



W₃C Specifications Project

Project Documentation

Jefferson University UX Team

Introduction

We the students at Jefferson University have been working on possible solutions for the W3C specifications pages under the guidance of Lauren Dillard, an adjunct professor at Jefferson University.

We have been working closely with members of the W3C community including Erika Etemad.



The Start

Students of Master's in User Experience and Interaction Design program at Jefferson University, have been invited by former Jefferson student and current W3C member Elika Etemad to participate in improving the usability of the specifications through changes to the design.

We will not be addressing changes to the content, structure or HTML code of the specifications themselves.

Outline

The Problem

Design Process

Solution Proposal

Next Steps

The Problem

The Problem

Design Process

Next Steps

Design Brief

Heuristic Evaluation

Competitor Analysis

Surveys

Interviews

Problems

Defining the Problem

Define and present project objectives - Design Brief

Evaluate the current state of the product - Heuristic Evaluation

Conduct, analyze and present user research - Task Analysis Findings

Conduct, analyze and present user research - Interview Findings

Design Brief

Defining the scope of the project, target audience, research methodologies, expected outcome and the timeline.

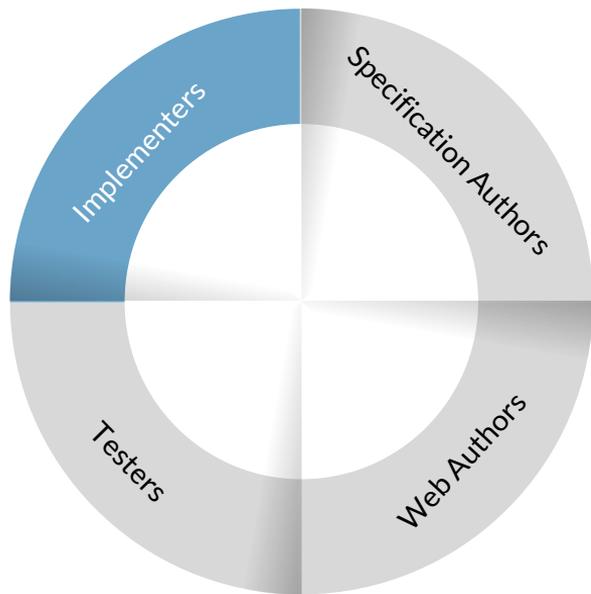
After our interviews and discussions with W3C, this design brief was formed to state the vision for our collaboration with the W3C. We have compiled this brief to inform and guide us through the process.

Users

The end users of W3C web standards fall into different categories based on their purpose of using these standards. These are the key stakeholders who are either working to create or using these standards. We've listed down these users below to give a brief idea about their roles.

1. Implementers
2. Specification Authors
3. Web Authors
4. Testers

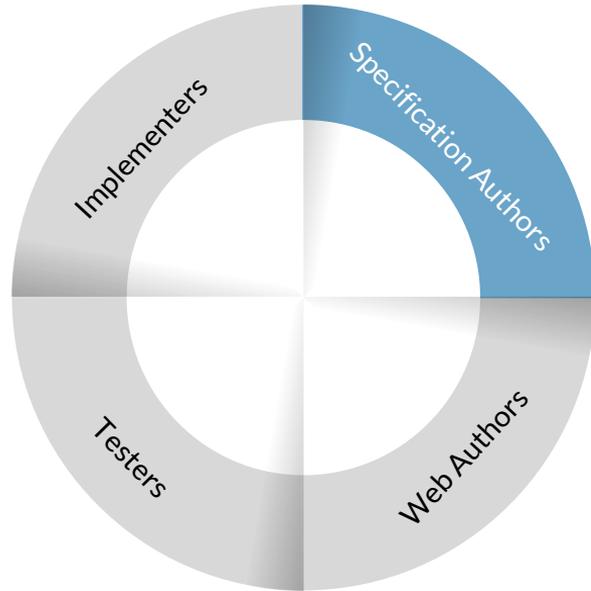
Implementers



The users in this group are generally using the specifications to implement the specification in their web browser.

Based on what is written in the specification, implementers can ensure their browser will read that code.

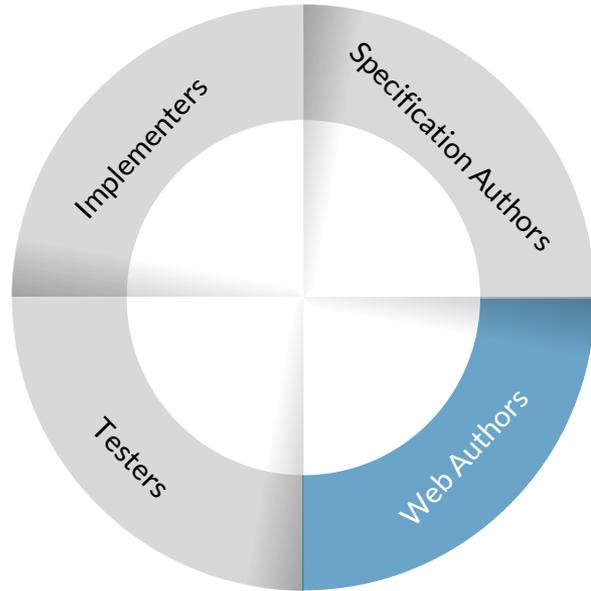
Specification Authors



Users who write new specifications. They are also members of a working group at W3C that has access to tools to write specifications and approve new specifications.

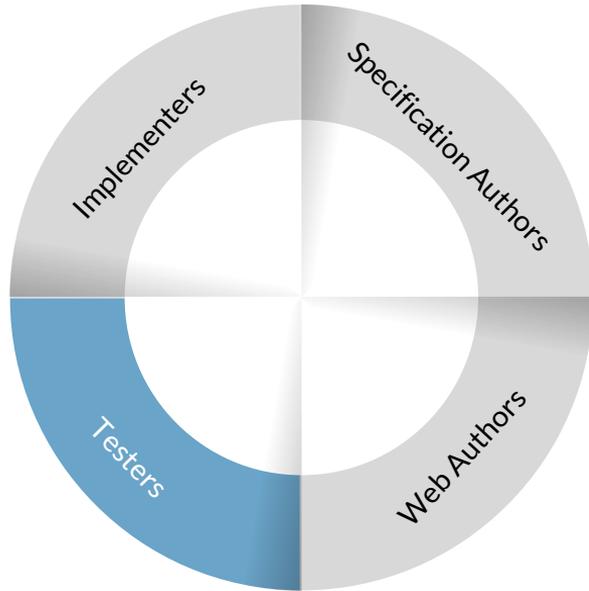
These authors are often informed by members of the public at large who show keen interest in the specifications and offer recommendations, feedback, and comments.

Web Authors



This group comprises of front-end developers, web designers, software developers or anyone creating content for the web.

Testers



This group comprises quality assurance specialists who test the specifications and sample code to report and assure compliance with current web standards and best practices.

The Tone

The W3C is an authority for the development of the web. The existing design of the site is very traditional, functional and definitive. Though it is our goal to improve the usability of the site, we intend to retain the existing tone of authority.

The intention is to adopt modifications in visual design to improve the experience of searching, reading and referring to the specifications. We aim to develop a design that will have a long shelf life that is sustainable over time.

Deliverables

- The project is expected to run through early May over a few months.
- The first few steps would involve intensive discussions with the client, understanding their requirements and hence defining the scope of the project.
- We have begun, with a duration of two weeks assigned to conducting user interviews, which will be followed by task analysis and evaluation.
- This will help define the personas and journey maps that will be used during the design phase of the project to inform our work.

Deliverables

Finally, we will dedicate the next step would comprise a larger section of time dedicated to creating high- fidelity wireframes, conducting user testing and iterating on the designs. This will be one of the most crucial phases in finalising the design.

The last few weeks will be entirely focussed on visual design and compilation of work. If time in our semester allows, we will deliver updated high-fidelity wireframes, a style guide, usability test findings and a functional (clickable) prototype.

Heuristic Evaluations

A heuristic evaluation is a usability inspection method for computer software that helps to identify usability problems in the user interface (UI) design. It specifically involves evaluators examining the interface and judging its compliance with recognized usability principles (the "heuristics").

Designers can obtain feedback in the early stage of the design process. Assigning a heuristic can help determine the best corrective measures. It can be done very quickly and without recruiting external users. You can use it together with other usability testing methods.

Findings

Status of Draft

After accessing any of the dozens of specification topic areas, the user is directed to a table with a list of completed work. This list cannot be reordered (sorted, filtered or searched) and uses language to describe the status that is unclear on first glance. As a user, I need to know what I'm looking for (and likely use the Ctrl+F function) to find a specification. This organizational schema is system specific and not designed for the user.

2008-04-11	Cascading Style Sheets (CSS1) Level 1 Specification	Recommendation
2000-11-13	Document Object Model (DOM) Level 2 Style Specification	Recommendation
2017-02-16	Ready-made Counter Styles	Group Note <i>Nightly Draft</i>
2017-01-31	CSS Snapshot 2017	Group Note <i>Nightly Draft</i>
2017-01-17	TTML Media Type Definition and Profile Registry	Group Note <i>Nightly Draft</i>
2015-10-13	CSS Snapshot 2015	Group Note <i>Nightly Draft</i>

Findings

Browsing between specifications

There is no indication of how the different version of specifications differ from one another. They all look the same even if the content is different. There is no proper introduction. It is deceiving to the user, especially if they're using the specifications for the first time.

Digital Publishing Accessibility API Mappings



W3C Recommendation 14 December 2017

This version:

<https://www.w3.org/TR/2017/REC-dpub-aam-1.0-20171214/>

Latest published version:

<https://www.w3.org/TR/dpub-aam-1.0/>

Latest editor's draft:

<https://w3c.github.io/dpub-aam/>

Implementation report:

<https://w3c.github.io/test-results/dpub-aam/>

Previous version:

<https://www.w3.org/TR/2017/PR-dpub-aam-1.0-20171102/>

Editors:

Richard Schwerdtfeger, Knowbility, richschwer@gmail.com
Joanmarie Diggs, Igalia, S.L., jdiggs@igalia.com

Please check the **errata** for any errors or issues reported since publication.

See also **translations**.

Copyright © 2015-2017 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and document use rules apply.

