Distributed Systems
Lecture 1: Overview

CS425/CSE424/ECE428

Fall 2011

Nikita Borisov

Objectives

- Define *Distributed System*
- Overview of distributed systems issues
- Course information

Examples of Distributed Systems

Class suggestions:

- Internet
- Google File System
  - http://labs.google.com/papers/gfs.html
- World Wide Web
- US Postal Service
- Peer-to-peer Networks
- Email

Properties

Class suggestions:

- Multiple Machines
- Redundant / Fault-tolerant
- Complex coordination
- Consistency

Definitions

A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable

-- Leslie Lamport

Definitions

A distributed system consists of multiple autonomous computers that communicate through a computer network. The computers interact with each other in order to achieve a common goal.

-- *Wikipedia (as of today!)*

Definitions

A collection of (probably heterogeneous) automata whose distribution is transparent to the user so that the system appears as one local machine. This is in contrast to a network, where the user is aware that there are several machines, and their location, storage replication, load balancing and functionality is not transparent. Distributed systems usually use some kind of client-server organisation.

Distributed systems are considered by some to be the “next wave” of computing.

-- Free On-Line Dictionary of Computing (FOLDOC)

Definitions

A distributed system is a collection of independent computers that apperas to its users as a single coherent system

-- Tanenbaum & Steen

Definitions

We define a distributed system as one in which hardware or software components located at networked computers communicate and coordinate their actions only by passing messages

-- *Coulouris, Dollimore, Kindberg*

Key Properties

- ***Multiple* computers**
  - *Concurrent* execution
  - *Independent* failures
  - *Autonomous* administrators
  - *Heterogeneous* capacities, properties
  - *Large* numbers (scalability)
- ***Networked* communication**
  - *Asynchronous* execution
  - *Unreliable* delivery
  - *Insecure* medium
- ***Common* goal**
  - *Consistency* – can discuss whole-system properties
  - *Transparency* – can use the system without knowing details

Comparison – Operating Systems

- ***Multiple* computers**
  - *Concurrent* execution
  - *Independent* failures
  - *Autonomous* administrators
  - *Heterogeneous* capacities, properties
  - *Large* numbers (scalability)
- ***Networked* communication**
  - *Asynchronous* execution
  - *Unreliable* delivery
  - *Insecure* medium
- ***Common* goal**
  - *Consistency* – can discuss whole-system properties
  - *Transparency* – can use the system without knowing details

Comparison – Networking

- ***Multiple* computers**
  - *Concurrent* executio
  - *Independent* failure
  - *Autonomous* admini
  - *Heterogeneous* capacities, properties
  - *Large* numbers (scalability)
- ***Networked* communication**
  - *Asynchronous* execution
  - *Unreliable* delivery
  - *Insecure* medium
- ***Common* goal**
  - *Consistency* – can discuss whole-system properties
  - *Transparency* – can use the system without knowing details

Note: Networks use Distributed Algorithms (DNS, BGP)

Example: WWW

- ***Multiple* computers – Web servers, clients**
  - *Concurrent* execution
  - *Independent* failures
  - *Autonomous* administrators
  - *Heterogeneous* capacities, properties
  - *Large* numbers (scalability)
- ***Networked* communication – Internet (TCP/IP)**
  - *Asynchronous* execution
  - *Unreliable* delivery
  - *Insecure* medium (HTTPS)
- ***Common* goal – Hyperlinked information system**
  - *Consistency* – can discuss whole-system properties
  - *Transparency* – can use the system without knowing details

Example: Domain Name Service

- ***Multiple* computers – DNS server, clients, caches**
  - *Concurrent* execution
  - *Independent* failures
  - *Autonomous* administrators
  - *Heterogeneous* capacities, properties
  - *Large* numbers (scalability)
- ***Networked* communication – Internet (UDP + TCP/IP)**
  - *Asynchronous* execution
  - *Unreliable* delivery
  - *Insecure* medium (DNSSEC)
- ***Common* goal – Hierarchical Naming System**
  - *Consistency* – can discuss whole-system properties
  - *Transparency* – can use the system without knowing details

