WWW

(Fitzgerald and Dennis, 2005 Figure 2.8)
Browsers

- Example Browsers (User Agent)
  - Firefox
  - Chrome
  - Internet Explorer

- Present online information in a readable format to the user
  - Done by interpretation of a web programming language called HTML
HTML

- Hypertext Markup Language (HTML)
  - World Wide Web Programming Language
    - tree of nested tags that tell browsers how to read and present data to the user
    - Uses tags to accomplish page formatting
  - Not actually a programming language
    - Lacks computing ability (try to compute the value of an arithmetic expression in HTML—you cannot)
Navigating Around the Web

- Links or URLs
  - The URL (Uniform Resource Locator) is the HTML’s specific web address
  - URL by Book-marking or Favorites
Addressing on the Internet

- InterNIC
  - Domain Name registration
  - Sets rules
  - Assigns IP Addresses
  - Organizes participant categories
    - Universities, Businesses
    - Networks, Organizations
    - Military, Government
Domains

- **First-Level Domain Extender**
  - a.k.a. Top-Level Domain
    - Thee letter code added to each address (e.g.: .COM)
    - Com = commercial
    - Edu = education (university)
    - Gov = Government
    - Mil = Military
    - Net = Network support company
    - Org = nonprofit organization
Flat addressing

- Does not provide anything other than a unique identifier. They provide no real information about where the object being addressed resides.
- Example: SSN# (may provide insight to where the person was born, but not to where they are now)
Hierarchical Addressing

- Provides layers or a hierarchy to the address that provide information about where the addressed object exists within the hierarchy
- Example: phone numbers (area code, local prefix, and four digit number unique to that area code/prefix combination)
DNS

(Fitzgerald and Dennis, 2005 Figure 5.8)
IP Numbering

- IP Addresses on the Internet are unique
- Serve as a locator
- ⟨Source IP, port number⟩, ⟨destination IP, port number⟩ pair uniquely defines a network connection
History

- Internet vs. internet
  - An internet (small “i”) refers to any network of computers
  - The Internet (large “I”) is a specific name
    - Given to the communication network comprised of hundreds of thousands of inter-connected networks
History (cont)

- The Internet is not new
  - In the early 1960’s
    - Paul Baran of Rand Corp. envisioned email
    - Outlined proposal to US Govt. regarding the need for a communications network in case of a nuclear attack against the US
  - ARPAnet (Advanced Research Projects Agency) was created to facilitate network research
History (cont)

- 1980’s:
  - Tim Berners-Lee of CERN (European Laboratory of Particle Physics) began working on WWW
    - Software for world-wide interconnectivity
    - Berners-Lee -- Father of WWW
      - Developed the Web as a means of sharing scientific information
      - Written as an “...Internet-based hypermedia initiative for global information sharing.”
  - Internet surfing—pre-browser era
    - Before the Web, information retrieval was accomplished through a number of complicated steps and commands
    - Until the birth of Mosaic 1993
How Does The Web Work?

- The internet as a Packet-Switched network
  - Bundles of data are broken up, transmitted, and reassembled at the other end
  - Transmitted data are mixed together with all the other data on the Internet until it reaches its final destination

- Clients, Servers, and Routers
  - Clients: Originating from User’s Computer
  - Server: Powerful, Continuous Internet Access
  - Router: Computers that link networks together
Underlying Protocol

- **TCP/IP**
  - Sets definitions for how data transmission takes place
  - **TCP (Transmission Control Protocols)**
    - Delivery of data
  - **IP (Internet Protocols)**
    - Logical addressing
7-Layer Open Systems Interconnection (OSI) Model

<table>
<thead>
<tr>
<th>Layer</th>
<th>Responsible For:</th>
</tr>
</thead>
<tbody>
<tr>
<td>7) Application</td>
<td>Provides Services to User Apps</td>
</tr>
<tr>
<td>6) Presentation</td>
<td>Data Representation</td>
</tr>
<tr>
<td>5) Session</td>
<td>Communication Between Hosts</td>
</tr>
<tr>
<td>4) Transport</td>
<td>Flow Ctrl, Error Detection/Correction</td>
</tr>
<tr>
<td>3) Network</td>
<td>End to End Delivery, Logical Addr</td>
</tr>
<tr>
<td>2) Data Link</td>
<td>Media Access Ctrl, Physical Addr</td>
</tr>
<tr>
<td>1) Physical</td>
<td>Medium, Interfaces, Puts Bits on Med.</td>
</tr>
</tbody>
</table>
## OSI (1984) VS TCP/IP (older)

<table>
<thead>
<tr>
<th>TCP/IP</th>
<th>OSI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Application</td>
</tr>
<tr>
<td></td>
<td>Presentation</td>
</tr>
<tr>
<td></td>
<td>Session (Layers 7-5)</td>
</tr>
<tr>
<td>Transport</td>
<td>Transport (Layer 4)</td>
</tr>
<tr>
<td>Internet</td>
<td>Network (Layer 3)</td>
</tr>
<tr>
<td>Network Interface</td>
<td>Data Link</td>
</tr>
<tr>
<td></td>
<td>Physical (Layers 1-2)</td>
</tr>
</tbody>
</table>
## Examples

