CSE 3231
Computer Networks

Introduction & Syllabus

William Allen, PhD

Spring 2022

Syllabus

- William Allen, PhD
  - Office: Harris 309
  - Office Hours information will be posted in Canvas
  - Email: wallen@fit.edu
    - this is the best way to contact me

Syllabus

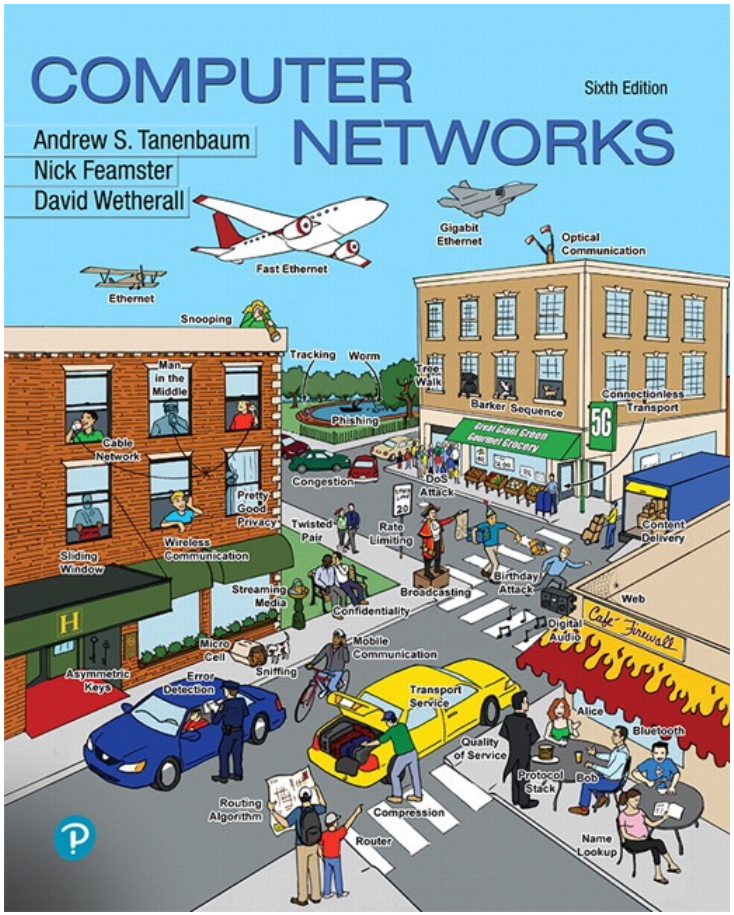
- Course Description:

  Provides an overview of computer networks, applications and protocols. Includes network architectures, routing, addressing, medium access control, connection-oriented and connectionless services, network performance analysis, network programming and common transport-layer, network-layer and link-layer protocols.

  Prerequisite: CSE 2010

Textbook

- Textbook:
  - **Computer Networks**, Andrew S. Tanenbaum and David J. Wetherall, Sixth Edition, 2021
    - A list of textbook sections that are related to the class lectures will be provided in Canvas



Syllabus

Your final course grade will be based on the following components:

- Quizzes, homework assignments and attendance in class: 55% total
- Midterm Exam: 20% of the final grade
  - Exam Date: *Tuesday, March 1*
- Final Exam: 25% of the final grade
  - Exam Date/Time: *Monday, May 2, 3:30–5:30 p.m*

To Get a Good Grade:

- Attend or watch the class lectures
  - the slides will not show all of the details that I will explain in class
- Read the assigned textbook sections
  - they will contain examples and more details than what is in the slides
- Read the assigned extra readings
  - these *may* be on exams!

Florida Tech Safe: *Return to Learn*

- In accordance with “Florida Tech Safe: Return to Learn” procedures, instructors will enforce Florida Tech’s *mandatory face covering policy* in all classrooms and teaching labs.
- All employees and students MUST wear appropriate face coverings that cover their mouth and nose when they are indoors.
- Students who fail to comply with this policy WILL BE REQUIRED to leave the classroom immediately.
- Students unable to comply may contact the Dean of Students, *Rodney Bowers*, for further options.

Academic Honesty:

All exams, quizzes and assignments for this class must be your own work. You may discuss assignments with other students and with your instructor, but *what you turn in is to be completely your own work*. Duplicate assignments, beyond reasonable coincidence, will be considered as cheating and dealt with as allowed by University policy. Department and University policies allow severe penalties (see the FIT Student Handbook) up to and including an F in the course and/or expulsion from the University, for plagiarism, i.e., submitting work as your own when it was, in fact, complete by others.

