

Introduction to Systems Engineering

Chapter 5

Preliminary and Detailed Design

ASTU, 2024

- The systems engineering processes begin in earnest with the first activity in the Acquisition Phase: Conceptual Design (or logical design or problem- domain design), which aims to articulate the needs, to analyze and document the system-level requirements flowing from the needs, and to complete a logical design of the system.
- The major product of the Conceptual Design is the Initial Functional Baseline (FBL)
- FBL provides a system-level logical architecture that is the basis for subsequent lower-level design.

Preliminary Design

- Preliminary Design starts with the Initial FBL and continues to translate system-level requirements into design requirements for the system elements that will combine to form the system.
- Specific requirements for the elements making up the system need to be determined.
- Trade-off studies are conducted and the result of the Preliminary Design effort is the establishment of an Allocated Baseline (ABL), in which requirements are 'allocated' to specific physical system elements that combine to form the system.

Activities during preliminary design

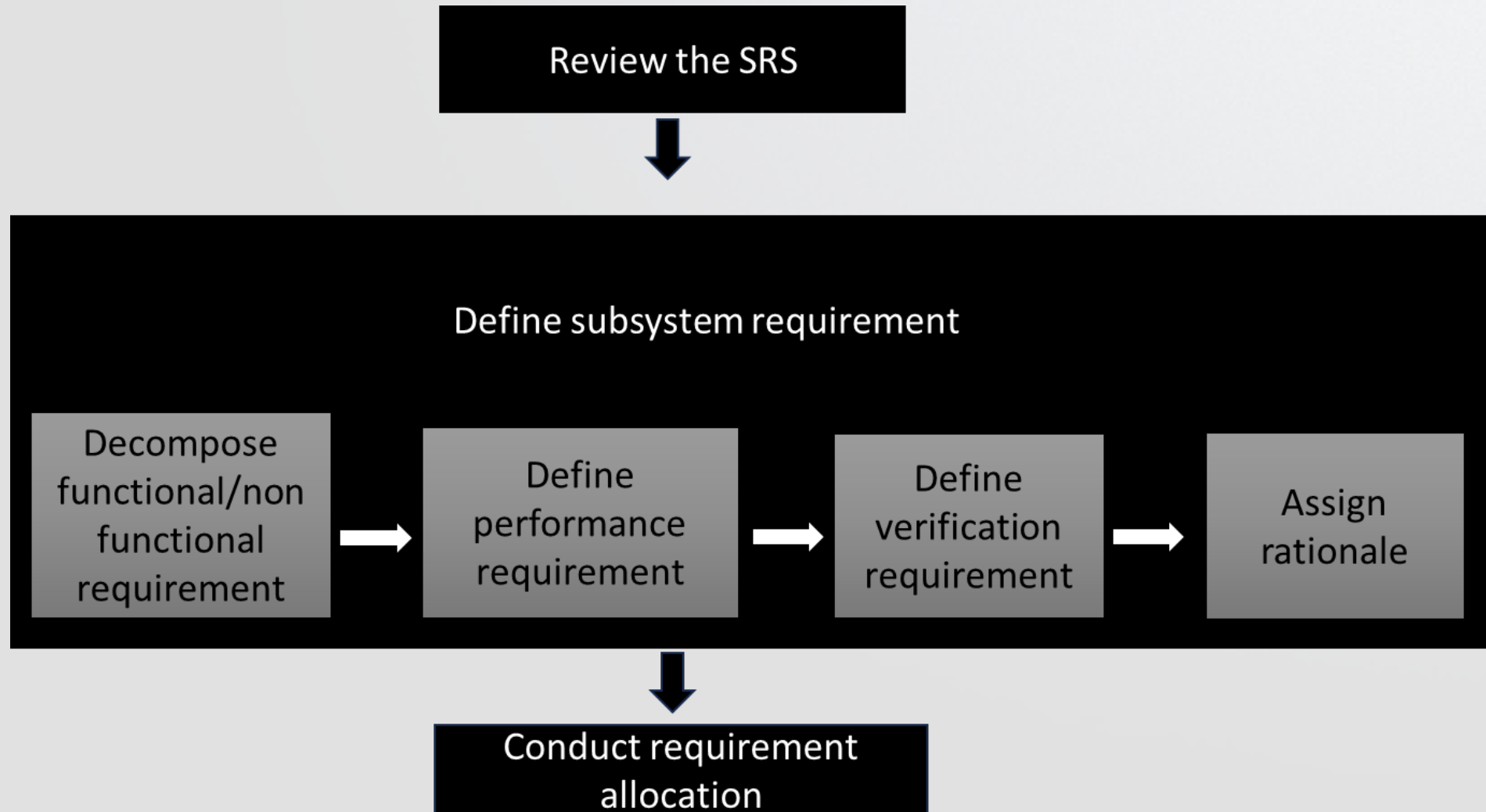
- Subsystem requirements analysis
- Requirements allocation
- Interface identification/design
- Subsystem-level synthesis
- Preliminary Design Review (PDR)

