CS550

• Advanced Operating Systems (Distributed Operating Systems)

• Instructor: Xian-He Sun
  – Email: sun@iit.edu, Phone: (312) 567-5260
  – Office hours: 1:30pm-2:30pm Tuesday, Thursday at SB229C, or by appointment

• TA: TBA
  – Email: xxx@iit.edu
  – Office: xxx
  – Office hours: TBA

• Blackboard:
  – http://blackboard.iit.edu

• Class Web site
Future Computing: Human-centered Service

A new IT booming is coming

Devices become smaller and powerful

They are connected to form `smart space’

A device is an entry of the cyber world

Grids link `smart spaces’ to support `global smartness’
What Is Computer Science?

Computer science is laying the foundations and developments the real search paradigms and scientific methods for the exploration of the world of information and intellectual processes that are not directly governed by physical laws.

By Juris Hartimanis, Turing Award Lecture

Such people (computer scientists) are especially good at dealing with situations where different rules apply in different cases; they are individuals who can rapidly change levels of abstraction, simultaneously seeing things “in the large” and “in the small”.

By Donald Knuth, Turing Award Lecture
What Is an OS?

An OS is a program that manages various computer resources
• A program that acts as an interface between users and bare hardware
• Resources: CPU(s), memory, file systems, I/O, etc.

• OS: software or hardware?
Figure 1  Layers and views of a computer system
Evolution of Operating Systems

Systems with single CPU
• Multitasking, multiprogramming

Systems with many CPUs
• Parallel processing and multiprocessors
• Networking and distributed systems
Technology Impacts

• CPU Technology
  – getting faster and less expensive, Moore’s law
  – used to be time-sharing (overhead for context switching)
  – Now space-sharing: systems with multiple CPUs
    – Multi-core, many-core architecture

• Memory Technology (memory wall)
  – Unbalanced technology advance
Processor-memory performance gap

- Processor performance increases rapidly
  - Uni-processor: ~52% until 2004, ~25% since then
  - New trend: multi-core/many-core architecture
    - Intel TeraFlops chip, 2007
  - Aggregate processor performance much higher
- Memory: ~9% per year
- Processor-memory speed gap keeps increasing
Memory Hierarchy

- Multiple levels of memory hierarchy
- Level closer to the CPU is faster to access
- Cache memories work well if spatial and temporal locality exists among data accesses
Memory Hierarchy

Deeper levels of cache memory
Large memories are slow, fast memories are small.

- **On chip Cache**
  - Speed (ns): 1ns
  - Size (bytes): 32KB

- **Second level cache (SRAM)**
  - Speed (ns): 10ns
  - Size (bytes): 1-2MB

- **Main memory (DRAM)**
  - Speed (ns): 100ns
  - Size (bytes): 1-8GB

- **Secondary storage (Disk)**
  - Speed (ns): 10 ms
  - Size (bytes): 20-200GB

- **Tertiary storage (Disk/Tape)**
  - Speed (ns): 10 sec
  - Size (bytes): TB
Contemporary memory hierarchy

- Registers
- Cache
- Main Memory
- Disk Cache
- Magnetic Disk
- Magnetic Tape
- Optical Disk

Layers:
- L1
- L2

Remote memory and remote disk connections.
The Principle of Locality

• The Principle of Locality:
  – Programs access a relatively small portion of the address space at any instant of time.

• Two Different Types of Locality:
  – **Temporal Locality** (Locality in Time): If an item is referenced, it will tend to be referenced again soon (e.g., loops, reuse)
  – **Spatial Locality** (Locality in Space): If an item is referenced, items whose addresses are close by tend to be referenced soon (e.g., straight line code, array access)
    • Cache Block or Cache Line

• Last 20 years, HW relied on locality for speed
Technology Impacts (cont’d)

• Disk (I/O wall)
  – large capacity, slow access time
  – was the most expensive item in computers
  – file system – file storage

• Storage is one of those technologies that we tend to take for granted. And yet, if we look at the true status of things today, storage is king. One can even argue that servers, which have become commodities, are now becoming peripheral to storage devices.

  --Michael Vizard
I/O Bottleneck

Technology Impacts (cont’d)

• Power Consumption (power wall)
  – Moore’s law
  – Green computing

• Networking (local area networks)
  – Speed increases faster than computing speed
  – diskless workstations
  – memory/disk sharing rather than CPU sharing
  – Remote data access
What Are We Doing with the Total System Silicon?

