The World Wide Web: Web-based Applications and Web Forms

Introduction to the World Wide Web and HTML

- HTML: HyperText Markup Language
  - Language used to specify hypertext document content and how it is to be displayed
- Hypertext
  - Non-sequential reading and writing
  - Text contains embedded hot words that are links to other documents
  - Hypermedia
    - Links can be references to non-textual information
- Most recently, XHTML: Extensible Hypertext Markup Language

World Wide Web (WWW)

- Created at CERN (Switzerland high energy physics lab) by Tim Berners-Lee (1991)
- Hypertext-based system for finding and accessing internet resources
- Huge set of hypertext-linked documents on many computers

WWW Important Acronyms

- URL (Uniform Resource Locator)
  - The web "address" of a document (page)
- HTTP (HyperText Transfer Protocol)
  - Protocol that specifies how a document is transferred
    - Defines how web browsers and web servers communicate with each other over a TCP/IP connection

URL Format

access protocol //: domain address / directory path / filename
(type of object) (computer) (virt. directory) (file name)

- Example:
  - http://cs.binghamton.edu/~reckert/360/topics.html

Web Applications, Web Servers and Web Browsers

- Windows applications (Windows Forms) run on one computer
- Web applications require two programs
  - Usually run on two different computers connected over the internet
- Web Server
  - Program that provides web documents (pages) to client applications running on other machines on the Internet
  - Pages are stored on the Web Server computer
- Web Browser
  - Client program that interprets the HTML of a page provided by a Web Server and displays it
    - Common ones: Internet Explorer, Netscape Navigator, Mozilla FireFox
    - Pages may also contain programming logic in the form of a script that's executed on the client machine (VBScript, JavaScript, etc.)

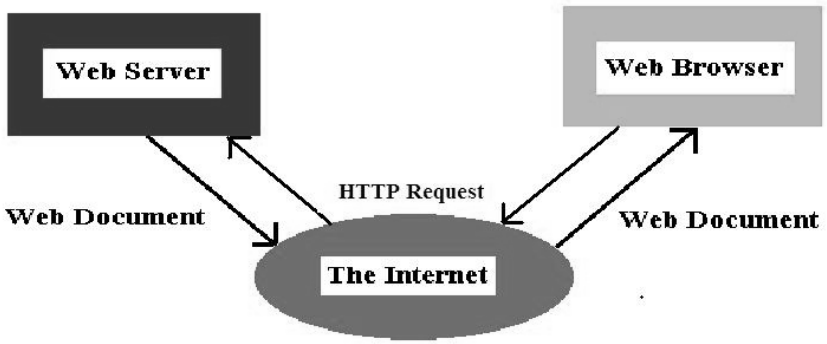
Brief Introduction to JavaScript

- Not a true programming language
- Only works in a web browser
- How it works:
  - Browser loads the page
  - Detects JavaScript <script> tag
  - Passes the script to the JavaScript Interpreter
  - Interpreter evaluates and performs the script
  - Interpreter passes HTML back to Browser
  - Browser displays the page
- Note that this is client-side processing

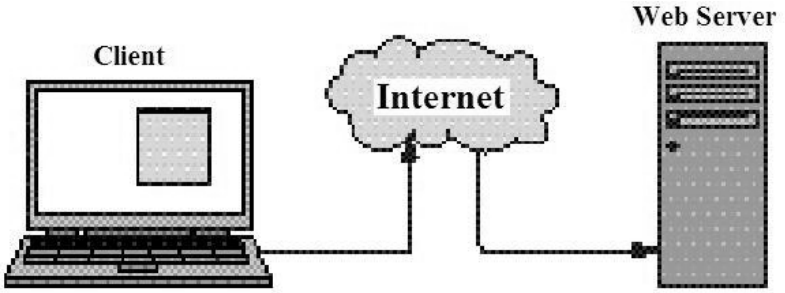
What Happens when a Web Page is Started

- When user starts a web page, the browser sends an HTTP request to the Server
- Server responds by sending the page
  - May be a preformatted HTML file
  - Or the program running on the Server may dynamically generate the HTML
    - This is server side processing
- A request to view a Web page requires a round trip to server that stores the page

Web Servers and Browsers



A Simple HTTP Transaction (1)

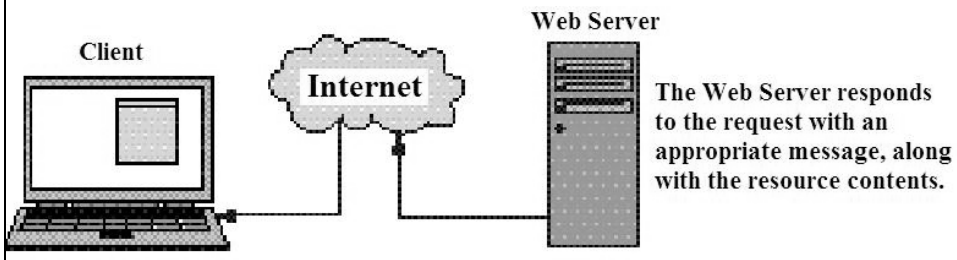


1. The client sends a GET request to the Web Server

2. After the Web server receives the request, it searches through its system for the resource

Client interacting with Web server.
Step 1: The **GET** request
`GET /books/downloads.htm HTTP/1.1`.

