DASH-IF implementation guidelines: MPD Patch

Living Document, 26 February 2025

This version:
https://dashif.org/Guidelines/master/mpd-patch.html

Issue Tracking:
GitHub

Editor:
SVTA DASH-IF Working Group

Table of Contents

§ 1. MPD Patch

§ 1.1. Introduction

This feature description is an update to DASH-IF IOP Guidelines v4.3 [IOP43]. It adds an additional feature, the MPD Patch, a way provide a differential update to a previous dynamic MPD.

§ 1.2. Scenarios

The typical use case for MPD patch is for dynamic manifests with long SegmentTimeline elements due to non-constant segment duration prohibiting efficient compression using repeat count. This happens when the media frame rate is not commensurable with the segment duration.

For example, a 2s average segment duration is not compatible with AAC 48kHz audio, since it corresponds to 93.75 frames leading to a cycle of 4 segments (8s) and the corresponding SegmentTimline pattern:

```
<SegmentTimeline>
  <S t="83498463744000" d="96256" r="2"></S>
  <S d="95232"></S>
  <S d="96256" r="2"></S>
  <S d="95232"></S>
  <!-- continued pattern -->
</SegmentTimeline>
```

For a long sliding window, this results in a huge MPD. With MPD Patch one can instead request a delta document, a Patch, describing the changes relative to a `publishTime`.

§ 1.3. Content Offering Requirements and Recommendations

MPD Patch is essentially only useful in the case of `SegmentTemplate with SegmentTimeline`. It is especially useful when the segment durations are varying leading to long `SegmentTimeline` nodes. It may also help in the case of multiple periods of a dynamic MPD.

MPD Patch should NOT be used for for `SegmentTemplate with $Number$` since such MPDs typically rarely change, meaning that their content including the `publishTime` is the same over a longer period of time.

The `PatchLocation` element in the MPD contains an optional `ttl` attribute providing the availability end time relative to the `publishTime`. It is recommended to use this value, and set it to relatively small number like 1 minute.

The node serving the MPD Patch requests can cache the first response with an updated `publishTime` respect to the referred one, provided that the time difference is less than the `ttl` value.

A client can therefore NOT assume that the Patch response is providing information about the latest publishTime. It follows that he client may need to make more MPD Patch requests to arrive at the live edge.

The server response to a too early request for an MPD Patch, i.e. before there is a new publishTime, should be the same as when asking for a segment before its availability time. That could be `404 Not Found` or `425 Too Early`.

§ 1.4. Client Implementation Requirements and Guidelines

Clients should ignore the `<PatchLocation>` element if not understood. If used, they should make a request for an Patch at the same instant that they would ask for an updated MPD.

If they get a `4XX` response to the Patch request, they should either wait and redo the request, or switch to fetching a full MPD.

§ 1.5. Examples

Below is an example with a `<PatchLocation>` element and `publishTime="2024-04-16T07:34:38Z"`.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:xsi="http://www.w3.org/2001/XMLScher
    xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 DASH-MPD.xsd" id="auto-patch-id"
    profiles="urn:mpeg:dash:profile:isoff-live:2011,http://dashif.org/guidelines/dash-if-simple"
    type="dynamic" availabilityStartTime="1970-01-01T00:00:00Z" publishTime="2024-04-16T07:34
```

```xml
      minBufferTime="PT2S" timeShiftBufferDepth="PT1M" maxSegmentDuration="PT2S">
  <ProgramInformation moreInformationURL="https://github.com/dash-Industry-Forum/livesim-
    <Title>Basic MPD with 640x480@30 video at 300kbp and 48kbps audio</Title>
    <Source>VoD source for DASH-IF livesim2</Source>
  </ProgramInformation>
  <PatchLocation ttl="60">/patch/livesim2/patch_60/segtimeline_1/testpic_2s/Manifest.mpp?publ
  <Period id="P0" start="PT0S">
    <AdaptationSet id="2" lang="en" contentType="audio" segmentAlignment="true" mimeType="a
      <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"></Role>
      <SegmentTemplate media="$RepresentationID$/$Time$.m4s" initialization="$Representation
        <SegmentTimeline>
          <S t="82236135168000" d="96256" r="2"></S>
          <S d="95232"></S>
          <S d="96256" r="2"></S>
          <S d="95232"></S>
          <S d="96256" r="2"></S>
          <S d="95232"></S>
          <S d="96256" r="2"></S>
          <S d="95232"></S>
          <S d="96256" r="2"></S>
          <S d="95232"></S>
          <S d="96256" r="2"></S>
          <S d="95232"></S>
          <S d="96256" r="2"></S>
          <S d="95232"></S>
          <S d="96256" r="2"></S>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="A48" bandwidth="48000" audioSamplingRate="48000" codecs="mp4a.40
        <AudioChannelConfiguration schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configu
      </Representation>
    </AdaptationSet>
    <AdaptationSet id="1" contentType="video" par="16:9" minWidth="640" maxWidth="640" minH
      <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"></Role>
      <SegmentTemplate media="$RepresentationID$/$Time$.m4s" initialization="$Representation
        <SegmentTimeline>
          <S t="154192753440000" d="180000" r="30"></S>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="V300" bandwidth="300000" width="640" height="360" sar="1:1" frameRa
    </AdaptationSet>
```