Plagiarism

- Copying/pasting from other people's written work into your own assignments
- Not citing the source of a quote or description that you insert into your work
- Make sure that you understand what plagiarism is so you can avoid it
  - *First offense: zero on the assignment*
  - *Second offense: F in the course*

Check Canvas Often

- Slides, examples, reading assignments and homework assignments will all be posted in Canvas
  - due dates for assignments will be in Canvas
  - and you will submit assignments to Canvas
- Exams and quizzes will also be in Canvas
  - questions can be multiple choice or essay
  - you will have some flexibility in when you start a quiz, but a limited time to complete it

What does networking give us?

- Without networking, computers would be isolated machines and only users with physical access would be able to process data that had been stored locally → 1950's and 1960's
- With networking:
  - computers can share data
  - users can access remote resources
  - users can communicate using computers
  - users can distribute tasks across many computers

Protocols

- Rules or Procedures for conducting computer communication
- Created by government or industry to standardize communications:
  - ISO - International Organization for Standardization
  - IETF - Internet Engineering Task Force
  - IEEE - Institute of Electrical and Electronics Engineers
  - ITU - International Telecommunication Union
  - W3C - World Wide Web Consortium

History of Communications

- Human Courier with oral/written message
  - Benefits?
    - can be delivered to the correct person
    - courier can provide verification of delivery
  - Limitations?
    - can be very slow if distance is great
    - courier could fail to deliver (death, injury, escape)
    - message could be intercepted, modified, replaced
    - how does the receiver know it was valid?

History of Communications

- Signal Flags, Drums, Smoke Signals etc.
  - Benefits?
    - faster delivery over longer distances
  - Limitations?
    - only works for “line of sight” communications, or requires relaying through other senders/receivers
    - sender and receiver must know the “code”
    - can be seen by anybody that is within range
    - the message can be modified without detection
    - won’t work at night or in bad weather

History of Communications

- Carrier Pigeons
  - Benefits?
    - not limited to “line of sight”
    - harder to intercept message (not impossible)
  - Limitations?
    - limited bandwidth (they can only carry so much)
    - how to verify delivery?
    - must return pigeons to sender
  - Internet Protocol Specification
    - https://www.ietf.org/rfc/rfc1149.txt

Network Working Group D. Waitzman, BBN STC

Request for Comments: 1149 **1 April 1990**

A Standard for the Transmission of IP Datagrams on Avian Carriers

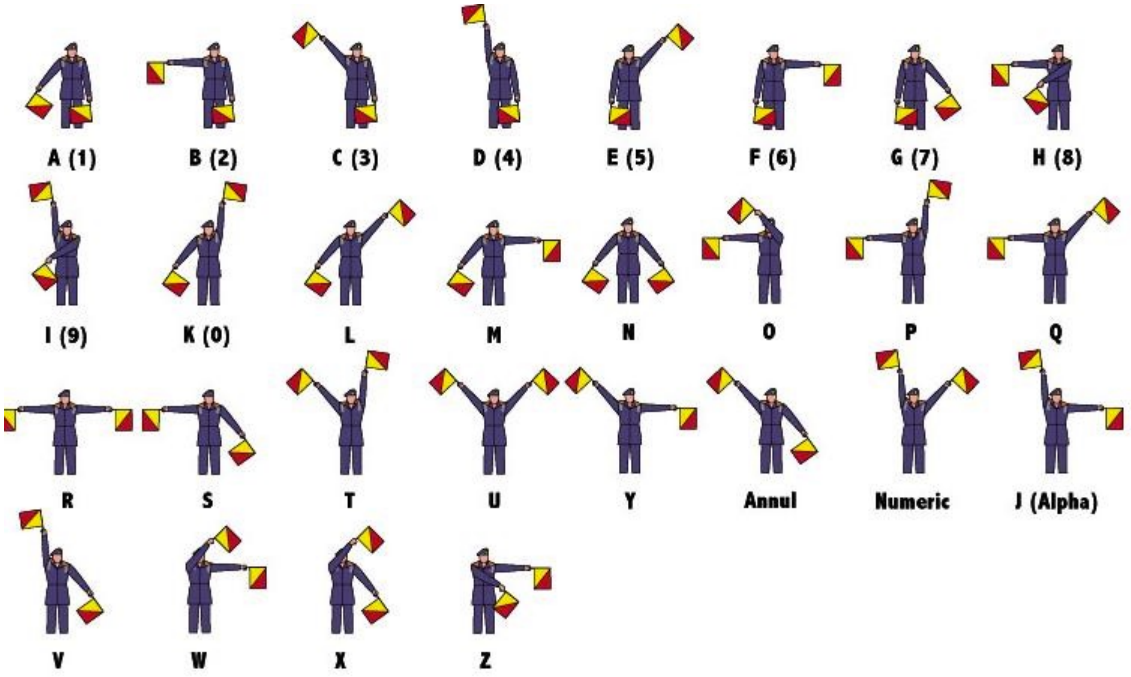
Avian carriers can provide high delay, low throughput, and low altitude service. The connection topology is limited to a single point-to-point path for each carrier, used with standard carriers, but many carriers can be used without significant interference with each other, outside of early spring. This is because of the 3D ether space available to the carriers, in contrast to the 1D ether used by IEEE802.3. The carriers have an intrinsic collision avoidance system, which increases availability. Unlike some network technologies, such as packet radio, communication is not limited to line-of-sight distance. Connection oriented service is available in some cities, usually based upon a central hub topology.

