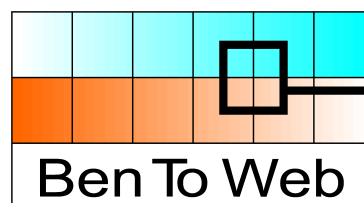


**Benchmarking Tools and
Methods for the Web
(FP6—004275)**



Sixth Framework Programme
Information Society Technologies Priority

D4.6 Guidelines for the Development of Test Suites

Contractual Date of Delivery to the EC:	30 September 2007 + 45 days
Actual Date of Delivery to the EC:	3 December 2007
Editor:	Christophe Strobbe (K.U. Leuven)
Contributors:	Christophe Strobbe (K.U. Leuven), Carlos A Velasco (FIT), Evangelos Vlachogiannis (AEGEAN)
Workpackage:	WP4
Security:	Public
Nature:	Report
Version:	B
Total number of pages:	15

Keywords: test case, test suite, guidelines, accessibility, Web Content Accessibility Guidelines, WCAG, Web Accessibility Initiative, WAI.

DOCUMENT HISTORY			
Version	Version date	Responsible	Description
A	30 August 2007	K.U.Leuven	First TOC and draft
B	15 November 2007	K.U.Leuven	Final version

Table of Contents

1	Executive Summary.....	4
2	Introduction.....	5
3	Development of Test Cases.....	6
3.1	Define a Metadata Scheme.....	6
3.2	Define the Scope of the Test Suite.....	7
3.3	Define Completeness Criteria for the Test Suite.....	7
4	Validation of Test Cases.....	9
4.1	Define a Validation Process.....	9
4.2	End-User Evaluation.....	9
5	Test Case Metadata.....	11
5.1	Define Criteria for the Metadata Elements.....	11
5.2	Define How to Evolve the Metadata Format.....	11
6	References.....	13

1 Executive Summary

This report discusses guidelines for the development of suites of test cases for web content accessibility guidelines such as the Web Content Accessibility Guidelines 2.0 (WCAG 2.0). It is based on the experience of the BenToWeb project in developing three versions of an XHTML test suite for three drafts WCAG 2.0. The guidelines relate to the development of test case, their validation and the metadata format that supports the process. BenToWeb also undertook end-user evaluation for a subset of the test cases; guidelines for this type of evaluation are also provided.

2 Introduction

This document defines guidelines for the development of test suites for web content accessibility specifications. This type of test suites is somewhat different from other test suites developed for W3C specifications: whereas most W3C test suites are based on a single specification for a content format, the type of test suite discussed here is based on two types of specifications: one for a content format (for example, XHTML) and one that defines accessibility guidelines (WCAG). This makes the development and validation of a test suite somewhat less straightforward than for some other test suites.

The guidelines document that the BenToWeb project targeted, WCAG 2.0, was under development during the complete duration of the project and went through several new drafts. Consequently, the original XHTML test suite was updated to two newer drafts. In the process, the tools, processes and data formats developed by the project were refined and improved. The guidelines in this document reflect some of the experience gathered during the development and validation of three versions of an XHTML test suite for WCAG 2.0.

In this document, we will assume that a test case consists of one or more test files with associated metadata.

3 Development of Test Cases

3.1 Define a Metadata Scheme

A test case consists of one or more test files, and these need to be accompanied by metadata. How metadata are stored (a database, XML, RDF, a wiki, or even combinations of these) is probably a matter of preference. However, it is necessary that the metadata scheme supports the development and validation process of the test suite. At the very least, these metadata should contain a description, the purpose and the status of the test case. Test case metadata in W3C test suites are often more detailed than this; the W3C Quality Assurance Working Group has published a note that defines 14 metadata elements [W3C 2005].

In BenToWeb, metadata are stored in an XML language called “Test Case Description Language 1.1” (TCDL 1.1)¹, which describes metadata such as:

- formal metadata, such as author, date, title and a short description;
- status of the test case;
- technologies (the technologies and technology features used in the test);
- test mode and test scenarios for end user evaluation;
- the “rules” that are tested (WCAG 2.0 success criteria or other rules);
- namespace mappings used in some of the previous sections.

BenToWeb's TCDL 1.1 has elements and attributes that map to most of the metadata elements defined by the Quality Assurance Working Group; TCDL 2.0 has elements and attributes that map to all but one of those elements (only 'grouping' is not included).

A good metadata format will support a number of important tasks such as

- monitoring the completeness of test suite,
- track the status of test cases;
- evolving the metadata format to changing needs or to better support existing needs (this is discussed in a separate section).

