Protocol Architectures

Dr Steve Gordon
ICT, SIIT

Contents

- Protocol Architectures and Standards
- Open Systems for Interconnection (OSI)
- TCP/IP
- Traffic and Performance

Need For Protocol Architecture

- Example: File Transfer from one computer to another
  - Require path between two computers: either direct link or communication network
  - But more is needed to transfer the file:
    - Source must activate the link/network in preparation for transmission
    - Source must know that destination is ready to receive
    - Source file transfer application must know that destination application is prepared to accept
    - File formats may need to be translated
- High degree of cooperation is needed between two computer systems
- Data exchange is a complex task – it is very hard!
- So apply the divide-and-conquer principle:
  - Break the communication tasks into subtasks
  - Implement tasks separately in layers in stack
  - Each layer provides functions needed to perform communications for layers above
  - Each layer uses functions provided by layers below
  - Peer layers communicate with a protocol
  - Combine the layers together to get *Protocol Architecture*

Key Elements of a Protocol

- What is a protocol?
  - Set of rules (or conventions) that two (or more) peer entities obey in order to communicate
- Main elements of a protocol:
  - Syntax: the types of messages that can be exchanged, and the format of each message
  - Procedures (or rules): the set of rules that each entity must follow
    - E.g. what to do when receive a message of type X; what to do when a timer expires
    - Includes information and the meaning of messages and timing of events
  - Assumptions about operating environment

Protocol and Standards

- Protocols are rules that communicating entities follow
  - Protocols are implemented in hardware and software on computing devices
- Standards are agreed-upon rules, i.e. protocols that some organisation has agreed upon
  - Standards are essential in creating open and competitive market
    - If all equipment manufacturers follow one standard, then you, as a purchaser, can select the equipment that best suites your need and know that it will interoperate with other equipment
  - Guarantee national and international interoperability
  - De jure standards: standards that have been officially recognised or are part of law
  - De facto standards: not approved by standards organisation, but in widespread use

Standards Organisations

- *International Organisation for Standardisation (ISO)* – formed from national standards bodies to create global standards
- *International Telecommunication Union – Telecommunication Sector (ITU-T)* – formed from national telecom operators and other organisations to create global standards for telecoms
- *Institute of Electrical and Electronics Engineers (IEEE)* – professional engineering society that develops standards in electronics, radio and electrical engineering
- *American National Standards Institute (ANSI)* – US standards organisations
- *Electronic Industries Association (EIA)* – electronics manufacturing standards
- *Internet Engineering Task Force (IETF)* – part of the Internet Society, develops most standards for the Internet
- *World Wide Web Consortium (W3C)* – develops web based standards (e.g. HTML)
- Others:
  - Forums and Special Interest Groups: usually companies get together to work on specific technologies
  - Regulatory agencies: government agencies that set regulations on use of communication technologies

