

Making Accessibility Part of Your Web Site Management Routine

Rick Ells

Making accessible design part of your Web site management routine is not simple. Habit, practicalities, and even your Web publishing software seem to work against you, making it easy to skip important aspects of accessible design. This paper attempts to outline a framework for including accessible design in your Web publishing routine. Accessible design is the result the interaction of seven design aspects and two management processes. The design aspects are standards, structure, alternative texts, labeling, timing, linearity, and usability. The management processes are site design and page design. Developing a workflow that incorporates the seven aspects into the two processes, dealing with each aspect appropriately within each process, is our objective.

SEVEN DESIGN ASPECTS

The W3C-WCAG guidelines and the Section 508 standards can be broken down into seven general categories, or aspects, of Web design. Each plays an essential role and the combination of all seven ensures that the overall site is accessible.

Standards

HTML, like most Web languages, is strictly defined. At the same time, browsers tend to be forgiving of bad HTML, allowing many Web authors to produce pages that do not comply with the standard. Adhering to the standard usage ensures the compatibility of the page with the widest variety of user agents and with many tools you may want to use to develop and validate your code.

Routine checking against language standards, such as by processing pages with the W3C Validator or with Tidy, will identify errors in logic and coding that would be missed if the page were only checked by viewing it with a browser. Keep in mind that browsers are not designed to point out problems in the HTML since usually the person viewing the page with the browser has no control over the code in the page being viewed.

Structure

HTML is designed around a specific semantic model of a report in the sense that elements in HTML are defined the different parts of a report – headers, paragraphs, lists and list items, blockquotes, etc.

Using the elements to identify logical types of text, rather than to achieve particular visual effects, gives the

whole page a logical structure that client programs can use to help readers navigate and understand the text. The voice browser JAWS, for example, can use headings within a page to help the reader navigate the topics on the page. Of course, the method only works if headers were used to indicate the main page topic (H1), subtopics (H2s), subsubtopics (H3s) as intended.

Alternative Text

The alt attribute provides an text that a reader can hear or see when a graphic cannot be seen or displayed.

Labeling

Forms and tables can be very difficult to navigate by ear. Proper labeling gives the browser information about the relationship between nearby text and a form field. With tables, labeling lets the browser describe each cell as it reads its contents.

Timing

Handicapped persons may be very slow in interacting with a process (such as a person with poor motor skills) or they may be very fast (such as a person using a voice browser turned up to a high scan rate). Measures to indicate that a process is timed or is of a specific duration are necessary to avoid failure.

Linearity

Sighted persons interact with a page as a two-dimension surface, allowing them to correctly interpret a wide range of layouts without difficulty. By definition, a person reading a page with a voice browser will encounter the page contents linearly as a one-dimensional stream of words. The goal of accessible design is to have an approach that gives reasonable freedom for 2D layout, yet is intelligible when read linearly.

One aspect of linearity is the tendency of page designs to put large obstacles, such as a table of contents repeated on every page, between the beginning of each page and its contents. Systematic use of “Skip to Main Content” links at the beginning of the page is one approach to solving this problem.

Usability

Usability relates to whether the page is easily understood by the reader and actually helps the reader reach their goals in coming to the page.

Pages that comply with all other design aspects can have poor usability and therefore be essentially inaccessible. Common problems include complex page designs, inconsistent terminology, inconsistent page designs, scripts that refresh pages without warning, placing key information only in images, and having a long series of items in a list that begin with the same words (“What is...”). Mary Frances Theofanos and Ginny Redish are among the very few researchers who have investigated how Web sites can be both accessible and usable.

Keep in mind too that to be accessible a site must be fully keyboard navigable (usable without using the mouse).

TWO MANAGEMENT PROCESSES

Creating and maintaining a Web site in general consists of two parallel and interrelated processes. Whether you are using simple HTML page, a system of include files, a content management system, or whatever, work must be done to design and set up site-wide processes. Once they are defined, work shifts largely to page level processes consisting of the systematic creation and maintenance of pages, within the framework provided by the site-level setup.

Site Level Process

Setting up a Web site usually begins with a series of steps to define common design elements that will be used across all pages in the site.

