

Accessible Testing in WPT: From Dream to Reality

Valerie Young (spectranaut)
2026 Web Engines Hackfest

Standards Tests



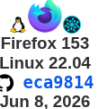
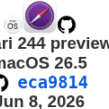


Web Platform Tests

web-platform-tests dashboard

[Latest Run](#)
[Recent Runs](#)
[🌟 Interop 2026 🌟](#)
[Insights](#)
[Processor](#)
[About](#)

Showing 120393 tests (2227882 subtests) from the latest master test runs for chrome[experimental], edge[experimental], firefox[experimental], safari[experimental]

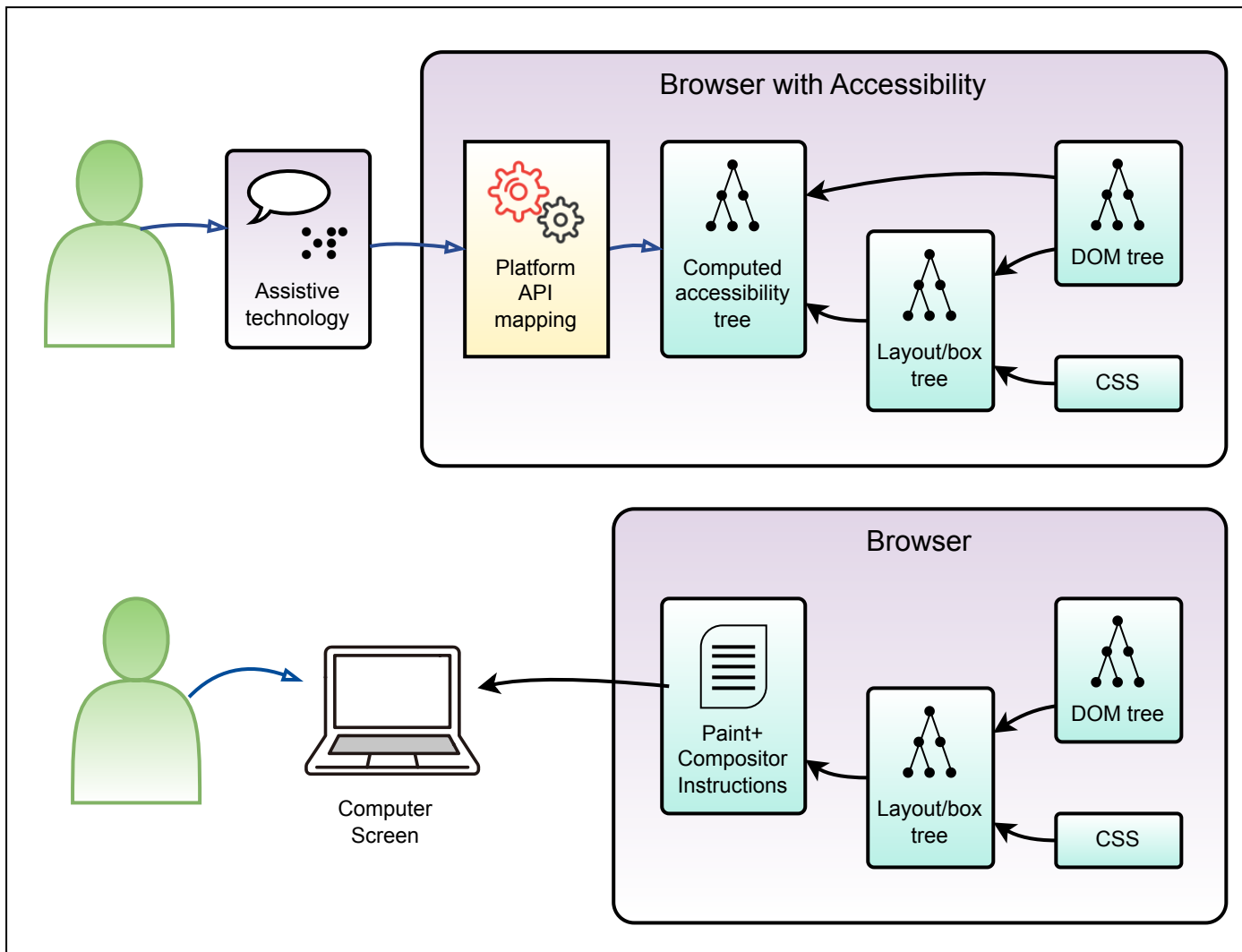
[LINK](#)
[EDIT](#)