TCP/IP Protocol Architecture

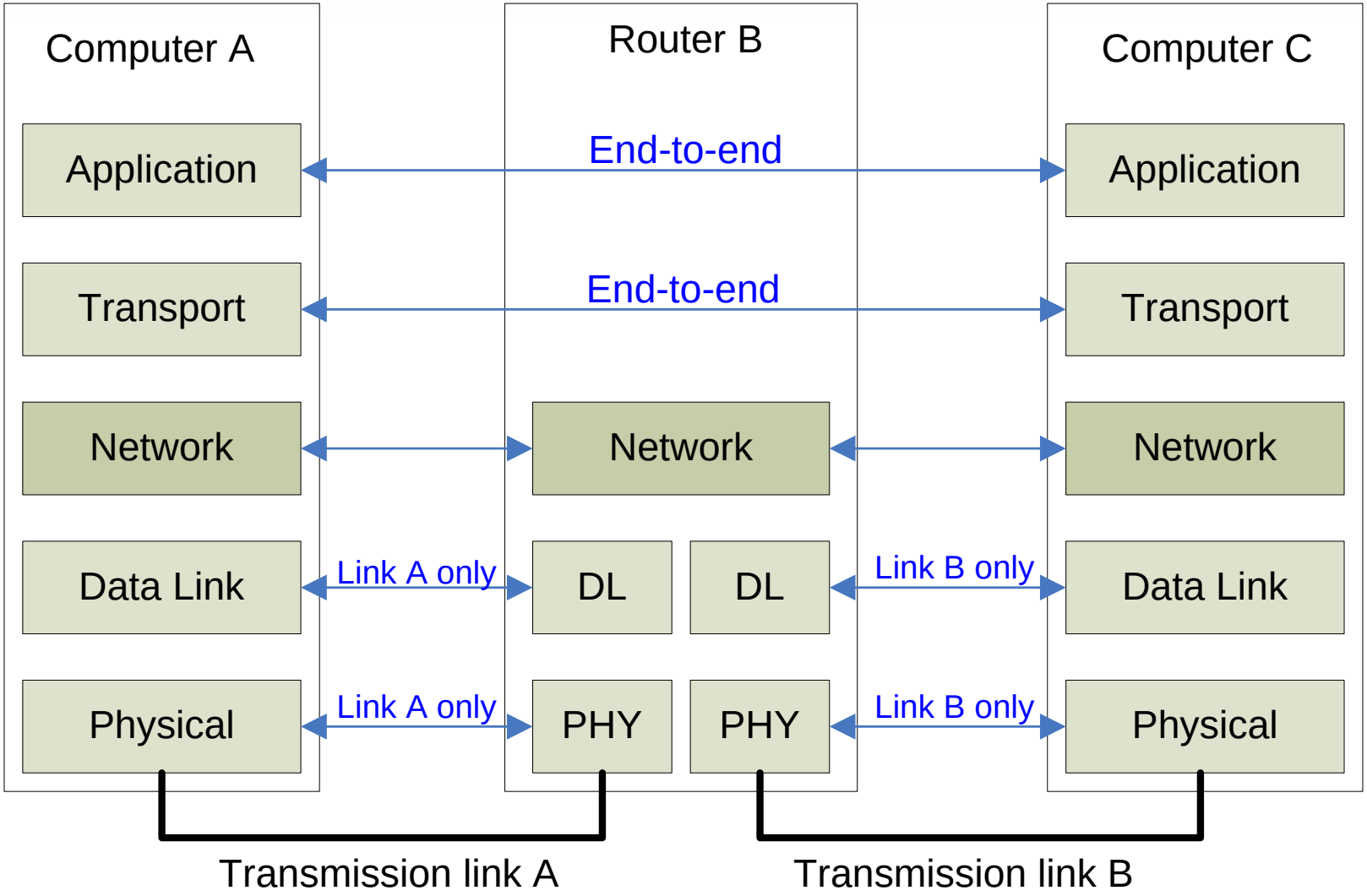
Also called “Internet stack”

TCP/IP Protocol Architecture

- Developed by US Defense Advanced Research Project Agency (DARPA)
  - For ARPANET packet switched network
- Used by the global Internet today
- Protocol suite comprises a large collection of standardized protocols
- There is no official layered model (unlike 7-layer OSI)
  - But many people (textbook authors, lecturers) have tried to characterize Internet protocols into a layered model
  - Usually 5 layers (sometimes the names and functionality differ)
  - We will use this layered architecture in remainder of course

- Note: TCP is a protocol; IP is a protocol; but TCP/IP most often refers to a set (or suite) of protocols used on the Internet
  - E.g. TCP, IP, UDP, ICMP, IGMP, ARP, …
  - TCP/IP does not mean “only TCP and IP”
  - TCP/IP architecture may also be called “Internet Architecture” or “Internet Stack”

