Nomor Research GmbH

# Usage Guide for DASH-IF Conformance Software

Nomor Research GmbH
Munich, Germany
info@nomor.de

21 January 2019

Nomor Research GmbH
Brecherspitzstraße 8
81541 München

# Table of Contents



# Table of Figures

# Table of Tables

# 1 Introduction

This document presents the usage guide for the DASH-IF Conformance Software tool. The screenshots of the User Interface (UI) are added to explain how the testing of the DASH content and visualization of the results can be done.

In Section 2, usage guide for the DASH-IF conformance software is presented whereas in Section 3, the live conformance tool usage guide is provided.

# 2 Usage Guide on DASH-IF Conformance Software

Conformance testing can be performed in two ways, namely using user interface and using command line and the following subsections provide details on the usage of these, respectively.

## *2.1 Usage via Web User Interface*

The web-based UI of the DASH-IF Conformance Software tool is shown in Figure 1. This page can be accessed either

1. Publicly from the website, https://conformance.dashif.org or
2. Locally from the directory, Conformance-Frontend/Conformancetest.html

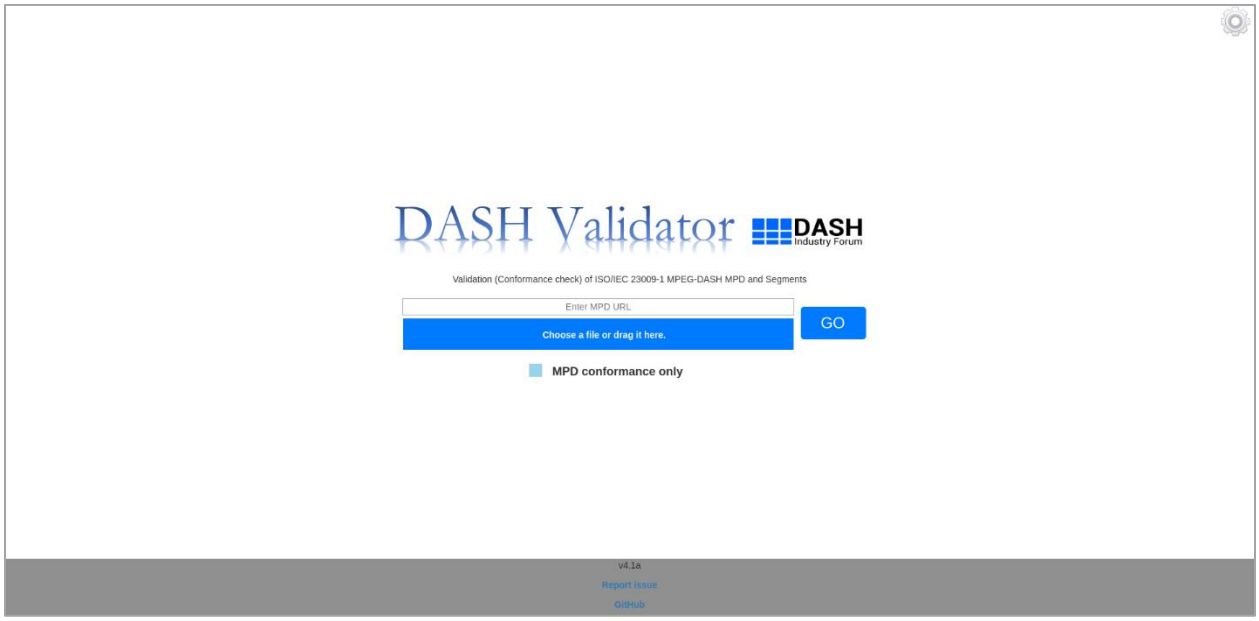


**Figure 1 User Interface of DASH-IF Conformance Software**

### 2.1.1 Starting Conformance Testing

At the start of conformance testing, the user interface has the following components as shown in Figure 2.


**Figure 2: User interface components at the start of conformance testing.**

These components can be defined as the following:

- MPD URL input bar, denoted by 1a
- MPD file upload bar, denoted by 1b
- Extension profile enforcement option selector, denoted by 2
- MPD-only conformance checkbox, denoted by 3
- Go button, denoted by 4

For starting the conformance testing, we have three steps:

1. ***Provide MPD:*** This can be done in two ways, specifically, MPD URL providing and local file uploading. MPD URL can be provided in the *MPD URL input bar* as in Figure 3. Alternatively, a local MPD file can be provided to the *MPD file upload bar* by either clicking on 'Choose a file' option or directly dragging the file and dropping it on the highlighted area shown in Figure 4.

