

This is a working document to formulate the 2020 renewed charter for the WebPerfWG. It is based on the [2018 charter](#).

## Open questions

- Admin
  - Can we make sure all specs' TR/ is up-to-date with the latest changes? Can we set up auto-publishing for everything which isn't up to date?
    - Do we need to say something in the charter about that?
  - If we go for a "CR draft" Living Standard model, when do we check that all implementations pass tests?
    - Snapshots?
- Larger themes
  - SPA performance
  - Runtime performance
  - UX metrics
  - Improved privacy & security - Impact of Isolated Contexts on WebPerf APIs, opt-in consolidation, etc
- Spec status
  - Publishing - maybe we can get those out the door before the charter change?
    - Performance Timeline L2 is ready for PR or even CR
      - Some [failing tests](#)
    - Resource Timing L1 - can we move to REC?
  - Almost there - Q3 2020 timelines for publications?
    - requestIdleCallback L1
      - Long standing issues - need to revive them
    - Page Visibility L2
      - HTML rewrite
    - Beacon
      - One lingering [test](#)
    - Paint Timing
      - Spec is (almost) good to go, but implementations don't yet pass tests
  - "Regressed" - Q4 2020/Q1 2021?
    - Resource Timing L2
    - Navigation Timing L2
  - "Stuck" - missing second implementer interest
    - Long Tasks
    - Device Memory
    - NEL
  - Incubations - can we adopt?
    - Event Timing
    - Element Timing

- Largest Contentful Paint
  - Layout Instability
  - JS self-profiling
  - Memory Measurement API
  - IsInputPending - already concluded that we should adopt
  - Client Hints infra/features ???
  - postTask API
  - ??
- 

## Web Performance Working Group Charter

The **mission** of the [Web Performance Working Group](#) is to provide methods to observe and improve aspects of application performance of user agent features and APIs.

### Scope

---

Web developers are building sophisticated applications where application performance is a critical feature. Web developers want the ability to observe the performance characteristics of their applications, and they want the ability to write more efficient applications, using well-defined interoperable methods. Their methods must be both secure and privacy-enabling by design, using well-defined interoperable methods that conform to the current Web browser security model.

The Web Performance Working Group's scope of work includes user agent features and APIs to observe and improve various aspects of application performance:

- Measurement
  - The WG will deliver APIs to measure network and rendering performance, responsiveness and interactivity, memory and CPU use, application failures, and other aspects of the user's experience, from the perspectives of both application loading and continuous user interaction. In particular, the group will deliver means to measure user experience metrics in popular web architectures, such as single-page apps, as well as means to collect those measurements.
- Scheduling
  - The WG will deliver APIs that would enable developers to improve scheduling of application work as well as use of network resources, to minimize loading times and ensure ideal user experience.
- Adaptation

- The WG will deliver APIs that would enable developers to deliver resources adapted to the user's device, preferences, conditions, or other circumstances.

Such deliverables will apply to desktop and mobile browsers and other non-browser environments where appropriate, and will be consistent with Web technologies designed in other working groups including HTML, CSS, Web Application Security, Web Platform, Device and Sensors, and SVG. The Group will strive for its deliverables to be well-integrated with other specifications where appropriate.

In addition to developing Recommendation Track documents, the Web Performance Working Group may provide review of specifications from other Working Groups.

## Out of Scope

---

The following features are out of scope, and will not be addressed by this working group.

- **performance data analysis techniques or algorithms.**

## Success Criteria

---

In order to advance to [Proposed Recommendation](#), each specification is expected to have at least two independent implementations of each feature defined in the specification.

Each specification should contain a section detailing any [known security](#), fingerprinting, and privacy implications, and suggested [mitigation strategies](#) for implementers, Web authors, and end users. The group should not publish a specification if acceptable mitigation strategies cannot be found.

Normative specification changes are generally expected to have a corresponding set of tests, either in the form of new tests or modifications to existing tests, or must include the rationale for why test updates are not required for the proposed update.

## Deliverables

---

Note: Latest milestone and publication schedules for the group's deliverables listed below can be found on the [publication status page](#).

*Draft state* indicates the state of the deliverable at the time of the charter approval. *Next Recommendation expected completion* indicates when the deliverable is projected to become a Recommendation, or otherwise reach a stable state.

## Normative Specifications

---

The working group will deliver incremental revisions of the following W3C normative specifications:

### High Resolution Time

This document defines an API that provides the current time in sub-millisecond resolution such that it is not subject to system clock skew or adjustments.

