Introduction, Internet and Web Basics
XHTML and HTML

January 31, 2006
Harvard University
Division of Continuing Education
Extension School

Course Web Site: http://cscie12.dce.harvard.edu/
Copyright 1998-2005 David P. Heitmeyer
My email: david_heitmeyer@harvard.edu
Course staff email: cscie12@fas.harvard.edu

CSCI E-12, Fundamentals of Web Site Development

The Course

Course Syllabus | Course Schedule

I. Content for the Web
   a. Markup (XHTML, HTML)
   b. Images
   c. Style (CSS)
   d. Building a Site
II. "Everything" else
   a. Hypertext Transfer Protocol (HTTP)
   b. The Apache HTTP Server
   c. Keeping a site healthy
   d. Dynamic Sites

The Course Staff

- Myself
- Teaching Fellows
  - Mike Cynn
  - Rei Diaz
  - Sonal Patel
  - Susan Smith

What the Course "Is" and "Is Not"

Is | Is Not
---|---
provides a foundation of many facets of Web development | strictly a design class
learn concepts and rules of markup and styles | vocational training in a specific software package
prepare for extending knowledge into specific area of interest or need | a programming class

Enough about the course, what about you?

- Used a computer?
- Used the Web?
- Edited a Web page?
- Published a Web page?
- Experience with XHTML/HTML?
- Maintained existing Web site?
- Built a new site?
- Experience with programming?
The Internet

(adapted from *The Internet Book, 2nd edition* by Douglas E. Comer)

The World Wide Web

The irony is that in all its various guises -- commerce, research, and surfing -- the Web is already so much a part of our lives that familiarity has clouded our perception of the Web itself.

*Tim Berners-Lee* in *Weaving the Web*

- HyperText Information System
- Cross-Platform
- Distributed
- Dynamic
- Web Browser: Single Interface to many services and gateways.
- Interactive

Rule nature by obeying her laws

- The Web site creator is not in control of the User's environment
- The User is in control of her environment
  - A variety of Platforms
  - A variety of Connection Speeds
  - A variety of Displays
  - A variety of HTTP Clients (i.e. Web browsers)
  - A variety of User Settings
  - A variety of Devices
  - A variety of Users
- Open Standards
- Suggestion: Open Source

Concepts, Technologies, Tools

<table>
<thead>
<tr>
<th>Task</th>
<th>Technology</th>
<th>Specific Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creating Resources (Text)</td>
<td>HTML and XHTML</td>
<td>Text editor, Dreamweaver, GoLive, etc.</td>
</tr>
<tr>
<td>Formatting XHTML and HTML</td>
<td>Cascading Style Sheets (CSS)</td>
<td>Text editor</td>
</tr>
<tr>
<td>Creating Image Resources</td>
<td>GIF, JPEG, PNG</td>
<td>Photoshop, Fireworks, Paint Shop Pro</td>
</tr>
<tr>
<td>Locating Resources</td>
<td>Uniform Resource Identifier (URI), may also see URL and URN</td>
<td></td>
</tr>
<tr>
<td>Retrieving Resources</td>
<td>HyperText Transfer Protocol (HTTP)</td>
<td>HTTP Client (Web Browser), Specific examples are IE, NS, Opera, Mozilla, etc.</td>
</tr>
<tr>
<td>Delivering Content</td>
<td>HyperText Transfer Protocol (HTTP)</td>
<td>HTTP Server (Web Server), Specific examples are Apache HTTP Server, Microsoft IIS</td>
</tr>
<tr>
<td>Interactive Command Line Session</td>
<td>Secure Shell (SSH)</td>
<td>SecureCRT, PuTTY</td>
</tr>
<tr>
<td>Transferring Files</td>
<td>SFTP (Secure File Transfer Protocol) or FTP over SSH</td>
<td>SecureFX, PuTTY</td>
</tr>
</tbody>
</table>

Acronyms, Jargon, etc.

*Webopedia* is an excellent online technical glossary and acronym expander. See for example:
HTTP Clients and HTTP Servers

client-server computing

The interaction between two programs when they communicate across a network. A program at one site sends a request to a program at another site and awaits a response. The requesting program is called a client; the program satisfying the request is called the server. (definition from The Internet Book, 2nd edition by Douglas E. Comer)

Client-Server Architecture from Webopedia, http://www.webopedia.com/

Uniform Resource Identifier (URI)

http://cscie12.dee.harvard.edu/lecture/index.html

- URI, Uniform Resource Identifier
- URL, Uniform Resource Locator
-URN, Uniform Resource Name

For more information, see Naming and Addressing (http://www.w3.org/Addressing/) from the W3C.

