The Wayback Machine - https://web.archive.org/web/20080730184142/http://www.w3.org:80/TR/2008/REC-mobile-bp-20080729/

Mobile Web Best Practices 1.0

Basic Guidelines

W3C Recommendation 29 July 2008

This version:
http://www.w3.org/TR/2008/REC-mobile-bp-20080729/
Latest version:
http://www.w3.org/TR/mobile-bp/
Previous version:
http://www.w3.org/TR/2006/PR-mobile-bp-20061102/
Editors:
Jo Rabin, mTLD Mobile Top Level Domain (dotMobi)
Charles McCathieNevile, Opera Software [Early Drafts]

Please refer to the **errata** for this document, which may include some normative corrections.

See also **translations**.

Copyright © 2008 W3C® (MIT, ERCIM, Keio), All Rights Reserved. W3C liability, trademark and document use rules apply.

Abstract

This document specifies Best Practices for delivering Web content to mobile devices. The principal objective is to improve the user experience of the Web when accessed from such devices.

The recommendations refer to delivered content and not to the processes by which it is created, nor to the devices or user agents to which it is delivered.

It is primarily directed at creators, maintainers and operators of Web sites. Readers of this document are expected to be familiar with the creation of Web sites, and to have a general familiarity with the technologies involved, such as Web servers and HTTP. Readers are not expected to have a background in mobile-specific technologies.

Status of this Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

This document was developed by the Best Practices Working Group (BPWG) as part of the Mobile Web Initiative.

Please see the Working Group's implementation report. Changes since the previous version of the document are editorial. A complete list of changes made to this document is available. Note the document stayed in Proposed Recommendation for more than a year as it depended on the progress of XHTML Basic 1.1 on the Recommendation track.

Please send comments about this document to public-bpwg-comments@w3.org (with public archive).

This document combines the experience of many mobile Web stakeholders into one set of best practices, regarded as essential by the participants of the Working Group.

This document has been reviewed by W3C Members, by software developers, and by other W3C groups and interested parties, and is endorsed by the Director as a W3C Recommendation. It is a stable document and may be used as reference material or cited from another document. W3C's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and interoperability of the Web.

This document was produced by a group operating under the 5 February 2004 W3C Patent Policy. This document is informative only. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

Twitch Interactive EX1027 Page 1

Table of Contents

1 Introduction
- 1.1 Purpose of the Document
- 1.2 How the Best Practices are Organized
- 1.3 Audience
- 1.4 Scope
  - 1.4.1 Phasing
- 1.5 Relationship to other Best Practices and recommendations
- 1.6 Longevity and Versioning

2 Requirements
- 2.1 Presentation Issues
- 2.2 Input
- 2.3 Bandwidth and Cost
- 2.4 User Goals
- 2.5 Advertising
- 2.6 Device Limitations
- 2.7 Advantages

3 Delivery Context
- 3.1 One Web
- 3.2 Background to Adaptation
- 3.3 Adaptation Implementation Model
- 3.4 Assumptions about Adaptation
- 3.5 Establishing Context
- 3.6 Choice of User Experience
- 3.7 Default Delivery Context

4 Structure of Best Practice Statements

5 Best Practice Statements
- 5.1 Overall Behavior
  - 5.1.1 Thematic Consistency of Resource Identified by a URI
  - 5.1.2 Exploit Device Capabilities
  - 5.1.3 Work around Deficient Implementations
  - 5.1.4 Testing
- 5.2 Navigation and Links
  - 5.2.1 URIs of Site Entry Points
  - 5.2.2 Navigation Bar
  - 5.2.3 Balanced Structure
  - 5.2.4 Navigation Mechanisms
  - 5.2.5 Access Keys
  - 5.2.6 Link Target Identification
  - 5.2.7 Image Maps
  - 5.2.8 Refreshing, Redirection and Spawned Windows
  - 5.2.9 Externally Linked Resources
- 5.3 Page Layout and Content
  - 5.3.1 Page Content
  - 5.3.2 Page Size
  - 5.3.3 Scrolling
  - 5.3.4 Navigation Bars etc. (Extraneous material)
  - 5.3.5 Graphics
  - 5.3.6 Color
  - 5.3.7 Background Images
- 5.4 Page Definition
  - 5.4.1 Title
  - 5.4.2 Frames
  - 5.4.3 Structural Elements
  - 5.4.4 Tables
  - 5.4.5 Non-Text Items
  - 5.4.6 Image Size
  - 5.4.7 Valid Markup
  - 5.4.8 Measures
  - 5.4.9 Style Sheets
  - 5.4.10 Minimize
  - 5.4.11 Content Types
  - 5.4.12 Character Encoding
  - 5.4.13 Error Messages
  - 5.4.14 Cookies
  - 5.4.15 Cache Headers
  - 5.4.16 Fonts
- 5.5 User Input
  - 5.5.1 Input
  - 5.5.2 Tab Order
  - 5.5.3 Labels for Form Controls

6 Conformance and mobileOK

Twitch Interactive EX1027 Page 2

6.1 Classes of Products
6.2 Extensibility

Appendices

A Sources (Non-Normative)
B Related Reading (Non-Normative)
C Acknowledgements (Non-Normative)
D References (Non-Normative)
D.1 MWI References
D.2 Sources
D.3 Device Independence
D.4 Web, Protocols and Languages
D.5 Other References

List of Best Practices

The following Best Practices are discussed in this document and listed here for convenience. There is also a free-standing summary.

1. [THEMATIC_CONSISTENCY] Ensure that content provided by accessing a URI yields a thematically coherent experience when accessed from different devices.
2. [CAPABILITIES] Exploit device capabilities to provide an enhanced user experience.
3. [DEFICIENCIES] Take reasonable steps to work around deficient implementations.
4. [TESTING] Carry out testing on actual devices as well as emulators.
5. [URIS] Keep the URIs of site entry points short.
6. [NAVBAR] Provide only minimal navigation at the top of the page.
7. [BALANCE] Take into account the trade-off between having too many links on a page and asking the user to follow too many links to reach what they are looking for.
8. [NAVIGATION] Provide consistent navigation mechanisms.
9. [ACCESS_KEYS] Assign access keys to links in navigational menus and frequently accessed functionality.
10. [LINK_TARGET_ID] Clearly identify the target of each link.
11. [LINK_TARGET_FORMAT] Note the target file's format unless you know the device supports it.
12. [IMAGE_MAPS] Do not use image maps unless you know the device supports them effectively.
13. [POP_UPS] Do not cause pop-ups or other windows to appear and do not change the current window without informing the user.
14. [AUTO_REFRESH] Do not create periodically auto-refreshing pages, unless you have informed the user and provided a means of stopping it.
15. [REDIRECTION] Do not use markup to redirect pages automatically. Instead, configure the server to perform redirects by means of HTTP 3xx codes.
16. [EXTERNAL_RESOURCES] Keep the number of externally linked resources to a minimum.
17. [SUITABLE] Ensure that content is suitable for use in a mobile context.

18. [CLARITY] Use clear and simple language.
19. [LIMITED] Limit content to what the user has requested.
20. [PAGE_SIZE_USABLE] Divide pages into usable but limited size portions.
21. [PAGE_SIZE_LIMIT] Ensure that the overall size of page is appropriate to the memory limitations of the device.
22. [SCROLLING] Limit scrolling to one direction, unless secondary scrolling cannot be avoided.
23. [CENTRAL_MEANING] Ensure that material that is central to the meaning of the page precedes material that is not.
24. [GRAPHICS_FOR_SPACING] Do not use graphics for spacing.
25. [LARGE_GRAPHICS] Do not use images that cannot be rendered by the device. Avoid large or high resolution images except where critical information would otherwise be lost.
26. [USE_OF_COLOR] Ensure that information conveyed with color is also available without color.
27. [COLOR_CONTRAST] Ensure that foreground and background color combinations provide sufficient contrast.
28. [BACKGROUND_IMAGE_READABILITY] When using background images make sure that content remains readable on the device.
29. [PAGE_TITLE] Provide a short but descriptive page title.
30. [NO_FRAMES] Do not use frames.
31. [STRUCTURE] Use features of the markup language to indicate logical document structure.
32. [TABLES_SUPPORT] Do not use tables unless the device is known to support them.
33. [TABLES_NESTED] Do not use nested tables.
34. [TABLES_LAYOUT] Do not use tables for layout.
35. [TABLES_ALTERNATIVES] Where possible, use an alternative to tabular presentation.
36. [NON-TEXT_ALTERNATIVES] Provide a text equivalent for every non-text element.
37. [OBJECTS_OR_SCRIPT] Do not rely on embedded objects or script.
38. [IMAGES_SPECIFY_SIZE] Specify the size of images in markup, if they have an intrinsic size.
39. [IMAGES_RESIZING] Resize images at the server, if they have an intrinsic size.
40. [VALID_MARKUP] Create documents that validate to published formal grammars.
41. [MEASURES] Do not use pixel measures and do not use absolute units in markup language attribute values and style sheet property values.
42. [STYLE_SHEETS_USE] Use style sheets to control layout and presentation, unless the device is known not to support them.

