Types of Software Architects

<https://medium.com/@nvashanin/types-of-software-architects-aa03e359d192>

<https://medium.com/@nvashanin/the-path-to-becoming-a-software-architect-de53f1cb310a>

Let’s continue investigating Software Architecture. In every field of professional activity there are various specializations. For example, in medicine we have surgery, cardiology, ophthalmology and many other areas. Or, for example, management unites resource managers, PR managers, even cleaning managers!

Specialization is necessary when the amount of knowledge in the field exceeds a rational limit. And since the software architecture is a huge amount of knowledge, it is important to reduce the duties of a person for better productivity. Therefore, the types of software architects will be discussed in this article.

### Types of architects

Let’s consider [the following division](https://en.wikipedia.org/wiki/Software_architect):

● **System architect.**

○ Affects one system and builds connections within it.

○ Focuses on the technical component of the development.

○ Helps the project manager to make management decisions.

○ Has deep knowledge of the technologies.

● **Solution architect.**

○ Participates in discussions of business.

○ Creates connections between several systems.

○ Provides communication between several teams.

○ Designs connections between systems.

○ Codes independently only solution prototypes.

○ Acts as a universal soldier of business and technology.

● **Enterprise architect.**

○ Affects all development of the company.

○ Works with high-level abstractions of the created systems.

○ Provides technical communications throughout the company.

○ Does not interact with the code.

○ Focuses on the business component.

○ Has a broad technical horizon.

○ Owns several domains.

### Domain architects

I often heard the question — is it possible for an architect to exist in mobile applications? And this question is so resonant that it is extremely difficult to get a reasoned response to it. Can there be an architect who knows only the JavaEE technology stack, or just .NET? How these specialists should be called? Tech leads or something else?

While thinking, during my own career, how to answer this question, I changed my mind several times. In my discussions with specialists who hold the positions of highly specialized architects, they naturally tried to give arguments in order to justify their position. Others said it was pointless and you should call such a person anything, but not an architect.

Consider a situation where a three-person startup has a CEO (chief executive officer), a CTO (chief technical officer), and a CMO (chief marketing officer). Can there be a technical officer for one developer? Or a marketing officer for a department of one specialist? Is it possible to determine whether a fictional rank is in pursuit of a beautiful name or a real necessity?

Let’s try to answer these questions. If there is a post, then there must be job descriptions that the specialist does. For architects, we described them in [the previous article](https://medium.com/@nvashanin/the-path-to-becoming-a-software-architect-de53f1cb310a).

In large projects, there are posts whose instructions coincide with those described in the above article. If the project consists of one or more platforms, each of them requires an expert who will act as a domain architect and perform the following duties:

● **Identifying the stakeholders on the project.** It is important to note that a domain architect should select stakeholders that affect his platform and work with them.

● **Identifying business requirements and requirements of the stakeholders on the project**. If the architect finds platform-specific stakeholders, then he/she should find the requirements with the restrictions, specifically for the domain.

● **Designing the entire system based on the received requirements.**For a domain architect, it is more important not to design an entire system, but to integrate the platform into the project outline. And also, to consider the connections between the components that affect the platform.

**● Choosing the technologies for the implementation of each component and connections between the components.**Unlike higher-level architects, a domain architect, as a rule, has the greatest impact on the choice of application technologies for her/his platform. For example, in mobile applications the architect deals with such issues as: what kinds of testing to use on the project, whether it needs code generation, how to organize the service and presentation layers, what architectural patterns to use and why it is generally necessary for the project.

**● Writing project documentation and its support.** If there is an architecture, then it should be documented. Even in terms of one platform. As written in the book “[Software Architecture in Practice](https://www.amazon.com/Software-Architecture-Practice-3rd-Engineering/dp/0321815734)”, if the architecture is not documented, then it is not architecture. In the mobile development, it can be a scheme for working with databases, a description of network interactions, class diagrams and so on.

● **Creating unified development standards in the company.** This point is especially suitable for a domain architect, because all standards are usually developed for a specific platform.

● **Controlling the architecture during the next iteration of the system release**.