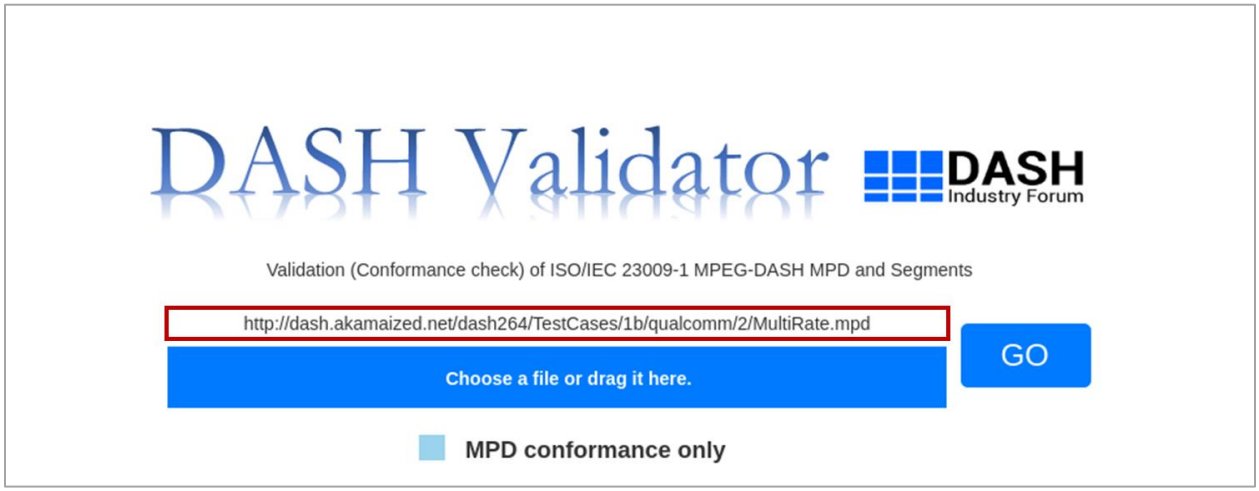


**Figure 3 MPD URL input method**

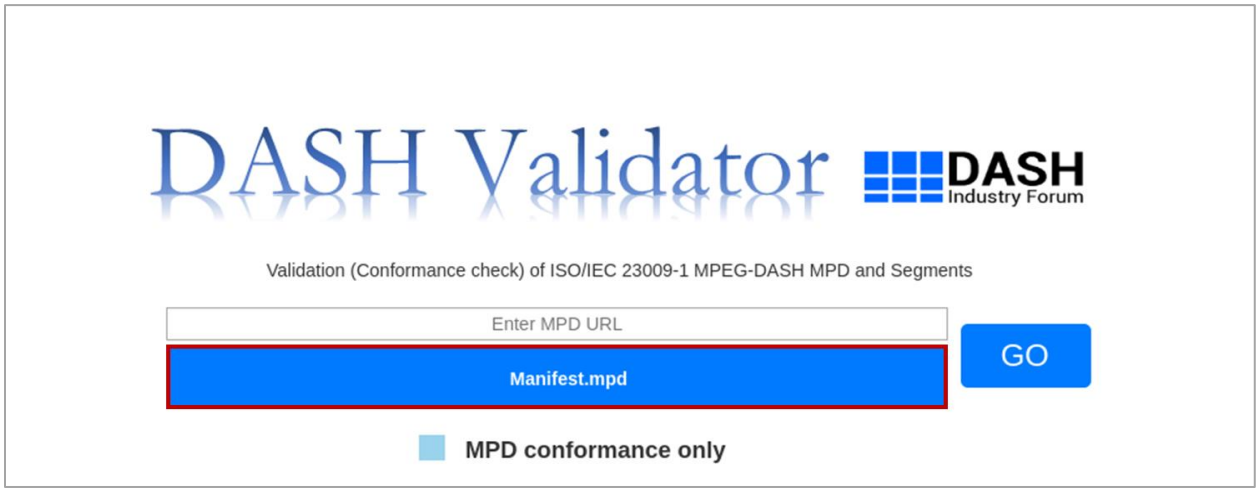


**Figure 4 MPD input using File Upload option**

2. ***Provide optional conformance settings:*** Once the MPD has been provided as described in previous step, one can optionally provide additional conformance settings. These settings correspond to *Extension profile enforcement option selector* and *MPD-only conformance checkbox*. The former – when clicked on – shows additional profiles against which the provided MPD is desired to conformed as shown in Figure 5. Any combination of these profiles can be selected. The latter can be ticked if one desires to only do MPD conformance without conformance testing of media segments pointed to by the MPD. This option is shown in Figure 6.

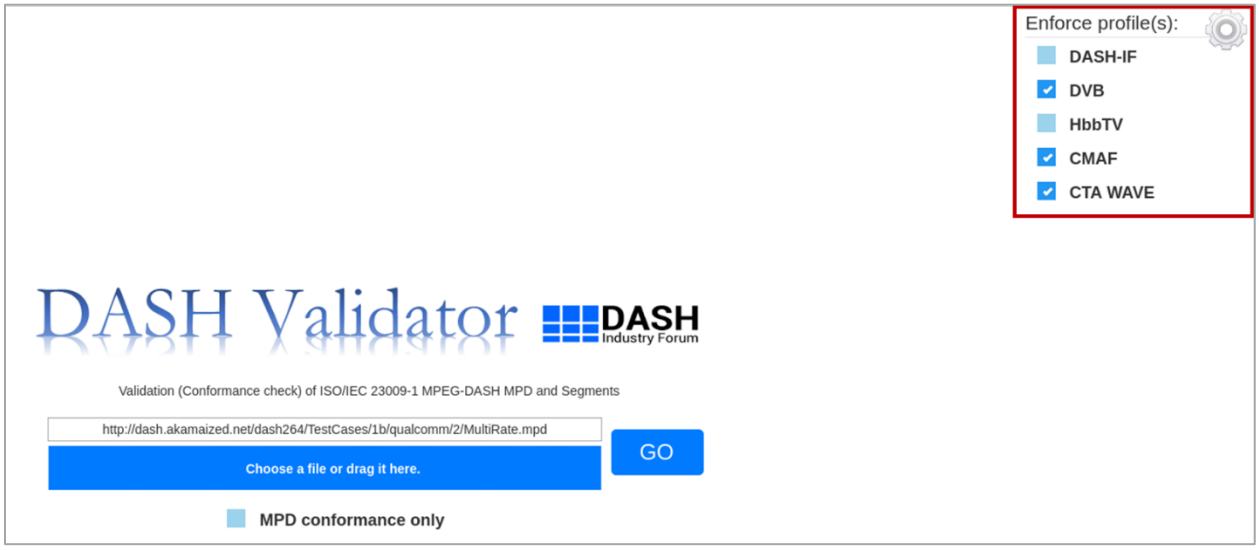


**Figure 5 Optional extension profile enforcement.**

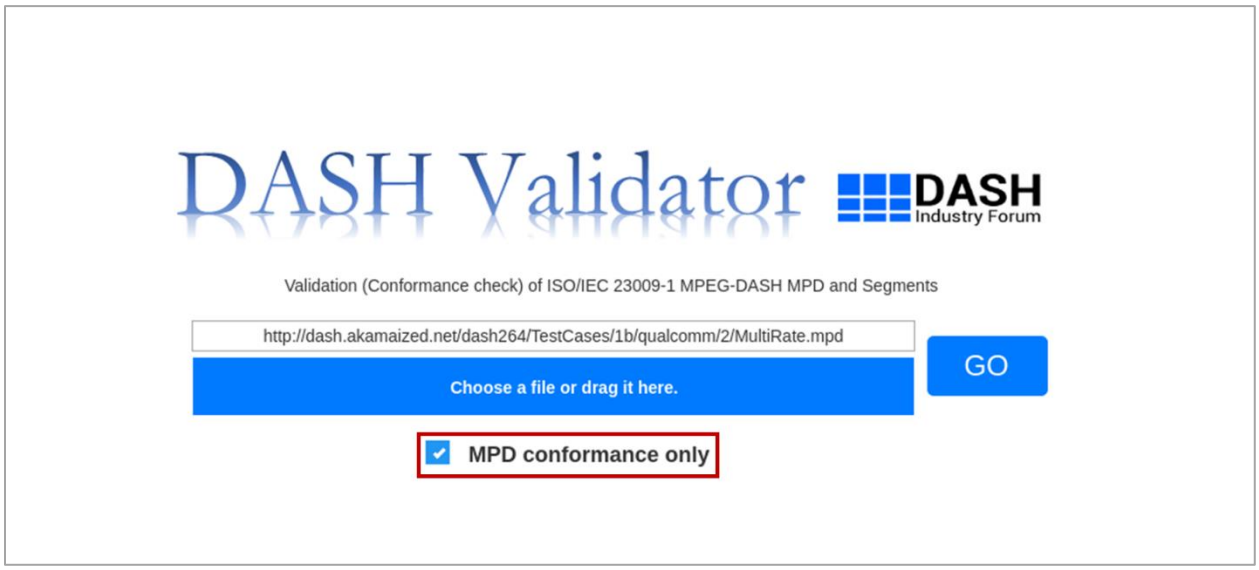


**Figure 6: Optional MPD-only conformance checkbox ticking.**

3. ***Run the test:*** After providing MPD and optionally selecting conformance settings, the last step to start the conformance testing is to click on the *Go button*. When it is clicked on, it will change colour to indicate that the test has started as can be seen in Figure 7.