TCP/IP Layering Concepts



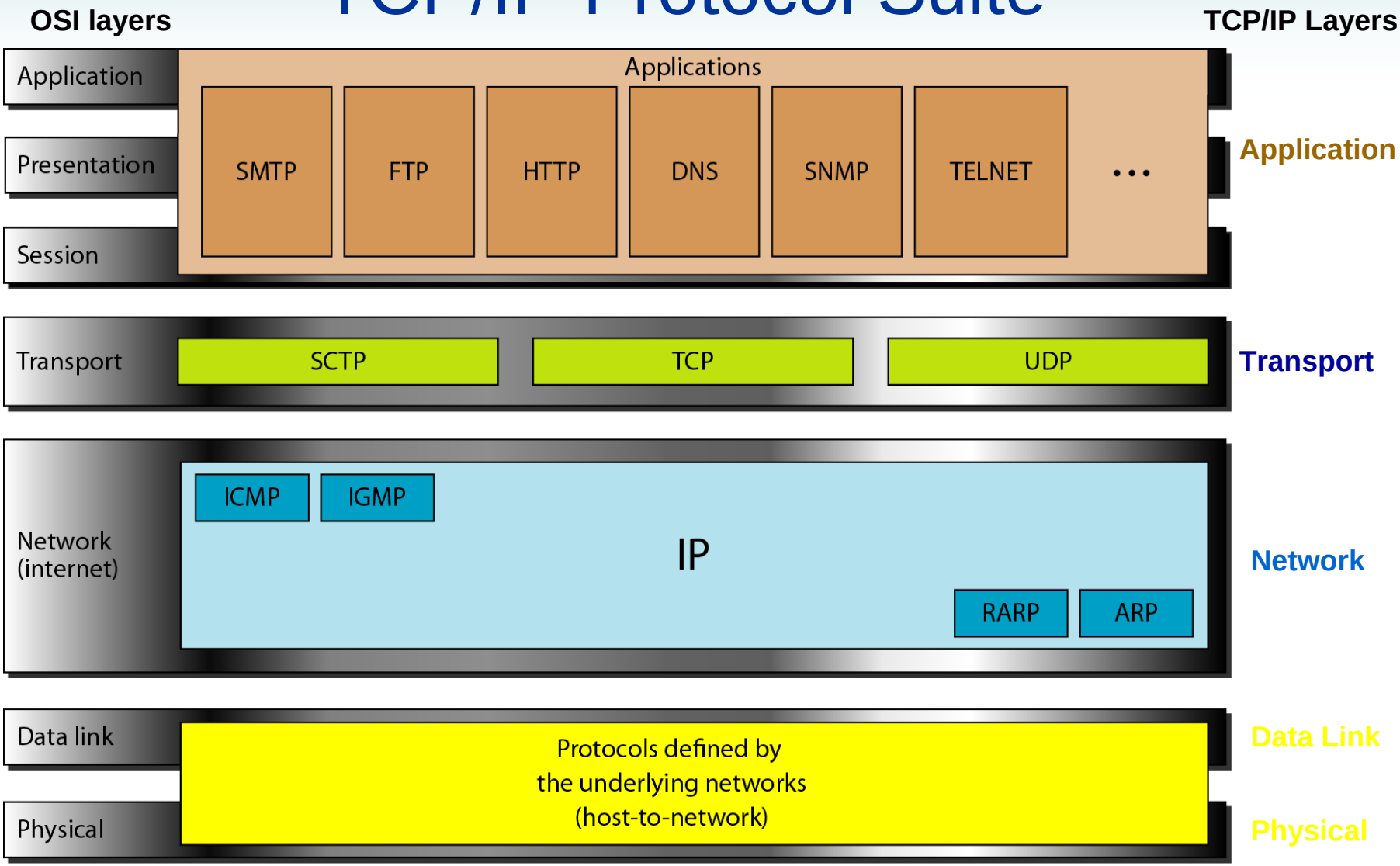
TCP/IP Layered Model

- 5 Layered Model (from bottom)
  1. Physical Layer
     - Physical interface between transmission device and medium
     - How to send bits over transmission medium: data rate, signalling, electrical signals, codecs, modems, …
  2. Data Link Layer
     - Sometimes called: “Network Access”, “MAC”, “Link”, “Hardware”
     - Transmission of data over link or network to which the device is attached
     - Addressing scheme of destination device
     - Allows layers above to ignore details of links
  3. Network Layer
     - Sometimes called: “Internet” or “IP” layer
     - Core of the Internet; uses the Internet Protocol
     - Allows hosts to communicate across different networks
     - Provides routing across the Internet; determine the path to take
     - Provides unreliable, connectionless deliver of packets

TCP/IP Layered Model

- 5 Layered Model (from bottom)
  4. Transport Layer
     - Provides connections between applications (processes) running on the end hosts. Three standardised protocols:
