

**HTTP Client Hints**  
**draft-grigorik-http-client-hints-01**

Abstract

An increasing diversity of Web-connected device form factors and software capabilities has created a need to deliver varying, or optimized content for each device.

HTTP Client Hints can be used as input to proactive content negotiation; just as the Accept header allowed clients to indicate what formats they prefer, Client Hints allow clients to indicate a list of device and agent specific preferences.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 9, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	<a href="#">Introduction</a>	<a href="#">3</a>
<a href="#">1.1.</a>	<a href="#">Notational Conventions</a>	<a href="#">3</a>
<a href="#">2.</a>	<a href="#">Client Hint Request Header Fields</a>	<a href="#">3</a>
<a href="#">2.1.</a>	<a href="#">Hint Values</a>	<a href="#">4</a>
<a href="#">2.2.</a>	<a href="#">Sending Client Hints</a>	<a href="#">4</a>
<a href="#">2.3.</a>	<a href="#">Server Processing of Client Hints</a>	<a href="#">5</a>
<a href="#">2.3.1.</a>	<a href="#">Advertising Support for Client Hints</a>	<a href="#">5</a>
<a href="#">2.3.2.</a>	<a href="#">Interaction with Caches</a>	<a href="#">5</a>
<a href="#">3.</a>	<a href="#">The CH-DPR Client Hint</a>	<a href="#">6</a>
<a href="#">4.</a>	<a href="#">The CH-RW Client Hint</a>	<a href="#">6</a>
<a href="#">4.1.</a>	<a href="#">Confirming Selected DPR</a>	<a href="#">6</a>
<a href="#">5.</a>	<a href="#">Examples</a>	<a href="#">7</a>
<a href="#">5.1.</a>	<a href="#">Relationship to the User-Agent Request Header</a>	<a href="#">7</a>
<a href="#">6.</a>	<a href="#">IANA Considerations</a>	<a href="#">7</a>
<a href="#">7.</a>	<a href="#">Security Considerations</a>	<a href="#">8</a>
<a href="#">8.</a>	<a href="#">Normative References</a>	<a href="#">8</a>
	<a href="#">Author's Address</a>	<a href="#">9</a>



## 1. Introduction

There are thousands of different devices accessing the web, each with different device capabilities and preference information. These device capabilities include hardware and software characteristics, as well as dynamic user and client preferences.

One way to infer some of these capabilities is through User-Agent (UA) detection against an established database of client signatures. However, this technique requires acquiring such a database, integrating it into the serving path, and keeping it up to date. However, even once this infrastructure is deployed, UA sniffing has numerous limitations:

- o UA detection cannot reliably identify all static variables
- o UA detection cannot infer any dynamic client preferences
- o UA detection requires an external device database
- o UA detection is not cache friendly

A popular alternative strategy is to use HTTP cookies to communicate some information about the client. However, this approach is also not cache friendly, bound by same origin policy, and imposes additional client-side latency by requiring JavaScript execution to create and manage HTTP cookies.

This document defines a set of new request header fields that allow the client to perform proactive content negotiation [[I-D.ietf-httpbis-p2-semantics](#)] by indicating a list of device and agent specific preferences, through a mechanism similar to the Accept header which is used to indicate preferred response formats.

### 1.1. Notational Conventions

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

This document uses the Augmented Backus-Naur Form (ABNF) notation of [[RFC5234](#)] with the list rule extension defined in [[I-D.ietf-httpbis-p1-messaging](#)], [Appendix B](#). It includes by reference the OWS, field-name and quoted-string rules from that document, and the parameter rule from [[I-D.ietf-httpbis-p2-semantics](#)].

## 2. Client Hint Request Header Fields

A Client Hint request header field is a HTTP header field that is



used by HTTP clients to indicate configuration data that can be used by the server to select an appropriate response. Each one conveys a list of client preferences that the server can use to adapt and optimize the response.

Client Hint request headers share a common syntax. As a convention, those defined in this specification have names prefixed with "CH-", but this is only a convenience.

