

Detailed Design Definition

Detailed Design Definition

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

Lead Authors: *Alan Faisandier, Rick Adcock*

The purpose of the System Design is to supplement the system architecture by providing information and data useful and necessary for implementation of the system elements. Design definition is the process of developing, expressing, documenting, and communicating the realization of the architecture of the system through a complete set of design characteristics described in a form suitable for implementation.

□

Contents

Concepts and Principles

- Design Notion

- Design Characteristics and Design Enablers

- Relation with System Architecture

- Design Descriptor

- Holistic Design

Process Approach

- Purpose

- Activities of the Process

1. Initialize design definition

2. Establish design characteristics and design enablers related to each system element

3. Assess alternatives for obtaining system elements

4. Manage the design

Practical Considerations

Pitfalls

Proven Practices

References

Works Cited

Primary References

Additional References

Concepts and Principles

Design Notion

In industrial practices, the term *design* is often used to mean both architecture and design. In the recent past, professionals used the term *design* when they dealt with simpler technological products - ones that do not include several different and interconnected technological components such as hardware, software, operators, services, etc. In the development of new multi-technology products and services, professionals have recognized the usefulness of the notion of *system* in dealing with complexity (interconnections level, multi-techno, emergence, etc.).

It was due to complexity that structuring the elements that comprise a system became necessary. This structure explains the functional, behavioral, temporal, physical, and other aspects of a system as described in System Architecture. Practitioners found the term *structure* inadequate to describe all these aspects of a system. The terms *architecture* and *architectural design* have been used for approximately 30 years, especially in software intensive systems and other domains, such as the space industry. The set of different types and interrelated structures can be understood as the architecture of the system.

The trend today is to consider system architecture and system design as different and separate sets of activities, but concurrent and strongly intertwined.

System design includes activities to conceive a set of system elements that answers a specific, intended purpose, using principles and concepts; it includes assessments and decisions to select system elements that compose the system, fit the architecture of the system, and comply with traded-off system requirements. It is the complete set of detailed models,

properties, and/or characteristics described into a form suitable for implementation.

Design Characteristics and Design Enablers

Every technological domain or discipline owns its peculiar laws, rules, theories, and enablers concerning transformational, structural, behavioral, and temporal properties of its composing parts of materials, energy, or information. These specific parts and/or their compositions are described with typical design characteristics and enablers. These allow achieving the implementation of every system element through various transformations and exchanges required by design characteristics (e.g., operability level, reliability rate, speed, safeguard level) that have been assigned during the system architecture definition process.

The design definition provides the description of the design characteristics and design enablers necessary for implementation. Design characteristics include dimensions, shapes, materials, and data processing structures. Design enablers include formal expressions or equations, drawings, diagrams, tables of metrics with their values and margins, patterns, algorithms, and heuristics.

- Examples of generic design characteristics in mechanics of solids: shape, geometrical pattern, dimension, volume, surface, curves, resistance to forces, distribution of forces, weight, velocity of motion, temporal persistence
- Examples of generic design characteristics in software: distribution of processing, data structures, data persistence, procedural abstraction, data abstraction, control abstraction, encapsulation, creational patterns (e.g., builder, factory, prototype, singleton), and structural patterns (e.g., adapter, bridge, composite, decorator, proxy)

Relation with System Architecture

System design is intended to be the link between the system architecture (at whatever point this milestone is defined in the specific application of the systems engineering process) and the implementation of technological system elements that compose the physical architecture model of the system.

Design definition is driven by specified requirements, the system architecture, and more detailed analysis of performance and feasibility. It addresses the implementation technologies and their assimilation. Design provides the “how-” or “implement-to” level of the definition.

Design concerns every system element composed of implementation technologies, such as mechanics, electronics, software, chemistry, human operations and services for which specific engineering processes are needed. System design provides feedback to the parent system architecture to consolidate or confirm the allocation and partitioning of architectural characteristics and design properties to system elements.

Design Descriptor

A design descriptor is the set of generic design characteristics and of their possible values. If similar, but not exact system elements exist, it is possible to analyze these in order to identify their basic characteristics. Variations of the possible values of each characteristic determine potential candidate system elements.

Holistic Design

Holistic design is an approach that considers the system being designed as an interconnected whole, which is also part of something larger. Holistic concepts can be applied to the system as a whole along with the system in its context (e.g., the enterprise or mission in which the system participates), as well as the design of mechanical devices, the layout of spaces, and so forth. This approach often incorporates concerns about the environment, considering how the design will impact the environment and attempting to reduce environmental impact. Holistic design is about more than merely trying to meet the system requirements.

Process Approach

Purpose

The purpose of the System Design process is to provide sufficient detailed data and information about the system and its system elements to enable the implementation

consistent with architectural entities as defined in models and views of the system architecture (ISO/IEC/IEEE 15288 [ISO 2015]).

Generic inputs include architecture description of the parent system and system element requirements.

Generic outputs are the description of the design characteristics and design enablers necessary for implementation.