SUBSYSTEM REQUIREMENTS ANALYSIS

SRA Process and Initiation

- elaboration from the system level to the subsystem level and so on until all functions, parameters and interfaces have been defined and the necessary resources have been identified.

Major activities in subsystem requirements analysis



SUBSYSTEM REQUIREMENTS ANALYSIS

- It is not possible to prescribe the level of detail that needs to be decomposed/derived during subsystem requirements analysis.
- The detail must be sufficient to define completely the functions required of the major subsystems comprising the system, and must be able to be used either to procure, or to design and produce, the major subsystems and their assemblies and components during subsequent stages of the systems engineering process.

SUBSYSTEM REQUIREMENTS ANALYSIS

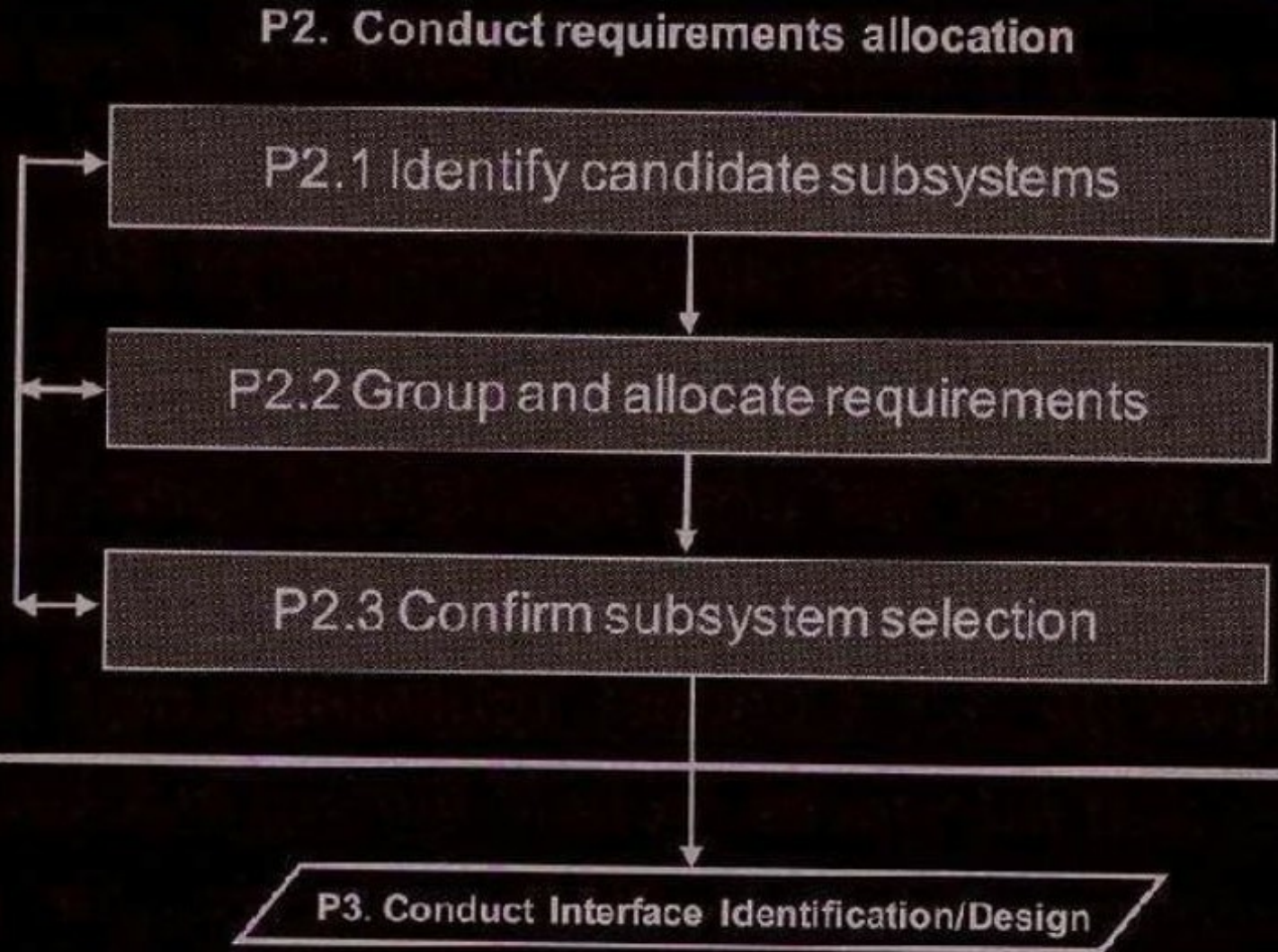
- The diagrams that are generated during the Preliminary Design effort are more detailed than in Conceptual Design and are aimed at defining the subsystems (rather than the system).
- Note that this stage is therefore the beginning of the conversion of the 'what' the system needs to do into the 'how' it is going to do it, which is more commonly referred to as the translation of logical design into physical design (or from the problem domain into the solution domain).

