

Syllabus

Course Code	Course/Subject Name	Credits
CSC802	Distributed Computing	4

Objectives :

1. To provide students with contemporary knowledge in distributed systems.
2. To equip students with skills to analyze and design distributed applications.
3. To provide master skills to measure the performance of distributed synchronization algorithms.

Outcomes : On successful completion of course learner will be able to:

1. Demonstrate knowledge of the basic elements and concepts related to distributed system technologies ;
2. Illustrate the middleware technologies that support distributed applications such as RPC, RMI and Object based middleware.
3. Analyze the various techniques used for clock synchronization and mutual exclusion.
4. Demonstrate the concepts of Resource and Process management and synchronization algorithms.
5. Demonstrate the concepts of Consistency and Replication Management.
6. Apply the knowledge of Distributed File System to analyze various file systems like NFS, AFS and the experience in building large-scale distributed applications

Module No.	Unit No.	Topics	Hours
1.0	Introduction to Distributed Systems	1.1 Characterization of Distributed Systems : Issues, Goals, and Types of distributed systems, Distributed System Models, Hardware concepts, Software Concept.	06
		1.2 Middleware : Models of Middleware, Services offered by middleware, Client Server model. (Refer Chapter 1)	
2.0	Communication	2.1 Layered Protocols, Interprocess communication (IPC) : MPI, Remote Procedure Call (RPC), Remote Object Invocation, Remote Method Invocation (RMI)	10
		2.2 Message Oriented Communication, Stream Oriented Communication, Group Communication (Refer Chapter 2)	
3.0	Synchronization	3.1 Clock Synchronization, Logical Clocks, Election Algorithms, Mutual Exclusion, Distributed Mutual Exclusion-Classification of mutual Exclusion Algorithm, Requirements of Mutual Exclusion Algorithms, Performance measure.	

(Book Code : ME90A)

Module No.	Unit No.	Topics	Hours
		3.2 Non Token based Algorithms : Lamport Algorithm, Ricart-Agrawala'sAlgorithm, Maekawa's Algorithm	10
		3.3 Token Based Algorithms : Suzuki-Kasami'sBroadcast Algorithms,Singhal'sHeurastic Algorithm, Raymond's Tree based Algorithm,Comparative Performance Analysis (Refer Chapter 3)	
4.0	Resource and Process Management	4.1 Desirable Features of global Scheduling algorithm, Task assignmentapproach, Load balancing approach, load sharing approach.	06
		4.2 Introduction to process management, process migration, Threads.Virtualization, Clients, Servers, Code Migration. (Refer Chapter 4)	
5.0	Consistency, Replication and Fault Tolerance	5.1 Introduction to replication and consistency, Data-Centric and Client-Centric Consistency Models, Replica Management.	08
		5.2 Fault Tolerance: Introduction, Process resilience, Reliable client-server andgroup communication, Recovery. (Refer Chapter 5)	
6.0	Distributed File Systems and Name Services	6.1 Introduction and features of DFS, File models, File Accessing models,File-Caching Schemes, File Replication, Case Study: Distributed FileSystems (DSF), Network File System (NFS), Andrew File System (AFS)	12
		6.2 Introduction to Name services and Domain Name System, DirectoryServices, Case Study: The Global Name Service, The X.500 DirectoryService	
		6.3 Designing Distributed Systems: Google Case Study (Refer Chapter 6)	
Total			52

**Module 1**

Chapter 1 : Introduction to Distributed System		1-1 to 1-40
1.1	Introduction.....	1-1
1.2	Examples of Distributed System	1-3
1.2.1	Internet	1-3
1.2.2	Intranet	1-3
1.2.3	Mobile and Ubiquitous Computing	1-3
1.3	Resource Sharing and the Web	1-4
1.3.1	World Wide Web (WWW).....	1-4
1.4	Issues and Goals	1-5
1.4.1	Heterogeneity	1-5
1.4.2	Making Resources Accessible	1-5
1.4.3	Distribution Transparency	1-6
1.4.4	Scalability	1-7
1.4.5	Openness	1-9
1.5	Types of Distributed Systems	1-10
1.5.1	Distributed Computing Systems.....	1-10
1.5.1(A)	Cluster Computing Systems	1-10
1.5.1(B)	Grid Computing Systems	1-11
1.5.2	Distributed Information Systems	1-12
1.5.2(A)	Transaction Processing Systems.....	1-12
1.5.3	Enterprise Application Integration	1-13
1.5.4	Distributed Pervasive Systems	1-14
1.6	Distributed System Models	1-14
1.6.1	Architectural Models	1-15
1.6.1(A)	Software Layers.....	1-15
1.6.1(B)	System Architecture.....	1-16
1.6.1(C)	Variations in Basic Client Server Model	1-17
1.6.1(D)	Interfaces and Objects.....	1-18
1.6.1(E)	Design Requirements for Distributed Architectures.....	1-18
1.6.2	Fundamental Models	1-20
1.6.2(A)	Interaction Model	1-20
1.6.2(B)	Failure Model	1-22
1.6.2(C)	Security Model	1-24
1.7	Hardware Concepts	1-24
1.7.1	Multiprocessors.....	1-25
1.7.2	Homogeneous Multicomputer Systems.....	1-26
1.7.3	Heterogeneous Multicomputer Systems.....	1-27