Example: Bank

- ***Multiple* computers – ATMs, teller computers, servers, credit card scanners**
  - *Concurrent* execution
  - *Independent* failures
  - *Autonomous* administrators
  - *Heterogeneous* capacities, properties
  - *Large* numbers (scalability)
- ***Networked* communication – Internet, local networks, modems, leased lines**
  - *Asynchronous* execution
  - *Unreliable* delivery
  - *Insecure* medium
- ***Common* goal – Financial Institution**
  - *Consistency* – can discuss whole-system properties
  - *Transparency* – can use the system without knowing details

Course Objective

- **Concepts** in distributed computing
  - Properties
  - Challenges
  - Impossibility results
- **Designs** of distributed systems
  - Abstractions
  - Algorithms
  - Implementations
- **Case studies**

Course Information: Staff

- Instructor: Prof. Nikita Borisov
  - Office: 460 Coordinated Science Lab
  - Office hours: 1:30–3:30PM Mondays
- TAs:
  - Ghazale Hosseinabadi
    - Office hours: 2–4PM Fridays
  - Sonia Jahid
    - Office hours: 3–5PM Wednesdays, 0207 Siebel

Sources of Information

- Course website: *(will be running by Thursday)*
  - Announcements, homework, MPs,
  - Lecture list, reading assignments, slides
- Course newsgroup: class.fa11.cs425
  - Announcements, questions, clarifications
    - Monitor daily; announcements will not be emailed
  - SLA: one business day response time
- Email: **`cs425-help@cs.illinois.edu`**
  - SLA: slower than newsgroup

Books

- *Distributed Systems: Concepts and Design*, Coulouris et al., 4th ed.
  - Earlier eds may be acceptable
  - Your responsibility to find correct homework questions & reading sections
- Other texts
  - *Distributed Systems: An Algorithmic Approach*, Ghosh
  - *Distributed Systems: Principles and Paradigms,* Tanenbaum & Steen
  - *Distributed Algorithms*, Lynch

Grade Components

Assignments

- Homeworks (16%)
  - Approx. every 2 weeks
  - Must be **typed**
  - Must be done **individually**
- MPs (32%)
  - 3 projects
  - Groups of 2

Exams

- Midterm (16%)
  - Date TBA
- Final (32%)
  - Friday, Dec 16, 7–10pm
  - (may be changed)

Participation (4%)

- i>Clickers (2%)
  - Available at bookstore
    - $36 new, $25 used
    - Can be re-sold
  - Review quiz at each lecture
  - Points for answering
  - No points for correctness
- Subjective participation (2%)
  - Lecture involvement
- Perfect attendance *not* needed to get full 4%

Grading

- Grades *may* be curved
- Undergrads & grads curved separately
- Academic integrity violations have serious consequences
  - Min: 0% on assignment
  - Max: expulsion
  - All cases are reported to CS, your college, and senate committee
- Note: any sharing of code outside group is forbidden

- Guaranteed grades:
  - >90%: A
  - >80%: B
  - >70%: C
  - >60%: D
  - >50%: pass

Acknowledgments

- Material borrowed from:
  - Prof. Jennifer Hou
  - Prof. Mehdi Harandi
  - Prof. Klara Nahrstedt
  - Prof. Indranil Gupta
  - Prof. Nitin Vaidya
  - Prof. Sayan Mitra

Lecture Summary

- *Distributed Systems* properties
  - Multiple computers
  - Networked communication
  - Common goal
- Course goals
  - Concepts, designs, case studies
- Your responsibilities
  - Read assigned sections
  - Monitor newsgroup
  - Participate in lectures

Next Lecture

- Failure Detection
  - Readings: §2.3.2, §12.1