Path	 Chrome 151 Linux 22.04 eca9814 Jun 8, 2026	 Edge 150 Windows 10.0 eca9814 Jun 8, 2026	 Firefox 153 Linux 22.04 eca9814 Jun 8, 2026	 Safari 244 preview macOS 26.5 eca9814 Jun 8, 2026
^	^	^	^	^
accelerometer/	119 / 150	59 / 150	8 / 150	8 / 150
accessibility/ 	58 / 58	58 / 58	58 / 58	58 / 58
accname/ 	466 / 484	466 / 484	462 / 484	462 / 484
acid/	102 / 102	102 / 102	100 / 102	102 / 102
ai/	85 / 395	77 / 404	3 / 404	3 / 404
ambient-light/	39 / 50	19 / 50	4 / 50	4 / 50
animation-worklet/	18 / 86	18 / 86	18 / 86	18 / 86
apng/	2 / 2	2 / 2	2 / 2	2 / 2
attribution-reporting/	19 / 24	19 / 24	2 / 24	2 / 24
audio-output/	20 / 45	20 / 45	45 / 45	22 / 45
audio-session/	14 / 40	14 / 40	14 / 40	35 / 40
autoplay-policy-detection/	14 / 22	14 / 22	22 / 22	14 / 22
avif/	1 / 1	1 / 1	0 / 1	1 / 1
background-fetch/	209 / 244	207 / 235	34 / 244	34 / 244

What has it meant to *not* have accessibility testing?

- When using a browser with assistive technology (like a screen reader):
 - Additional W3C standards apply
 - Different code paths are executed
- For a long time, wpt.fyi could tell you the status of a new feature for sighted users... but not if accessibility support was complete or correct!

⊗ You cannot know if new feature was implemented for all users.



Briefly

Testing the computed accessibility tree

- Would be nice to test! It's platform-independent, full of familiar concepts.
- But, exposing that information in javascript is no-go.
- In 2023: Webdriver endpoint for role and name!
- Working on webdriver end points for many other attributes! (checked, selected, focus, etc...)

To really know if there is full browser support...

Test the Platform Accessibility API

- "Platform" as in operating system: Linux, Mac, two on Windows, Android
- They are all trees, but are otherwise pretty different.
- ARIA WG of the W3C standardizes how a website is translated to/exposed through these APIs with the **AAMs**.

Long history of Igalia trying to solve this lack of tests

- **2017** "Manual" tests
 - Joanie Diggs started and maintained **"manual" tests in WPT**.
 - Tests needed a program outside of WPT, different for each API.
 - Used for generating proof of "adequate implementation experience" when publishing ARIA 1.1.
- **2022** Chromium inspired tests
 - Desire to have a single, cross platform, fully automated tests.
 - I experimented with using a **utility in Chrome**...

Then in 2024.. it finally occurred to me:

- 1.** The WPT test runner is written in Python
- 2.** There are Python bindings for all Accessibility APIs....

Automated tests should be possible in WPT!!!

Opened RFC for WPT

Problems to Solve for AAM tests in WPT

1. How to design the tests/testing API
2. How to handle platform-specific test results
3. Find the right browser in accessibility API
4. Turn on "accessibility" in the browser for these tests

First attempt: use `testharness` test type and extended `testdriver.js`

```
<!doctype html>
<title>core-aam: blockquote role</title>
<script src="/resources/testharness.js"></script>
<script src="/resources/testharnessreport.js"></script>
<script src="/resources/testdriver.js"></script>

<body>
  <div id=test role=blockquote>quote</div>

  <script>
    promise_test(async t => {
      const node = await test_driver.get_platform_accessibility_node('test');

      if (node.API == 'atspi') {
        assert_equals(node.role, 'block quote', 'Atspi role');
      }
      if (node.API == 'axapi') {
        assert_equals(node.role, 'AXGroup', 'AX API role');
      }
      ...
    }, 'role blockquote');
```