43. [STYLE_SHEETS_SUPPORT] Organize documents so that if necessary they may be read without style sheets.
44. [STYLE_SHEETS_SIZE] Keep style sheets small.
45. [MINIMIZE] Use terse, efficient markup.
46. [CONTENT_FORMAT_SUPPORT] Send content in a format that is known to be supported by the device.
47. [CONTENT_FORMAT_PREFERRED] Where possible, send content in a preferred format.
48. [CHARACTER_ENCODING_SUPPORT] Ensure that content is encoded using a character encoding that is known to be supported by the device.
49. [CHARACTER_ENCODING_USE] Indicate in the response the character encoding being used.
50. [ERROR_MESSAGES] Provide informative error messages and a means of navigating away from an error message back to useful information.
51. [COOKIES] Do not rely on cookies being available.
52. [CACHING] Provide caching information in HTTP responses.
53. [FONTS] Do not rely on support of font related styling.
54. [MINIMIZE_KEYSTROKES] Keep the number of keystrokes to a minimum.
55. [AVOID_FREE_TEXT] Avoid free text entry where possible.
56. [PROVIDE_DEFAULTS] Provide pre-selected default values where possible.
57. [DEFAULT_INPUT_MODE] Specify a default text entry mode, language and/or input format, if the device is known to support it.
58. [TAB_ORDER] Create a logical order through links, form controls and objects.
59. [CONTROL_LABELLING] Label all form controls appropriately and explicitly associate labels with form controls.
60. [CONTROL_POSITION] Position labels so they lay out properly in relation to the form controls they refer to.

1 Introduction

1.1 Purpose of the Document

This document sets out a series of recommendations designed to improve the user experience of the Web on mobile devices.

The recommendations are offered to creators, maintainers and operators of Web sites and are intended as the basis for assessing conformance to the mobileOK trustmark, which is described in the Mobile Web Best Practices Working Group Charter and is not developed in this document. At the time of writing of this document, documents describing mobileOK and techniques for implementing the Best Practice recommendations are being worked on.

1.2 How the Best Practices are Organized

The document is organized as follows:

1. Introduction. Describes the audience, purpose and scope of the document.
2. Requirements. An illustration of the type of problems that the Best Practices are intended to ameliorate.

3. Delivery Context. Discusses the environment within which mobile access to the Web is realized, with particular reference to adaptation.
4. Overview of Best Practices. A discussion of the organization of the Best Practices, and sources from which they were derived.
5. Best Practices. The statements themselves.
6. Conformance and mobileOK. A brief conformance statement and reference to the mobileOK documentation.
7. Appendices

   Sources

   Related Reading

   Acknowledgements

   References

1.3 Audience

Readers of this document are expected to be familiar with the creation of Web sites, and to have a general familiarity with the technologies involved, such as Web servers and HTTP. Readers are not expected to have a background in mobile-specific technologies.

Our intention is to make it clear to all involved what the Best Practices are, and hence establish a common basis of understanding. As a result of wishing to be clear to those not already involved in the development of mobile friendly content, some of our statements may appear to be obvious or trivial to those with experience in this area.

The document is not targeted solely at developers; others, such as interaction and graphic designers are encouraged to read it.

1.4 Scope

The scope of these Best Practices is laid out in "Scope of Mobile Web Best Practices" [Scope]. In summary, this document refers primarily to the extension of Web browsing onto mobile devices.

The Best Practice recommendations refer to delivered content. While they are clearly relevant to the processes of content creation and rendering on devices, they are not intended to be Best Practices for those activities.

As the goal of the document is to specify Best Practices for delivery to mobile devices, statements that do not have a specific mobile aspect are not included. In particular, many Web Content Accessibility [WCAG] guidelines are general to all forms of Web access and are not repeated here unless they have a specific mobile interpretation. Examples of general good practice which have a specific mobile interpretation include "Error Messages" and "Color".

See **B Related Reading** for information about the related topics of Internationalization, Web Accessibility and Device Independence.

1.4.1 Phasing

As discussed in the Scope document [Scope] there are many aspects to Mobile Web Best Practices. At present, for example, the design and construction of many Web sites and pages make for a poor user experience when they are viewed on a mobile device.

The quality of the user's Web experience via a mobile device depends significantly on the usability of Web sites, of browsers, and of the device itself. Although browser usability and device usability are important (for reading, navigating, and interacting with content), this document focuses primarily on Best Practices for improving site usability.

In future phases other aspects may be considered - e.g. Best Practices as applied to adaptation and devices. Also in future phases the scope of the recommendations may be extended beyond "Traditional Web Browsing" into fields such as multimodal interaction.

1.5 Relationship to other Best Practices and recommendations

These recommendations are in part derived from the Web Content Accessibility Guidelines [WCAG]. As noted above, WCAG guidelines are supplementary to the Mobile Web Best Practices, whose scope is limited to matters that have a specific mobile relevance.

This document builds on some of the concepts described by the Device Independence Working Group (DIWG) in the Device Independence Principles [DIP]. The document discusses device and delivery channel characteristics, which the DIWG has named "Delivery Context" [DCODI]. In addition, the document uses some terminology from DIWG's Glossary of Terms for Device Independence [DIGLOSS].

Twitch Interactive EX1027 Page 6

The BPWG is developing a companion document describing techniques [Techniques] by which the Best Practice statements in this document can be implemented.

1.6 Longevity and Versioning

The Best Practices have been written at a level of generality that allows them to be applicable across a range of markup languages. They have been written with enduring properties of mobile access to the Web in mind. While the factors identified in 3.7 Default Delivery Context, such as screen dimensions, will change over time, it seems likely that the distinguishing features of mobile access such as cost and difficulty of input will remain issues.

This document may be reviewed from time to time. When necessary, an updated version will be released with clear documentation as to the changes that have been introduced.

2 Requirements

This section discusses the requirements of the Mobile Web Best Practice statements in section 5. The statement of requirements is intended to be illustrative rather than exhaustive or complete.

2.1 Presentation Issues

Today, Many Web pages are laid out for presentation on desktop size displays, and exploit capabilities of desktop browsing software.

Accessing such a Web page on a mobile device often results in a poor or unusable experience. Contributing factors include pages not being laid out as intended. Because of the limited screen size and the limited amount of material that is visible to the user, context and overview are lost.

Because of the limited screen size, the subject matter of the page may require considerable scrolling to be visible, especially if the top of the page is occupied by images and navigation links. In these cases the user gets no immediate feedback as to whether their retrieval has resulted in the right content.

It is particularly important in the mobile context to help the user create a mental image of the site. This can be assisted by adopting a consistent style and can be considerably diminished by an uneven style.

2.2 Input

Mobile device input is often difficult when compared with use of a desktop device equipped with a keyboard. Mobile devices often have only a very limited keypad, with small keys, and there is frequently no pointing device.

One of the difficulties of the mobile Web is that URIs are very difficult to type. Lengthy URIs and those that contain a lot of punctuation are particularly difficult to type correctly.

Because of the limitations of screen and input, forms are hard to fill in. This is because navigation between fields may not occur in the expected order and because of the difficulty in typing into the fields.

While many modern devices provide back buttons, some do not, and in some cases, where back functionality exists, users may not know how to invoke it. This means that it is often very hard to recover from errors, broken links and so on.

2.3 Bandwidth and Cost

Mobile networks can be slow compared with fixed data connections and often have a measurably higher latency. This can lead to long retrieval times, especially for lengthy content and for content that requires a lot of navigation between pages.

Mobile data transfer often costs money. The fact that mobile devices frequently support only limited types of content means that a user may follow a link and retrieve information that is unusable on their device.

Even if the content type can be interpreted by their device there is often an issue with the experience not being satisfactory - for example, larger images may only be viewable in small pieces and require considerable scrolling.

Web pages can contain content that the user has not specifically requested - especially advertising and large images. In the mobile world this extra material contributes to poor usability and may add considerably to the cost of the retrieval.

2.4 User Goals

Mobile users typically have different interests to users of fixed or desktop devices. They are likely to have more immediate and goal-directed intentions than desktop Web users. Their intentions are often to find out specific pieces of information that are relevant to their context. An example of such a goal-directed application might be the user requiring specific information about schedules for a journey they are currently undertaking.

Equally, mobile users are typically less interested in lengthy documents or in browsing. The ergonomics of the device are frequently unsuitable for reading lengthy documents, and users will often only access such information from mobile devices as a last resort, because more convenient access is not available.

2.5 Advertising

Developers of commercial Web sites should note that different commercial models are often at work when the Web is accessed from mobile devices as compared with desktop devices. For example, some mechanisms that are commonly used for presentation of advertising material (such as pop-ups, pop-unders and large banners) do not work well on small devices and are therefore contrary to Best Practice recommendations such as [CENTRAL_MEANING], [LARGE_GRAPHICS] and [POP_UPS].

It is not the intention of the MWI to limit or to restrict advertising; rather it is the intention that the user experience of the site as a whole, including advertising, if any, is as effective as possible.

2.6 Device Limitations

As noted above, the restrictions imposed by the keyboard and the screen typically require a different approach to page design than for desktop devices. As detailed in the Scope document [Scope], various other limitations may apply and these have an impact on the usability of the Web from a mobile device.

Mobile browsers often do not support scripting or plug-ins, which means that the range of content that they support is limited. In many cases the user has no choice of browser and upgrading it is not possible.

Some activities associated with rendering Web pages are computationally intensive - for example re-flowing pages, laying out tables, processing unnecessarily long and complex style sheets and handling invalid markup [T-MOB]. Mobile devices typically have quite limited processing power which means that page rendering may take a noticeable time to complete. As well as introducing a noticeable delay, such processing uses more power as does communication with the server.

Many devices have limited memory available for pages and images, and exceeding their memory limitations results in incomplete display and can cause other problems.

2.7 Advantages

In discussing the limitations of mobile devices for delivery of Web content it is easy to lose sight of the fact that they are extremely popular and very common.

This popularity largely stems at present from them being:

- personal
- personalizable
- portable
- connected

and increasingly multi-functional beyond their original purpose of voice communications.

In addition to these factors, the advantages of mobile devices will increasingly include:

- location awareness
- one-handed operation
- always on
- universal alerting device

By way of illustration of some of these factors: the Web can go where you go. You do not have to remember to do something on the Web when you get back to your computer. You can do it immediately, within the context that made you want to use the Web in the first place.

Moreover, with mobile devices appearing in all shapes and forms, and with a growing variety of features like location technology, cameras, voice recognition, touch screens etc, the Web can reach a much wider audience, and at all times in all situations. It has the opportunity to reach into places where wires cannot go, to places previously unthinkable (e.g. providing medical info to mountain rescue scenes) and to accompany everyone as easily as they carry the time on their wristwatches.