       - User Datagram Protocol (UDP): unreliable, connectionless sending of packets
       - Transmission Control Protocol (TCP): reliable, connection-oriented sending of packets
       - Stream Control Transmission Protocol (SCTP): combines features of TCP and UDP to better support voice and other applications
  5. Application Layer
     - Everything else!
     - Contains functionality needed for various applications used on the Internet
       - E.g. for web browsing (HTTP), file transfer (FTP), email (SMTP), …

TCP/IP Protocol Suite



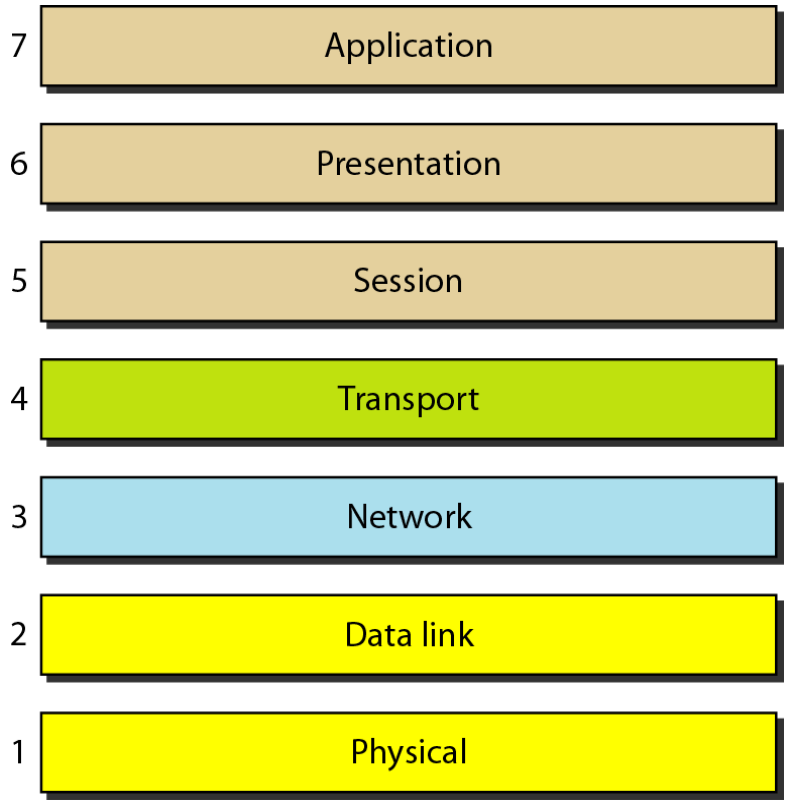
Other Protocol Architectures

OSI

Open Systems for Interconnection

- Open Systems for Interconnection (OSI)
  - Developed by the International Organization for Standardization (ISO), introduced in late 1970's
- The OSI 7-layer reference model
  - Defines concepts that are helpful in thinking about layering, architectures and describing protocols
  - Within each layer, one or more protocols are standardized
- Not used in practice today!
  - Implementations of TCP/IP were mature before OSI implementations were available
  - Overly complex compared to TCP/IP

