

Southern Illinois University Carbondale

CS420: Introduction to Distributed Systems and Applications

Fall 2011

Instructor: Mengxia Zhu

Office: 2142 Faner

Phone: 453-6057

Office Hours: MWW 2pm~4pm or walk in at your convenience or by appointment

Email Address: mzhu@cs.siu.edu (Email is a good way to contact me!)

Web Page for CS420 <http://www.cs.siu.edu/~mengxia/Teaching.htm>

T.A: Dabin Ding

Office: Faner 3131

Office Hours: 1:30pm to 3:30pm Tuesday and Thursday

Email Address: dabin@siu.edu

Phone: (618) 453 6037

Texts:

- *Distributed Systems: Concepts and Design* by George Coulouris, Jean Dollimore and Tim Kindberg, 4th edition, Addison-Wesley, 2005.

With the fast development of the Internet and other Network-based applications and services, the design and implementation of distributed systems has become more and more important. This course emphasizes on the theoretical contents of algorithm design and complexity analysis and also has some practical exercises underlying distributed system.

Prerequisite:

CS335 or equivalent OS course

Written and Programming Assignments:

There will be many written assignments and a few programming assignments. Your written assignments and programs should be prepared in electronic documents submitted through Blackboard system. Upon receiving a graded assignment, if you have a dispute about the grading, please return the assignment to T.A. within a week, with a note explaining your concerns. T.A. will decide if he needs to re-grade it or not. The re-graded paper will be returned to you and the grade posted on the blackboard will be updated. If you still have questions, you can talk to me in person about the disputes during my office hours.

Final Project:

You will investigate a particular real-life distributed computing application that is of interest to you. I will also provide several candidate final projects for you to choose from. Design and

implement the system that integrates several techniques you learnt in the class. Project proposal should be submitted to get instructor's approval. You will write a report and give a presentation with live demo to the class during the last week of the semester. If you prefer, you can also implement a project designed by the instructor.

Quizzes:

Every Friday we will have a 10 minute quiz. The quizzes consist of questions on material covered in the past week. To be excused from a quiz, you must contact the instructor before the Friday class.

Exams:

There will be a Midterm exam and a Final exam. There will be no makeup exams. If you cannot take an exam during scheduled time, you should consult the instructor in advance.

Late Policy:

Homework and lab assignments must be submitted to blackboard by the due time unless you have prior approval of the instructor. The late policy for all assignments is as follows: 10% points off, if submitted within 24 hours after the due date; 25% off, if submitted 24-48 hours after the due date; no credit if submitted two days or more days after the due date unless prior arrangements are made with the instructor with acceptable reasons. Partially finished assignments will receive partial credit.

Academic Integrity:

Plagiarism behavior in any form is unethical and *will be punished*. All work submitted by a student (home works, projects, programming assignments, and exams etc.) has to be a student's own work. Students are allowed and encouraged to discuss with other students and look up resources in the literature for their assignments, but *appropriate references must be included for the materials consulted*, and appropriate citations should be made when the material is taken verbatim.

Definitions and policies on academic dishonesties are described in detail on the website (<http://www.cs.siu.edu/department/cheating.html>). The instructor will expect students to be aware of these guidelines and to conduct themselves accordingly. If cheating occurs, the student will receive a failing grade on the assignment and (at the instructor's discretion) a failing grade in the course. The instructor may also decide to forward the incident to the University Judicial Affairs Office for further disciplinary action.

Emergency Procedures:

Southern Illinois University Carbondale is committed to providing a safe and healthy environment for study and work. Because some health and safety circumstances are beyond our control, we ask that you become familiar with the SIUC Emergency Response Plan and Building Emergency Response Team (BERT) program. Emergency response information is available on posters in buildings on campus, available on BERT's website at www.bert.siu.edu, Department

of Safety's website www.dps.siu.edu (disaster drop down) and in Emergency Response Guideline pamphlet. Know how to respond to each type of emergency.

Instructors will provide guidance and direction to students in the classroom in the event of an emergency affecting your location. It is important that you follow these instructions and stay with your instructor during an evacuation or sheltering emergency. The Building Emergency Response Team will provide assistance to your instructor in evacuating the building or sheltering within the facility.

Grades:

The course grade will be distributed as follows:

Attendance	5%
Written and programming assignments	25%
Projects	15%
Quizzes	15%
Midterm Exam	20%
Final Exam	20%

Tentative Topics

Introduction

- What is distributed computing?
- Different forms of computing
- The weaknesses and strengths of distributed computing
- The architecture of distributed applications
- The challenges of distributed computing

Interprocess Communications (IPC)

- Definitions: IPC, unicast, multicast
- Client-server model
- Basic IPC Application Program Interface
- Event synchronization: block vs. non-blocking operations
- Deadlocks and timeouts
- Data marshalling; object serialization
- Protocols; text-based protocol; request-response protocol

Group Communications

- Unicasting versus Multicasting
- Multicast API operations: Join, leave, send, receive
- Reliable multicast vs. unreliable multicast
- Ordered delivery of multicast messages:
- The Java Basic Multicast API: InetAddress, DatagramPacket, MulticastSocket, time-to-live

Distributed Objects and Remote Invocation

- Message Passing versus Distributed Objects
- The Architecture of a Distributed Object System

- Remote Procedure Calls (RPC)
- Remote Method Invocations
- Java RMI: architecture, remote interfaces, rmiregistry, rmic, threaded execution of remote methods

Time and Global States

- Clocks, events and process states
- Synchronizing physical clocks
- Logical time and logical clock
- Global states

Coordination and Agreement

- Distributed Mutual exclusion
- Election
- Consensus

Transaction and Concurrency Control

- Transactions
- Locks
- Optimistic concurrency control
- Timestamp ordering
- Method comparison for concurrency control

Distributed Transactions

- Atomic commit protocols
- Concurrency control in distributed transactions
- Distributed deadlocks
- Transaction recovery

Peer-to-peer System

- Peer-to-peer middleware
- Routing overlays
- Application case studies

Distributed File System

- File service architecture
- Sun network file system
- Enhancement and further development