Requirements Allocation

Requirements Allocation

- During Preliminary Design, requirements allocation takes a new dimension. Unlike Conceptual Design, where groups were based on the Requirements Breakdown Structure (RBS) headings, in Preliminary Design, groups are formed around a preliminary physical architecture crafted by the designers.
- The identified groups of similar functions become pivotal in determining the design of major system elements, referred to as Configuration Items (CIs). These CIs are either selected or designed to execute the group of requirements assigned to them. The primary focus during this phase is the allocation of requirements to these Configuration Items.

Requirements Allocation



Significance of Requirements Allocation

- It provides a structured approach for designers to define the major elements of the system, ensuring that each Configuration Item is tailored to perform the specific set of requirements assigned to it.
- In essence, Requirements Allocation in Preliminary Design streamlines the design process, facilitating a systematic and organized development of the major system components.

Requirements Breakdown Structure (RBS) and the Work Breakdown Structure (WBS)

Definition

a. Requirements Breakdown Structure (RBS):

The Requirements Breakdown Structure (RBS) is a hierarchical representation that systematically decomposes and organizes the logical requirements of a system. It outlines the relationships and dependencies between various requirements, helping to structure and categorize the essential functionalities and characteristics needed for the successful development of a system.

Definition

a. Work Breakdown Structure (WBS):

The Work Breakdown Structure (WBS) is a hierarchical decomposition of the total scope of work to be carried out by a project. It breaks down the project into smaller, more manageable components, known as work packages. The WBS is organized in a way that reflects the physical structure of the project and serves as a foundation for project planning, management, and control. It outlines the tasks, activities, and deliverables required to complete the project successfully.

RBS VS WBS

- The RBS articulates the logical design, representing the breakdown of logical requirements into subsystems or Configuration Items (CIs).
- On the other hand, the WBS documents the products and work packages needed for system development, encompassing the design, development, integration, and testing of subsystems/CIs.

RBS VS WBS

- While the RBS focuses on logically organizing requirements, the WBS is structured physically, emphasizing project management imperatives.
- The WBS includes not only the subsystems/CIs but also encompasses additional work related to design, development, integration, and testing. This structural difference reflects the distinct purposes of each
 - logical organization versus physical project management.

Interface Identification and Design

- During the selection of the subsystems/CIs comprising the system, the interfaces between the subsystems/CIs are identified.
- Identification of interfaces is a critical part of Preliminary Design because interfaces not only determine successful operation of the system once integrated, but they also place additional limitations and requirements on the design of the individual subsystems/CIs.

Importance of interface management

- Interface management is crucial in systems engineering due to the prevalence of problems that often come in at the interfaces between different components or subsystems.
- Recognizing and defining interfaces with precision can mitigate risks associated with miscommunications, compatibility issues, and errors during the integration and operation phases of a system.
- Effective interface management contributes to the overall reliability, performance, and longevity of a system.

Types of interfaces

- ❖ The types of interfaces utilized in the design process are tied to the system architecture, by considering how different elements of the system interact, and this consideration guides the selection and definition of interfaces.

