The BenToWeb XHTML 1.0 Test Suite for the Web Content Accessibility Guidelines 2.0 - Last Call Working Draft

Christophe Strobbe[1], Jan Engelen[1], Johannes Koch[2], Carlos Velasco[2], Evangelos Vlachogiannis[3], and Daniela Ortner[4]

[1] Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Heverlee-Leuven, Belgium
{Christophe.Strobbe, Jan.Engelen}@esat.kuleuven.be
[2] Fraunhofer-Institut für Angewandte Informationstechnik (FIT), Schloss Birlinghoven, D-53757 Sankt Augustin, Germany
{Johannes.Koch, Carlos.Velasco}@fit.fraunhofer.de
[3] University of the Aegean, Voulgaraktonou 30, GR11472 Exarchia, Athens, Greece
evlach@aegean.gr
[4] University of Linz "integriert studieren - integrated study" (i3s3), Altenbergerstrasse 69, A-4040 Linz, Austria
Daniela.Ortner@jku.at

Abstract. This paper presents the work carried out under the umbrella of the EU-funded project BenToWeb to develop a complete XHTML 1.0 test suite for the W3C's Web Content Accessibility Guidelines 2.0. Initial work covered the June 2005 working draft, which was subsequently updated to the April 2006 working draft ("last call"). At the time of writing, a thorough evaluation, involving end users, is being carried out.

Keywords: test suite, accessibility, Web Content Accessibility Guidelines (WCAG), evaluation and repair tools (ERT), HTML, XHTML, CSS, Web Accessibility Initiative (WAI), World Wide Web Consortium (W3C), Last Call Working Draft.

Introduction

The EC-funded project BenToWeb (Benchmarking Tools and Methods for the Web[1]) has a goal to develop test suites for the forthcoming Web Content Accessibility Guidelines 2.0. The development of these test suites serves a dual purpose. First, they support the WAI Working Groups in the development of support documents, such as technology-specific techniques, for WCAG 2.0. The test files prove whether techniques can be implemented, and help find out ambiguities and loopholes in

[1] http://www.bentoweb.org/

WCAG documents. Second, the test suite can be used to benchmark accessibility evaluation and repair tools (ERT). Failing test files can also be used to test conformance to the Authoring Tools Accessibility Guidelines (ATAG)[2], for example to test if a tool's user interface allows authors to avoid flashing in a WYSIWYG editing view.

WCAG 2.0 currently has general techniques, HTML techniques, CSS techniques, client-side scripting techniques and server-side techniques. BenToWeb will not cover all of these technologies exhaustively, but at least XHTML 1.0 + CSS 2.0. In 2005, BenToWeb created a test suite for the June 2005 working draft of WCAG 2.0. After the WCAG Working Group published the Last Call Working Draft of WCAG 2.0[3] in April 2006, this test suite was updated to the newest draft.

Testing and Comparing Evaluation and Repair Tools

Since the publication of Web Content Accessibility Guidelines (WCAG) 1.0, there have been many efforts to automate the evaluation of web pages or even complete websites against these guidelines. There is great variability in the strengths and features of these tools, and comparing them is a time-consuming task. Several researchers have already reported on this type of work, including Brajnik [1] and Melody Ivory [3,4,5]. The problem with these comparisons is that they usually rely on web pages or on samples from websites that are not available for other researchers to repeat the study with other tools. The outcome of the research is partially based on an unknown input. A publicly available suite of test cases that have been individually described and validated, would eliminate this the problem of repeatability. For each test case, it should be clear which accessibility guidelines are violated (or that specific guidelines are not violated), and how often they are violated. This documentation can then be compared with the report of an evaluation tool that has "checked" the test suite, in order to find out what guidelines are covered by the product, what guidelines trigger false positives, false negatives, etcetera. A publicly available test suite thus enables comparisons or benchmarking of products, and comparisons of the same product over time.

