System Integration

INCOSE SFBAC

David Mason, CSEP
Chin-An Cheng, CSEP

Agenda

- Trends of System Integration
- Definition of System Engineering
- SE Process
- Definition of System Integration
- Division of System Integration
- Traditional View of SI
- Early System Integration Process
- System Integration in program phases
- System Integration

Trends of System Integration

Systems Integration has progressed from an individual effort to cross organizational teams

Definition of Systems Engineering

- System: An integrated set of elements that accomplish a defined objective.

- System Engineering: An interdisciplinary approach encompassing the technical effort to evolve and verify an integrated and balanced system that satisfies stakeholders needs.

Program Phases →		0	I	II	III	
Legend: ● = Major Use, o = Minor Use System Engineering Processes	PRE CON	CE	PD&RR	EMD	PFD&OS	DISPOSAL
1. Pre-Proposal Activities						
Mission, SRD, SOW, RFP, CDRL	●	●	●	●	●	●
2. Requirements Analysis						
Capture Source Requirements	●	●	●	●		●
Develop Operational Concept	●	●	●	●		
Functional Performance Reqts.	o	●	●	●		
Design Constraint Reqts.	o	●	●	●		
Requirements Allocation	o	●	●	●		
3. Functional Analysis	o	●	●	●		
4. Sys. Architecture Synthesis						
Synthesize Multiple Arch's	o	●				
System Element Reqts.	o	●	●	●		
Eval/Select Pfd. Architecture	o	●	●	●		
Integrated Sys. Physical Config.	o	●	●	●		
Define/Refine Interfaces	o	●	●	●		
Develop Spec. Tree & Specs.	o	●	●	●		
5. Systems Analysis						
Tradeoff Studies	o	●	●	●		
System Modeling & Simulation	o	●	●	●	o	
Risk Management	o	●	●	●	●	●
Life Cycle Cost Analysis	●	●	●	●	o	
Cost & Effectiveness Analysis	●	●	●	●	o	●
6. SE Product Control		●	●	●	●	
7. SE Process Control						
SEMP, SEMS/SEDS, TPM, Audits		●	●	●	●	
Mfg. Involvement in SE Process	o	o	●	●	●	
8. Sys. Implementation Support						
System Integration	o	●	●	●	o	●
System Verification		●	●	●	o	●
Baseline Maintenance		●	●	●	●	
Sustaining Engineering				●	●	

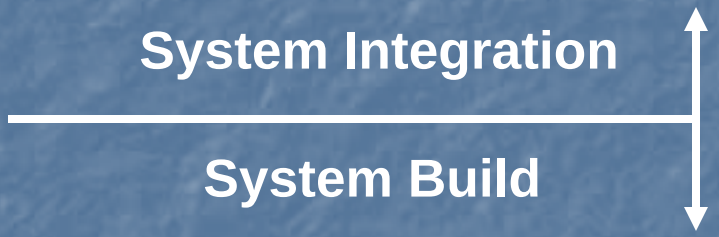
The System Engineering process identifies System Integration being performed from the beginning in the Concept/Exploration Phase or earlier

Definitions of Systems Integration

- System Integration (SI) :
  - the multiphase and multilevel effort to merge two or more component, or configuration items, into a higher level system element and ensuring the logical and physical interfaces are satisfied, and the integrated system satisfies its intended purpose.
  - the key objective of the systems engineering process performed throughout the product development and validation process to create a unified and functional product.

Division of System Integration

- System Integration: the external interfaces between the system and other systems, entitled System Integration with External Systems.



