

SYSTEM ARCHITECTURE

Definition: *An architecture is "the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution [1, 2]."*

Keywords: *architecture*

MITRE SE Roles & Expectations: MITRE systems engineers are expected to understand the role that an architecture plays in system development (e.g., conceptualization, development, and certification), the various purposes for architecture, and the different types of architectures. They are also expected to understand various architecture frameworks, models and modeling, views and viewpoints, as well as when and why each would apply. MITRE systems engineers are expected to understand different architectural approaches and their applications, including the use of architectural patterns.

Context

At this point in the systems engineering life cycle, an operational need has been expressed and turned into a concept and set of operational requirements (refer to "[Concept Development](#)" topic). They are then analyzed and transformed into a set of system requirements (refer to "[Requirements Engineering](#)" topic). The next step is to develop an architecture (or update an existing architecture for fielded systems) as a basis or foundation to guide design and development.

This topic contains articles on Architectural Frameworks, Models, and Views, Approaches to Architecture Development, and Architectural Patterns.

The article [Architectural Frameworks, Models, and Views](#) discusses the ways in which an architecture can be described. Various frameworks are used in different domains. Two well-known examples are the Zachman framework and the Department of Defense Architecture Framework. Whatever their specific form, all frameworks focus on defining a set of models, views, and viewpoints to support a range of systems engineering and program management activities and decisions across the system life cycle.

The article [Approaches to Architecture Development](#) provides an overview of ways to tailor and apply architecture approaches, process, and methodologies to support decision making.

Architectural patterns are a method of arranging blocks of functionality. They can be used at the subsystem (component), system, or enterprise level. The article [Architectural Patterns](#) describes patterns and discusses how they can simplify and expedite the development process.

Architecture Best Practices

- Ensure purpose before architecting.
- Ensure that stakeholders have an opportunity to vet architectural trade-offs as they occur.
- Evaluate the architecture throughout system development. Although an architecture is intended to be a persistent framework during the life cycle (and life) of a system, unforeseen changes (e.g., new missions) can influence the best of "first version" architectures.
- Construct the architecture to help understand technology readiness and evolution, and avoid getting locked in to proprietary or potentially obsolete technologies or captured by a specific vendor.

References & Resources

- ANSI/IEEE 1471-2000, "Recommended Practice for Architecture Description of Software-Intensive Systems."
- ISO/IEC 42010:2007, "Systems and Software Engineering—Recommended Practice for Architectural Description of Software-Intensive Systems."

Additional References & Resources

Bass, L., P. Clements, and R. Kazman, December 30, 1997, *Software Architecture in Practice*, 1st edition, Addison-Wesley Professional.

"System Architecture," *MITRE Project Leadership Handbook*, accessed February 23, 2010.

"Architecture," *MITRE Systems Engineering Competency Model*, v 1.13, section 2.3, accessed February 23, 2010.

"Concept Development," *MITRE Systems Engineering Guidebook*, accessed February 23, 2010.