A Simple HTTP Transaction (2)

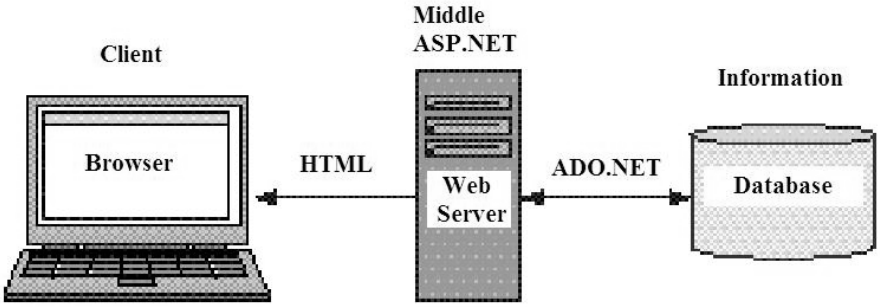


Client interacting with Web server.
Step 2: The HTTP response
`HTTP/1.1 200 OK`.

Distributed System Architecture on the Web

- Multi-tier Applications
  - Web-based applications (n-tier applications)
  - Tiers are logical groupings of functionality
  - Can be on the same computer, but usually on different ones
- Information Tier (data tier or bottom tier)
  - Maintains data pertaining to the applications
  - Usually stores data in a database management system on a separate computer
- Middle Tier (business logic)
  - Acts as an intermediary between data in the information tier and the application's clients
  - Processes client requests and retrieves and processes data from the Information Tier
  - Typically a Web Server or Web Application
- Client Tier (top tier)
  - Application's user interface
  - Typically a browser

Distributed System Architecture



Web Application Development

- Different ways of developing web pages/sites in the Microsoft Windows world:
  - Write them in HTML
    - "First generation" programming model (early 90s)
    - Good for static pages: No user data input or client/server processing
    - Dynamic pages: client or server side processing required
      - HTML Can be used with CGI for dynamic pages
    - Usable on any platform, but slow with CGI
  - Use ASP (Active Server Pages)
    - "Second generation" programming model (late 90s)
    - More powerful and easier to use
    - Relatively slow: interpreted
  - ASP.NET
    - "Third generation" programming model (21st century)
    - Powerful, flexible, and easy to use
      - Object-oriented, event-driven
      - Same paradigm as Windows Forms
    - Fast: compiled

ASP.NET and Web Forms

- ASP.NET
  - Provides libraries, controls, & programming support to write Web applications that interact with the user, render controls, display data, and generate appropriate HTML
  - Using C# and ASP.NET we can create object-oriented, event-driven Web applications
- Web Forms
  - Windows Form applications run standalone on the local machine's Windows environment
  - Web Form applications run on a server on a different computer
    - Web pages are built around controls and event handlers
    - ASP.NET Extends the ideas of Windows Forms to distributed computing over the Web

C# and ASP.NET

- An ASP.NET Web Form has two pieces:
  - HTML needed to render the page
    - A file with .aspx extension
  - Code that contains program logic to:
    - Interact with user, respond to events, render controls, display data, generate appropriate HTML
    - In C#, a file with .aspx.cs extension
      - The "code-behind" file
    - Notice the separation of page content and processing

Web Application Development with Visual Studio 2005/2008

Host computer must have a "Web Server" program running

- Microsoft's Internet Information Services (IIS) is usually the Web server in the Microsoft Windows environment
  - It's part of Windows 2000, XP Professional, Vista and Server 2003 OS
- When running IIS on a local machine you are hosting a Web server that must be secured
- Must have administrative rights to create IIS-served Web applications
- By default, the files are stored in c:\inetpub\wwwroot

- An alternative is a File System Web site
  - Stores Web pages and associated files in a folder on the local machine
  - Pages are tested using the new Visual Studio Web Server program
    - But cannot be served to browsers running on other computers
  - VS Web Server doesn't expose the computer to security vulnerabilities and doesn't require administrative rights
  - Can run on Windows XP Home OS
  - VS Web Server web applications can be ported to IIS

First, Look at Web Application Development using HTML

- See Appendices F and G of your Deitel text
- Markup language for preparing WWW hypertext documents
- Specifies what is to be displayed and how it is to be displayed
- Subset of SGML
  - Standard Generalized Markup Language
- Result--just a text file (a script)
  - extension .html or .htm
- Used to set up static web pages