History of Communications

- Surface (Postal) Mail
  - Benefits?
    - allows individual addressing of messages
      - but, needs a standardized addressing scheme
    - delivery handled by an organized infrastructure
      - can provide verification of delivery
  - Limitations?
    - requires significant resources to operate
    - sender must know address of receiver
    - less efficient for shorter distances

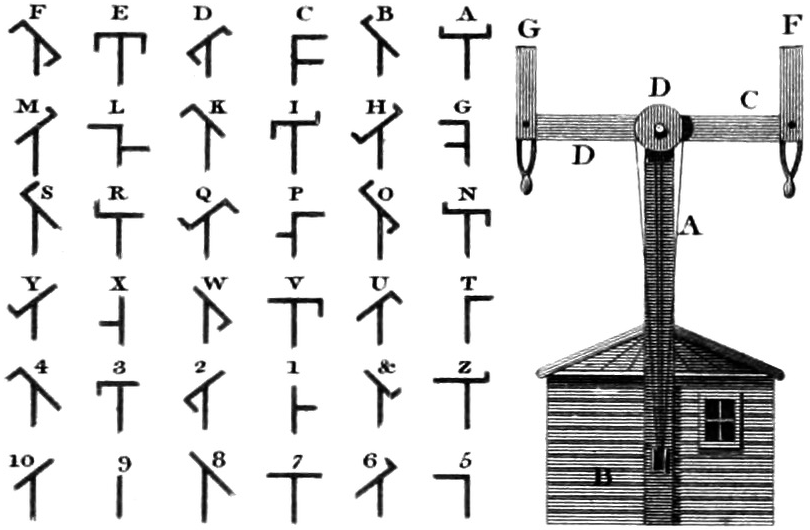
Signal Flags

- Flag position indicates the letter or number
- Two flags (Alpha) and (Numeric) indicate when to switch between letters and numbers
- Range:
  a few miles
- Speed:
  depends on receiver's experience



