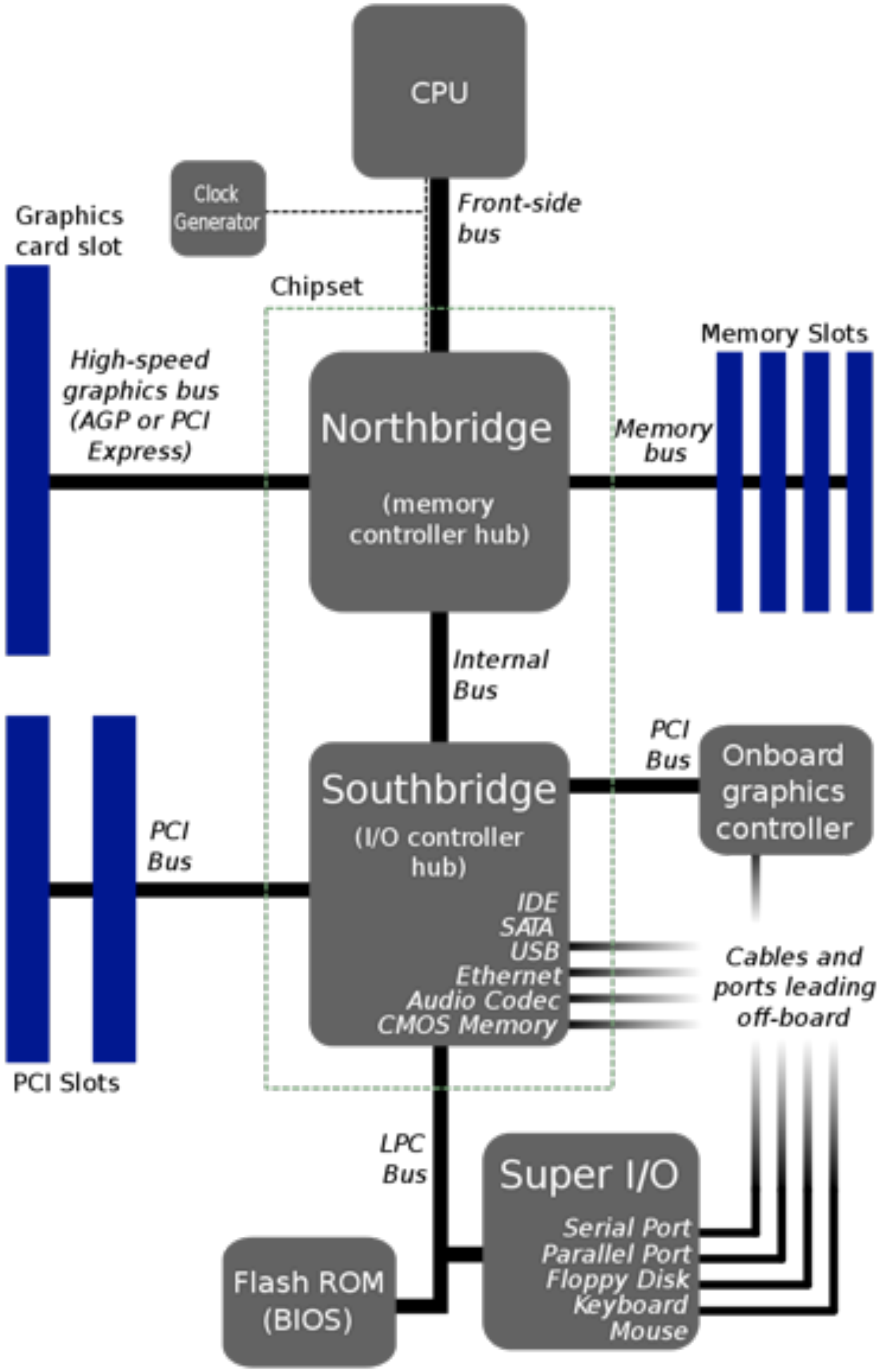
OS and Networking Review

15-712, Spring 2010

OS Definition

- Resource manager (bottom-up view):
  - Users access the common resources of the system (CPU, memory, files)
  - OS multiplexes access
- Extension of the machine (top-down view):
  - OS abstracts the physical machine
  - simpler operations for ease of use (e.g. file access)