Anatomy of a URI

- Scheme
  http://www.harvard.edu/academics/index.html
- Common schemes: http, https, ftp, mailto, file, rtsp
- Host
  http://www.harvard.edu/academics/index.html
- Port
  http://www.harvard.edu:80/academics/index.html
- Path Information
  http://www.fas.harvard.edu/academics/index.html
- Fragment Information
  http://cscie12.dee.harvard.edu/lecture/index.html/#2
HTML and HTTP

- Hypertext Markup Language (HTML and XHTML), the *lingua franca* of the Web.
- Hypertext Transfer Protocol (HTTP)

**XHTML: a simple example**

View it.

```xml
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
  <head>
    <title>My Schools</title>
  </head>
  <body>
    <h1>My Schools</h1>
    <ul>
      <li><a href="http://www.harvard.edu/">Harvard University</a><br/>
          <img src="images/veritas.gif" alt="Harvard Shield" height="84" width="72"/></li>
      <li><a href="http://www.ukans.edu/">University of Kansas</a>
          <img src="images/KUSeal.gif" alt="University of Kansas Seal" height="73" width="72"/></li>
    </ul>
  </body>
</html>
```

(X)HTML Document Structure

- Document Type Declaration
- Head
- Body

**A Tree**

A "tree" structure view of XHTML produced by Amaya, the open source Web editor/browser from the W3C.
Components of XHTML Elements

- Start Tag
- Element Name
- Attributes and Values
- Content
- End Tag

<a href="http://www.harvard.edu/">Harvard</a>

Start Tag
<a href="http://www.harvard.edu/">Harvard</a>

Element Name
<a href="http://www.harvard.edu/">Harvard</a>

Attribute
<a href="http://www.harvard.edu/">Harvard</a>

Attribute Value
<a href="http://www.harvard.edu/">Harvard</a>

Content
<a href="http://www.harvard.edu/">Harvard</a>

End Tag
<a href="http://www.harvard.edu/">Harvard</a>

HTML and XHTML: Content vs. Appearance

HyperText Markup Language
- Primary purpose: content

Cascading Style Sheets (CSS)
- Primary purpose: style and appearance

This is really more of a "separation of concerns" -- let each one do that they do best.

XHTML: a simple example with Cascading Style Sheets

View it.

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
    <title>My Schools</title>
    <link rel="stylesheet" href="simple-style.css" type="text/css"/>
</head>
<body>
    <h1>My Schools</h1>
    <ul>
        <li>
            <a href="http://www.harvard.edu/">Harvard University</a>
            <img src="images/veritas.gif" alt="Harvard Shield" height="84" width="72"/>
        </li>
        <li>
            <a href="http://www.ukans.edu/">University of Kansas</a>
            <img src="images/KUSeal.gif" alt="University of Kansas Seal" height="73" width="72"/>
        </li>
    </ul>
</body>
</html>

And here is the stylesheet:

body {
    margin-left: 10%;
    margin-top: 5em;
    margin-right: 10%;
    background-color: #efefec;
}

h1 {
    font-family: sans-serif;
    color: navy;
}

li {
    margin-top: 0.5em;
}

a {
    text-decoration: none;
}

a:link, a:visited {
    color: blue;
}
a:hover {
    color: white;
    background-color: blue;
}
Well-formed (X)HTML and Valid (X)HTML

- well-formed
- valid

Well-formed

Required for well-formed XML

- elements must be properly nested
- elements must have a start and end tag
- element names case-sensitive
- attribute values must be enclosed in quote marks
- attributes may not be repeated

Good practices for HTML

- Choose lower case conventions for element names and attributes.
- Use double quotes for all attribute values.
- Close tags even if they are optional in HTML.
- Use new lines and indentation (spaces or tabs) for readability.