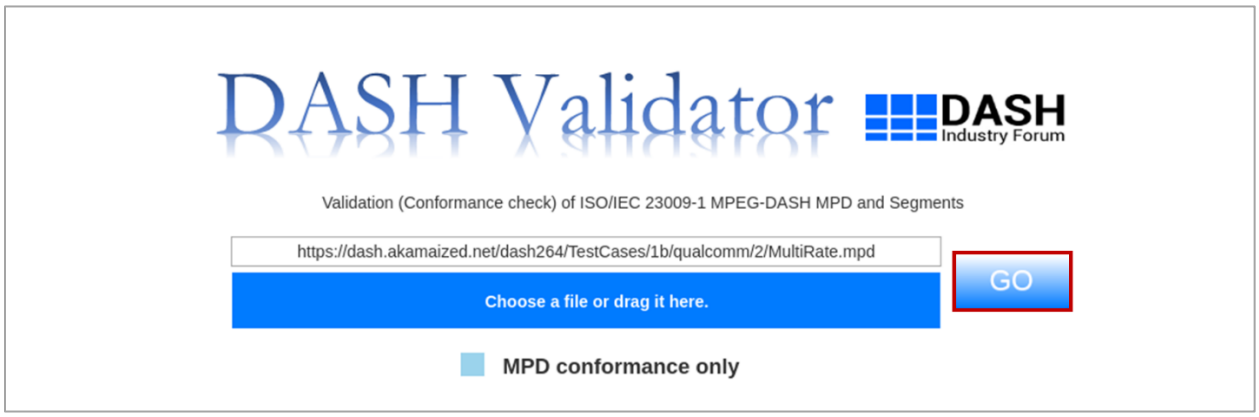


**Figure 7: Running the test.**

### 2.1.2 During Conformance Testing

When conformance testing has started, the user interface looks as provided in Figure 8.


**Figure 8: User interface during conformance testing.**

The changes that appear on the user interface can be defined as the following:

- Progress status text, denoted by 1
- Profiles information signalled in the MPD, denoted by 2
- Feature list link, denoted by 3
- Result tree, denoted by 4

During conformance testing, we have four steps:

1. ***Observe the conformance test progress status:*** The progress information is printed on the user interface once the conformance test starts to update the user. The *progress status text* tells whether the conformance test is in the MPD validation phase or media segment validation phase, and if latter, to what percentage that phase is completed.

2. ***Get information on the profiles:*** *Profiles information signalled in the MPD* is also printed on the user interface, providing that the conformance testing is being done against these profiles.

3. ***View feature list:*** To view the list of features, i.e., all the elements and attributes present in the given MPD, click on the link *Feature list link*. When clicked on, a new tab opens with the feature list as depicted in Figure 9.

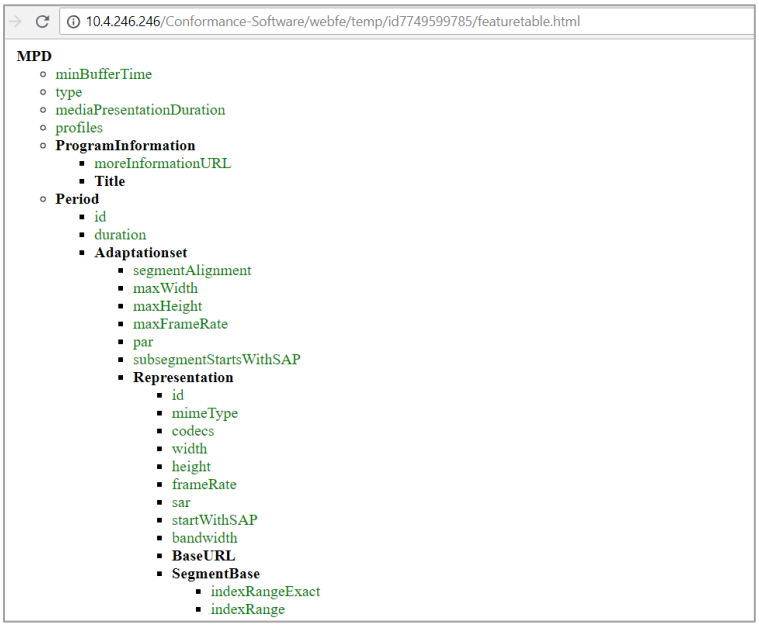


**Figure 9 A look of Feature list**

4. ***Check the results tree:*** The results of the conformance testing get updated continuously after the *Go button* is pressed. These are reflected on the *results tree*. When a result section passes the related conformity checks, that part of the section is ticked off with green. When there is even a single error, that part of the tree is assigned a wrong sign in red. When there is no error but only warnings related to a conformance check part, that part of the tree is assigned a warning sign in yellow. In the intermediate *result tree* shown in Figure 8, the conformance testing parts are ticked off with green, which means that there were not any errors or warnings.
   Each result section is attached a log file which contains the error and/or warning messages concerning the conformance checks or a statement indicating full conformance. The

provided logs can be opened by a single click on the log. The respective report opens in a new tab.
Additionally, each representation part of the tree is attached an “Estimate bitrate” feature, which provides information on possible buffer underrun events when using the specified Minimum Buffer Time (MBT) and bitrate. It can be opened by a single click and they open in a tab.
For a more detailed explanation of the *results tree*, logs, and estimate bitrate reporting, please refer to section 2.1.3.

### 2.1.3 After Conformance Testing

Once the conformance testing is completed, the user interface looks provided in Figure 10.

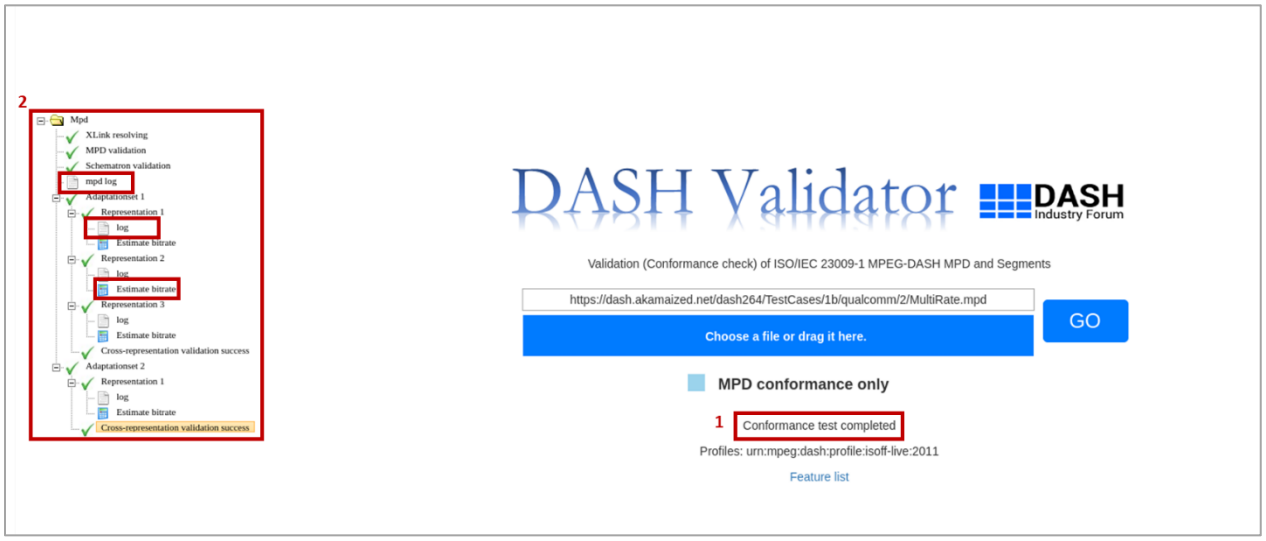


**Figure 10: User interface after conformance test completed.**

The final changes appear on the user interface can be defined as the following:

