

Course Title: Distributed Systems

Course Type: Main Course

Unit Type: Theoretical

Units: 3

Prerequisites: Advanced Computer Architecture

Project: None

Teaching Hours: 48

Objectives:

Distributed systems are common. Computer scientists and engineers need to understand what are the principles and paradigms underlying distributed systems software and be familiar with several real world examples. This course systematically examines the underlying principles and how they are applied to a wide variety of distributed systems in depth.

Syllabus:

1. **Introduction:** Definition, Goals, Hardware and Software Concepts, Client-Server Model
2. **Communication:** Layered Protocols, Remote Procedure Call, Remote Object Invocation, Message-Oriented Communication, Stream-Oriented Communication
3. **Processes:** Threads, Clients, Servers, Code Migration, Software Agents
4. **Naming:** Naming Entities, Locating Mobile Entities, Removing Unreferenced Entities
5. **Synchronization:** Clock Synchronization, Logical Clocks, Global State, Election Algorithms, Mutual Exclusion, Distributed Transactions
6. **Consistency and Replication:** Data- And Client-Centric Consistency Models, Distribution Protocols, Consistency Protocols
7. **Fault Tolerance:** Process Resilience, Reliable Client-Server and Group Communication, Distributed Commit, Recovery
8. **Security:** Secure Channels, Access Control, Security Management, Kerberos
9. **Exemplar Distributed Systems:** Object-Based, Document-Based, Coordination-Based

1. Andrew S. Tanenbaum, Maarten Van Steen, *Distributed Systems: Principles and Paradigms*, 3rd ed., Prentice Hall (2017)
2. G. Coulouris, Jean Dollimore, *Distributed Systems Concepts and Design 5th ed.*, ISBN: 0132143011, Addison Wesley (2011)
3. Maarten Van Steen, Andrew S. Tanenbaum, *Reliable Distributed Systems: Technologies, Web Services, and Applications*, Springer (2005)
4. Hagit Attiya, Jennifer Welch, *Distributed Computing: Fundamentals, Simulations, and Advanced Topics*, John Wiley (2004)