¹ TCDL 1.1 RDDL file: <http://bentoweb.org/refs/TCDL1.1/>. See also BenToWeb deliverable 4.2 and [Strobbe].

3.2 Define the Scope of the Test Suite

What is the test suite meant to cover? A test suite can contain different types of test cases. For example, BenToWeb makes the following distinction:

- atomic test case: a test case that addresses one checkpoint (WCAG 1) or success criterion (WCAG 2) and that uses only a single file;
- compound test case: a test case that addresses one checkpoint (WCAG 1) or success criterion (WCAG 2) and that uses a set of files (e.g. to test consistency of navigation);
- complex test case: a test case that addresses multiple checkpoints (WCAG 1) or success criteria (WCAG 2), to allow for more thorough testing of evaluation tools; complex test cases may use one or more test files (i.e. they can be atomic or compound).

In BenToWeb, these distinctions are linked to the following rules for creating test cases:

- If a test case is atomic, there can be only one test file in the `files` section of the TCDL description. If one has multiple files that are different variations of the same test, for example video in different formats, then these variations really constitute different test cases, each requiring a separate TCDL file.
- Whether a test case is atomic or complex is indicated in the `complexity` attribute of the `testCase` element. The `complexity` attribute does not accept the value "compound". Whether a test case is compound or not is inferred from the number of files (referenced in the `files` element) that constitute the test.

In addition to the types of test cases, it is also necessary to define which human languages will be covered, especially for those parts of the specification that address language issues, for example, the criteria under Guideline 3.1 in WCAG 2.0 [W3C 2007a]. The choice of languages has implications for the validation process, especially when it involves end-user testing.

3.3 Define Completeness Criteria for the Test Suite

Define what "completeness" means for the test suite. In the case of test suites for WCAG 2.0 there are several ways to define completeness;

taking an XHTML test suite for WCAG 2.0 as an example one could do the following:

- for each success criterion, make sure that there is a test case for each relevant feature of XHTML that there is at least one “fail” example and one “pass” example for each success criterion; or
- for each WCAG 2.0 technique or failure that applies to XHTML, check that there is at least one test case.

The first approach will lead to the creation of many test cases that do not map to a WCAG 2.0 technique or failure (because the techniques document is not exhaustive²) and to the creation of test cases that may be considered as “exotic” (for example, an object element that embeds a complex image, with the long description of the image nested inside the object element).

The second approach will lead to good coverage of the WCAG 2.0 techniques and failures, but may miss many types of failures that would be covered by the first approach.

In each case, the metadata scheme will need features that enable developers to document how a test case relates to the technical specification or document format (XHTML in the above examples) and/or the techniques and failures. TCDL 1.1 and TCDL 2.0 support both approaches. In TCDL 2.0 [TCDL2], references to features of document formats can be documented in the 'technicalSpec' element³, and references to techniques and failures can be documented in the 'techniques' element⁴.

² The WCAG 2.0 techniques document does not define failures that would merely state the opposite of the relevant success criterion: failures need to be more specific and fairly common.

³ <http://bentoweb.org/refs/TCDL2.0.html#edef-technicalspec>

⁴ <http://bentoweb.org/refs/TCDL2.0.html#edef-techniques>

4 Validation of Test Cases

4.1 Define a Validation Process

A validation process or review process defines the phases that a test cases needs to go through before it is finally accepted as a valid test case. This process defines several states and input and output channels for each state. The validation process needs to be shown to be complete and unambiguous. It is good practice to test the validation process by running a few example test cases through the complete process.

For BenToWeb's validation process, see “Updated methodology for the validation of test cases” in BenToWeb deliverable D3.7b. For examples of other validation or review processes see:

- WCAG 2.0 Test Samples Development Task Force (TSD TF) Review Process [TSD TF Process];
- Conformance Test Process for WCAG 2.0 [WCAG 2005b],
- “Test Review Guidelines” in [Henderson 2001].

4.2 End-User Evaluation

Evaluation or validation processes often rely on the opinion of experts, but it is also possible to involved end users. There may be several reasons for doing this, for example

- for evaluating test cases that are too “exotic” to be covered in any specification,
- for comparative evaluation of test cases that use different techniques to address the same accessibility issue (or WCAG 2.0 success criterion),
- for comparative evaluation of test cases that are variations on the same technique or failure;
- for evaluating test cases that illustrate techniques that may not be completely reliable;
- for evaluating test cases that may lead to different outcomes depending on support by user agent and/or assistive technologies.

