Consultation hours

... will be held on
- 3am Wednesday Nov. 10, 2004
- noon Friday Nov. 12 and 19, 2004

No clash with exams for: ELEC2032, ELEC2042, ELEC3014, ELEC3017, MATH2509 (Wed. Nov. 17th)
Starting at lecturer’s office (341) then moving to a tutorial room (probably 224 – sign on door of 341).

Procedure:
1. Solicitation for topics – specify lecture & slide
2. Voting on topics (Lecturer may prioritise further)
3. Topics in order of decreasing popularity
4. One-on-one consultation: ≤ 5 min ea round robin

Exam content

Style of questions:
- Cover sheet on the course web page
  - 25% multiple choice
  - 75% "short answer"
    - say 55% describing concepts;
    - 20% applying concepts

Rough estimates of coverage:
- 85% covered in lectures
  - 5% ea in tutorials, labs, text only
- 70% covered by review lectures
- 20% of exam covers material covered by the mid-session

Exam tips

Advertisement

- Tired of UNSW’s email system?
- Tired of other unreliable network services?
- Consider doing research on network dependability, e.g. for 4th year thesis.
  - 2004 theses included:
    - communication about service outages, and
    - pioneering measurements of email reliability
- See
  http://uluru.ee.unsw.edu.au/~tim/dependable/ for details

Course review

Warnings:
- Obviously can’t comprehensively repeat 13 weeks of material in one week.
  - Yell out if you want something covered in depth.
- From Lecture 1: "There is a difference in kind between eating an ice cream cone every day, and eating 365 ice cream cones on your birthday" [Tanenbaum, 2nd edition, p. 86]
- These review slides are from last year – if there are inconsistencies, this year’s lecture slides have precedence.
Review of weeks 1-5

Architecture
- Network structures & scope; client-server, peer-to-peer
- Protocols, layering & reference models
- Packet and (virtual) circuit switching; with/without connections

The analog domain
- Audio and video coding
- Transmission coding
- Modulation, transmission coding

Link layer
- Framing and error coding
- Link layer protocol principles
- Specific link protocols
- Link layer switching

Don’t worry about

What’s the Internet: “nuts and bolts” view

millions of connected computing devices: hosts, end-systems
- PCs workstations, servers
- PDAs phones, toasters
running network apps

communication links
- fiber, copper, radio, satellite
- transmission rate = "bandwidth"

routers/switches: forward packets (chunks of data)
protocols control sending, receiving of msgs:
- e.g., TCP, IP, HTTP, FTP, PPP

nodes:
- local ISP
- regional ISP
- company network

ad hoc & peer-to-peer systems blur the distinctions between hosts/routers & workstations/servers

Client-server paradigm

Typical network app has two pieces: client and server

Client:
- initiates contact with server ("speaks first")
- typically requests service from server,
- Web: client implemented in browser; e-mail: in mail reader

Server:
- provides requested service to client
- one server usually serves many clients
- e.g., Web server sends requested Web page, mail server delivers e-mail

Communication protocols

“agreement between the communicating parties on how communication is to proceed”

Communication protocols regulate:
- **syntax**: structure/format of the information
- **semantics**: the meaning of fields within this structure
- **behaviour**: timing of when information is exchanged
How layers handle data units (1)

Each layer takes data from above
• **Encapsulation**: adds header information to create new data unit
• passes new data unit to layer below

When information ascends the protocol stack, headers are removed.

Internet protocol stack

**application**: supporting network applications
  - FTP, SMTP, HTTP
**transport**: host-host data transfer
  - TCP, UDP
**network**: routing of datagrams from source to destination
  - IP, routing protocols
**link**: data transfer between neighboring network elements
  - PPP, Ethernet
**physical**: bits “on the wire”

Strictly, the Internet protocols aren’t concerned with layering below the network layer, so the link & physical layers are often treated as a merged layer.

Layering: physical communication

Note that different nodes in the network may have different varieties of each layer, e.g. PPP over modem on left & Ethernet link/phy on right. Routers may provide conversion, by having varied ports.

Functions of OSI layers

**Application**: Protocols that are commonly needed by end-users for specific applications.
**Presentation**: Formatting of payload
**Session**: Coordinate communicating entities, e.g. turn-taking.
**Transport**: Match path to end-users: e.g. enhance reliability
**Network**: Concatenate links together to create an end-to-end path. Routing functions to determine possible paths.
**Data Link**: Aggregate bits to form frames. Sharing of access to broadcast channel.
**Physical**: Transmit raw bits, e.g. physical transmission characteristics.
Outline

Connection management

Communication is often classified as being either:

**Connection-oriented:**
3 distinct phases:
1. Connection establishment
2. Data transfer
3. Connection release
   Classic example: Telephone network.

**Connectionless:** No connections; data is pushed into the network
   Classic example: Postal network.

Packet Switching: Statistical Multiplexing

Sequence of A & B packets does not have fixed pattern ➔ *statistical multiplexing.*
In TDM each host gets same slot in revolving TDM frame.

Note that the actual time depends on many factors: packet switching isn't necessarily the fastest.
Four sources of packet delay

1. Transmission delay
2. Propagation delay
3. Processing in each node
   - determine output link
   - check bit errors
4. Queueing
   - time waiting at output link for transmission
   - depends on congestion level of router

Circuit switching also has these

Forms of packet-switching

There are connection-oriented and connectionless forms of packet switching:
- datagram network: Connectionless
  - destination address in packet determines next hop
  - routes may change during session
  - analogy: driving, asking directions
- virtual circuit network: Connection-oriented
  - each packet carries tag (virtual circuit ID), tag determines next hop
  - fixed path determined at call setup time, remains fixed thru call
  - routers maintain per-call state

Outline

Data rate vs baud rate vs bandwidth

Units:
- Data rate (bits / second): Information carrying capacity
- Baud rate (symbols / second): How often the signal can vary
- Bandwidth (Hz): Range of frequencies that carry significant energy.

These measures are correlated, but are different:
- Each signal carries some bits ⇒ more signals = more bits.
  e.g. V32 modem: 2400 baud, each symbol carries 4b=9600b/s
- Shannon’s capacity formula relates data rate to bandwidth, but the relationship depends on the SNR.

The term "bandwidth" is often used to describe data rate, even though the technically correct term is "data rate".
Framing using length fields

First characters indicate how long frame is.
- Receiver can allocate buffer of appropriate size when first characters of frame arrive.
- Transmitter needs to know how long frame is before starting.

Sensitive to transmission errors:
- Error may change a length field, and so cause mis-interpretation of subsequent characters as length fields, continuing indefinitely.
- Rarely used at link layer, but may be used at higher layers (e.g. IP) after link layer error detection.

Stuffing

Problem with binary data: The frame may contain arbitrary strings of bits, including the flag string.

- e.g. FLAG H1 H2 A B C FLAG D E F FLAG

Don't want it to be interpreted as delineating 2 frames

Solution: Stuffing: When transmitting, stuff in extra characters where flag characters occur. Extra characters distinguish apparent flags in payload from genuine flags.

- e.g. stuff in an "Escape" character before apparent flag characters & before apparent escape characters.

Issues:
- Errors: Can convert payload into flag, or vice versa, but effect will only persist until the next flag.
- Extent to which stuffing expands the data depends on content
- Often scramble content to prevent long string of characters needing stuffing.