<table>
<thead>
<tr>
<th>Layer</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>7) Application</td>
<td>HTTP, FTP, SMTP</td>
</tr>
<tr>
<td>6) Presentation</td>
<td>ASCII, JPEG, PGP</td>
</tr>
<tr>
<td>5) Session</td>
<td>BOOTP, NetBIOS, DHCP, DNS</td>
</tr>
<tr>
<td>4) Transport</td>
<td>TCP, UDP, SPX</td>
</tr>
<tr>
<td>3) Network</td>
<td>IP, IPX, ICMP</td>
</tr>
<tr>
<td>2) Data Link</td>
<td>Ethernet, Token Ring, Frame Relay</td>
</tr>
<tr>
<td>1) Physical</td>
<td>Bits, Interfaces, Hubs</td>
</tr>
</tbody>
</table>
## Protocol Data Units (PDUs)

<table>
<thead>
<tr>
<th>Layer</th>
<th>PDU Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>7) Application</td>
<td>Data</td>
</tr>
<tr>
<td>6) Presentation</td>
<td>Data</td>
</tr>
<tr>
<td>5) Session</td>
<td>Data</td>
</tr>
<tr>
<td>4) Transport</td>
<td>Segment</td>
</tr>
<tr>
<td>3) Network</td>
<td>Packet</td>
</tr>
<tr>
<td>2) Data Link</td>
<td>Frame</td>
</tr>
<tr>
<td>1) Physical</td>
<td>Bits</td>
</tr>
</tbody>
</table>
Layer 1

- Defines physical medium and interfaces
- Determines how bits are represented
- Controls transmission rate & bit synchronization
- Controls transmission mode: simplex, half-duplex, & full duplex
- PDU: Bits
- Devices: hubs, cables, connectors, etc...
Layer 2: The Data Link Layer

- PDU: Frames
- Keeps Link alive & provides connection for upper layer protocols
- Based on physical (flat) address space
- Physical addresses are fixed and don’t change when the node is moved
- Medium/media access control
The Data Link Layer (cont)

- Flow control and error detection/correction at the frame level. Think collisions...
- Topology
- Ex: Ethernet, Token Ring, ISDN
- Sublayers: MAC (framing, addressing, & MAC) & LLC (logical link control – gives error control & flow control)
- Devices: switches, bridges, NIC’s
Layer 3: The Network Layer

- PDU: Packet
- End to end delivery of packets
- Creates logical paths
- Path determination (routing)
- Hides the lower layers making things hardware independent
- Uses logical hierarchical addresses
The Network Layer (cont)

- Logical hierarchical addresses do change when a node is moved to a new subnet
- Devices: routers, firewalls
Layer 4: The Transport Layer

- PDU: Segment
- Service Point Address (more often called a port) used to track multiple sessions between the same systems. SPA’s are used to allow a node to offer more than one service (i.e. it could offer both mail and web services)
- This layer is why you have to specify TCP or UDP when dealing with TCP/IP
The Transport Layer (cont)

- Must reassemble segments into data using sequence numbers
- Can use either connectionless or connection oriented sessions
- Connectionless sessions rely on upper layer protocols for error control and are often used for faster less reliable links
- Ex: UDP (used by things like NFS & DNS)
Connection oriented sessions require the sender to first request a connection, the receiver to acknowledge the connection, and that they negotiate how much data can be sent/received before its reception is acknowledged.

- Uses acknowledgements & retransmission for error correction.
- Example: TCP (used by things like telnet, http).
Layer 5: The Session Layer

- PDU: Data (from here on up)
- Sometimes called the dialog controller, this layer establishes, maintains, and terminates sessions between applications
- Sets duplex between applications
- Defines checkpoints for acknowledgements during sessions between applications
The Session Layer (cont)

- Provides atomization – Multiple connections can be treated as one virtual session. If one fails or is terminated, all should be terminated.
- Identifies raw data as either application data or session control information.
- Uses fields provided by layers 3 & 4 to track dialogs between applications / services.
- Provides translations for naming services.
- Ex: RPC, X-Windows, LDAP, NFS.
Layer 6: The Presentation Layer

- Data formatting, translation, encryption, and compression
- Ex: ASCII, EBCDIC, HTML, JPEG
Layer 7: The Application Layer

- Provides communication services to applications
- Ex: HTTP, FTP, SMTP
3-Tier Architecture

(Fitzgerald and Dennis, 2005 Figure 2.5)
3-Tier Architecture

- **Client tier (top tier)**
  - Application’s user interface
  - Users interact directly with the application through this tier

- **Middle tier**
  - Implements business logic and presentation logic
  - Control interactions between application clients and application data

- **Information tier (data or bottom tier)**
  - Maintains data for the application
  - Stores data in a relational database management system (RDBMS)
Scripting: Client-Side VS Server-Side

- **Client-side**
  - Validate user input
    - Reduce requests needed to be passed to server
    - Access browser
    - Enhance Web pages with DHTML, ActiveX controls, and applets

- **Server-side**
  - Executed on server
  - Generate custom response for clients
  - Wide range of programmatic capabilities
  - Access to server-side software that extends server functionality
References


- PHP Tutorial
  
  http://www.youtube.com/watch?v=afgyNp5HueQ&feature=related