HTTPbis Working Group
Internet-Draft
Intended status: Informational
Expires: January 16, 2014

H. Ruellan
Canon CRF
J. Fujisawa
Canon, Inc.
R. Bellessort
Y. Fablet
Canon CRF

July 15, 2013

# HTTP/2.0: Header Reference Set Definition
# draft-ruellan-reference-set-definition-00

## Abstract

This document describes modifications of the proposed HTTP/2.0 header compression mechanism for the definition of the reference set used to encode a set of headers.

## Status of This Memo



## Copyright Notice

Table of Contents



## 1. Introduction

In HTTP/2.0 [http2.0], HTTP headers are compressed according to the HTTP/2.0 Headers Compression [header-compression] draft. This compression format relies on the previous set of headers to improve compression. Two consecutive sets of headers often comprise a significant number of headers in common. Encoding the differences between those two sets is generally more efficient than encoding the set of headers itself.

The messages exchanged between a client and a server (or between a server and a client) may however comprise different types of header sets: GET requests and POST requests may have more differences than similarities; "200 OK" responses and "304 Not Modified" responses may be very different. In those cases, allowing the encoder to select a specific reference set of headers allows reducing the size of encoded headers.

A new mechanism is proposed in this draft to allow the encoder defining a reference set of headers directly from the header indexing table.

## 2. Overview

### 2.1. Design Principles

This proposal builds on HTTP/2.0 Header Compression [header-compression] to further improve the compactness of HTTP headers representation. It reuses the notion of header group as defined in Header Delta-Compression for HTTP/2.0 [delta2] in a more limited way. First, NO additional indexing structure or storage requirement is added to the existing mechanisms defined in HTTP/2.0 Header Compression [header-compression]. Second, the impact on implementations, particularly on decoder side, should be as limited as possible. This proposal may also bring processing improvements. qdfsq

### 2.2. Outline

Using this proposal, when encoding a new set of headers, the encoder indicates whether it relies on the default reference set (i.e. the previous set of headers) or whether it defines a specific reference set.

A specific reference set is represented as a start index SI and a length L; the reference set is defined as the L headers occurring from index SI in the header indexing table.

Once this specific reference set is defined, it is used in place of the default reference set to encode the new set of headers: only the differences between this specific reference set and the new set of headers are represented.

## 3. Integration Within Current Draft

A new flag, REFERENCE_SET, is defined for the HEADERS frame. This flag uses bit 5 (0x10) and determines whether the default reference set or another one is used for the given Header Block.

The value of 0 indicates that the default reference set (i.e. the previous set of headers) is used and the header block contains the differences between the reference set and the set of headers represented.

The value of 1 indicates that the header block starts by the definition of a specific reference set, which is then followed by the

differences between this specitic reference set and the set of headers represented.

# 4. Detailed Format

## 4.1. Header Block Organization

A header block is organized in two parts:

- o An optional reference set definition. This definition is present only if the REFERENCE_SET flag is set.

- o A list of difference between the reference set and the encoded set of headers. The difference is represented as defined in HTTP/2.0 Header Compression [header-compression].

## 4.2. Reference Set Definition

A reference set is described on 2 bytes.

```
  0                                       1
  0   1   2   3   4   5   6   7   8   9   0   1   2   3   4   5
+-------------------+-------------------------------------------+
|    Length (5)     |             Start Index (11)              |
+-------------------+-------------------------------------------+

                      Reference Set Definition
```

The fields defining a reference set are:

Length: The length of the reference set, encoded on 5 bits. This length is comprised between 0 and 31.

Start Index: The index of the first entry of the reference set in the header table. If the reference set length is strictly greater than zero, the start index is encoded on 11 bits. If the length is equal to zero, the start index value is zero and is encoded on 3 bits.

# 5. Empty Reference Set

If the reference set length is equal to zero, the reference set is empty and the reference set definition is optimized to use only one byte.

```
  0   1   2   3   4   5   6   7
+-------------------+-----------+
|         0         |     0     |
```

```
+-------------------+-----------+
```

Empty Reference Set Definition

This special case may be useful in a number of applications. First, it can allow a more compact encoding of a header set. For instance, measurements have shown that using the empty as the reference set is, for a fair number of test samples, more efficient than using the default reference set for HTTP responses.

Second, using the empty set allows reducing the coding complexity for the encoder. This can be useful for server or intermediaries under a heavy load. It could also be used by lightweight devices that prefer using static encoding strategies.

# 6. Open Questions

## 6.1. Reference Set Removals

The definition of a specific reference set is based on a contiguous set of header entries in the header indexing table. Removing a header from the set is typically encoded using one or two bytes. It may be beneficial to optimize the removal of a header in the case of a specific reference set which would be constituted of several contiguous sets of headers separated by a few other headers not belonging to the set of headers to encode.

For instance, for a sixteen headers reference set, two bytes are needed to define all the removals as an array of booleans. Two bytes is also the cost for encoding one or two removals using HTTP/2.0 Header Compression [header-compression]. Once a reference set of sixteen headers contains more than two headers that should be removed, using an array of booleans is more efficient.

Early experiments show that this approach allows selecting larger reference sets and brings noticeable compaction benefits. Additional investigation should be done to understand how optimized indexing table building would decrease the benefits.

Encoding the removals as an array of booleans has also the advantage of making decoder implementations simpler. As removals are not encoded in the same way as indexed values, there is no problem of handling possible discrepancies between the index values in the reference set and the index values in the header table.

Optimized encoding of the header removal could also be envisioned for the default reference set. In such a case, an ordering of the default reference set headers should be defined. Such an ordering could be based on the indexes of the headers.