Involving end users affects the tools and the metadata used in the development and validation process: tools and metadata need to support

the definition of evaluation scenarios, the presentation of these scenarios to users, and the collection and analysis of the users' feedback. In BenToWeb, the actions that were taken included the following:

- TCDL (the metadata format) defined several types of questions that could be presented to users (yes-no questions, open questions, Likert scales, ...) that could be used in a "scenario";
- TCDL allowed test case validators to define the disabilities and the combinations of user agents and assistive technologies that are needed to go through the scenario;
- a test case evaluation framework (called "Amfortas" see [Herramhof]) was developed in which users could sign in and create a user profile (disabilities, user agent, assistive technologies, ...);
- a "profile matching algorithm" was defined to match test cases to end user profiles;
- the validation process defined a state in which feedback from end-user testing could be integrated into the test case.

The following things need to be kept in mind when organizing end-user testing.

- It is important that the scenarios are validated by someone who has a good understanding of the relevant accessibility guidelines (e.g., WCAG 2.0), in order to make sure that the scenario actually tests conformance to, for example, the success criterion.
- It is necessary to define the minimum number of users that need to go through the scenarios for a test case, both with regard to representativeness of the evaluations and for practical reasons (spreading users over test cases; knowing when to stop evaluation of a test case).
- Some end users may get few test cases or even none at all. Decide in advance what you will tell them.
- Do no harm: test cases with flashing content (photosensitive epilepsy) or that trap the keyboard must be excluded from end-user testing.
- Don't include too many test cases that can easily be evaluated as "pass" examples by end users; otherwise, end users may find the evaluation trivial. It is better to present a mix of test cases with different levels of difficulty. The user profile matching can be used to make sure that the level of difficulty is adapted to the user's experience level.

5 Test Case Metadata

5.1 Define Criteria for the Metadata Elements

The purpose of each metadata element needs to be described. For example, BenToWeb's TCDL format uses elements such as 'title', 'description' and 'purpose' and defines the following rules for these elements.

- The title should be sufficiently descriptive to allow a quick identification of the issue illustrated by the test case. It does not need to be unique. It must not be formulated as a guideline (for example: "All data tables must have a summary");
- The description contains a summary of the test materials and how they are to be used in the test case. It indicates to the accessibility expert/scenario author what to expect and what will happen when a user interacts with the test materials. The importance of accurate descriptions can not be underestimated because bad descriptions cause many requests for clarifications, invalid scenarios and other problems that slow down the validation process.
- The purpose contains a description of the intention of the test materials. The purpose contains an explanation of the expected evaluation result in regard to the relevant rules, checkpoints or success criteria. It does not need to repeat the "rule" or success criterion for which the test case is developed.

5.2 Define How to Evolve the Metadata Format

One should expect the requirements of the metadata format to change over time, especially when a test suite goes through several versions as the targeted specification evolves. Assuming that an XML format is used, one will need to consider the following questions.

- Should the schema be updated or replaced with a new one? (Example, TCDL 1.1 and TCDL 2.0).
- If the schema does not need to change drastically and it is just updated, what is the impact on existing metadata, both those in the current test suite and in previous test suites?
- If the schema is replaced with a new one, will metadata for previous test suites be migrated to the new format or not?

- What is the impact of the changes to the tools (including XSLT)?

In BenToWeb, a number of changes to TCDL 1.1 were implemented in the following way:

- new elements and attributes that were not used in previous test suites were added as optional elements and attributes, in order not to break validity of older test suites;
- changes to attribute values were implemented by first adding the new values to the schema, then replacing old values with new values in the metadata files, and finally deleting the old (“deprecated”) values from the schema.

