

# Functional Architecture

---

Functional Architecture

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 Author:** Caitlyn Singam and Jeffrey Carter

---

A system's functional architecture is the inter-related set of transformative processes and purposeful input-output tasks (i.e., functions) that a system performs, or can perform, on input(s) from external or internal sources in order to produce output(s) that supports the achievement of mission objectives. More simply, functional architecture defines the various actions a system can perform in support of specific goals, and how those actions relate to each other in order to collectively give the system the appropriate capabilities to meet those goals. The handling of internal inputs and outputs (such as those generated by and passed between sub-functions) are encompassed in functional architecture as well.

Since a system's functional architecture only defines the system in relation to transformative functions and their inputs/outputs, it is not in and of itself a complete definition of all aspects of the system design. Functional architecture is thus considered as an architectural "view" or "perspective", rather than as a standalone sub-unit of the overall system architecture. Other system architectural views, such as logical architecture and physical architecture, can be used to provide limited context for a given system's functional architecture without the need to examine all aspects of a complete system architecture.

Functional architectures are typically represented via abstract system models when used as part of documentation capturing and/or defining the characteristics and behaviors of a system of interest. Capturing a system's functional architecture is a key part of system architecture design definition and is typically done early on in a system's life cycle. The

closely associated task of functional architecture modeling is generally performed at the same time and in advance of any system development efforts, though there are instances (e.g., in studies of natural systems) where there may be an interest in documenting the functional architecture of a system already in operation. Once developed, functional architecture models provide a useful means of elucidating the processes which underlie a system's abilities and behaviors to stakeholders, and consequently can be used later on in the system life cycle as well, in support of verification and validation (V&V) efforts and system design review activities.

This article discusses how functional architecture relates as a concept to other facets of system architecture, and discusses key principles related to the development and documentation of functional architecture in practice.

□

## **Contents**

---

Purpose

    Relation to Behavioral Architecture

Application in practice

Modeling

References

    Works Cited

    Primary References

    Additional References

## **Purpose**

---

The purpose of functional architecture is essentially stated directly in the name: it defines what functions and sub-functions exist within a system of interest (SoI), and the metaphorical "layout" or architecture which connects or otherwise relates these functions. Specifically, these function definitions involve the identification of the system's inputs and outputs (both main and intermediary), and the expression of system activities in terms of the actions they perform, either directly or in association with auxiliary system processes, to transform those inputs into output elements and behavioral outcomes (ISO/IEC/IEEE 15288:2023). The idea of a "function" in the specific context of functional architecture is thus arguably closer to how it is used in mathematics (in the sense of "y is a function of x") or

computer programming than in other aspects of systems engineering, where "function" takes on a broader definition generally along the lines of actions performed in pursuit of a desired outcome (Hitchins 2008) and not directly tied to input-output transformations.

## **Relation to Behavioral Architecture**

Functional architecture is distinct from the closely related concept of behavioral architecture, which it can be easily conflated with due to the similar technical meanings of the terms "behavior" and "function". Both functional and behavioral architecture concern purposeful actions undertaken by a given system, and the manner in which those actions enable the production of observable features such as system modes and mode transitions, but differ in terms of the lenses through which they view and define the system's actions.

Functional architecture relates to input-output transformations: the manner in which a system operates on, and/or in relation to, intangible and tangible inputs from the entities, users, and environments which exist around it in the surrounding system context through the transformation of inputs into outputs. Behavioral architecture, in contrast, is more concerned with the sequencing and execution of system actions, and how system actions interact with checkpoint conditions and the initiation/completion of other system tasks in order to produce behavior.

## **Application in practice**

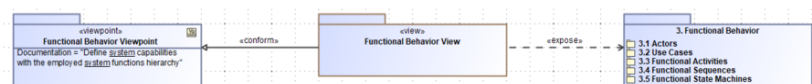
An overall system architecture is generally defined very early on in the system life cycle, as it plays a key role in translating the system concept definition into a viable system design (INCOSE 2023). Functional architecture, as one of many potential "views" of that system architecture, offers a lens through which one can consider the system in question in order to ensure that key elements of the system architecture are not omitted or overlooked, and to furthermore ensure that changes in aspects of the system architecture that were defined outside the scope of the functional view do not conflict or otherwise cause issues with the system's functionality.