- **Physical Interfaces:**

Physical interfaces involve the tangible connections between different hardware components. This can include electrical connectors, cables, ports, and other physical linkages.

- **Electronic interfaces.** An electronic interface is the name given to the flow of electronic signals (analogue or digital) between two points.
- **Electrical interfaces.** Electrical interfaces are normally associated with a physical interface of some sort such as a power cable or data cable. The type of information required in this sort of interface will include

Continued ...

- **Hydraulic/pneumatic interfaces.** Hydraulic and pneumatic interfaces are the mechanical equivalent of the electronic or electrical interfaces already described. A physical interface definition will coexist with this type of interface. Additional information such as flow rate, pressure, temperature, and fluid type will be required to define completely the hydraulic and pneumatic interfaces.
- **Software interfaces.** Software interfaces refer to passing of data between different computer software CIs. This may occur within the one piece of hardware using internal hardware buses or it may be between two different pieces of hardware.
- **Environmental interfaces.** Environmental interfaces are sometimes not considered to be interfaces but they form the interface between the system and its external environment. In some case, the environment provides the system with important information such as

Interface induced failures

- ❖ Interface-induced failures refer to issues and problems within a system that arise specifically due to inadequacies or problems at the interfaces between different components, subsystems, or systems.

- **Neglect and Weak Points:**

Overlooking interfaces during design creates vulnerabilities, potential weak points, leading to system failures or suboptimal performance.

- **Bottlenecks:**

Inadequate or mismatched interfaces cause bottlenecks, impeding the flow of information, energy, or matter and causing delays reduced efficiency, and system degradation.

Continued...

- **Structural Failures:** Interface-induced failures contribute to structural issue within a system, particularly when physical connections between components are poorly designed.
- **Erroneous Function Calls:** Miscommunication at interfaces can result in incorrect function calls, leading to unexpected behaviors, system malfunctions, or safety hazards.
- **Compatibility Issues:** Lack of interface compatibility, such as mismatched data formats, causes failures during system integration, emphasizing the importance of ensuring compatibility to prevent data corruption and misinterpretation.
- **Mitigation and Prevention:** crucial during design and development, involving thorough testing, verification, and validation processes to address issues before operational deployment

Interface control documents (ICD)

- ✓ In large and complicated system acquisitions, a wide range of technical specialists (or experts) will be involved with producing a wide variety of subsystems.
- ✓ The specialists perform work in the traditional engineering fields such as electrical, civil, mechanical and aeronautical engineering, and in very specialized and detailed areas such as safety, human factors, electromagnetic interference/compatibility (EMI/EMC) and corrosion control.
- ✓ With all of these specialized areas working on the same project, there is a significant requirement for a coordinated approach to the

Interface control documents (ICD)

- ✓ The task of systems integration falls within the bounds of systems engineering management and can be managed effectively by ensuring the adequate attention is paid both to the logical and physical interfaces between the different areas.
- ✓ When two or more CIs have an interface, the teams developing those CIs will no doubt have an interest in and specific requirements for that interface.
- ✓ To ensure that the teams developing the CIs have their needs met, and to facilitate communication between the teams, an interface control working group (ICWG) may be formed to bring together those involved in the interface definition and design.
- ✓ ICWGs must meet regularly to generate the initial ICD and should then meet

Subsystem-Level Synthesis and Evaluation

- Once the allocation of requirements has been performed, it is time to continue the familiar loop of analysis, synthesis and evaluation and concentrate on synthesis.
- Subsystem-level synthesis results in a Preliminary Design that meets the subsystem-level requirements.
- **Review Sources of Subsystem Requirements**
- The allocation process that has allocated requirements to individual CIs has completed the initial analysis.
- Designers responsible for each CI continued the analysis to an additional level of detail necessary to support the subsystem-level synthesis.
- Technical specifications and ICDs have been produced to detail the requirements, performance and interface requirements for each CI. As a first step in the synthesis process, the specification and ICDs for each CI should be reviewed, since it is these specifications that must be satisfied

Investigate Design Alternatives

- The initial synthesis is conducted to determine a preliminary design that identify the specification using one of the three broad design options available.
- There are three broad design options available including
 - Commercial-off-the-shelf (COTS),
 - Modified COTS, and
 - Developmental items.
- Factors that impact on the designer's decision to use COTS, modified COTS or specially developed items include the specific requirements to be performed by the item, the availability and stability of the current technology, size of the market, supportability and cost, and any contractual directives. The following sections discuss each of these options.