OSI 7-Layer Model



- Application: allows users (human or software) to access network; provides user interfaces and support for applications
- Presentation: translation, encryption and compression of data formats
- Session: creates and manages connections (sessions) between applications
- Transport: reliable transfer of data between end-points (processes)
- Network: delivery of data across networks; establish connections between end-points
- Data Link: reliable transfer across a link, including addressing and error control
- Physical: mechanical, electrical and functional means of transferring bits over medium

Other Protocol Architectures

- Old protocol architectures
  - IBM SNA
  - Appletalk
  - Novel IPX
- Domain specific protocol architectures:
  - Signalling System 7 (SS7) for telephone signalling
  - UMTS for 3G mobile telecommunications

Addressing in TCP/IP Protocol Architecture

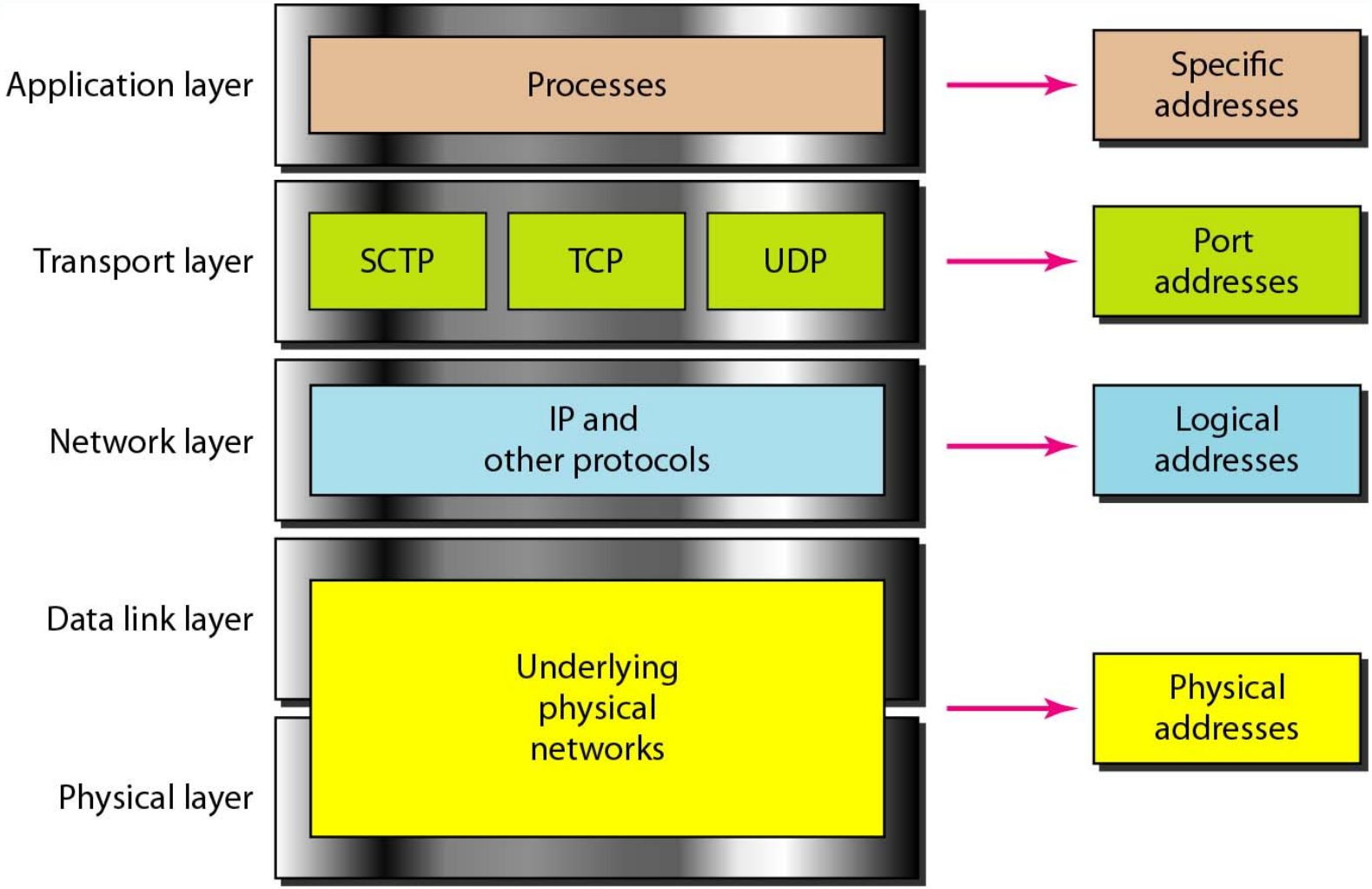
Addressing in TCP/IP

- Physical Addresses
  - Also referred to as “Data Link”, “Link”, “MAC”, “Hardware” addresses
  - Address of a physical interface on a device
  - Address type depends on the LAN/WAN technology being used
    - E.g. IEEE 48-bit addresses are used in Ethernet LANs; some Apple protocols use 8-bit dynamic addresses
  - Example: 07:01:02:01:2C:4B (48-bit IEEE address in hexadecimal)
- Logical Addresses
  - Also referred to as “Network” address
  - IP addresses are the format used in TCP/IP
    - Currently IP addresses are 32-bit addresses
    - In theory, all interfaces on a computer on the Internet may have a unique IP address
    - For example, your wired Ethernet interface may have an IP address, while your wireless WiFi interface may have another IP address
  - Example: 125.25.71.189 (32-bit IP address in dotted decimal notation)

Addressing in TCP/IP

- Port Addresses
  - Also referred to as a “Transport” address
  - IP address identify an interface on a computer
  - Port addresses identify software processes on that computer
  - Allows multiple Internet applications to run on the one computer at the same time
  - Example: 80 (port number used by web servers); 41067 (random port number used by a client application)
- Application-specific Addresses
  - Applications may use specific addresses
    - URLs, Email, P2P application addresses, …
  - Example: http://www.google.co.th/; steve@siit.tu.ac.th

