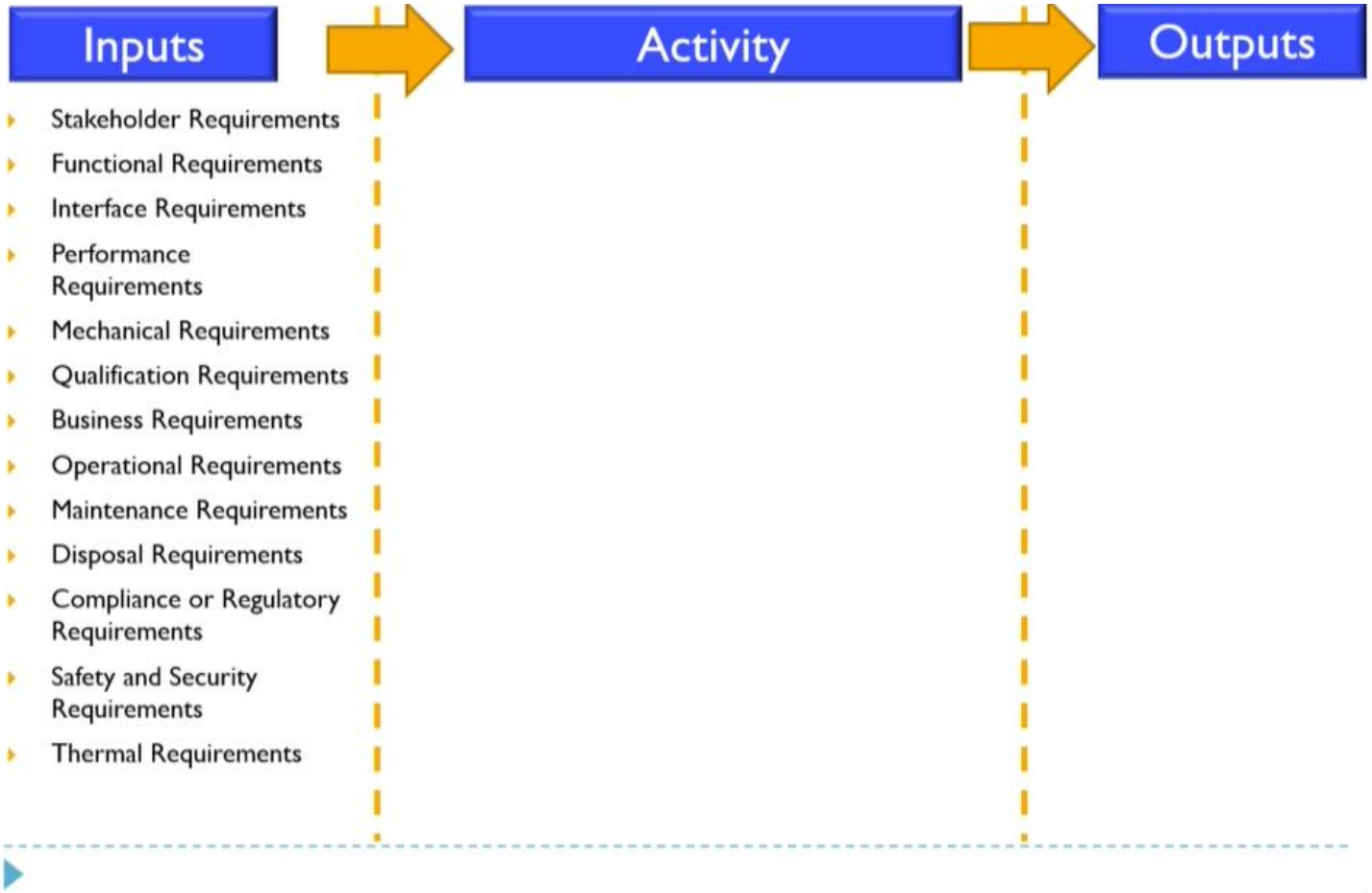
System Requirements

V Model

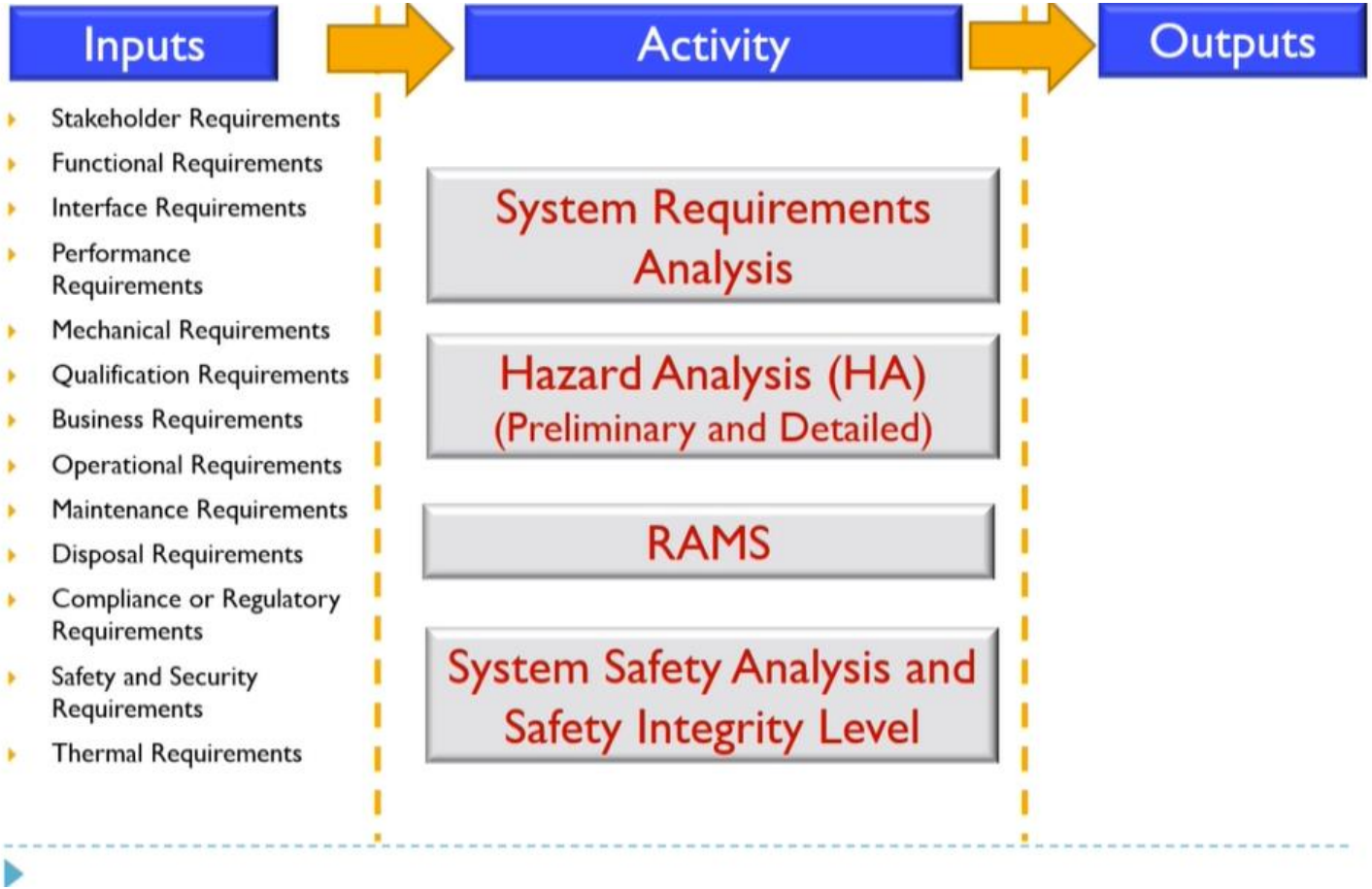

System Requirements
System Testing
System Architecture
Interface Testing
System Design
Sub System Requirements
Sub System Testing
System Team
System Implementation
Hardware Team
Software Team
Mechanical Team