- Type of HTML (DOCTYPE statement)
- Page templates, often with variations depending on the role of each type of page
- Site-wide navigation system
- Include files for common page features such as headers and footers and navigation
- Style sheets defining presentation properties and positioning of page elements and divisions
- The types of media that will be used on the site (PDF, Flash, etc.)
- Scripting functions

In general, site level setup focuses on the DOCTYPE, the contents of the HEAD element in each page (including links to stylesheets) and the beginning and end of the contents of the BODY element.

Page Level Process

Pages are created and maintained within the context of the site level process using selected Web publishing tools. While the site setup may be done only once and then only occasionally updated, the page level process will be constantly repeated as new pages are created and old ones updated.

Page level work focuses on specific page content, seldom going outside the BODY element on each page.

MANAGING ACCESSIBILITY

Each of the management processes offers different opportunities to deal with the accessibility aspects.

Site Level Accessibility

Site level setup can address many accessibility issues.

- A standard DOCTYPE statement on all pages will support standards-based coding, assist validation to ensure the page conforms to the chosen standard, and help the client program in interpreting and displaying the pages.
- Controlling site navigation features at the site level will make possible greater consistency across the site.
- Standard page templates can address the “Skip to main content” need in a uniform, efficient way that ensures the implemented solution is present on all pages in the site.
- A standard library of scripting functions (such as for pull down menus) can allow accessibility considerations of the scripts to be addressed efficiently. Any script based navigation should be optional, or at least fully functional from the keyboard only (without using the mouse).
- Usability aspects of navigation and general page layout can be addressed independent of specific page content.

Page Level Accessibility

With page templates controlling page layout, page level accessibility can focus on the following factors:

- Properly structured content (placing text content into logically appropriate elements) creates an easily interpretable content.
- Alternative text, labeling, and timing can be addressed as content is entered, particularly if the Web publishing tool being used is set to prompt for the information on entry
- Linearity within the page content can be addressed with CSS positioning, if the visual layout sequence is different from the linear sequence. On the other hand, since the page content is within the navigation framework provided by the templates, no positioning may be necessary at the page level.
- Timing issues must be addressed at the page level with appropriate explanations. For example, it is now common to set time limits on forms provided for the entry of identity information. Explanatory text should appear at the top of the form stating the time limit.

THE IMPORTANCE OF TOOLS

Most Web publishing tools now have configuration options that will encourage diligence in sticking to accessible design methods. Oddly, most of these products have these features turned off by default.

Configuring Tools

Useful configurations include the following:

- Specify the type of HTML being used. Having your software set to add a DOCTYPE statement specifying XHTML Transitional and to produce compliant HTML code to that standard would be a good approach to sticking to standards. In practice, it does not make much difference whether you pick HTML 4.01 or XHTML and Transitional or Strict, as long as you are willing to follow through and comply with the standard you declare. The combination of the DOCTYPE statement and compliant code will significantly help client programs correctly interpret and present your content.
- Setting software to prompt for alternative texts for all non-text objects, summary attributes for all tables, and labeling within tables will help encourage consistency in taking these important steps.
- Built in validators are extremely helpful in ensuring your code is fully compliant with the logic and syntax of HTML.

Useful Toolsets

Many helpful tools are available for including accessibility in your work routine:

- Tidy – A delightful tool for checking and cleaning HTML, once you get in the habit of using DOCTYPE statements and aiming for fully compliant code.
- AIS Accessibility Toolbar – Available from Australia’s National Information Library Service, the Toolbar is a useful set of tools for understanding how accessible your pages are.
- HTML-KIT – A delightful HTML editor with a nicely integrated version of Tidy.

REFERENCES

(1) Theofanos, Mary Frances, and Janice (Ginny) Redish
“Guidelines for Accessible and Usable Web Sites:
Observing Users Who Work With Screen Readers”
<http://www.redish.net/content/papers/interactions.html>

(2) W3C, “Authoring Tool Accessibility Guidelines 2.0”
<http://www.w3.org/TR/ATAG20/>

(3) AIS Accessibility Toolbar – <http://www.nils.org.au/ais/>

(4) HTML-KIT – <http://www.chami.com/>

Rick Ells
Senior Computer Specialist
University of Washington
Box 352830
Seattle, Washington 98195-2830 USA
206-543-2875

Rick Ells is a member of the Communications Group of Computing & Communications at the University of Washington. Projects include Web design, training, project management, information architecture, and graphical design.

