DCPAPERS

Metadata Generation and Accessibility Auditing

Liddy Nevile
La Trobe University, Australia
Tel. 61-4-1931-2902
Mail mailto:liddy@SunriseResearch.org

https://doi.org/10.23106/dcmi.952107936

Abstract: This paper describes using software to manage and assist in the process of developing a database of metadata. It can be used easily for a number of purposes, including managing digital content's accessibility. In this case a typical university site was audited in 2004 and a lot of metadata was generated.

Keywords: digital library, metadata applications, accessibility, metadata creation, generation

1. Introduction

Many organisations with large digital collections and Web sites are keen to determine and manage the accessibility of their Web content. This is not a simple matter given that at least half of the testing process must be undertaken 'manually' and that many 'pages' have their content changed frequently. This paper describes a process for using software to manage and assist in the process of developing a database of metadata. The resulting, extensive collection of metadata can be used for a number of purposes, including managing the content's accessibility.

A typical university site was audited in 2004. Accessibility is tested for two reasons: to determine compliance with requirements and to assist in the process of increasing accessibility. Whichever, the metadata is then available for other purposes.

2. Audit Scope and Style

2.1.Identifying Players

The first thing when doing judgmental work, as auditing can be, is to obtain permission from someone with authority to do all that is required. The second, also important, is to gain the confidence and support of the people who own the material to be tested. Auditing of sites is often initiated by someone with high-level responsibility within an organisation but gaining access to the content of the sites is not always easy, and the cooperation and authority of someone who can engage with the actual content is essential. Although it may be possible to discover and work on content without the involvement of actual content owners, it is always better and easier if they are available to explain both their understanding of the content and where it is. That the content owners have different understandings from those gained by users, may be of significance.

2.2.Identifying Standards

Next, it must be determined exactly what standards are to be applied. In the US, typically, there are external standards and internal standards to which organisations want to conform. In the US it is usually what are known as Section 508 standards (i) that are relevant. When contracts involving the US Federal Government, or funding provided by them, are connected with the Web content in some way. In other countries, there may be statutory standards, such as in Australia where all public content is covered by anti-discriminatory legislation. There, the content is expected to be available to all without discrimination, and the regulations point to the W3C Web Content Accessibility Guidelines (ii). Currently, Level AA compliance of WCAG is the standard to which public content should conform. In Australia, unlike in the US, legal compliance is not determined by satisfying the specifications of the WCAG but it is a measure to be taken into account when a judge is determining a case of discrimination. So organisations are likely to want to know how their content matches the legal standards. Such differences in standards and their use are common throughout the world.

In addition, many organisations have local standards to which they expect their content developers to conform. It may be that special tests are developed for inclusion in the auditing process, and compliance with these local standards is

https://doi.org/10.23106/dcmi.952107936

audited. Alternatively, it may be that the organisation is interested in how well its local standards contribute to ensuring that the organisation's content matches the external standards to which it may be subject.

2.3.Defining Compliance

In the case study, the team opted for compliance with Australian standards. We interpreted Australian standards as W3C/WAI WCAG 1.0, Level AA for Web content and W3C/WAI ATAG 1.0 for authoring tools (iii). Within this choice, there are objective and subjective tests that must be applied.

2.4.Mapping the Content

Once the standards are determined, the next step is to determine the scope of the audit. Content may be organised in such a way that this is a trivial matter but often content is distributed across a number of locations and it is necessary to make decisions about what is to be in the audit and what is not.

Generating a site map is one way of starting this process. We chose to try Mercury Interactive's Astra Site-Management software (iv) to develop the site map. This software was available for free once the user had given full details of themselves and their workplace. After downloading the software, the user is given seven days before they must register it.