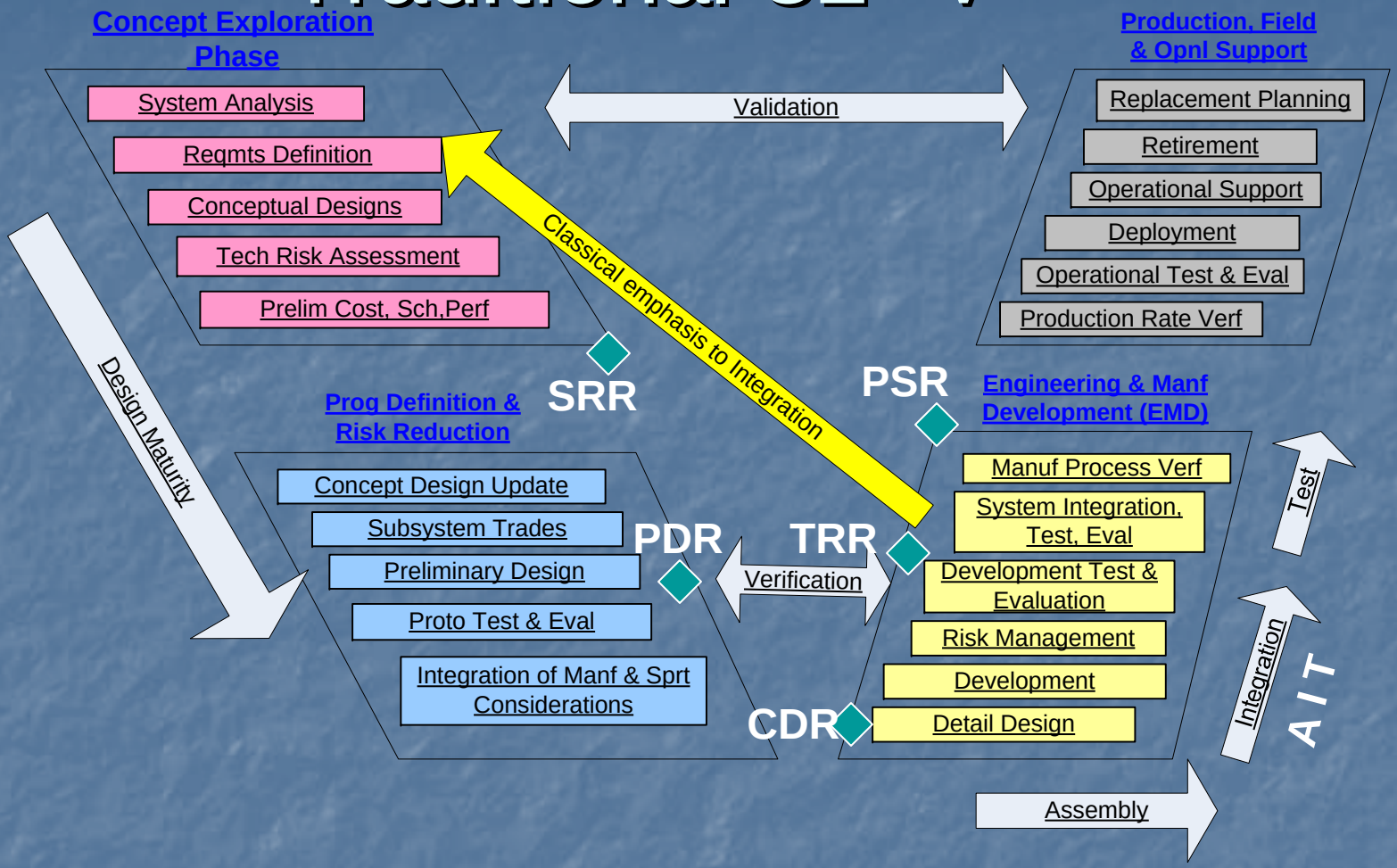
- System Build: the internal interfaces among the components and subsystems comprising the system.

This presentation will focus on ‘System Build’

Traditional View of System Integration

- Associated with Assembly Integration and Test (AIT).
  - Combining system components after build
  - Focuses on the assembly and testing to verify predicted performance.
- Results
  - Integration verification is performed late in the product life cycle.
    - High program costs
    - Extended schedules.

Traditional SE "V"



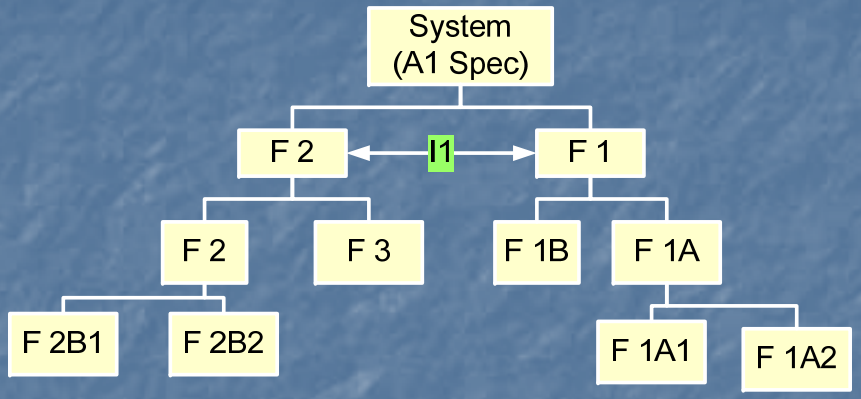
Classical AIT results in higher risks, late verification, higher program costs, and late delivery schedule