Findings

Link Behavior

When the user clicks on links within the document, some links open in a new tab while some open in the same browser window. This creates a situation where the user is not oriented to the action before it happens.

The key words **MAY**, **MUST**, **MUST NOT**, **OPTIONAL**, **RECOMMENDED**, **REQUIRED**, **SHALL**, **SHALL NOT**, and **SHOULD** are to be interpreted as described in [\[RFC2119\]](#).

This specification indicates whether a section is *normative* or *informative* and the classification applies to the entire section. A statement "This section is normative" or "This section is informative" applies to all sub-sections of that section.

Normative sections provide requirements that *user agents* must follow for an implementation to conform to this specification. The keywords **MUST**, **MUST NOT**, **REQUIRED**, **SHALL**, **SHALL NOT**, **SHOULD**, **RECOMMENDED**, **MAY**, and **OPTIONAL** in this document are to be interpreted as described in [Keywords for use in RFCs to indicate requirement levels \[RFC2119\]](#). RFC-2119 keywords are formatted in uppercase and contained in a **strong** element with `class="rfc2119"`. When the keywords shown above are used, but do not share this format, they do not convey formal information in the RFC 2119 sense, and are merely explanatory, i.e., informative. As much as possible, such usages are avoided in this specification.

Informative sections provide information useful to understanding the specification. Such sections may contain examples of recommended practice, but it is not required to follow such recommendations in order to conform to this specification.

Findings

Contact Information

The names under the editor section in each document are links. Some of these links open web sites, others open a new email addressed to that recipient and it is unclear which is which.



Findings

Outdated Versions

When the user clicks on the “previous version” link of any specification, it redirects the user to a different page as well as shows an error message that says “This version is outdated!” The user can get confused because they don’t know if the current page is outdated or the page before.



Findings

Consistency of Contents

In some specifications, the table of contents is on the left and others have the contents within the text. When the user clicks on a topic within the specification, they're taken to that section of the specification. But there is no sense of place as the table of contents is not consistently there with the user. It results in increased motor load as the user has to scroll to get back to the table.

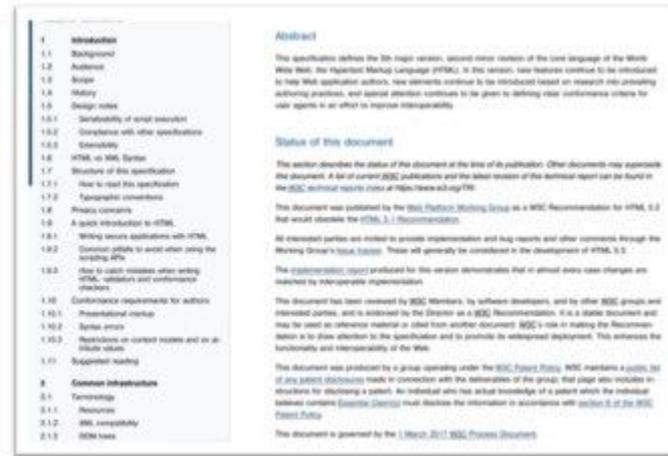
Table of Contents

- Abstract
- Status of This Document
- 1. Introduction
 - Excluded from Scope
 - Document Overview
 - Document Conventions
- 2. Key Terms
 - Accessibility Services of Platform Software
 - Content (on and off the Web)
 - Document
 - Set of Documents
 - Set of Software Programs
 - Software
 - User Agent
- 3. Closed Functionality
- 4. Text / Command-line / Terminal Applications and Interfaces
- 5. Comments on Conformance
- 6. Comments by Guideline and Success Criterion
 - Principle 1: Perceivable
 - Guideline 1.1: Text Alternatives
 - Success Criterion 1.1.1: Non-text Content (Level A)
 - Guideline 1.2: Time-based Media
 - Success Criterion 1.2.1: Audio-only and Video-only (Prerecorded) (Level A)
 - Success Criterion 1.2.2: Captions (Prerecorded) (Level A)
 - Success Criterion 1.2.3: Audio Description or Media Alternative (Prerecorded) (Level A)
 - Success Criterion 1.2.4: Captions (Live) (Level AA)
 - Success Criterion 1.2.5: Audio Description (Prerecorded) (Level AA)
 - Guideline 1.3: Adaptable
 - Success Criterion 1.3.1: Info and Relationships (Level A)
 - Success Criterion 1.3.2: Meaningful Sequence (Level A)
 - Success Criterion 1.3.3: Sensory Characteristics (Level A)

Findings

Availability of Contents

In some left side table of contents, all the topics are shown on the documents, while in some, only the boilerplate, abstract and status of the document is mentioned. The user has to click on the other topics to see that section. The experience is not consistent.



Findings

Search

Most of the document has a lot of text and, at times, the user needs to reach one section of the specification really fast. But there is no intuitive search option. This again leads to longer scrolling and excessive motor and mental load as they have to skim through a lot of content to get to where they want.

TTML defines a MIME type/subtype, `application/ttml+xml`, that may be used to identify the content type of TTML resources. In addition, TTML as well as other W3C and non-W3C specifications define a number of processor profiles which define requirements on compliant processors that may decode and process a TTML document. This registry can be used by other entities to exchange processor profiles in a compact way. In certain TTML use cases, it is desirable for a processor to proceed with TTML resource fetch, decode, and processing only if it can be determined that the referenced resource is tentatively processable. In order to satisfy such use cases, it is possible for the referencing context to enumerate one or more named profiles, which, if supported by the processor, would allow a first-order determination to be made about whether a resource may be processed. We say *first order* here since during actual decoding of a TTML resource, the processor profile declarations contained within the resource may result in the resource being rejected.

This registry is intended to provide a central location for enumerating identified TTML profiles, or, more strictly speaking, TTML decoder/processors, where each entry in the registry identifies a particular profile which is understood to implement a processor capable of satisfying the constraints of a defined TTML processor profile that takes the form of a TTML Profile Definition Document. By utilizing a common registry, it is possible to avoid name collisions among different profile defining fora.

Note Well that, in the context of this registry, when we use the term *profile*, we mean *processor profile*. We explicitly do not mean *content profile*. That is, nothing about the use of the `profile` parameter described here is intended to be used to identify or make claims about whether a TTML resource conforms with a TTML Content Profile or any type of TTML Profile that may be interpreted in whole or in part as making statements about the conformance of a TTML resource or the features of TTML (or other external specifications) actually used in the resource.

Findings

Flexibility and efficiency of use

- **Lack of Tools for Super users:** Sometimes, expert users want to search for some specific information within the document. Unfortunately, there are no quick tools to assist them in their search.
- **Lack of Between or On-Specification Search:** When the user wants to search for a certain specification for reference or search within a specification, it results in motor and mental load as they search and scroll to get to the desired information.
- **Lack of Tools for Code Snippets:** While users agree that code snippets are incredibly helpful, there is no consistent tool for easy use of the snippet.

Findings

View Selector

The other pages of the W3C site have the option to view the site on different devices (website/tablet/mobile). But when you go to the specification, this option is not available.

Views: desktop mobile print

STANDARDS | PARTICIPATE | MEMBERSHIP | ABOUT W3C

W3C » Standards » All Standards and Drafts

ALL STANDARDS AND DRAFTS

W3C is [teaming up](#) with Jefferson University [to improve the design of the specification template](#).
We would greatly appreciate your participation in [our survey](#), open until March 30, 2018, in order to better understand your n

Select sorted by

Show details Hide details

Findings

Link Styling

There are a lot of links in each specification. Often, these links will include the whole text instead of a hyperlink, which is not aesthetically pleasing.

Digital Publishing Accessibility API Mappings



W3C Recommendation 14 December 2017

This version:

<https://www.w3.org/TR/2017/REC-dpub-aam-1.0-20171214/>

Latest published version:

<https://www.w3.org/TR/dpub-aam-1.0/>

Latest editor's draft:

<https://w3c.github.io/dpub-aam/>

Implementation report:

<https://w3c.github.io/test-results/dpub-aam/>

Previous version:

<https://www.w3.org/TR/2017/PR-dpub-aam-1.0-20171102/>

Editors:

Richard Schwerdtfeger, Knowbility, richschwer@gmail.com
Joanmarie Diggs, Igalia, S.L., jdiggs@igalia.com

Please check the **errata** for any errors or issues reported since publication.

See also **translations**.

Copyright © 2015-2017 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and document use rules apply.

Findings

Print Layout

Some users might want to print a specification. But there is no print version of the specifications available.

The screenshot displays a document viewer interface. On the left, a 'TABLE OF CONTENTS' sidebar lists sections: Abstract, Status of This Document, 1. Introduction, 2. Conformance, 3. Important Terms, 4. Supporting Key..., 5. Mapping WAI-ARIA..., 5.1 General rules for semantics, 6. Conflicts between..., 6.1 Conflicts between..., 7. Exposing attributes..., 8. Role mapping, 8.1 General Rules, 8.2 Role Mapping T..., 9. State and Property Mapping, 10. Special Processing Requiring Additional... A 'Print' dialog box is open in the center, showing 'Total: 15 sheets of paper' and 'Cancel' and 'Print' buttons. Below this, it shows the destination 'Srishti N2 n N3 RICOH Aficio MP C3501 ...' with a 'Change...' button. The 'Pages' section has a radio button for 'All' and a text input 'e.g. 1-5, 8, 11-13'. The 'Copies' section has a text input '1'. The 'Layout' section has a dropdown menu set to 'Portrait'. The 'Colour' section has a dropdown menu set to 'Colour'. The 'Options' section has a checkbox for 'Two-sided'. Below the print dialog, there are links: '+ More settings', 'Print using system dialogue... (%P)', and 'Open PDF in Preview'. The main document area shows a 'Table of Contents' page with a list of sections: Abstract, Status of This Document, 1. Introduction, 2. Conformance, 3. Important Terms, 4. Supporting Keyboard Navigation, 5. Mapping WAI-ARIA to Accessibility APIs, 5.1 General rules for exposing WAI-ARIA semantics, 6. Conflicts between native markup semantics and WAI-ARIA, 7. Exposing attributes that do not directly map to accessibility API properties, 8. Role mapping, 8.1 General Rules, 8.2 Role Mapping Table, 9. State and Property Mapping, 10. Special Processing Requiring Additional Computation, 10.1 Name and Descriptions, 10.2 Widget Values, 10.3 Relations, 10.4 Group Position, 11. Actions, 12. Events, 13. Special Document Handling Procedures, 14. Appendix, 14.1 Substantive changes since the last public working draft, 14.2 Acknowledgments, 14.2.1 Participants active in the DPUW ARIA task force at the time of publication. At the bottom of the page, there is a section for '0993/2018 Digital Publishing, Accessibility API Mappings' with sub-sections: A.2.2 Enabling features, B. References, B.1 Normative references, B.2 Informative references, and 1. Introduction. A dark vertical bar on the right side of the document viewer contains a speech bubble with the number '2' and three circular navigation buttons (back, home, forward). To the right of the document viewer, there is a text block: 'not feasible, 1.0 Recom- in errata or ts but future -progress up- BC groups and document and Recommen- s enhances the ns a public list o includes in- e individual s of the W3C'.