From the main starting page, SiteManager worked its way to 48,084 URIs. 14,432 were available (the http server returned them) and 32,826 were 'unread', probably because they were unprocessed files, such as images etc., or because they were off-site files or had broken links. 2 files were unavailable, maybe because of server problems; 174 URIs had 'access denied' responses, and there were 650 404 errors (broken links). SiteManager found 37,919 local links (URLs) and 10,165 external links. A fast connection was used and the process took 17 minutes. In this time, SiteManager generated a comprehensive report.

From this result, it is obvious that there is a lot to be gained from the exercise. This was only a small section of the university's site, and without the site map it would have been very difficult to gain an insight into the scope of the audit. SiteManager produced a number of useful tables and images that made it easy to see at a glance that many of the problems were in fact only single problems, but linked to from many places that made them show up as more serious and numerous than they really were.

https://doi.org/10.23106/dcmi.952107936

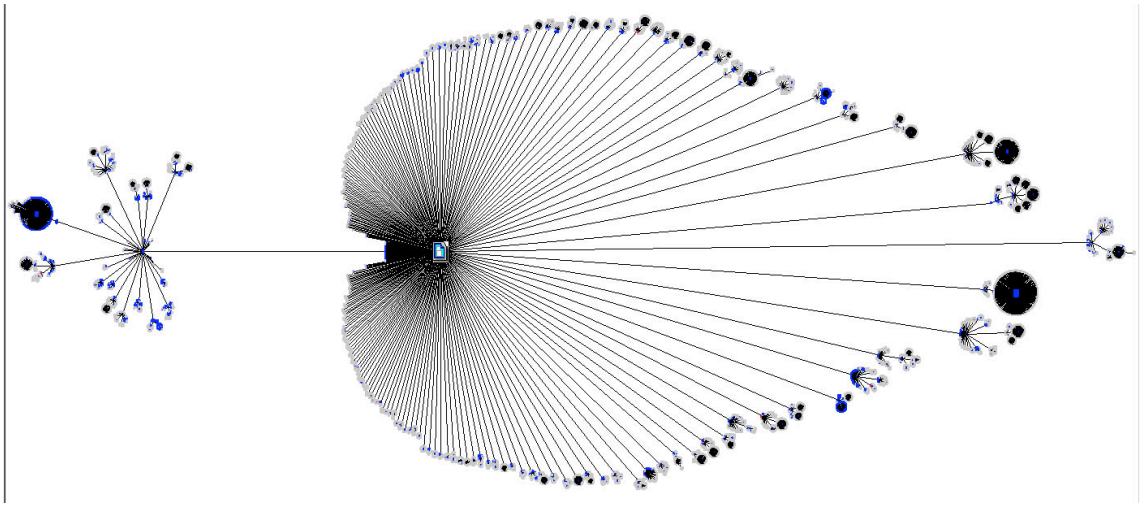


Figure 1. A site map image.

Figure 1 shows the many pages of the section of the site being audited and relationships between those pages. This is important because it can be used to identify trouble spots with greater precision, where traffic is concentrated, etc. Figure 2 shows one section of the relevant part of the site in greater detail.