**Level 3 Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q2 2021

Latest publication: [01 October 2020](#)

Reference Draft: <https://www.w3.org/TR/2018/CR-hr-time-2-20180301/>

associated [Call for Exclusion](#) on 01 March 2018 ended on 30 April 2018

Produced under Working Group Charter: <https://www.w3.org/2016/07/webperf>

### Performance Timeline

This specification defines a unified interface to store and retrieve performance metric data.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q1 2021

Latest publication: [24 October 2019](#)

Reference Draft: <https://www.w3.org/TR/2019/WD-performance-timeline-2-20191024/>

associated [Call for Exclusion](#) on 08 December 2016 ended on 06 February 2017

Produced under the [2016 Working Group Charter](#)

### Resource Timing

An interoperable means for site developers to collect real-world performance information from the user agent while loading resources as specified from the root document of a webpage. The user agent captures the end to end latency associated with loading resources from a web server. This includes the time associated with fetching the resource from the network and the time associated to load the resource within the user agent. This might include:

- timing associated with the network;
- timings associated with loading the resource in the document;

- a resource may be one of the following elements: iframe, img, script, object, embed and link.

**Level 1 Draft state:** [Candidate Recommendation](#)

**Next Recommendation expected completion:** Q1 2021

Latest publication: [30 March 2017](#)

Reference Draft: <https://www.w3.org/TR/2017/CR-resource-timing-1-20170330/>

associated [Call for Exclusion](#) on 30 March 2017 ended on 29 May 2017

Produced under the [2016 Working Group Charter](#)

**Level 2 Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Living Standard

Latest publication: [18 February 2020](#)

Reference Draft: <https://www.w3.org/TR/2020/WD-resource-timing-2-20200218/>

associated [Call for Exclusion](#) on 03 November 2016 ended on 02 April 2017

Produced under the [2016 Working Group Charter](#)

## **[Navigation Timing](#)**

An interoperable means for site developers to collect real-world performance information from the user agent while loading the root document of a webpage. The user agent captures the end to end latency associated with loading a webpage from a web server. This includes the time associated with fetching the document from the network to the time associated to load the document within the user agent.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Living Standard

Note: after Level 2 of Navigation Timing is ratified, future development of new Navigation Timing features will be done within Resource Timing; Navigation Timing will be merged with Resource Timing.

Latest publication: [21 January 2020](#)

Reference Draft: <https://www.w3.org/TR/2020/WD-navigation-timing-2-20200121/>

associated [Call for Exclusion](#) on 31 January 2013 ended on 30 June 2013

Produced under [2013 Working Group Charter](#)

### **User Timing**

An interoperable means for site developers to capture timing information with a developer supplied name. The user agent captures the time stamp at the point and time specified in the code executing in the user agent.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Living Standard

Latest publication: [26 March 2020](#)

Reference Draft: <https://www.w3.org/TR/2020/WD-user-timing-3-20200326/>

associated [Call for Exclusion](#) on 01 November 2016 ended on 31 March 2017

Produced under the [2018 Working Group Charter](#)

### **Page Visibility**

An interoperable means for site developers to programmatically determine the current visibility of a document and be notified of visibility changes.

**Draft state:** [Proposed Recommendation](#)

**Next Recommendation expected completion:** Q1 2021

Latest publication: [17 October 2017](#)

Reference Draft: <https://www.w3.org/TR/2017/CR-page-visibility-2-20170328/>

associated [Call for Exclusion](#) on 29 March 2017 ended on 27 May 2017

Produced under the [2016 Working Group Charter](#)

### **Beacon**

An interoperable API for site developers to asynchronously transfer data from the user agent to a web server, with a guarantee from the user agent that the data will be eventually sent.

**Draft state:** [Candidate Recommendation](#)

**Next Recommendation expected completion:** Q4 2020

Latest publication: [13 April 2017](#)

Reference Draft: <https://www.w3.org/TR/2017/CR-beacon-20170413/>

associated [Call for Exclusion](#) on 13 April 2017 ended on 12 June 2017

Produced under the [2016 Working Group Charter](#)

### **Cooperative Scheduling of Background Tasks**

An API that web page authors can use to cooperatively schedule background tasks such that they do not introduce delays to other high priority tasks that share the same event loop, such as input processing, animations and frame compositing.