Finally, today, many more people have access to mobile devices than access to a desktop computer. This is likely to be very significant in developing countries, where Web-capable mobile devices may play as similar a role in deploying wide-spread Web access as the mobile phone has played for providing "plain old telephone service".

3 Delivery Context

Delivery Context is used with the specific meaning defined in the Device Independence Glossary [DIGLOSS].

3.1 One Web

Twitch Interactive EX1027 Page 8

The recommendations in this document are intended to improve the experience of the Web on mobile devices. While the recommendations are not specifically addressed at the desktop browsing experience, it must be understood that they are made in the context of wishing to work towards "One Web".

As discussed in the Scope document [Scope], *One Web* means making, as far as is reasonable, the same information and services available to users irrespective of the device they are using. However, it does not mean that exactly the same information is available in exactly the same representation across all devices. The context of mobile use, device capability variations, bandwidth issues and mobile network capabilities all affect the representation. Furthermore, some services and information are more suitable for and targeted at particular user contexts (see 5.1.1 Thematic Consistency of Resource Identified by a URI).

Some services have a primarily mobile appeal (location based services, for example). Some have a primarily mobile appeal but have a complementary desktop aspect (for instance for complex configuration tasks). Still others have a primarily desktop appeal but a complementary mobile aspect (possibly for alerting). Finally there will remain some Web applications that have a primarily desktop appeal (lengthy reference material, rich images, for example).

It is likely that application designers and service providers will wish to provide the best possible experience in the context in which their service has the most appeal. However, while services may be most appropriately experienced in one context or another, it is considered best practice to provide as reasonable experience as is possible given device limitations and not to exclude access from any particular class of device, except where this is necessary because of device limitations.

From the perspective of this document this means that services should be available as some variant of HTML over HTTP (see 3.7 Default Delivery Context).

3.2 Background to Adaptation

The widely varying characteristics of mobile devices can make it difficult for a Web site to provide an acceptable user experience across a significant range of devices. For example different devices support different markup features and different screen sizes may demand different sized images. Consequently, it is very common when delivering content to mobile devices to vary the details of the markup, format of images, image sizes, color depths and so on to suit the characteristics of the device in question. The process of altering content to enhance the user experience on particular devices is referred to as *Content Adaptation*.

We do not describe adaptation in detail in this document. For a more detailed description, readers are referred to the Device Independence Principles [DIP].

In addition, the sister group of the Best Practices Working Group, the Device Description Working Group, is currently defining requirements for a repository of mobile device characteristics that are relevant to content adaptation.

3.3 Adaptation Implementation Model

There are a number of different implementation models for content adaptation. On the one hand, adaptation may be quite simple and consist of determining the device type and choosing the most appropriate set of previously prepared content to match the device characteristics. At the other extreme it may be carried out in a completely dynamic way, with content formatted at the time of retrieval, taking into account not only statically determined properties, such as screen dimension, but also dynamically determined properties, such as the temporary attachment of a fully featured keyboard.

Adaptation can be carried out in a number of different points in the delivery of content to the device [DCODI]:

Server Side adaptation implies that the content is delivered by the originating content server or application.

In-Network adaptation is where the content is altered as it passes through one or more network components. Some network operators, for example, compress images before they are passed over the air to the mobile device.

Client Side adaptation consists of the device accepting content and displaying it in an appropriate way for its characteristics.

Whatever the adaptation model at work, the process of adaptation should not diminish accessibility.

3.4 Assumptions about Adaptation

In phase 1 (See 1.4.1 Phasing) it is assumed that content adaptation, if any, is carried out Server Side. Future phases may consider the implications of content adaptation elsewhere, especially the issues concerning the granting of authority to third parties to carry out adaptation, prohibiting adaptation and so on. Later phases may also address multiple adaptation - i.e. the possibility that adaptation can be applied at more than one point and that In-Network adaptation may occur more than once.

It is also assumed that it is possible to create a site that is consistent with the Best Practice recommendations without carrying out adaptation at all. However it is likely that a more sophisticated and enhanced user experience will be achieved if adaptation is used.

3.5 Establishing Context

Providing variations on the user experience that are appropriate in different cases requires the content provider to know a significant amount about the characteristics of the device, the properties of the browser in use and the transparency of the network connection to the device.

Twitch Interactive EX1027 Page 9

For simple sites that present an interface which is similar across a broad range of contexts the need for such information is diminished when compared with a sophisticated site that has an optimized navigation structure, presents different size images or carries out other adaptations to suit the particular delivery context.

There are several methods by which a content provider can discover information about the delivery context, such as CC/PP, UAPROF, CSS Media Queries and various outputs of the Device Independence Working Group. The companion Techniques [Techniques] document describes these methods.

3.6 Choice of User Experience

In the interests of "One Web" (see 3.1 One Web) considerations, the content provider may choose to allow the user to select from broad categories such as mobile or desktop presentation, where these are distinguished in the application. If the presentation option has been determined automatically, the content provider may choose to allow the user to override the automatic determination. Where a choice of presentations is available, it is good practice to record the user's preferences and to allow them to be changed.

Given an appropriate server environment, it is unlikely that the content provider will be unable to find out anything about the delivery context. However this can happen, either because details of the delivery context are not available in sufficient detail or because the server does not provide the ability to inspect and act on the information provided. In this case a "reasonable default experience" should be provided.

The details of the default experience depend upon a number of factors including, but not limited to, the geographic region in which the service is offered and the primary intention of the service (e.g. considering whether the service is primarily desktop focused vs. primarily mobile focused).

3.7 Default Delivery Context

In order to allow content providers to share a consistent view of a default mobile experience the BPWG has defined the Default Delivery Context. This allows providers to create appropriate experiences in the absence of adaptation and provides a baseline experience where adaptation is used. The Default Delivery Context has been determined by the BPWG as being the minimum delivery context specification necessary for a reasonable experience of the Web. It is recognized that devices that do not meet this specification can provide a reasonable experience of other non-Web services.

It is also recognized that this specification is made against the background of demographic, cultural and economic assumptions. Content providers may choose to provide services that demand a different or lower delivery context specification, but should try to provide an experience that exploits the capabilities of the Default Delivery Context in order to provide the best possible experience for that context.

It is stressed that many devices exceed the capabilities defined by the DDC. Content providers are encouraged not to diminish the user experience on those devices by developing only to the DDC specification, and are encouraged to adapt their content, where appropriate, to exploit the capabilities of the actual device.

In summary, the purpose of defining the DDC is to support the following rules:

- If an adaptation process is used, then information that is known about the actual Delivery Context should (see 5.1.2 Exploit Device Capabilities) be used to vary the delivered content to make it more suitable for that specific Delivery Context or to provide an enhanced user experience.
- If the delivered content does not result from an adaptation process - e.g. the content is statically defined as HTML stored in files, or the details of the Delivery Context cannot adequately be determined, then the delivered content should be suitable for the Default Delivery Context and should comply with the Best Practice statements.

The Default Delivery Context is defined as follows:

Usable Screen Width

120 pixels, minimum.

Markup Language Support

XHTML Basic 1.1 [XHTML-Basic] delivered with content type `application/xhtml+xml`.

Character Encoding

UTF-8 [UTF-8].

Image Format Support

JPEG.

GIF 89a.

Maximum Total Page Weight

20 kilobytes.

Twitch Interactive EX1027 Page 10

Colors

256 Colors, minimum.

Style Sheet Support

CSS Level 1 [CSS]. In addition, CSS Level 2 [CSS2] `@media` rule together with the `handheld` and `all` media types (see CSS 2 Media Types).

HTTP

HTTP/1.0 [HTTP1.0] or more recent [HTTP1.1].

Script

No support for client side scripting.

4 Structure of Best Practice Statements

The Heading

The functional area that is addressed by the statements.

The Statements

One or more Best Practice statements, identified in the following way:

[EXAMPLE] This is a Best Practice statement.

What it means

An explanation of the significance of the statements under this heading.

How to do it

A discussion of techniques and some suggestions as to how to implement. The BPWG is creating a separate document describing techniques [Techniques] in more detail.

What to Test

The aspects of the delivered content that an external validator could examine to assess conformance with the Best Practice statements. This section is not present for process related statements.

In this section it is noted whether the statement is *Machine Testable* (Automated testing is possible) or *Human Testable* (Testing requires human assessment). Some Best Practices are partially machine testable, i.e. based on the result of an automated test, some human interaction may be required. In such cases both a Machine Testable and a Human Testable statement are present.

Some Best Practice statements use words such as "minimize" and "avoid" which are intentionally non-prescriptive. This is in order to provide guidance while leaving room to accommodate a wide variety of applications whose requirements cannot be anticipated. It also allows creativity and diversity within the same Best Practice framework. More prescriptive advice can be found in the Techniques document [Techniques].

References

Where appropriate, references to related WCAG points and other immediate references from the preceding text.

5 Best Practice Statements

The Best Practice statements are grouped under the following headings

- 5.1 Overall Behavior
- 5.2 Navigation and Links
- 5.3 Page Layout and Content
- 5.4 Page Definition
- 5.5 User Input

5.1 Overall Behavior

There are some general principles that underlie delivery to mobile devices.

Twitch Interactive EX1027 Page 11

5.1.1 Thematic Consistency of Resource Identified by a URI

[THEMATIC_CONSISTENCY] Ensure that content provided by accessing a URI yields a thematically coherent experience when accessed from different devices.

5.1.1.1 What it means

This is a realization of the One Web (see 3.1 One Web) principle, whereby content should be accessible on a range of devices irrespective of differences in presentation capabilities and access mechanism. Web sites may paginate their content in various ways corresponding to differences in device characteristics; therefore the navigation structure of the site, and possibly its technical realization, may vary according to the device class that is being served. (See also [WebArch] Section 3.5.1).