Making Accessibility Part of Your Web Site Management Routine

Rick Ells
University of Washington
Computing & Communications
<http://staff.washington.edu/rells/stc2005/>
rells@cac.washington.edu

Principles

- **Importance of Adherence to Standards:** Adaptive and assistive technology need standards compliant input to work properly.
- **Content Has Structure:** Good logical (semantic) structure aids in proper processing of the content, allowing logical, structured views of the content.
- **Attributes Enable Alternatives:** Page element attributes provide alternative ways to get the content (alt, longdesc, etc.) and provide supplemental information (id, names, etc.).
- **Quality Content Is User Supportive:** Technically correct content that does not meet the user's needs is an obstacle, not a solution.
- **Accessibility Takes in the Whole Site:** A comprehensive approach to accessibility is necessary. Do it part way is not helpful

Resources

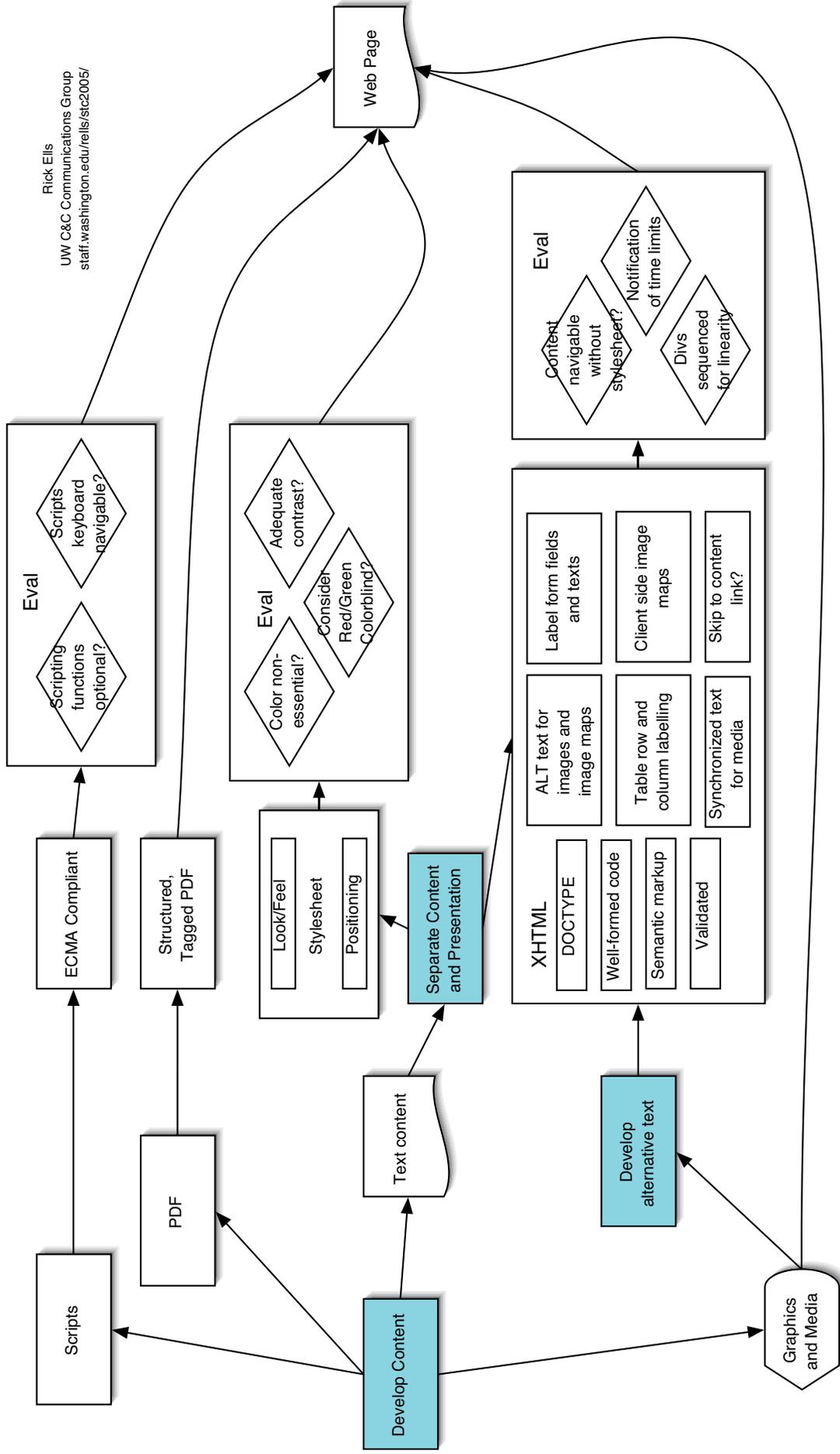
- Boiko, Bob, **Content Management Bible**, Hungry Minds Press
- Musciano, Chuck, and Bill Kennedy, **HTML & XHTML: The Definitive Guide**, O'Reilly Press
- Paciello, Michael, **Web Accessibility for People With Disabilities**, CMP Press
- Rosenfeld, Louis, and Peter Morville, **Information Architecture for the World Wide Web**, O'Reilly Press
- Zeldman, Jeffrey, **Designing With Web Standards**, New Riders Press

