Fachhochschule Nordwestschweiz
Hochschule für Technik

Software Architecture

Exercise – Software Architecture Definition

BSc

Ingo Arnold

Exercise Opening
Overview

There are many different answers to the question of **what exactly software architecture is** — wickedly put, as many as there are software architects.

One world-renowned institution that has long been involved in software architecture is the **Software Engineering Institute (SEI)** at Carnegie Mellon University in Pittsburgh, Pennsylvania.

Various such **software architecture definitions** have been collected and made available to the general public ([SEI Software Architecture 2021])

Exercise Agenda

- Software Architecture Definition

Software Architecture Definition
Exercise

There are many different answers to the question of **what exactly software architecture is** — wickedly put, as many as there are software architects.

One world-renowned institution that has long been involved in software architecture is the **Software Engineering Institute (SEI)** at Carnegie Mellon University in Pittsburgh, Pennsylvania.

Various such **software architecture definitions** have been collected and made available to the general public ([SEI Software Architecture 2021])

Software Architecture Definition
Exercise

However, the SEI also offers its own definition[1] (text 1):

“The software architecture of a program or computing system is a depiction of the system that aids in the understanding of how the system will behave. Software architecture serves as the blueprint for both the system and the project developing it, defining the work assignments that must be carried out by design and implementation teams. The architecture is the primary carrier of system qualities such as performance, modifiability, and security, none of which can be achieved without a unifying architectural vision. Architecture is an artifact for early analysis to make sure that a design approach will yield an acceptable system. By building effective architecture, you can identify design risks and mitigate them early in the development process.”

[1] http://www.sei.cmu.edu/architecture/

Software Architecture Definition
Exercise

Does this definition contradict common agile process attitude and models? Architecture (as defined by the SEI) sounds like a lot of *planning ahead*. Some thoughts worth reading on the role of architecture and architects in the article *who needs an architect* ([Fowler 2003]) in which Martin Fowler embeds postings by Ralph Johnson (text 2):

> “In most successful software projects, the expert developers working on that project have a shared understanding of the system design. This shared understanding is called *architecture*. This understanding includes how the system is divided into components and how the components interact through interfaces. These components are usually composed of smaller components, but the architecture only includes the components and interfaces that are understood by all the developers.”

Software Architecture Definition
Exercise

What are the relationships between architecture and code? What significance does the architecture still have when a system has been completely programmed? In this context, the abstract of a presentation at the Java User Group Switzerland (JUGS) by Simon Brown (software architecture vs. code[1]) is worth reading (text 3):

“Software architecture and coding are often seen as mutually exclusive disciplines, despite us referring to higher level abstractions when we talk about our software. You've probably heard others on your team talking about components, services and layers rather than objects when they're having discussions. Take a look at the codebase though. Can you clearly see these abstractions or does the code reflect some other structure? If so, why is there no clear mapping between the architecture and the code? Why do those architecture diagrams that you have on the wall say one thing whereas your code says another?”

[1] https://www.jug.ch/html/events/2015/software_architecture_vs_code.html

Software Architecture Definition
Exercise

Finally, another interesting angle via which Ralph Johnson marks an elementary difference between traditional architecture or engineering disciplines and software architecture in Martin Fowler's article ([Fowler 2003]) (text 4):

“Software is not limited by physics, like buildings are. It is limited by imagination, by design, by organization. In short, it is limited by properties of people, not by properties of the world. We have met the enemy, and he is us.”

An angle Melvin Convey had already picked up in *how do committees invent?* ([Convey 1968]) (text 5):

“The basic thesis of this article is that organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.”

Software Architecture Definition

Exercise

What is the essence of texts 1, 2, 3, 4, and 5? Summarize the key messages of each text compactly.

Bibliography
Lecture

[SEI Software Architecture 2021]
Software Engineering Institute, *What is your definition of software architecture?*, https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=513807, 2021

[Fowler 2003]
Fowler, Martin, *Who needs an Architect?*, https://martinfowler.com/ieeeSoftware/whoNeedsArchitect.pdf, 2003

Questions