A bookmark captured on one device should be usable on another, different type of device even if it does not yield exactly the same experience. If the page that was bookmarked is not appropriate for the device that is now using it, an alternative that is suitable should be provided.

URIs may be decorated to provide session or other information. If a URI is decorated with session information that is no longer current, then the user should be directed to a point in the navigation hierarchy that is appropriate to their device, in order to establish appropriate session and other parameters.

5.1.2 Exploit Device Capabilities

[CAPABILITIES] Exploit device capabilities to provide an enhanced user experience.

5.1.2.1 What it means

While encouraging content providers to be sensitive to the needs of the Default Delivery Context, it is not intended that this will result in a diminished experience on more capable devices. Develop sites that target the Default Delivery Context. In addition, where appropriate, use device capabilities to provide a better user experience on more capable devices.

5.1.3 Work around Deficient Implementations

[DEFICIENCIES] Take reasonable steps to work around deficient implementations.

5.1.3.1 What it means

Just as in the desktop world, there are browsers that do not respect the intentions of the content provider. There are differences in interpretation between browsers and there are also deficiencies in implementation. By deficient we mean non-support of mandatory features of a relevant standard or recommendation and other bugs or errors in implementation.

Because the software in mobile devices is frequently embedded in the device, there is no easy way to correct or enhance it once it is in the field. It is a particular challenge to provide work-arounds for these deficiencies and differences in interpretation. It is recognized that content providers may need to violate specific Best Practices in order to support their intentions on devices that exhibit deficiencies in implementation. If a device is not known to have specific limitations then content providers must comply with Best Practices.

Just as it is not the intention to recommend a least common denominator approach, neither is it the intention to recommend avoiding features that exhibit problems on some class of devices.

It is also not the intention to suggest that content providers should restrict their support to certain device types. Content providers should aim to support as wide a range of device types as is practical.

5.1.4 Testing

[TESTING] Carry out testing on actual devices as well as emulators.

5.1.4.1 What it means

Any Web site should be tested in a range of browsers. Mobile browsers often show markedly different characteristics to desktop browsers. As well as assessing a site's suitability for display in reduced format, content providers are encouraged to test that the features they rely on work in actual devices.

Content providers should also test with specific features disabled, such as using text-only modes and with scripting disabled.

5.1.4.2 How to do it

Twitch Interactive EX1027 Page 12

Many manufacturers provide emulators for their device that can provide a convenient preliminary means of testing. However, in practice, many of the emulators behave in a different way to the devices they emulate. Consequently testing should be carried out in as wide a range of real devices and specific software versions as is practical.

5.2 Navigation and Links

Because of the limitations in display and of input mechanisms, the possible absence of a pointing device and other constraints of mobile devices, care should be exercised in defining the structure and the navigation model of a Web site.

5.2.1 URIs of Site Entry Points

[URIS] Keep the URIs of site entry points short.

5.2.1.1 What it means

Typing URIs on mobile devices can be difficult, and it is expected that users will prefer to use alternative methods of obtaining URIs when available - such as following a hyperlink (from an e-mail, SMS or other Web page), WAP Push, 2D bar code, color bar code, RFID tag and Bluetooth. However, typing a URI may in some cases be the only option available. By keeping site entry point URIs short it is possible to reduce the chance of error and provide a more satisfactory user experience.

5.2.1.2 How to do it

When accessing site entry points users should not have to enter a filename as part of the URI. If possible, configure Web sites so that they can be accessed without having to specify a sub-domain as part of the URI.

Example: Instead of requiring users to type

```
"http://www.example.org/index.html"
```

allow

```
"http://example.org"
```

and instead of

```
"http://www.example.org/example.html"
```

allow

```
"http://example.org/example"
```

5.2.2 Navigation Bar

[NAVBAR] Provide only minimal navigation at the top of the page.

5.2.2.1 What it means

Provide basic navigation, which should be placed on the top of the page. Any other secondary navigational element may be placed at the bottom of the page if really needed. It is important the users should be able to see page content once the page has loaded without scrolling (see 5.3.4 Navigation Bars etc. (Extraneous material)).

5.2.2.2 How to do it

Provide the basic links on a single line.

5.2.3 Balanced Structure

[BALANCE] Take into account the trade-off between having too many links on a page and asking the user to follow too many links to reach what they are looking for.

5.2.3.1 What it means

Twitch Interactive EX1027 Page 13

The design should aim to provide a balance between having a large number of navigation links on a page and the need to navigate multiple links to reach content.

Scrolling a page when there are many links on it can be very cumbersome, as the scrolling action on many mobile devices selects each link in turn. On the other hand, each retrieval of a navigation page takes time and adds cost, so the number of links on a page should not be minimized at the expense of adding page retrievals.

Design the service so that frequently accessed information is easily reached with a minimum number of page retrievals. Navigation to less frequently accessed information may take more retrievals as a result. A guideline is that users become frustrated if it takes more than four retrievals to reach their objective. Whether this can be achieved depends on the nature of the site and, in particular, how items in menus group together to provide understandable themes.

5.2.4 Navigation Mechanisms

[NAVIGATION] Provide consistent navigation mechanisms.

5.2.4.1 What it means

Using the same navigation mechanisms across a service helps users orient themselves and allows them to identify navigation mechanisms more easily.

Users of devices that do not have pointing devices have to scroll between hyperlinks using the keypad. Intelligent grouping, perhaps optimized through adaptation according to usage patterns, can assist usability.

5.2.4.2 How to do it

A "drill-down" method, based on major headings, can often provide an effective means of navigation; because of the linearized arrangement of content, small screen size and lack of pointing device, it is often useful to provide a means to jump entire sections of content.

At each target of the drill-down navigation an "up" link should be provided to allow the user to jump up an entire section.

5.2.4.3 References

This relates to WCAG 13.4.

5.2.5 Access Keys

[ACCESS_KEYS] Assign access keys to links in navigational menus and frequently accessed functionality.

5.2.5.1 What it means

Where there is no pointing device, assigning an access key (keyboard short cut) to a link can provide a convenient way for users to access the link and avoid navigating to the link by repeated pressing of the navigation key.

Provide the same access key for links that are repeated across pages such as links to the home page.

5.2.5.2 What to test

Machine Test: Test for the presence of the `accesskey` attribute.

Human Test: Verify the presence of the `accesskey` attribute on links such as the home page.

5.2.5.3 References

This relates to WCAG 9.5.

5.2.6 Link Target Identification

[LINK_TARGET_ID] Clearly identify the target of each link.

[LINK_TARGET_FORMAT] Note the target file's format unless you know the device supports it.

Twitch Interactive EX1027 Page 14

5.2.6.1 What it means

Users of mobile devices may suffer undue delay and cost as a result of following links. It is important to identify where a link leads so users can make an assessment of whether following it will be of interest to them. While it is unlikely that the cost in monetary terms of a particular user following a particular link can be specified, it should be possible to give an idea of the size of the resource (in bytes or in an abstract way, e.g. large file).

Links to content that is in a different format or different language to that of the page the link is on (i.e. content that can only be interpreted by other applications or downloads) should be human signposted, so that users are not lead to download content that their device may not be able to use. However, bear in mind that some devices support the rendering of those formats by other applications once downloaded (e.g. music files). Additionally, users may wish to download content for later transfer to other devices altogether. So even if it is known that the user agent does not support a particular content type, that content should still be made available.

5.2.6.2 How to do it

Use clear, concise, descriptive link text to help users decide whether to follow a link. Identify the implications of following a link if the target is notably large and the user might not anticipate this from the context.

For the Default Delivery Context all formats other than XHTML, GIF and JPG should be noted.

5.2.6.3 What to test

Human Test: Check for proper descriptions (e.g. no use of "Click here").

Machine Test: Check for links to non-HTML formats.

Human Test: If present check whether there is information about the format of the target of the link.

5.2.6.4 References

This relates to WCAG 11.3 and 13.1.

5.2.7 Image Maps

[IMAGE_MAPS] Do not use image maps unless you know the device supports them effectively.

5.2.7.1 What it means

Image maps allow fast navigation providing the requesting device can support the image involved and providing there is a means of navigating the map satisfactorily. Up, down, left, right and enter are available on most mobile devices, even if there is no pointing device. This is usually sufficient to allow navigation of the active regions of client-side image maps where they are defined as geometric shapes.

Many mobile devices lack a pointing device and server-side image maps cannot be used on such devices.

5.2.7.2 How to do it

If only small images can be displayed, break larger images up into smaller sections and deal with them separately.

For the Default Delivery Context, or if a satisfactory image map cannot be displayed, use a list of links with descriptive text instead.

5.2.7.3 What to test

IMAGE_MAPS Machine Test: Send a request to the site with a device that does not support client-side image maps and check the `map` element is not present.

5.2.7.4 References

This relates to WCAG 1.2 and 9.1.

5.2.8 Refreshing, Redirection and Spawned Windows

Twitch Interactive EX1027 Page 15

[POP_UPS] Do not cause pop-ups or other windows to appear and do not change the current window without informing the user.

[AUTO_REFRESH] Do not create periodically auto-refreshing pages, unless you have informed the user and provided a means of stopping it.

[REDIRECTION] Do not use markup to redirect pages automatically. Instead, configure the server to perform redirects by means of HTTP 3xx codes.

5.2.8.1 What it means

Each of these activities is likely to cause the user confusion, or add cost and delay to their interaction.

Some mobile devices use a separate window for input; this section does not refer to such windows.

Many mobile devices cannot support more than one window and consequently, attempting to open one will have unpredictable results.

Auto-refreshing pages are widely recognized as presenting accessibility problems. In a mobile environment they may expose the user to undue cost as a result of such a page being left open or put unnoticed into the background. If an auto-refreshing page is demanded by the application, always provide a means of ceasing the refresh and always inform the user that the page will refresh and may expose them to higher usage costs.