Main Tasks in HTML

- Define Tags
  - Basic element of HTML
  - Specify what is to be displayed and how it is to be displayed
- Define hypertext Anchors and Links
  - For navigating nonsequentially (hypertext)
- Format the document
  - In general terms (details handled by browser)

Preparing HTML Scripts

- Can use a text editor to type in the HTML tags
  - The most basic way to go
- But there are many HTLM editors
  - All generate HTML text files
  - Dreamweaver is a popular one (not free)
    - Easy to use GUI environment
    - Very powerful
  - Netscape Composer
    - From Netscape Navigator Browser: "File" | "Edit Page"
    - Or for a new page: "File" | "New" | "Composer Page"
  - Microsoft Word
    - "File" | "Save As" | 'Web Page"
      - » Result can be kind of "quirky"
  - Microsoft Internet Explorer
    - "File" | "Edit with …"
  - Microsoft FrontPage

HTML Basic Components

- Tags
  - Inform browser to perform some action (display, format, link to, etc.)
    - E.g., <h1>A Simple Home Page</h1> <! Header 1>
    - Not case sensitive; often nested
- Attributes
  - Provide more information related to the tag
  - Like variables--you give their values
  - E.g., <IMG SRC="myimage.gif">
- URLs
  - Attribute values often specify links to other documents
  - For these, the document's URL is the value of the attribute
  - Example (including an inline graphic image): <IMG> tag:

<IMG SRC="http://____.____.____/____.GIF">
 ^ ^ ^
 | | |
tag attribute value is a URL

Other Document "Information" Tags

- Give other information to Browser
  - Don't affect document content
- Document HTML Tag:
  - <HTML> .. entire document script .. </HTML>
    - Specifies it's an HTML document
- Document HEAD Tag:
  - <HEAD> ... Header Info ... </HEAD>
  - Usually Contains Document Title
- Document TITLE Tag:
  - <TITLE> ... Document Title ... </TITLE>
- Document BODY Tag:
  - <BODY> ... Body of Document ... </BODY>

XHTML Skeleton Script

```
<HTML>
  <HEAD>
    <TITLE>  Doc Title </TITLE>
  </HEAD>
  <BODY>
    .....
    .....
  </BODY>
</HTML>
```

Links and Anchors

- Establish how user is guided through a body of hypertext information
- Use the HREF attribute of the A tag

```
<A HREF="filename or URL"> some-text </A>
        |                  |
        |                  |
        |                  |
```

Link to this place

Make this text an anchor; Will be highlighted & underlined in document

Example of Links & Anchors

<A HREF="xxx.html">click here for xxx </A>

----> Click here for xxx

- Here file xxx.html is on same computer and in same directory as active page
- When user clicks on underlined text in the browser, the file is displayed
- To link to another server--give URL as linked item:

<A HREF="http://www.cs.binghamton.edu/~reckert"> R. Eckert's Home Page </A>

----> R. Eckert's Home Page

  - When user clicks on underlined text, linked page is displayed

More on HTML

- Lots of references and books available
  - Appendices F and G of Deitel Text Book
- Some of many Web Sites:
  - http://www.deitel.com/XHTML
  - http://www.htmlprimer.com/htmlprimer/html-primer
  - http://www.w3schools.com/html/default.asp
  - http://www.davesite.com/webstation/html/
  - http://www.htmlcodetutorial.com/
  - http://www.htmlgoodies.com/primers/html/
  - http://www.echoecho.com/html.htm
  - http://www.2kweb.net/html-tutorial/
- Also see BU's Information Technology Services "Instructional Web Pages"
  - http://computing.binghamton.edu/web/bingweb

MyPage0.html
A Very Simple HTML document

```
<html>
  <head>
    <title> My First Home Page </title>
  </head>
  <body>
    My name is RE and this is my first home page
  </body>
</html>
```

http://cs.binghamton.edu/~reckert/330/mypage0.html

- A more complicated one:

  http://cs.binghamton.edu/~reckert/330/TestPage08.html

What Happens when a URL such as www.whatever.com/fn.html is Typed into a Browser

- Browser uses the DNS to convert the URL into an IP address
- Then opens a socket connection to the server on port 80 and transmits an HTTP request:

  GET /fn.html HTTP/1.1

  8 more lines of message header containing information about the request

  [Blank line]
- Start line: GET is a method requesting the desired resource
- Blank line (CR/LF) marks end of header and end of request

Web Server Response to GET Command

- If fn.html is a valid resource identifier and security settings don't prevent it from being returned:
  - Server transmits an HTTP response like:

HPPT/1.1 200 OK
7 lines of header information
[blank line]
<html>
<body>
Hello, world
</body>
</html>

{This is the HTML of the returned page}

- The browser then parses the returned HTML and displays the Web Page