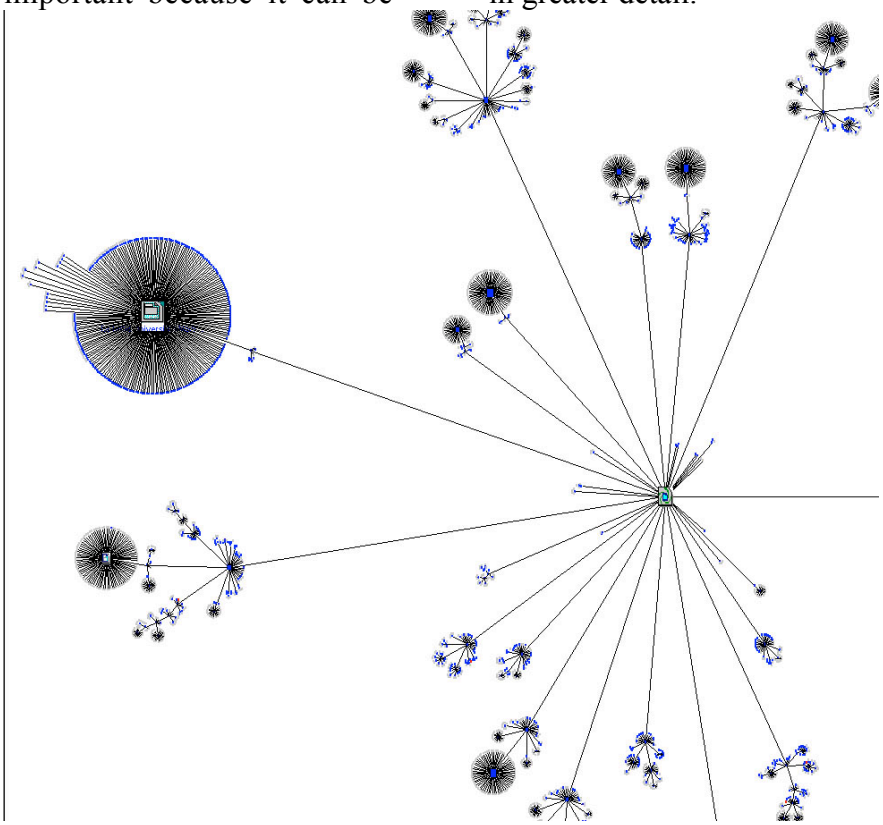


Figure 2. Detail of one section of the site map.

https://doi.org/10.23106/dcmi.952107936

The site information was used to identify specific pages or ranges of pages to be set up for auditing by AccVerify, the chosen testing software (v). The next task, then, was to extract the necessary information from SiteManager into a file that could be 'fed' to AccVerify to direct the process of evaluation. In fact, SiteManager has its own file format and so this process requires several steps, during which it turned out to be useful to have the data in a spreadsheet for easy viewing and manipulation. It was also useful to have the data in a database for bulk handling with respect to some information of interest. From this phase, the information could be considered and made available for other purposes. Once a file of URIs for testing was established, the file was saved as text for use by AccVerify.

The following information discovered by SiteManager was considered to be of interest in the case study: FileName, PageName, Annotation, URL, Last Modified, File Size, Load Size, Incoming Links, Outgoing Links, Broken Links. All AccVerify needed was a list of URIs but some of the other information was useful in deciding which pages should be evaluated. The URIs were not printed on the report page but they lie behind the links for each page, so were easily extracted from the page encoding.

2.5. Setting up AccVerify for Audit Content

Next, AccVerify had to be set up for the evaluation process. The parameters of particular interest were: the standards against which the evaluations were to be made, the type of report to be generated and the format for it,

AccVerify could be set to work immediately or scheduled to work at some future time.

3. Testing Content

Finally, the software was scheduled to undertake the testing task. AccVerify generates reports that can be used in a variety of ways. It provides for automatic and manual evaluation of content. Such questions as:

- does the content contain an image can be answered in such a way as to identify if there is a further need to test for the existence of an ALT tag;
- if there is an ALT tag, it can automatically be tested to see if it has just a typical default value, such as "insert ALT text here" but it requires a human to determine if it is a meaningful ALT tag.

AccVerify does what automated testing it can do and very often this is sufficient to discover that the content does not satisfy all the criteria. On the other hand, even if the automated tests are passed, it is not necessarily true that all accessibility criteria have been met. Almost always there will be aspects that need to be tested manually.

The list of items to be tested by a standard AccVerify test numbers in the sixties. For accessibility purposes, it is important to know not just the format of content, but the genre, for

https://doi.org/10.23106/dcmi.952107936

example. It is possible to have content in text format in the sense that it is meant to be read, but be displayed by an image of writing. Such content would be described as of ‘genre’ text but of ‘format’ image./ This would render the text inaccessible to a device such as a screen reader. So the relationship between genre and format is as important as each of them individually.