**Silicon Area Distribution**
- Memory: 86%
- Processors: 3%
- Routers: 3%
- Random: 8%

**Power Distribution**
- Memory: 9%
- Processors: 56%
- Routers: 33%
- Random: 2%
Latency Lags Bandwidth (last ~20 years)

• **Performance Milestones**
  
  • **Processor**: ‘286, ‘386, ‘486, Pentium, Pentium Pro, Pentium
  
  4 (21x,2250x)
  
  • **Ethernet**: 10Mb, 100Mb, 1000Mb, 10000 Mb/s (16x,1000x)
  
  • **Memory Module**: 16bit plain DRAM, Page Mode DRAM, 32b, 64b, SDRAM, DDR SDRAM (4x,120x)
  
  • **Disk**: 3600, 5400, 7200, 10000, 15000 RPM (8x, 143x)
Networking (Distributed Systems)

- A *distributed* system is a collection of processors that do not share memory or a clock. Each processor has its own local memory.
- The processors in the system are connected through a *communication network*.
- A distributed system provides user access to various system resources.
- Access to a shared resource allows:
  - Computation speed-up
  - Increased data availability
  - Enhanced reliability
Parallel Processing

• Parallel Processing
  – Several working entities work together toward a common goal

• Parallel Processing
  – A kind of information processing that emphasizes the concurrent manipulation of data elements belonging to one or more processes solving a single problem

• Parallel Computer
  – A computer designed for parallel processing
Multiprocessors: Shared-Memory Multiprocessors

- Uniform Memory Access (UMA)
- NonUniform Memory Access (NUMA)
Multiprocessors: Distributed-Memory

Multiprocessors

INTERCONNECTION NETWORK

Note: NO HARDWARE SUPPORT FOR REMOTE MEMORY ADDRESSING
Homogeneous Multicomputer Systems

a) 2D-mesh

(b) Hypercube
Hardware Concepts

Parallel and Distributed Systems

Multiprocessors (fast hw network)

Shared-Memory Multiprocessors

Distributed-Memory Multiprocessors

UMA

NUMA

Multicomputers (slow hw network)

Homogeneous Multicomputers

Heterogeneous Multicomputers

Network of Workstation

Workstation Cluster

PC Cluster

Distributed Systems

loosely coupled

tightly coupled
Software Concepts

- Operating system:
  - Interface between users and hardware
  - Implements a virtual machine that is easier to program than raw hardware
- Primary functions:
  - Services: file system, virtual memory, networking, CPU scheduling, ...
  - Coordination: concurrency, memory protection, security, networking, ...
Uniprocessor Operating Systems

Microkernel architecture
• Small kernel
• user-level servers implement additional functionality

![Diagram of microkernel architecture]

- No direct data exchange between modules
- User mode
- Kernel mode
- Hardware
- Microkernel
- File module
- Process module
- Memory module
- User application
- System call
- OS interface
Multiprocessor Operating Systems

- Like a uniprocessor operating system
- Manage multiple CPUs transparently to the user
- Each processor has its own hardware cache
  - Maintain consistency of cached data
  - Scalability issues
- Shared variable versus message passing
Multicomputer Operating Systems

• More complex than multiprocessor OS
  – Because communication has to be through explicit message passing
Network Operating System

- Machine A
- Machine B
- Machine C

Distributed applications

- Network OS services
- Kernel

Network
Network Operating System

- Employs a client-server model
  - Minimal OS kernel
  - Additional functionality as user processes
Network-Operating Systems

• Users are aware of multiplicity of machines. Access to resources of various machines is done explicitly by
  – Remote logging into the appropriate remote machine.
  – Transferring data from remote machines to local machines, via the File Transfer Protocol (FTP) mechanism.
Distributed Operating System

- Users not aware of multiplicity of machines.
- Manages resources in a distributed system
  - Seamlessly and transparently to the user
- Looks to the user like a centralized OS
  - But operates on multiple independent CPUs
- Provides transparency
  - Location, migration, concurrency, replication,…
- Presents users with a virtual uniprocessor
Middleware-based Systems

• General structure of a distributed system as middleware.

```
Machine A    Machine B    Machine C

Distributed applications

Middleware services

Network OS services  Network OS services  Network OS services

Kernel             Kernel             Kernel

Network
```