Birla Institute of Technology & Science, Pilani
Hyderabad Campus
Computer Science and Information Systems Department
First Semester 2019-2020 Course Handout (Part II)

Date: 02/08/ 2019

In addition to Part-I (General handout for all courses appended to the timetable) this portion gives further specific details regarding the course:

COURSE NO.: CS G623

Advanced Operating Systems

Instructor In-Charge: **Chittaranjan Hota** (hota@hyderabad.bits-pilani.ac.in)

Scope:

Over the last few decades considerable amount of research has been done in "Distributed OS". The aim of this course is to introduce the design and implementation issues of Distributed OS. Distributed OS's work in an environment where we have independent machines (both hardware and software) connected with each other over a computer network. Distributed OSs have at their center the reasoning that you should use faster machines for more tasks that need speed, and slower ones for the tasks that don't. Also, central to the design of distributed OS's is making this design transparent to the user. A Distributed OS makes a Distributed System a virtual uniprocessor system.

Objectives:

- The distributed OS to be studied in this course is *microkernel* based. It's just that the user level processes that are separated from the kernel can run on remote machines.
- Few case studies like Sun NFS, HDFS, GFS, MapReduce, Vector Clocks, Causal ordering, Agreement protocols, Mutual Exclusion, distributed file systems etc. shall be discussed and also implemented as part of the coding practices in the laboratories.

Text Book:

T1 M. Singhal & N. Shivaratri, "Advanced Concepts in Operating Systems: Distributed, Database and Multiprocessor Operating Systems", Tata McGraw Hill, 2001.

Reference Books:

R1 Distributed Operating Systems – The Logical Design by A. Goscinski, AW
R2 Modern Operating Systems by A. S. Tanenbaum, PHI
R3 Distributed Systems-Concepts and Design by G. Coluris, AW

Plan of Study

S.No.	Learning Objectives	TOPIC	CHAPTER REF (Text)	Lect.s
1.	Why do we develop distributed operating systems?	What is a distributed system, Characteristics of a distributed system, Challenges in building distributed systems, Examples: Planetlab, Aadhaar, GARUDA, SETI@home etc.	Chapter 1	2
2.	Why do we need to have a virtualized environment, and what are its' characteristics?	Traditional Vs Virtualized architecture, Different types of hypervisors, Need for virtualization, Uses of Virtual machines	Class notes	2
3.	Why do we need logical clocks in building a distributed system? How does causal ordering of messages help?	Limitations of a distributed system, Lamport's logical clock, Fidge-Mattern's vector clock, causal ordering of messages, global state of a distributed system, Cuts.	Chapter 5	4

4.	How to solve the problem of critical section in a distributed environment?	Lamport's DME, Recart-agrawala, and Maekawa's algorithms; Suzuki-kasami broadcast algorithm, and Raymond's tree based algorithm.	Chapter 6	4
5.	Why deadlocks are seen as challenges in building distributed systems? How to handle those without using avoidance algorithms like Bankers?	Resource Vs. Communication deadlock, Strategies to handle deadlock, Ho-Ramamoorthy, Path-Pushing, Edge-Chasing, Diffusion Computation-based algorithms.	Chapter 7	4
6.	What is the importance of consensus in distributed systems?	System model, Classification of agreement problems, Solutions to Byzantine agreement problems.	Chapter 8	4
7.	Distributed transparent storage: how to access data stored at different machines? What are the characteristics of Bigdata?	Bigdata characteristics, Structured vs. unstructured data, Mechanisms for building DFSs, Design Issues, Case studies: Sun DFS, Sprite DFS, Hadoop Distributed File System (HDFS): Hadoop cluster, HDFS communications, HDFS read/write operations.	Chapter 9 and Class notes.	6
8.	Why do we need to run a single computation at multiple places?	Issues in Load Distribution, Components of a load distribution algorithm, Load Distribution Algorithms, Case studies: System-V, Sprite, and Condor, HDFS MapReduce: Programming on Hadoop.	Chapter 11 and Class notes.	6
9.	Research paper discussions.	Recent research on DFS and Distributed schedulers.	IEEE/ACM	2
9.	How to build a logical memory consisting of RAM contents of different machines?	Algorithms for implementing Distributed Shared Memories (DSMs), Memory Coherence models, and Coherence Protocols, Case study: Integrated Virtual Memory at Yale (IVY).	Chapter 10	4
10.	How to recover from failures in a distributed computation?	Classification of failures, Synchronous and Asynchronous Check Pointing and Recovery algorithms.	Chapter 12	4

Evaluation Scheme:

Sl No.	Component & Nature	Duration	Weightage	Date and Time
1.	Term paper presentation.	30 mins	10%	Will be announced in the class.
2.	Lab evaluations (Open Book)	2 hrs	25%	In the regular labs
3.	Technical report with implementation of the term paper.	-	10%	Take home
4.	Mid-Sem Test (Closed Book)	1.5 hours	25%	28/09 (11.00am-12.30pm)
5.	Comprehensive Exam (Open-50%+ Closed-50%)	3 hours	30%	02/12 (AN)

Coding labs for the course will be based on the design aspects of various components of Distributed Operating Systems like distributed middleware, thread synchronizations, logical clock implementations, distributed file systems, and distributed scheduling or load balancing etc. using HDFS/ MapReduce platforms.

Note: All notices related to the course will be displayed on the Computer Sc. Department's Notice Board, and / or course website at google class (class code: wjliee). Make ups shall be granted to genuine cases with a request for makeup reaching I/C on or before the test. There will be no make ups for lab evaluations.

Chamber Consultation Hour: Will be announced in the class.

Academic Honesty and Integrity Policy: Academic honesty and integrity are to be maintained by all the students throughout the semester and no type of academic dishonesty is acceptable.

Instructor-in-charge, CS G623