Layers and Addresses in TCP/IP



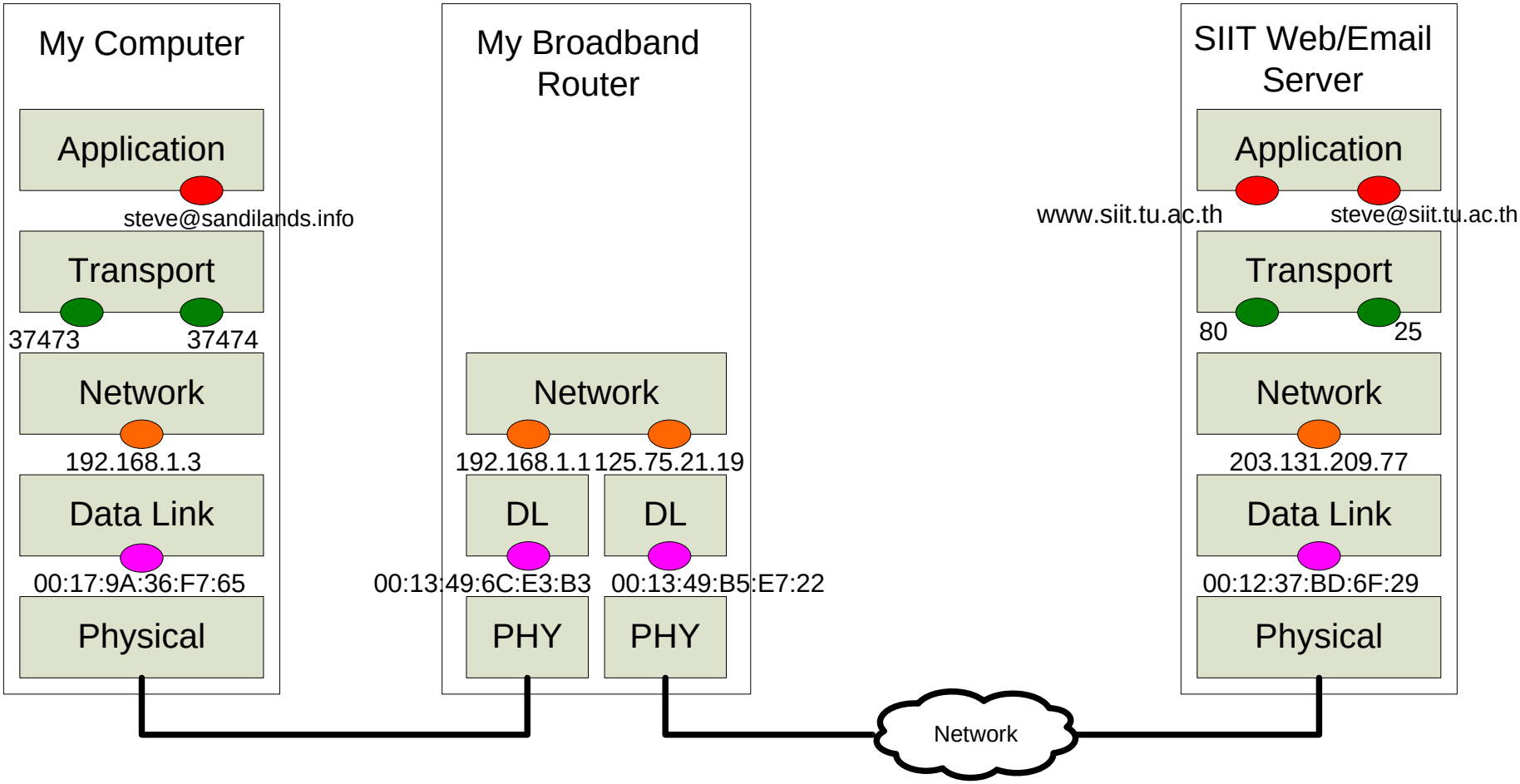
Addressing Example

Networking configuration from my home computer

```
sgordon@basil:~/its323$ ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:17:9A:36:F7:65
          inet addr:192.168.1.3  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::217:9aff:fe36:f765/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:105041 errors:0 dropped:0 overruns:0 frame:0
          TX packets:128616 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:12129260 (11.5 MiB)  TX bytes:56865017 (54.2 MiB)
          Interrupt:12 Base address:0xcf00

sgordon@basil:~/its323$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.1              ether   00:13:49:6C:E3:B3   C                     eth1

sgordon@basil:~/its323$ nslookup www.siit.tu.ac.th
Server:         192.168.1.1
Address:        192.168.1.1#53
Non-authoritative answer:
Name:   www.siit.tu.ac.th
Address: 203.131.209.77

sgordon@basil:~/its323$ netstat -t -n
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        1      1 192.168.1.3:37473       203.131.209.77:80       CLOSING
tcp        1      1 192.168.1.3:37474       203.131.209.77:80       CLOSING
tcp6       0    368 ::ffff:192.168.1.3:22   ::ffff:61.19.242.1:2109 ESTABLISHED
```


Addressing Example

Assume web browser and email client on My Computer communicating with web and email server at SIIT



Layers in Practice

Data Transfer

Implementing Layers

Data and Layers

=

message

segments

packets/datagrams

frames

bits 0110001010110101000111 10

Application

Application protocol adds a header and sends data as message (may break into multiple messages)

Transport

Transport protocol breaks into segments, adding header to each, and sends each segment

Network

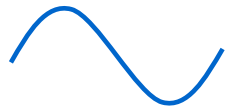
Network protocol adds header to each segment and sends datagram