Valid

Well-formed + Conforms to DTD = Valid
HyperText Markup Language

W3C HyperText Markup Language Home Page

Specifications from the W3C

- **XHTML 1.1**, Module-based XHTML
  - May 2001
- **XHTML 1.0**, a reformulation of HTML 4.0 into XML 1.0
  - January 2000
- **HTML 4.01**
  - December 1999
- **HTML 4.0**
  - December 1997
- **HTML 3.2**
  - January 1997
- **HTML 2.0**
  - November 1995

SGML, XML, HTML, and XHTML

Defining Markup Languages

- Standard Generalized Markup Language (SGML)
- Extensible Markup Language (XML)

Specific Markup Languages

- Hypertext Markup Language (HTML)
- Extensible HyperText Markup Language (XHTML)

Relationships

- SGML
- XML is a simplified subset of SGML
- HTML is an application of SGML
- XHTML is an application of XML

Document Type Declaration and Document Type Definition (DTD)

The **Document Type Declaration** for an XHTML 1.0 strict document is:

```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

A closer look at the components follows:

- **html**
  - The name of the root element.
- **PUBLIC**
  - An indication that the DTD is a public standard. Private standards would use "SYSTEM".
- "-//W3C//DTD XHTML 1.0 Strict//EN"
  - The public identifier for the DTD (Document Type Declaration). This identifier can be used to locate the parser to a local copy of the DTD.
  - **W3C**
    - The owner of the DTD (in this case the W3C, the World Wide Web Consortium).
  - **DTD**
    - The type of document that is referenced (in this case a Document Type Definition, DTD).
  - **XHTML 1.0 Strict**
    - The name of the document that the public identifier references.
  - **EN**
    - The language identifier (in this case, "EN" = English). Note that the language code is case-sensitive.
- "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"
  - The URI at which the parser can locate the referenced Document Type Declaration. If the parser has the DTD available in a local library, the URI is not necessary.
Common Document Type Declarations and DTDs

- XHTML 1.0 Transitional
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

- XHTML 1.0 Strict
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

- HTML 4.01 Frameset
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html401/frameset.dtd">
```

- HTML 4.01 Strict
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Strict//EN" "http://www.w3.org/TR/html401/strict.dtd">
```

- HTML 4.01 Transitional
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html401/loose.dtd">
```

- HTML 4.0 Transitional
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

- HTML 4.0 Strict
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Strict//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

- HTML 4.0 Frameset
  
```xml
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
```

Specification vs. Implementations

Pragmatic Steps: Software Tools

- HTTP Client (Web Browser)
- SSH or Client
- SFTP Client
- Text/HTML Editor or Authoring Package
- Graphics Program
- Validators and Checkers
HTTP Clients

- Mozilla Firefox
  - Web Developer Extension for Firefox
- Netscape Navigator/Communicator
- Microsoft Internet Explorer
- Opera
- Safari (Apple)
- Lynx
- Amaya

SSH Clients and SFTP Clients

- PuTTY, http://www.chiark.greenend.org.uk/~sgtatham/putty/
- OpenSSH, http://www.openssh.org/

Text or HTML Editor

Windows

- Dreamweaver
- Adobe GoLive
- Editpad or Wordpad or Notepad

Macintosh

- BBEdit
  BareBones Software
  http://www.barebones.com/
- Dreamweaver
- Adobe GoLive
- SimpleText

UNIX

- Emacs
  http://www.gnu.org/software/emacs/emacs.html
  (UNIX, Windows, Macintosh)

(X)HTML Well-formedness and Validation Resources

- W3C HTML Validation Service
  http://validator.w3.org/
- HTML Tidy
  http://www.w3.org/People/Raggett/tidy/
  minerva% tidy -help
**HTML Tidy**