As well as sections marked as non-normative, all authoring guidelines, diagrams, examples, and notes in this specification are non-normative. Everything else in this specification is normative.

Findings

Notes / Code Snippets / Tables

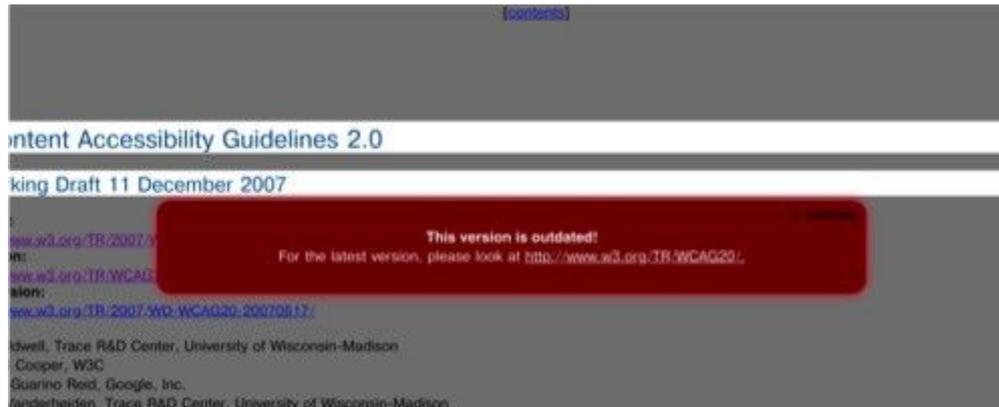
Users prefer to scan. It is usually better to present them with small, visually differentiated chunks of information. The user might find it difficult to read the whole document or go through the table of contents as there is lot of information without any breathing space. Additionally, the code snippets do not provide an improvement to the aesthetic design or achieve minimalism.

1.	Introduction	Operating systems and other platforms provide a set of interfaces that expose information about objects and events to assistive technologies . Assistive technologies use these interfaces to get information about and interact with those widgets . Examples of accessibility APIs are Microsoft Active Accessibility [MSAA], Microsoft User Interface Automation [UI-AUTOMATION], MSAA with UIA Express [UIA-EXPRESS], the Mac OS X Accessibility Protocol [AXAPI], the Linux/Unix Accessibility Toolkit [ATK] and Assistive Technology Service Provider Interface [AT-SPI], and Accessible2 [Accessible2].
2.	Conformance	
3.	Important Terms	
4.	Supporting Keyboard Navigation	Accessible Description An accessible description provides additional information, related to an interface element, that complements the accessible name . The accessible description might or might not be visually perceivable.
5.	Mapping WAI-ARIA to Accessibility APIs	Accessible Name The accessible name is the name of a user interface element. Each platform accessibility API provides the accessible name property. The value of the accessible name may be derived from a visible (e.g., the visible text on a button) or invisible (e.g., the text alternative that describes an icon) property of the user interface element. See related accessible description .
5.1	General rules for exposing WAI-ARIA semantics	A simple use for the accessible name property may be illustrated by an "OK" button. The text "OK" is the accessible name. When the button receives focus, assistive technologies may concatenate the platform's role description with the accessible name. For example, a screen reader may speak "push-button OK" or "OK button". The order of concatenation and specifics of the role description (e.g., "button", "push-button", "clickable button") are determined by platform accessibility APIs or assistive technologies .
6.	Conflicts between native markup semantics and WAI-ARIA	Assistive Technologies Hardware and/or software that: <ul style="list-style-type: none">• relies on services provided by a user agent to retrieve and render Web content• works with a user agent or web content itself through the use of APIs, and• provides services beyond those offered by the user agent to facilitate user interaction with web content by people with disabilities
7.	Exposing attributes that do not directly map to accessibility API properties	This definition may differ from that used in other documents. Examples of assistive technologies that are important in the context of this document include the following: <ul style="list-style-type: none">• screen magnifiers, which are used to enlarge and improve the visual readability of rendered text and images
8.	Role mapping	
8.1	General Rules	
8.2	Role Mapping Table	
9.	State and Property Mapping	
10.	Special Processing Requiring Additional Computation	
10.1	Name and Description	
10.2	Widget Values	
10.3	Relations	
10.4	Group Position	
11.	Accessibility	

Findings

Specification Status

When the user clicks on the “previous version” link of any specification, it redirects the user to a different page as well as shows an error message that says “This version is outdated!” The user can get confused because they don’t know if the current page is outdated or the page before.



Findings

Status of Draft

For a first time user, it's difficult to understand terminologies like “nightly draft,” “recommendation,” etc. Also, within the specifications, the non-intuitive search makes it harder to navigate in the document. It would be helpful if there was a guide to how to use the specification.

▶ Accessibility (All)		
More about Accessibility (All)		
Completed Work		
2017-12-14	Digital Publishing WAI-ARIA Module 1.0	Recommendation <i>Nightly Draft</i>
2017-12-14	Accessible Rich Internet Applications (WAI-ARIA) 1.1	Recommendation <i>Nightly Draft</i>
2017-12-14	Digital Publishing Accessibility API Mappings	Recommendation <i>Nightly Draft</i>
2017-12-14	Core Accessibility API Mappings 1.1	Recommendation <i>Nightly Draft</i>
2015-09-24	Authoring Tool Accessibility Guidelines (ATAG) 2.0	Recommendation
2014-03-20	WAI-ARIA 1.0 User Agent Implementation Guide	Recommendation
2014-03-20	Accessible Rich Internet Applications (WAI-ARIA) 1.0	Recommendation
2013-03-28	Role Attribute 1.0	Recommendation
2008-12-11	Web Content Accessibility Guidelines (WCAG) 2.0	Recommendation
2002-12-17	User Agent Accessibility Guidelines 1.0	Recommendation
2000-02-03	Authoring Tool Accessibility Guidelines 1.0	Recommendation
1999-05-05	Web Content Accessibility Guidelines 1.0	Recommendation
2017-12-14	WAI-ARIA Authoring Practices 1.1	Group Note <i>Nightly Draft</i>
2017-02-02	Pointer Methods in RDF 1.0	Group Note
2017-02-02	Representing Content in RDF 1.0	Group Note
2017-02-02	Developer Guide for Evaluation and Report Language (EARL) 1.0	Group Note

Surveys

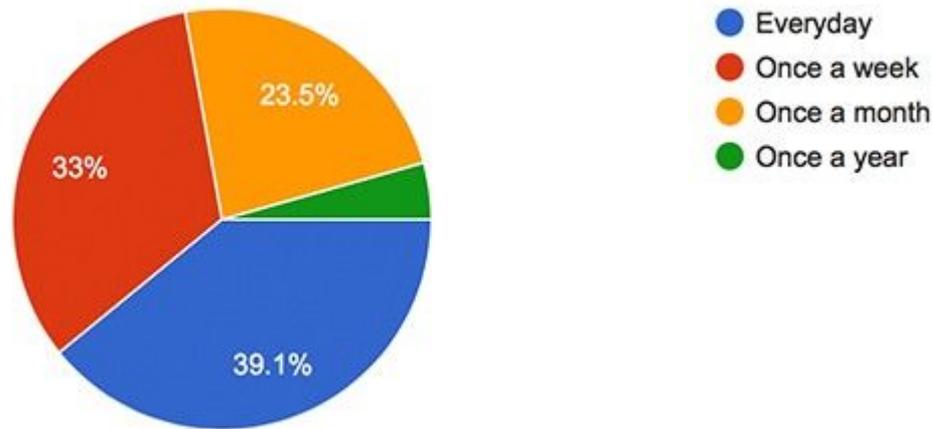
After conducting a round of surveys with the W3C community and affiliated members, we have come up with answers to some critical questions that will help us understand the user base and target audience and their interactions with W3C specifications. The survey was sent out to the W3C community, including developers from Google Chrome, Mozilla and other developers who interact with the W3C specifications.

The open-ended feedback that we received pointed to the need for improvement of navigation and readability. Respondents also asked for more examples and clear definitions to be added to be added to the specifications.

Survey Results

W3C Usage

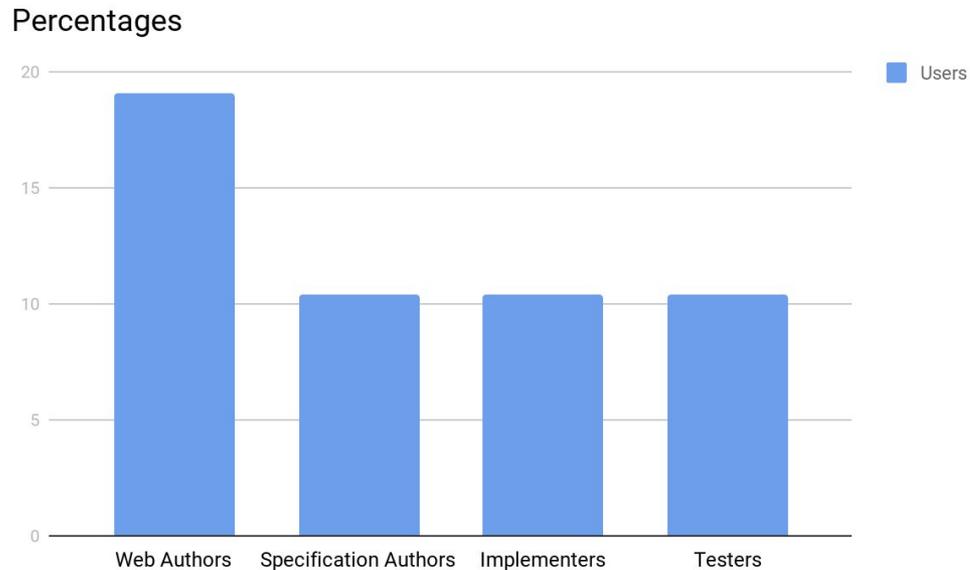
We have received more than 130 responses to the survey questionnaires. Of those, around 90% of the respondents had used W3C specifications at the time of the survey. Around 39% of respondents use the specifications on a daily basis, while 33% of respondents use them on a weekly basis, 23.5% of respondents use the specifications on a monthly basis and the rest on a yearly basis.