While redirection is a commonly employed mechanism, it must be remembered that redirection usually requires a round-trip to the browser. This adds to delay on slow links; so use a maximum of one redirect per page and limit the number of pages that are redirected.

5.2.8.2 What to test

POP_UPS Machine Test: Look for the `target` attribute on links and if present check to see if it has a value different from `_self`, `_parent` or `_top`.

AUTO_REFRESH Machine Test: Check whether `meta http-equiv="refresh" content="<the same URI>"` is used.

AUTO_REFRESH Human Test: If auto-refresh is used, check that options are provided to stop any page using auto-refresh.

REDIRECTION Machine Test: Check whether `meta http-equiv="refresh" content="<a different URI>"` is used.

5.2.8.3 References

This relates to WCAG 7.4, 7.5 and 10.1.

5.2.9 Externally Linked Resources

[EXTERNAL_RESOURCES] Keep the number of externally linked resources to a minimum.

5.2.9.1 What it means

Each linked resource (images, style sheets and other objects) requires a separate request across the network. This may add significantly to the load time of the page in the mobile context.

5.2.9.2 How to do it

Minimize the number of images on a page and consolidate style information into a single sheet per page (see also 5.4.9 Style Sheets).

5.2.9.3 What to test

Machine Test: Count the number of linked images, style sheets and other linked items.

Human Test: Review whether a similar effect could be obtained using fewer links.

5.3 Page Layout and Content

Twitch Interactive EX1027 Page 16

This section refers to the user's perception of the delivered content. It concentrates on design, the language used in its text and the spatial relationship between constituent components. It does not address the technical aspects of how the delivered content is constructed, which is discussed in 5.4 Page Definition.

5.3.1 Page Content

[SUITABLE] Ensure that content is suitable for use in a mobile context.

[CLARITY] Use clear and simple language.

[LIMITED] Limit content to what the user has requested.

5.3.1.1 What it means

Users in a mobile context are often looking for specific pieces of information, rather than browsing. Content providers should consider the likely context of use of information and, while providing the option to access all information, should offer appropriate information first. See also discussion under 2.4 User Goals and 3.1 One Web.

The general prescription to use clear language is of particular importance for mobile delivery, where brevity and directness are generally more desirable than a discursive style.

Writing content in the traditional journalistic "front loaded" style can assist users determining whether information is of interest to them and allow them to skip it more easily if it is not. Placing distinguishing information at the beginning of headings, paragraphs, lists, etc. can also help the user contextualize when using devices with limited screen area. See also 5.3.4 Navigation Bars etc. (Extraneous material) for a discussion of making sure that the subject matter of the page is near the top.

Mobile users often pay for bandwidth, so offering them content that is extraneous to their needs, especially advertising, costs them time and money and contributes to an unsatisfactory experience. In general, the user's consent should be sought before initiating the download of content.

5.3.1.2 What to test

Human Test: Examine content to determine if, given the subject matter, it is appropriate in a mobile context.

5.3.1.3 References

This relates to WCAG 13.8 and 14.1.

5.3.2 Page Size

[PAGE_SIZE_USABLE] Divide pages into usable but limited size portions.

[PAGE_SIZE_LIMIT] Ensure that the overall size of page is appropriate to the memory limitations of the device.

5.3.2.1 What it means

If pages are too big they may take an unduly long time to load. In addition, mobile devices typically have restrictions on the largest page they can accommodate.

On the other hand, if pages are too short then the user will be required to make multiple requests to read the relevant information. This can lead to an unnecessary delay, since each request typically takes a measurable time to complete.

The balance between pagination and scrolling is partly a matter of taste and partly a matter of necessity. Devices with severe memory restrictions can only have small pages delivered to them. Equally some devices offer a poor scrolling experience and a better page retrieval experience.

Some studies [MF] have been carried out in this area to test for user preferences. Some of these indicate that users prefer scrolling to click-throughs and some indicate the contrary. More research is likely to be needed in this area.

5.3.2.2 How to do it

For the Default Delivery Context assume the limits specified in 3.7 Default Delivery Context.

5.3.2.3 What to test

Twitch Interactive EX1027 Page 17

PAGE_SIZE_USABLE Machine Test: Measure the total size of the markup for a page; check that it does not exceed 10 kilobytes for the Default Delivery Context.

Human Test: Check that the page is still usable (e.g. not cut in the middle of a sentence, just before the end of a section, and so on).

PAGE_SIZE_LIMIT Machine Test: Measure the total size of markup and images for a page; check that it does not go over the allowed size for the device - 20 kilobytes for the Default Delivery Context.

5.3.2.4 References

This relates to WCAG 12.3.

5.3.3 Scrolling

[SCROLLING] Limit scrolling to one direction, unless secondary scrolling cannot be avoided.

5.3.3.1 What it means

The page should lay out so that simple repeated scrolling in the same direction (axis) allows the user to experience all its content. However some content (such as maps and other images) cannot be displayed without secondary scrolling.

If some element on the page requires secondary scrolling it must not cause the remainder of the page to require this. For example, if an object causes subsequent text to lay out with a significant margin to its left, then this text may not be visible once a user has scrolled past the object.

Equally, if the presence of such an object causes text to render beyond the right boundary of the page then the user will be required to scroll to read each line of text.

5.3.3.2 How to do it

If it is not possible to avoid presenting images that are larger than the screen size, then consider providing these images on a separate page with a link back to the main content.

In the Default Delivery Context assume a width of 120 pixels.

5.3.3.3 What to test

SCROLLING Machine Test: Check for `width` attributes and width style properties wider than the screen size - for the Default Delivery Context, 120 pixels.

Human Test: If it is wider than the screen size, check that the use case warrants it (e.g. maps).

Browse URIs within a site with a mobile device and observe that on pages with elements that require secondary scrolling only those elements require it, and the rest of the page requires only primary scrolling.

5.3.4 Navigation Bars etc. (Extraneous material)

[CENTRAL_MEANING] Ensure that material that is central to the meaning of the page precedes material that is not.

5.3.4.1 What it means

Many Web pages are designed with significant navigational and other elements at the top of or to the side of the page (e.g. Menu Bars, Breadcrumb Trails and Search Functions). This provides a convenient and well-understood navigational metaphor on large displays. However, on small displays this can result in the navigation appearing instead of the actual content of the page when the page is first retrieved.

Because it is important for the user to gain an idea of the content of the page on initial view, there should be a minimum amount of clutter preceding this - including navigation, decorative images, advertising and other material that is not central to the user's experience of the page. The user should not have to scroll significantly to find the primary content of the page.

See also 5.3.1 Page Content for a discussion of how writing style can help the user identify meaning.

5.3.4.2 How to do it

Twitch Interactive EX1027 Page 18

Menu selections can be placed away from the top of the page with a simple link to the selection at the top of the page. Alternatively, use meta navigation on top of the page with simple text links to major sections of the Web site.

5.3.4.3 What to test

Human test: Browse URIs within a site with a mobile device and observe that the most important/relevant information is conveyed first.

5.3.4.4 References

This relates to WCAG 13.5.

5.3.5 Graphics

[GRAPHICS_FOR_SPACING] Do not use graphics for spacing.

[LARGE_GRAPHICS] Do not use images that cannot be rendered by the device. Avoid large or high resolution images except where critical information would otherwise be lost.

5.3.5.1 What it means

The popular mechanism of using a 1 pixel graphic for absolute positioning does not work on a variety of screens.

Graphics that are larger than necessary, for example by having a higher resolution than is displayable on the device or by having too many colors, waste bandwidth.

5.3.5.2 What to test

GRAPHICS_FOR_SPACING Machine Test: Check for very small and/or transparent graphics.

LARGE_GRAPHICS Machine Test: Check dimensions of graphics.

5.3.6 Color

[USE_OF_COLOR] Ensure that information conveyed with color is also available without color.

[COLOR_CONTRAST] Ensure that foreground and background color combinations provide sufficient contrast.

5.3.6.1 What it means

Mobile devices often do not have good color contrast and are often used in less-than-ideal lighting conditions. Hence information highlighted in color may not be visible to users. If color is used to indicate a feature then that feature should generally also be indicated in a way that is not color dependent. In particular, do not use blue or purple text, as this may be confused with hyperlinks, especially on devices that do not underline links.

5.3.6.2 What to test

USE_OF_COLOR Human Test: Browse the page in a monochrome environment.

COLOR_CONTRAST Human Test: Browse the page under a strong light parallel to the screen.

Machine Test: There are automatic tools to test color contrast.

5.3.6.3 References

This relates to WCAG 2.1 and 2.2.

5.3.7 Background Images

[BACKGROUND_IMAGE_READABILITY] When using background images make sure that content remains readable on the device.

Twitch Interactive EX1027 Page 19

5.3.7.1 What it means

Images that are used indiscriminately can lead to content that is hard to view, particularly with the limited contrast often found on mobile devices and in the hostile viewing conditions in which mobile devices are frequently used.

Before using background images, consider carefully your objectives for doing so and try to use alternative techniques to achieve similar objectives. If you use a background image ensure that the content is readable with and without the background image for devices that do not support them.

5.3.7.2 What to test

Machine Test: Test for the presence of a background image.

Human Test: Test readability both on devices that support them and devices that do not.

5.4 Page Definition

5.4.1 Title

[PAGE_TITLE] Provide a short but descriptive page title.

5.4.1.1 What it means

Provide a descriptive title for the page to allow easy identification. Keep the title short to reduce page weight, and bear in mind that it may be truncated.

Many mobile browsers do not display the title of a page. Where the title is displayed the available space may be limited.

The device may use the page title as the default label for bookmarks. Again, space may be limited, so use it to help identify the content and not for other purposes.

5.4.1.2 What to test

Machine Test: Test for presence of the `title` element.

Human Test: Test that the title is descriptive of content.