This document defines a selection of Client Hint request header fields, and can be referenced by other specifications wishing to use the same syntax and processing model.

### **2.1. Hint Values**

Client-Hint field-values consist of either a token or a comma-delimited list of parameters.

`client-hint-value = token | 1#parameter`

When the value is a token, it can either be boolean or a numeric value. Where possible, this form SHOULD be used, so that the hints don't consume too much space in requests.

Boolean values are indicated by the presence of the hint field-name in the request headers. If the hint name is absent in the last message containing the client hint header field, it is considered false.

Numeric values are indicated by the full field-value contents (single value), or by the digits after "=" of the hint name (parameter value), up to the first non-digit character. If the hint does not have an argument, its value is assumed to be 0.

Note that HTTP allows headers with comma-separated values to be conveyed using multiple instances of the same header field; as a result, the hints are collected from all instances of the same header on the message in question before being considered complete. If the same hint is used more than once, then the last hint overrides all previous occurrences, and the final ordering of unique hints is not significant.

### **2.2. Sending Client Hints**

Clients control which Client Hint headers and their respective header fields are communicated, based on their default settings, user configuration and/or preferences. The user may be given the choice to enable, disable, or override specific hints.



The client and server, or an intermediate proxy, may use an opt-in mechanism to negotiate which fields should be reported to allow for efficient content adaption.

### **2.3. Server Processing of Client Hints**

Servers can modify the response sent based upon Client Hints. When doing so, it **MUST** confirm the selection for certain hints and indicate the value of selected resource via corresponding response header. For example, this specification defines "DPR" that corresponds to the "CH-DPR" request header field.

#### **2.3.1. Advertising Support for Client Hints**

Servers can advertise support for Client Hints using the Accept-CH header or an equivalent HTML meta element with http-equiv attribute.

Accept-CH = #token

For example:

Accept-CH: DPR, RW

When a client receives Accept-CH, it **SHOULD** append the Client Hint headers that match the advertised field-values. For example, based on Accept-CH example above, the client would append CH-DPR and CH-RW headers to subsequent requests.

#### **2.3.2. Interaction with Caches**

Client Hints **MAY** be combined with Key ([\[I-D.fielding-http-key\]](#)) to enable fine-grained control of the cache key for improved cache efficiency. For example, the server may return the following set of instructions:

Key: CH-DPR;r=[1.5:]

Above examples indicates that the cache key should be based on the CH-DPR header, and the resource should be cached and made available for any client whose device pixel ratio is 1.5, or higher.

Key: CH-RW;r=[320:640]

Above example indicates that the cache key should be based on the CH-RW header, and the resource should be cached and made available for any request whose display width falls between 320 and 640px.

In absence of support for fine-grained control of the cache key via





the Key header field, Vary response header can be used to indicate that served resource has been adapted based on specified Client Hint preferences.

Vary: CH-DPR

Above example indicates that the cache key should be based on the CH-DPR header.

Vary: CH-DPR, CH-RW

Above example indicates that the cache key should be based on the CH-DPR and CH-RW headers.

### **3. The CH-DPR Client Hint**

The "CH-DPR" header field indicates the client's current Device Pixel Ratio (DPR), the ratio between physical pixels and density independent pixels on the device.

CH-DPR = 1\*DIGIT [ "." 1\*DIGIT ]

### **4. The CH-RW Client Hint**

The "CH-RW" header field indicates the client's current Resource Width (RW), the display width of the requested resource in density independent pixels on the device.

CH-RW = 1\*DIGIT

#### **4.1. Confirming Selected DPR**

The "DPR" header field indicates the ratio between physical pixels and density independent pixels of the selected response.

DPR = 1\*DIGIT [ "." 1\*DIGIT ]

DPR ratio affects the calculation of intrinsic size of the image on the client (i.e. typically, the client automatically scales the natural size of the image by the DPR ratio to derive its display dimensions). As a result, the server must explicitly indicate the DPR of the resource whenever CH-DPR hint is used, and the client must use the DPR value returned by the server to perform its calculations. In case the server returned DPR value contradicts previous client-side DPR indication, the server returned value must take precedence.