Survey Results

W3C User Demographics

The majority of the respondents fall into these four main categories:



Respondents prefer to work on the specifications using desktop or laptop computers, while some respondents prefer to view them on the mobile phones or tablets. Others print them for convenience.

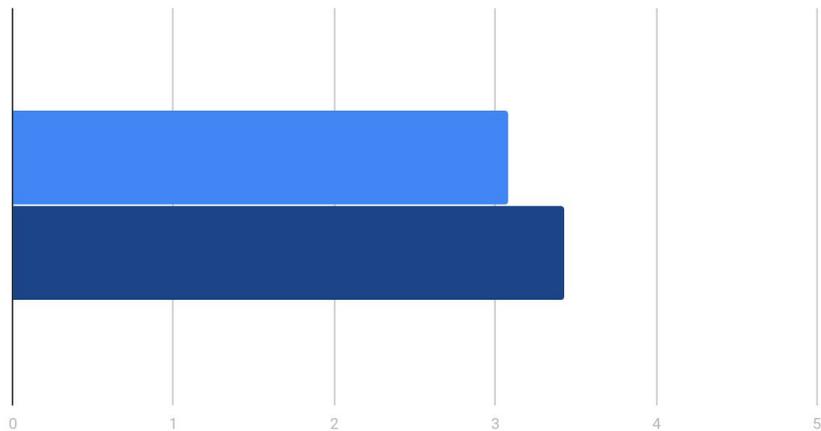
Survey Results

W3C Net Promoter Score

We asked respondents to rate their level of agreement with two statements. These statements serve to set a baseline for the existing specification to which we can compare future scores. With an average score of 3.08, respondents were basically neutral about the statement, “It’s easy to find the information I need.” With an average score of 3.43, respondents lean a little bit better than neutral on the statement, “I would recommend the specifications to a friend.”

Averages

■ It's easy to find the information I need. ■ I would recommend the specifications to a friend.



Interviews

After identifying the target demography of W3C specifications, we explored the issues from the perspective of the people who care most – W3C users.



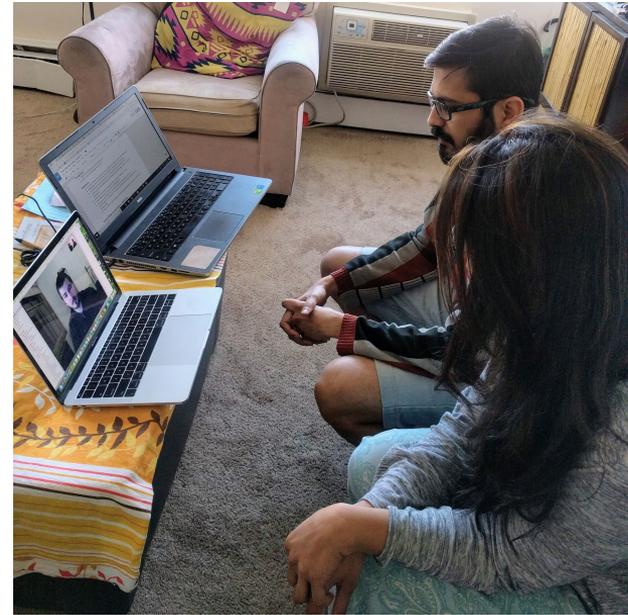
Interviews

To identify the complications of using a W3C specification, we listened to the users' experiences with it.

We wanted to identify their pain points.

Methodology

A general set of questions were structured that formed a common ground for the interview sessions, but we geared them to be of open-ended, casual conversations so our participants would feel comfortable enough to reveal anything important. All the interviews were online video sessions that lasted 45–60 minutes.



Problems

Participants talked about the nature of their affiliation with the W3C, usage patterns, media, purpose, frequency, search patterns, opinion on the current layout, usability, readability and also walked us through their use of specifications.

We analyzed each interview, pulling out key findings, and used an affinity exercise to find the common problem areas. To view our affinity diagram document, please [click here](#).

The most typical pain points were:

Navigation

Current navigation system is confusing and non-linear. Participants had trouble navigating within a single specification as well as between specifications. This problem is further complicated by the W3C tendency to include hyperlinks. Participants were inevitably unsure whether a hyperlink would lead them to an anchor tag, a new URL or launch their email client. As participants increased their reliance on the W3C, this problem became more pronounced. To view the analysis of the keyword mapping document, [click here](#).

Version Control

As we know, W3C specifications are constantly being revised. We found that it's not always clear to our participants which version they are looking at. Though they've seen those notifications that the version they are looking at is out of date, they're not sure what to do about it. This leads to confusion and redundancy for users.

Readability

Specifications consist of a lot of great content — text, code snippets, images, both internal and external references, acknowledgements, etc. Though there were plenty of comments about the content itself, we're interested in the impacts to usability. This bounty of content can, at times, create clutter that distracts the user from their end goal: finding the answer to a question and getting back to writing their code.

Interviews Conclusion

While revealing the key problem areas above, the affinity exercise lead us to begin modeling personas and journey maps. The interview sessions helped us connect with our target audience and empathize with their problems. Our research is a valuable resource that our team can refer to throughout the design process.



Stages of development of Specification

User goals	Group Copy/ Draft/Nightly/ Editor's Draft	Working Draft	Series of Working drafts	Candidate Recommendation	Proposed Recommendation	Recommendation
Touchpoints and emotional response	Writers working on their own copies on their site. Latest input from the writers. Important for advanced Users Github Repo	Published version		when good enough for implementation looking for feedback mainly from implementers	Mainly administrative	Standard Specification
Pain points						
Overall experience						

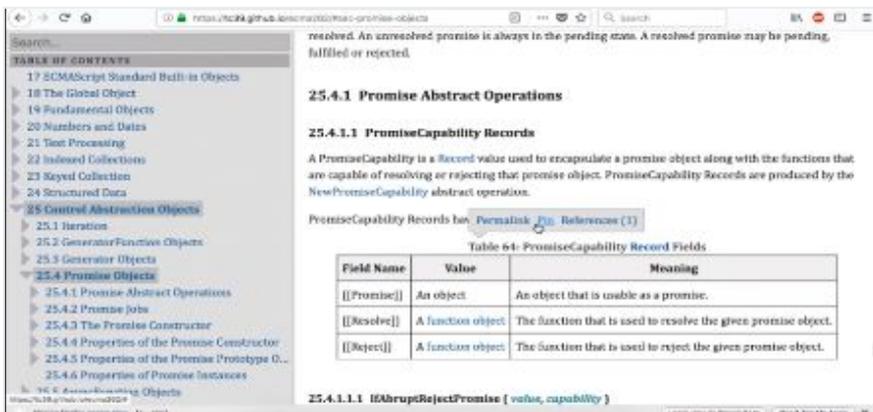
Competitor Analysis

We conducted a competitor analysis to identify the competitors and evaluate their strategies to determine their strengths and weaknesses relative to those of W3C's product or service.

Competitors

Who are the Major Competitors

- whatwg.org spec (foundational spec, Users generally depend on them a lot) Its the most widely used website
- DOM
- ECMASCRIPT
 - (useful but not super great in terms of layout)
 - Sidebar is useful
 - They have the fragment Identifier /pop up tooltips



resolved. An unresolved promise is always in the pending state. A resolved promise may be pending, fulfilled or rejected.

25.4.1 Promise Abstract Operations

25.4.1.1 PromiseCapability Records

A PromiseCapability is a *Record* value used to encapsulate a promise object along with the functions that are capable of resolving or rejecting that promise object. PromiseCapability Records are produced by the *NewPromiseCapability* abstract operation.

PromiseCapability Records has [Permalink](#) [Pop](#) [References \(1\)](#)

Table 64: PromiseCapability Record Fields

Field Name	Value	Meaning
[[Promise]]	An object	An object that is usable as a promise.
[[Resolve]]	A function object	The function that is used to resolve the given promise object.
[[Reject]]	A function object	The function that is used to reject the given promise object.

25.4.1.1.1 `[[AbruptRejectPromise]]` (*value, capability*)

Competitors

Who are the Major Competitors

- IETF
 - Usually just text
 - Started at 1950s (super interesting history)
 - We depend very heavily on them
 - Language test are defined by this spec
- ISO

Users depend very heavily on normative references

The screenshot shows a web browser window with the URL <https://www.w3.org/TR/payment-request/#references>. The page is titled "C. References" and contains a list of normative references. The left sidebar shows a table of contents with sections like "Security Considerations", "Privacy Considerations", "Dependencies", "Conformance", "IDL Index", "Acknowledgements", "References", and "Normative references". The main content area lists several normative references, including BCP47, DOM, E.164, i18n-402, ECMAScript, HTML, and INFra.