Requirements

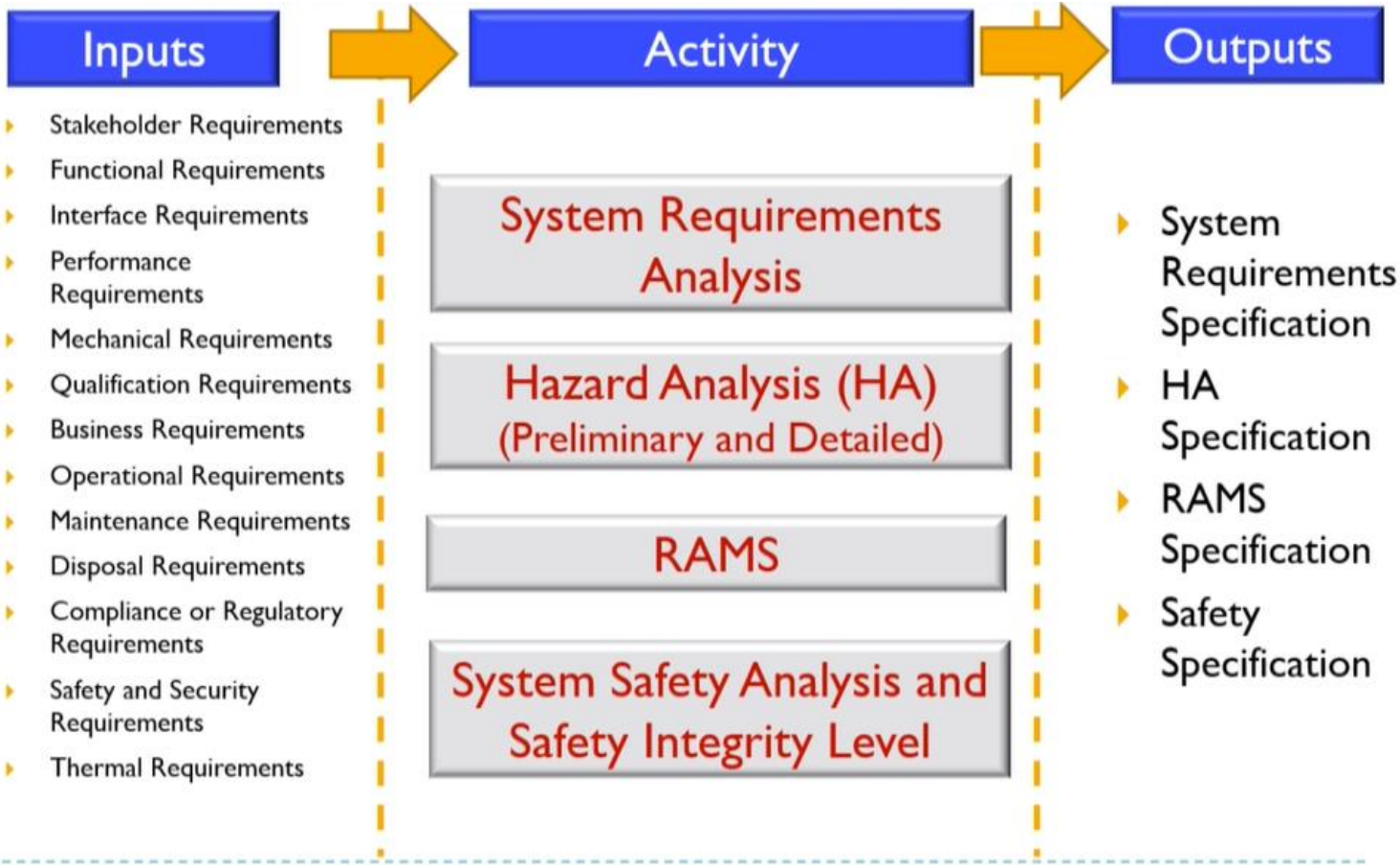
Input: System Requirements



Activity: System Requirements

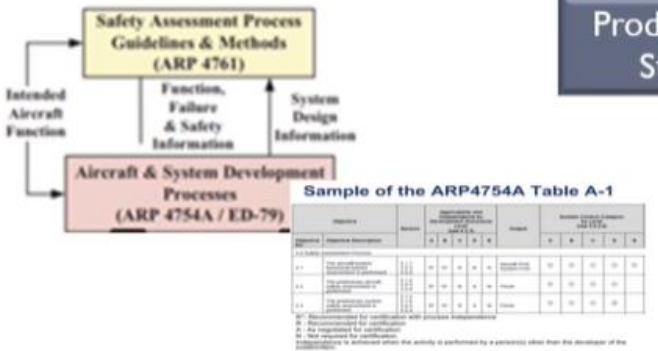


Output: System Requirements



Standards and Tools

- Domain Standards (Aerospace, Rail, Automotive)
- Requirement Standards and Templates
- Representation (SysML, Textual)
- Tools: Requirement Management Tools, SysML Tools, FMEA Tools, Reliability Tools



Product Break Down Structure (PBS)

System Break Down Structure (SBS)

Functional Break Down Structure (FBS)

ISO 26262

ASIL

AUTOSAR

Standards and Tools

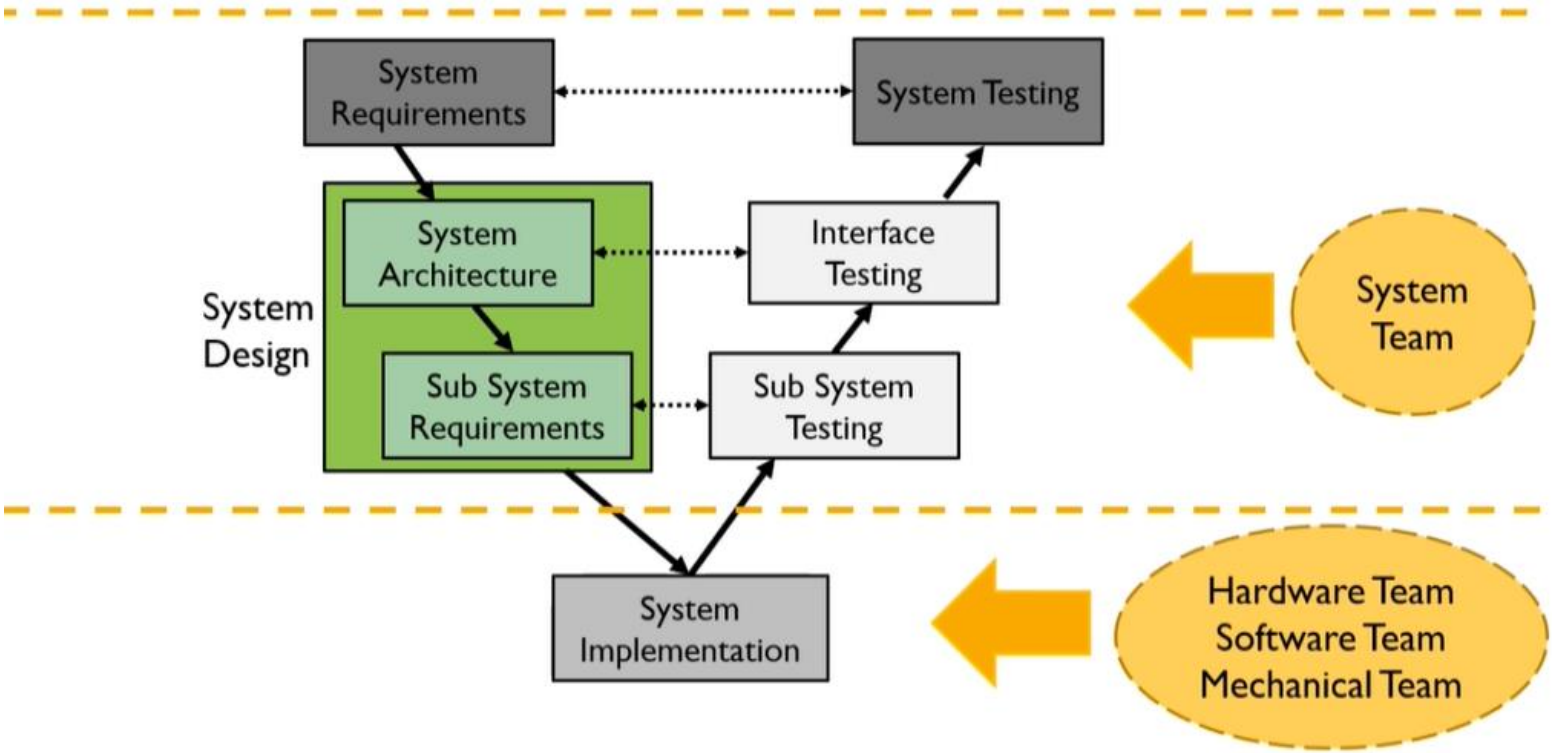
- Requirements Management Tools
  - DOORS
  - LDRA
  - SCADE
  - Casespec
- Modelling Tools
  - Rhapsody, ARTiSAN, MagicDraw, Visual Paradigm, SCADE, Enterprise Architect
- RAMS Tools
  - ReliaSoft, Ramp Software, Isograph

Automotive Standards

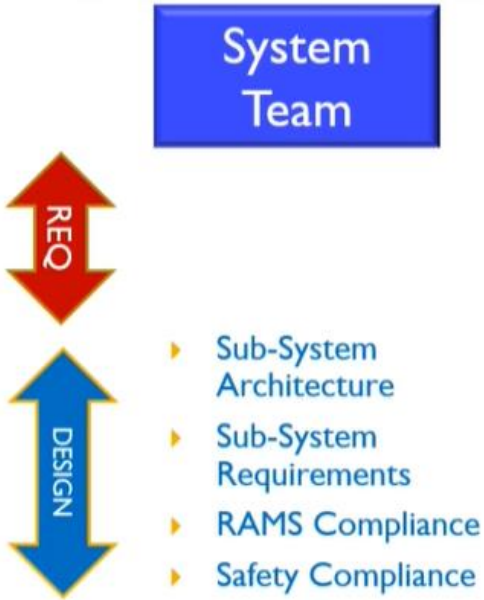
- Automotive Standards
  - AUTOSAR – Automotive Open System Architecture
  - ASIL - Automotive Safety Integrity Level
  - ASPICE – Automotive SPICE Process Development Model
  - ISO26262 - Functional safety standard for electrical and electronic (E/E) systems (Derived from IEC 61508)
  - ISO9001
  - ISO/IEC 15288:2008 - System life cycle processes
  - ISO/IEC 15504 – Process assessment, also termed Software Process Improvement and Capability Determination (SPICE)

