Large Scale Systems Architecture

Professor
Dr. Sahar Adabi

Author
Mohammad Khorshidi

Course Overview

Contents

Chapter 1

Introduction to Software Architecture

Large-scale systems architecture refers to the design and management of complex computing systems that are capable of handling vast amounts of data and serving numerous users simultaneously. These systems are typically distributed, comprising multiple components that work together to provide seamless services at scale. The architecture of such systems must ensure high availability, fault tolerance, scalability, and performance, often under demanding conditions. As more applications move to the cloud and become data-intensive, building robust architectures becomes critical to maintaining both user experience and system reliability.

In large-scale systems, several key tasks need to be addressed, including the management of distributed databases, load balancing, horizontal scaling, and efficient communication between system components. Furthermore, designing these systems requires careful attention to the trade-offs between consistency, availability, and partition tolerance (the CAP theorem). Architecting for resilience against failure, ensuring security, and optimizing for performance are also essential considerations.

1.1 Software Architecture Document

A Software Architecture Document (SAD) is a critical part of the software development process. It provides a comprehensive description of the system's architecture, serving as a vital communication tool between different teams and stakeholders involved in the project. The SAD helps document the design decisions and illustrates how the system should be structured and implemented to meet the given requirements.

The document typically covers several key sections:

- **Introduction and Overview:** This section offers a general introduction to the system, outlining its objectives and goals. It provides a broad view of the architecture and specifies the intended audience and the document's purpose.

- **Architectural Overview:** In this part, a high-level representation of the system architecture is provided, explaining the major components and how they interact. This section usually includes diagrams that illustrate the main modules, services, and their connections.

- **Logical View:** The logical view describes the abstract structures of the system and the relationships between different components. It focuses on how the system is organized in terms of modules, classes, and software interfaces (APIs), often supported by class diagrams.

- **Physical View:** This section explains how the software components are distributed across the hardware infrastructure. It covers details about servers, networks, and physical deployment environments, helping to visualize how the system will be deployed in a real-world setting.

- **Process View:** Here, the document describes the dynamic aspects of the architecture, particularly how different processes run and interact during runtime. It also addresses concurrency and how the system manages processes during execution.

- **Quality Attributes:** One of the most important parts of the SAD is the description of the system's quality attributes. This includes requirements for performance, security, scalability, reliability, and maintainability, which guide the architectural decisions.

- **Design Considerations and Decisions:** In this section, key design decisions made during the architecture process are documented. It includes explanations for the choice of technologies, tools, frameworks, and any trade-offs or constraints the team faced.

- **Integration with Other Systems:** This section covers how the system will interact with external systems or components. It highlights any dependencies and describes how integration will be achieved.

The SAD plays a crucial role in ensuring transparency in design decisions and serves as a roadmap for developers during the implementation phase. Moreover, it helps mitigate risks and challenges early in the process and facilitates easier maintenance and updates in the future.