(PDR)

- The Preliminary Design Review (PDR) is a crucial phase in the engineering and product development process.
- Preliminary Design Review (PDR): It is a structured assessment and evaluation of the initial design concepts and plans for a project or system. This review takes place after the completion of the conceptual design phase and before moving into the detailed design phase.
- The primary purpose of the PDR is to examine the proposed design, ensuring that it aligns with the project requirements and objectives. It serves as a checkpoint to validate the feasibility and soundness of the design before significant resources are committed to the detailed design and development phases

Key Aspects

Design Concepts: Evaluation of the overall design concepts, including architecture, functionality, and features.

Requirements Compliance: Verification that the design meets the specified requirements and objectives set forth in the project plan.

Risk Assessment: Identification and assessment of potential risks associated with the proposed design and strategies for risk mitigation.

Feasibility: Examination of the technical feasibility and practicality of the design approach.

Cost and Schedule Estimates: Review of initial cost estimates and project schedules to ensure alignment with organizational constraints and expectations.

Outcome and benefits

- The outcome of a successful PDR is the approval to proceed to the next phase of detailed design and development. Alternatively, it may lead to recommendations for design modifications, additional analyses, or further refinement before moving forward.
- Conducting a PDR helps in identifying and addressing design issues early in the development process, reducing the likelihood of costly changes later. It ensures that the design aligns with the project goals, meets requirements, and is technically and economically feasible.

Tools

- There are a number of tools that may be used during Preliminary Design (in some cases, these tools will have already been used in Conceptual Design and will also be used in later design activities).
- This section aims to introduce the more common tools and briefly explain their characteristics.
- ❑ **Schematic Block Diagrams:** Schematic block diagrams (or simply schematics) are perhaps the most basic and popular of the synthesis tools.
- ❖ It allow designers to visually describe a solution to a given logical need as defined during the analysis stage of the process.
- ❖ It show all of the elements of the solution and show their interrelationships and interfaces. These interfaces will include internal interfaces (between schematic elements) and external interfaces (with other Cis). Schematics can be drawn at different levels of detail during systems engineering processes.

Tools ...

- **Physical Modelling:** Physical modelling can take anyone of a number of forms. Perhaps the best- known physical models are engineering models, such as the wind-tunnel models used during vehicle design to determine aerodynamic stability and characteristics. These models can include models of aircraft, land vehicles and sea going vehicles such as ships and submarines. They are normally scale models and are built to exacting standards to ensure that the results obtained during the trials will be representative of the full-scale variant.
- **Mathematical Modelling and Simulation** Mathematical modelling and simulation being useful tools for requirements exploration-they are also used during the physical design stages of system developments.
- Models and simulations are excellent tools for performing what if analyses and determining the limits of the design.
- Variables can be changed and conditions easily altered to allow designers to investigate the likely behaviour of the design in varying conditions. Simulators are widely used for a variety of different

Tools ...

➤ **Trade-Off Analysis** Throughout this text, references are made to the evaluation of alternative solutions against multiple criteria and the selection of the preferred approach. The process of analysing alternatives and selecting the preferred solution is often generically called trade-off analysis (or more generically, multi-criteria decision analysis or decision making). The analysis process must consider the different alternatives and balance the many (potentially conflicting) requirements prior to selecting the preferred solution. It must be emphasized that trade-off analysis is a decision-making aid only, and should be used by the decision makers as a tool in making decisions.

Detailed Design and Development

- ❖ The Detailed Design and Development activity continues the development effort and makes use of the FBL and ABL developed during Conceptual Design and Preliminary Design.
- ❖ The detailed design effort takes these definitions of the overall system (as contained in the FBL) and of the major CIs (contained in the ABL) and finalizes the design of specific components that make up the CIs (and subsystems).
- ❖ The realization and documentation of individual components used to support production is referred to as the Product Baseline (PBL).
- ❖ Product Baseline (PBL). A number of tasks are undertaken during Detailed Design and Development. The major technical activities include:
 - ❖ describing the lower-level assemblies and components making up the CIs (and their interrelationships);
 - ❖ defining the characteristics of the above items through specifications and

Detailed Design and Development ...