C. References

C.1 Normative references

[BCP47]
Tags for Identifying Languages. A. Phillips; M. Davis. IETF. September 2009. IETF Best Current Practice. URL: <https://tools.ietf.org/html/bcp47>

[DOM]
DOM Standard. Anne van Kesteren. WHATWG. Living Standard. URL: <https://dom.spec.whatwg.org/>

[E.164]
The international public telecommunication numbering plan. ITU-T. November 2010. Recommendation. URL: https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-E.164-201011-!!PDF-E&type=items

[i18n-402]
ECMAScript Internationalization API (Specification). Ecma International. URL: <https://tc39.github.io/i18n-402/>

[ECMAScript]
ECMAScript Language Specification. Ecma International. URL: <https://tc39.github.io/ecma262/>

[HTML]
HTML Standard. Anne van Kesteren; Domenic Denicola; Ian Hickson; Philip Jägenstedt; Simon Pieters. WHATWG. Living Standard. URL: <https://html.spec.whatwg.org/hu/!page/>

[INFra]
Infra Standard. Anne van Kesteren; Domenic Denicola. WHATWG. Living Standard. URL: <https://infra.spec.whatwg.org/>

Analysis

Why do Users' use these other Specifications

- Functionality
- Previous mentioned boilerplate stuff are gone
- Really nice clean version
- Test link quite usable
- Rest are pretty same



Analysis

Fragment Identifier/Popup tooltips :

- A great feature
- When you click on a subject it tells where everything is in the sections
- Takes you to the actual reference
- Both definitive link also
- The context in which it gets used as well
- Takes you to the other connected links, this is like super useful
- Generally you need to copy that URL to use it



A screenshot of a technical document, likely a specification, with a tooltip displayed over a section. The document text includes:

- A **URL** is local if its scheme is a local scheme.
- Note** This definition is also used by [Retriever Policy](#), [RUFWRREQ](#).
- An **HTTP(S) scheme** is "http" or "https".
- A **network scheme** is "ftp", "mailto", "tel", "urn", or an HTTP(S) scheme.
- A **fetch scheme** is a "blob", "data", "file", "filesystem", or a network scheme.
- Note** HTTP(S) scheme, network scheme, and fetch scheme are also used by [HTML](#), [HTML](#).
- A **response URL** is a URL, for which implementations need not store the fragment as it is never exposed. When serialized, the exclude fragment flag is set, meaning implementations can store the fragment nonetheless.

2.2. HTTP

While **fetching** encompasses more than just HTTP; it borrows a number of concepts from HTTP and applies these to resources obtained via other means (e.g., data URLs).

The **HTTP whitespace bytes** are 0x09, 0x0A, 0x0D, and 0x20.

File an issue about the selected text.

Working Draft, Working Draft, 12, 2021

Copyright © 2021 W3C. All Rights Reserved. W3C, 2021. All Rights Reserved.

Analysis

Section/Paragraph markers

- User describe it as cute
- Gives them visual queues

Most annoying thing about these specs

- They are printer hostile
- To demonstrate printer walkthrough user faces problem with firefox browser and switches to Safari
- It used to be like each section was broken into a page, so you would end up with a lot of pages to print

Design Process

The Problem

Design Process

Next Steps

Personas

Journey Mapping

Wireframes

Visual Design

Style Guide

User Testing Findings

Personas

Once we generated enough data from the interviews and identified problems with affinity clustering, it was important to wrap those problems around a personality. Creating a personality is quite helpful, it gives us the opportunity to connect the needs to a face and also motivates us to solve a problem that is much more user-focused.

Before building a persona, it is important to identify the key goals and pain points of the user. This information is essential to determine what approach should be taken to address the recognized issues.

Identification

Affinity clustering was an immensely helpful exercise we did to group ideas together and search for patterns based on the the interviews conducted. It was important to further narrow it down and prioritize the problems. To do this, we categorized the users of the specs into two groups based on usage behavior.

The first was the web authors who use the specifications to create their own digital products and the other was specification creators, implementers, and testers who work closely to create W3C specifications. We then drew a chart to classify users navigation, search and, reading behavior on the pages.

Identification

Another approach was to note how many times a certain keyword was mentioned. For instance, queries similar to “search”, “navigation”, “github”, “boilerplate”, etc. We called it the “keyword usage sheet. This guided us into making observations about very specific user challenges.

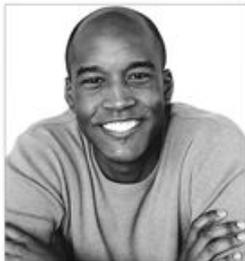
NAVIGATION 4 Times				
A	B	C	D	E
NAVIGATION 4 Times	SEARCH 23 Times	VERSIONS 11 Times	LINKS 10 times	READ 13 Times
Another difficulty is the referencing. A page has a link to a different page. Navigation becomes difficult.	Visual presentation seems ok. It works for me as I can search and read text easily.	Well versed with the existing system. Difficulties while knowing the version of the specs.	Difficulties in figuring out the context based on external links is annoying.	Visual presentation seems ok. It works for me as I can search and read text easily.
Navigation of multiple pages becomes confusing.	Quick Search	If I look at Github I know I've got a fairly decent version.	The links direct to other website and the information is hard to relate it back to the original reference.	Having examples helps guiding the spec reading and also to sketch basic concept of how to use.
There is no indication of the different types of link and navigation becomes difficult.	I designed Respec tool search other ref/spec,definition,export.	A ref test it will be good to have an authoring mode to compare different versions of HTML and CSS, so you can iterate and work.	More clarity of links inside any spec. Disruptive.	Proof reading by screen reader. Highlight text then use screen reader.
There is no indication of the different types of link and navigation becomes difficult.	Searching for media events. (Captions or subtitles)	I use Github to get latest versions. I use Unicode and BSI (British Standards Institute).	Popup Tooltips. Reference. Definitive link. context. Other connected links.	Her developers enjoy reading specs in other sites but she always tells them to come back to W3C because that's the best.
	Browser search	Availability of visual cues is very important to identify versions at any part of the document.	I use MDN. Cross links to other specs.	Font is readable.
	Remember keywords to search quickly.	There is no indication of the versions if you scroll down. It's only at the top.	Opens links in different pages while content page is open in a different tab.	So I am trying to make the examples more readable. Some use very bizarre, complicated languages.
	Uses browser search.	Searching on Google is a problem cause it has all the old versions as well so it's confusing.	Some websites use icons if its going externally for differentiating links.	Reads specs and compare them.
	I think searching it is the most complex thing. I don't think there is a search function.	Uses older versions of standards. CSS to HTML. CSS to XML.	Iconography will be useful to classify links.	W3C too much of text but easily readable.
	Prefer to search in browser not in PDF.	In built search will reduce the issue of coming across older versions.	Internal or exterior links distinction should be there.	Depends on length of specs. Eg. HTML is long and hard to read through.
	It is important to have search function within specs.	Specifications evolve over time but we keep the old versions so people can access it. Which confuses people.	There may be better ways of showing the information of links indicating without excessive hyperlinks.	I tend to not pay attention to the abstract. status part, all the meta-data. Except if it is a full read through.
	Uses quick search on browser.	I don't care about previous versions or editors or status of the document.		Every once in a while I'll get a link on phone and to read that never works.
	I use specs all the time, I don't search. I have tons of bookmarks about this stuff			If I already don't have a link to a document, it takes me a while to figure out which CSS spec is going to have the info I need.

Integration

Once we had enough data and the keyword usage chart, we started narrowing information for the persona. The information that goes on to the persona is crucial. The solution that will be created has to fulfill all the user requirements. We created two personas, one for each group. Each group has a different set of requirements and expectations. We identified those differences and created the personas.

The information was split into different identifiable sections:

1. Background/story – It is a good stepping stone to understand the user
2. Motivations – The reasons behind using W3C specifications
3. Goals – Their expected goals
4. Pain points – Difficulties they face
5. Search/reading behavior – How do they search specifications
6. Our Recommendations – A way to improve the overall experience



All rounder Harry

Invited expert at W3C



As a simultaneous implementer and tester, the nature of my work involves both looking for a specific section in the specifications and pursuing the whole specification line by line. Review and reference specifications for various web standards.

Intent on generating content for W3C

Contribute building up a platform where web authors would find a standard set of information to create websites and achieve their goals. It is crucial to consider how each set of information is defined and being consumed by web authors. I intend to create structured data which can be used for reference and a tool/guideline for users.

Usage goals

- To make the content accurate (technically and grammatically).
- Review and reference appropriate content for specific sections.

Pain points

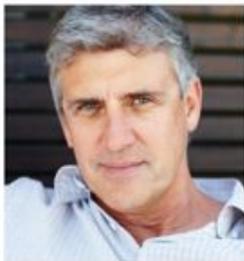
- As an advanced user, landing on the W3C TR page is annoying, generally, I look for Editor's Drafts.
- The Boilerplate section at the top is quite overwhelming. Some content is redundant for me.
- Scrolling to the top of the page, again and again, to report a bug or have access to the test suit is exhausting.
- Unfavorable print version.

Search/reading behavior

- Bookmark a whole bunch of specifications that I use.

Recommendation

- I should get all the information upfront, instead of a huge boilerplate/metadata.
- Version control, and indicating which version am I viewing more prominently.
- Reporting a bug could be intuitive and easy from any point of the webpage.



Broadcaster Chris

Principle software engineer



My primary role is to code and find intuitive ways to make the broadcasting of shows much more accessible, for eg. A media query which will help me use subtitles. It is a tough task as finding for the exact information becomes hard on the vast information database on W3C. I also need to consider how to make media accessible. As this job is very demanding, I hope not to waste much time on searching; and rather spend time on making company's product.

Intent on using W3C

W3C provides me required guidelines on how I should be developing my product or tools for consumers. Snippet codes and descriptive information about an idea is explained elaborately, which helps me in my development process. W3C is my to go standards apart from the broadcasting (ETSI) and accessibility standards (British standards for web accessibility) which I refer.

Usage goals

- Using this guidelines for checking if websites and applications are following the guidelines and suggest appropriate actions.
- Using the application to define media and entertainment solutions for streaming, subtitles etc.

Pain points

- Which version of specification I am referring.
- When I start reading, I want content upfront and not the metadata and other links.

Search/reading behavior

- The internal website for quick references (Quick links)
- Browser search on W3C website
- Using the table of content.

Recommendation

- A robust query searchability on W3C website, replacing browser search.
- Version control, and indicating which version am I viewing more prominently.

Journey Mapping

Journey maps are created to understand experience from the user's perspective. Once the map is created, it is easy to identify the pain points of the user throughout their journey of completing a task.

Identification

Mapping the user journey by considering what are the user goals and how they access the specifications, what are their search methods, and other reading behaviors, surfaced some very minute details of their experience. Defining the story also provided a better overview of all the problems and frustrations.

We started with simply recategorizing all responses received in interviews into a very linear flow starting from how they search, access, read, and use a specification.

Integration

As we proceeded with defining user journeys, we mapped out the different users and broadly categorized their responses under “Search”, “Read/Locate”, and “Use” of the specifications. This then led to a further breakdown into details of their actions performed at each stage, which included their ways of navigation, how they searched for the required section within the document, understanding their difficulties in identifying the versions of specifications, different links, etc.