5.4.2 Frames

[NO_FRAMES] Do not use frames.

5.4.2.1 What it means

Many mobile devices do not support frames. In addition, frames are recognized as being generally problematic.

5.4.2.2 What to test

Machine Test: Test for presence of frame related elements - check for `frameset` and `iframe` elements.

5.4.2.3 References

See http://www.w3.org/TR/xframes/#s_intro for a discussion of problems with frames.

5.4.3 Structural Elements

[STRUCTURE] Use features of the markup language to indicate logical document structure.

5.4.3.1 What it means

It is good practice for all but the simplest documents to indicate their structure through headings and sub-headings. Using structural markup, rather than formatting effects, allows easier adaptation of content where it needs to be divided into several pages, as well as potentially facilitating access to the sections of the document that a user is interested in.

Twitch Interactive EX1027 Page 20

Where headings are used they should be used in accordance with the specification, i.e. they should be properly nested according to their level.

Structural markup must not be used solely to create a font effect (see also 5.4.3 Structural Elements).

5.4.3.2 How to do it

Markup languages like HMTL contain many constructs to indicate structure.

5.4.4 Tables

[TABLES_SUPPORT] Do not use tables unless the device is known to support them.

[TABLES_NESTED] Do not use nested tables.

[TABLES_LAYOUT] Do not use tables for layout.

[TABLES_ALTERNATIVES] Where possible, use an alternative to tabular presentation.

5.4.4.1 What it means

Tables do not work well on limited size screens and may result in the user having to scroll horizontally to read them. Putting navigational links into tables may result in the user having both to scroll horizontally and vertically to see possible navigational choices.

5.4.4.2 What to test

TABLES_SUPPORT Machine Test: Send a request to the site with a device that does not support tables and check the `table` element is not present.

Machine Test: Check that there are no nested tables.

TABLES_LAYOUT Machine Test: Check that no column or row in a table is empty or contains only a 1x1 transparent GIF.

Machine Test: If there is a `table` element, check to see whether there is rendered content outside the element. If there is not then it is likely that the table is being used for layout.

5.4.4.3 References

This relates to WCAG 5.1, 5.2, 5.3, 5.5 and 5.6.

5.4.5 Non-Text Items

[NON-TEXT_ALTERNATIVES] Provide a text equivalent for every non-text element.

[OBJECTS_OR_SCRIPT] Do not rely on embedded objects or script.

5.4.5.1 What it means

A *non-text item* is defined by Non-text content in the WAI Glossary [WAIGlossary].

Downloading images to a mobiledevice adds to the time to display an image and the cost of displaying the page. Making the page readable in text-only mode can help the user assess its usefulness before images arrive.

Many mobile devices do not support embedded objects or script and in many cases it is not possible for users to load plug-ins to add support. Content must be designed with this in mind.

Even where a device does support scripting, do not use it unless there is no other way of accomplishing your objectives. Scripting increases power consumption and so decreases battery life.

5.4.5.2 How to do it

Design pages so that they are useful when rendered as text-only. See also 5.1.4 Testing.

Twitch Interactive EX1027 Page 21

Always use features of the markup designed to support alternate rendering such as the `longdesc` and `alt` attributes in XHTML.

Use only features from the markup that are known to be supported by the device in question.

Avoid things like CSS image replacement and pictures of words.

If scripting is used, do not use `onmouse` and `onkey` triggers, use `onclick`.

5.4.5.3 What to test

NON-TEXT_ALTERNATIVES Machine Test: Test for presence of `alt` attribute on images and text content on objects.

Human Test: Check the relevance of the meaning of the content of `alt` attributes.

OBJECTS_OR_SCRIPT Machine Test: Test for the presence of `object` or `script` elements in content delivered to a device that does not support them and if present carry out the Human Test below.

Human Test: If present, test that the user experience is acceptable.

5.4.5.4 References

This relates to WCAG 1.1, 3.1, 6.2, 6.3, 6.5 and 9.2.

5.4.6 Image Size

[IMAGES_SPECIFY_SIZE] Specify the size of images in markup, if they have an intrinsic size.

[IMAGES_RESIZING] Resize images at the server, if they have an intrinsic size.

5.4.6.1 What it means

Images such as bitmaps have an intrinsic size. Telling the browser in advance what the size is avoids it having to re-flow the page when it receives it. Resizing images at the server reduces the amount of data transferred and the amount of processing the device has to carry out to scale the image.

Note that this recommendation contrasts with 5.4.8 Measures, which recommends using relative measures where possible.

5.4.6.2 What to test

IMAGES_SPECIFY_SIZE Machine Test: Test for presence of `width` and `height` attributes on `img` elements.

IMAGES_RESIZING Machine Test: Check `width` and `height` attributes are equal to image dimensions.

5.4.7 Valid Markup

[VALID_MARKUP] Create documents that validate to published formal grammars.

5.4.7.1 What it means

If the page markup is invalid this will result in unpredictable and possibly incomplete presentation.

5.4.7.2 What to test

Machine Test: Validate documents.

5.4.7.3 References

This relates to WCAG 3.2, 11.1 and 11.2.

See http://www.w3.org/QA/Tools/#validators.

5.4.8 Measures

Twitch Interactive EX1027 Page 22

[MEASURES] Do not use pixel measures and do not use absolute units in markup language attribute values and style sheet property values.

5.4.8.1 What it means

Avoiding pixel and absolute measures allows the browser to adapt content to fit the display. An exception to rule is where an image has been specifically sized for a particular display (see 5.4.6 Image Size). In this case references to the image in markup may specify the exact dimensions of the image in pixels, in order to help the browser to flow the page and avoid re-flowing it after the page has been retrieved. Devices may realize the intentions of authors more accurately if margins, borders and padding are specified in pixels.

5.4.8.2 How to do it

Use percentage and relative measures such as `em`, `ex`, `bolder`, `larger` and `thick`.

5.4.8.3 What to test

Machine Test: Send a request to the site with a device that supports relative measures correctly and check the values of `font-size` are not absolute or pixels.

5.4.9 Style Sheets

[STYLE_SHEETS_USE] Use style sheets to control layout and presentation, unless the device is known not to support them.

[STYLE_SHEETS_SUPPORT] Organize documents so that if necessary they may be read without style sheets.

[STYLE_SHEETS_SIZE] Keep style sheets small.

5.4.9.1 What it means

Style information may be contained in an externally linked style sheet or, in HTML, may be contained either in a style element or in a style attribute on specific elements.

Mobile devices offer varying support for style sheets. Some provide full implementations, including caching of external style sheets; some do not cache external style sheets; some do not support the `style` element; some implementations do not support more than one style sheet and some do not support style sheets at all.

If style sheets are turned off or not supported, content will be rendered in document order, so it is important that the content makes sense when read in document order.

5.4.9.2 How to do it

It is preferable to share style information between pages, but if the device does not support caching of style sheets then this approach would result in the same style sheet being retrieved for each page. Consequently, in order of preference: if the device caches style sheets, put style information in a single external style sheet (see also 5.2.9 Externally Linked Resources); if the device supports the `style` element, use it; otherwise use an external style sheet.

Optimize style information so that only styles that are used are included.

When creating style sheets, take advantage of the CSS media types (these may be used both in the CSS `@media` rule and in the `media` attribute of the `link` element) to specify styles that apply to handheld rendering. The CSS Media types that apply are "handheld" and "all". If handheld rendering is not specified, browsers may download other style sheets even if they are identified as applicable to non-handheld rendering

5.4.9.3 What to test

STYLE_SHEETS_USE Machine Test: Send a request to the site with a device that supports CSS and check that style sheets are used and that the page does not use formatting tags (e.g. `font`).

STYLE_SHEETS_SUPPORT Human Test: Disable style sheets and check that the page is still readable.

STYLE_SHEETS_SIZE Machine Test: Check that the elements in a style sheet are used in at least one of the pages that reference it.

5.4.9.4 References

This relates to WCAG 3.3 and 6.1

5.4.10 Minimize

[MINIMIZE] Use terse, efficient markup.

5.4.10.1 What it means

Content which is marked up in languages such as XML can often be made smaller while preserving exactly the same semantics merely by removal of redundant white space (i.e. spaces and new lines).

Marking fonts, colors and other stylistic effects in-line can cause considerably larger page sizes when compared with using logical markup, and use of the HTML `class` attribute for application of style (see also 5.4.9 Style Sheets).

5.4.10.2 How to do it

While it is not intended that authors should create their content in a single line to remove white space altogether, it is suggested that authors should not contribute to page weight by introducing unnecessary white space. Note that "pretty printing" (the formatting of markup with indentation) can generate large amounts of white space and hence add to page weight.

If "pretty printing" is an important part of the authoring process, then try to arrange that redundant white space is stripped when serving a page.

Even though some network proxies strip white space that they think is redundant, not all do so, so it is not best practice to rely upon this behavior.

Use of structural markup (see 5.4.3 Structural Elements) contributes to minimizing the size of the markup on a page, as does centralizing the style descriptions using CSS [CSS].

5.4.10.3 What to test

Machine Test: Count the number of non-significant white space characters in the document.

5.4.11 Content Types

[CONTENT_FORMAT_SUPPORT] Send content in a format that is known to be supported by the device.

[CONTENT_FORMAT_PREFERRED] Where possible, send content in a preferred format.

5.4.11.1 What it means

Transferring content that a device cannot display wastes users' time and money. A device may express a preference for formats. In this case it is good practice to respect the device's preference, as it may have a fuller implementation of those formats.

5.4.11.2 How to do it

To determine what formats a device supports, Web sites may use any combination of device profile information such as the HTTP User-Agent header, HTTP Accept headers and UAProf.

There are problems with using any one approach to the exclusion of the others. Some issues that have been noted by the BPWG in this context are:

- Some devices do not supply accept headers;
- Some devices mis-state their capabilities;
- Some operator gateways supplement the accept headers to include formats that they adapt;
- User agent headers do not always uniquely identify the device;
- UAProf information may not be available or may be incomplete.

