

[Open in app](#)[Sign up](#)[Sign in](#)

Most Common Software Architecture Styles

Dr Milan Milanović · [Follow](#)

5 min read · Dec 22, 2023



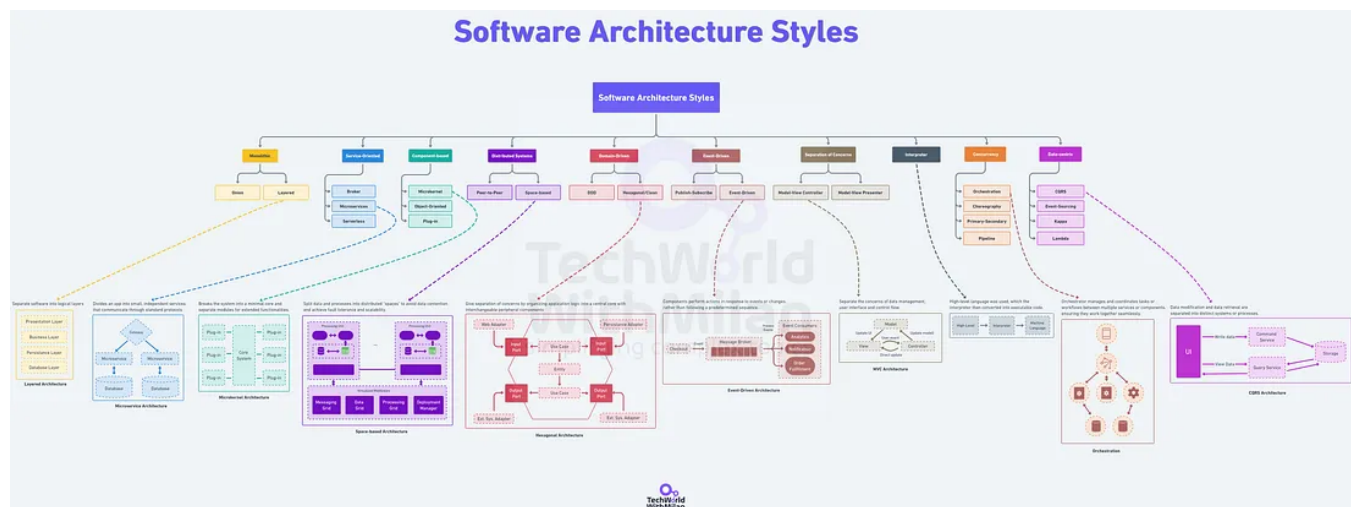
Listen



Share

Software **architecture styles** are the foundational blueprints for constructing various software systems, ensuring they meet specific requirements and quality attributes. By adhering to a suitable architecture style, organizations can ensure that their software systems are built to align with their strategic goals, accommodate future changes, and are resilient in the face of evolving technological landscapes and user demands.

An **architectural pattern**, on the other hand, communicates a fundamental organizational structure for software systems. By selecting the appropriate patterns for your issue, you can avoid creating anything from scratch and potentially dangerous traps that might arise if you devise a novel solution.



Most common software architecture styles

Here are the most common architectural styles:

1. **Monolithic:** Builds the entire application as a single unit, where all functionalities and components are managed and served from one place. Examples of monolithic architectures are **Onion** and **Layered**.

***Pros:** simple to implement, enhanced maintainability, as issues or updates can be addressed in specific modules or layers without affecting the entire system, and improved reusability and scalability, as modules can be reused in different projects and layers can be independently scaled or modified.*

***Cons:** performance issues due to data transmission overhead between layers and complexity in managing and ensuring interaction and consistency among various modules and layers.*