The functional architecture view of a completed system architecture should, at minimum, define the following:

1. the functional capabilities that the system uses to

- fulfill mission objectives and meet stakeholder needs;
- 2. the required inputs and outputs for those system functions;
- 3. the steps through which the system will transform provided inputs into the desired outputs;
- 4. the grouping of those steps into independently executable sub-functions, if sub-functions are being employed, with corresponding definitions of the inputs and outputs for those sub-functions;
- 5. the variations in the inputs, outputs, and execution of the system's function/sub-functions for each system mode, if the system has multiple modes; and
- 6. the internal and external interfaces enabling input flow and output flow for each defined function, sub-function, and auxiliary function.

Given that functional architecture is merely a partial "view" of the overall system architecture, it is generally a good practice to consider a system's functional architecture in juxtaposition with other architectural views for context. Logical architecture, physical architecture, and functional architecture are often discussed in conjunction with each other (Pineda and Smith 2010; Boriky and Bradley 2019; Stief et al. 2018) since they provide distinct but mutually relevant perspectives of how the system's capabilities are implemented in a given system design (Broy et al. 2009). Taken together as a trio, they provide a fairly comprehensive portrait of the overall system architecture, and can be as key lenses through which to develop design concepts for new systems or examine the architecture of pre-existing ones. Figure 1 provides the functional architecture viewpoint and view within the system design model.



**Figure 1. Functional Architecture Viewpoint and View expressed in SysML. (SEBoK Original)**

## Modeling

A system's functional architecture is an abstract concept; consequently, while it might be possible to deduce the functional architecture of an existing system through observation and testing of the system while it is in operation (as might be necessary when examining/documenting a natural system), doing so is

generally a non-trivial, time-intensive task. Thus, a model of a system's functional architecture, which provides a representation of the system's functionality in a manner that can be accessed and understood by relevant parties, is a useful means through which individuals can interact with information about system's functional architecture concept. It is worth noting, however, that a functional architecture model is not equivalent to the system's functional architecture itself: while a system's functional architecture refers to the conceptual idea of how a system transforms its inputs into outputs, a functional architectural model serves as a representation or expression of that concept to communicate understanding of the system's functionality. As with any model, a functional architectural model will not necessarily capture every possible nuance of what is being modeled, and in cases where one is attempting to model the functional architecture of a system without complete knowledge of the system requirements, etc. that impacted the details system's functionality (natural systems again being an example of this), it is entirely feasible to have a functional architecture model that does not match the functional architecture of the system. The terms "functional architecture view" and "functional architecture model" should thus not be used interchangeably as the former refers an idea (a specific aspect of the overall system architecture), and the latter to a representation used to communicate the key aspects of that idea in a manner that may be imperfect and/or incomplete.

There are multiple options for how one can express a system's functional architecture in model form. The optimal type of model for representing any one given system's functional architecture is something that is usually best assessed on a case-by-case basis depending on the system and scope of the model's intended use. Text-based descriptive architecture models, for instance, are suitable solutions for systems with straightforward functional architectures that only involve a few functions and inputs/outputs since those architectures can be described in paragraph form without too much difficulty. In cases where the functional architecture involves a large number of conditional operations or highly interconnected processes, a visual diagram-based model (Youngs et al. 1999) or digital software model (Lemazurier, Chapurlat, and Grossetête 2017; Brinkkemper and Pachidi 2010) are likely to be preferable to text-based descriptions in terms of accurately representing the full gamut of system functionality while still making it feasible to extract

information from the model without undue effort (Nowodziński and Navas 2023; Younse, Cameron, and Bradley 2021). The principles of model-based systems engineering (MBSE) tend to be useful in ensuring the development of a quality system architecture model, and it is thus recommended for individuals to apply such principles when performing architecture modeling in various contexts (Estefan 2007; Kaslow et al. 2017). Software packages for generating digital MBSE-friendly models sometimes include tools that automatically check the model for errors (e.g., incorrectly expressed data flows), which can be useful when modeling informational flow through system functions.

