CIS 630 - Fall 2005
Distributed (Operating) Systems

Lecture 1
Characterization of Distributed Systems

University of Oregon
Department of Computer and Information Science
Course Information

  - However, course description is correct.
- Now the required core course in graduate curriculum.
  - Replaces CIS 629.
  - Prerequisite: 415 or equivalent, 429/529.
- Instructor:
  - Prof. Allen D. Malony (malony@cs.uoregon.edu).
- Course webpage:
  - http://www.cs.uoregon.edu/classes/05F/cis630.
- Class: Tues./Thurs., 12:00-1:20 pm, 106 Condon.
Course Text

- Distributed Systems: Concepts and Design
  - G. Coulouris, J. Dollimore, T. Kindberg
  - Addison-Wesley, Fourth Edition
  - 2005

- Text webpage
  - [http://www.cdk4.net/](http://www.cdk4.net/)

- Many reference texts listed on the course webpage

- Will follow text closely
Lectures

☐ All lectures will use computer slide presentation
   ☑ All lectures slides will be posted on webpage
   ☑ No later than end of week of lecture

☐ Lecture content
   ☑ Some text / figures come from online book materials
   ☑ Other sources of lecture content will be cited

☐ Please do not waste paper printing the slides
Assignments

- Problem sets (2) (not graded) (5%)
  - Third and seventh week
  - Enforce topics and practice for exam
- Programming exercise (5%)
  - Java RMI client-server application
  - Early experience programming distributed systems
- Reading summaries (10%)
- Term exam (in class and take home) (25%)
- Term paper (25%)
- Term project (30%)
Reading Summaries

☐ Four to five research papers assigned to read
  ☐ Weekly, starting in third week
☐ Turn in two-page summaries
☐ Gain practice reading research papers
☐ Useful for term paper
Term Exam

- There will be one exam in the course
- It will take place in Week 9 (before Thanksgiving)
- It will cover topics through Week 8
  - Lecture topics
  - Research papers
  - Book chapters
- Two parts
  - Exam in class (open book)
  - Take-home problem (due 12:00 pm, Nov. 28)
Term Paper

☐ One of the main assignments in the course
☐ Give you an opportunity to explore a topic of interest
  ☐ Might not be covered by lecture or assigned readings
☐ Give you experience in reading research literature
☐ Give you experience in assimilating information
☐ You will present your paper in class
  ☐ During Week 10
☐ See course webpage for paper requirements
Term Project

- Second main assignment in course
- Enhance your knowledge of distributed systems
  - Hands-on distributed application development
- Done in teams
  - 3-4 people
  - Individual and group effort identified
  - Skills survey and team preferences
- Deliverables
  - Written report of accomplishments
  - Demonstration during finals week
  - Project presentations during final exam period
Course Topics

- Distributed system characterization
- Networking
- Distributed processing and programming
- Understanding time and global states
- Coordination and agreement
- Distributed data management and file systems
- Concurrency control and distributed transactions
- Replication
- Name services and peer-to-peer computing
- Web services and grid computing
Lecture Objectives

- Distributed system characteristics
  - components coordinating actions with messages
  - Concurrency
  - Independent failure and lack of global clock
- Place distributed systems in context
  - Internet, intranet, mobile computing
- Motivate benefits of resource sharing
  - Web as an example.
- Understanding of challenges for distribute systems
  - heterogeneity, openness, security, scalability
  - failure handling, concurrency, transparency
Characterization of Distributed Systems

- A distributed system is defined as one in which components (software primarily) at networked computers communicate and coordinate their actions only by passing messages.

- Definition allows for:
  - Concurrent execution of programs
  - Prevents possibility of a global clock
  - Means that components can fail independently

- Why construct and use distributed systems?
  - Stems from a desire to share resources
  - Coordinate distributed operations
Examples of Distributed Systems

- Internet
  - Very large collection of computer networks
  - Very large distributed system of networked computers
  - Enables users to make use of a vast number of services

- Intranet
  - Portion of Internet separately administered
  - Use firewall to enforce own local security policies
  - Supports standard and specialized services

- Mobile and ubiquitous computing
  - Nomadic computing, location-aware, embedded
  - Always present and available
Typical Portion of the Internet

* Graphics from Distributed Systems: Concepts and Design, Coulouris, Dollimore, and Kindberg
Typical Intranet

* Graphics from Distributed Systems: Concepts and Design, Coulouris, Dollimore, and Kindberg
Devices in a Distributed System

- Laptop
- Mobile phone
- Camera
- Internet
- Host intranet
- Wireless LAN
- Home intranet
- WAP gateway
- Printer
- Hosting site
- WAP: Wireless Access Protocol

* Graphics from Distributed Systems: Concepts and Design, Coulouris, Dollimore, and Kindberg
Resource Sharing

- Hardware sharing and computing/data sharing
- Variety of patterns and types of resource sharing
- A service manages a collection of related resources and presents their functionality to users / applications
  - Shared resources are managed by *server* processes
  - Accepts service requests from *client* processes running on other computers and responds accordingly
  - Well-defined set of operations
  - Requests are sent in messages
- Scalability of services is a key aspect
Types of Distributed Process Interactions

- Client-server systems
  - Clients request services from servers
- Versus peer-to-peer
  - Processes have equal status
- Resources may be encapsulated as objects
  - Methods are invoked by client objects
- Basis of distributed processing mechanisms
  - Naming
  - Management of state
WWW

- Evolving system for publishing and accessing resources and services across the Internet
- The WWW and the Internet are not the same thing
- The WWW is an open system
  - Extensible in services and service providers
  - Extensible in resource types and content
- Illustrates approach to addressing scale
  - Use of hierarchical naming
  - Partitioned data
  - Caching and replication
Web Servers, Web Browsers, and Web Services

Web servers

www.google.com

www.cdk3.net

www.w3c.org

File system of www.w3c.org

URL: Uniform (Universal) Resource Locator

Protocols

Activity.html

HTTP: HyperText Transfer Protocol

XML: eXtensible Markup Language

* Graphics from Distributed Systems: Concepts and Design, Coulouris, Dollimore, and Kindberg
Challenges

- Heterogeneity
- Openness
- Security
- Scalability
- Failure handling
- Concurrency
- Transparency
Transparencies

- Transparency hides the separation of components
- *Access transparency*: enables local and remote resources to be accessed using identical operations
- *Location transparency*: enables resources to be accessed without knowledge of their location
- *Concurrency transparency*: enables several processes to operate concurrently using shared resources without interference between them
- *Replication transparency*: enables multiple instances of resources to be used to increase reliability and performance without knowledge of the replicas by users or application programmers
Transparencies (continued)

- **Failure transparency**: enables the concealment of faults, allowing users and application programs to complete their tasks despite the failure of hardware or software components.
- **Mobility transparency**: allows the movement of resources and clients within a system without affecting the operation of users or programs.
- **Performance transparency**: allows the system to be reconfigured to improve performance as loads vary.
- **Scaling transparency**: allows the system and applications to expand in scale without change to the system structure or the application algorithms.