Application Layer – Overview

Kameswari Chebrolu

Applications

- A huge variety of applications: Web, email, chat, audio/video streaming, file-transfer etc

- Role of application developers:

  – Understand the building blocks and their interaction

  – Make the right choices and implement required functionality

History of Internet Services

- 1970's and 1980's: Text based applications

  [97|]
  – Email, Telnet, File-transfer, Newsgroups, Chat

- 1990's: World-wide-web, Search, E-commerce (Amazon, Ebay, Paypal, Craigslist)

- 2000's: P2P file transfer (Napster, Bit-torrent), Social-Networking (Facebook,Twitter), Real-time applications (Skype, Youtube)

History of Internet Services

- Past few years: Cloud based services

  - Storage services: Dropbox, Apple's iCloud, Amazon's S3 (simple storage service)

  - Virtual Servers: Amazon's EC2 (elastic compute cloud)

  - Data processing: Amazon's Elastic Map-reduce

  - Database services: Amazon's RDS (relational database service)

Application Architecture

- Three types

- Client-Server

  - E.g. Web, Email, FTP

- Peer-to-Peer

  - Bit-torrent, Gnutella

- Hybrid of client-server and peer-to-peer
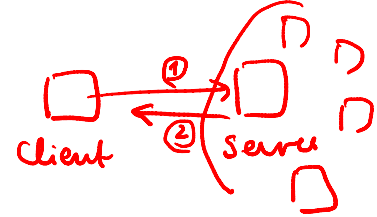
  - Skype, Instant messenger

Client-Server Architecture

- Client:

  - Initiates connection to server;

  - Intermittently connected; dynamic IP address

- Server:

  - Provides specific services (e.g. google server provides search functionality); may employ server farm

  - Always on; fixed IP address

Pure Peer-to-Peer Architecture

↳ *music file, video games etc*

- No server; end-systems coordinate to provide required service

- End-systems maybe intermittently connected; can have dynamic IP addresses

- Scalable system, but tends to be complex

Hybrid Architecture

- Initial contact is to a central server
  - Used to determine info about other end-systems
- After this, end-systems talk directly

IM → Instant Messaging

register IP add - Server

buddy

chat

user

IP address
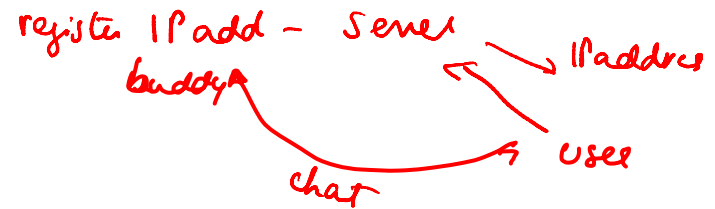
Application Protocols

Application	Protocol	Transport
E-mail	SMTP (RFC 2821)	TCP
Remote terminal access	Telnet (RFC 854)	TCP
Web	HTTP (RFC 2616)	TCP
File Transfer	FTP (RFC 959)	TCP
Streaming Multimedia	Proprietary	TCP or UDP
Internet Telephony	Proprietary	Often UDP

Application Protocols

- Define types of messages exchanged,

  – e.g., request, response

- Message syntax:

  – Fields in messages

- Message semantics

  – meaning of information in fields

- Rules for when to send and how to act on messages

Application Protocols

- Many protocols also have a companion protocol that specifies format of the data exchanged

  web ser
  - HTML is companion protocol of HTTP
 - RFC 822 and MIME define format of email messages; companion of SMTP → Email

Application Process/program

User Space

DPP

Process

Socket

Kernel Space

Transport
Layer

- Application programs use the application protocol to achieve intended task

  – E.g. Internet Explore, Chrome, Firefox all use HTTP to enable web access

- Processes identified by IP address:Port

  – Popular services have well know port numbers (e.g. 80 for web; 25 for email server)