http://www.w3.org/People/Raggett/tidy/

```
minerva% tidy -h

Tidy [option...] [file...]
Utility to clean up and pretty print HTML/XHTML/XML
See http://www.w3.org/People/Raggett/tidy/

Options for HTML Tidy for Linux/x86 released on 1st September 2002:

Processing directives
---------------------
  -indent or -i  to indent element content
  -omit or -o    to omit optional end tags
  -wrap <column> to wrap text at the specified <column> (default is 68)
  -upper or -u    to force tags to upper case (default is lower case)
  -clean or -c    to replace FONT, NOBR and CENTER tags by CSS
  -bare or -b     to strip out smart quotes and em dashes, etc.
  -numeric or -n  to output numeric rather than named entities
  -errors or -e   to only show errors
  -quiet or -q    to suppress nonessential output
  -xml            to specify the input is well formed XML
  -asxml          to convert HTML to well formed XHTML
  -ashtml         to convert HTML to well formed XHTML
  -asxhtml        to force XHTML to well formed HTML
  -slides         to burst into slides on H2 elements
  -access <level> to do additional accessibility checks (<level> = 1, 2, 3)

Character encodings
-------------------
  -raw            to output values above 127 without conversion to entities
  -ascii           to use US-ASCII for output, ISO-8859-1 for input
  -latin1          to use ISO-8859-1 for both input and output
  -iso2022         to use ISO-2022 for both input and output
  -utf8            to use UTF-8 for both input and output
  -mac             to use MacRoman for input, US-ASCII for output
  -win1252         to use Windows-1252 for input, US-ASCII for output

File manipulation
-----------------
  -config <file>  to set configuration options from the specified <file>
  -f <file>       to write errors to the specified <file>
  -modify or -m   to modify the original input files

Miscellaneous
-------------
  -version or -v  to show the version of Tidy
  -help, -h or -? to list the command line options
  -help-config    to list all configuration options
  -show-config    to list the current configuration settings

You can also use --blah for any configuration option blah

Input/Output default to stdin/stdout respectively
Single letter options apart from -f may be combined
as in: tidy -f errs.txt -imu foo.html
For further info on HTML see http://www.w3.org/MarkUp
```

**Your Minerva Account**

1. Hostname: minerva.dce.harvard.edu
2. ssh minerva.dce.harvard.edu
3. Username: first letter of first name concatenated with up to the first 7 letters (only count "word" characters) of your last name, all lower-case
4. Password: up to the first 8 characters of your last name, all lower-case

```
minerva% passwd
```

**URI to Filename Mapping**

**User directories**

<table>
<thead>
<tr>
<th>Public_html</th>
<th>jharvard/home/courses/j/h/jharvard</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI</td>
<td><a href="http://minerva.dce.harvard.edu/~jharvard/index.html">http://minerva.dce.harvard.edu/~jharvard/index.html</a></td>
</tr>
<tr>
<td>File</td>
<td>/home/j/h/jharvard/public_html/index.html</td>
</tr>
</tbody>
</table>

**Document Root**

<table>
<thead>
<tr>
<th>Htdocs/WWW/</th>
<th>Htdocs</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI</td>
<td><a href="http://www.fas.harvard.edu/academics/index.html">http://www.fas.harvard.edu/academics/index.html</a></td>
</tr>
<tr>
<td>File</td>
<td>/www/htdocs/academics/index.html</td>
</tr>
</tbody>
</table>
Absolute and Relative Locations

Absolute

Absolute, or fully-qualified, URIs specify the complete information.

Example 1.1

Example 1.1 Source:

```html
<a href="http://cscie12.dce.harvard.edu/lecture/2/notes/slide1.html" >Slide 1</a>
```

Example 1.1 Rendered:

Slide 1

Example 1.2

Example 1.2 Source:

```html
<a href="http://www.whitehouse.gov" >White House</a>
```

Example 1.2 Rendered:

White House

Relative

Relative, or partial, URIs specify partial information. The information not given is assumed from the current location.

Example 1.3

Example 1.3 Source:

```html
<a href="slide1.html" >Slide 1</a>
```

Example 1.3 Rendered:

Slide 1

Location Shortcuts: ../ and ./

- ..
- ./

Example 1.4

Example 1.4 Source:

```html
<a href="../index.html" >Up a level</a>
```

Example 1.4 Rendered:

Up a level

Directory Requests and index.html

Directory without index.html. Note that this assumes the directory permissions are set correctly. If not, you may see Forbidden.

Example 1.5

Example 1.5 Source:

```html
<a href="http://cscie12.dce.harvard.edu/images" >Images from the CSCIE12 Web Site</a>
```

Example 1.5 Rendered:

Images from the CSCIE12 Web Site

Example 1.6

Example 1.6 Source:

```html
<a href="http://www.harvard.edu/academics" >Harvard Academics</a>
```

Example 1.6 Rendered:

Harvard Academics (index.html)
Download/Viewing Source of a Web Page from another site.

- Browser: "View Source" or "Save"
- Minerva Account:
  fas\~ lwp-download http://cscie12.dce.harvard.edu/syllabus.html
- Minerva Account:
  fas\~ lwp-download http://cscie12.dce.harvard.edu/syllabus.html

Online Web Development Resources

Some of my favorites:

- World Wide Web Consortium
  http://www.w3c.org/
- O'Reilly Web and Internet Center
  http://web.oreilly.com/

Table of Contents | All Slides | Link List | Examples | CSCI E-12