COURSE DESCRIPTION

Dept., Number	CSci 524	Course Title	Distributed Operating System Design
Semester hours	3	Course Coordinator	P. Tobin Maginnis, Associate Professor

Current Catalog Description

Analysis of operating system design principles for multiple computers. A distributed operating system model is presented and compared to selected network and distributed operating system examples.

Textbook

Tanenbaum, A. and Maarten van Steen, *Distributed Systems: Principles and Paradigms,* 2nd edition, Prentice-Hall, 2006.

Plus various research articles.

References

Class website: http://pix.cs.olemiss.edu/csci524/

Course Outcomes

Upon successful completion of this course, the students can:

1. describe categories of distributed and parallel computer architectures independent of circuit packaging;

2. compare and contrast various distributed operating system designs based upon resource sharing, reliability, and performance design goals;

3. describe system architecture and operating design parameters that significantly impact algorithm performance.

Relationship between Course Outcomes and Program Outcomes

This is a course taken primarily by computer science graduate students; it is sometimes taken by undergraduate computer science students as an elective to enrich their programs. The course outcomes contribute to the program outcomes as follows: (1) to (a); (2) to (c); and (3) to (b).

Prerequisites by Topic

Operating systems (CSci 423)

Major Topics Covered in the Course

1. Introduction to distributed systems 2. Performance, reliability, and resource sharing design trade offs 3. Transparency design strategies and interconnection trade offs 4. Communication models in distributed systems 5. Synchronization issues 6. Processes versus processors 7. Logical versus literal file systems 8. Distributed shared memory 9. Example systems

Assessment Plan for the Course

This is an elective course offered infrequently (e.g., only once in the past four years) and primarily to computer science graduate students. An offering typically has 4 examinations and a semester project. Outcome 1 is assessed by exam questions and 1 homework assignment, outcome 2 by exam questions and 3 homework assignments, and outcome 3 by exam questions and 1 homework assignment. The instructor evaluates the student performance informally and makes changes to the course content, organization, and pedagogy as indicated.

How Data in the Course are Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

Estimate Curriculum Category Content (Semester hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms			Software design		1
Data structures		1	Concepts of programming languages		1