

# Web Development

---

Technical Summer School 2019, IIT Bombay – Parth Patil

Part 1 – Introduction to the Web and HTML

Courtesy – Varun Patil

# Parth Patil

2<sup>nd</sup> Year Undergraduate, Electrical Engineering

Django, Angular, Node.js, REST, ROS, GTK, Android Studio, ...

Python, C++, HTML, JS/TS, Java ...



# Prerequisites

---

- Basic Hardware
- Desktop OS – Windows, GNU/Linux or macOS
- Python
- Web Browser - Mozilla Firefox or Google Chrome
- Code Editor – Notepad++, VS Code, Sublime etc.
- Basics of Programming – CS101

# What is the World Wide Web?

---

- What we see in a Web Browser!
- HTML – not really
- A set of documents connected to each other.
- A system of **Internet servers** that support specially formatted documents, supporting links to other documents as well as graphics, audio and video files.
- Are *Web* and *Internet* synonymous? – No!

# What is the Internet?

---

- A lot of connected devices – a network – which talk to each other
- A global computer network providing a variety of information and communication facilities, consisting of interconnected networks using **standardized communication protocols**



# The Seven Layers of OSI\*

---

1. Physical Layer
2. Link Layer
3. **Network Layer**
4. **Transport Layer**
5. Session Layer
6. Presentation Layer
7. **Application Layer**

# The Network Layer

---

- Addressing (IPV4/IPV6)
- Routing
- Path Planning

# Internet Protocol

---

- What is a protocol – standardized communication
- Headers and body of packets
- Protocol used by the network layer
- Each device has a unique **IP Address** – like your postal address
- 32-bits – xxx.xxx.xxx.xxx (in IPv4)
- Best effort



# The Network Layer

---

- $A \rightarrow B$  – A and B can talk – we already have this
- $A \rightarrow B \dots C$
- $A \rightarrow B \dots B \rightarrow C$
- $A \rightarrow B \dots B \rightarrow C \dots A \rightarrow C$
- What if D comes up? -  ${}^4C_2 = 6$
- Eventually ...  ${}^{50}C_2 = 1225 \dots {}^{300}C_2 = 44850$

# Relaying Information

---

- $A \rightarrow B \rightarrow C \rightarrow D$
- Or maybe  $A \leftarrow B \rightarrow C, D$
- Routers – devices designed for this – **B**
- Switches – Layer 2

# The Seven Layers of OSI

---

1. Physical Layer
2. Link Layer
3. **Network Layer** ✓
4. **Transport Layer**
5. Session Layer
6. Presentation Layer
7. **Application Layer**



# The Transport Layer

---

- Flow Control
- Error Control
- Congestion control
- Order of receiving - multiple paths of communication

# Transmission Control Protocol

---

- Built into Operating Systems – with standards
- Performs error detection/correction
- Ensures correct ordering of data
- Allows multiple applications to communicate with **Ports**