For individuals using Unified Modeling Language (UML), Systems Modeling Language (SysML), or other standardized graphical modeling approaches to express their system concept, the functional architecture and behavioral architecture views are often considered and evaluated jointly due to their coordinated expression in SysML diagrams. Figures 2-5 are presented below for the reader's edification as illustrative exemplars of some of the diagram types of relevance in a functional architecture view of a SysML model, including a Use Case Diagram for a system (Figure 2), a Block Definition Diagram of hierarchical system functions (Figure 3), an Activity Diagram describing an individual system function (Figure 4), and a State Machine Diagram of the overall system (Figure 5). The diagrams below are presented in the example context of an automobile powertrain application.

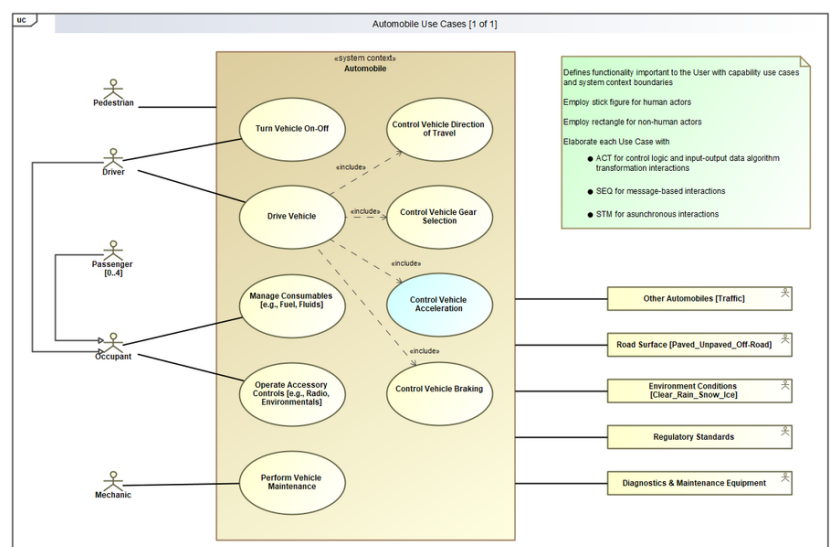
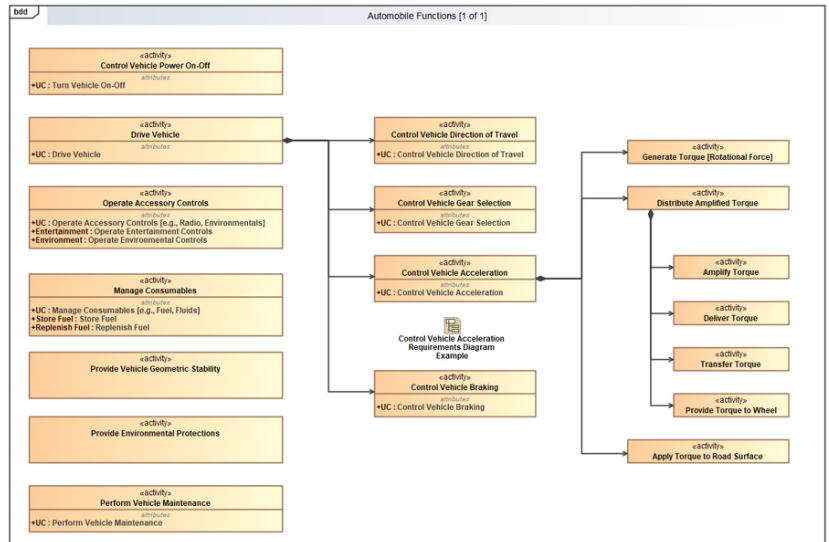
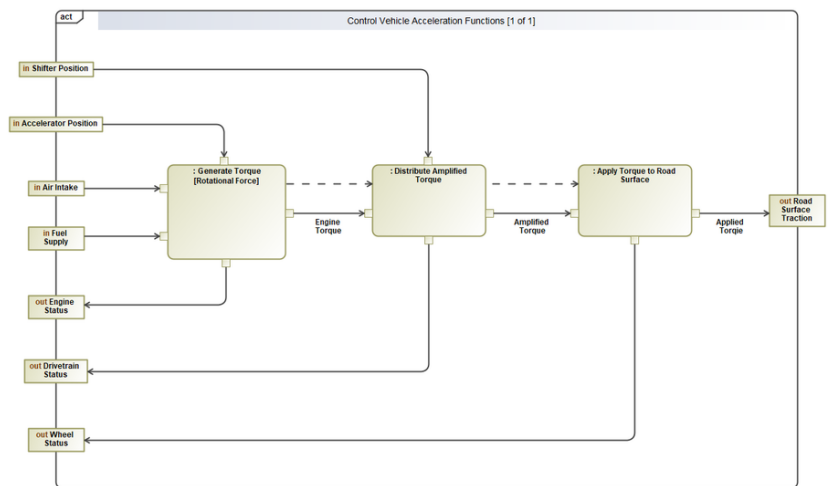