Unfortunately, many people do not realise this, and there are many invalid claims of compliance: they have not had the required human attention. Tools such as Bobby (vi) include instructions at the bottom of the reports explaining that only that part of the test that can be automated has been shown compliant by the tool and that more is required if compliance is to be asserted. The AccVerify warning note says: "Files Requiring Visual Verification are files containing HTML elements, identified through automated checks, that require visual verification to determine accessibility. All files should be verified visually, in addition to any automated remediation, to ensure compliance to all checkpoints."

https://doi.org/10.23106/dcmi.952107936

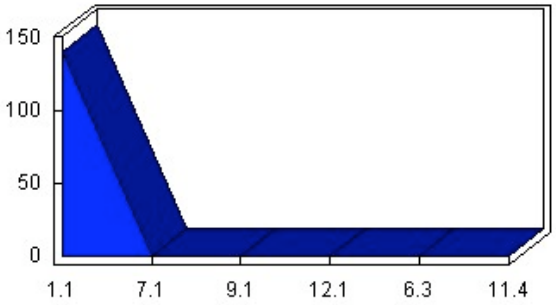
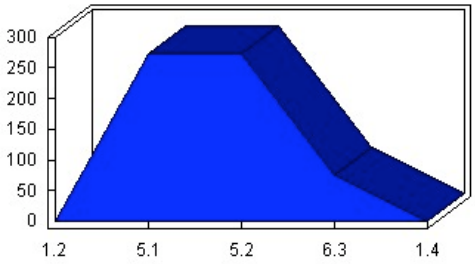
Testing Results	
Date and Time: 1/12/2003 10:45:55 AM Total Files Reported: 75 Total Files Passed: 0 Total Files Failed: 75 View Accessibility Statistics Summary Percentage Passed: 0.0 % Percentage Failed: 100.0 %	
Error Checkpoint Summary (Priority 1) Checkpoint 1.1 / (a): 140 Checkpoint 7.1 / (j): 0 Checkpoint 9.1 / (f): 0 Checkpoint 12.1 / (i): 0 Checkpoint 6.3 / (l),(m): 0 Checkpoint 11.4 / (k): 0	
Visual Checkpoint Summary (Priority 1) Checkpoint 1.2 / (e): 0 Checkpoint 5.1 / (g): 272 Checkpoint 5.2 / (h): 272 Checkpoint 6.3 / (l),(m): 74 Checkpoint 1.4 / (b): 0	
Visual Verification Summary (Priority 1) Total Files Requiring Visual Verification: 74 Total Files Not Requiring Visual Verification: 1 Percentage Requiring Visual Verification: 98.666% Percentage Not Requiring Visual Verification: 1.334%	

Figure 3. Sample of Test results.

3.1.Interpreting the Evaluation

In the case study, about 100 pages were selected for careful testing and none passed the automated test. This was not necessarily a 'bad' result. It meant that not everything in that test was satisfied but it does not mean that the content was not close to satisfactory. Fine-granularity of test results is important for this. So while

https://doi.org/10.23106/dcmi.952107936

the gross evaluation result was interesting, it was the finer detail that was of real significance.

Having seen how many times a single object was included in the range of Web pages, what mattered was how easily those single objects that contained errors could be repaired. That is, if there is, say, a navigation bar that contains some inaccessible content, it may be the single pollutant in a huge number of pages, and so fixing it might fix all the pages that include it.

In a summary of AccVerify's evaluation of a set of 75 pages, the following information was provided and in some cases linked through to reports that offer more detail (linked sections are underlined).

The information available in the linked 'Accessibility Statistics Summary' was useful to help identify the errors by type, making it clear what sort of repairs were going to be required. In many cases, as can be seen, fixing a single type of problem would fix many pages.