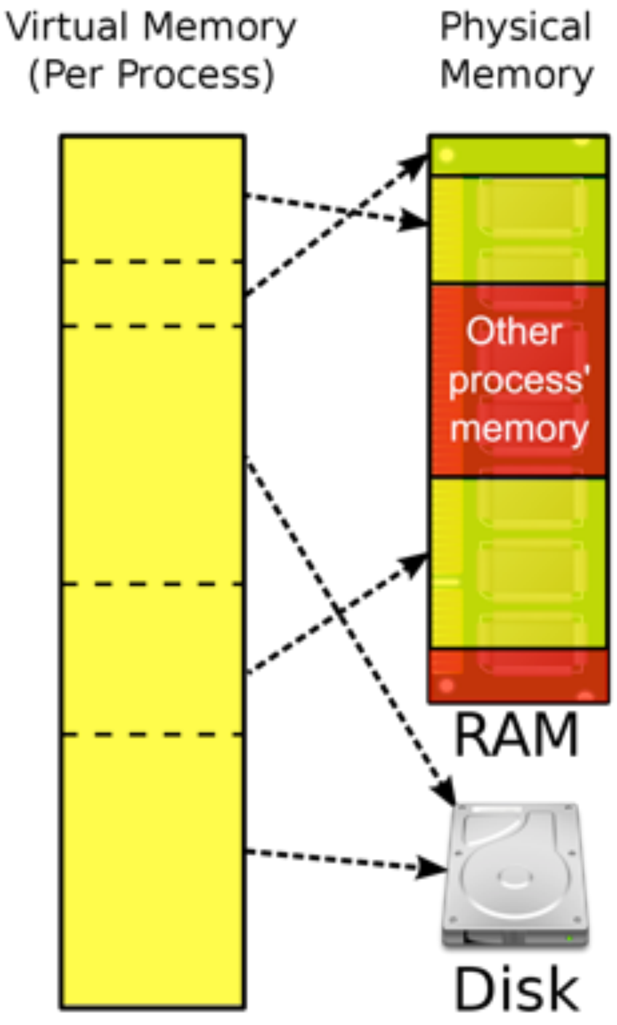
Hardware Resources



Source: Wikipedia

Virtual Memory

- Each process sees its own contiguous address space
- Virtual pages ↔ physical frames (usually 4KB)
- The mapping is stored in a page table (one table per process)
- MMU (Memory Management Unit) does the translation – caches table entries in the TLB
- Demand paging, page fault, swapping, thrashing
- Applications can access only part of the address space – the rest is reserved for the kernel
    - why not different address space for kernel ?



OS Kernel

- Lowest-level abstraction layer for hardware resources – has direct access to hardware
- Runs in privileged CPU mode (kernel mode):
  - All instructions available
  - Can access pages with supervisor flag set (kernel space)
- Accessible through **syscalls**
- Multiple types of kernels: monolithic, microkernel, exokernel
- Some parts are permanently memory resident
- Device driver: code that handles specific hardware resource
  - Common source of bugs
  - Part of the kernel in monolithic kernels (kernel modules)

Processes, Threads

- **Process** = The primary abstraction in an OS
- Is an instance of a program
- Has several resources: address space (pages and page table), memory, open files
- Can have one or multiple concurrent threads
    - **Thread** = sequence of instructions inside a process
    - Threads share the process resources
- Processes can communicate through inter-process communication: semaphores, message queues, shared memory

Concurrency

- Two tasks are **concurrent** if the order in which they execute is not predetermined
- Concurrency ≠ Parallelism
- Concurrency is a property of the program
- Synchronizing concurrent threads (or programs):
  - Semaphore: P (proberen = to test),V (verhogen = to increment)
  - Mutex: acquire, release
  - Condition Variables
- Deadlock, livelock, race condition

Parallel/Concurrent Programming Styles

- Threads. Event-driven programming.
- Event-driven programming: program flow is determined by external events
- Sometimes an alternative to threads
    - E.g. multiplexing multiple TCP connections on one thread (select, epoll)
    - Usually easier to program than threads
    - Can be combined – e.g. I/O Completion Ports