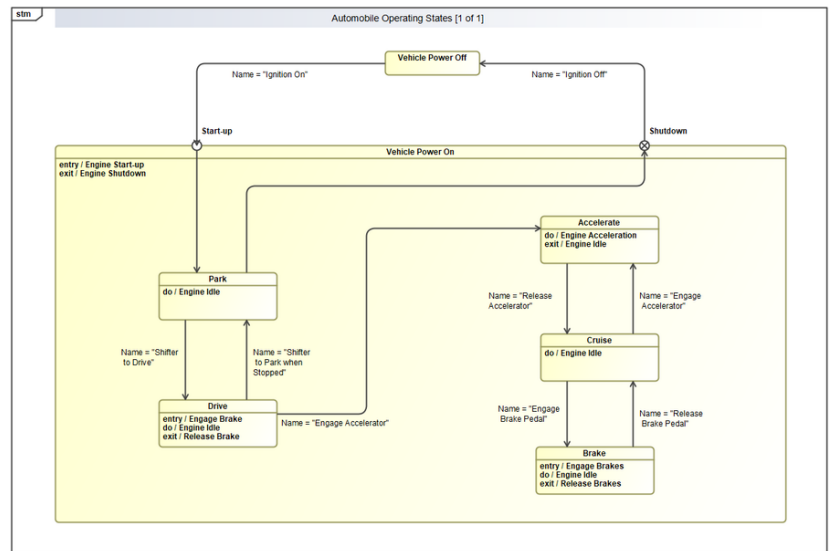
Figure 2. Example SysML Use Case Diagram for an Automotive System. (SEBoK Original)



**Figure 3. Example SysML Block Definition Diagram Representation of Hierarchical Functions for an Automotive System. (SEBoK Original)**



**Figure 4. Example SysML Activity Diagram for the Function "Control Vehicle Acceleration" in an Automotive System. (SEBoK Original)**



**Figure 5. Example SysML State Machine Diagram for an Automotive System. (SEBoK Original)**

## References

### Works Cited

- Borky, John M., and Thomas H. Bradley. 2019. "Designing in a Logical/Functional Viewpoint." In *Effective Model-Based Systems Engineering*, edited by John M. Borky and Thomas H. Bradley, 153-216. Cham: Springer International Publishing. [https://doi.org/10.1007/978-3-319-95669-5\\_5](https://doi.org/10.1007/978-3-319-95669-5_5).
- Brinkkemper, Sjaak, and Stella Pachidi. 2010. "Functional Architecture Modeling for the Software Product Industry." In *Software Architecture*, edited by Muhammad Ali Babar and Ian Gorton, 198-213. Berlin, Heidelberg: Springer. [https://doi.org/10.1007/978-3-642-15114-9\\_16](https://doi.org/10.1007/978-3-642-15114-9_16).
- Broy, Manfred, Mario Gleirscher, Stefano Merenda, Doris Wild, Peter Kluge, and Wolfgang Krenzer. 2009. "Toward a Holistic and Standardized Automotive Architecture Description." *Computer* 42 (12): 98-101. <https://doi.org/10.1109/MC.2009.413>.
- Estefan, Jeff A. 2007. "Survey of Model-Based Systems Engineering (MBSE) Methodologies." *INCOSE MBSE Focus Group* 25 (8): 1-12.
- Hitchins, Derek K. 2008. *Systems Engineering: A 21st Century Systems Methodology*. John Wiley & Sons.
- INCOSE, ed. 2023. *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and*



*Activities*. 5th edition. Hoboken, NJ: Wiley.