- Conformance test completion status text, denoted by 1
- Conformance result tree final version, denoted by 2

After the conformance testing:

1. ***Understand the completion of the test:*** Once the conformance testing is completed, it is indicated on the user interface via *conformance test completion status text* stating “Conformance test completed”. Hence, this means all the results are available for the user.
2. ***Check the results tree:*** All the results are available to the user at this stage on the *results tree*. In this step, let us review only the results section of the user interface in detail. The results are formatted in a tree structure. Results can be categorized in three sections as follows:
    a. ***MPD results***

This section consists of Xlink Validation, MPD Validation and Schematron Validation results. When these three parts of this section does not pass the conformance test, the even if *MPD-only conformance checkbox* was not selected, the media segment validation is not performed and the conformance test terminates after this section is completed. In case an enforced profile option is selected from *Extension profile enforcement option selector* and that profile contains MPD validation checks, then this section is extended by adding another item on the tree for that specific enforced profile.

***b. Representation/Segments validation results***

This section consists of representation validation results performed for each individual Representation. In the case that *MPD-only conformance checkbox* is ticked, this validation is not performed; and hence, this section does not appear on tree. Given that *MPD-only conformance checkbox* is not ticked and the *MPD results* section passes the conformance checks, then the conformance checks related to this section are performed.

***c. Cross-representation validation results***

This section consists of cross-representation validation results performed for the Representations within the same Adaptation Set. In the case that *MPD-only conformance checkbox* is ticked, this validation is not performed; and hence, this section does not appear on tree. Given that *MPD-only conformance checkbox* is not ticked and the *MPD results* section passes the conformance checks, then the conformance checks related to this section are performed. In case an enforced profile option is selected from *Extension profile enforcement option selector* and that profile contains cross-representation validation checks, then this section is extended by adding another item on the tree for that specific enforced profile.

Figure 11 highlights the abovementioned sections (a, b, c) of results. When a result section passes the related conformity checks, that part of the section is ticked off with green. Similarly, when there is no error but only warnings related to a conformance check part, that part of the tree is ticked off with green. When there is even a single error, that part of the tree is assigned a wrong sign in red. Figure 11 shows an example where all result sections passed all the checks related to them. Figure 12 shows an example of an unsuccessful MPD validation. As can be seen, it is indicated with a wrong sign in front of the respective result section. Figure 13 shows an example of an unsuccessful representation validation.


**Figure 11: Highlighting all the sections of results.**

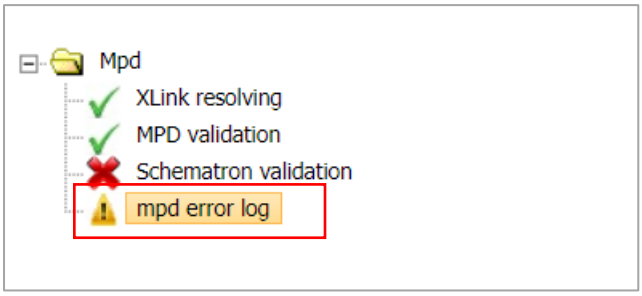


**Figure 12 MPD error results**

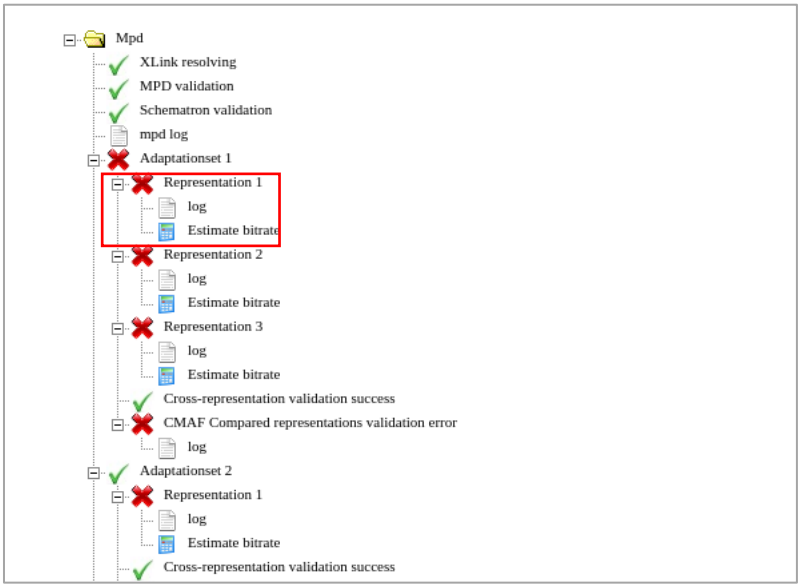


**Figure 13: Representation error results.**

Additionally, each result section is attached a log file which contains the error and/or warning messages concerning the conformance checks or a statement indicating full conformance. The provided logs can be opened by a single click on the log. The respective report opens in a new tab. An example MPD report shown in Figure 14 highlights the abovementioned error from the Schematron validation part of the section. As can be seen the error provides an error statement and points to the location in MPD where this error occurred. Figure 15 shows a sample representation validation error report from. As can be observed, it also consists of detailed information, showing the locations of error (which ISOBMFF box in the representation file) and the error statements.

It should be noticed that the log messages are printed in different colours. As suggested by the “Legend” at the beginning of the reports, blue colour is used for informational messages, yellow is used for indicating a warning and red is used for stating error messages.