System Design

V Model

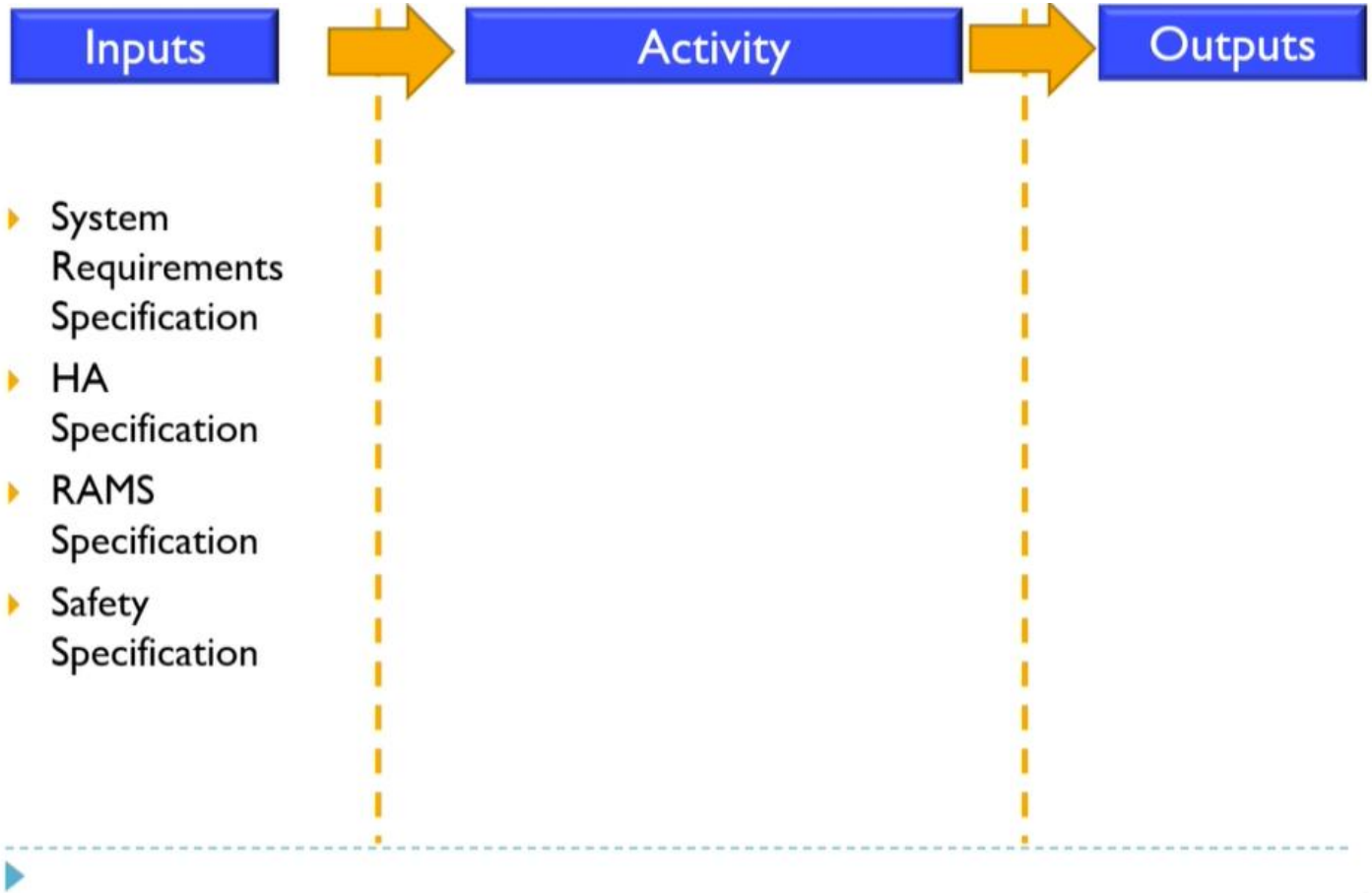

System Requirements
System Testing
System Architecture
Interface Testing
System Design
Sub System Requirements
Sub System Testing
System Team
System Implementation
Hardware Team
Software Team
Mechanical Team

System Design

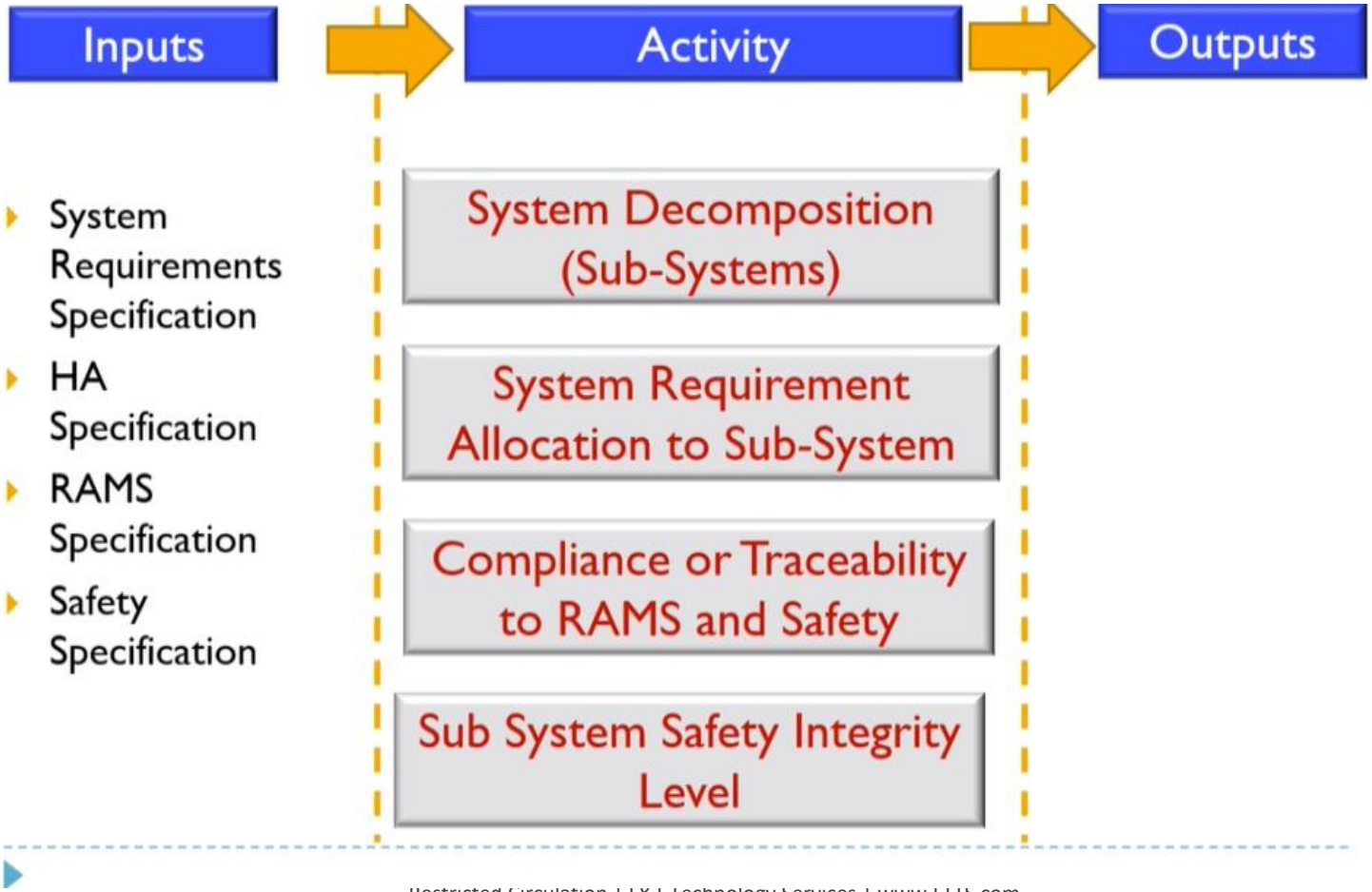


- Sub-System Architecture
- Sub-System Requirements
- RAMS Compliance
- Safety Compliance