6 References

- [Bobroskei] Bobroskie, A.; Çelik, T. 2002. "HTML 4 Test Suite Documentation".
<http://www.w3.org/MarkUp/Test/HTML401/current/htmltestdocumentation.html>
- [Dubost 2006] Dubost, K.; Rosenthal, L. 2006. Test Assertion Guide. Editor's Draft 28 April 2006.
<http://www.w3.org/2006/03/test-assertion-guide>.
- [Herramhof] Herramhof, S.; Petrie, H.; Strobbe, C.; Vlachogiannis, E.; Wiemann, 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.
- [Henderson 2001] Henderson, L. 2001. "SVG Conformance Test Suite — Test Builder's Manual".
<http://www.w3.org/Graphics/SVG/Test/svgTest-manual.htm>
- [Strobbe] 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.
- [TCDL2] Strobbe, C., ed. (no date) Test Case Description Language 2.0: Specification and Guide.
Latest version: <http://bentoweb.org/refs/TCDL2.0.html>.
- [TSD TF Process] WCAG 2.0 Test Samples Development Task Force (TSD TF) Review Process. <http://www.w3.org/WAI/ER/tests/process>.
- [WCAG 2005a] DRAFT WCAG 2.0 Implementation Testing Framework.
<http://www.w3.org/WAI/GL/implementation-testing/Overview.html>.
- [WCAG 2005b] Conformance Test Process for WCAG 2.0. W3C Working Draft 2005-02-23. (Edited by Chris Ridpath and Jenae Andershonis).
<http://www.w3.org/WAI/GL/WCAG20/tests/ctprocess.html>.
- [WCAG 2005c] HTML Test Suite for WCAG 2.0 Sorted by Guideline. W3C Editor's Draft 2005-08-11. (Edited by Chris Ridpath and Jenae Andershonis). <http://www.w3.org/WAI/GL/WCAG20/tests/>.

- [W3C 1999] World Wide Web Consortium. 1999. Web Content Accessibility Guidelines 1.0. (Edited by Wendy Chisholm, Gregg Vanderheiden and Ian Jacobs). W3C Recommendation 5-May-1999. <http://www.w3.org/TR/WAI-WEBCONTENT>.
- [W3C 2002a] World Wide Web Consortium. 2002. Conformance Testing and Certification Model for W3C Specifications. W3C Working Draft 2 January 2002. <http://www.w3.org/QA/2002/01/Note-qa-certif-20020102.html>.
- [W3C 2002b] World Wide Web Consortium. 2002. Requirements for WCAG 2.0. (Edited by Gregg Vanderheiden and Wendy Chisholm). W3C Working Draft 26 April 2002. <http://www.w3.org/TR/wcag2-req/>.
- [W3C 2002c] World Wide Web Consortium. 2002. Authoring Tool Conformance Evaluations. (Last updated by Ian Richards, 28 May 2002.) <http://www.w3.org/WAI/AU/2002/tools>.
- [W3C 2004a] World Wide Web Consortium. 2004. World Wide Web Consortium Process Document. (Edited by Ian Jacobs.) 5 February 2004. <http://www.w3.org/2004/02/Process-20040205/>.
- [W3C 2004b] World Wide Web Consortium. 2004. QA Framework: Test Guidelines. (Edited by Patrick Curran and Dimitris Dimitriadis.) W3C Working Draft 25 February 2004. <http://www.w3.org/TR/2004/WD-qaframe-test-20040225/>. (Progression and publications of this document is on hold since August 2004⁵; the Quality Assurance Working Group's charter ended in August 2005⁶ and has not been renewed since then.)
- [W3C 2004c] World Wide Web Consortium. 2004. The QA Handbook. (Edited by Lofton Henderson.) W3C Working Group Note 22 November 2004. <http://www.w3.org/TR/qa-handbook/>.
- [W3C 2005] World Wide Web Consortium. 2005. Test Metadata. (Edited by Patrick Curran and Karl Dubost.) W3C Working Group Note 14 September 2005. <http://www.w3.org/TR/2005/NOTE-test-metadata-20050914/>.
- [W3C 2007a] World Wide Web Consortium. 2007. Web Content Accessibility Guidelines 2.0. (Edited by Ben Caldwell, Michael Cooper, Loretta Guarino Reed and Gregg Vanderheiden). W3C Working Draft 17 May 2007. <http://www.w3.org/TR/2007/WD-WCAG20-20070517/>.

⁵ <http://www.w3.org/TR/2004/WD-qaframe-test-20040820/>

⁶ <http://www.w3.org/QA/WG/2005/11/wg-overview>

- [W3C 2007b] World Wide Web Consortium. 2007. Understanding WCAG 2.0: A Guide to Understanding and Implementing WCAG 2.0. (Edited by Ben Caldwell, Michael Cooper, Loretta Guarino Reed and Gregg Vanderheiden). W3C Working Draft 17 May 2007.
<http://www.w3.org/TR/2007/WD-UNDERSTANDING-WCAG20-20070517/>.
- [W3C 2007c] World Wide Web Consortium. 2007. Techniques for WCAG 2.0: Techniques and Failures for Web Content Accessibility Guidelines 2.0. (Edited by Ben Caldwell, Michael Cooper, Loretta Guarino Reed and Gregg Vanderheiden). W3C Working Draft 17 May 2007.
<http://www.w3.org/TR/2007/WD-WCAG20-TECHS-20070517/>.