Activities of the Process

Major activities and tasks to be performed during this process include the following:

1. Initialize design definition

- Plan for technology management for the whole system. Identify the technologies (mechanics, electricity, electronics, software, biology, operators, etc.) that would compose and implement the system elements and their physical interfaces.
- Determine which technologies and system elements have a risk to become obsolete or evolve during the operation stage of the system. Plan for their potential replacement.
- Identify types of design characteristics or properties for each technology of each system element.
- Periodically assess design characteristics and adjust as the system evolves.
- Document the design definition strategy, including the need for and requirements of any enabling systems, products, or services to perform the design.

2. Establish design characteristics and design enablers related to each system element

- Perform, consolidate or detail system requirements allocation to system elements for all requirements and system elements not fully addressed in the System Architecture process (normally, every system requirement would have been transformed into architectural entities and architectural characteristics within the System Architecture process, which are then allocated to system elements through direct assignment or some partitioning).

- Define the design characteristics relating to the architectural characteristics and check that they are implementable. Use design enablers, such as models (physical and analytical), design heuristics, etc. If the design characteristics are not feasible, then assess other design alternatives or implementation option, or perform trades of other system elements definition.
- Define the interfaces that were not defined by the System Architecture process or that need to be refined as the design details evolve. This includes both internal interfaces between the system elements and the external interfaces with other systems.
- Record the design characteristics of each system element within the applicable artifacts (they depend on the design methods and techniques used).
- Provide rationale about selection of major implementation options and enablers.

3. Assess alternatives for obtaining system elements

- Identify existing implemented system elements (COTS/NDI, reused, or other non-developed system elements). Alternatives for new system elements to be developed may be studied.
- Assess design options for the system element, using selection criteria that are derived from the design characteristics.
- Select the most appropriate alternatives.
- If the decision is made to develop the system element, the rest of the design definition process and the implementation process are used. If the decision is to buy or reuse a system element, the acquisition process may be used to obtain the system element.

4. Manage the design

- Capture and maintain the rationale for all selections among alternatives and decisions for the design, architecture characteristics, design enablers, and sources of system elements.
- Assess and control the evolution of the design characteristics, including the alignment with the architecture.
- Establish and maintain traceability between design

characteristics and architectural characteristics, and with requirements as necessary.

- Provide baseline information for configuration management.
- Maintain the design baseline and the design definition strategy.

Practical Considerations

Key pitfalls and proven practices related to system design are described in the next two sections.

Pitfalls

Some of the key pitfalls encountered in performing system design are provided in Table 1.

Table 1. Pitfalls with System Design.(SEBoK Original)

Pitfall	Description
Consider the design of each system element separately	This would be conducted using heterogeneous implementation of a given technology or between technologies within the system-of-interest. The design strategy for the complete system is defined to find synergies and/or commonalities that could help operation and maintenance of system elements.

Proven Practices

Some proven practices gathered from the references are provided in Table 2.

Table 2. Proven Practices with System Design.(SEBoK Original)

Practice	Description
Architecture and design mutual support	Discipline engineers perform the design definition of each system element; they provide strong support (knowledge and competencies) to systems engineers or architects in the evaluation and selection of candidate system architectures and system elements. Inversely, systems engineers, or architects, must provide feedback to discipline engineers to improve knowledge and know-how.

References

Works Cited

INCOSE. 2015. *INCOSE Systems Engineering Handbook*, Version 4. San Diego, CA, USA: International Council on Systems Engineering (INCOSE), INCOSE-TP-2003-002-03.2.2.

ISO/IEC/IEEE. 2015. *Systems and Software Engineering - System Life Cycle Processes*. Geneva, Switzerland: International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/Institute of Electrical and Electronics Engineers (IEEE). ISO/IEC/IEEE 15288:2015.

Faisandier, A. 2012. *Systems Architecture and Design*. Belberaud, France: Sinergy'Com.

Primary References

ISO/IEC/IEEE. 2015. *Systems and Software Engineering - System Life Cycle Processes*. Geneva, Switzerland: International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/Institute of Electrical and Electronics Engineers (IEEE). ISO/IEC/IEEE 15288:2015.

Faisandier, A. 2012. *Systems Architecture and Design*. Belberaud, France: Sinergy'Com.

Additional References

Baldwin, C.Y. and K.B. Clark. 2000. *Design Rules*. Cambridge, MA, USA: MIT Press.

Buede, D.M. 2009. *The Engineering Design of Systems: Models and Methods*. 2nd ed. Hoboken, NJ, USA: John Wiley & Sons Inc.

DoD. 2010. *DOD Architecture Framework*. Version 2.02. Arlington, VA, USA: US Department of Defense. Available at: <http://cio-nii.defense.gov/sites/dodaf20/>

< Previous Article | Parent Article | Next Article >

SEBoK v. 2.6, released 20 May 2022

Retrieved from

"https://www.sebokwiki.org/w/index.php?title=Detailed_Design_Definition&oldid=64996"

This page was last edited on 18 May 2022, at 09:41.