Other Accessibility Test Suites

The World Wide Web Consortium's Quality Assurance Activity[4] has a goal to ensure that its deliverables – W3C Recommendations – are implemented correctly. The development of test suites can contribute to this goal. In the area of web accessibility, both inside and outside the Web Accessibility Initiative (WAI), there have been several efforts to build test suites. The scope, size and status of these test suites vary

[2] http://www.w3.org/TR/ATAG20/
[3] http://www.w3.org/TR/2006/WD-WCAG20-20060427/
[4] http://www.w3.org/QA/

greatly: in the beginning of 2005, it was not clear if any of these could be considered as finalized [6].

There are test suites for the User Agent Accessibility Guidelines 1.0[5], for the Web Content Accessibility Guidelines 2.0[6], and for several software products, such as screen readers and the Mozilla browser. The HTML Test Suite for WCAG 2.0 comes closest to the type of test suite developed by BenToWeb. The test suite is no longer being updated, but the WCAG Working Group has integrated tests into the techniques document for WCAG 2.0[7]. In 2006, the Evaluation and Repair Tools Working Group (ERT WG) and the WCAG Working Group also set up a joint "Test Samples Development Task Force", with the objective to develop test samples for WCAG 2.0 Techniques[8].

Structure of the Test Suite

In this context, a test suite is not a set of tests that can be used to validate web content, but a set of test files with accompanying metadata both for human and machine consumption. The test suite is a collection of "test cases", where a test case consists of one or more XHTML files that implement or fail a requirement specified by a WCAG 2.0 success criterion, and an accompanying metadata file. The metadata are recorded in an XML format specially created for this purpose: Test Case Description Language (TCDL) [8]. The metadata include a short title and a description of the test file or files, a statement on whether the test files pass or fail the success criterion, the location of the issue if the test files fail, and guidance on what is necessary to validate the test case, for example scenarios for end-user evaluation.

For each WCAG 2.0 success criterion, at least two test cases need to be created: at least one that fails and at least one that passes the success criterion. This is because there needs to be at least one test for a false positive, and at least one test for a false negative, respectively. When the test suite is complete and validated, running the test files through an accessibility evaluation tool should then provide data on the completeness of the tool's coverage of WCAG 2.0 and whether it generates false positives and negatives.

BenToWeb distinguishes between several types of test cases. "Atomic test cases" address only one success criterion and use only a single XHTML file (supporting files such as images, client-side scripts or style sheets do not count in this context; the XHTML file can either be static or generated with JavaServer Pages (JSP)). However, some accessibility requirements apply to sets of web pages instead of pages in isolation: the Last Call Working Draft of WCAG 2.0 contains success criteria about consistency of navigational mechanisms such as site navigation (SC 3.2.3 and 3.2.4), bypassing repeated content (SC 2.4.1), finding information in a website (SC 2.4.2) and information about a user's location in a website (SC 2.4.7). Test cases for these success criteria use multiple XHTML files and are called "compound test cases".

[5] For example at http://www.w3.org/WAI/UA/TS/html401/.
[6] HTML Test Suite for WCAG 2.0: http://www.w3.org/WAI/GL/WCAG20/tests/.
[7] Techniques for WCAG 2.0: http://www.w3.org/TR/WCAG20-TECHS/.
[8] WCAG 2.0 Test Samples: http://www.w3.org/WAI/ER/tests/.

Ideally, a test suite for benchmarking evaluation software should also contain test cases for combinations success criteria. In BenToWeb, these would be called "complex test cases". The Test Case Description Language supports the description of such test cases: it is possible to reference multiple success criteria, and to state for each whether the test case passes or fails. Previous accessibility test suites do not appear to contain this type of test cases [6].

The Development and Evaluation Process

The development process requires that each test case moves through several steps before it is finally accepted in the test suite. Each test case starts out as a draft and goes through two levels of evaluation. First, it is reviewed by an accessibility or HCI expert. If any issues are found, the test case is sent back to the test case author. These issues may be editorial or content-related. Sometimes they relate to the interpretation of a success criterion. In this phase, the evaluator can also decide that the test case would benefit from end-user evaluation and set up scenarios that match a certain user profile (disabilities, experience with user agents, assistive technologies).