The analysis of the process allowed us to identify specific problems and propose solutions to solve a section within the flow. You can find a detailed journey map in the next slide.



Web Authors General Journey Map:

This group comprises of front-end developers, web designers, software developers or anyone creating content for the web.

	SEARCH			READ/LOCATE		USE
User goals	Accessing W3C specifications	Search for required spec/ specs	Finding version / other requirements	Searching sections within documentation	Reading the document	Using for authoring (Undecided)
Touchpoints and actions performed	<ul style="list-style-type: none">- Github- Google search sometimes give results for old versions- Bookmarks- Referring to other documentation MDN	<ul style="list-style-type: none">- No one uses the search function on W3C website- Navigation is difficult- Searching is complex	<ul style="list-style-type: none">- Uses quick search on browser- Difficulties while knowing versions of specs	<ul style="list-style-type: none">- A page has a link to a different page. Navigation becomes difficult.- Using quick browser search- Figuring out context based on external links is difficult- Not easy to switch between specific sections within the document- Table of contents is useful	<ul style="list-style-type: none">- Links redirect- Description is terse- Trying to understand spec is a little difficult because each spec is written by someone different.	<ul style="list-style-type: none">• Having examples helps guiding the spec reading and also to sketch basic concept of how to use.• Some websites use icons if its going externally for differentiating links.• I usually open multiple tabs go to Github or make notes.• Auditing of websites and applications.• Sends clients links of specs to inform them how to implement.• Uses links to the advisory techniques.• Lacks examples in some section.
Pain points	<ul style="list-style-type: none">- Do not fully rely on W3C- Bookmarking is useful for frequently used specs	<ul style="list-style-type: none">- Navigation and search functionality is not upto mark- It is time consuming to look for the exact spec required	<ul style="list-style-type: none">- Knowing some high level information like versions, authors, is difficult.- Reading through specs and not knowing some key information may lead to irritation	<ul style="list-style-type: none">- Links are not differentiated by colours or icons so there is a constant back and forth from the spec document to evaluate where it directs.- The only option to search is by using quick browser search.	<ul style="list-style-type: none">- Clarity of information is missing at times- No consistent style and colours- Multiple authors make the content inconsistent- No indication of differences between links	
Ideas to improves	<ul style="list-style-type: none">- Github can be referred to understand why it is more reliable sometimes.- SEO improvement will lead to better search results- A way to bookmark within the W3C specs can improve in quick access	<ul style="list-style-type: none">- The search experience can be improved by adding features that will allow users for search results based on key words or other specific criteria.	<ul style="list-style-type: none">- An overview of each documentation should include versions, labeling index used for links, all authors who have worked.- This will help user locate what they are looking for.	<ul style="list-style-type: none">- Availability of visual cues is very important to identify versions at any part of the document.- Creation of a style guide and pattern library will largely help in improving consistency.	<ul style="list-style-type: none">- Definitions are precise. Example code is nice to have examples in context.- It would be nice to see diagrams and/or structure of things. (Media Source Execution)	



Implementers General Journey Map:

The users in this group are generally using the specifications to implement the specification in their web browser. Based on what is written in the specification, implementers can ensure their browser will read that code.

	SEARCH			READ/LOCATE		IMPLEMENT
User goals	Accessing W3C specifications	Search for required spec/ specs	Finding version / other requirements	Reading the document	Searching sections within documentation	Using for implementation
Touchpoints and emotional response	<ul style="list-style-type: none">- Github- Type in from Memory- Google- Bookmark (within the browser)- W3C TR page Feel comfortable because of being habituated to accessing in certain ways		<p>Latest Editors Draft</p> <p>It's difficult to find versions and refer to the accurate draft</p>	<ul style="list-style-type: none">- User like having the sidebar.- Algorithm layout are pretty good.- Like spacing Between each one of the steps.- Blue bar helps when you have nested algorithms	<ul style="list-style-type: none">- No search Field- Search By memory	
Pain points	<ul style="list-style-type: none">- Unwanted step- No Hierarchy of information in boilerplate.		<p>Some specifications miss the link.</p> <p>Very Important as a advanced User</p>			



Testers General Journey Map:

This group comprises quality assurance specialists who test the specifications and sample code to report and assure compliance with current web standards and best practices.

	SEARCH			READ/LOCATE		IMPLEMENT
User goals	Track down Id mentioned	To Track user use print version	Peruse	Test Suit Link	Latest Editors Draft	Test Suit
Touchpoints and emotional response		No designed print version available.	User manually cross through line by line to track and put assertions.	User has to scroll up to the top everytime.	User created own infrastructure in the editors draft.	
Pain points						

Wireframes

In this process of redesign, we have started wireframing and proposing different ways to tackle some of the problems discovered. Wireframing is an important step in any screen-based design process. It will allow us to define an information hierarchy and roadmap of functionality.

In order to get a clearer picture of the different ideas we were considering, we divided our class into two teams and approached the problem in different ways. This gave us the opportunity to test alternative design decisions.

The first team was comprised of Sarika Joglekar and Pratik Joglekar. The second team was comprised of Ishita Ferdousi and Raeesha Alteef.

Sarika Joglekar & Pratik Joglekar

Designed a classic approach and tested the following:

- The navigation at top with drop-down to reveal version history
- Open table of contents with visual mark to locate section
- Use of icons to indicate link function within a document
- Need of global search for searching specifications
- Use of category tags w/ specification name

To check out the live wireframe, [click here](#).

We did a first round of usability testing with two users. We asked about their first impression of the visual style, layout and iconography. Both participants found the design comprehensible and clean and the layout, colors, contrast effective. We then gave them certain scenarios and asked them to think aloud about their experience.

Sarika Joglekar & Pratik Joglekar

W3C All Standards and Drafts

Title Tag Status Version ?

W3C >> STANDARDS >> ALL STANDARDS AND DRAFTS

ARIA in HTML Accessibility

W3C Working Draft 10 March 2018

[More info](#) [GitHub w3c/wpub](#) [File a bug](#) [Commit history](#) [Pull requests](#)

TABLE OF CONTENTS

- 1. Web developer requirements for use of ARIA in HTML
- 2. Document conformance requirements for use of ARIA
- 3. Case requirements for ARIA role,
- 4. Allowed ARIA roles, states and
- 5. Requirements for implementers
 - 5.1 Conformance Checker implementers
- 6. Conformance Requirements
- A. References
 - A.1 Normative references
 - A.1 Informative references

Abstract

This specification defines the web developer rules (author conformance requirements) for the use of **[wai-aria-1.1]** and **[dpub-aria-1.0]** attributes on **[HTML52]** elements. It also defines requirements for Conformance Checking tools.

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](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.

ARIA in HTML is a **[HTML52]** specification module. Any HTML features, conformance requirements, or terms that this specification module makes reference to, but does not explicitly define, are defined in the **[HTML52]** specification.

NOTE

Feedback and bug reports are welcome. [File a bug](#)

This is a draft document and its contents are subject to change without notice.

This document was published by the [Web Platform Working Group](#) as a Working Draft. This document is intended to become a W3C Recommendation. Comments regarding this document are welcome. Please send them to public-html@w3.org ([subscribe](#), [archives](#)).

Sarika Joglekar & Pratik Joglekar

We were interested in insights on certain key actions like accessing Github to file a bug, switching between different versions of the specifications, and searching for other related specifications. Our testing gave some great insights about what the user needs. Some key findings from this testing revealed as listed :

- Global search: Global search will work when the user knows the parameters. The header is taking up a lot of space leaving less room for the documentation itself.
- Links (Icons) within spec: The links accompanied with icons was breaking the flow of reading and hence was increasing cognitive load.
- Table of contents highlights to locate section: Our participants recommended we take a look at specification that has an absurdly long table of contents, just to make sure the idea works for all specs.
- Icons on the top (Header): The icons at the top were well received and they specifically liked the 'File a bug' icon but found 'More info' a little confusing, since it was not a call to action.
- Changing version from drop down: It was a little difficult to figure out that the drop down offers different versions, hence it would help to write what action to expect right at the top.

Sarika Joglekar & Pratik Joglekar

We iterated on the recommendations and came up with an improved set of wireframes and sent it out to our testers again. One of the concerns pointed out was of responsive design and how the on hover links will work on mobile. Most of the other improvements we did were well received.

One of the comments for this version was *“Love the dropdown for version. Also like the collapsable side menu. My only concern is with the hover text. I often browse the spec on my phone – so I don’t have a mouse pointer. How does it work in that example?”*

Please view the next slide for the iterated version.

Sarika Joglekar & Pratik Joglekar

 All Standards and Drafts
W3C >> STANDARDS >> ALL STANDARDS AND DRAFTS

Title Tag

ARIA in HTML

Accessibility

Version : W3C Working Draft 10 March 2018

 File a bug  GitHub  Commit history  Pull requests  Authors

TABLE OF CONTENTS

- 1. Web developer requirements for use of ARIA in HTML
- 2. Document conformance requirements for use of ARIA
- 3. Case requirements for ARIA role,
- 4. Allowed ARIA roles, states and
- 5. Requirements for implementers
 - 5.1 Conformance Checker implementers
- 6. Conformance Requirements
- A References
 - A.1 Normative references
 - A.1 Informative references

Abstract

This specification defines the web developer rules (author conformance requirements) for the use of **[wai-aria-1.11 and ldub-aria-1.0]** attributes on **[HTML52]** elements. It also defines requirements for Conformance checking tools.

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](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.

W3C Patent Policy
5 February 2004 (updated 1 August 2017)

ARIA in HTML is a **[HTML52]** specification module. Any HTML features, conformance requirements, or terminology that this specification makes reference to, but does not explicitly define, are defined in the **[HTML52]** specification.

NOTE

Feedback and bug reports are welcome.  File a bug

This is a draft document and its contents are subject to change without notice.

Ishita Ferdousi & Raeesha Altaf

Designed with a modern approach and tested the following:

- Use tabs for nav, putting boilerplate and history on tabs, also testing the toolkit on right rail
- Use an accordian for the table of contents
- Collapsible table of contents

We conducted user testing with three users who are working with the W3C. We showed them our wireframes and asked what they thought on the overall layout, navigation between different versions of the same specifications, boilerplate content, table of content and tool buttons.

To check out the wireframes, please [click here](#).