- **Creating Accessible Adobe PDF Files** - Adobe
http://www.adobe.com/enterprise/accessibility/pdfs/acro6_pg_ue.pdf
- **IBM Accessibility Center**
<http://www-3.ibm.com/able/guidelines/web/accessweb.html>
- **ECMA International: Standards@Internet Speed**
<http://www.ecma-international.org/>
- **A List Apart** - Jeffrey Zeldman
<http://www.alistapart.com/>
- **Vischeck**
<http://www.vischeck.com/>
- **CSS Zen Garden**
<http://www.csszengarden.com/>
- **W3C Web Accessibility Initiative (WAI)**
<http://www.w3.org/WAI/>
- **Cynthia Says Portal**
<http://cynthia.contentquality.com/>

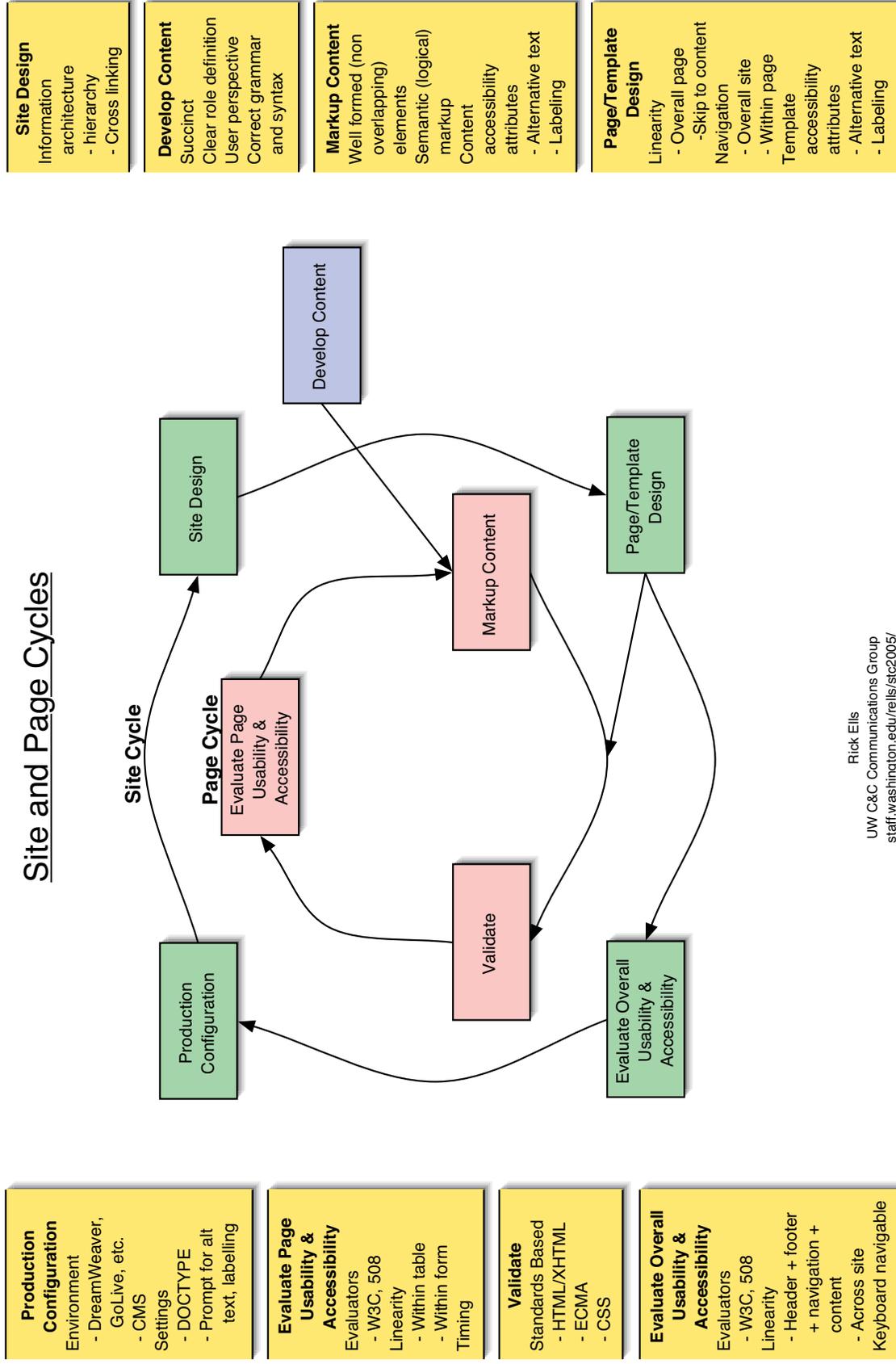
Section 508 Guidelines (1194.22 Standards)