The server does not need to confirm resource width (RW) selection as this value can be derived from the resource itself once it is decoded by the client.

## 5. Examples

For example, given the following request header:

```
CH-DPR: 2.0  
CH-RW: 160
```

The server knows that the device pixel ratio is 2.0, and that the intended display width of requested resource is 160px, as measured by density independent pixels on the device.

If the server uses above hints to perform resource selection, it must confirm its selection via the DPR response header to allow the client to calculate the appropriate intrinsic size of the image resource. The server does not need to confirm resource width, only the ratio between physical pixels and density independent pixels of the selected image resource:

```
DPR: 1.0
```

The DPR response header indicates to the client that the server has selected resource with DPR ratio of 1.0. The client may use this information to perform additional processing on the resource - for example, calculate the appropriate intrinsic size of the image resource such that it is displayed at the correct resolution.

### 5.1. Relationship to the User-Agent Request Header

Client Hints does not supersede or replace User-Agent. Existing device detection mechanisms can continue to use both mechanisms if necessary. By advertising its capabilities within a request header, Client Hints allows for cache friendly and proactive content negotiation.

## 6. IANA Considerations

This document defines the "Accept-CH", "CH-DPR", "CH-RW", and "DPR" HTTP request fields, and registers them in the Permanent Message Headers registry.



- o Header field name: CH-DPR
- o Applicable protocol: HTTP
- o Status: Informational
- o Author/Change controller: Ilya Grigorik, [ilya@igvita.com](mailto:ilya@igvita.com)
- o Specification document(s): [this document]
- o Related information: for Client Hints
- o Header field name: CH-RW
- o Applicable protocol: HTTP
- o Status: Informational
- o Author/Change controller: Ilya Grigorik, [ilya@igvita.com](mailto:ilya@igvita.com)
- o Specification document(s): [this document]
- o Related information: for Client Hints
- o Header field name: DPR
- o Applicable protocol: HTTP
- o Status: Informational
- o Author/Change controller: Ilya Grigorik, [ilya@igvita.com](mailto:ilya@igvita.com)
- o Specification document(s): [this document]
- o Related information: for Client Hints
- o Header field name: Accept-CH
- o Applicable protocol: HTTP
- o Status: Informational
- o Author/Change controller: Ilya Grigorik, [ilya@igvita.com](mailto:ilya@igvita.com)
- o Specification document(s): [this document]
- o Related information: for Client Hints

## 7. Security Considerations

The client controls which header fields are communicated and when. In cases such as incognito or anonymous profile browsing, the header can be omitted if necessary.

## 8. Normative References

[I-D.fielding-http-key]

Fielding, R. and M. Nottingham, "The Key HTTP Response Header Field", [draft-fielding-http-key-02](#) (work in progress), February 2013.

[I-D.ietf-httpbis-p1-messaging]

Fielding, R. and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", [draft-ietf-httpbis-p1-messaging-25](#) (work in progress), November 2013.

[I-D.ietf-httpbis-p2-semantics]

Fielding, R. and J. Reschke, "Hypertext Transfer Protocol



(HTTP/1.1): Semantics and Content",  
[draft-ietf-httpbis-p2-semantics-25](#) (work in progress),  
November 2013.

[I-D.nottingham-http-browser-hints]

Nottingham, M., "HTTP Origin and Hop Hints",  
[draft-nottingham-http-browser-hints-05](#) (work in progress),  
February 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate  
Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an  
IANA Considerations Section in RFCs", [BCP 26](#), [RFC 5226](#),  
May 2008.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax  
Specifications: ABNF", STD 68, [RFC 5234](#), January 2008.

#### Author's Address

Ilya Grigorik

Email: [ilya@igvita.com](mailto:ilya@igvita.com)

URI: <http://www.igvita.com/>