1.8	Software Concepts.....	1-27
1.8.1	Distributed Operating Systems	1-28
1.8.2	Network Operating Systems	1-33
1.9	Middleware	1-34
1.9.1	Positioning Middleware	1-34
1.10	Models of Middleware	1-35
1.11	Services Offered by Middleware.....	1-35
1.12	Client Server Model	1-37
1.12.1	Clients and Servers.....	1-37
1.12.2	Application Layering.....	1-38
1.12.3	Client Server Architectures	1-39

Module 2**Chapter 2 : Communication****2-1 to 2-25**

2.1	Layered Protocols	2-1
2.1.1	Lower Layer Protocols	2-2
2.2	Interprocess Communication (IPC)	2-2
2.2.1	Types of Communication	2-2
2.2.2	Message Passing Interface	2-3
2.3	Remote Procedure Call (RPC)	2-3
2.3.1	RPC Operation	2-3
2.3.2	Implementation Issues	2-4
2.3.3	Asynchronous RPC	2-5
2.3.4	DCE RPC	2-5
2.4	Remote Object Invocation	2-7
2.4.1	Distributed Objects (RMI : Remote Method Invocation)	2-7
2.5	Message-Oriented Communication	2-10
2.5.1	Persistence Synchronicity in Communication	2-10
2.5.2	Combination of Communication Types	2-11
2.5.3	Message-Oriented Transient Communication	2-13
2.5.4	Message-Oriented Persistent Communication	2-15
2.6	Stream-Oriented Communication	2-18
2.6.1	Continuous Media Support	2-18
2.6.2	Streams and Quality of Service (QoS)	2-19
2.6.3	Stream Synchronization	2-21
2.7	Group Communication	2-21
2.7.1	Application-Level Multicasting	2-22
2.7.2	Gossip-Based Data Dissemination	2-23

**Module 3****Chapter 3 : Synchronization****3-1 to 3-28**

3.1	Clock Synchronization	3-1
3.1.1	Physical Clocks.....	3-2
3.1.2	Global Positioning System (GPS)	3-2
3.1.3	Clock Synchronization Algorithms.....	3-3
3.2	Logical Clocks.....	3-6
3.2.1	Lamport's Logical Clocks	3-7
3.2.2	Application of Lamport Timestamp : Total Order Multicasting	3-8
3.2.3	Vector Clocks	3-9
3.3	Election Algorithms	3-9
3.3.1	Bully Algorithm.....	3-10
3.3.2	Ring Algorithm	3-11
3.3.3	Elections in Wireless Networks	3-12
3.4	Mutual Exclusion.....	3-14
3.4.1	Distributed Mutual Exclusion	3-14
3.4.2	Classification of Mutual Exclusion Algorithms	3-14
3.4.3	Requirements of Mutual Exclusion Algorithms	3-15
3.4.4	Performance Measure of Mutual Exclusion Algorithms	3-15
3.4.5	Performance in Low and High Load Conditions.....	3-15
3.5	Non-Token Based Algorithms	3-16
3.5.1	Lamport's Algorithm	3-16
3.5.2	Ricart-Agrawala's Algorithm	3-17
3.5.3	Centralized Algorithm.....	3-18
3.5.4	Maekawa's Algorithm.....	3-19
3.6	Token Based Algorithms.....	3-20
3.6.1	Suzuki-Kasami's Broadcast Algorithm	3-20
3.6.2	Singhal's Heuristic Algorithm	3-21
3.6.3	Raymond's Tree-Based Algorithm	3-23
3.6.4	Token Ring Algorithm	3-26
3.7	Comparative Performance Analysis of Algorithms	3-26
3.7.1	Response Time.....	3-26
3.7.2	Synchronization Delay	3-27
3.7.3	Message Traffic	3-27