(<http://www.access-board.gov/sec508/guide/1194.22.htm>)

- (a) A text equivalent for every non-text element shall be provided (e.g., via "alt", "longdesc", or in element content).
- (b) Equivalent alternatives for any multimedia presentation shall be synchronized with the presentation.
- (c) Web pages shall be designed so that all information conveyed with color is also available without color, for example from context or markup.
- (d) Documents shall be organized so they are readable without requiring an associated style sheet.
- (e) Redundant text links shall be provided for each active region of a server-side image map.
- (f) Client-side image maps shall be provided instead of server-side image maps except where the regions cannot be defined with an available geometric shape.
- (g) Row and column headers shall be identified for data tables.
- (h) Markup shall be used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers.
- (i) Frames shall be titled with text that facilitates frame identification and navigation.
- (j) Pages shall be designed to avoid causing the screen to flicker with a frequency greater than 2 Hz and lower than 55 Hz.
- (k) A text-only page, with equivalent information or functionality, shall be provided to make a Web site comply with the provisions of this part, when compliance cannot be accomplished in any other way. The content of the text-only page shall be updated whenever the primary page changes.
- (l) When pages utilize scripting languages to display content, or to create interface elements, the information provided by the script shall be identified with functional text that can be read by assistive technology.
- (m) When a Web page requires that an applet, plug-in or other application be present on the client system to interpret page content, the page must provide a link to a plug-in or applet that complies with §1194.21(a) through (l).
- (n) When electronic forms are designed to be completed on-line, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues.
- (o) A method shall be provided that permits users to skip repetitive navigation links.
- (p) When a timed response is required, the user shall be alerted and given sufficient time to indicate more time is required.



Site and Page Cycles



Site Design
 Information architecture
 - hierarchy
 - Cross linking

Develop Content
 Succinct
 Clear role definition
 User perspective
 Correct grammar and syntax

Markup Content
 Well formed (non overlapping) elements
 Semantic (logical) markup
 Content accessibility attributes
 - Alternative text
 - Labeling

Page/Template Design
 Linearity
 - Overall page
 - Skip to content
 Navigation
 - Overall site
 - Within page
 Template accessibility attributes
 - Alternative text
 - Labeling

Production Configuration
 Environment
 - DreamWeaver, GoLive, etc.
 - CMS
 Settings
 - DOCTYPE
 - Prompt for alt text, labelling

Evaluate Page Usability & Accessibility
 Evaluators
 - W3C, 508
 Linearity
 - Within table
 - Within form
 Timing

Validate
 Standards Based
 - HTML/XHTML
 - ECMA
 - CSS

Evaluate Overall Usability & Accessibility
 Evaluators
 - W3C, 508
 Linearity
 - Header + footer + navigation + content
 - Across site
 Keyboard navigable