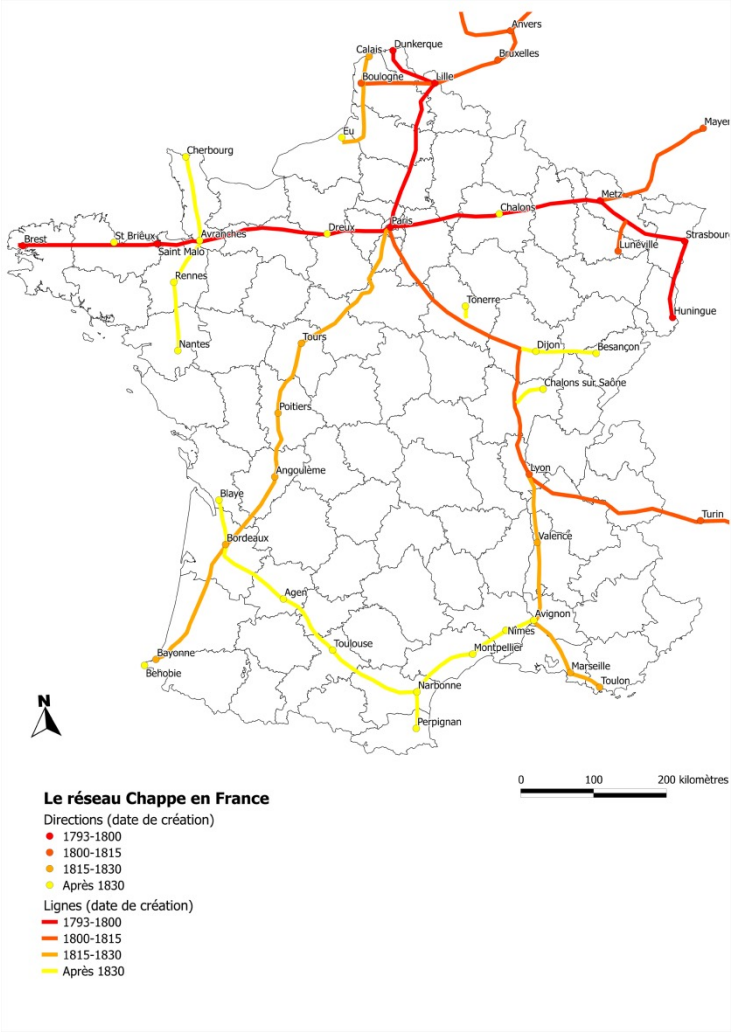
Optical Telegraph

- An "Optical Telegraph" was invented by Claude Chappe in France in the 1790's
- Called a "semaphore", from the Greek: 'sign carrying'



Optical Telegraph

- Chappe developed a large network of stations across France
- At its peak, it included:
  - 534 stations
  - covered 3000 miles
- Napoleon Bonaparte used Chappe's system to send messages to warships in port and to armies in the field