Data Link

Data link protocol adds header and trailer to each datagram and sends frame

Physical

Physical protocol adds header and sends bits over physical medium (e.g. as waveform)



Implementing Layers

User Processes
- implement application protocol in each application (along with GUI and other functionality)

Application layer

Operating System
- implements TCP/IP set of protocols in software

Router Operating System
- Implements IP in software and hardware

Transport layer

Network layer

Network Interface Card (NIC)
- implemented in hardware and software (driver in OS)
- separate devices or individual chips

Data Link layer

Physical layer

Traffic and Performance

Traffic in Data Communications

- Traffic refers to the data/information moving through a communications system
- What types of traffic (or data)?
  - Data: Numbers, text, images
  - Audio: Telephone calls, radio stations, …
  - Video: TV, video conferencing, movies, …

- Data communication systems have progressed from initialling supporting data, through to audio and then video
  - Because the amount of information in video is usually much more than for audio, and audio much more than data
- Different types of traffic have different performance requirements
  - Remember: delivery, accuracy and timelines make an effective communications system

Performance of Networks

- Bandwidth
  - Bandwidth in Hertz (Hz): range of frequencies a channel can pass (next lecture)
    - E.g. the bandwidth of a telephone line is 4kHz
  - Bandwidth in bits per second (bps): number of bits a channel (or network) can transmit
    - E.g. the bandwidth of Fast Ethernet is 100Mb/s
    - Often referred to as “capacity” (how much is theoretically possible)
  - Relationship between the two depends on transmission system and modem (covered in next lectures)
- Throughput
  - How fast we can actually send data
  - Bandwidth is capacity of link/network; throughput is real data rate we achieve
    - Bandwidth and throughput are different because there are often overheads and other limiting factors on throughput
    - E.g. Fast Ethernet throughput may be 40Mb/s

Performance of Networks

- Delay (or Latency)
  - How long it takes for entire message to arrive at destination (from when first bit is sent)
    - Propagation time + Transmission time + Queuing time + Processing time
  - Propagation time = Distance / Speed
    - Speed of light (3×10^8m/s) is the best; air is slower, and cable is much slower
    - E.g. 12,000km across Atlantic ocean at 2.4×10^8 gives 50ms
  - Transmission time = Message Size / Bandwidth
    - E.g. 2.5KB email over 1Gb/s channel: 0.020ms
  - Queuing time: intermediate devices hold messages in queues in a network. Not a fixed factor
  - Processing time: end computers and intermediate devices process each message in CPU
    - Usually very small compared to propagation/transmission time (so we often ignore it)
- Jitter (or Delay Variance)
  - The difference in delay between subsequent packets

Performance Requirements of Traffic

- Data traffic:
  - In most cases, accuracy must be 100%
  - Throughput is also very important, as is delay
  - Examples:
    - Web browsing: a user wants the requested page to be returned in a short time. Response time in order of seconds
    - Email: a user wants the email to arrive; it doesn't matter if it takes 10ms or 10 seconds (or even 10 minutes)
    - File transfer: a user wants the complete file transferred as fast as possible. Usually, it doesn't matter if there is a small delay (seconds) to start the transfer, although expect the transfer time will be proportional to size

Performance Requirements of Traffic

- Audio/Video traffic:
  - Accuracy is not always important. If part of the audio or video does not arrive at destination, then usually the application can compensate
    - E.g. 100ms of audio is lost; or a frame of video is not displayed – in both cases the user may not notice
  - Delay is important (especially for interactive applications). If you are talking on the phone and there is a 2 second delay from when you speak to when the other person hears, you cannot converse!
  - Jitter is important. The delay should be constant.
    - E.g. a video source may generate 30 frames per second. Buffering can be used to compensate for some delay, but if the delay changes, then the video may play normal, then freeze, then normal, then freeze, …
  - Throughput is important. Audio and especially video require a large amount of information to be transferred, hence require large bandwidth and throughput.