2 The Making of the Internet

2.1 The Beginning

- **Mid-1940s**: Early vision — maximize computer usage during idle time.
- **Progress**: Decoupling of terminals from processors.
- **1969 - DARPA Project (California)**:
    - Goal: Connect DARPA scientists and provide access to university research.
    - Outcome: First computer network — **ARPANET**.

2.2 First Challenge: The Switching Problem

- **Circuit Switching**:
    - Inefficient for data transmission.
    - Used by phone lines at the time.

- **Packet Switching (Still in use today)**:
    - Breaks data into smaller chunks (packets).
    - Packets travel individually through the network to the destination.
    - Each computer:
        - Has a unique address.
        - Passes packets forward based on destination info.
        - Stores packets until all arrive, then reassembles them.

2.3 Second Challenge: Keeping Address Records Updated

- Problem: A single change in address could break the network.

- **1973**: Stanford becomes central address keeper.

    - Only 60 computers connected.

    - Now, instead of each computer storing all addresses, they reference a central **address book**.

2.4 Third Challenge: Standardizing Communication (Packet Formatting)

- Problem: Different networks = different formats.
- Solution: **1983 – TCP/IP Protocol** (still the backbone of the Internet).

🧱 TCP/IP Explained

- **Application Layer**:
  - HTTP (Browsers like Chrome)
  - FTP (e.g., Cyberduck)
  - POP3, SMTP (Emails)
  - SNMP (Printers/Routers)

- **Transport Layer**:
  - **TCP**: Reliable delivery, error checking (e.g., website loading).
  - **UDP**: Faster, but no delivery guarantee (e.g., DNS).

- **Network Layer**: IP (identifies computers).

- **Physical Layer**:
  - Ethernet (LANs)
  - ARP (Finds device IPs in local networks)

➡️ For web developers, the **Application Layer** (especially HTTP & FTP) is most relevant.

➡️ TCP/IP enabled **inter-network communication**, giving rise to what we now call:

🌍 THE INTERNET

In 2019, over 12 billion devices were connected to it.

2.5 Fourth Challenge: Managing Growing Address Lists

- **1983**: ~1,000 computers on ARPANET.
- Manual IP tracking became inefficient.
- **DNS (Domain Name System)** invented.

🔍 What is DNS?

- Acts like a **phonebook** for the Internet.
- Converts IP addresses into human-readable **domain names** (e.g., google.com).
- Managed by **ICANN**, via **Domain Registrars** (GoDaddy, Google Domains, etc.).

➡️ You'll learn more about DNS in the **Deployment** section later.

2.6 Rise of Internet Service Providers (ISPs)

- Late 1980s: DARPA hands off ARPANET → **NFSNET**.
- ISPs emerge to connect individuals to the internet.
    - Phone companies (AT&T, Verizon) lead initially via dial-up.
    - Later, cable providers (e.g., Comcast) dominate via broadband.
- **1995**: NFSNET shuts down, ISPs take over fully.

2.7 Fifth Challenge: Organizing Internet Content

- Problem: Rapidly growing information, hard to navigate.
- **1989**: **Tim Berners-Lee (CERN)** proposes:
    - Break large docs into smaller ones.
    - Add links (**HyperText**).
    - Use tags to provide structure.

🌐 His Solution:

- **HTML**: HyperText Markup Language
- **HTTP**: HyperText Transfer Protocol
- **1993**: Released to the public as the **World Wide Web (WWW)**

➡️ Your role as a web developer: Use **HTML** to structure and share content online.

2.8 Sixth Challenge: Information Overload

- Mid-1990s: Explosion of shared content.
- **Two Stanford students** set out to solve:
    1. What info exists online?
    2. What topic does each doc cover?
    3. Which doc is most relevant?

Their Solution:

- Crawl → Index → Rank (via backlinks)

➡️ In **1998**, they released:

🔍 GOOGLE — for free!

2.9 The Rise of Web 2.0

- **What is Web 2.0?**
  - Shift from static to **dynamic** sites.
  - Data stored in databases.
  - User-generated content becomes central.

- **Platforms**: Facebook, YouTube, Wikipedia, etc.

- Web developers now handle:
  - Business logic
  - Databases
  - Templates
  - Design (a key differentiator)

2.10 The Rise of Web Frameworks

- As web development got complex, developers needed **starting points**.

- Frameworks bundled common functionality for faster builds.

Popular Web Frameworks:

- **Backend**: Express (JS), Django (Python), Rails (Ruby), Laravel (PHP), ASP.NET (C#)

- **CMS**: WordPress, Drupal, SquareSpace, Wix, SharePoint

- **Frontend**: Bootstrap, Angular, Vue.js

EVANGADI

2.11 The Smartphone Revolution

- **2007**: First iPhone launch.
- **Mobile traffic stats**:
    - 2013: 16%
    - 2018: 52%
- Shift in priority: **Mobile-first** design dominates.

2.12 Power Players of the Internet (Hierarchy of Control)

1. **Device Makers**: Apple, Samsung, Huawei
2. **OS Providers**: iOS, Android, Windows
3. **ISPs**: AT&T, Verizon, Comcast
4. **Search Engines**: Google, Bing
5. **Browsers**: Safari, Chrome, Edge
6. **Apps & Platforms**: Facebook, Gmail, TikTok, IG

The real game now: **Who controls your data & attention**.

2.13 What to Focus On as a React Web Developer

🖥️ On Your Computer

- Master your OS
- Improve speed (shortcuts, typing, performance)

🌐 On the Internet

- Master:
    - Google & StackOverflow
    - HTML, CSS, JavaScript
    - HTTP & DNS
    - **React**

🔗 Useful Links:

- StackOverflow:
    - React
    - HTML
    - CSS
    - JavaScript

📆 For Next Week:

- HTML Tutorial (YouTube)
- CSS Tutorial (YouTube)