**Draft state:** [Proposed Recommendation](#)

**Next Recommendation expected completion:** Q2 2021

Latest publication: [10 October 2017](#)

Reference Draft: <https://www.w3.org/TR/2017/CR-requestidlecallback-20170131/>

associated [Call for Exclusion](#) on 02 February 2017 ended on 01 April 2017

Produced under the [2016 Working Group Charter](#)

### **Long Tasks**

This specification defines an API that web page authors can use to detect presence of "long tasks" that monopolize the UI thread for extended periods of time and block other critical tasks from being executed - e.g. reacting to user input.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q4 2021

Latest publication: [07 September 2017](#)

Reference Draft: <https://www.w3.org/TR/2017/WD-longtasks-1-20170907/>

associated [Call for Exclusion](#) on 07 December 2017 ended on 04 February 2018

Produced under the [2016 Working Group Charter](#)

## **Paint Timing**

An interoperable API for site developers to capture the set of key moments during pageload - e.g. first paint and first contentful paint.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q1 2021

Latest publication: [07 September 2017](#)

Reference Draft: <https://www.w3.org/TR/2017/WD-paint-timing-20170907/>

associated [Call for Exclusion](#) on 07 December 2017 ended on 04 February 2018

Produced under the [2016 Working Group Charter](#)

## **Server Timing**

This specification enables a server to communicate performance metrics about the request-response cycle to the user agent. It also standardizes a JavaScript interface to enable applications to collect, process, and act on these metrics to optimize application delivery.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q2 2021

Latest publication: [02 January 2020](#)

Reference Draft: <https://www.w3.org/TR/2017/WD-server-timing-20170713/>

associated [Call for Exclusion](#) on 13 July 2017 ended on 10 December 2017

Produced under the [2016 Working Group Charter](#)

## **Device Memory**

This specification defines a JavaScript API and HTTP Client Hint header to surface device capability for memory — i.e. device RAM, in order to enable web apps to customize content depending on device memory constraints.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q4 2021



Latest publication: [25 September 2018](#)

Produced under the [2018 Working Group Charter](#)

## **Reporting**

This specification defines a generic reporting framework which allows web developers to associate a set of named reporting endpoints with an origin. Various platform features (like Content Security Policy, Network Error Reporting, and others) will use these endpoints to deliver feature-specific reports in a consistent manner.

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q4 2021

Latest publication: [25 September 2018](#)

Reference Draft: <https://www.w3.org/TR/2018/WD-reporting-1-20180925/>

associated [Call for Exclusion](#) on 7 April 2016 ended on 4 September 2016

Produced under the [2016 Working Group Charter](#)

## **Network Error Logging**

This specification defines a mechanism that enables developers to declare a network error reporting policy for a web application. A user agent can use this policy to report encountered network errors that prevented it from successfully fetching requested resource

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Q4 2021

Latest publication: [25 September 2018](#)

Reference Draft: <https://www.w3.org/TR/2014/WD-navigation-error-logging-20140211/>

associated [Call for Exclusion](#) on 12 February 2014 ended on 11 July 2014

Produced under the [2016 Working Group Charter](#)

## **Past Deliverables**

---

The Group plans to transition the contents of the following past-deliverables to the WHATWG and publish them as a Note.

### **Resource Hints**

This specification defines the dns-prefetch, preconnect, prefetch, and prerender relationships of the HTML Link Element (`<link>`).

**Draft state:** [Working Draft](#)

**Next Recommendation expected completion:** Transition to WHATWG HTML

Latest publication: [02 July 2019](#)

Reference Draft: <https://www.w3.org/TR/2014/WD-resource-hints-20141021/>

associated [Call for Exclusion](#) on 23 October 2014 ended on 20 March 2015

Produced under [2013 Working Group Charter](#)

## [Preload](#)

This specification defines the link relation type preload, a declarative fetch primitive that initiates an early fetch and separates fetching from resource execution.

**Draft state:** [Candidate Recommendation](#)

**Next Recommendation expected completion:** Transition to WHATWG HTML

Latest publication: [26 June 2019](#)

Reference Draft: <https://www.w3.org/TR/2019/CR-preload-20190626/>

associated [Call for Exclusion](#) on 26 October 2017 ended on 25 December 2017

Produced under the [2016 Working Group Charter](#)

## Other Deliverables

---

The Group will maintain a [non-normative Primer document](#) to facilitate the understanding of the interrelationships between its deliverables.

Other non-normative documents may be created such as:

- [Security and Privacy considerations for Performance APIs](#)
- Test suites for each specification;
- Use cases documents to support development and exploration of performance-critical web platform features and APIs;
- Primer or Best Practice documents to support web developers when designing applications with performance in mind;

- Developer and user guides for its normative specifications;
- Best practice document explaining how data is commonly gathered between the client and the server.

## Coordination

---

For all specifications, this Working Group will seek [horizontal review](#) for accessibility, internationalization, performance, privacy, and security with the relevant Working and Interest Groups, and with the [TAG](#). Invitation for review must be issued during each major standards-track document transition, including [FPWD](#) and [CR](#), and should be issued when major changes occur in a specification.

Additional technical coordination with the following Groups will be made, per the [W3C Process Document](#):

### W3C Groups

---

#### [Technical Architecture Group \(TAG\)](#)

The WG may ask the Technical Architecture Group to review some of its specifications.