- ❖ either procuring the above items off-the-shelf, or designing them if they are unique to the system under development;
- ❖ developing prototypes or engineering models of the CIs (or even the final system) by integrating the above items (for the purpose of design verification, and system-level test and evaluation);
- ❖ redesigning and retesting (as required); and
- ❖ conducting the Critical Design Review (CDR) to confirm that the design is ready for construction and production.

Detailed Design Requirements

- ❖ System-level requirements and constraints were derived and recorded in the SyRS during Conceptual Design.
- ❖ A system-level solution was selected during Conceptual Design as the preferred way of delivering the requirements. The solution was developed further during Preliminary Design where details of the various CIs and interfaces were refined.
- ❖ The requirements associated with each of the CIs, including the internal and external interfaces were recorded in Development Specifications.
- ❖ The design approach to realizing the CIs was also considered during Preliminary Design, including what elements of the design would be satisfied by hardware-dominated CIs and what elements of the design would be allocated to software CIs.
- ❖ Regardless of whether the requirements were allocated to hardware, software, or a combination of the two, there were three broad options available to realize the design: commercial-off-the-shelf (COTS) modified

Detailed Design of Hardware

- It is not possible to cover all of the categories of hardware associated with complex system developments. It may be useful to consider what the term may mean to engineers from different engineering disciplines as follows:
- ✓ **Electrical/electronic engineer.** Hardware may be an electrical or electronic system of some description that relies on a combination of electronic components arranged in such a way as to manage, condition, or distribute electrical power or signals.
- ✓ **Civil engineer.** Hardware may be the foundations and structural components of a bridge or other piece of physical infrastructure.
- ✓ **Mechanical engineer.** Hardware may be a physical system such as an engine, chassis, or drive train.
- ✓ **Aeronautical engineer.** Hardware may be the structures and components associated with the wings and fuselage of an aircraft.
- ✓ **Software engineer.** Hardware may be considered to be anything that is not software including computer processors, network infrastructure, and

Detailed Design of Hardware ...

- The design artefacts will support the production and construction of the hardware-aspects of the system in subsequent phases of the lifecycle.
 - Generic Hardware Detailed Design Process
 - Preparing for Detailed Hardware Design
 - Performing Detailed Hardware Design
 - Proving the Detailed Hardware Design
 - Documenting the Detailed Hardware Design

Detailed software design processes

1. Defining the scope

Detailed Software design

- Begins after requirements are determined and documented in subsystem-level documents.
- Utilizes various development environments and methodologies like Agile, XP, RAD.
- The optimal approach depends on the system's complexity, risk tolerance, and desired development pace.
- Rapid development methods blur the lines between design and production.

Detailed software design processes

- Modern software development tools have revolutionized the detailed design landscape.
- Integrated development environments streamline coding, debugging, and testing processes.
- These tools empower detailed design teams to iterate rapidly, optimize efficiency, and minimize errors.

contd

2. Preparing for Software Design

The software development specifications, allows the software developers to prepare for software design by starting with a clear understanding of what their software eventually needs to be able to do.

3. Performing Detailed Software Design

performing the details design using the chosen methodology and tools

Contd...