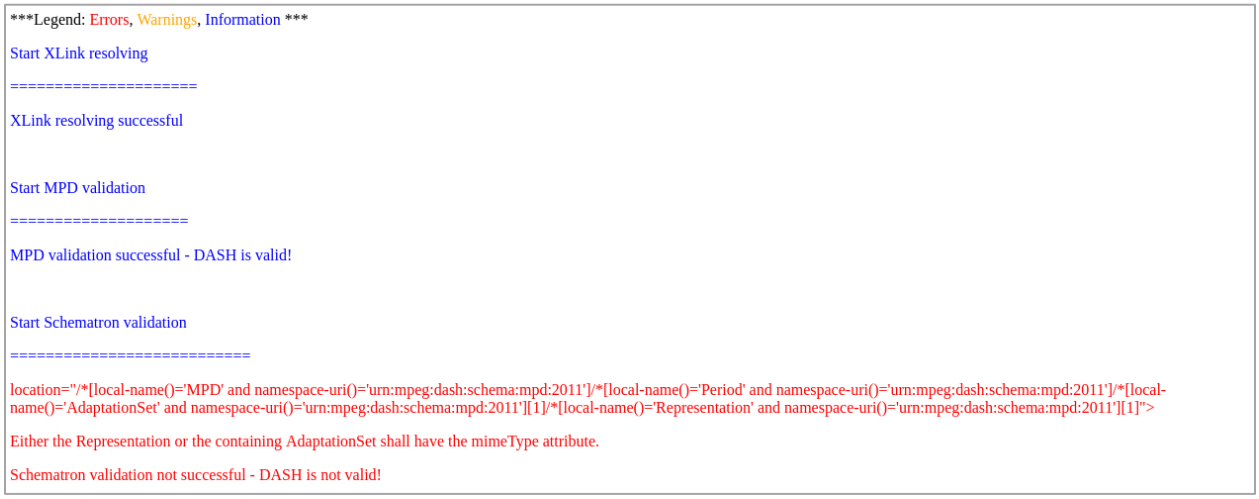


**Figure 14 MPD error report**

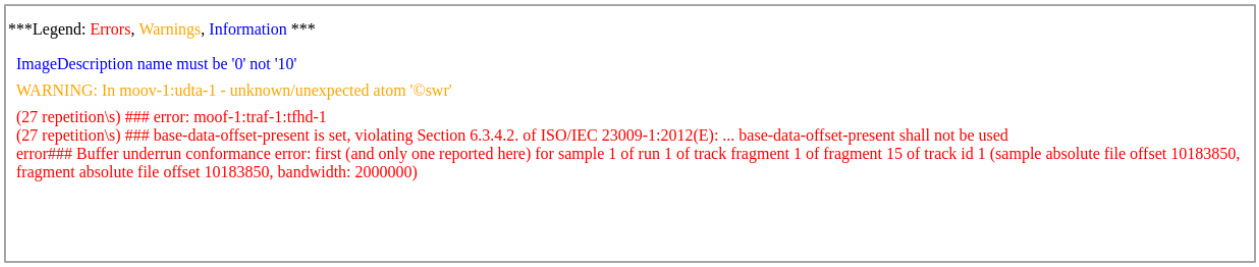


**Figure 15: Representation error report.**

Each representation part of the tree is also attached an “Estimate bitrate” feature, which provides information on possible buffer underrun events when using the specified Minimum Buffer Time (MBT) and Bandwidth. The user interface of this feature is shown in Figure 16.

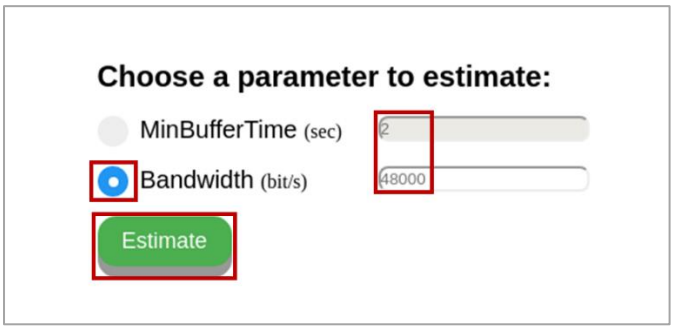


**Figure 16: Estimate bitrate user interface.**

This page provides the upper boundary value for MBT given the Bandwidth and vice versa such that the buffer underrun does not occur when clicked on 'Estimate' button. The input values are set to those found in the MPD for the specific representation. Two examples depicting the estimating MBT and Bandwidth are provided in Figure 17 and Figure 18, respectively.

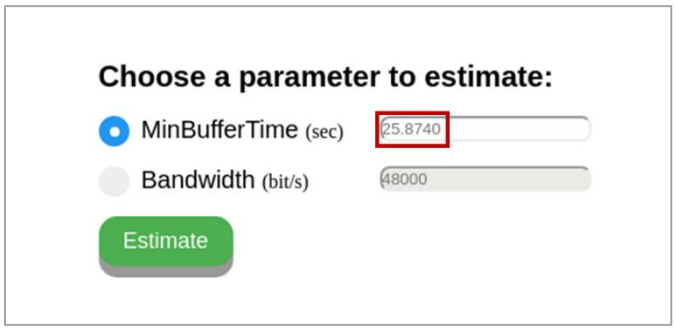


**Figure 17: MBT estimation given bandwidth.**

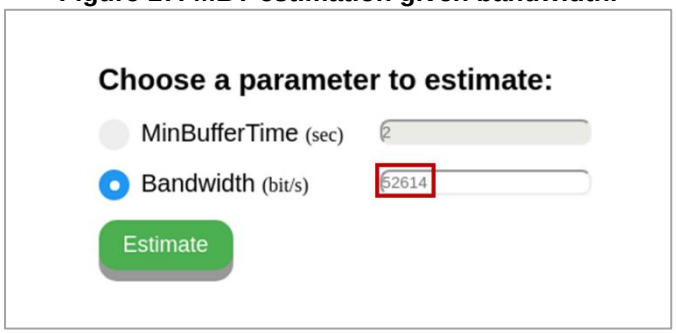


**Figure 18:Bandwidth estimation given MBT.**

## *2.2 Usage via Command Line*

Conformance testing via command line has following components:

- Software tool for retrieving content with command line utility
- Input
  - Mandatory parameters
  - Optional parameters
- Output

Information on these components are as provided in Table 1. Using these components, the command line conformance testing comprises of 1) conformance test command forming, 2) running conformance test and 3) obtaining the results.

**Table 1: Command line components.**

| Components | | Information |
|---|---|---|
| **Software tool** | | • curl<br>• wget |
| **Input** | **Mandatory parameters** | • MPD URL<br>  ○ url<br>• MPD local file<br>  ○ afile |
| | **Optional parameters** | • mpdonly<br>• dashif<br>• cmaf<br>• dvb<br>• hbbtv<br>• ctawave<br>• noerror<br>• nowarning<br>• noinfo |
| **Execution path** | | • The full path of Process.php file in the local server |
| **Output** | | • MPD validation result<br>• Representation/segment validation result<br>• Cross-representation validation result |

### 2.2.1 Forming Conformance Test Command

When forming the command one of the two supported software tool should be used and the following subsections provide explanations and examples on each, respectively.

Another important note is to provide at least the mandatory input parameters. The mandatory parameter is regarding the MPD input, which can be provided either in the form of a URL or in the form of a full local path.

Lastly, as the name suggests optional input parameters are optional. Any combination of these parameters can be provided as input in the command. These parameters can be divided into three categories depending on their functionalities:

- ***MPD-only conformance testing:*** This option can be used if one desires to only do MPD conformance without conformance testing of media segments pointed to by the MPD. The enabler of this functionality is:
  - mpdonly
- ***Extension profile enforcement:*** This functionality provides additional extension profiles against which the conformance of the provided MPD is desired to be performed. The enablers for this are:
  - dashif – DASH-IF profile extension

- cmaf – CMAF profile extension
- dvb – DVB profile extension
- hbbtv – HbbTV profile extension
- ctawave – CTAWAVE profile extension

- ***Output result suppression:*** If one desires to obtain reports with specific type of messages, such as only error messages or no information messages, this feature provides the suppression of specific category of the message types. The enablers of this functionality are as follows:
  - noerror – Suppress error messages in the reports
  - nowarning – Suppress warning messages in the reports
  - noinfo – Suppress informational messages in the reports

#### 2.2.1.1 Using curl

When using curl, the input parameters are provided with "-d" flag or "-F" flag in case of MPD URL providing and MPD local file uploading, respectively.

