CCS3300: Software Architecture

Week 04 – In-class Activity

22UG2-0035 Sahan Hansaja
22UG2-0009 Deemantha Rathnayake
22UG2-0560 Adithya Ramanayake
22UG2-0004 Lahiru Dilshan

---

1. Overview of middleware technologies
    a. What is middleware and what role does it play in modern software architecture?
    b. How does middleware contribute to system scalability and maintainability?
    c. What are the key characteristics that define effective middleware solutions?
    d. How has the evolution of middleware technologies been influenced by changing business requirements?
    e. What are the primary challenges organizations face when implementing middleware solutions?
2. Technology classification
    a. How are middleware technologies typically classified, and what are the main categories?
    b. What distinguishes synchronous middleware from asynchronous middleware, and when should each be used?
    c. How do object-oriented middleware technologies differ from service-oriented middleware approaches?
    d. What role does platform independence play in middleware technology classification?
    e. How has the emergence of cloud computing influenced middleware technology classification?
3. Distributed objects
    a. What are distributed objects, and how do they enable transparent remote method invocation?
    b. How does CORBA (Common Object Request Broker Architecture) facilitate interoperability between different programming languages and platforms?
    c. What are the advantages and limitations of using Java RMI for distributed object computing?
    d. How do distributed object systems handle issues like network failures, object lifecycle management, and security?
    e. What factors contributed to the decline in popularity of traditional distributed object technologies, and what has replaced them?
4. Message oriented middleware
    a. What is Message Oriented Middleware (MOM), and how does it differ from other middleware approaches?

b. How do message queues work, and what benefits do they provide in distributed systems?
c. What are the key differences between point-to-point and publish-subscribe messaging patterns?
d. How do modern message brokers like Apache Kafka and RabbitMQ address scalability and reliability challenges?
e. What role does message transformation and routing play in enterprise integration scenarios?