Distributed Systems

Instructor: Farnam Jahanian (farnam@umich.edu)
Schedule: TTH 10:30-12:00 EECS 3433
TH 5:00-6:00pm EECS 3427
Office: EECS 2229
Telephone: 734-936-2974

Course Description:

Significant advances in computer and communication technologies have enabled the discovery of several distributed computing paradigms and the development of numerous distributed systems during the last decade. The primary objective of this course is to examine the state of the art and practice in distributed computing and to provide students hands-on experience in developing distributed protocols. The course aims to convey insight into the principles underlying the design of distributed systems, enabling students to evaluate existing systems or design new ones.

This course presents the dominant paradigms for building distributed systems/applications including the traditional model of client-server computing, distributed objects, group communication, remote-procedure calls, distributed shared memory, and the evolving model of distributed computing on Web. The topics explored in this course include: kernel support for distributed systems, failure models and programming paradigms, group multicast protocols, multicast over the Internet, replicated data management based on the state-machine and the primary-backup schemes, group membership protocols, clock synchronization and logical timestamps, distributed synchronization, the design of fault-tolerant real-time systems, distributed objects systems, network security, and the Web-based client/sever models. Concepts are illustrated using several case studies of experimental and commercial systems, and discussions on emerging standards. Concepts are illustrated using several case studies of experimental and commercial systems.

The course also requires active student participation in a semester-long group project. Groups of no more than three students will be responsible for the design and development of a collection of distributed services such as data replication, caching, naming, group membership, object support and multicast communication. The group project is intended to complement the reading material by allowing the students develop experimental skills for designing and building distributed protocols. Each group will have an opportunity to present its work to the class.
Prerequisites:

- EECS 482 and graduate standing.
- Familiarity with UNIX or NT, and extensive C/C++ programming experience.
- Familiarity with Socket programming.

Reading List:

- Collection of papers and handouts on distributed systems.

Books on Reserve:


Grade Assignment:

The course grade is graded based on a final exam, several short quizzes, and a large programming project. The course will involve a substantial amount of time for the reading of material from the required text books and the assigned research papers.

- Project - Part I and II 40%
- Mid-term Exam 25%
- Final Exam 25%
- Quiz & Participation 10%
List of Topics

1. Characteristics of distributed systems, distributed programming models, failure models, synchronous vs. asynchronous

2. Communication issues and network technologies

3. Distributed computations, global system states, consistent cuts and runs

4. Client-Server Model, Remote Procedure Call

5. Fault-tolerant group communication: reliable, FIFO, causal, atomic, timed, and uniform

6. Distributed agreement (consensus)

7. Distributed Synchronization: Mutual Exclusion, Leader Election Algorithms, Clock synchronization protocols, Distributed Deadlock Detection/Prevention

8. Replication management:
   - state machine approach
   - primary-backup approach
   - gossip architecture
   - database techniques

9. Distributed shared memory

10. Distributed object-based systems

11. Client/Server computing on the Web

12. Multicast in wide-area networks

13. Routing protocols

14. Real-time fault-tolerant systems

15. Operating system support for distributed systems

16. Network security, intrusion detection systems

17. Checkpointing and recovery

Note: Several important topics such as distributed name service, distributed file systems and distributed transactions are covered in other graduate CSE courses.