After the author and the first evaluator have solved the issues, the test case is ready for the second level of evaluation. The second evaluator can accept the test case for end-user evaluation, send it back to the test case author, or recommend that the test case be included in or rejected from the test suite. Test cases that contain scenarios for end-user evaluation are loaded into a test case evaluation framework (described by Herramhof (2006) [2]). The framework matches scenarios with user profiles and saves the users' input for later analysis. After evaluation and when all data are definitive, the test case is finally "accepted" into the test suite.

Current Status of the Test Suite

Size of the Test Suite

The first version of the test suite covered the 30 June 2005 Working Draft of WCAG 2.0. It was made up of 477 test cases, each consisting of one metadata file (in TCDL) and one or more content files. The test suite contained over 520 XHTML files (or JSP files that generated XHTML)[9], which often use supporting files, such as JavaScript, CSS, GIF, JPEG, WMV (audio/video), WMA, WAV, MP3 and Java applets.

At the time of writing, the second version of the test suite contains over 530 test cases for the 56 success criteria in the Last Call Working Draft. These test cases contain over 600 XHTML test files (or JSP files that generate XHTML), and a

[9] The numbers cited here are slightly lower than in a previous conference paper (Strobbe (2006) [7]: 481 test cases, over 530 XHTML files) because that paper was written when the evaluation of the test suite was not yet finished.

smaller number of supporting files (JavaScript, CSS, images, audio, video and applets). Both versions of the test suite are publicly available[10].

Interpretation of WCAG Success Criteria

The second version of the test suite covers the 27 April 2006 Working Draft ("last call") of WCAG 2.0. This means that the working draft published on 23 November 2005 was skipped. In the June 2005 working draft, some success criteria were unfinished or open to interpretation. This led to discussions during the development and evaluation of the BenToWeb test suite, and to feedback to the WCAG Working Group. We refer to the conference paper on the first version of the test suite for examples [7]. Between June 2005 and April 2006, the WCAG Working Group solved many issues and published additional supporting documents: "Understanding WCAG 2.0"[11] and "Techniques for WCAG 2.0"[12] (later supplemented with the "WCAG 2.0 Quick Reference"[13]). As a result, there were fewer interpretation issues in BenToWeb; they were also easier to solve internally and were about specific details rather than a success criterion as a whole. Nevertheless, if after internal consultation, project participants were not fully confident about the interpretation of a success criterion, the issue was fed back to the WCAG Working Group.

Comprehensiveness of the Test Suite

The publication of the additional supporting documents by the WCAG Working Group also helps BenToWeb to develop a more comprehensive test suite. When the first test suite was developed, the WCAG Working Group had only defined a relatively small set of HTML techniques for WCAG 2.0. Test case authors were free to draw on any documentation of techniques or failures they could find, regardless whether the source was WCAG or not. When the WCAG Working Group published the Last Call Working Draft of WCAG 2.0, it also published "Techniques for WCAG 2.0", containing both techniques and common failures related to accessibility. BenToWeb test cases can now also be mapped to these techniques and failures, which enables a gap analysis. Gap analyses against the HTML and CSS specifications will also be conducted. Evaluating the comprehensiveness of the test suite and searching for techniques and failures not documented by the WCAG Working Group are ongoing tasks.

Variability in the Test Suite

In the first version of the test suite, some success criteria had only two test cases, while others had more than thirty. This variability is also present in the second version

[10] http://www.bentoweb.org/ts
[11] http://www.w3.org/TR/UNDERSTANDING-WCAG20/
[12] http://www.w3.org/TR/WCAG20-TECHS/
[13] http://www.w3.org/WAI/WCAG20/quickref/

and seems to be inherent in any XHTML test suite for WCAG 2.0. The variability in the number of test cases per success criterion is often related to the number of XHTML elements or attributes that can be used to pass or fail a success criterion. For example, success criterion 3.1.1 requires that "[t]he primary natural language or languages of the Web unit can be programmatically determined." The primary language is usually declared by means of the `lang` and/or the `xml:lang` attribute on the element that contains the complete document (the `html` element). Declaring the primary language on the `body` element may also be acceptable, but this exhausts all the options. There are few techniques to declare the primary language, and few ways to fail this requirement. By contrast, success criterion 1.4.1 requires a luminosity contrast ratio of 5:1 between foreground and background colours. However, colours can be set on basically any element that can appear in the body of a document, and these elements can be nested in many different ways. Success criterion 1.3.1 is an even better example. It requires that "[i]nformation and relationships conveyed through presentation can be programmatically determined, and notification of changes to these is available to user agents, including assistive technologies." This covers the use of heading elements to express the structure of a document, connections between form fields and their labels, semantic markup versus abuse of presentational markup, etcetera. There are many ways to fail this success criterion, each with dozens of variations. For this reason, the variability in the number of test cases per success criterion can be said to be inherent in the test suite.