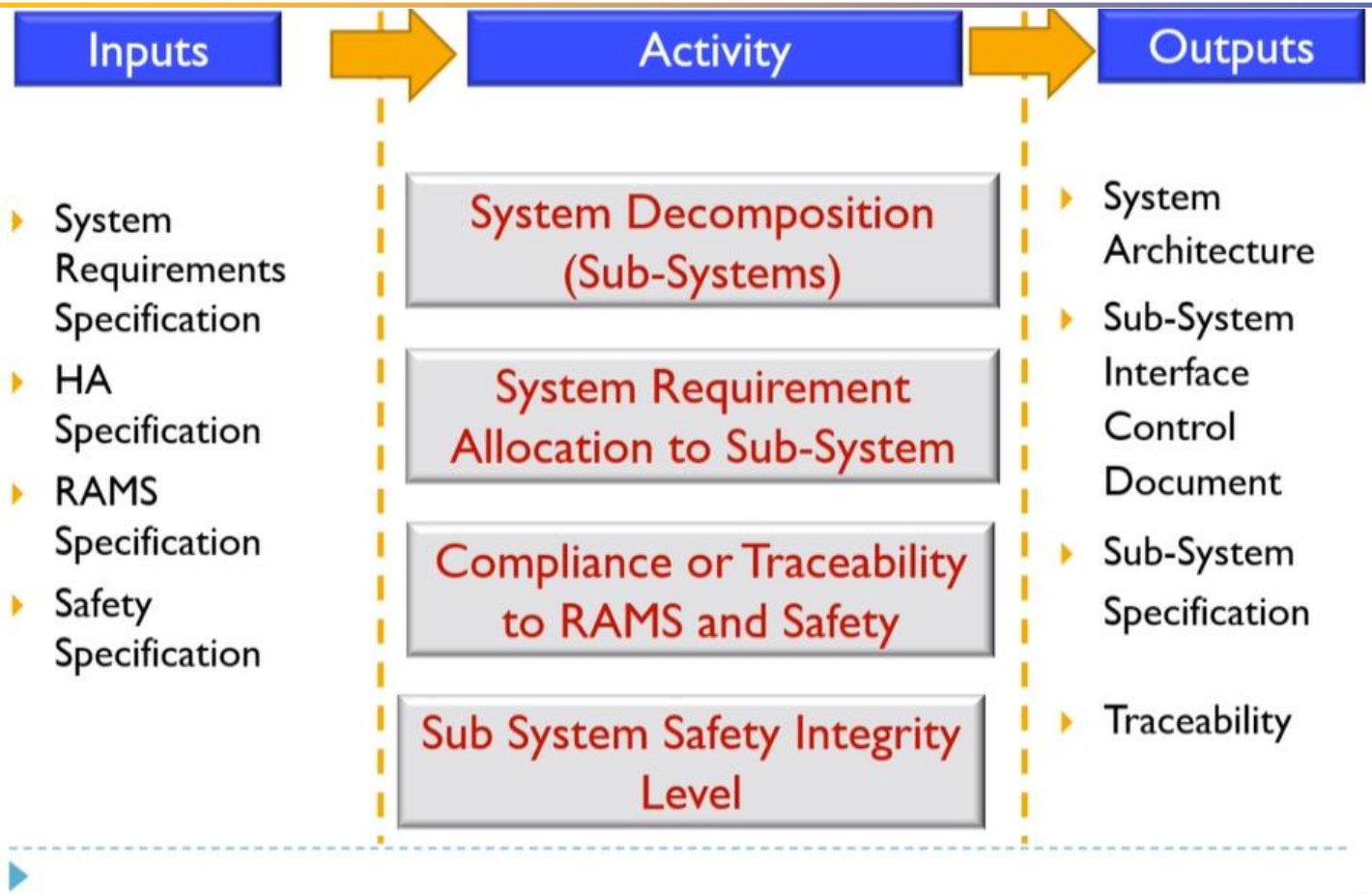
Input: System Design



Activity: System Design

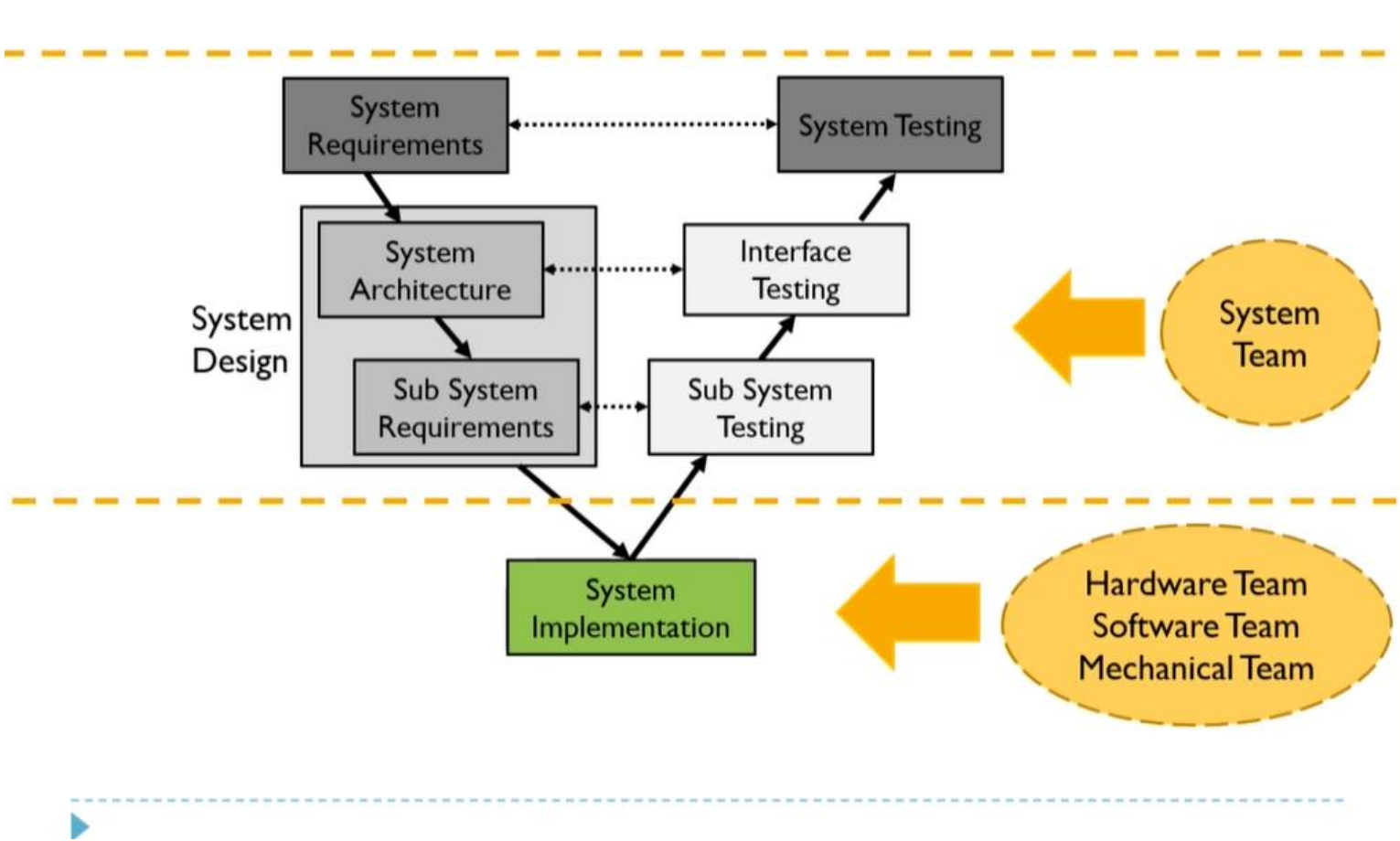


Output: System Design



System Implementation

V Model


System Requirements
System Testing
System Architecture
Interface Testing
System Design
Sub System Requirements
Sub System Testing
System Team
System Implementation
Hardware Team
Software Team
Mechanical Team