5.4.11.3 What to test

Twitch Interactive EX1027 Page 24

CONTENT_FORMAT_SUPPORT Machine Test: Check MIME types of content with various devices.

CONTENT_FORMAT_PREFERRED Machine Test: Check MIME types of content with various devices and check that the preferred format is sent or that the format is compatible with the Default Delivery Context.

5.4.12 Character Encoding

[CHARACTER_ENCODING_SUPPORT] Ensure that content is encoded using a character encoding that is known to be supported by the device.

[CHARACTER_ENCODING_USE] Indicate in the response the character encoding being used.

5.4.12.1 What it means

As in the previous section, content should not be sent to a device if it can not use it.

5.4.12.2 How to do it

The supported character encodings for a device may be obtained either from a device profile or by examining the value of the HTTP Accept-Charset header.

The character encoding being used in a response may be indicated using the HTTP Content-Type header.

```
Example:
Content-Type: text/html; charset=utf-8
```

Additionally for XML [XML] documents the character encoding may be indicated in the encoding declaration, although this will generally be ignored if an HTTP Content-Type header is present.

```
Example:
<?xml version="1.0" encoding="UTF-8"?>
```

Encoding of the content to a desired character encoding is dependent on the authoring tools being used, Web server configuration and the server side scripting technology being used (if any). For a discussion of this see [CHARSET1] and [CHARSET2].

Unicode is a good choice for representing content when served in multiple languages. The amount of bandwidth required to transmit content can vary significantly depending on the character encoding used. Text consisting principally of characters from the Latin alphabet will encode more efficiently in UTF-8, whereas text consisting principally of characters from ideographic scripts will encode more efficiently in UTF-16. When choosing a character encoding, consider the efficiency of the available encodings.

Since the Default Delivery Context specifies use only of UTF-8, all applications should support UTF-8.

5.4.12.3 What to test

Machine Test: Check that the encoding is declared in some way and is supported. The content type may be declared in one or more of the following ways: The Content-Type HTTP header, the XML declaration for XML-based content, the CSS @charset rules for CSS, the Content-Type Meta element for HTML content.

5.4.12.4 References

See [XML] Character Encoding in Entities for a discussion of character encoding in XML documents.

5.4.13 Error Messages

[ERROR_MESSAGES] Provide informative error messages and a means of navigating away from an error message back to useful information.

5.4.13.1 What it means

It is inevitable that, on occasions, a mobile user will not be successful in accessing the content or information they sought. Providing easy navigation away from the error is particularly important in the mobile arena, where browsers may not have an easy-to-find "back" button, where contextualization is frequently difficult and where re-entry of URIs as a means of error recovery is particularly difficult.

It is noted that errors due to networking, connection and some kinds of mistyping of URIs are not within the control of the content provider, which has no way to influence how such errors are presented to the user. However, where errors are within the control of the content provider the user should be provided with clear information regarding the fault they have experienced. This should help them to understand whether the fault was temporary or permanent, whether they should retry the attempt to access the content and how they may be able to escalate the problem.

It should also be possible for the user to escape from the error condition. They should either be able to return to the page they were on prior to the error, or to be able to move onwards to a convenient part of the service from where they can retry or alter the transaction they were attempting.

5.4.13.2 How to do it

It is noted that many Web servers provide a default error page, especially in the event of a request for a non-existent page (404) or an internal error (500). Where possible (see [TOMCAT], [APACHE] and [IIS]), applications should trap all error conditions by overriding the default pages if necessary, and handle them in a user-friendly, and graceful, way.

Error messages should be provided in the same language as the application that was being used. They should be clear and concise, adhering to the same Best Practices as the rest of the application. They should be provided in a format that the device can handle.

The error message should detail whether the issue is likely to be temporary or permanent, whether the user may be able to solve the issue themselves (for example, by changing input data or a handset setting), or whether it is an issue that can be escalated to the content provider or network operator. In the latter case, contact details, such as an SMS address or a support line number, might be appropriate.

The error message should provide one or more of the following navigational constructs:

1. A "back" link to return to the previous page (particularly for devices that do not have an easy to find back button);
2. A "retry" link to attempt the relevant part of the transaction again (note that this may not be equivalent to a page "refresh");
3. A "home" link to allow the user to return to the main part of the application.

The error message can provide an error code to be used for diagnosis of the issue. However, the use of an error code is not a substitute for a human-readable message. While some users may understand "404" to mean "page cannot be found", this must not be assumed to be true for all users.

5.4.13.3 What to test

Enter an extraneous URI, known not to represent an actual resource on the site, and check that a HTTP 404 error response is accompanied by a page whose markup is appropriate for the requesting device, or the default context.

Human Test: Check that the page returned contains an explanation of the error and appropriate corrective actions, without assuming any technical knowledge on the part of the end user.

5.4.14 Cookies

[COOKIES] Do not rely on cookies being available.

5.4.14.1 What it means

Cookies are frequently used to carry out session management, to identify users and to store user preferences. Many mobile devices do not implement cookies or offer only an incomplete implementation. In addition, some gateways strip cookies and others simulate cookies on behalf of mobile devices.

5.4.14.2 How to do it

Test that cookies are supported by the device on its current access path. If they are not supported, use URI decoration for session management, being careful not to exceed the device's maximum length for such strings. Some gateways provide user identification without setting cookies.

5.4.14.3 What to test

Machine Test: Check that an alternative to cookies is used for session management when they are not available.

5.4.15 Cache Headers

Twitch Interactive EX1027 Page 26

[CACHING] Provide caching information in HTTP responses.

5.4.15.1 What it means

Limited bandwidth and high latency can reduce the usability of Web sites on mobile devices. Using caching information effectively can reduce the need to reload data such as style sheets, images and pages, thus improving performance and reducing cost of use. It can also prevent the reuse of content where this is not appropriate, for example content that is adapted for one device should not be re-used by different devices. Devices and network caches are both affected by caching information.

5.4.15.2 How to do it

Set expiry times in a way that is appropriate to your application. Consider using `Cache-Control: public` to allow sharing of pages between devices, `Cache-Control: private` to allow re-use but only by the requesting device and `Cache-Control: nocache` to prevent caching.

The HTTP 1.1 specification [HTTP1.1] and Techniques document [Techniques] contain discussions of caching.

5.4.15.3 What to test

Machine Test: Check for the presence of cache headers on the HTTP response.

5.4.15.4 References

Section 13 Caching in HTTP of [HTTP1.1] discusses caching.

5.4.16 Fonts

[FONTS] Do not rely on support of font related styling.

5.4.16.1 What it means

Mobile devices often have few fonts and limited support for font sizes and effects (bold, italic etc.) As a result of this, the use of font size, face or effect, for example to highlight an answer or a stressed word, may not achieve the desired effect. See also 5.4.3 Structural Elements.

5.4.16.2 How to do it

For the Default Delivery Context do not use font related styling.

5.4.16.3 What to test

Machine Test: Check for the presence of font related styling in an environment that does not support it.

Human Test: If present, ensure that the author's intentions remain clear.

5.5 User Input

This section contains statements relating to user input. This is typically more restrictive on mobile devices than on desktop computers (and often a lot more restrictive). For example, mobile devices may lack pointing devices and often do not have a standard keyboard for text entry.

5.5.1 Input

[MINIMIZE_KEYSTROKES] Keep the number of keystrokes to a minimum.

[AVOID_FREE_TEXT] Avoid free text entry where possible.

[PROVIDE_DEFAULTS] Provide pre-selected default values where possible.

[DEFAULT_INPUT_MODE] Specify a default text entry mode, language and/or input format, if the device is known to support it.

Twitch Interactive EX1027 Page 27

5.5.1.1 What it means

Given the typical input limitations of a mobile device, the interface must as far as possible minimize user input. Where possible, use selection lists, radio buttons and other controls that do not require typing.

Some markup languages allow the specification of an input mode, which is particularly useful in cases where user input is to be restricted, for example to numeric only. It is anticipated that XHTML-Basic [XHTML-Basic] will support this functionality in the future.

5.5.1.2 How to do it

There are a number of techniques available for this, including:

- Where possible use previous entries as defaults.
- Make it possible to select items using navigation keys and/or numeric input.

5.5.1.3 What to test

AVOID_FREE_TEXT Machine Test: Check whether `input type="text"` and `textarea` are used.

Human Test: If one of them is used, check whether it can be replaced by a pre-determined entry.

PROVIDE_DEFAULTS Machine Test: Check if there is a pre-selected value in controls (selected or checked attribute set).

Human Test: If not, check if there could be sensible pre-selection in the context (e.g. most common choice).

DEFAULT_INPUT_MODE Machine Test: Send a request with a device known to support the `inputmode` attribute and if the response is in a language that supports this attribute, check that it is present on `input type="text"` and `textarea` elements.

5.5.1.4 References

This relates to WCAG 10.4.

5.5.2 Tab Order

[TAB_ORDER] Create a logical order through links, form controls and objects.

5.5.2.1 What it means

It is important that as the user navigates through the page the various fields and objects are presented in a logical order, especially as many of them will not be visible at the same time as the focus item.

5.5.2.2 How to do it

Use document order to control layout and tab order.

5.5.2.3 What to test

Machine Test: Check that there are no `tabindex` attributes or layout effects that affect the order of presentation.

If there are `tabindex` attributes check that all controls have a tab index and that they are used consistently.

Human Test: If there are either `tabindex` attributes or layout effects that might affect the order of presentation, then check that the order is usable.

5.5.3 Labels for Form Controls

[CONTROL_LABELLING] Label all form controls appropriately and explicitly associate labels with form controls.

[CONTROL_POSITION] Position labels so they lay out properly in relation to the form controls they refer to.