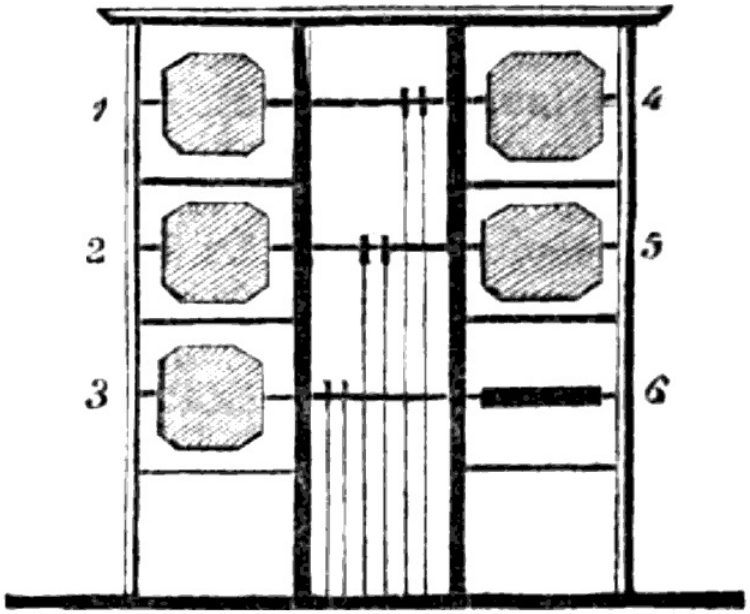


Optical Telegraph

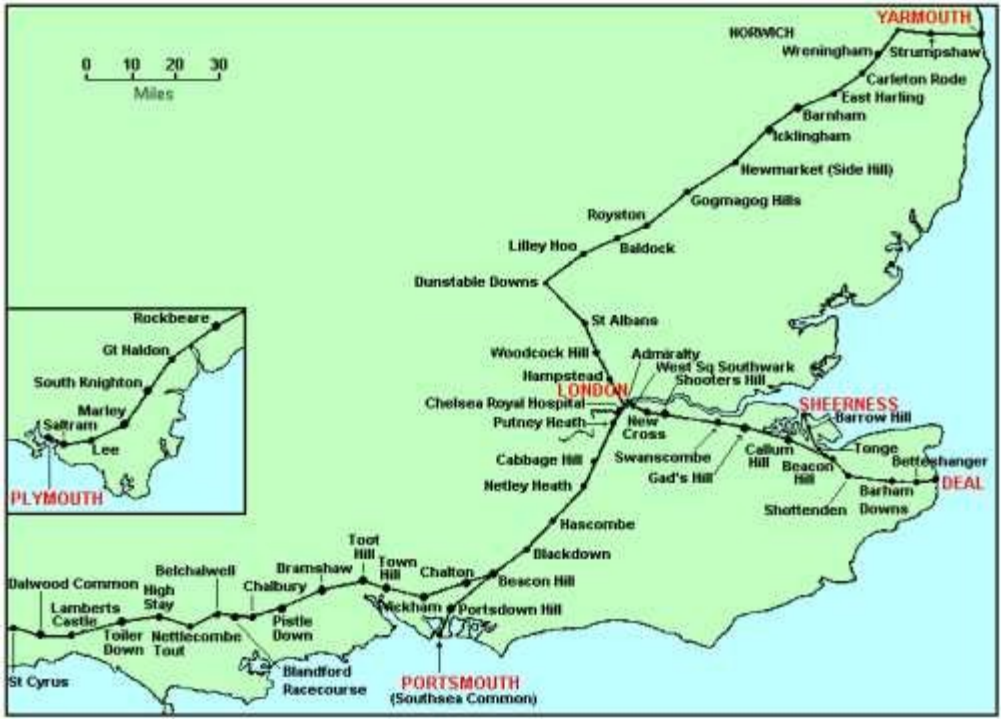
- British developed two different versions:
  - one used used arms on poles, like Chappe's
  - another had octagonal panels (shutters) that turned on an axis to show or hide the panel
- A series of towers from coastal towns to London could pass messages in minutes
  - Optical telegraphs were used in Europe until the electric telegraph was developed in the middle of the 1800's

Optical Telegraph

- Optical shutter telegraph
  - similar to a 6-bit binary code



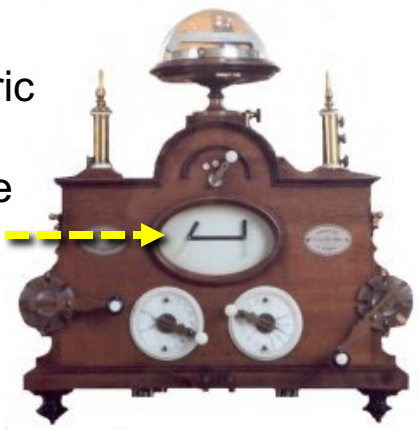
- Chain of stations in England

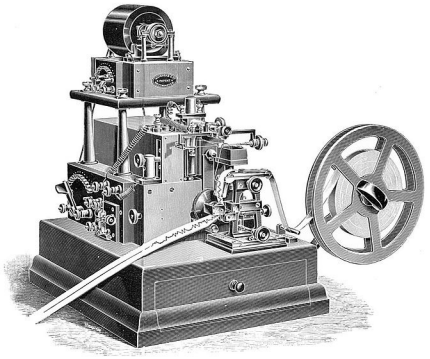


Electromagnetic Telegraph

- Based on the invention of the Electromagnet by William Sturgeon (1783-1850)
- Refined in the 1830s and 1840s
  - Sir William Cooke (1806-79) and Sir Charles Wheatstone (1802-75) in England, Samuel Morse (1791-1872), Leonard Gale (1800-83) and Alfred Vail (1807-59) in the US
- Several technical developments and improvements