# Port

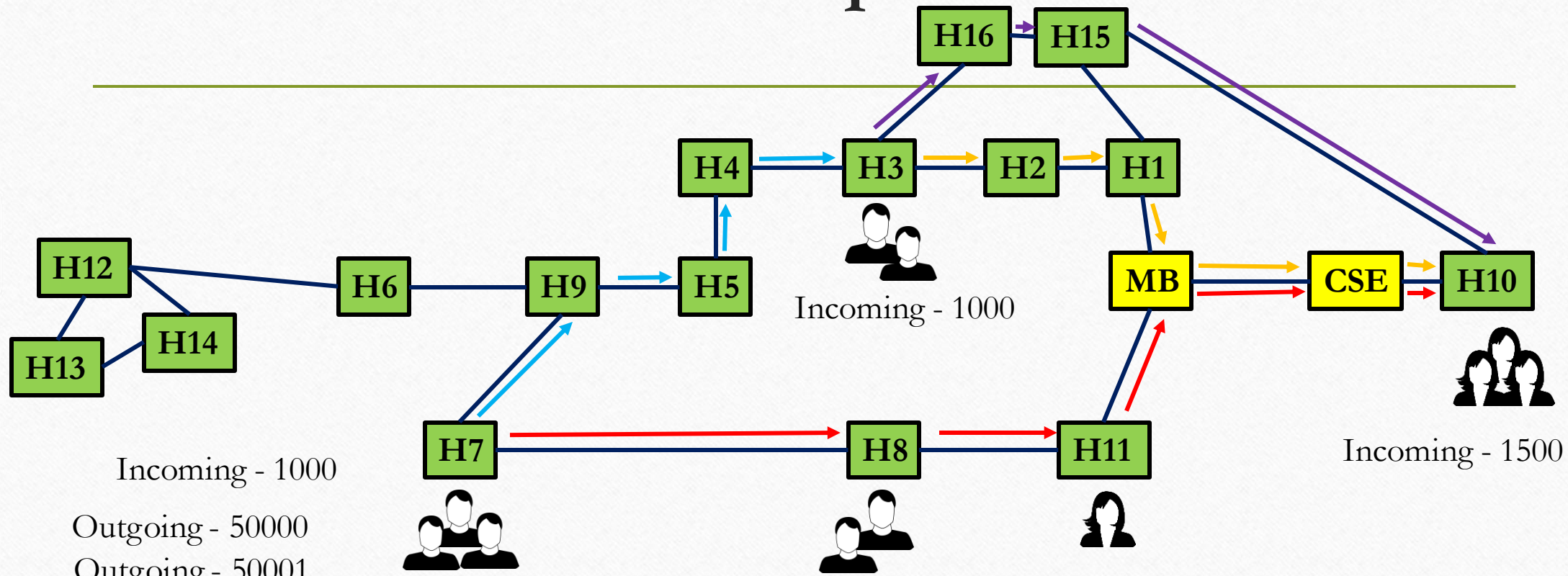
---

- 16-bit number – 0 to 65535
- Outgoing and incoming ports
- Can receive multiple connections on one port
- 4-tuple – identifying a unique connection
  - IP Address of A
  - Port of A
  - IP Address of B
  - Port of B



# An Analogy

# A Map?!



Incoming - 1000  
Outgoing - 50000  
Outgoing - 50001  
Outgoing - 50005

Incoming - 1000

Incoming - 1500

# The Seven Layers of OSI

---

1. Physical Layer
2. Link Layer
3. **Network Layer** ✓
4. **Transport Layer** ✓
5. Session Layer
6. Presentation Layer
7. **Application Layer**



# The Application Layer

---

- Multiple protocols like HTTP, FTP etc.
- **HTTP** – Hypertext Transfer Protocol
  - Can transfer any type of content
  - Primarily for text – Hypertext i.e. with Hyperlinks
  - Protocol takes care of only transferring data
  - Understood by **Web Browsers**
  - Not the same as HTML – HTML is usually transferred over HTTP

# HyperText Transfer Protocol

---

- Headers
  - What is being transferred - **URL**
  - Size of content
  - Type of file – MIME\* type
  - Extra information related to server
  - Extra information related to content
- Body
  - Actual contents of the file – the message

# Uniform Resource Locator

---

- Reference to a **web** resource that specifies its location on a computer network
- Usually used with HTTP
- Send as part of HTTP header when requesting a resource



# Uniform Resource Locator

---

- `scheme://authority/path?query#fragment`  
authority = `userinfo@host:port`
- `scheme` - usually `http`, can be ftp etc.
- `path` - path of resource we want - **known**
- `port` is usually (and defaults to) `80`
- `query` - for passing extra information

# Uniform Resource Locator

---

- `http://www.iitb.ac.in/newacadhome/timetable.jsp`
  - `http://` → using Hypertext Transfer Protocol
  - `www.iitb.ac.in` → authority
    - port not specified → 80
    - “resolves” (see DNS) to an IP like 10.102.1.111
  - `newacadhome/timetable.jsp` → path
  - No query or fragment specified
- Server sends a response with the requested page

# Web Browser

---

- Takes in a URL and makes an HTTP request for you
- Receives the content and understands it
- Displays it to the user
- Allows the user to interact with the received content
- Makes more requests



# The Seven Layers of OSI

---

1. Physical Layer
2. Link Layer
3. **Network Layer** ✓
4. **Transport Layer** ✓
5. Session Layer
6. Presentation Layer
7. **Application Layer** ✓

# HyperText Markup Language

---

- A markup language is a system for **annotating** a document in a way that is **syntactically distinguishable** from the text
- Hypertext Markup Language is the **standard** markup language for creating web pages and web applications
- HTML elements are the building blocks of HTML pages
- Represented by **tags**

# HTML Tags

---

- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page
- For example
  - `<b>` - Make the text bold
  - `<p>` - Begin a new paragraph
- Closed as `</tag>` e.g. `<b>This is bold</b>`
- Just syntax



# Basic HTML Skeleton

---

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

# <a> and <img>

---

```
<a href="http://www.iitb.ac.in/newacadhome/timetable.jsp">
```

```
    IITB Timetable
```

```
</a>
```

```

```

- Things to note:
  - **href** and **src** are **attributes**, the expressions in quotes are **values**
  - One tag can have one or more attributes (or none)
  - Attributes control content in the tag
  - **img** has no end tag
  - **image.jpg** indicates same path as the open page

# Tag nesting

---

```
<ul>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ul>
```

- Coffee
- Tea
- Milk

`<ul>` - Unordered List  
`<li>` - List Item  
`<ol>` - Ordered List



# List of common tags (Non-exhaustive)

---

- `<html>` - HTML document
- `<body>` - Main body
- `<h1>` - Biggest heading, `<h2>`, `<h3>` are progressively smaller
- `<b>` - Bold text
- `<a>` - Hyperlink
- `<img>` - Image - no end tag
- `<button>` - Button!
- `<div>` - Division
- `<p>` - Paragraph
- `<br>` - Line Break - no end tag
- `<pre>` - Preformatted text

Thank You!