System Implementation

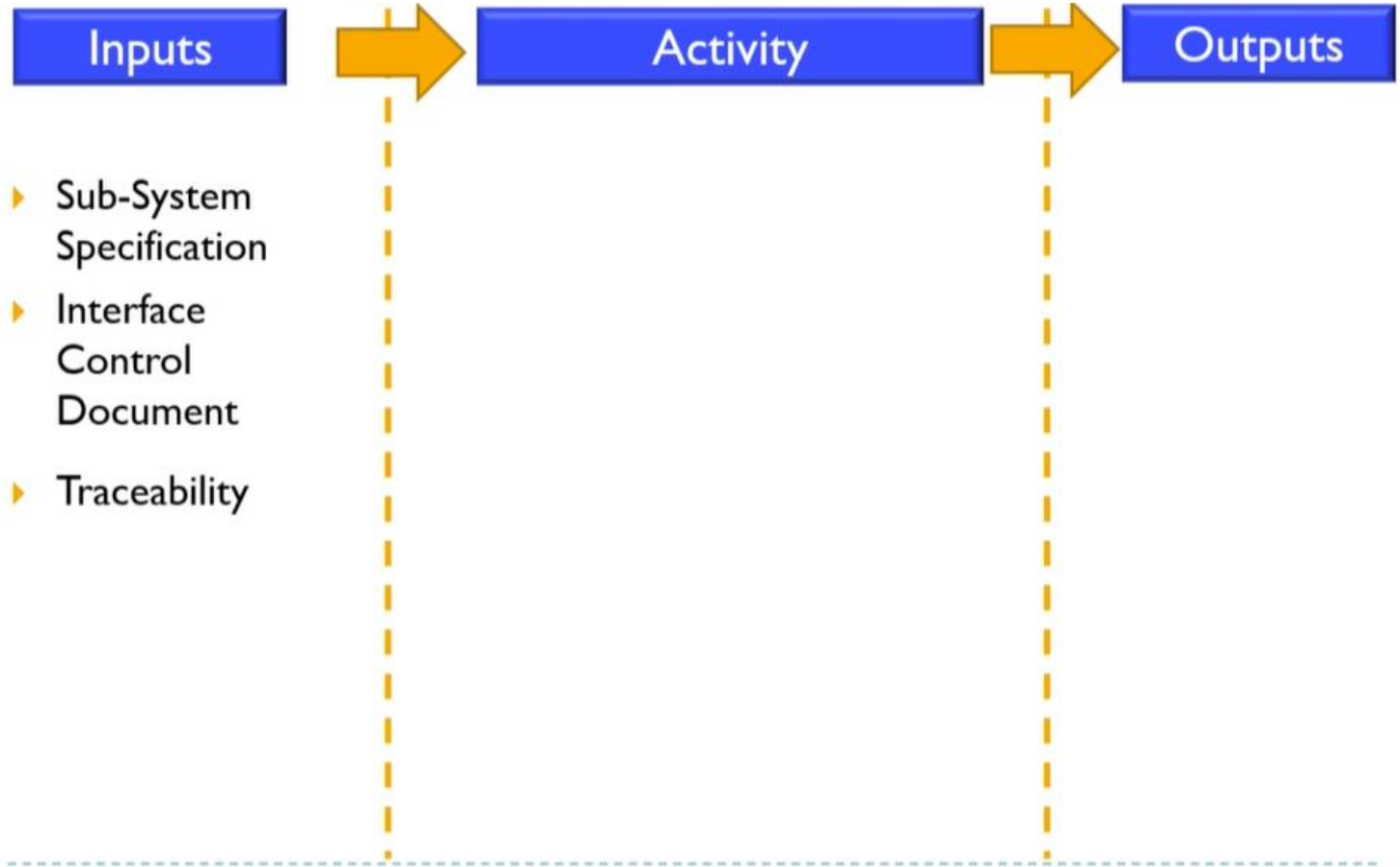
Hardware Team

Software Team

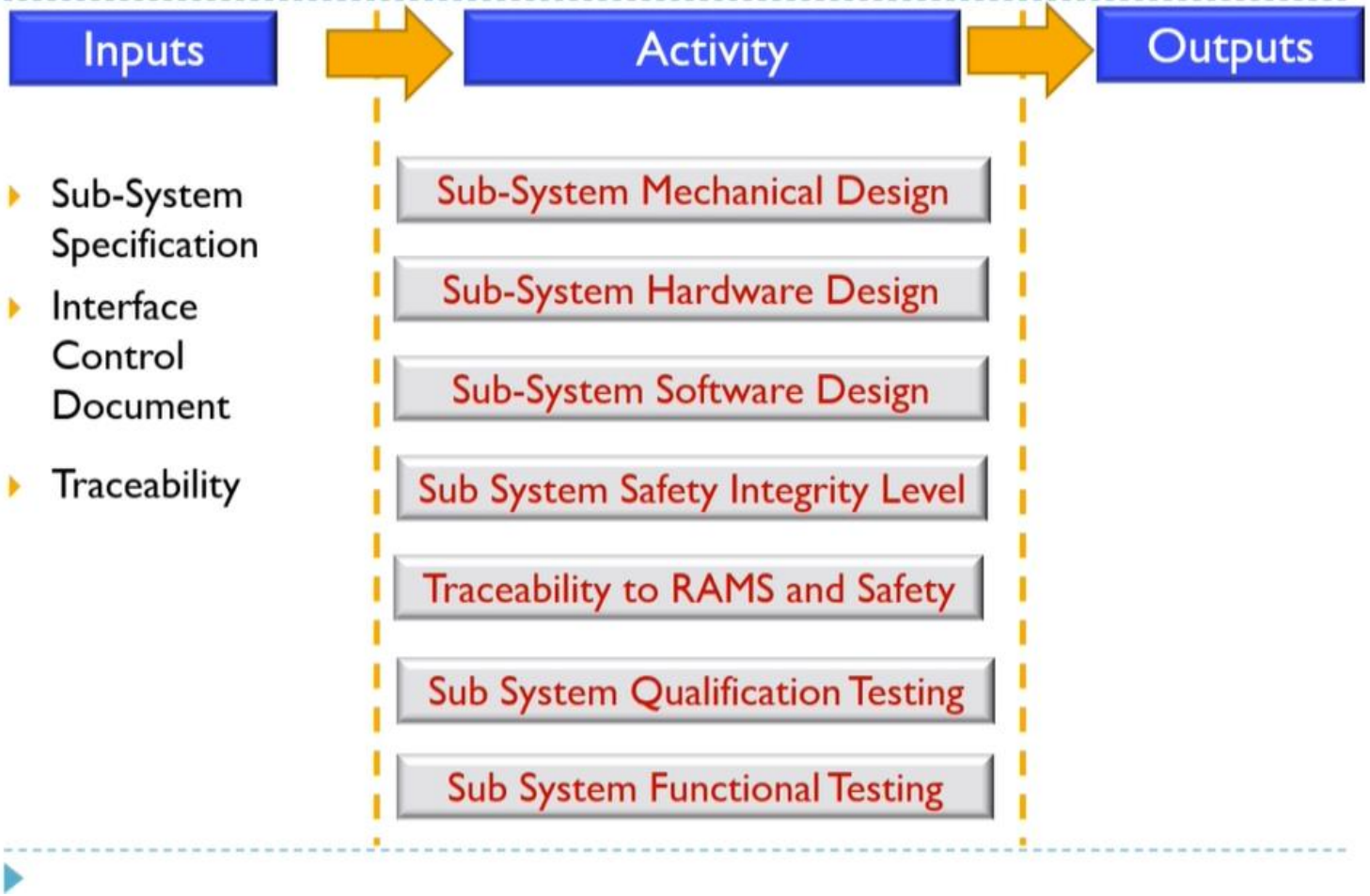
Mechanical Team

- Sub-System Implementation (Hardware Design)
- Sub-System Implementation (Software Design)
- Sub-System Implementation (Mechanical Design)