(1850) Electric Telegraph, Still using the semaphores proposed by Chappe

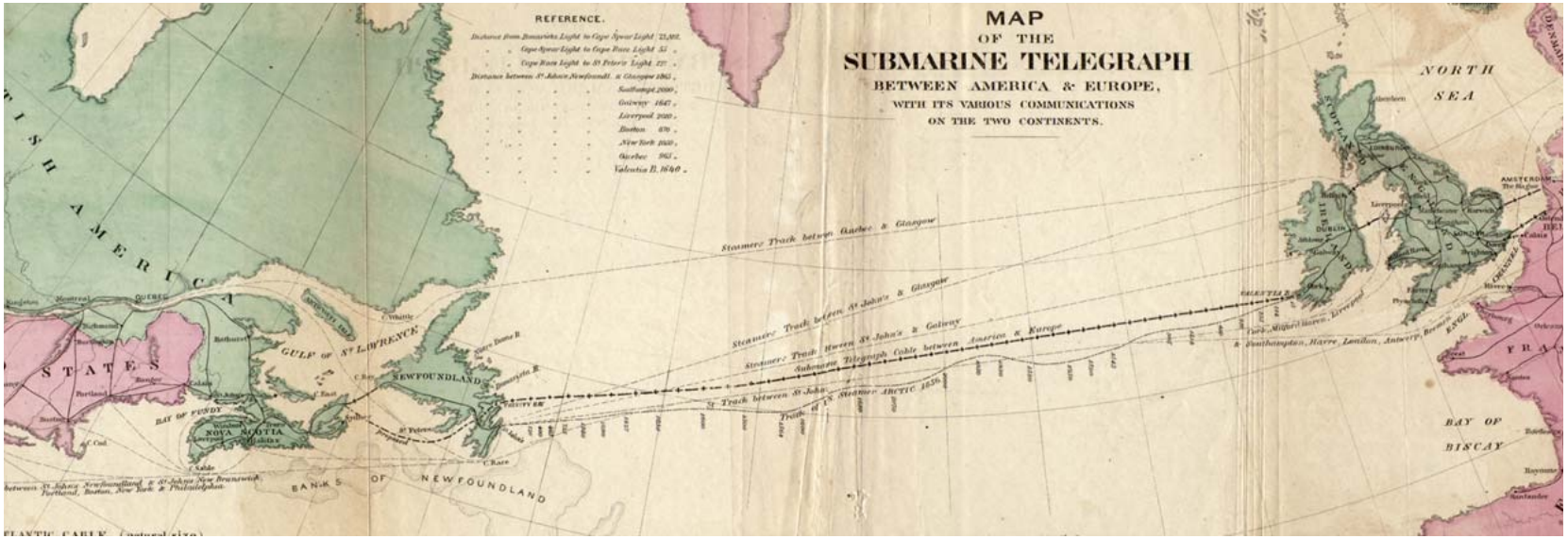




printing telegraph

Electromagnetic Telegraph

- Oceanic cables were deployed and by the late 1800's the telegraph covered much of the world

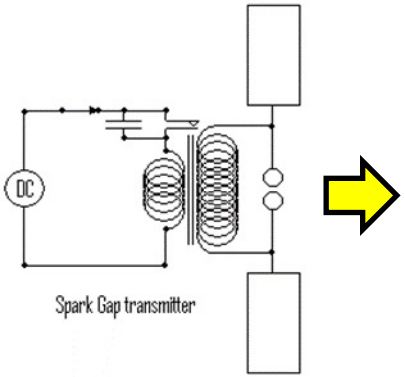


- In 1876, the invention of the telephone started the transition of wired communications to voice

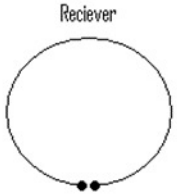
Wireless Communications

Heinrich Rudolf Hertz

- Born 22 February 1857 (Germany)
- 1880 – PhD from University of Berlin
- 1885 – Professor at University of Karlsruhe
- Based on the 1887 Michelson-Morley experiment (which tried to detect the existence of *aether*)



Created a spark at the transmitter end and detected it at the receiver with no physical connection

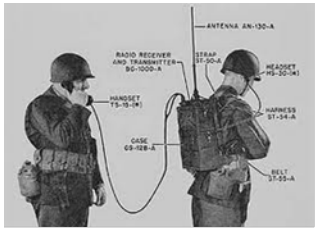


The Father of the Radio

Guglielmo Marconi

- Born 25 April 1874 (Bologna, Italy)
  - In 1894 starts studying the work of Heinrich Hertz and recreating Hertz‘s experiments
- Greatly improves the performance of a Hertz apparatus, reaching a 2 Km range in 1895.
- By 1899, Marconi was transmitting messages across the English Channel and to ships at sea
- He received a U.S. patent on an “Apparatus for Wireless Telegraphy” in June, 1901
- In 1909, he received the Nobel Prize in Physics
- Marconi died in Rome in 1935 at the age of 63.
  - Wireless stations throughout the world fell silent for 2 minutes in tribute.