5.5.3.1 What it means

This means use the `label` element in HTML, or its equivalent in other languages. Make sure that where the label goes is consistent and close to the form control so re-flowing or adapting the content intelligently will always recognize label controls and keep them together.

5.5.3.2 What to test

Machine Test: Check if the `label` element is used in forms.

Human Test: Check whether the labels are properly positioned with regard to the controls.

5.5.3.3 References

This relates to WCAG 10.2 and 12.4.

6 Conformance and mobileOK

The Best Practice statements are intended to be capable of having conformance statements constructed around them in support of the mobileOK trustmark and for other purposes. Work on the mobileOK trustmark will develop specific recommended requirements for a trustmark, which will be based on some profile, or subset, of the Statements in this document.

As such, the mobileOK trustmark will serve as the main conformance claim for the Best Practices document.

All of the Best Practice statements have a fragment identifier to allow formal reference to them and allow the construction of compliance claims that refer to them.

6.1 Classes of Products

This specification applies to one class of product: content delivered to a mobile device, including the metadata transferred as part of the delivery protocol.

6.2 Extensibility

This specification may be compatible with other specifications that describe a different set of requirements for content, insofar as such requirements do not conflict with the current specification.

A Sources (Non-Normative)

The Best Practice statements have been assembled by the BPWG from a number of sources. Primary among those are:

- Various BPWG group meetings and discussions
- WCAG Guidelines 1.0 [WCAG]
- iMode Guidelines [iMode]
- Opera's "Making Small Devices Look Great" [Opera]
- Openwave Guidelines [OpenWave]
- Nokia's Series 60 XHTML-MP Guidelines [Nokia-MP]
- Browsing on Mobile Phones, Nokia [Nokia-VR]
- Little Spring Design [LSD]

While the Best Practice statements have mainly been assembled by secondary research, the sources for that research have in many cases been assembled from primary research. In addition, group members' contributions are to some extent informed by primary research carried out by their company.

B Related Reading (Non-Normative)

Readers interested in the topic of this document will find a variety of other publications of interest. As noted in the Scope paragraph above, topics such as internationalization and accessibility have been addressed separately by the W3C and have not been covered here.

The Character Model for the World Wide Web and other materials prepared by the W3C Internationalization (i18n) Activity cover important interoperability drivers for content prepared for the One Web and the mobile-services arena.

The Web Accessibility Initiative has prepared a variety of Guidelines and Techniques that likewise bear on the preparation and processing of content in and for the Web.

Twitch Interactive EX1027 Page 29

Section 3.2 Background to Adaptation above introduced the idea of content adaptation. Readers who contemplate implementing server-side adaptation will be interested in the ongoing work of the Device Independence Activity.

C Acknowledgements (Non-Normative)

The editors would like to thank members of the BPWG for contributions of various kinds. The editors would also like to thank contributors to the public list, and contributors of Last Call comments whose comments have been taken into account in the creation of this document.

The editors acknowledge significant written contributions from:

- Greg Aaron, Afilias Limited
- Daniel Appelquist, Vodafone (*Mobile Web Best Practices Working Group Chair*)
- Phil Archer, ICRA
- Rotan Hanrahan, MobileAware
- Dominique Hazaël-Massieux, W3C/ERCIM (*W3C Team Contact*)
- Philipp Hoschka, W3C/ERCIM (*W3C Team Contact*)
- Richard T. Kennedy, The Boeing Company
- Rhys Lewis, Volantis Systems Ltd
- Luca Passani, Openwave Systems Inc.
- Giles Payne, NTT DoCoMo
- James Pearce, Argo Interactive Ltd
- Kai-Dietrich Scheppe, Deutsche Telekom AG, T-Com, Geschäftsbereich T-Online
- Andrea Trasatti, W3C Invited Expert
- Paul Walsh, Segala M Test

D References (Non-Normative)

D.1 MWI References

Scope
Scope of Mobile Web Best Practices, Phil Archer, Ed Mitukiewicz, Editors, W3C Working Group Note, 20 December 2005 (See http://www.w3.org/TR/2005/NOTE-mobile-bp-scope-20051220/)

Techniques
Mobile Web Techniques for Best Practices [in development] (See http://www.w3.org/2005/MWI/BPWG/techs/TechniquesIntro)

mobileOK
mobileOK Basic Tests 1.0, Sean Owen, Jo Rabin (eds), W3C Working Draft, 10 June 2008 (See http://www.w3.org/TR/2008/WD-mobileOK-basic10-tests-20080610/)

D.2 Sources

WCAG
Web Content Accessibility Guidelines 1.0, W Chisholm, I. Jacobs, G Vanderheiden, Editors, W3C Recommendation, 5 May 1999. (See http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/)

WAIGlossary
WAI (Printable) Glossary, Katie Haritos-Shea, Charles McCathieNevile, Internal Working Draft, 1 March 2003 (See http://www.w3.org/WAI/GL/Glossary/)

iMode
iMode (See http://www.iMode.nl/pdf/download/How_to_create_an_i-mode_site_1_3.pdf)

Opera
Opera's Making Small Devices Look Great (See http://my.opera.com/community/dev/device/)

OpenWave
Openwave (See http://developer.openwave.com/dvl/support/documentation/guides_and_references/best_practices_in_xhtml_design/index.htm)

Nokia-MP
Nokia Guidelines for XHTML-MP on Series 60 (See http://sw.nokia.com/id/4f7b6805-47d7-4914-885c-6ef2b487adf6/Series_60_Platform_Designing_XHTML_MP_Content_v1_4_en.pdf)

Nokia-VR
Browsing on Mobile Phones, paper by Virpi Roto, Nokia (See http://www.research.att.com/~rjana/WF12_Paper1.pdf)

LSD
Little Spring Design (See http://patterns.littlespringsdesign.com)

D.3 Device Independence

DIP
Device Independence Principles, R. Gimson, Editor, W3C Working Group Note, 1 September 2003 (See http://www.w3.org/TR/2003/NOTE-di-princ-20030901/)

DCODI
Delivery Context Overview for Device Independence, , R. Gimson, R. Lewis, S. Sathish, Editors, W3C Working Group Note, 20 March 2006 (See http://www.w3.org/TR/2006/NOTE-di-dco-20060320/)

DIGLOSS

Glossary of Terms for Device Independence, R. Lewis, Editor, W3C Working Draft (work in progress), 18 January 2005 (See http://www.w3.org/TR/2005/WD-di-gloss-20050118/)

D.4 Web, Protocols and Languages

WebArch
Architecture of the World Wide Web, Volume One, N. Walsh, I. Jacobs, Editors, W3C Recommendation, 15 December 2004 (See http://www.w3.org/TR/2004/REC-webarch-20041215/)

XML
Extensible Markup Language (XML) 1.0 (Fourth Edition), Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, François Yergeau, Editors, W3C Recommendation 16 August 2006, edited in place 29 September 2006 (See http://www.w3.org/TR/2006/REC-xml-20060816)

XHTML-Basic
XHTML™ Basic 1.1, Shane McCarron, Masayasu Ishikawa, Editors, W3C Recommendation, 29 July 2008 (See http://www.w3.org/TR/2008/REC-xhtml-basic-20080729/)

CSS
Cascading Style Sheets (CSS1) Level 1 Specification, Håkon Wium Lie, Bert Bos, Editors, W3C Recommendation, 11 Jan 1999 (See http://www.w3.org/TR/1999/REC-CSS1-19990111)

CSS2
Cascading Style Sheets, level 2 CSS2 Specification, Bert Bos, Håkon Wium Lie, Chris Lilley, Ian Jacobs, Editors, W3C Recommendation, 12 May 1998 (See http://www.w3.org/TR/1998/REC-CSS2-19980512/)

HTTP1.0
Hypertext Transfer Protocol -- HTTP/1.0 Request for Comments: 1945, T. Berners-Lee, R. Fielding, H. Frystyk, May 1996 (See http://www.w3.org/Protocols/rfc1945/rfc1945)

HTTP1.1
Hypertext Transfer Protocol -- HTTP/1.1 Request for Comments: 2616, R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999 (See http://www.w3.org/Protocols/rfc2616/rfc2616.html)

UTF-8
UTF-8, a transformation format of ISO 10646 Request for Comments: 3629, F. Yergeau, November 2003 (See http://www.ietf.org/rfc/rfc3629.txt)

CHARSET1
Tutorial: Character sets & encodings in XHTML, HTML and CSS (See http://www.w3.org/International/tutorials/tutorial-char-enc/)

CHARSET2
FAQ: Setting encoding in Web authoring applications (See http://www.w3.org/International/questions/qa-setting-encoding-in-applications)

D.5 Other References

UAPROF
Open Mobile Alliance OMA-TS-UAProf-V2_0-20060206-A User Agent Profile Approved Version 2.0 – 06 Feb 2006 (See http://www.openmobilealliance.org/technical/release_program/docs/UAProf/V2_0-20060206-A/OMA-TS-UAProf-V2_0-20060206-A.pdf)

TOMCAT
Tomcat FAQ How do I get a customized error page? (See http://tomcat.apache.org/faq/misc.html#q6)

APACHE
Apache Core Features ErrorDocument directive (See http://httpd.apache.org/docs/1.3/mod/core.html#errordocument)

IIS
IIS Custom HTTP Error Messages (See http://msdn.microsoft.com/library/default.asp?url=/library/en-us/iissdk/html/ee7a8c53-f9bc-4cd4-b954-e32066105cf1.asp)

T-MOB
T-Mobile International - Position Statement for W3C Mobile Web Initiative Version: 1.0 18 Dec 2004 (See http://www.w3.org/2004/10/MWIWS-papers/W3C_Mobile_Web_Initiative_-_T-Mobile_Position_Statement-final.pdf)

MF
Study by Segala M Test on Scrolling vs. Pagination (See http://www.mobilefriendly.org)

Twitch Interactive EX1027 Page 31