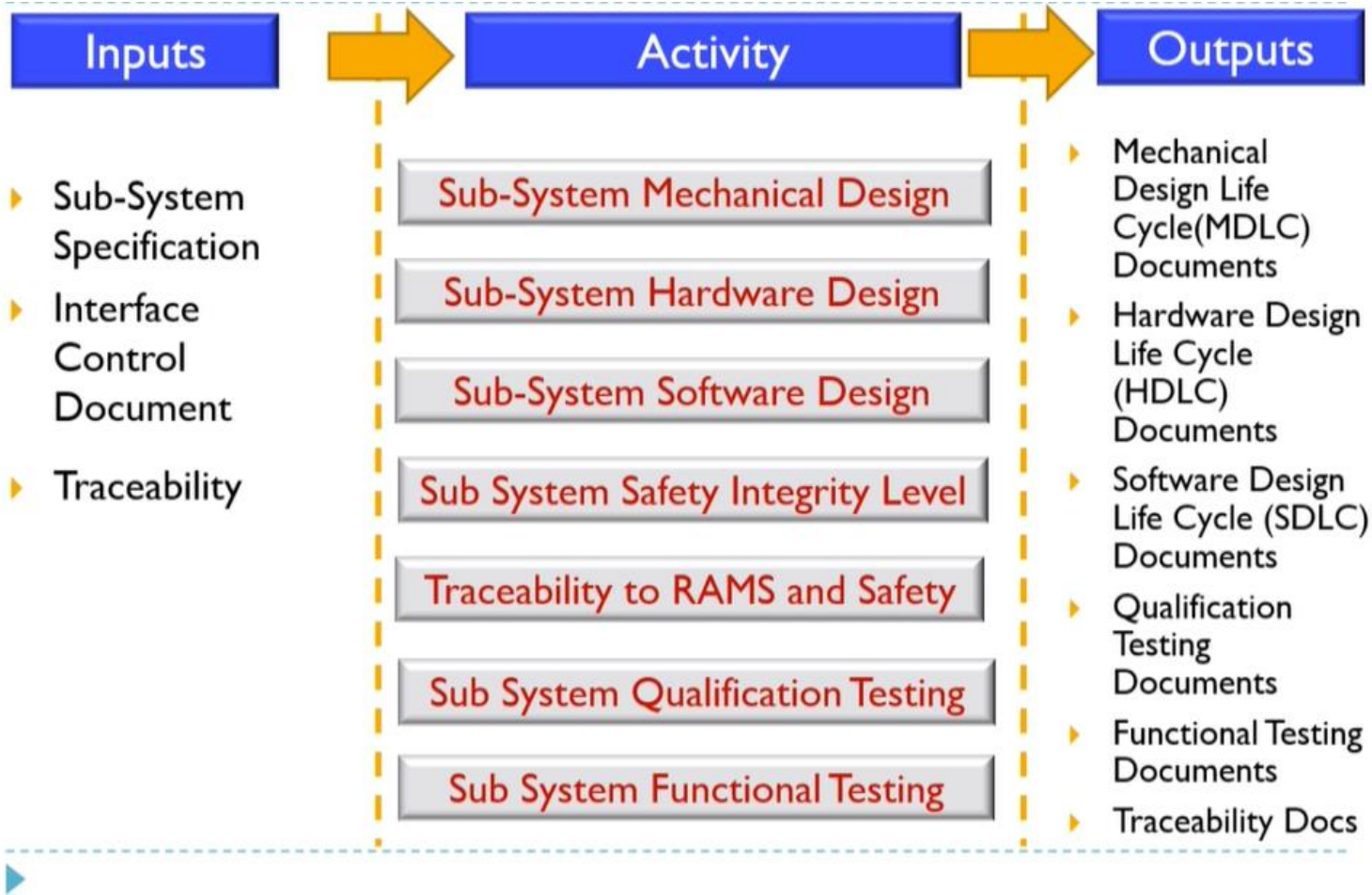
Input: System Implementation



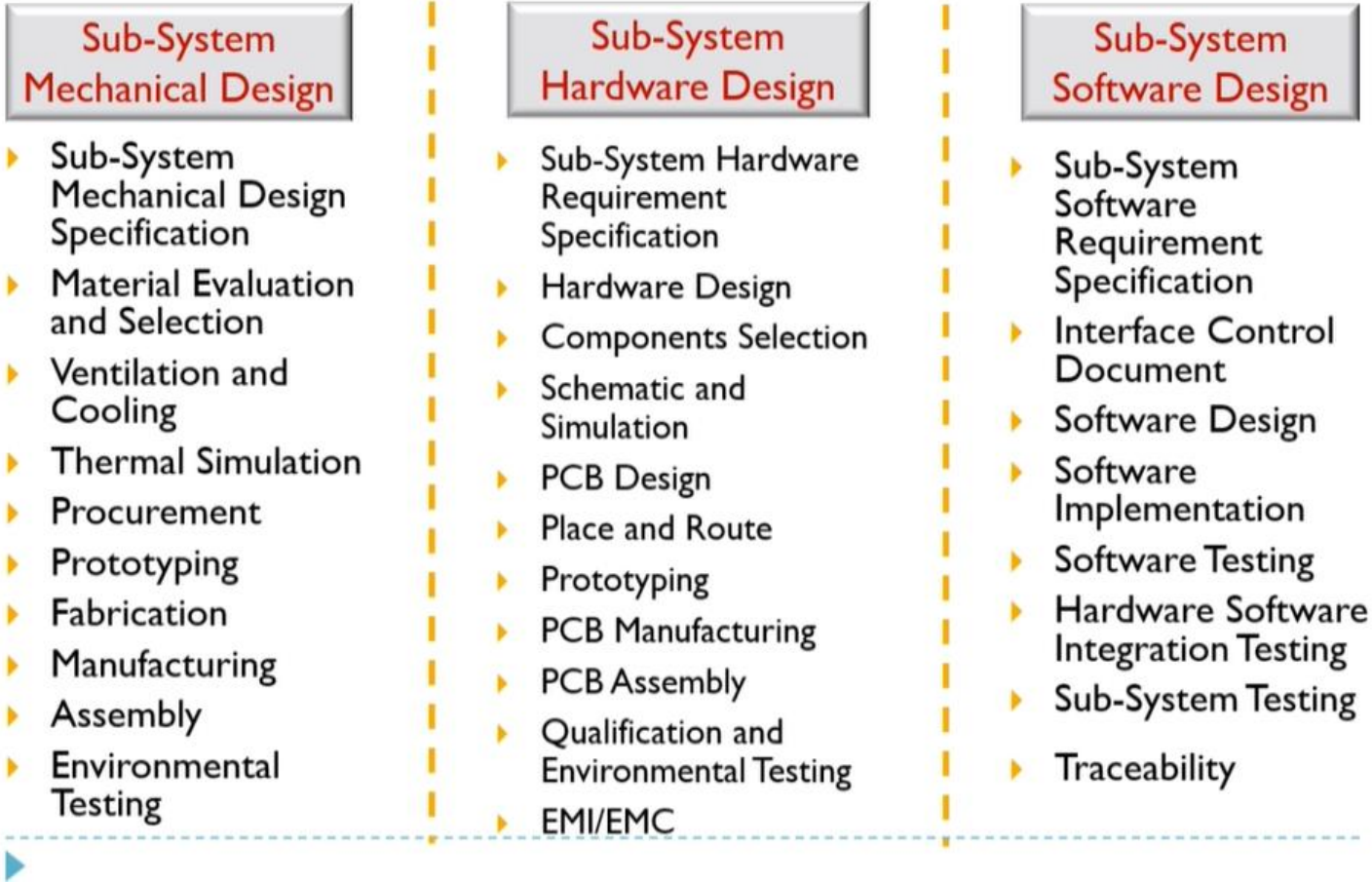
Activity: System Implementation



Output: System Implementation

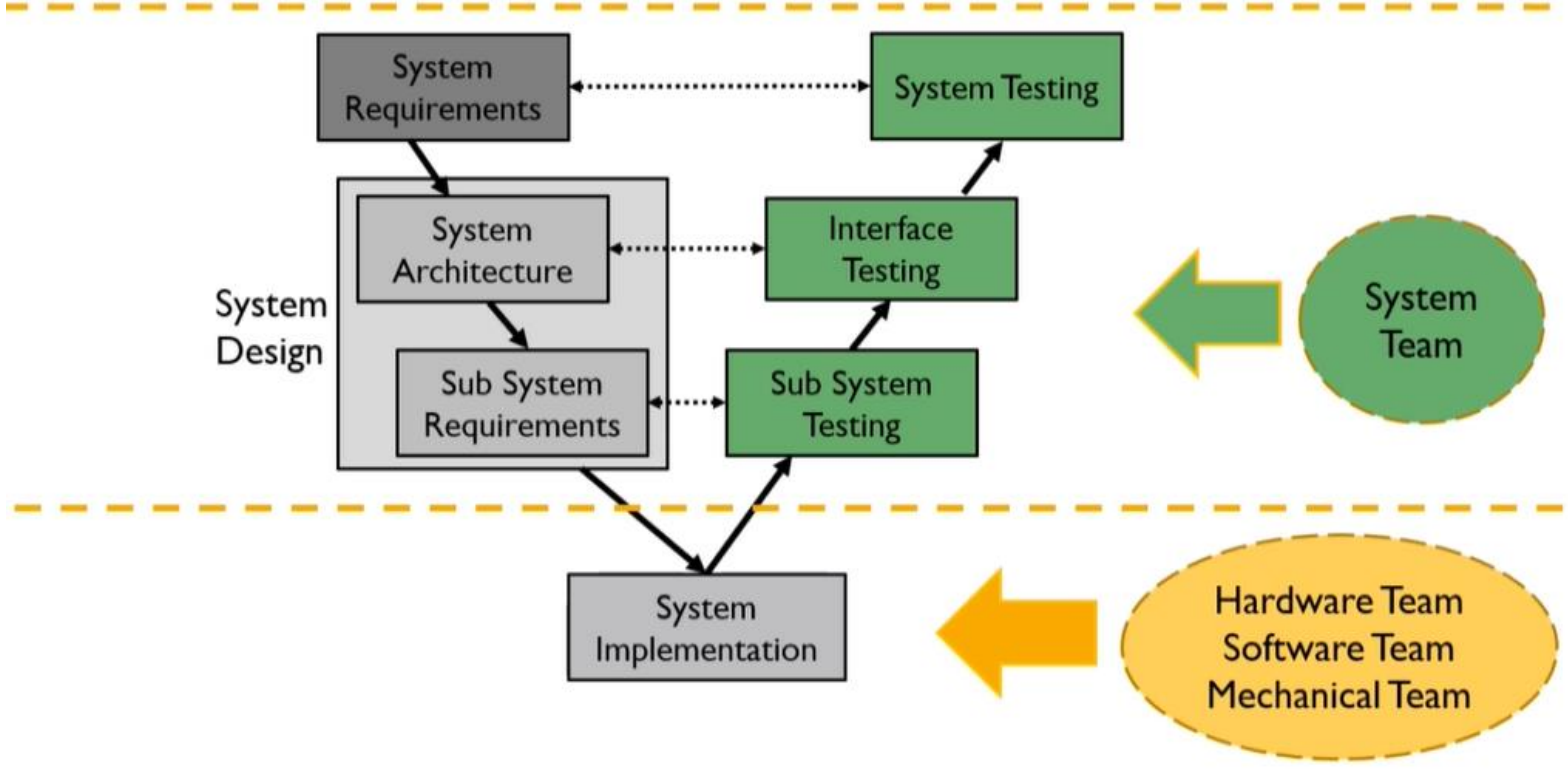


Activity: System Implementation

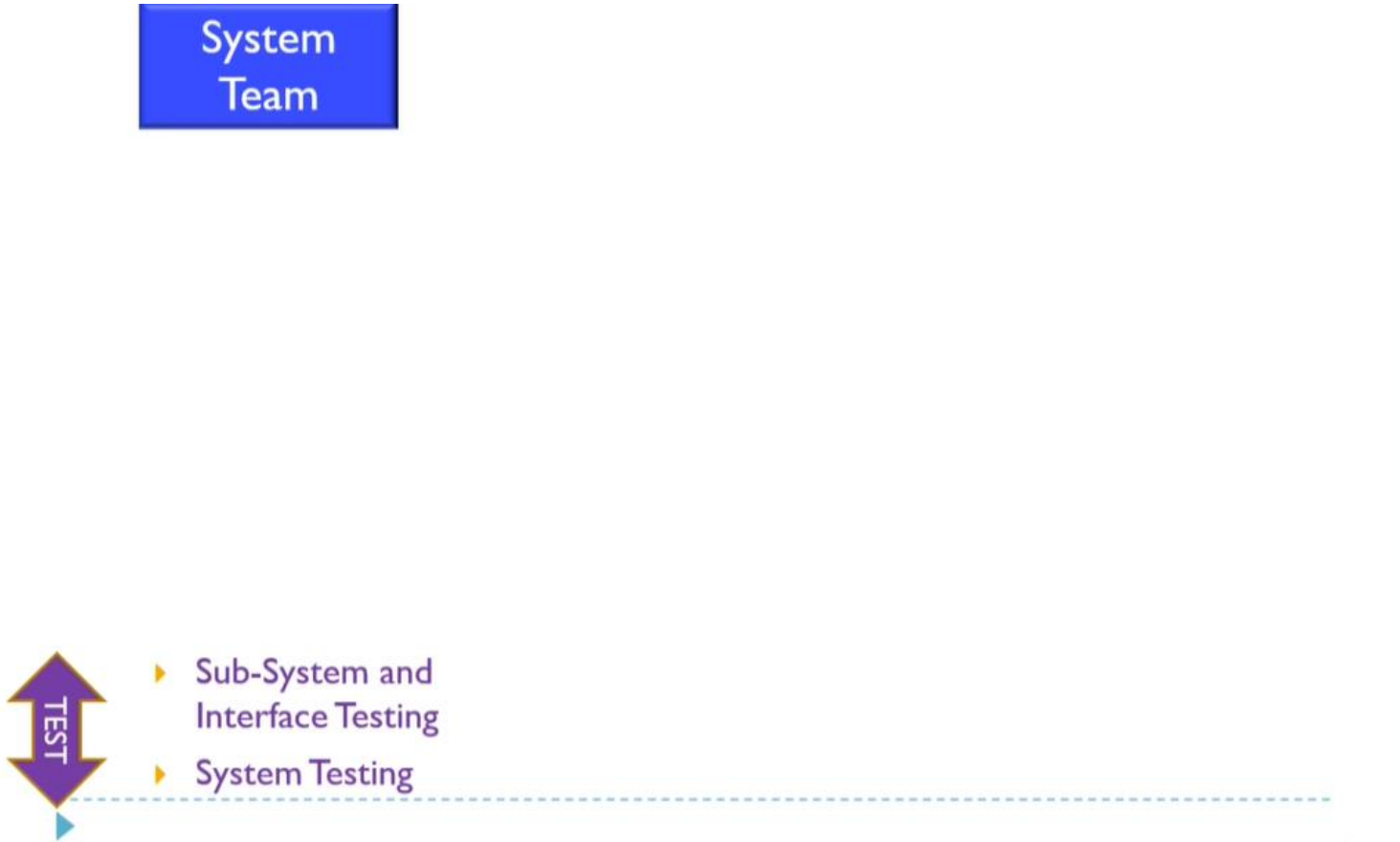