| End-Host-1 | Intermediate-Host-1 | Intermediate-Host-2 | End-Host-2 |
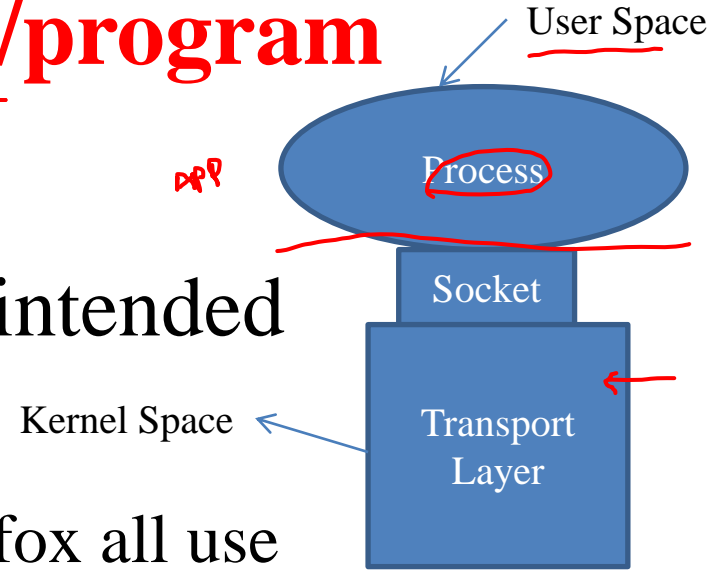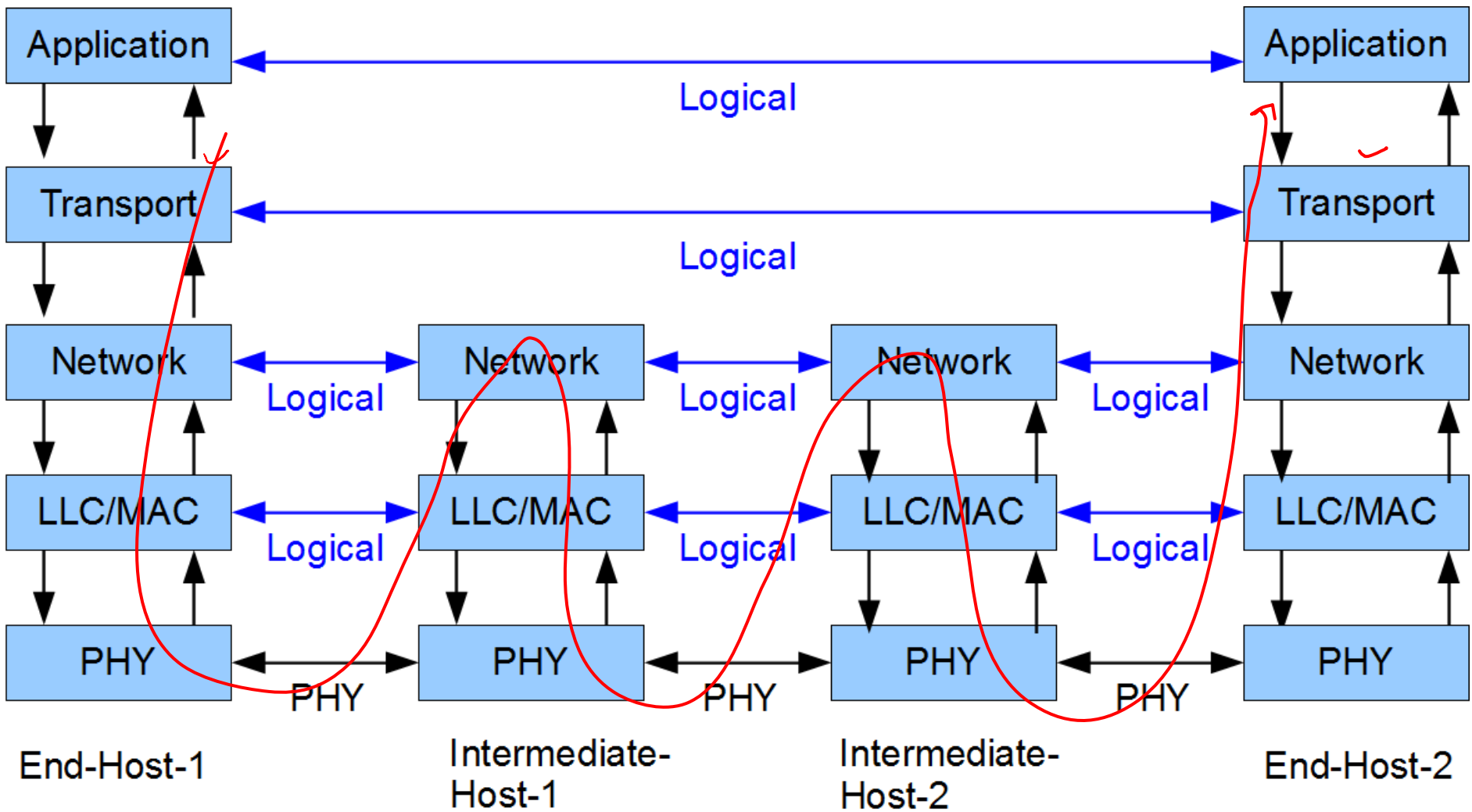
Summary

- Over the past 4 decades, many new and interesting networking applications have emerged

- Three predominant architecture types are used

  – Client-server, peer-to-peer, hybrid

- There are many application protocols; supporting protocols and implementations

- Ahead: Look at some popular applications