

Department of Computer Science

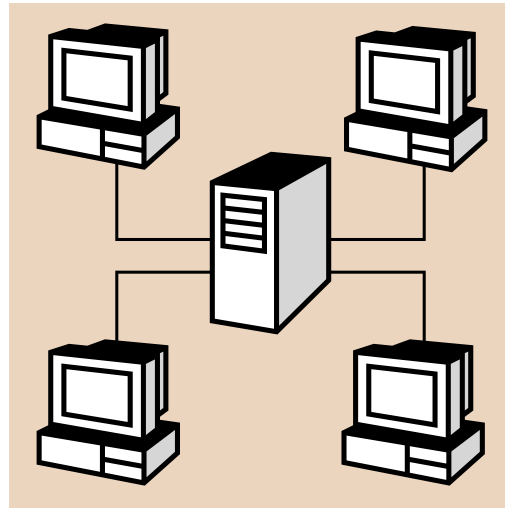
PURDUE
UNIVERSITY

Distributed Systems

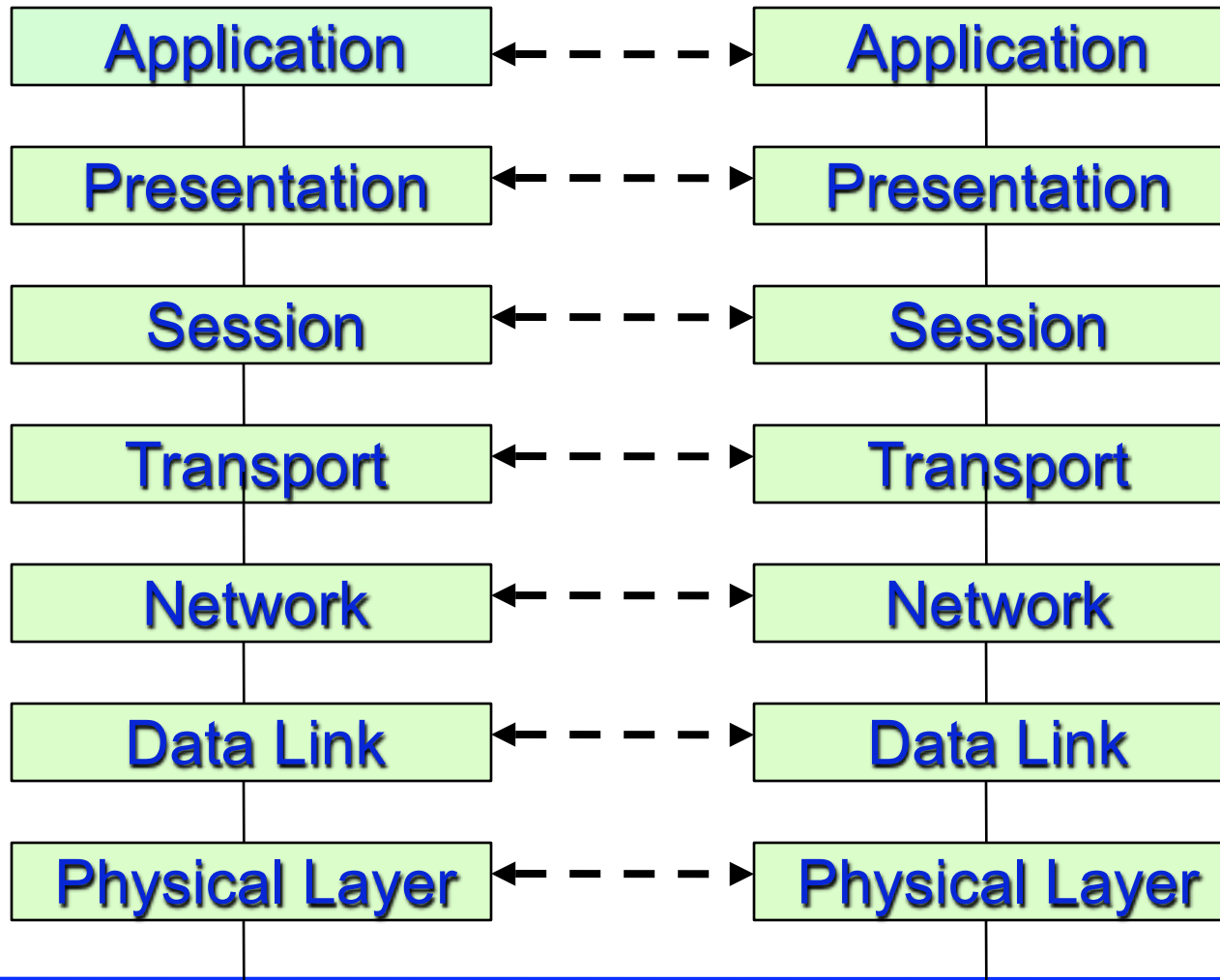
Lecture 3: Background

PURDUE
UNIVERSITY

Basic Communication Services



OSI/ISO Model



Internet Protocol -- IP (Network Layer)