Kaslow, David, Bradley Ayres, Philip T. Cahill, Laura Hart, and Rose Yntema. 2017. "A Model-Based Systems Engineering (MBSE) Approach for Defining the Behaviors of CubeSats." In *2017 IEEE Aerospace Conference*, 1-14. <https://doi.org/10.1109/AERO.2017.7943865>.

Lemazurier, L., V. Chapurlat, and A. Grossetête. 2017. "An MBSE Approach to Pass from Requirements to Functional Architecture." *IFAC-PapersOnLine*, 20th IFAC World Congress, 50 (1): 7260-65. <https://doi.org/10.1016/j.ifacol.2017.08.1376>.

Nowodzienski, Pierre, and Juan Navas. 2023. "From Model-Based to Model and Simulation-Based Systems Architectures—Achieving Quality Engineering through Descriptive and Analytical Models." *INSIGHT* 26 (1): 40-50. <https://doi.org/10.1002/inst.12428>.

Pineda, Ricardo L., and Eric D. Smith. 2010. "Functional Analysis and Architecture." In *Systems Engineering Tools and Methods*. CRC Press.

Stief, Paul, Jean-Yves Dantan, Alain Etienne, and Ali Siadat. 2018. "A New Methodology to Analyze the Functional and Physical Architecture of Existing Products for an Assembly Oriented Product Family Identification." *Procedia CIRP*, 28th CIRP Design Conference 2018, 23-25 May 2018, Nantes, France, 70 (January): 47-52. <https://doi.org/10.1016/j.procir.2018.02.026>.

Youngs, R., D. Redmond-Pyle, P. Spaas, and E. Kahan. 1999. "A Standard for Architecture Description." *IBM Systems Journal* 38 (1): 32-50. <https://doi.org/10.1147/sj.381.0032>.

Younse, Paulo J., Jessica E. Cameron, and Thomas H. Bradley. 2021. "Comparative Analysis of a Model-Based Systems Engineering Approach to a Traditional Systems Engineering Approach for Architecting a Robotic Space System through Knowledge Categorization." *Systems Engineering* 24 (3): 177-99. <https://doi.org/10.1002/sys.21573>.

## **Primary References**

ANSI/IEEE. 2000. *Recommended Practice for Architectural Description for Software-Intensive Systems*. New York, NY, USA: American National

Standards Institute (ANSI)/Institute of Electrical and Electronics Engineers (IEEE), ANSI/IEEE 1471-2000.

INCOSE. 2023. *Systems Engineering Handbook - A Guide for System Life Cycle Processes and Activities*, version 5.0. Hoboken, NJ, USA: John Wiley and Sons, Inc, ISBN: 978-1-119-81429-0.

ISO/IEC/IEEE. 2015. *Systems and Software Engineering -- System Life Cycle Processes*. Geneva, Switzerland: International Organisation for Standardisation / International Electrotechnical Commissions / Institute of Electrical and Electronics Engineers. ISO/IEC/IEEE 15288:2015.

ISO/IEC/IEEE. 2011. *Systems and Software Engineering - Architecture Description*. Geneva, Switzerland: International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/Institute of Electrical and Electronics Engineers (IEEE), ISO/IEC/IEEE 42010.

## **Additional References**

None.

---

< [Previous Article](#) | [Parent Article](#) | [Next Article](#) >  
**SEBoK v. 2.11, released 25 November 2024**

---

Retrieved from  
"[https://sebokwiki.org/w/index.php?title=Functional\\_Architecture&oldid=73170](https://sebokwiki.org/w/index.php?title=Functional_Architecture&oldid=73170)"

---

**This page was last edited on 24 November 2024, at 19:36.**