System Testing

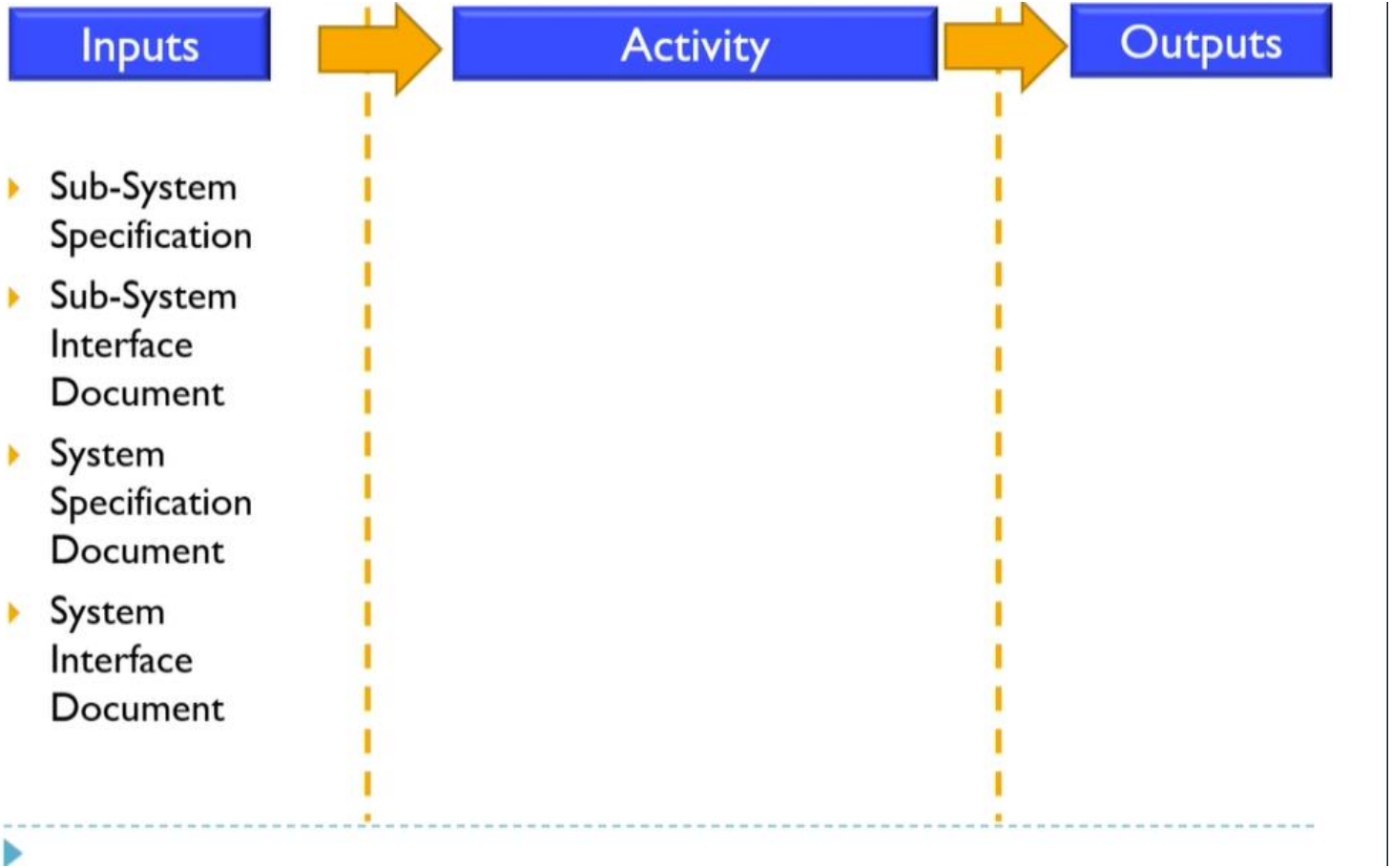
V Model


System Requirements
System Testing
System Architecture
Interface Testing
System Design
Sub System Requirements
Sub System Testing
System Team
System Implementation
Hardware Team
Software Team
Mechanical Team