4. Documenting the design:

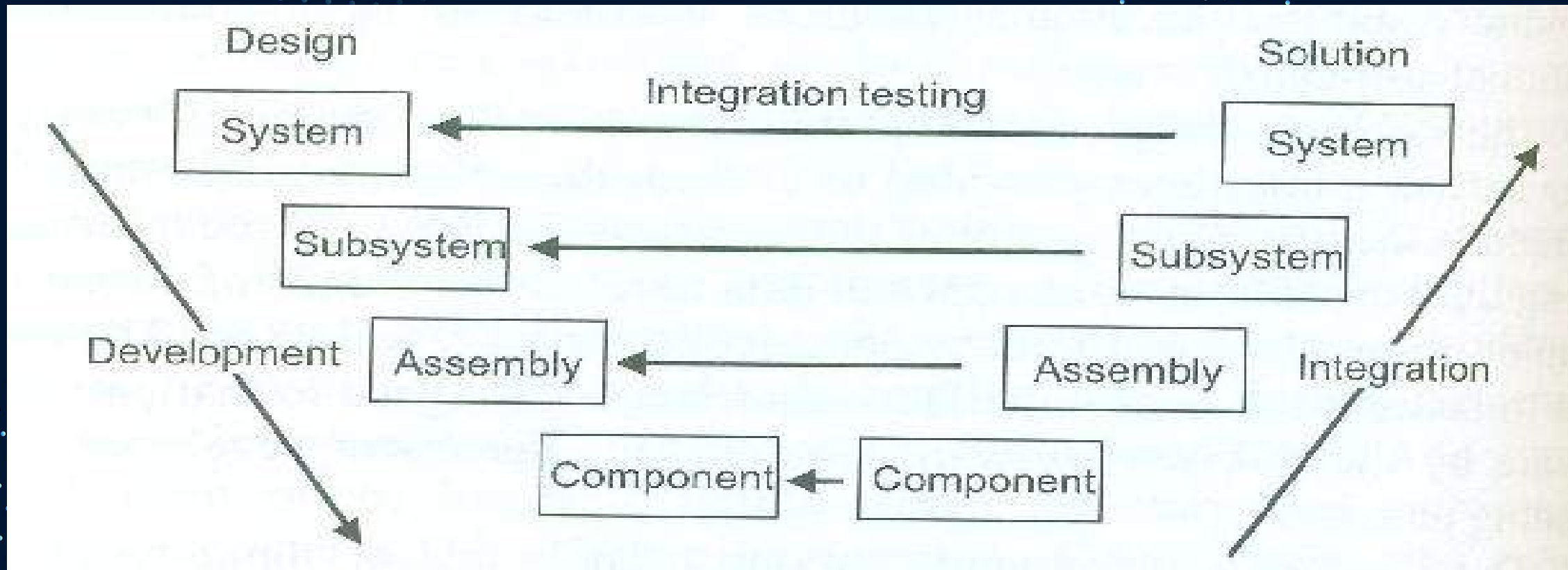
- Required by IEEE-STD-12207 for subsequent phases.
- Requirements are determined and documented in subsystem-level documents.
- Poorly documented software becomes a major problem during the utilization phase of the lifecycle especially when the software needs to be modified or maintained in some way.
- Includes design decisions, constraints, functionality, user interfaces, performance characteristics, and architecture.
- Details software interfaces, databases, assumptions, and design conventions.

INTEGRATING SYSTEM ELEMENTS

- At this stage, All CIs (hardware and software) and their interfaces are designed and tested.
- Low-level items are integrated into higher-level assemblies and tested against requirements.
- COTS items are procured and checked against supplier specifications.
- System-level prototypes are created to confirm design requirements.
- This prototype can be used to confirm that the system-level design requirements have been met.

INTEGRATING SYSTEM ELEMENTS

Top down and bottom up integration approaches



DETAILED DESIGN REVIEWS

~~Design review~~ (or a series of reviews) is normally conducted at the conclusion of Detailed Design and Development to ensure that the system design is complete prior to **Construction** and/or **Production**.

1. Equipment/Software Design Reviews:

- ❑ focused on particular items of software and hardware to ensure that the specific design approach meets the requirements.
- ❑ These design reviews, sometimes called equipment or software design reviews, are focused on particular items of software and hardware to ensure that the specific design approach meets the requirements.
- ❑ Associated documentation such as **trade-off** reports that justify and explain the design approach will also normally be investigated during these reviews.

DETAILED DESIGN REVIEWS

1. Equipment/Software Design Reviews:

- ❑ These reviews investigate the drawings, computer programs, material lists and the necessary specifications to thoroughly review the item under investigation.
- ❑ All equipment/software design reviews will be completed prior to the major formal review of the phase, which is called the Critical Design Review (CDR).

DETAILED DESIGN REVIEWS

2. Critical Design Review (CDR)

- ❑ Critical Design Review (CDR) is the final review before construction/production.
- ❑ CDR aims to satisfy requirements, confirm compatibility, and finalize test/production plans.
- ❑ CDR establishes the Product Baseline and freezes design activities.

AIMS OF CDR

~~The success of CDR is a key indicator of design maturity.~~

The aims of the CDR include:

1. to evaluate the detailed design:

- Detailed design is documented in various forms including the mature Product Specifications and related engineering drawings.
- CDR should evaluate this documentation set and the detailed design to ensure that it adequately addresses the original requirements.

AIMS OF CDR

2. To Determine readiness for production/construction.

- CDR should review the process specifications and the material specifications and the production plan to ensure that the detailed design for the hardware items can progress to Construction and/or Production.

3. To Determine maturity of software

4. Determine design compatibility

5. Establish the Product Baseline