```
    </Period>
    <UTCTiming schemeIdUri="urn:mpeg:dash:utc:http-xsdate:2014" value="https://time.akamai.co
  </MPD>
```

The segments have an average duration of 2s, so a request 5s after the publishTime results in the following PATCH document with a more complex change to the audio part than to the video part.

```
<?xml version="1.0" encoding="UTF-8"?>
<Patch xmlns="urn:mpeg:dash:schema:mpd-patch:2020"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:mpeg:dash:schema:mpd-patch:2020 DASH-MPD-PATCH.xsd" mpdId=
  <replace sel="/MPD/@publishTime">2024-04-16T07:34:42Z</replace>
  <replace sel="/MPD/PatchLocation[1]">
    <PatchLocation ttl="60">/patch/livesim2/patch_60/segtimeline_1/testpic_2s/Manifest.mpp?pub
  </replace>
  <remove sel="/MPD/Period[@id='P0']/AdaptationSet[@id='2']/SegmentTemplate/SegmentTimeli
  <add sel="/MPD/Period[@id='P0']/AdaptationSet[@id='2']/SegmentTemplate/SegmentTimeline"
    <S t="82236135360512" d="96256"/>
  </add>
  <add sel="/MPD/Period[@id='P0']/AdaptationSet[@id='2']/SegmentTemplate/SegmentTimeline/
    <S d="95232"/>
  </add>
  <add sel="/MPD/Period[@id='P0']/AdaptationSet[@id='2']/SegmentTemplate/SegmentTimeline/
    <S d="96256"/>
  </add>
  <remove sel="/MPD/Period[@id='P0']/AdaptationSet[@id='1']/SegmentTemplate/SegmentTimeli
  <add sel="/MPD/Period[@id='P0']/AdaptationSet[@id='1']/SegmentTemplate/SegmentTimeline"
    <S t="154192753800000" d="180000" r="30"/>
  </add>
</Patch>
```

§ 1.6. Reference Tools

> NOTE: provide status for the following functionalities

- dash.js supports MPD Patch for `SegmentTemplate with SegmentTimeline`. It has been tested towards [livesim2][livesim2] including cases with multiple periods. If a 4XX

response is received, it will switches to ordinary full MPD requests (Daniel to confirm)

- [livesim2][livesim2] supports MPD DASH. There is [Wiki article][livesim2-wiki] describing how it works

- Test Vectors. One can get test vectors from the [DASH-IF instance of livesim2][livesim2-instance], e.g. [https://livesim2.dashif.org/livesim2/patch_60/segtimeline_1/testpic_2s/Manifest.mpd][livesim2-entry]. Use the [urlgen][urlgen] page to generate other test vectors.

- JCCP

§ 1.7. Additional Information

[livesim2]: https://github.com/Dash-Industry-Forum/livesim2 [livesim2-wiki]: https://github.com/Dash-Industry-Forum/livesim2/wiki/MPD-Patch [livesim2-instance]: https://livesim2.dashif.org [livesim2-entry]:https://livesim2.dashif.org/livesim2/patch_60/segtimeline_1/testpic_2s/Manifest.mpd [urlgen]: https://livesim2.dashif.org/urlgen

§ References

§ Normative References

[IOP43]
  Guidelines for Implementation: DASH-IF Interoperability Points. URL: https://dash-industry-forum.github.io/docs/DASH-IF-IOP-v4.3.pdf