- ▶ **IP is the current delivery protocol on the Internet, between hosts**
- ▶ **IP provides “best effort”, unreliable delivery of packets**
- ▶ **There are two versions**
 - IPv4 is the current routing protocol on the Internet
 - IPv6, a newer version, still not totally embraced by the community



Transport Protocols (Transport Layer)

- ▶ Provides communication between processes running on hosts
- ▶ The most common transport protocols are UDP and TCP
- ▶ OS provides support for developing applications on top of UDP and TCP



User Datagram Protocol -- UDP

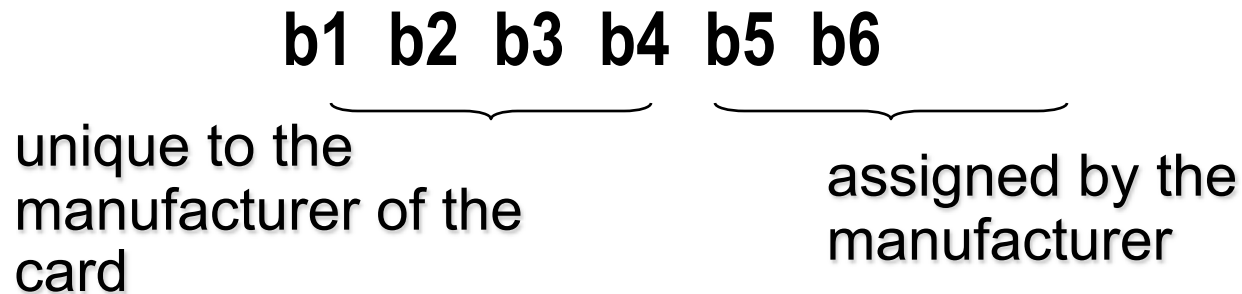
- ▶ **Connectionless protocol for a user process**
 - No connection established
 - Unreliable transmission
 - no guarantee that the packets reach their destination
 - Error detection
- ▶ **Runs on top of IP**
 - IP performs breakdown/wrapping of user space messages

Transmission Control Protocol -- TCP

- ▶ **Connection oriented protocol for a user process**
 - **Reliable, full-duplex channel**
 - Acknowledgements, retransmissions, timeouts, flow-control
 - **FIFO ordering**
 - Packets are delivered in the same order in which they were sent
 - **Flow control**
 - Acks, max allowed window size specified by *receiver*
 - **Congestion control (*sender* side)**
 - Congestion window for max number of packets in transit (w/o acks)
 - Multiplicative decrease (div. by 2) on ack timeout
 - Additive increase (by 1) if max packets acked successively
 - Slow-start phase; exponential increase (mult. by 2) until threshold is hit

Hardware Addresses

- ▶ Hosts access the physical medium via network cards
- ▶ Each network card is uniquely identified by a 48 bit (6 bytes) number, called hardware address, or Ethernet address.
- ▶ Ethernet addresses are hardwired into the electronics of the network device



- ▶ Address resolution protocols ARP/RARP protocols map IP addresses to hardware addresses and vice versa

IP Addresses

- ▶ **Hosts are identified in the network by IP addresses**
 - IPv4 addresses: 32 bits addresses, most used
 - IPv6 addresses: 128 bits addresses
- ▶ **Each decimal number represents eight bits of binary data (value between 0 and 255)**
- ▶ **Divided in classes**
 - Network addresses with first byte between 0 and 127 are class A
 - Network addresses with first byte between 128 and 191 are class B
 - Network addresses with first byte between 192 and 223 are class C
 - All other networks class D, for special functions or class E which is reserved
- ▶ **Class-less**
 - E.g. 192.168.0.0/16

Naming Services: DNS

- ▶ **People prefer names for hosts (hostnames)**
 - Name: `arthur`
 - Fully qualified name: `arthur.cs.purdue.edu`
- ▶ **DNS (Domain Name System) maps hostnames to IP addresses**
- ▶ **Example:**
 - `arthur.cs.purdue.edu` has the IP `128.10.2.1`

NATs and their Implications

- ▶ **There are not enough IP addresses**
- ▶ **Solutions: IPv6 orNetwork Address Translation (NAT)**
- ▶ **NAT allows a single device, to act as an agent between the Internet (or "public network") and a local (or "private") network: only a single, unique IP address is required to represent an entire group of computers**
- ▶ **Computers can not communicate directly, STUN client-server protocol allows computers to discover each other behind a NAT (learn their public addresses), but requires presence of STUN server**

