Consumer Technology Association™

Web Media API Snapshot Test Suite Enhancements

November 2020

NOTICE

Consumer Technology Association (CTA)™ Standards, Bulletins and other technical publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need. Existence of such Standards, Bulletins and other technical publications shall not in any respect preclude any member or nonmember of the Consumer Technology Association from manufacturing or selling products not conforming to such Standards, Bulletins or other technical publications, nor shall the existence of such Standards, Bulletins and other technical publications preclude their voluntary use by those other than Consumer Technology Association members, whether the standard is to be used either domestically or internationally.

Standards, Bulletins and other technical publications are adopted by the Consumer Technology Association in accordance with the American National Standards Institute (ANSI) patent policy. By such action, the Consumer Technology Association does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the Standard, Bulletin or other technical publication.

This document does not purport to address all safety problems associated with its use or all applicable regulatory requirements. It is the responsibility of the user of this document to establish appropriate safety and health practices and to determine the applicability of regulatory limitations before its use.

(Formulated under the cognizance of the CTA **WAVE Project**; for information please see cta.tech/WAVE.)

Published by
CONSUMER TECHNOLOGY ASSOCIATION
Technology & Standards Department
www.cta.tech

Web Media API Snapshot Test Suite Enhancements

By John Riviello, Fellow, Comcast and CTA WAVE HTML5 API Task Force Chair

What is WMATS?

The Web Media API Test Suite (WMATS) [https://github.com/cta-wave/WMAS] is a test suite for the Web Media API Snapshot specifications [https://github.com/w3c/webmediaapi/]. The test suite and specification are being developed as part of the CTA WAVE Project [https://cta.tech/WAVE]. The project is forked from W3C Web Platform Tests (WPT) [https://web-platform-tests.org/] and is customized to automate test runs on web browsers for embedded devices and appliances such as TV sets, set-top boxes, game consoles, etc.

These updates to WPT were contributed back to the project so that other users of WPT can take advantage of it, and to allow CTA WAVE to develop future enhancements from the latest WPT updates that can also be contributed back.

Why run WMAS Tests against our devices?

CTA WAVE's Web Media API Snapshot documents the web APIs that are supported across all four of the most widely used user agent code bases. As the web is a fast evolving platform, capturing this information alone is not enough. By executing the WMAS tests against our devices, we can know with certainty what level of web API support the browsers we embed are providing to our customers and the web developers that build web applications for them to use on our devices.

Improvements to the manual testing process

The current hosted version is available at https://webapitests2019.ctawave.org/wave/. For dedicated testing, it is recommended to download the software from https://github.com/cta-wave/WMAS and run the server on either your local network or in the cloud. Upon launching the URL from the device you wish to test, you are presented with a QR code. This enhancement was made to make it easy to configure, initiate execution of, and view the progress of the tests on a device separate from the device under test.

By default, all APIs will be tested. You can limit the test to a subset of APIs by deselecting the ones you wish to skip. This is useful if a test is causing issues on the device under test that cannot be addressed quickly, as the other tests can run while the problematic test can be skipped while engineers investigate the issue. Specific tests can also be excluded by providing paths to test files or directories to exclude them from the session. While a test is running, you can mark tests that time out as "malfunctioning" so you can easily run another test later that skips those tests. You can also configure the test suite to only execute the tests that are known to pass on a particular reference browser.

Once your test run is configured, tap "Start Session" and a live report of the test suite progress will appear in your browser while the device under test goes through the testing process. While the test is running, there are options available to pause or stop the testing. If a test is paused for any reason, the test can be easily resumed.

Upon completion of the test run, a report is available for analysis. The report shows the test results sorted by the type of failure if applicable (failures vs. timeouts), and then sorted by feature.

Comparing test results over time

Labels can be added to a test run either before, during, or after the tests are executed. These labels can serve whichever purpose you wish, such as enabling the test suite to be integrated into and run by another test suite.

In addition to overall test results, a report is also available for each set of APIs that were tested. These test reports are available as JSON or HTML, complete with each failed assert message.

Previously run sessions are easily available from the bottom of the results screen. By default, recently run and pinned tests are displayed, but any prior test can be imported by providing the session token or importing an export of a prior test. These previous sessions can easily be sorted, and you can select multiple sessions to view a high-level comparison report in the web UI.

Test automation

A major feature that was added to the WMAS test suite was a REST API. The REST API contains a full suite of features that brings many of the same functionality of the web UI to the REST API, including creating, reading, deleting, and interacting with a test session and its results. At Comcast, we leverage this to create, configure and execute new test runs as part of our automation testing. The REST API gives us full control to be able to execute scripts that interact with the test session as it runs, including getting an in-progress update on the test statistics, and the ability to pause and resume an active test.

The REST API is fully documented at https://github.com/cta-wave/WMAS/blob/wmas2018/tools/wave/docs/rest-api/README.md.