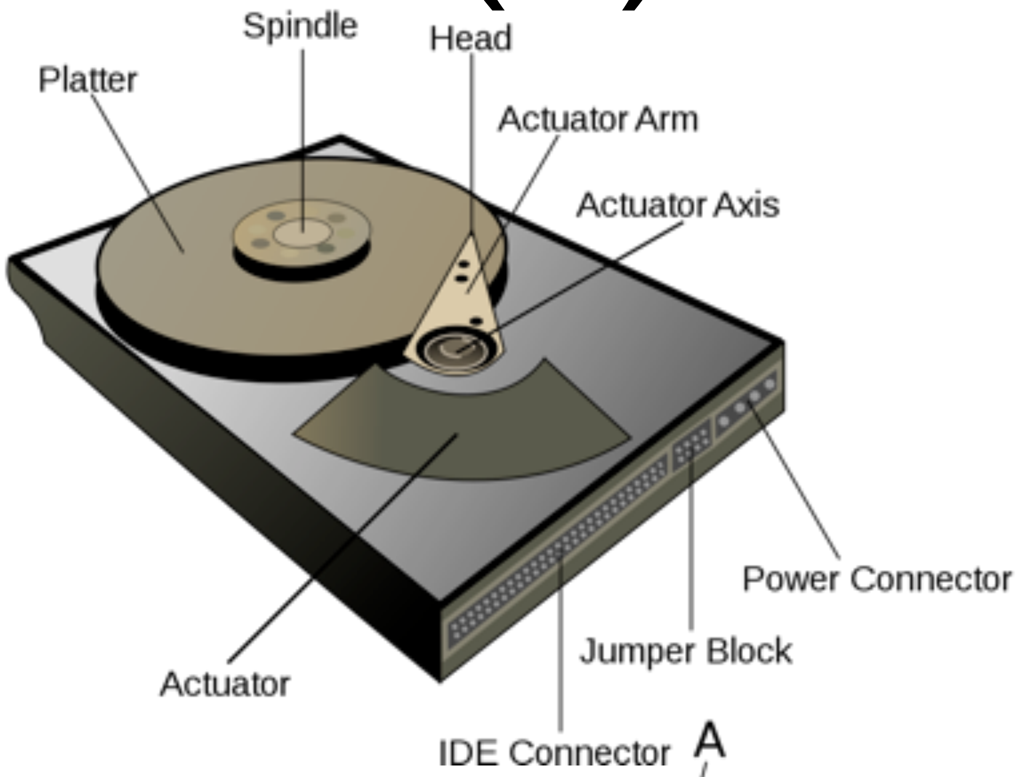
Security (goals)

- **Authentication**: establish that an entity is who it claims it is
- **Access control**: enables an authority to control access to resources in a systems
- **Confidentiality**: only authorized entities have access to information
- **Privacy**: the ability to control what information is accessible about oneself
- **Non-repudiation**: ensuring that a contract or statement cannot be repudiated
- **Integrity**: Ensure a message has not been tampered with

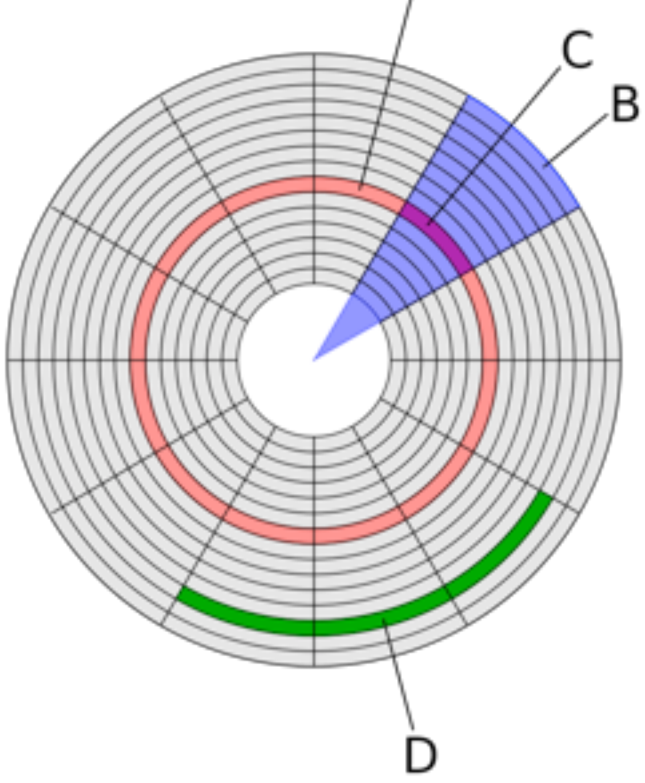
Security Primitives

- **Symmetric-key cryptography**: based on a shared secret (e.g. DES, AES)
- **Public key cryptography**: based on one-way functions (e.g. RSA, ElGamal)
- **Cryptographic hash function**: (e.g., SHA-1) easy to compute, but infeasible to:
  - revert
  - modify message without changing hash
  - find two messages with same hash
- **MAC**: Message Authentication Code
  - E.g. encrypt the hash of the message with the private key of a pair of asymmetric keys
- **One-time passwords**: passwords that change after each login

Hard Disk Drives (1)



- Data recorded by magnetizing ferromagnetic material
- Multiple platters, two read/write heads per platter (one on each side)
- Each platter is divided into tracks and sectors
  - A: track, B: geometrical sector, C: sector, D: cluster
  - Cylinder = the set of tracks of the same radius