3.2.Accessibility Statistics Summary

Image Summary Images: 2458 Images without Alt attribute: 140 Images with Alt attribute: 2318 Images with blank Alt attribute: 0 Images with null Alt attribute: 2094 *Image Counts by file extension:* .gif = 2444 .jpg = 14	**Form Summary** Forms: 75 Forms with Labeled Controls: 0 Forms with Inputs using the Alt attribute: 0 Forms without Input elements: 0 Forms with Input Images not using the Alt attribute: 0 Forms without any use of TabIndex: 75 Forms without any use of AccessKey: 75
Table Summary Tables: 857 Tables with Summary attribute: 0 Tables with Caption: 0 Tables with Summary and Caption: 0 Tables with ID attribute: 0 Tables with Cells with ID attribute: 0 Tables with Cells that use Scope: 0 Data Tables: 272	**Frame Summary** Frames: 0 Frames without Title Attribute: 0 IFrames: 0 IFrames without element content: 0
Object Summary Objects: 0 Objects without element content: 0	**Script Summary** Script Elements: 75 Pages using Script Elements: 74 Pages using Script Elements without NOSCRIPT: 74
Applet Summary Applets: 0 Applets without either the Alt attribute or element content: 0	**Link Summary** Links with the phrase "Click Here" in the link text: 0 Links to DOC files: 0 Links to MP3 files: 0

https://doi.org/10.23106/dcmi.952107936

	Links to MPG files: 0 Links to PDF files: 0 Links to PPT files: 0 Links to SWF files: 0 Links to XLS files: 0

Figure 4. Accessibility statistics summary.

The links from the Site Evaluation Summary page to individual pages provided the following information for each page:

- Verified File: http://www.random-university.edu.au/handbook/index.htm
- Verification Date: 1/12/2003 10:43:03 AM
- Report: CheckLists/www.random-university.edu.au/handbook/c_index.htm

The links on this page in turn pointed to the reports for the individual pages and these provided still more detail. They pointed to the individual checkpoints of the WCAG and the section of the university page's content being evaluated and the current state of the evaluation. The links to WCAG pointed directly into the list of techniques for making the content accessible and to information about how to ensure it was accessible. Where it was shown that there was not yet a value, because visual checking by a human was required, this was identified and the user could make the assessment and change that value.

3.3.Repairing Inaccessible Content

In the case of the software being used, once it is discovered that some content has an accessibility flaw, the user can switch from the evaluation software to repair management software and be led through the process of correcting the problem.

4. The Metadata's Role

It is clear that detailed information about accessibility problems is necessary for evaluation, repair, and management of the evaluation process and the management decisions to be made after the evaluation.

In the test case, it was obvious that it was a few errors in templates that were causing a vast number of problems. This information made it very much easier for management decisions to be made about the content, including those to do with levels of compliance as well as processes for improvement. The on-going management of the metadata, integrated into the repair process, and suitable for regeneration on a regular basis, is a perfect task for a metadata repository.

The detailed information developed by AccVerify, as with other similar software, can be exported as metadata in Resource Description Framework (RDF) (vii) format. This metadata is, in fact, in a special form of RDF encoding that constrains the use of RDF to provide for details that are considered to be of special value in

https://doi.org/10.23106/dcmi.952107936

this context. This format is known as Evaluation and Reporting Language (EARL) (viii) and was developed by W3C for this purpose. The constraints ensure that an RDF statement is accompanied by information about when it was made and by whom or what. These details are often of particular significance when the veracity of the evaluation is being considered.

John Foliot claims that P3P provides a model for using the <link> tag for attaching information to pages. He says the same mechanism could be used for attaching EARL information. Whether this is the best way to attach the EARL statement is one question, but the motivation for doing it is another:

"Named accountability for accessibility would force developers to take the testing and evaluation process more seriously than clicking a button on a software application. The EARL report could/would clearly indicate the benchmarks/standards against which the page was tested, when the testing took place, and which tools and methodology were used to assess the resource. If you missed manual checks, the EARL report would indicate as such [sic]. When you have to sign your name to your work (perhaps using digital signatures) you are more likely to ensure that the report is accurate and thorough. Companies and other institutions could (should!) make inclusion of EARL reports mandatory via Internal Policy, etc., thus closing the development loop.