One of the comments for this version was *“The buttons are fantastic. We can argue what buttons we need or not. But, I love the idea.”*

Ishita Ferdousi & Raeesha Altaf

Some of the insights that we got are as follows:

- Overall layout: First impression is really good. Things are more structured than before and it was time to get those icons on the specs. But check for long titles and work more on the real estate of the page.
- Version: We rightfully managed to address the problem regarding the versions. But think about including all the other available versions and give the users cue to let them know that other versions are available.
- Boilerplate content: Users like the tab idea. They understood that tabs are organized and put together according to the nature of the meta data.
- Table of contents: Content at a glance was appreciated by the users. The idea of scrolling and the visual cue following you at the table content would be very nice. But the triangle doesn't feel like a cue to collapse and open the table of content. Also, how can we see the whole content whenever we want?
- Tool Button (icons on the right): Users love the icons. Users believed that it was high time that those icons were put in the specs for a more visual experience.

Ishita Ferdousi & Raeesha Altaf

W3C Home > All Standards and Drafts > HTML 5.3

HTML 5.3

▼ TABLE OF CONTENTS

- ▼ 1. Introduction
 - 1.1. Background
 - 1.2. Audience
 - 1.3. Scope
 - 1.4. History
 - ▼ 1.5. Design Notes
 - 1.5.1. Serialisability of script execution
 - 1.5.2. Compliance with other Specifications
 - 1.5.3. Extensibility
- ▶ 2. Common Infrastructure
- ▶ 3. Semantics, Structure and APIs of HTML Documents
- ▶ 4. The Elements of HTML
- ▶ 5. User Interaction
- ▶ 6. Loading Webpages
- ▶ 7. Web Application APIs
- ▶ 8. The HTML Syntax
- ▶ 9. The XML Syntax
- ▶ 10. Rendering
- ▶ 11. Obsolete Features
- ▶ 12. IANA considerations

1. Introduction

1.1. Background

This section is non-normative.

HTML is the World Wide Web's core markup language. Originally, HTML was primarily designed as a language for semantically describing scientific documents. Its general design, however, has enabled it to be adapted, over the subsequent years, to describe a number of other types of documents and even applications.

1.2. Audience

This section is non-normative.

This specification is intended for authors of documents and scripts that use the features defined in this specification, implementors of tools that operate on pages that use the features defined in this specification, and individuals wishing to establish the correctness of documents or implementations with respect to the requirements of this specification.

This document is probably not suited to readers who do not already have at least a passing familiarity with Web technologies, as in places it sacrifices clarity for precision, and brevity for completeness. More approachable tutorials and authoring guides can provide a gentler introduction to the topic.

In particular, familiarity with the basics of DOM is necessary for a complete understanding of some of the more technical parts of this specification. An understanding of Web IDL, HTTP, XML, Unicode, character encodings, JavaScript, and CSS will also be helpful in places but is not essential.

1.3. Scope

This section is non-normative.

This specification is limited to providing a semantic-level markup language and associated semantic-level scripting APIs for authoring accessible pages on the Web ranging from static documents to dynamic applications.

Style Guide

A style guide is a set of standards for the writing and design of documents, either for general use or for a specific publication, organization, or field. A style guide establishes and enforces style to improve communication.

Our team member, Sarika Joglekar has come up with a style guide that uses the aesthetic appeal for this project. To view the style guide document, please [click here](#).

H1 ROBOTO 30PT BOLD

This is an example headline.

H2 ROBOTO 24PT BOLD

This is an example of a headline.

H3 ROBOTO 16PT BOLD

This is an example of headline text.

H4 ROBOTO 14PT BOLD

This is an example of headline text.

BODY TEXT ROBOTO 14PT | LINE HEIGHT 25 | REGULAR

This is an example of body text. W3C standards define an Open Web Platform for application development that has the unprecedented potential to enable developers to build rich interactive experiences, powered by vast data stores, that are available on any device. Although the boundaries of the platform continue to evolve, industry leaders speak nearly in unison about how HTML5 will be the cornerstone for this platform. But the full strength of the platform relies on many more technologies that W3C and its partners are creating, including CSS, SVG, WOFF, the Semantic Web stack, XML, and a variety of APIs.

SIDEBAR HEADER TEXT ROBOTO 18 PT REGULAR

Table of contents : 14PT Medium

SMALL SECTION LABEL TEXT ROBOTO 12PT BOLD UPPERCASE

Tool tips, definitions and other descriptive labels within document.

PRIMARY COLORS

#005A9C



#DCECF7



#FFFFFF



#F7F7F7



#101010



#707070



SECONDARY COLORS

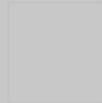
#727272



#BABABA



#BABABA



#BABABA



#BABABA



#BABABA



TO BE COMPLETED

TABLE OF CONTENTS

TABLE OF CONTENTS

1. Introduction

1.1 Context Extensions

2. Terminology

3. Namespaces

4. Conformance

5. Vocabulary Definition

5.1 Overview

5.2 Classes

5.2.1 [Thing](#)

5.2.2 [Interaction Pattern](#)

5.2.3 [Property](#)

5.2.4 [Action](#)

5.2.5 [Event](#)

5.2.6 [DataSchema](#)

5.2.7 [Form](#)

5.2.8 [Link](#)

5.2.9 [Security](#)

6. JSON-LD Serialization

6.1 Thing as a Whole

6.2 Interaction Resources

6.2.1 Property

6.2.2 Action

6.2.3 Event

GLOBAL SEARCH



ON CLICK

LOGO, TITLE AND BREADCRUMBS



W3C >> STANDARDS >> ALL STANDARDS AND DRAFTS

DROP DOWN

This Version : W3C Working Draft 10 March 2018 ▾

Latest published version

Latest editor's draft

All other versions

This Version : W3C Working Draft 10 March 2018 ▾

Latest published version

Latest editor's draft

All other versions

ON HOVER

ICONS



File a bug



Pull requests



Commit history



Github

DOCUMENTATION HEADERS AND BODY TEXT

1. This headline is Roboto 24pt Regular | Line height 34pt

This text is Roboto 14pc Regular| Line height 25pt | Paragraph width 750px.

Development that has the unprecedented potential to enable developers to build rich interactive experiences, powered by vast data stores, that are available on any device. Although the boundaries of the platform continue to evolve, industry leaders speak nearly in unison about how HTML5 will be the cornerstone for this platform. But the full strength of the platform relies on many more technologies that W3C and its partners are creating, including CSS, SVG, WOFF, the Semantic Web stack, XML, and a variety of APIs.

ICONS WITHIN BODY TEXT

-  **More information**
-  **File a bug**
-  **External link**
-  **Email**

NOTE HIGHLIGHTS

NOTE

Feedback and bug reports are welcome.  [File a bug](#)

This is a draft document and its contents are subject to change without notice.

Same page links

DOCUMENTATION LINK TYPES

External page different standard

Development that has the unprecedented potential to enable developers to build rich **interactive experiences**, powered by vast data stores, that are available on any device. Although the boundaries of the platform continue to evolve, industry leaders speak nearly in unison about how HTML5 will be the cornerstone for this platform. But the full strength of the platform relies on many more technologies that W3C and its partners are creating, including [CSS], SVG, WOFF, the Semantic Web stack, [XML], and a variety of APIs.

Full forms

Definition

This document was published by the [Web Platform Working Group](#) as a Working Draft. This document is intended to become a W3C Recommendation. Comments regarding this document are welcome. Please send them to public-html@w3.org ([subscribe](#), [archives](#)).

Open email

TOOL TIPS ON HOVER ON WEB OR LONG PRESS ON MOBILE

W3C technical reports index



 Mailing list of the W3C HTML Working Group and Web Platform WG

ARIA



Accessible rich internet applications

Visual Design

We aim to improve the user experience through considering the effects of illustrations, typography, space, layouts, and color on the usability of the specification pages. To do so, we have iterated on visual designs for different platforms to help achieve this.

By considering how we can form or arrange visual elements to address the principles of good visual designs, we have shaped the user experience in order to elicit user responses and behaviors that suit the use and purpose of the specifications. Inconspicuous, small details of a product's aesthetics can thus play a significant role in the design of the user experience.

The Desktop Version

To view the full resolution desktop version designs, please click here.

The screenshot displays the W3C website interface for the document 'Identifiers for WebRTC's Statistics API'. The top navigation bar includes the W3C logo, a breadcrumb trail (W3C > Standards > All Standards and Drafts > Identifiers for WebRTC's Statistics API), a search icon, and a version dropdown menu set to 'Working Draft 28 March 2018'. The main content area is divided into three sections: '1. Introduction', '2. Conformance', and '3. Terminology'. The 'Introduction' section contains two paragraphs of text. The 'Conformance' section contains two paragraphs of text. The 'Terminology' section contains three paragraphs of text. A sidebar on the left provides a 'TABLE OF CONTENT' with links to various sections. A right-hand sidebar contains four utility icons: a search icon, a bug report icon, a GitHub icon, and a commit history icon.

W3C > Standards > All Standards and Drafts > Identifiers for WebRTC's Statistics API

Identifiers for WebRTC's Statistics API MEDIA WEB API VERSION Working Draft 28 March 2018

TABLE OF CONTENT

- Contributors +
- Meta Data +
- Reference +
- 1. Introduction**
- 2. Conformance
- 3. Terminology
- 4. Basic concepts
 - 4.1 Guidelines for design of stats objects
 - 4.2 Guidelines for implementing stats objects
 - 4.3 Lifetime considerations for monitored objects
 - 4.4 Guidelines for getStats() results caching/throttling
- 5. Maintenance procedures for stats object types
 - 5.1 Adding new stats objects
 - 5.2 Retiring stats objects
- 6. RTCStatsType
 - 6.1 RTCStatsType enum
- 7. Stats dictionaries
 - 7.1 The RTP statistics hierarchy
 - 7.2 RTCTrtpStreamStats dictionary
 - 7.3 RTCCodecStats dictionary
 - 7.3.1 RTCCodecType enum

1. Introduction

This section is non-normative.

Audio, video, or data packets transmitted over a peer-connection can be lost, and experience varying amounts of network delay. A web application implementing WebRTC expects to monitor the performance of the underlying network and media pipeline.

This document defines the statistic identifiers used by the web application to extract metrics from the user agent.

2. Conformance

As well as sections marked as non-normative, all authoring guidelines, diagrams, examples, and notes in this specification are non-normative. Everything else in this specification is normative.