Second attempt: `aamtest`, a new test type modeled after `wdspec` tests


```
TEST_HTML = "<div role='blockquote' id='test'>content</div>"

def test_atspi(atspi, session, inline):
    session.url = inline(TEST_HTML)
    node = atspi.find_node("test", session.url)
    assert atspi.Accessible.get_role(node) == atspi.Role.BLOCK_QUOTE







def test_axapi(axapi, session, inline):
    session.url = inline(TEST_HTML)
    node = axapi.find_node("test", session.url)
    role = axapi.AXUIElementCopyAttributeValue(node, "AXRole", None)[1]
    assert role == "AXGroup"

...
```

Problems to Solve for AAM tests in WPT

1. How to design the tests/testing API 
2. **How to handle platform-specific test results**
3. Find the right browser in accessibility API
4. Turn on "accessibility" in the browser for these tests

Tests that do not apply to a platform end up like "missing" tests, in order to have the correct subtest total in wpt.fyi.

Subtest	 Chrome 151 Linux 22.04  433496c Jun 8, 2026	 Firefox 153 Linux 22.04  433496c Jun 8, 2026	 Safari 245 preview macOS 26.5  433496c Jun 8, 2026
Harness status	OK	OK	MISSING
Duration	0.777 seconds	2.221 seconds	MISSING
test_atspi	PASS	FAIL	MISSING
Subtest Total	1/1	0/1	0/0

[timer test in wpt.fyi](#)




Problems to Solve for AAM tests in WPT

1. How to design the tests/testing API ✓
2. How to handle platform-specific test results ✓
3. **Find the right browser in accessibility API**
4. Turn on "accessibility" in the browser for these tests

By PID or by name? Ideally PID.

- GeckoDriver returns the PID by default, I updated ChromeDriver to do the same.
- Can find browser by PID:
 - Firefox on Linux, Windows and macOS
 - Chrome on Linux, Windows and macOS
- Must find browser by name:
 - Edge - need to update edgedriver
 - Safari - need to update safaridriver

Problems to Solve for AAM tests in WPT





1. How to design the tests/testing API 
2. How to handle platform-specific test results 
3. Find the right browser in accessibility API 
4. Turn on "accessibility" in the browser for these tests

WPT test runner restarts browser when encountering these tests...

So we can turn on accessibility then!

- For Firefox and Chrome, this can be done with a command-line flag.
- For Safari, this features is being worked on for this test type, by Tyler Wilcock.
- For Edge.... it should be the same as Chrome 🙄🙄

Problems to Solve for AAM tests in WPT

1. How to design the tests/testing API 
2. How to handle platform-specific test results 
3. Find browser in accessibility API by PID 
4. Turn on "accessibility" in the browser for these tests 

May 5 2026: AAM tests infrastructure merged!

You can now write tests of:

- Linux Accessibility API - AT-SPI
- Mac Accessibility API - AX API
- One of the Windows Accessibility APIs - IA2
- As of last week, Theo Hale opened a PR for UIA support!

Full support Firefox and Chrome on all three platforms, but Safari and Edge can't yet run these tests yet.

If you want:

- Safari or Edge to run these tests
- If you want to add support for more Accessibility APIs (UIA, Android)
- If you want to write tests

Come talk to me this week!

Acknowledgments - Helped with AAM tests landing!

- Alice Boxhall (Igalia)
- Joanie Diggs (Igalia)
- Ms2ger (Igalia)
- Theo Hale (independent)
- James Craig (Apple)
- Tyler Wilcock (Apple)
- Sam Sneddon (Apple)
- Jamie Teh (Mozilla)
- James Graham (Mozilla)
- Simon Pieters (Mozilla)
- Jonathan Lee (Google)
- Philip Jägenstedt (Google)