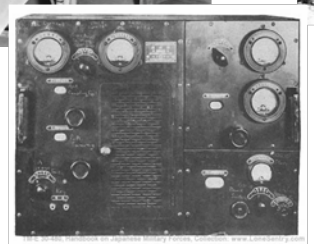
World War II
Advances in Wireless Communication



SCR-300 (1940)


Submarine Radio Room

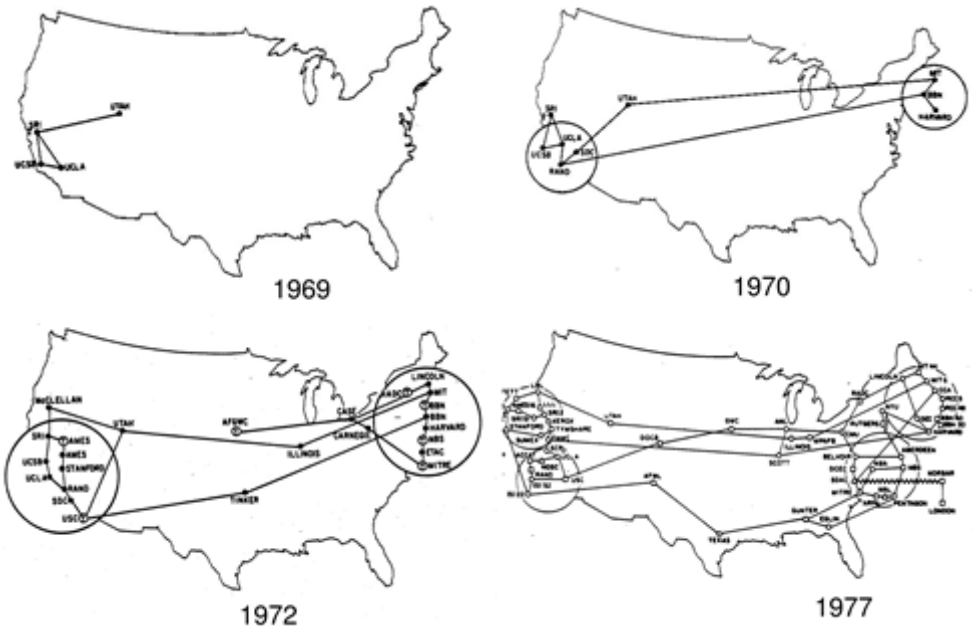

German Field Radio


Japanese Field Radio (1944)

- Military used encrypted wireless signals to transmit battle plans
- Electronic navigation systems used by both Allied and Axis planes for more accurate night-time bombing
  - *Knickebein* used by the Luftwaffe starting in 1939
  - Effect mitigated by interference from ground stations in England
    - ("Battle of the Beams")

The Cold War and ARPANET

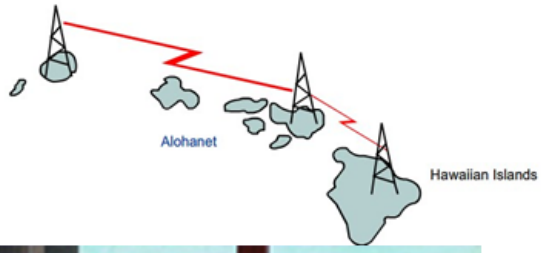
- In 1957, the USSR launched **Sputnik**, the first orbiting satellite
- In the same year, the Advanced Research Projects Agency (ARPA) was founded in the United States to develop information technologies that could survive a nuclear attack
  - In 1962, ARPA created the Information Processing Techniques Office to explore information technology issues, which lead to *ARPAnet*



ARPAnet - the world's first multiple-site computer network was created in 1969.

The Birth of Wireless Networking

- In 1971, researchers at the University of Hawaii, lead by Norman Abramson, created the first "packet-switched" radio communications network entitled "**Alohanet**"
  - Alohanet was also the first *wireless local area network*, (WLAN)
- The **ALOHA** protocol was specifically developed to allow geographically distributed users to share a single broadcast communication medium
  - In 1972, Alohanet connected with Arpanet on the mainland. The length of this network connection was ground-breaking in telecommunications between computers



The Aloha Protocol

- Numerous improvements followed…
- Carrier Sense Multiple Access Protocols
  - CSMA/CD – Carrier Sense Multiple Access with Collision Detection
  - CSMA/CA – Carrier Sense Multiple Access with Collision Avoidance
- These are Local Area Network Protocols, concerned with accessing and sharing a common medium (wire or air)
- These protocols are used in shared-media networks like Ethernet (wired LAN) and WiFi (wireless LAN)