Problems with NATs

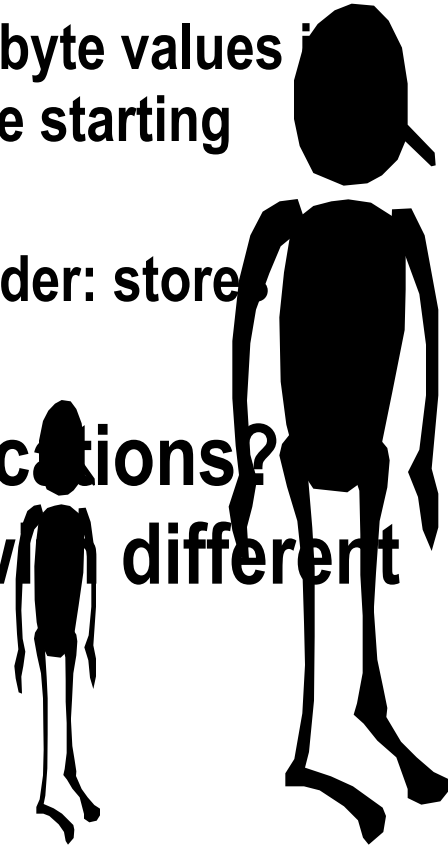
- ▶ **Break end-to-end control**
- ▶ **Hosts depend on same trusted point (the STUN server)**
- ▶ **Add complexity**
- ▶ **Prevent IP security deployment**

IP Multicast

- ▶ Provides support for group communication
- ▶ Groups specified by reserved IP multicast addresses 224 . 0 . 0 . 0 to 239 . 255 . 255 . 255
- ▶ Unreliable communication
- ▶ IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN
- ▶ Network cards recognize IP multicast addresses: hosts that did not subscribe to a particular group will not process those packets (unlike UDP broadcast that is processed by all hosts in a network segment)
- ▶ Issues with IP multicast: can be used to cause DOS, many ISP and enterprise network block IP multicast communication

Byte Order

- ▶ **Different systems store multibyte values (for example `int`) in different ways.**
 - HP, Motorola 68000, and SUN systems store multibyte values in Big Endian order: stores the high-order byte at the starting address
 - Intel 80x86 systems store them in Little Endian order: store low-order byte at the starting address
- ▶ **Why is this a problem for network applications?**
Data is interpreted differently on hosts with different architectures



Buffering and Fragmentation

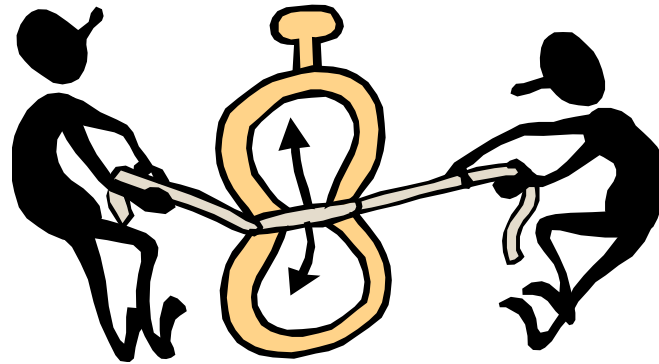
- ▶ **Buffering: OS maintains a set of buffers used to temporarily store incoming and outgoing messages**
- ▶ **THE BUFFERING SPACE is LIMITED**
- ▶ **Fragmentation: IP datagrams are fragmented, they can travel on different paths**
- ▶ **When processes send very quickly, packets can be dropped by the OS without any notification**
- ▶ **On sending: no OS memory can be obtained for one or several fragments**
- ▶ **On receiving: one or several fragments did not make it to the destination, entire datagram is dropped**

Enough?

- ▶ **Why do these protocols not provide better support for distributed applications?**

The End-to-End Argument

- ▶ End to end arguments in System Design. Saltzer, Reed, Clark TOCS 1990.



What Is It All About?

- ▶ Analyzes what services should be provided at low levels and what should be provided by the application
- ▶ Commonly cited as a justification for not addressing reliability at low levels and let application handle it
- ▶ Example: how to transfer a file: hop-by-hop or end-to-end
- ▶ Low-level mechanisms should focus on speed, not reliability
- ▶ The application should worry about “properties” it needs
- ▶ Reasoning about reliability is required at higher levels and can't be accomplished by basic reliable mechanisms

Analogy

	<i>Computation-oriented</i>	<i>Communication-oriented</i>
<i>Basic tools</i>	Programming Languages	Networking
<i>Composition</i>	Software Engineering	Distributed Systems

References



- ▶ Chapter 1 and 2 from *Reliable Distributed Systems*
- ▶ *Why do Internet Services Fail, and What Can be Done about It?* D. Oppenheimer, A. Ganapathi and D. A. Patterson, 2003.
- ▶ *Why Do Computers Stop and What Can be Done about It?* Jim Gray, 1985.
- ▶ *End to End Arguments in System Design.* Saltzer, Reed, Clark TOCS 1990.

Next Lecture

- ▶ ***Fundamentals of Fault-tolerant Distributed Computing in Asynchronous Environments.*** Felix C. Gartner, *ACM Computing Surveys* 31(1):1—26, 1999.