Architectural Design Overview

# Overview

Architectural design is the first phase in the software design process. It focuses on organizing the overall structure of a software system, acting as a bridge between requirements engineering and detailed design. The outcome is an architectural model outlining how components communicate within the system.

# Levels of Software Architecture

1. Architecture in the Small: Deals with the structure of individual programs and how they are broken down into components.

2. Architecture in the Large: Focuses on complex, distributed systems that consist of multiple programs and components across different computers.

# Benefits of Explicit Architecture Design

1. Stakeholder Communication: A high-level system overview for discussions among various stakeholders.

2. System Analysis: Helps ensure early decisions align with performance, reliability, and maintainability requirements.

3. Large-Scale Reuse: Supports reuse of architectural models in systems with similar requirements.

# Architectural Models and Diagrams

Block diagrams are often used to model architectures informally. However, these diagrams may not capture all relationships or properties between components. More detailed and formal architectural descriptions reduce misunderstandings but are time-consuming and expensive to create.

# Architectural Design Decisions

Designing architecture involves creatively organizing systems to meet both functional and non-functional requirements. The design is influenced by the system's nature, domain, and the architect’s experience.

Key Non-Functional Requirements Impacting Architecture:

- Performance  
- Security  
- Safety  
- Availability  
- Maintainability

# Architectural Views

Different perspectives are necessary for comprehensive architecture documentation:  
1. Logical View: Shows key system abstractions (e.g., object classes).  
2. Process View: Depicts runtime interactions between system processes.  
3. Development View: Breaks down the system for implementation by development teams.  
4. Physical View: Maps software components to the system hardware for deployment planning.

By considering these views, the architecture can be thoroughly documented and communicated to all relevant parties.