Ethernet

- **1973:** On May 22, 1973, Bob Metcalfe at Xerox PARC documented the invention of Ethernet in a memo, which described communication across different "*ethers*" - including cable, telephone, and radio - building on the ALOHAnet protocol.
  - **1973:** Bob Metcalfe, David Boggs, and Tat Lam built the first Ethernet prototype at 2.94 Mbps.
  - **1979:** Bob Metcalfe founded 3Com to commercialize the Ethernet.
  - **1983:** 802.3 (Wired LANs) specification formally approved by IEEE.
  - **1999:** 802.11a/b (Wireless LANs) specification approved
  - **2003:** 802.11g amendment
  - **2009:** 802.11n amendment
  - etc…..

Meanwhile, In the Early 70's

- Two projects began independently, with the same goal: to define a unifying standard for the architecture of **networking systems**.
  - One was administered by the *International Organization for Standardization (ISO)* (note: ISO is not an abbreviation of the name)
  - The other was undertaken by the *International Telegraph and Telephone Consultative Committee*, or *CCITT* (the abbreviation is from the French version of the name)
- In 1983, these two documents were merged together to form *The Open Systems Interconnection Reference Model (OSI Reference Model)* – Published in 1984/85.
- However at around the same time...

Back at DARPA…

- In **1973**, development of a full-fledged system of internetworking protocols for the ARPAnet began.
- The ideas behind a *transport level protocol* (TCP) were first published in 1974
- In **1977**, Jon Postel realized that the protocol should be separated into two different layers to improve modularity

  > *"I suggest that a new distinct internetwork protocol is needed, and that TCP be used strictly as a host level end to end protocol."*
  >
  > -- *Internet Engineering Note number 2*, Jon Postel, IEN 2, 1977

- The TCP/IP Architectural Model and the first formal standards for the versions of IP and TCP used in modern networks (IP version 4) were created in **1980**

Issues with Communications

- Potential Communication Problems:
  - the problem of *shared* communications media
    - how to recognize individual senders?
    - how to take turns sending?
  - the problem of *verifying delivery* of messages
    - verification of the receipt of messages
    - issues with timeouts, delays, error messages
  - *insufficient bandwidth* (there is never enough)
    - data volume grows as fast as bandwidth increases

Network Programming

- All modern languages and operating systems support network programming
  - easy to do in Python, Java and C/C++
  - they all have libraries to support creating network connections and transferring data
  - similar to file I/O, the first step is to *open* a connection between the computers
  - then, data is transmitted and received much like performing *read* and *write* on a file

Programing Example - C

```
#include<stdio.h>
#include<sys/socket.h>      // library for network sockets
#include<arpa/inet.h>       // library for internet addresses

int main(int argc , char *argv[])   {
      int socket_desc;
      struct sockaddr_in server;          // for the computer you are connecting to
      char message[1024], reply[1024];
      socket_desc = socket(AF_INET , SOCK_STREAM , 0);    // create socket
      server.sin_addr.s_addr = inet_addr("74.125.235.20");
      server.sin_family = AF_INET;
      server.sin_port = htons( 80 );
      connect(socket_desc , (struct sockaddr *)&server , sizeof(server));
      send(socket_desc , message , strlen(message) , 0);  // writing data
      recv(socket_desc, reply , 1024 , 0);                // reading data
      puts(reply);
      close(socket_desc);
}
```

Programing Example - Java

```
import java.io.*;
import java.net.*;
class simpleClient   {
   public static void main (String args[])  throws IOException   {
      BufferedReader stdin =
            new BufferedReader(new InputStreamReader(System.in));
      Socket sock = new Socket("host1.cs.fit.edu", 4567); // host, port
      PrintWriter output = new PrintWriter(sock.getOutputStream());
      System.out.println("\nType characters, then press Enter:");
      String line = stdin.readLine();
      output.println(line);
      sock.close();
   }
}
```

Socket creates a network connection to the computer at host1.cs.fit.edu and then an output stream is created to send data to that computer

Programing Example - Python

```
#!/usr/bin/python

import socket                          # import the socket module

sock = socket.socket()                 # create a socket object

host = socket.gethostname()            # get the name of the local machine

port = 12345                           # assign a port number

sock.connect((host, port))             # connect to that computer

print sock.recv(1024)                  # receive and print up to 1024 bytes
                                       # of data from the other computer

sock.close()                           # close the socket connection
```