Ongoing and Future Work

The current draft of WCAG 2.0 (the Last Call Working Draft) is not a final document. BenToWeb will update the test suite to the draft that will become available in early 2007. This means that the mapping of the test cases to success criteria will need to be updated. It also means that test cases will need to be created for success criteria that may have been added after the Last Call Working Draft. Moreover, some existing test cases may need to be reviewed because the related success criteria have changed. Other test cases may disappear if the related success criteria have been removed and do not map to other success criteria. In addition to this, more time will be devoted to evaluating the comprehensiveness of the test suite.

BenToWeb participants are also involved in the Test Samples Development Task Force (TSD TF), a joint task force set up by ERT Working Group and the WCAG Working Group with the objective to review the WCAG 2.0 techniques and failures and to develop test samples for them. This task force uses a subset of TCDL for its metadata[14] and the contribution of BenToWeb test cases will facilitate the uptake of BenToWeb's test suite efforts in WAI.

[14] WCAG 2.0 Test Samples Metadata: http://www.w3.org/WAI/ER/tests/usingTCDL.

Acknowlegements. This work has been undertaken in the framework of the project BenToWeb — IST-2-004275-STP — funded by the IST Programme of the European Commission.

References

1. Brajnik, G (2004). "Comparing accessibility evaluation tools: a method for tool effectiveness." Univ Access Inf Soc 3: pp. 252-263. DOI: 10.1007/s10209-004-0105-y
2. Herramhof, S, Petrie, H, Strobbe, C, Vlachogiannis, E, Weimann, K, Weber, G, Velasco C A (2006). "Test Case Management Tools for Accessibility Testing." In: Miesenberger K et al (eds). Proceedings of the 10th International Conference ICCHP 2006 (Linz, Austria, July 2006), LNCS 4061, pp. 215-222. Berlin-Heidelberg: Springer-Verlag.
3. Ivory, M Y, Sinh, R R, and Hearst, M A (2001). "Empirically validated web page design metrics." Proceedings of the Conference on Human Factors in Computing Systems (Seattle, WA, March), pp. 53—60. New York, NY: ACM Press.
4. Ivory, M Y, and Chevalier, A (2002). "A Study of Automated Web Site Evaluation Tools." Technical Report UW-CSE-02-10-01. University of Washington, Department of Computer Science and Engineering.
5. Ivory-Ndiaye, M Y (2003). "An Empirical Approach to Automated Web Site Evaluation." Journal of Digital Information Management, 1(2), June 2003, pp. 75-102.
6. Strobbe, C (2005). "Test-suites' State of the Art and Quality Assurance Methods for W3C Recommendations." BenToWeb deliverable D 4.1. Available at: http://www.bentoweb.org/html/BenToWeb_D4.1.html.
7. Strobbe C, Herramhof S, Vlachogiannis E, Koch J, Velasco C A (2006). "The BenToWeb XHTML Test Suite for the Web Content Accessibility Guidelines 2.0" In: Miesenberger K et al (eds). Proceedings of the 10th International Conference ICCHP 2006 (Linz, Austria, July 2006), LNCS 4061, pp. 172-175. Berlin-Heidelberg: Springer-Verlag.
8. Strobbe C, Herramhof S, Vlachogiannis E, Velasco C A (2006). "Test Case Description Language (TCDL): Test Case Metadata for Conformance Evaluation" In: Miesenberger K et al (eds). Proceedings of the 10th International Conference ICCHP 2006 (Linz, Austria, July 2006), LNCS 4061, pp. 164-171. Berlin-Heidelberg: Springer-Verlag.