Additional study should be conducted to evaluate whether this feature should be a mandatory part of the specific reference set definition or an option. Several ways to switch on/off this feature may be envisioned: using a specific HEADERS frame flag; using a specific bit within the reference set definition. Using a specific HEADERS frame flag would be appropriate if using deletions for the default reference set is envisioned.

### 6.2. Adapted Entries Insertion/Removal

Rules to handle incremental indexing and batch entry removal in case of buffer overflow are defined in HTTP/2.0 Header Compression [header-compression]. New rules may be defined when specific reference sets are used.

Incrementally indexed entries could be inserted just after the last entry in the header indexing table that is part of the specific reference set. If entry removal is needed, removal could start with the first entry after the last incrementally indexed header or just after the last entry that is part of the specific reference set.

Additional study of the processing cost of these rules should be envisioned.

### 6.3. Optimized Reference Set Definition

A single representation of the reference header set is currently specified. This representation generally requires 2 bytes to be encoded. Tests involving a single byte representation have been made and have shown a small benefit. Based on additional tests, a more optimized representation could be defined.

### 6.4. Re-indexing of Already Indexed Headers

If the same header happens in several reference sets, it may be beneficial to insert such header at several places in the header indexing table. While it makes the header indexing table size bigger, it may further improve future reference set definitions.

This approach is already feasible by encoding a header literally several times. It may be optimized by enabling to insert at a new place a header that is already represented as an indexed header.

## 7. Security Considerations

TBD.

## 8. IANA Considerations

This memo includes no request to IANA.

## 9. References

### 9.1. Normative References

[header-compression]
Ruellan, H. and R. Peon, "HTTP/2.0 Header Compression", draft-ietf-httpbis-header-compression-01 (work in progress), July 2013.

[http2.0] Belshe, M., Peon, R., Thomson, M., Ed., and A. Melnikov, Ed., "Hypertext Transfer Protocol version 2.0", draft-ietf-httpbis-http2-04 (work in progress), July 2013.

### 9.2. Informative References

[bohe] Snell, J., "HTTP/2.0 Discussion: Stored Header Encoding", draft-snell-httpbis-bohe-11 (work in progress), July 2013.

[delta2] Peon, R., "Header Delta-Compression for HTTP/2.0", draft-rpeon-httpbis-header-compression-03 (work in progress), March 2013.

## Appendix A. Example

After the transmission of a "200 OK" and "304 Not Modified" responses, the header indexing table is as follows:

| Index | Header Name | Header Value |
|---|---|---|
| 0 | :status | 200 |
| 1 | cache-control | public |
| 2 | content-type | text/html |
| 3 | content-length | 32 |
| 4 | expires | 01 Apr 2014 00:00:00 GMT |

```
|    5    | :status        | 304                      |
+---------+----------------+--------------------------+
|    6    | expires        | 01 Apr 2015 00:00:00 GMT |
+---------+----------------+--------------------------+
```

Initial Header Table

The current reference set is:

```
5    :status: 304
6    expires: 01 Apr 2015 00:00:00 GMT
```

Initial Header Table

The next response is a "200 OK" response. The following headers have to be encoded:

```
:status: 200
cache-control: public
content-type: text/html
content-length: 1034
expires: 01 Apr 2016 00:00:00 GMT
```

Initial Header Table

## A.1. Default Reference Set Example

The following stream illustrates the "200 OK" response header set encoded as reference to the previous message:

```
0x85    (Indexed header, index = 5, removal)
0x86    (Indexed header, index = 6, removal)
0x80    (Indexed header, index = 0)
0x81    (Indexed header, index = 1)
0x82    (Indexed header, index = 2)
0x44    (Literal header with incremental indexing, name index = 3)
0x04    (Header value string length = 4)
1034
0x45    (Literal header with incremental indexing, name index = 4)
0x18    (Header value string length = 24)
01 Apr 2016 00:00:00 GMT
```

Initial Header Table

The size of the encoded header set is 37 bytes.

## A.2. Empty Reference Set Example

The following stream illustrates the "200 OK" response header set encoded as reference to an empty reference set:

```
0x00    (Empty reference set)
0x80    (Indexed header, index = 0)
0x81    (indexed header, index = 1)
0x82    (indexed header, index = 2)
0x44    (Literal header with incremental indexing, name index = 3)
0x04    (Header value string length = 4)
1034
0x45    (Literal header with incremental indexing, name index = 4)
0x18    (Header value string length = 24)
01 Apr 2016 00:00:00 GMT
```

Initial Header Table

The size of the encoded header set is 36 bytes.

### A.3. Specific Reference Set Example

The following stream illustrates the "200 OK" response header set encoded as reference to a specific reference set (Length = 3, Start Index = 0):

```
0x18 0x00 (specific reference set, length = 3, start index = 0)
0x44    (Literal header with incremental indexing, name index = 3)
0x04    (Header value string length = 4)
1034
0x45    (Literal header with incremental indexing, name index = 4)
0x18    (Header value string length = 24)
01 Apr 2016 00:00:00 GMT
```

Initial Header Table

The size of the encoded header set is 34 bytes.

## Authors' Addresses

Herve Ruellan
Canon CRF

EMail: herve.ruellan@crf.canon.fr

Jun Fujisawa
Canon, Inc.
3-30-2 Shimomaruko
Ohta-ku, Tokyo 146-8501
Japan

EMail: fujisawa.jun@canon.co.jp

Romain Bellessort
Canon CRF

EMail: romain.bellessort@crf.canon.fr

Youenn Fablet
Canon CRF

EMail: youenn.fablet@crf.canon.fr