***MPD URL***

For this case, "url" parameter is mandatory to provide the MPD file location from which the MPD file will be retrieved. Figure 19 and Figure 20 show the two syntax options for the command forming by providing MPD URL.

```
curl -d 'url="<MPD_URL>"' -d <option1> -d <optionN> <execution_path>
```

**Figure 19: Conformance test curl command syntax with separate -d flags in case of MPD URL usage.**

```
curl -d 'url="<MPD_URL>"&<option1>&<option2>' <execution_path>
```

**Figure 20: Conformance test curl command syntax with concatenated options in case of MPD URL usage.**

In both cases the <MPD_URL> is provided between quotation marks. As can be seen the difference between these two syntaxes is the way of providing the input parameters. In the former one each option is provided separately, therefore before each option "-d" flag is present. In the latter one options are concatenated by the use of "&" and therefore only one "-d" flag is present.

Two examples regarding both syntaxes are provided below.

*curl -d 'url="http://dash.akamaized.net/dash264/TestCases/1a/qualcomm/2/MultiRate.mpd"' –d dashif –d nowarning -d cmaf -d mpdonly http://localhost/DASH-IF-Conformance/Utils/Process.php*

*curl -d 'url="http://dash.akamaized.net/dash264/TestCases/1a/qualcomm/2/MultiRate.mpd"& dashif &noerror' http://localhost/DASH-IF-Conformance/Utils/Process.php*

***MPD File Upload***

For this case, “afile” parameter is mandatory to provide the MPD file location from which the MPD file will be retrieved. Figure 21 and Figure 22 show the two syntax options for the command forming by providing a local path of the MPD.

```
curl -F 'afile=@<MPD_location>' -F '<option>=1' -F  '<option>=1' <execution_path>
```

**Figure 21: Conformance test curl command syntax with separate -F flags in case of MPD file upload usage.**

```
curl -F 'afile=@<MPD_location>' -F '<option>=1&<option>=1' <execution_path>
```

**Figure 22: Conformance test curl command syntax with concatenated options in case of MPD file upload usage.**

As can be seen the difference between these two syntaxes is the way of providing the input parameters. In the former one each option is provided separately, therefore before each option “-F” flag is present. In the latter one, options are concatenated by the use of “&” and therefore only one “-F” flag is present.

Two examples regarding both syntaxes are provided below.

*curl -F 'afile=@/home/Documents/Manifest.mpd' -F 'mpdonly=1' –F ‘dashif=1’ F ‘cmaf=1’ -F ‘nowarning=1’ http://localhost/DASH-IF-Conformance/Utils/Process.php*

*curl -F 'afile=@/home/Documents/Manifest.mpd' -F 'mpdonly=1&dashif=1&cmaf=1’ http://localhost/DASH-IF-Conformance/Utils/Process.php*

### 2.2.1.2 Using wget

When using wget, the syntax is as provided in Figure 23. It should be noted that “--post-data” is used to provide all the necessary input. <MPD_URL> corresponds to the URL at which the MPD file resides. The optional input parameters can be provided in a concatenated form in which case they are separated by “&” symbol.

```
wget --post-data 'url="<MPD_URL>”&<option1>&<option2>' <execution_path>
```

**Figure 23: Conformance test wget command syntax with separate -d flags in case of MPD URL usage.**

One example usage of this is provided below.

*wget --post-data 'url=http://dash.akamaized.net/dash264/TestCases/1a/qualcomm/2/MultiRate.mpd&mpdonly' http://localhost/DASH-IF-Conformance/Utils/Process.php*

### 2.2.2 Running Conformance Test

A command line tool of preference should be opened and the directory should be changed to the DASH-IF-Conformance project directory. Here, the formed command should be typed. When pressed “Enter”, conformance testing starts.

### 2.2.3 Obtaining Conformance Test Results

After the complete conformance testing finishes, the results are printed on the command line as shown in Figure 24.

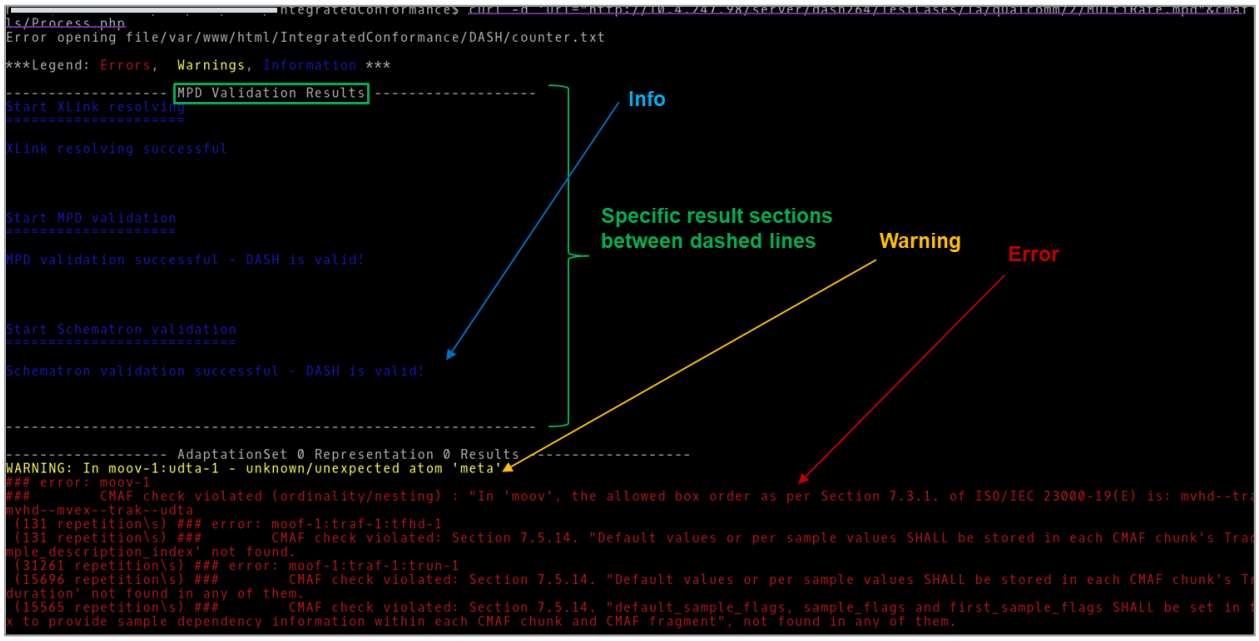


**Figure 24: Command line output.**

The output results are provided in sections assigned with well-defined names and separated by dashed-lines. The results are printed in different colors. As suggested by the “Legend” at the beginning of the complete report, blue colour is used for informational messages, yellow is used for indicating a warning and red is used for stating error messages. The results comprise of:

- **MPD results**
  This section consists of Xlink Validation, MPD Validation and Schematron Validation results. When these three parts of this section does not pass the conformance test, the even if *MPD-only conformance testing* option was not provided, the media segment validation is not performed and the conformance test terminates after this section is completed. In case an enforced profile option is provided from *Extension profile enforcement* and that profile contains MPD validation checks, then this section is extended by adding another item on the tree for that specific enforced profile.
- **Representation/Segments validation results**

This section consists of representation validation results performed for each individual Representation. In the case that *MPD-only conformance* is provided, this validation is not performed; and hence, this section does not appear on command line. Given that *MPD-only conformance* option is not provided and the *MPD results* section passes the conformance checks, then the conformance checks related to this section are performed.