Elements of Low Risk SI

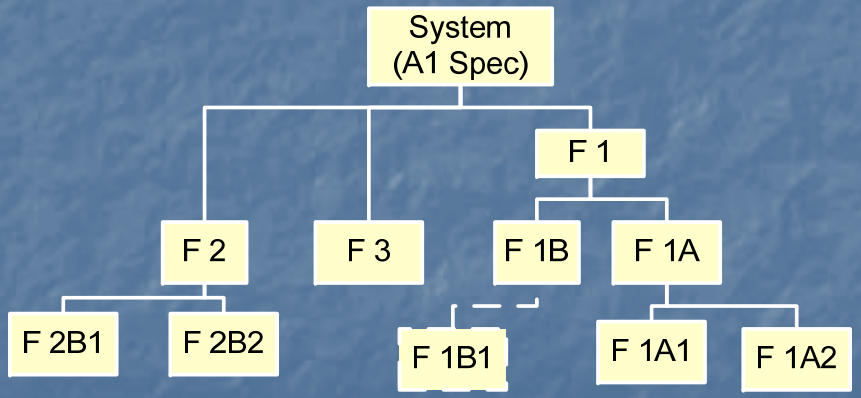
- Initiated early in the product Life Cycle
- Balanced level Architecture
- Component level Architecture
- Clearly Defined Functional, Performance, and Integration requirements.
- Identify Interfaces between elements.
- Establishes Interface Ownership.

Objective: start early in the program, perform component level architecture, verify early and at Lowest Level possible

Balanced Requirements Hierarchy



- Balanced
  - Balanced number of functional tiers
  - Enables component level verification.
  - Illustrates Parent/child
  - Identifies Interfaces
  - Used to define Orgn Responsibility



- Unbalanced
  - Unequal number of functional tiers
  - Difficult to perform component level verification
  - Difficult to verify interfaces from different hierarchy levels

Clearly Defined Requirements

- Functional
  - Each function has a single purpose, unique subsystems, redundant only if required, has a defined ownership
- Performance
  - One Parameter per specification line, quantified engineering units, easily measured by standard equipment and processes.
- Interface Type
  - Clearly identifiable as a Product, Process, Function, Type, Timing, Organization, or technology driver

Types of Integration

- **Program Integration.** Program Integration ensures that (1) program technical activity objectives, tasks, organizations, and schedules efficiently support system technical objectives within the constraints of the contract
- **Requirements Integration.** Ensures requirements are complete, self-consistent, and traceable throughout the system hierarchy and interfaces.
- **Design Integration.** Ensures design criteria and analysis are applied to the system and the resulting designs are compatible across all subsystems.
- **Product Integration.** Ensures physical and functional compatibility of hardware and software components, subsystems, and system elements as the product develops from concept, through assembly, integration, test, to final delivery to the customer
- **Process Integration.** Ensures that all process elements are tailored and integrated into the overall program planning and strategy.
- **End-Item Integration (Validation).** Ensures the integrated end item is validated for intended use in its intended service or operating environment.

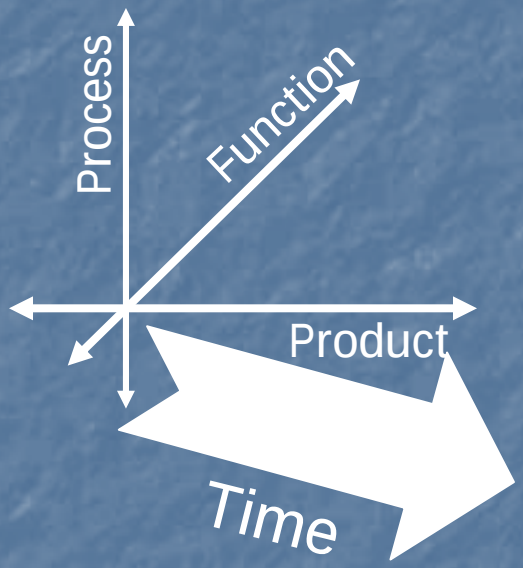
Integration Spaces

- System Integration: The rich mixture of three integration components applied in combination defined by the resultant integration spaces to the work confined to a finite number of Integration cells across the program world line that actually comprises system integration on any program.