#### [Web Platform Working Group](#)

This Working Group develops APIs for client-side development and for markup vocabularies for describing and controlling client-side application behavior.

#### [Web Application Security Working Group](#)

This Working Group develops security and policy mechanisms to improve the security of Web Applications, and enable secure cross-site communication.

#### [CSS Working Group](#)

The group develops and maintains CSS, which is relevant for specifications such as page visibility.

#### [Privacy Interest Group](#)

This group monitors ongoing privacy issues that affect the Web, investigates potential areas for new privacy work, and provides guidelines and advice for addressing privacy in standards development.

#### [Web Platform Incubator Community Group](#)

This group provides a lightweight venue for proposing, incubating and discussing new web platform features. The Web Performance Working group will incubate and review new proposals that are within scope of our charter within the WICG. Once such WICG-incubated proposal is implemented and available in at least one major browser, and has support from one more, it may be adopted by the Web Performance Working group.

## External Organizations

---

### [ECMA Technical Committee 39 \(TC39\)](#)

This is the group responsible for ECMAScript standardization, and related ECMAScript features like E4X. As the Web Performance Working Group will be developing ECMAScript APIs, it should collaborate with TC39.

#### [Internet Engineering Task Force](#)

The IETF is responsible for defining robust and secure protocols for Internet functionality, in particular HTTP. The Working Group should coordinate protocol-related work (e.g. profiles of hybi or HTTP) with the appropriate IETF WGs.

#### [Web Hypertext Application Technology Working Group \(WHATWG\)](#)

The Web Hypertext Application Technology Working Group (WHATWG) is a community of people interested in evolving the web through standards and tests.

## Participation

---

To be successful, this Working Group is expected to have 5 or more active participants for its duration, including representatives from the key implementors of this specification, and active Editors and Test Leads for each specification. The Chairs, specification Editors, and Test Leads are expected to contribute half of a day per week towards the Working Group. There is no minimum requirement for other Participants.

The group encourages questions, comments and issues on its public mailing lists and document repositories, as described in [Communication](#).

The group also welcomes non-Members to contribute technical submissions for consideration upon their agreement to the terms of the [W3C Patent Policy](#).

## Communication

---

Technical discussions for this Working Group are conducted in [public](#). Meeting minutes from teleconference and face-to-face meetings will be archived for public review, and technical discussions and issue tracking will be conducted in a manner that can be both read and written to by the general public. Working Drafts and Editor's Drafts of specifications will be developed on a public repository, and may permit direct public contribution requests.

Information about the group (including details about deliverables, issues, actions, status, participants, and meetings) will be available from the [Web Performance Working Group home page](#).

Most Web Performance Working Group teleconferences will focus on discussion of particular specifications, and will be conducted on an as-needed basis.

This group primarily conducts its technical work in its [public repositories](#). There is also a public mailing list [public-web-perf@w3.org](mailto:public-web-perf@w3.org) ([archive](#)). The public is invited to contribute to the github

repositories and post messages to the list. Regular activity summaries around the github repositories will be provided.

## Decision Policy

---

This group will seek to make decisions through consensus and due process, per the [W3C Process Document \(section 3.3\)](#). Typically, an editor or other participant makes an initial proposal, which is then refined in discussion with members of the group and other reviewers, and consensus emerges with little formal voting being required.

However, if a decision is necessary for timely progress, but consensus is not achieved after careful consideration of the range of views presented, the Chairs may call for a group vote, and record a decision along with any objections.

To afford asynchronous decisions and organizational deliberation, any resolution (including publication decisions) taken in a face-to-face meeting or teleconference will be considered provisional until 10 working days after the publication of the resolution in draft minutes sent, which will be published within 5 working days of the meeting to the working group's mailing list with a 'call for consensus' in the subject line. If no objections are raised on the mailing list by the end of the response period, the resolution will be considered to have consensus as a resolution of the Working Group.

All decisions made by the group should be considered resolved unless and until new information becomes available, or unless reopened at the discretion of the Chairs or the Director.

This charter is written in accordance with the [W3C Process Document \(Section 3.4, Votes\)](#), and includes no voting procedures beyond what the Process Document requires.

## Patent Policy

---

This Working Group operates under the [W3C Patent Policy](#) 5 February 2004 (updated 1 August 2017). To promote the widest adoption of Web standards, W3C seeks to issue Recommendations that can be implemented, according to this policy, on a Royalty-Free basis. For more information about disclosure obligations for this group, please see the [W3C Patent Policy Implementation](#).

## Licensing

---

This Working Group will use the [W3C Software and Document license](#) for all its deliverables.

## About this Charter

---

This charter has been created according to [section 5.2 of the Process Document](#). In the event of a conflict between this document or the provisions of any charter and the W3C Process, the W3C Process shall take precedence.