Module 4

4-1 to 4-27

Chapter 4 : Resource and Process Management

4.1	Introduction.....	4-1
4.2	Desirable Features of Global Scheduling Algorithm	4-1
4.3	Task Assignment Approach	4-2
4.4	Load-Balancing Approach	4-3
4.4.1	Classification of Load Balancing Algorithms.....	4-3
4.4.2	Issues in Designing Load-Balancing Algorithms.....	4-4
4.5	Load-Sharing Approach	4-7
4.5.1	Issues in Designing Load-Sharing Algorithms.....	4-7
4.6	Introduction to Process Management.....	4-8
4.7	Process Migration	4-9
4.7.1	Desirable Features of Good Process Migration Mechanism	4-9
4.7.2	Process Migration Mechanisms	4-10
4.7.3	Process Migration in Heterogeneous System.....	4-13
4.7.4	Advantages of Process Migration.....	4-13
4.8	Threads	4-14
4.8.1	Comparison between Process and Thread	4-14
4.8.2	Server Process	4-15
4.8.3	Models for Organizing Threads	4-16
4.8.4	Issues in Designing a Thread Package	4-16
4.8.5	Thread Scheduling.....	4-17
4.8.6	Implementing a Thread Package	4-17
4.9	Virtualization	4-18
4.9.1	The Role of Virtualization in Distributed Systems.....	4-19
4.9.2	Architectures of Virtual Machines.....	4-19
4.10	Clients.....	4-20
4.10.1	Networked User Interfaces.....	4-20
4.10.2	Client-Side Software for Distribution Transparency	4-20
4.11	Servers	4-21
4.11.1	General Design Issues.....	4-21
4.11.2	Server Clusters	4-22
4.11.3	Distributed Servers	4-23
4.12	Code Migration.....	4-23
4.12.1	Approaches to Code Migration.....	4-23
4.12.2	Migration and Local Resources.....	4-25
4.12.3	Migration in Heterogeneous Systems.....	4-26

Module 5

Chapter 5 : Consistency, Replication and Fault Tolerance

5-1 to 5-34

5.1	Introduction to Replication and Consistency	5-1
5.1.1	Reasons for Replication	5-1
5.1.2	Replication as Scaling Technique	5-2
5.2	Data-Centric Consistency Models	5-2
5.2.1	Continuous Consistency	5-3
5.2.2	Consistent Ordering of Operations	5-4
5.3	Client-Centric Consistency Models	5-9
5.3.1	Eventual Consistency	5-9
5.3.2	Monotonic Reads	5-10
5.3.3	Monotonic Writes	5-10
5.3.4	Read Your Writes	5-11
5.3.5	Write Follows Reads	5-11
5.4	Replica Management	5-12
5.4.1	Replica-Server Placement	5-12
5.4.2	Content Replication and Placement	5-13
5.4.3	Content Distribution	5-14
5.5	Fault Tolerance	5-16
5.5.1	Basic Concepts	5-16
5.5.2	Failure Models	5-17
5.5.3	Failure Masking by Redundancy	5-18
5.6	Process Resilience	5-19
5.6.1	Design Issues	5-19
5.6.2	Failure Masking and Replication	5-19
5.6.3	Agreement in Faulty Systems	5-20
5.6.4	Failure Detection	5-22
5.7	Reliable Client-Server Communication	5-22
5.7.1	Point-to-Point Communication	5-23
5.7.2	RPC Semantics in the Presence of Failures	5-23
5.8	Reliable Group Communication	5-25
5.8.1	Basic Reliable-Multicasting Schemes	5-25
5.8.2	Scalability in Reliable Multicasting	5-26
5.8.3	Atomic Multicast	5-27
5.9	Recovery	5-30
5.9.1	Introduction	5-30
5.9.2	Stable Storage	5-31
5.9.3	Checkpointing	5-32
5.9.4	Message Logging	5-32
5.9.5	Recovery-Oriented Computing	5-33

**Module 6****Chapter 6 : Distributed File Systems and Name Service****6-1 to 6-30**

6.1	Introduction.....	6-1
6.2	Desirable Features of a Good Distributed File System.....	6-2
6.3	File Models	6-3
6.3.1	Unstructured and Structured Files.....	6-3
6.3.2	Mutable and Immutable Files	6-4
6.4	File-Accessing Models	6-4
6.4.1	Accessing Remote Files	6-4
6.4.2	Unit of Data Transfer.....	6-4
6.5	File-Caching Schemes.....	6-5
6.5.1	Cache Location.....	6-5
6.5.2	Modification Propagation.....	6-6
6.5.3	Cache Validation Schemes.....	6-7
6.6	File Replication	6-8
6.6.1	Replication and Caching	6-9
6.6.2	Advantages of Replication	6-9
6.6.3	Replication Transparency	6-9
6.6.4	Multicopy Update Problem	6-10
6.7	Case Study : Distributed File Systems (DFS).....	6-12
6.7.1	Network File System (NFS).....	6-12
6.7.2	Andrew File System (AFS).....	6-16
6.8	Introduction to Name Services and Domain Name System.....	6-17
6.8.1	Names, Identifiers and Addresses	6-17
6.8.2	Name Services and the Domain Name System	6-18
6.8.3	Name Resolution	6-19
6.8.4	Domain Name System.....	6-20
6.9	Directory Services.....	6-22
6.10	The Global Name Service (GNS)	6-23
6.11	The X.500 Directory Service	6-24
6.12	Designing Distributed Systems : Google Case Study.....	6-25
6.12.1	Google Search Engine	6-25
6.12.2	Google Applications and Services.....	6-26
6.12.3	Google Infrastructure	6-27
6.12.4	The Google File System (GFS).....	6-28