- **Cross-representation results**
  This section consists of cross-representation validation results performed for the Representations within the same Adaptation Set. In the case that *MPD-only conformance* option is provided, this validation is not performed; and hence, this section does not appear on command line. Given that *MPD-only conformance* option is not provided and the *MPD results* section passes the conformance checks, then the conformance checks related to this section are performed. In case an enforced profile option is selected from *Extension profile enforcement* and that profile contains cross-representation validation checks, then this section is extended by adding another item on the command line for that specific enforced profile.

# 3 Usage Guide on Live Conformance Tool

In this section, the usage guide for Live Conformance tool or Dynamic Service Validator is presented. The web-based UI of the Dynamic Service Validator tool is shown in Figure 25.

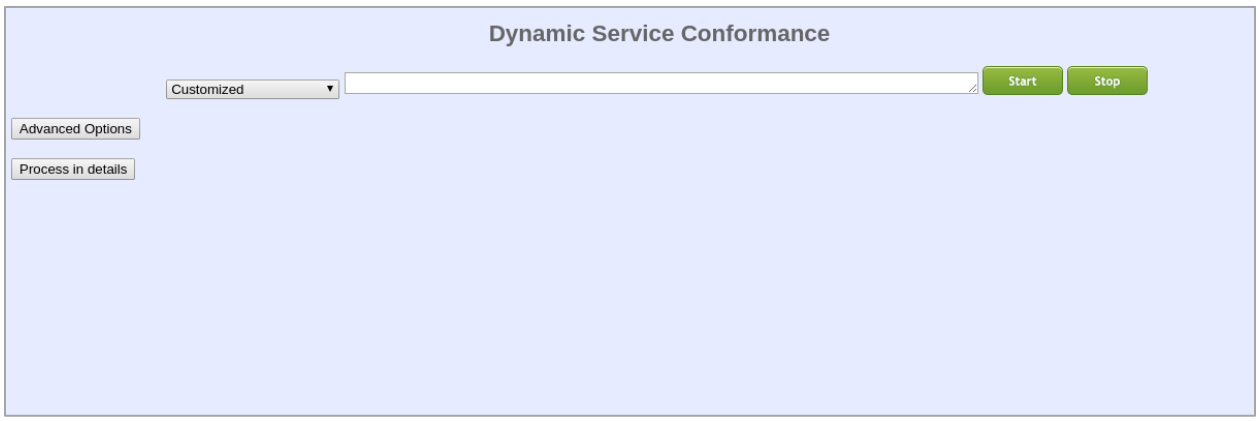


**Figure 25: Live conformance tool user interface.**

### 3.1.1 Accessing Live Conformance Tool

This page can be accessed either

1. From DASH-IF Conformance User Interface via https://conformance.dashif.org or Conformance-Frontend/Conformancetest.html
2. Publicly from http://vm1.dashif.org/DynamicServiceValidator
3. Locally from the directory DynamicServiceValidator/index.html

The first access option is triggered when a dynamic-type MPD is detected on the DASH-IF Conformance software. When this happens, the user interface provides a link to the Dynamic

Service Validator user interface as shown in Figure 26. When clicked on the link the user interface opens up in a new tab with MPD URL already input to the input bar of the user interface as depicted in Figure 27.

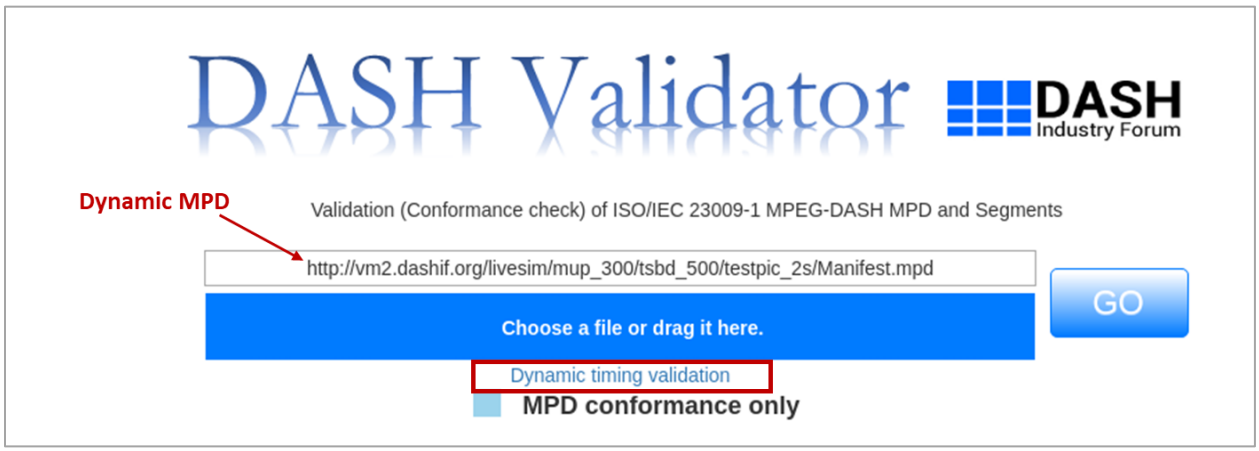


**Figure 26: Triggering dynamic validation tool from DASH-IF conformance user interface.**


**Figure 27: Dynamic service validator when accessed from DASH-IF conformance user interface.**

The second and third access options are simply opening the user interface via the appropriate URL. In this case, the page does not assign an MPD URL to the URL input bar rather leaves it empty, just as in Figure 25.

### 3.1.2 Starting Live Conformance Testing

At the start of conformance testing, the user interface has the following components as shown in Figure 28.

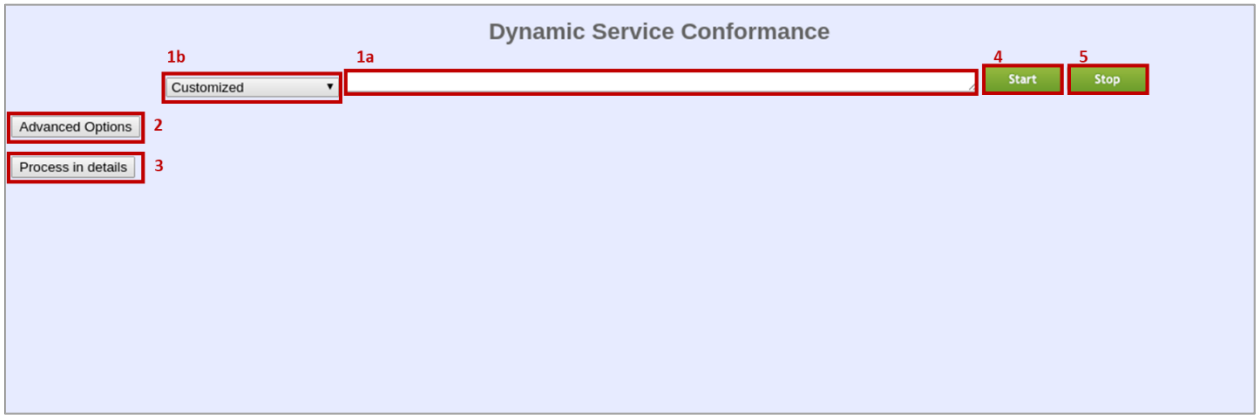


**Figure 28: Dynamic service validation user interface components at the start of conformance testing.**

These components can be defined as the following:

- MPD URL input bar, denoted by 1a
- Sample MPD dropdown menu, denoted by 1b
- Optional conformance settings, denoted by 2
- Detailed process section, denoted by 3
- Start button, denoted by 4
- Stop button, denoted by 5

For starting the live conformance testing, we have at most three steps:

1. **Provide MPD:** If the access to this user interface is done from the DASH-IF Conformance user interface, this step can be skipped, since the MPD URL will be automatically pasted to the *MPD URL input bar*. If this is not the case, there are two ways of providing an MPD.
    a. Typing the URL at which MPD resides
    b. Choosing a sample MPD from the sample MPD dropdown menu

    The user interface when these two options are used is shown in Figure 29 and Figure 30, respectively.


**Figure 29: MPD providing via MPD URL typing.**


**Figure 30: MPD providing via using sample MPDs.**

2. **Set optional parameters:** Before running the conformance testing, one can set optional parameters from *optional conformance settings* part of the user interface. When clicked on the user interface looks as shown in Figure 31.


**Figure 31: Optional conformance testing parameters section of the user interface.**

*RTT correction:* For dynamic live services, the segment availability start time (SAST), segment duration (SD) and segment availability end time (SAET) are important concepts for both the segment life time on the server and fetching DASH segments from the server as a client. in the case that the download of the current segment takes long time such that the time advance reaches the SAET or close to SAET of the next segment, the client can still request this next segment as it is in the availability time window. In this regard, the transmission delay of the request may exceed the announced SAET, which results in 404 Not Found message. Round-trip time (RTT) option is provided to avoid this problem. As can be seen from Figure 31, RTT can be manually set to a value in milliseconds or can be decided to be not applied when clicked on the radio button of “None”. By default, RTT of 300 milliseconds is applied.

*Dynamic clock skew correction:* DASH provides many possibilities to assure the synchronization, however there is always a possibility that the synchronization is not exactly trustable caused by problems like real-time process load at the server and server application not running on real-time operating system. In these cases, the SAST timing information would not be accurate, resulting in early requests on the server side. Taking

DASH interoperability guidelines into account, a safety margin for such possibility is introduced via this option. When ticked, a clock synchronization margin will be applied for testing SAST and SAET.

3. **Run live conformance testing:** After MPD URL providing and optionally conformance parameters setting, conformance testing can be run by clicking on the *start button*.

### 3.1.3 Observing Conformance Testing Results

The test results are provided in the *Detailed process section*. When clicked on, the user interface looks like as shown in Figure 32. Results are presented in different sections on the webpage and they are titled as:

1. 'Response information for the MPD request' (displaying MPD fetch and publish times, number of available segments),
2. 'Overall progress of segment requests' (displaying number of successful checks, mean RTT and clock skew),
3. 'Response information for segment requests' (displaying the status of availability start time and end time checks - Status: OK or Not Found).
4. 'FAIL' or 'PASS' (displaying the overall conformance testing result)

"Status: OK" indicates the conformity for the availability start time checks as the segment was available at the availability start time signaled by the MPD. "Status: Not Found" indicates the conformity for the availability end time checks as the segment was not available at the availability end time signaled by the MPD. It should be noted that even a single 404 Not Found message results in 'FAIL' status whereas if there is no 404 Not Found message the test status is displayed as 'PASS'.

Additionally, at any time of conformance testing, one can stop the test by clicking on the *stop button*.

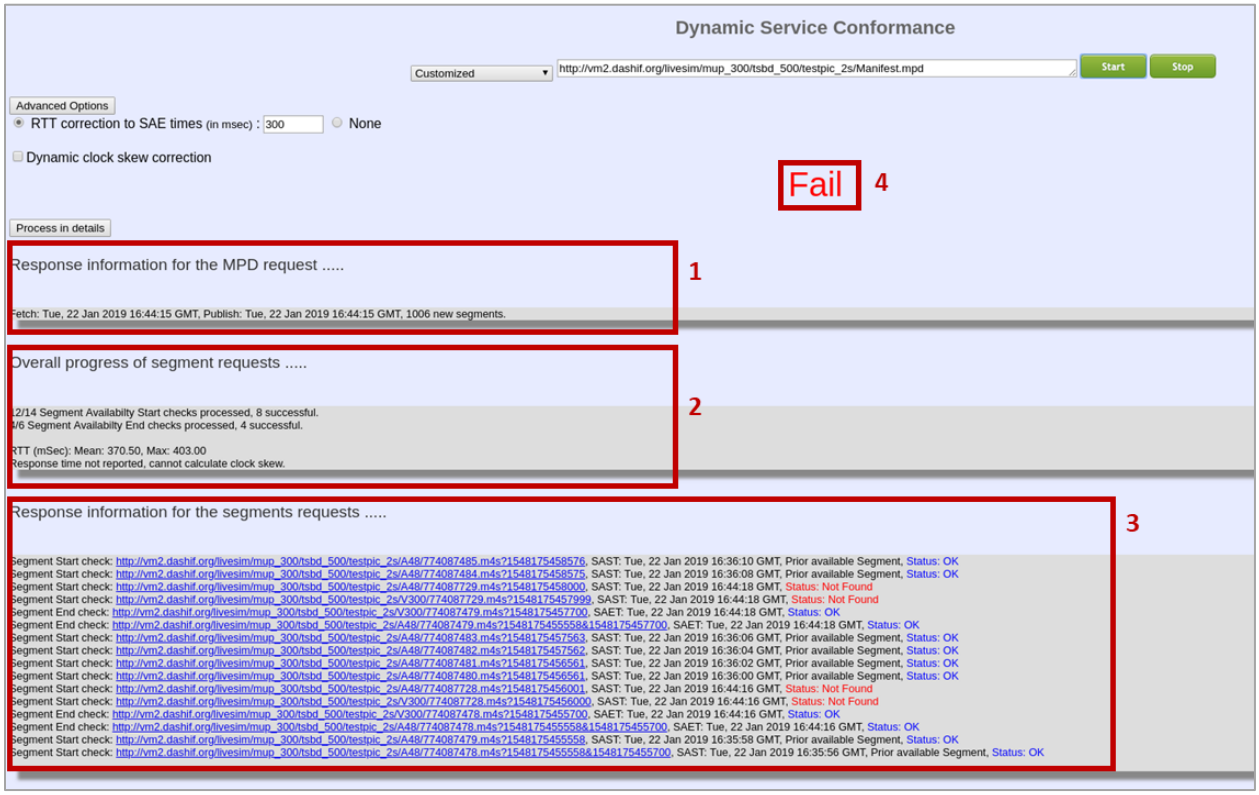


**Figure 32 Results of the live conformance**