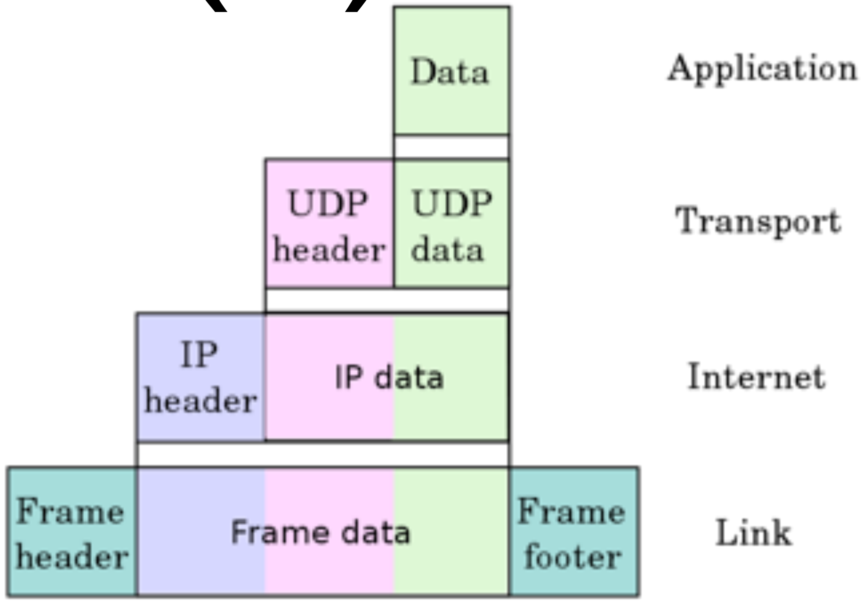
Hard Disk Drives (2)

- Random access slow because of:
  - **Seek time:** the time it takes to position the read head on the right track
  - **Rotational delay**: the time it takes for the addressed area to rotate into position
- Zone Bit Recording (ZBR): more sectors on outer tracks
  - Higher transfer rates on exterior tracks (almost 2X)
- Contiguous data transfer rate ~ 70 MB/s at 7200 rpm
- Seek time 2-15 ms

Transactions

- Units of work that have the ACID properties:
  - Atomic: all operations succeed or none succeeds
  - Consistency: The transaction leaves the system (e.g. database) in a consistent (legal) state
  - Isolation: While the transaction runs, its intermediate results are not visible from outside the transaction
  - Durability: If the transaction commits, its results are guaranteed to be persisted (e.g. even under system failure)
- In some contexts (e.g. speculative execution) some properties are relaxed

TCP/IP Basics (1)



- Multi-layered
  - Encapsulation
- Link layer (Ethernet): communication between hosts in the same network
- Internet layer (IP): communication over multiple (different) networks
  - Assume as little as possible about underlying network => doesn’t provide guarantees
  - Host identification – hierarchical addressing system
  - Packet routing (best effort)
  - Packets can be lost, delayed, reordered

TCP/IP Basics (2)

- Transport Layer: end-to-end transport
  - Application multiplexing (port numbers)
  - UDP: no delivery guarantees; optional checksum
  - TCP: connection-based, reliable, in-order byte stream
    - congestion control; tries to fill pipes without overfilling
- Typical application interface: **sockets**
  - Kernel objects
  - Applications access them through syscalls (connect, bind, listen, send, recv, sendto, recvmsg)

Network Delay

- **Transmission delay**: time it takes for sender to put the packet bits on the link
- **Propagation delay**: time for the signal to reach destination
- **Queueing delay**: amount of time packet spends in routing queues
- **Processing delay**: time routers spend processing the packet header

Networks - Misc.

- **Bandwidth-RTT product**
  - Ideal amount of data in pipeline – avoids stalls
- **Bandwidth-delay product**
  - The maximum amount of in-flight data in the network
  - Large value => Difficult for transport protocols (e.g. because of congestion avoidance, TCP may reach optimum rate slowly)

Numbers everyone should know

- L1 cache reference 0.5 ns
- Branch mispredict 5 ns
- L2 cache reference 7 ns
- Mutex lock/unlock 100 ns
- Main memory reference 100 ns
- Compress 1K bytes with Zippy 10,000 ns
- Send 2K bytes over 1 Gbps network 20,000 ns
- Read 1 MB sequentially from memory 250,000 ns
- Round trip within same datacenter 500,000 ns
- Disk seek 10,000,000 ns
- Read 1 MB sequentially from network 10,000,000 ns
- Read 1 MB sequentially from disk 30,000,000 ns
- Send packet CA->Netherlands->CA 150,000,000 ns