A domain architect must control the entire product development cycle. And being the closest person to the technical component of the platform and, at the same time, seeing it as a whole picture, the architect is fully responsible for the quality of the product on a particular platform.

As a result, there is a large number of different types of domain architects:

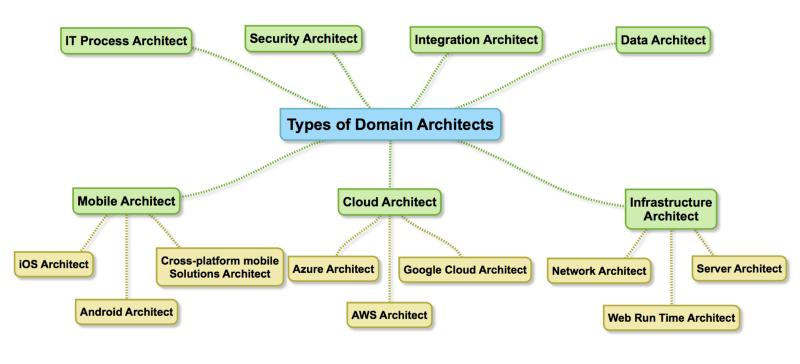


Image 2. Some types of domain architects

The picture shows only a small part of them. In reality, there are much more of them, as well as various technology stacks.

On the other hand, it is necessary to add the position of the architect only when the corresponding responsibilities are apparent. If you have a project from one platform and two developers, adding an architect’s position is redundant and one of the developers can perform this duties.

### Growing from one type of architect to another

To consider this topic, let’s use the concept of the [T-Shape model](https://en.wikipedia.org/wiki/T-shaped_skills). It assumes that a specialist can grow vertically and horizontally. Increasing the vertical line means improving the skills within your specialization, that is, Deep Knowledge. Increasing the horizontal line means getting skills and experience in a wide variety of domain areas and technological stacks, that is, the Wide Experience. By combining these two features, you can get information about the current level of knowledge and experience of the employee at any time.

For example, developer level growth is described by a vertical bar. The developer learns new knowledge about frameworks, language, and development tools within the limits of one technical stack. The architect, however, must possess not only deep knowledge, but also have broad technical erudition across several platforms.

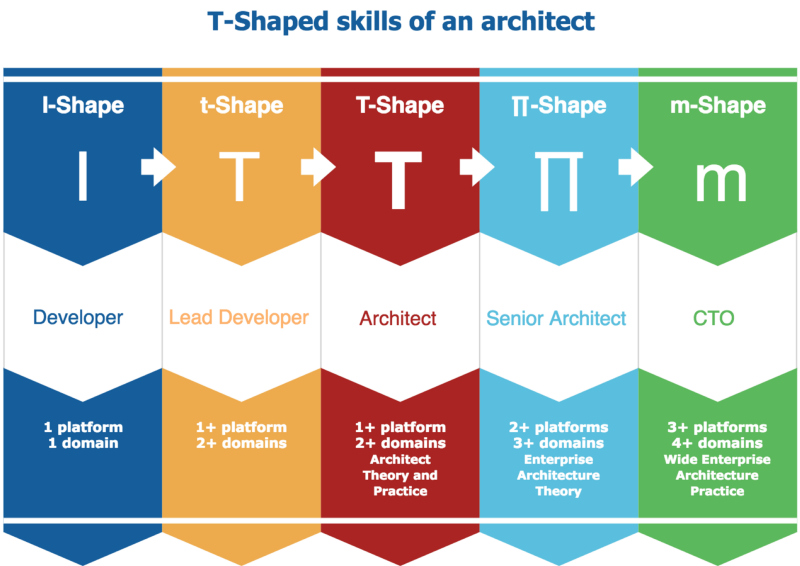


Image 3. T-Shaped skills of an architect

At the same time, the architect’s growth is closely related to the number of platforms in which he/she has theoretical and mainly practical expertise, and also, with the number of subject areas in which he/she know oats. The goal of the architect’s career development is the formation of the “m” — multiplatform & multidomain specialist.

Hereby we finish consideration of the architect types and their growth perspectives. In the next article, we will consider the quality attributes and their usage in the software architecture.