  (INCOSE Fellow: Jeffrey O. Grady)

What are components, spaces, cells, and program World Line?

Integration Spaces

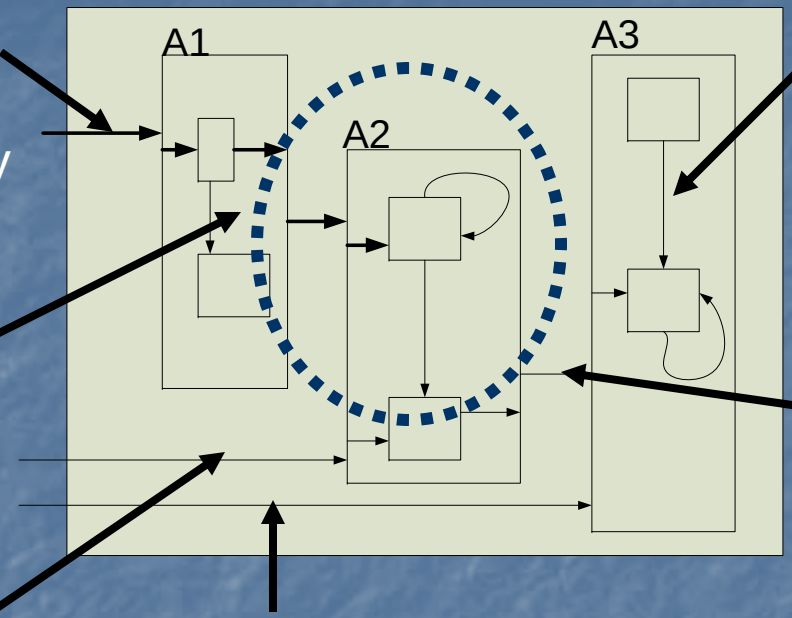


- Three integration components:
  - Product, Process, Function
- Three-Value situation:
  - Co, Cross, Null
- The number of combinations of multiple integration components and values is: $S=C^n$
- Types of Interfaces:
  Inner, outer, cross, external, extended, and inference.
- Interfaces change (evolve) over time.

Where are these Interfaces Located?

Types of Interfaces

- External: One terminal enters the responsibility of the Organization
- Inference: an indirect impact to a product or element
- Extended: Software I/F



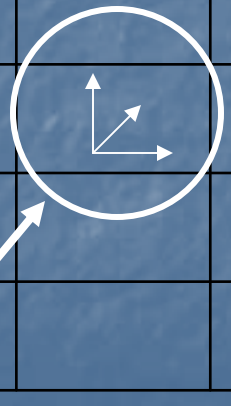
- Innerfaces: Where both terminals & message are within the Organizational responsibility
- Crossfaces: Where a terminals cross between Organization responsibility.
- Outerfaces: Where neither the terminals or messages enter the responsibility of the Organization