This would also force software developers to address current shortcomings in accessibility evaluation products or risk losing market share (the old carrot and stick method). If an evaluation tool fails to provide complete EARL reports as part of the reporting process (including the known requirement for manual checks), then the EARL report would not be available to link to the document." (ix)

In a workshop at CSUN in 2002, Wendy Chisholm pointed to two uses of EARL metadata:

"This information is stored in EARL so that other tools can make use of it. For example, if several authors use EARL to describe the accessibility of their Web sites, then a search engine could look through this information to find a site that meets specific accessibility requirements. Someone who does not read well could look only for sites that supplement text with images, multimedia, and other illustrations.

Our primary use of EARL during this session, will be to combine results from a variety of tools to help summarize the issues with a site." (x)

Chisholm's last point, about the mixing and matching of metadata, is an important one. There is no single tool that tests well for all aspects of accessibility. Experts use different tools for different parts of the process and then want to combine the results.

https://doi.org/10.23106/dcmi.952107936

Having results in EARL format enables this.

But accessibility metadata is not only useful in the process of determining compliance of content. There is ongoing work to make inaccessible content available by providing services, transforms, and other applications to match users to content. There is already a user profile that can be used to determine the needs and preferences of users, particularly for those with temporary or permanent disabilities (xi) This profile can be automatically matched to content descriptions which are currently being developed (xii). Content management systems are then able to match users to content, substituting more appropriate content on the fly, or replacing inaccessible to the user sections of content with accessible equivalent content. This means the user can expect to get access to the content they need, regardless of the need for the access device to be reconfigured, or images to be replaced by text, for example.

The accessibility metadata application profile will expect to find an EARL statement that will offer the sort of detail that will be necessary for the accessible rendering of all content. This would be a fanciful dream if it depended upon the human creation of metadata, but as it is now developed by many tools that work in similar ways to those used in the case study, this is not considered a long-term problem.

5. Conclusion

The DC Accessibility Working Group is confident that tools like AccVerify will make generating metadata about accessibility easier. They are also sure that the pressure for compliance will drive the adoption of such tools. To that end, the Working Group participated in an IMS-led effort to develop user profiles and matching resource for a new accessibility term. Crucial to the success of the overall effort to make Web resources more accessible is the availability of the metadata created by the tools. Once available, it can be re-purposed to satisfy not only the needs of those who care about compliance for regulatory reasons, but for those who work to ensure that resources are matched to users' needs and preferences.

Importantly, the process and tools described could be used to generate and manage other metadata.

6. Note

AccVerify is one of several tools that generate EARL statements - see also the Accessibility Checker from the Assistive Technology Resource Centre at the University of Toronto (xiii), Accessibility Valet Demonstrator (xiv) and now Wave 3.5 (xv). There is also significant development work going on in non-English speaking countries.

https://doi.org/10.23106/dcmi.952107936

References

i http://www.section508.gov/
ii http://www.w3.org/TR/WCAG10/
iii http://www.w3.org/TR/ATAG10/
iv http://www.testingautomation.com/sitemanager.html
v http://www.hisoftware.com/access/
vi http://bobby.watchfire.com/
vii http://www.w3.org/RDF/
viii http://www.w3.org/2001/03/earl
ix http://www.wats.ca/articles/accountabilityinaccessibilitytesting/54 on 19/04/2004
x http://www.csun.edu/cod/conf/2002/proceedings/260.htm on 19/04/2004
xi http://www.imsglobal.org/accessibility/index.cfm
xii http://www.imsglobal.org/accessibility/index.cfm
xiii http://checker.atrc.utoronto.ca/index.html
xiv http://valet.webthing.com/
xv http://www.wave.webaim.org/wave35/index.jsp

https://doi.org/10.23106/dcmi.952107936