The key words may, must, and must not are to be interpreted as described in [RFC2119].

This specification defines the conformance criteria that applies to a single product: the user agent.

Implementations that use ECMAScript to implement the objects defined in this specification must implement them in a manner consistent with the ECMAScript Bindings defined in the Web IDL specification [WEBIDL], as this document uses that specification and terminology.

This specification does not define what objects a conforming implementation should generate. Specifications that refer to this specification have the need to specify conformance. They should put in their document text like this:

- An implementation must support generating statistics for the type `RTCInboundRtpStreamStats`, with attributes `packetsReceived`, `bytesReceived`, `packetsLost`, `jitter`, and `fractionLost`.
- It must support generating statistics for the type `RTCOutboundRtpStreamStats`, with attributes `packetsSent`, `bytesSent` For all subclasses of `RTCTrtpStreamStats`, it must include `ssrc` and `kind`.
- When stats exist for both sides of a connection, in the form of an `inbound-rtp` / `remote-outbound-rtp` pair or an `outbound-rtp` / `remote-inbound-rtp` pair, the members `remoteId` and `localId` must also be present. It may support generating other stats.

Search

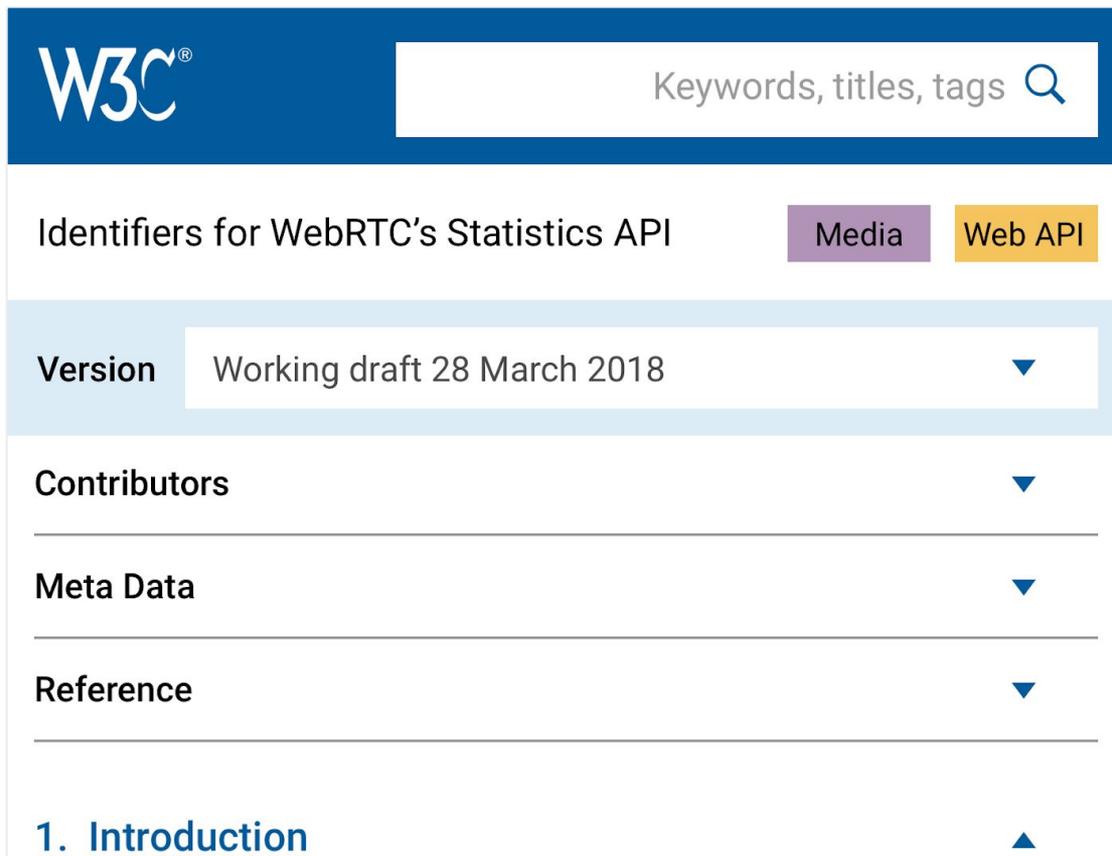
File a bug

GitHub

Commit history

The Mobile Version

To view the full resolution mobile version designs, please click here.



The screenshot shows the top portion of the W3C website. At the top left is the W3C logo. To its right is a search bar with the placeholder text "Keywords, titles, tags" and a magnifying glass icon. Below the search bar, the main title "Identifiers for WebRTC's Statistics API" is displayed, followed by two buttons: "Media" (purple) and "Web API" (yellow). A light blue navigation bar contains a "Version" dropdown menu currently set to "Working draft 28 March 2018". Below this are four menu items: "Contributors", "Meta Data", "Reference", and "1. Introduction", each with a downward-pointing triangle. The "1. Introduction" item has an upward-pointing triangle, indicating it is the active page.

W3C®

Keywords, titles, tags 

Identifiers for WebRTC's Statistics API [Media](#) [Web API](#)

Version Working draft 28 March 2018 ▼

Contributors ▼

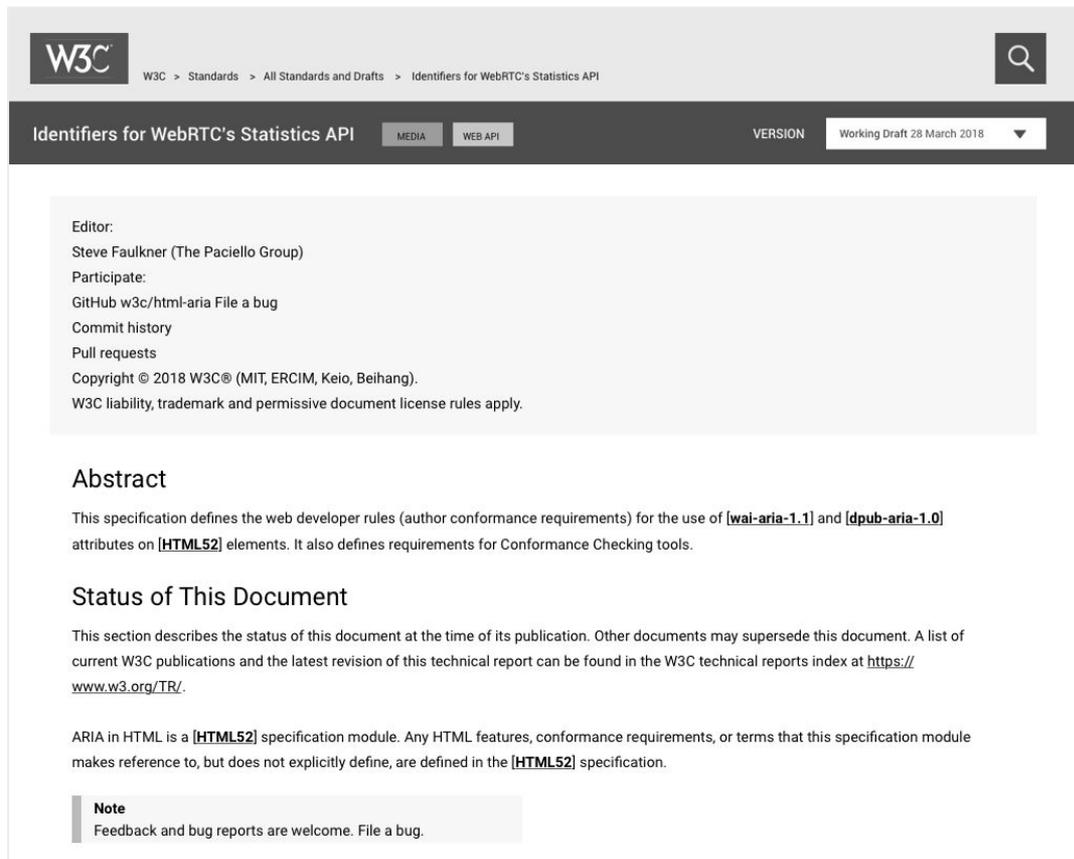
Meta Data ▼

Reference ▼

1. Introduction ▲

The Print Version

To view the full resolution print version designs, please click here.



The screenshot shows the W3C website for the document 'Identifiers for WebRTC's Statistics API'. The page has a dark header with the W3C logo, navigation links, and a search icon. Below the header, there are tabs for 'MEDIA' and 'WEB API', and a 'VERSION' dropdown menu set to 'Working Draft 28 March 2018'. The main content area is light gray and contains the following text:

Editor:
Steve Faulkner (The Paciello Group)

Participate:
[GitHub w3c/html-aria File a bug](#)
[Commit history](#)
[Pull requests](#)

Copyright © 2018 W3C® (MIT, ERCIM, Keio, Beihang).
W3C liability, trademark and permissive document license rules apply.

Abstract

This specification defines the web developer rules (author conformance requirements) for the use of [\[wai-aria-1.1\]](#) and [\[dpub-aria-1.0\]](#) attributes on [\[HTML52\]](#) elements. It also defines requirements for Conformance Checking tools.

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 <https://www.w3.org/TR/>.

ARIA in HTML is a [\[HTML52\]](#) specification module. Any HTML features, conformance requirements, or terms that this specification module makes reference to, but does not explicitly define, are defined in the [\[HTML52\]](#) specification.

Note
Feedback and bug reports are welcome. [File a bug.](#)

Next Steps

The Problem

Design Process

Next Steps

Next Steps

We have listed a few improvements that we have researched through our process that could be implemented in the future to improve usability and accessibility of the specifications.

Next Steps

- Create a documentation style guide to improve the consistency, length and formatting of language in specifications
 - Link behavior (open in new tab, etc) should be consistent
 - Contact links (a vs mailto) should be consistent
 - Documentation should be less wordy
 - Code samples should be informative
 - Diagrams and Information layout could be improved
 - Text shown on images cannot be searched

Next Steps

- Create key or guide to versions (e.g. help users navigate “working draft,” “public draft,” “editor’s draft”)
- Help Google find the current published version of a specification, unless the user indicates otherwise
- Add “return to top” tag after each section of specification, making it easier to browse specification
- Create a track changes or specification comparison tool to help users ID changes
- Consider providing more robust tools for copying / utilizing code snippets



Thank You!!!