The impact of Interfaces are based on Organization Point of Reference

SI and the N^2 Flow Diagram

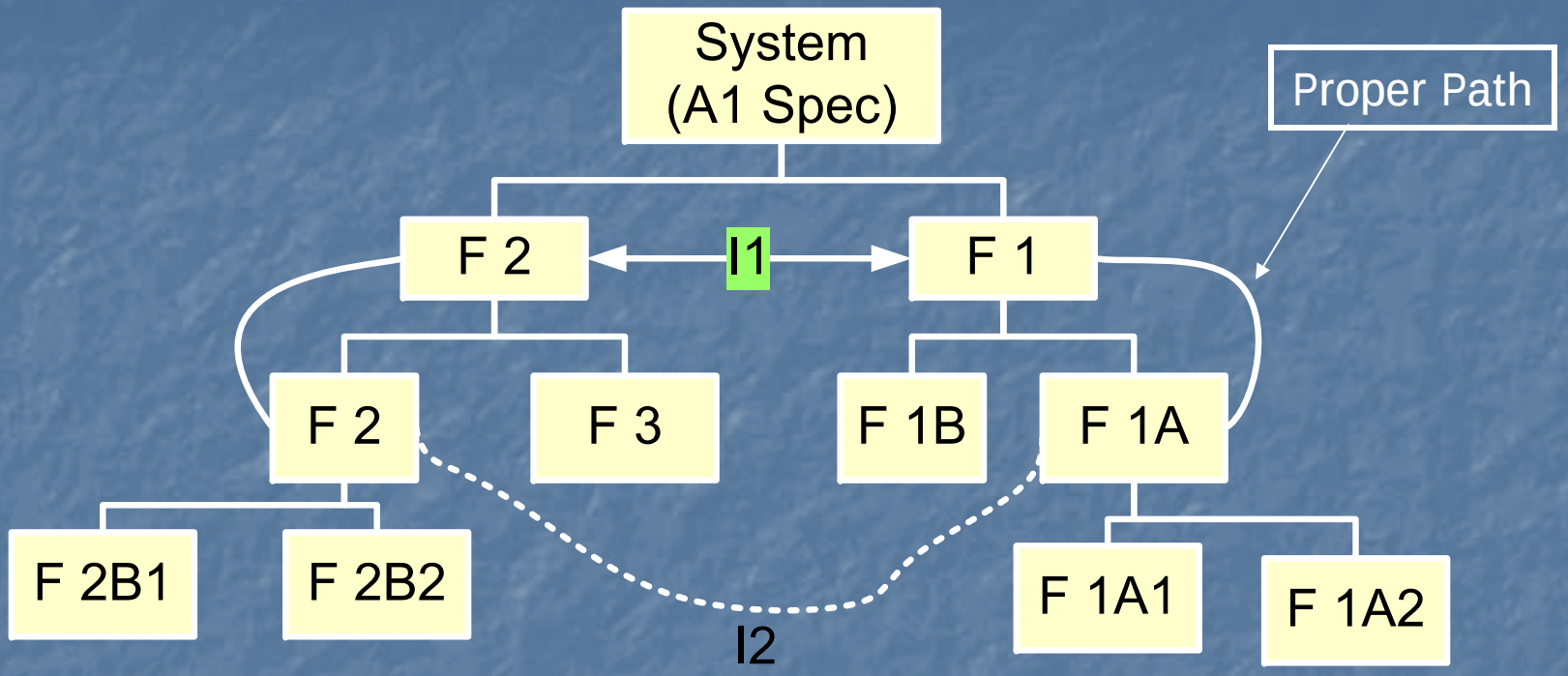
Capture Source Documents	Requirements	Required Capability	Mission Parameters	Source	External I/F Compliance Documents	Mission Requirements	Mission Requirements, External Interface Requirements
Requirements Validation	Develop Ops Concept	Operations	Scenarios	Equipment Experience	Organization & Personnel		Timeline Requirements
		Functional Analysis	Functional Sequences, Timelines	Functional Areas	Required Functionality	Required Capability	Derived Reqts, Functional Reqts, Internal Interface Requirements
			Simulation	Sizing	Sizing	Quantified Performance Capability	Derived Parameter Times
			Configuration	System Architecture	Candidate Approaches	Configuration Item Definition	Allocated Path
				Selected Design	Trade Studies	Quantified Performance	Quantified Requirements
Tailored MIL-STDs & MIL-SPECs				Ops Environment, GFE, Cost	Cost	Design Constraints	Interface Requirements	Engineering Specialty Reqts
Verification of Compliance			Margins & Deficiencies	Margins & Deficiencies	Adverse Consequences	Margins & Deficiencies	Performance Evaluation	Requirements
Traceability			Functionality	Functionality	Baseline Capability Required	Traceability of Flowdown	Required Capability	Requirements Database