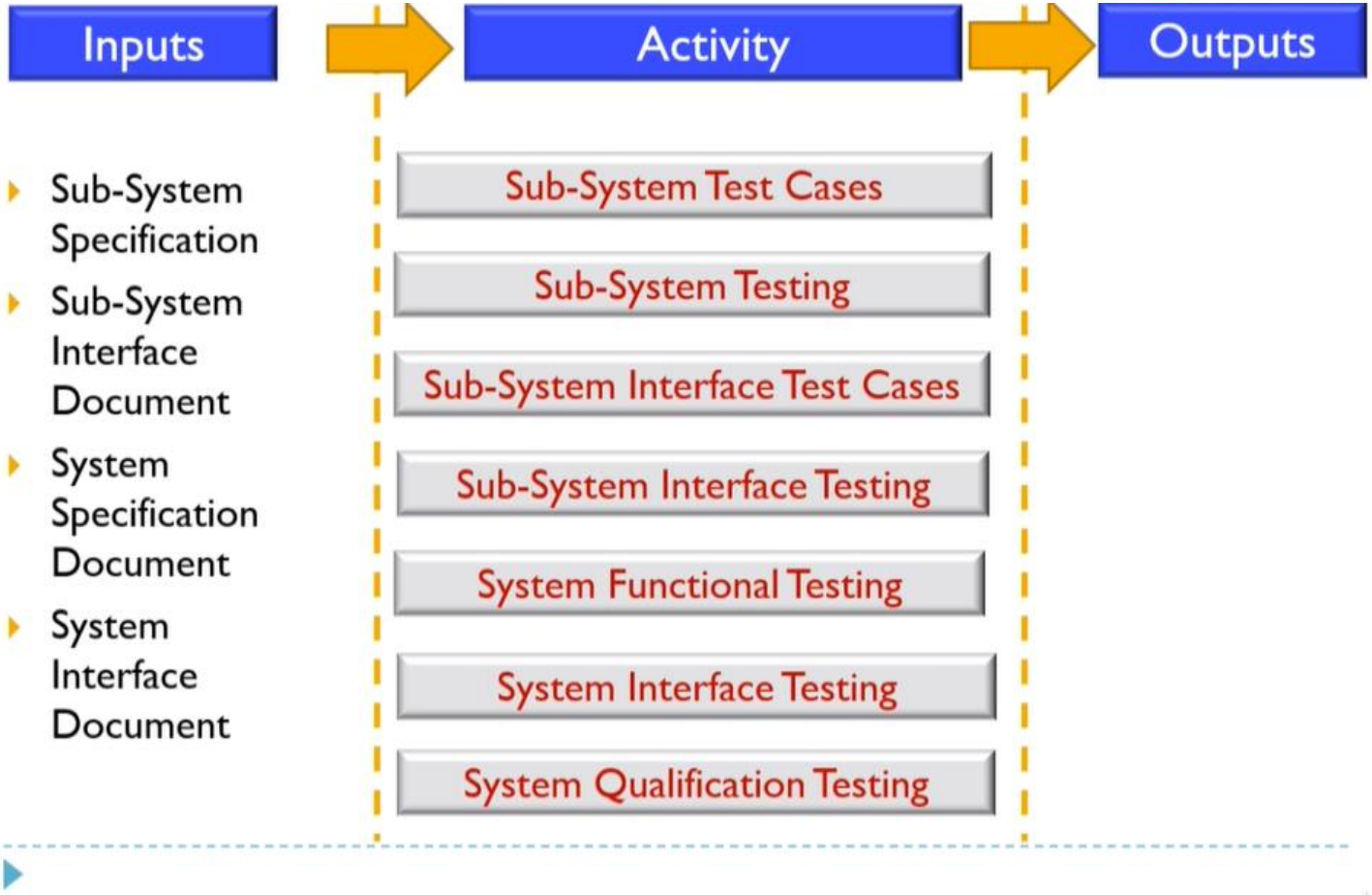
System Testing



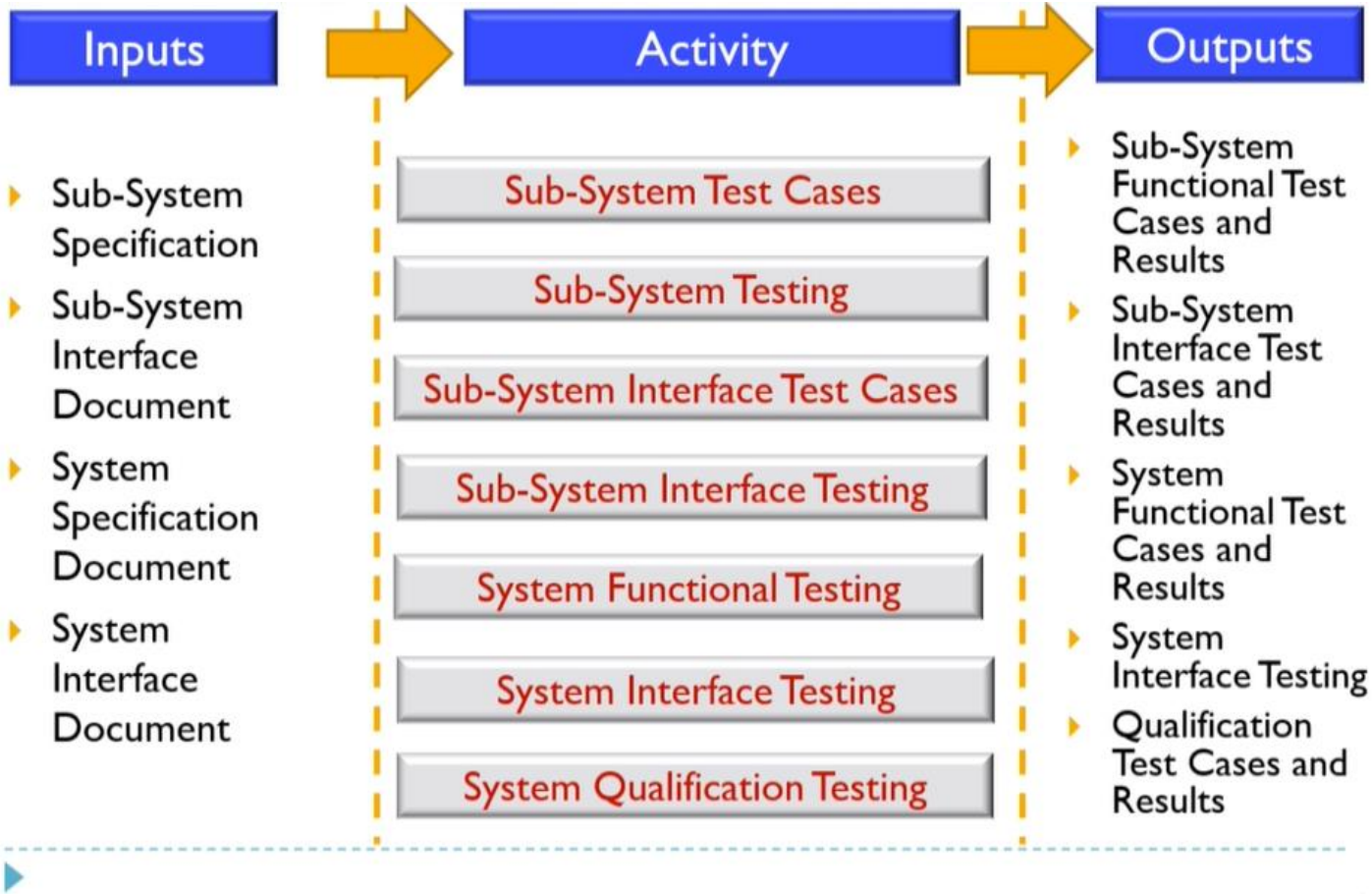
Input: System Testing



Activity: System Testing



Output: System Testing



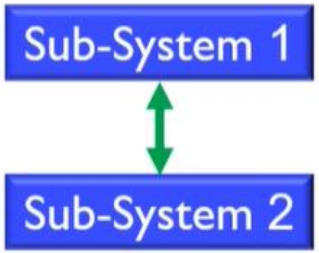
System Testing

Sub-System Testing

- Perform the functional testing of each Sub-System as per specification
- Perform the Qualification Testing
- Check for the different interfaces of Sub-System

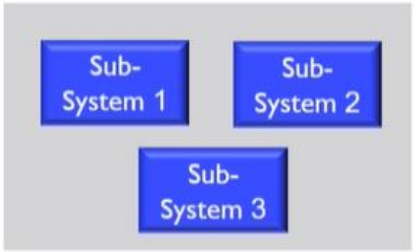
Sub-System Interface Testing

- Interconnect the Sub-Systems. Perform the Interface Testing between Sub-Systems



System Functional Testing

- Connect all Sub-Systems
- Test as a whole system w.r.t System Specifications



System Testing: Tools

- Test Tools
  - VectorCAST
  - LDRA
  - RTRT
  - dSPACE
  - Proprietary Agents

Consequence

Consequence is nothing but an end result of any Failure

In General, End Result of any System failure can lead to,

- Death of People
- Injury to People (Major or Minor)
- Damage to external environment
- Discomfort

Consequence

- **Death of People**
  - In case of Aerospace, when flight crashes due to engine failure, it causes death of people.
  - In case of Automotive or Rail, If Engine fails, it may not kill people, but may lead to discomfort for passenger or driver.
  - In case of Rail, the overspeed may derail the train and kill the passenger.
  - In case of Automotive, the overspeed may overturn the vehicle and may kill the passengers.

In case of Aerospace, when flight crashes due to engine failure, it causes death of passengers. But In case of Automotive or Rail, If Engine fails, it may not kill people, but may lead to discomfort for passenger or driver. From the above example, it is clear that Consequence will be different (death or discomfort) for Aero/Rail/Automotive for the same cause (i.e., Engine Failure)

Thank You !

L&T Technology Services