Destination



SI Interfaces are at the intersections of Life Cycle Phases

Interfaces and Protocols



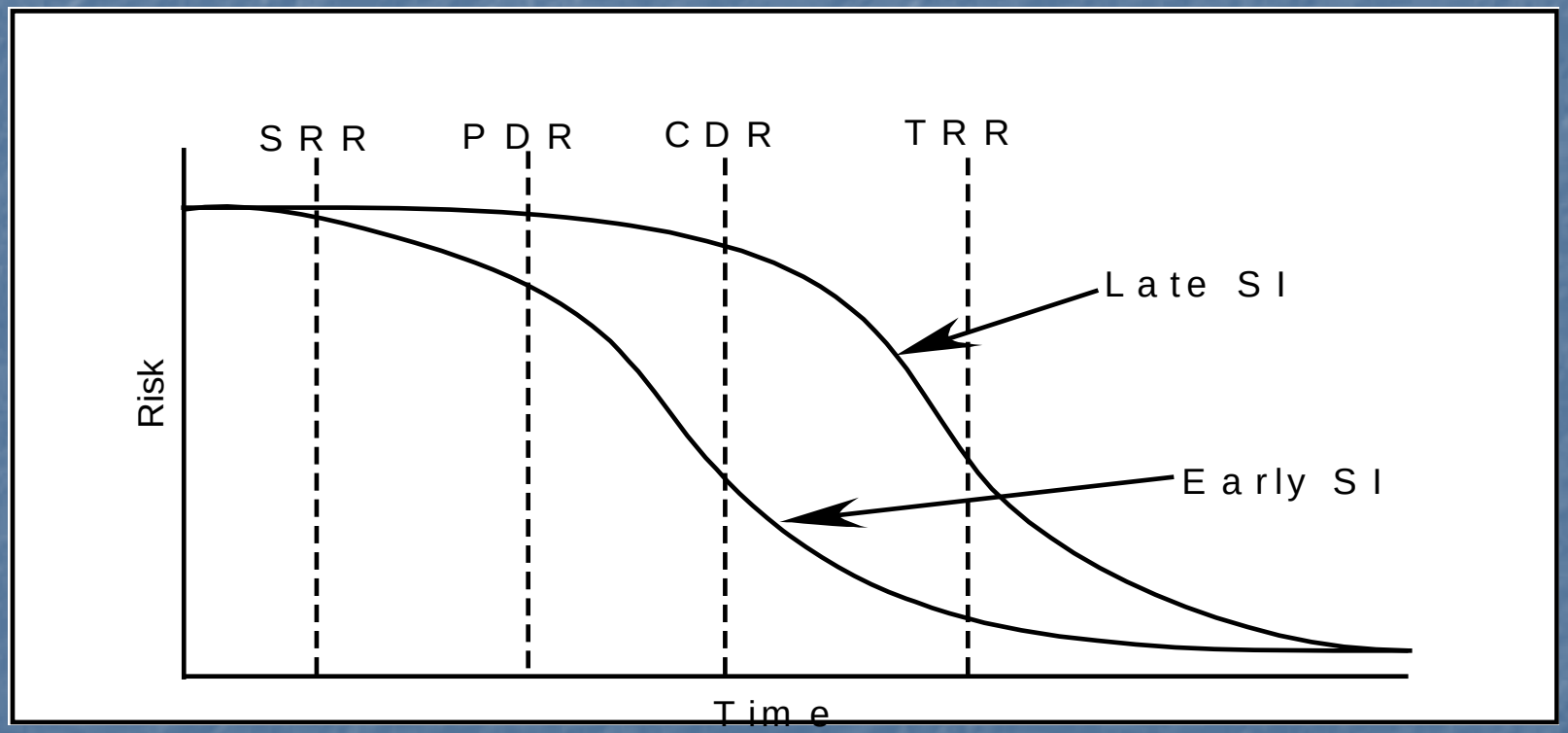
- Interfaces traveling across functions have a protocol to travel through the next higher level and then down to the required function.

System level interfaces are the essence of SI

Managing Interfaces

- Interfaces should be managed in the SE process
- Currently most interfaces are 'Controlled' with ICDs
  - A reactive position to interfaces
- Manage Interfaces by designing and verifying them early in the product Life Cycle.
  - Design the interfaces into the program hierarchy
  - Plan interface activities (mechanical, thermal, etc)
  - Plan process ownership- use a database hierarchy
  - Develop Interface Control Working Groups (ICWG)
  - Identify the type of interface and responsibility.

Early SI Risk Profile



Early System Integration reduce program Risk

Conclusion

- System Integration begins early in the Concept phase of the product Life Cycle.
- Functional, Performance, an Integration requirements receive equal importance and weighting during the allocation and Hierarchy construction process.
- Interfaces come in a variety of types depending on Frame of Reference.
- Interfaces occur at the intersection of integration spaces along the product life line.
- Interfaces need to be actively managed