The Advanced Television Systems Committee, Inc., is an international, non-profit organization developing voluntary standards and recommended practices for digital television. ATSC member organizations represent the broadcast, broadcast equipment, motion picture, consumer electronics, computer, cable, satellite, and semiconductor industries. ATSC also develops digital television implementation strategies and supports educational activities on ATSC standards. ATSC was formed in 1983 by the member organizations of the Joint Committee on Inter-society Coordination (JCIC): the Electronic Industries Association (EIA), the Institute of Electrical and Electronic Engineers (IEEE), the National Association of Broadcasters (NAB), the National Cable Telecommunications Association (NCTA), and the Society of Motion Picture and Television Engineers (SMPTE). For more information visit www.atsc.org.

Note: The user's attention is called to the possibility that compliance with the standard affected by these changes may require use of an invention covered by patent rights. By publication of the corresponding standard, no position is taken with respect to the validity of this claim or of any patent rights in connection therewith. One or more patent holders have, however, filed a statement regarding the terms on which such patent holder(s) may be willing to grant a license under these rights to individuals or entities desiring to obtain such a license. Details may be obtained from the ATSC Secretary and the patent holder.

Implementers with feedback, comments, or potential bug reports relating to this document may contact ATSC at https://www.atsc.org/feedback/.

Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tracking Initial Changes to A/344:2021 CS Working Draft</td>
<td>08 April 2021</td>
</tr>
</tbody>
</table>
# Table of Contents

1. **SCOPE** ................................................................................................................................. 5  
   1.1 Introduction and Background 5  

2. **SCHEMA FIXES** ..................................................................................................................... 5  
   2.1 Scope 5  
   2.2 Rationale for Changes 5  
   2.3 Compatibility Considerations 6  
   2.4 List of Changes 6  
      2.4.1 Mark Unused Request 6  
      2.4.2 Cache Request 6  
      2.4.3 Cache Request DASH 6  
      2.4.4 Event Stream APIs 7  
      2.4.5 Launch App Request 7  
      2.4.6 Service Guide Query Response 7  
      2.4.7 Notify Caption Display Prefs 8  
      2.4.8 Query Dialog Enhancement Preference Request and Response 8  
      2.4.9 Launch App Request 8  
      2.4.10 Query Device Info Response 9  
      2.4.11 Cancel Request 9  
      2.4.12 Example JSON Files Changed 9  

3. **DIALOG ENHANCEMENT API JSON SCHEMA FILE MISMATCH** ......................................... 11  
   3.1 Scope 11  
   3.2 Rationale for Changes 11  
   3.3 Compatibility Considerations 11  
   3.4 List of Changes 11  
      3.4.1 Text of A/344 11  
      3.4.2 Associated Resources 12  
      3.4.3 (Informative) Additional Information 12  

4. **DEVICE INPUT BAAPPEAR PROPERTY DESCRIPTION** ...................................................... 14  
   4.1 Scope 14  
   4.2 Rationale for Changes 14  
   4.3 Compatibility Considerations 14  
   4.4 List of Changes 14  

5. **SECTION 7 REWRITE** ........................................................................................................... 15  
   5.1 Scope 15  
   5.2 Rationale for Changes 15  
   5.3 Compatibility Considerations 15  
   5.4 List of Changes 15  

6. **REMOVE JSON RPC BATCH MODE REQUIREMENT** ........................................................... 16  
   6.1 Scope 16  
   6.2 Rationale for Changes 16  
   6.3 Compatibility Considerations 16  
   6.4 List of Changes 16  

7. **URI CLARIFICATION** ........................................................................................................... 17  
   7.1 Scope 17
7.2 Rationale for Changes 17
7.3 Compatibility Considerations 17
7.4 List of Changes 17

8. S38-332R0: DATE-TIME FORMAT UPDATES ................................................................. 20
   8.1 Scope 20
   8.2 Rationale for Changes 20
   8.3 Compatibility Considerations 20
   8.4 List of Schema Change Instructions 21

9. S38-335R0: CONTENT RECOVERY STATE API SCHEMA FIX ........................................ 21
   9.1 Scope 21
   9.2 Rationale for Changes 21
   9.3 Compatibility Considerations 21
   9.4 List of Schema Change Instructions 21
A/344:2021 Revision Change Log

1. SCOPE
This document describes one or more changes to the ATSC 3.0 Interactive Content candidate standard. These changes have been accepted by the S38 Specialist Group on Interactive Content but are pending full TG3 and membership approval. Readers are cautioned that these changes may be amended in the future but are encouraged to provide feedback and comments.

1.1 Introduction and Background
This document describes changes or updates to the ATSC 3.0 Interactive Content Candidate Standard. Each section identifies a single change including the scope, rationale, and backwards compatibility of the change. The amendment from which these changes were made is referenced by the S38 document number and that document number is used as the author when making changes to the working draft. This change log may also include change instructions for schemas since these cannot be easily tracked through red-line processes.

Note that this change log represents a change in process from the originally provided change logs. The first six sections (Sections 2 through 7) provide the entire text of each amendment, including all change instructions. The process was streamlined by removing the change instructions and using the document number to track changes in the working draft document. In addition, the working draft amendment resulting in the changes is referenced in this document to allow readers to refer back to the original if so desired.

For each of the subsections below, unless otherwise noted, editing instructions are given in italics, inserted text, tables, and drawings are shown in red; deletions of existing text are shown in red strikeout. The text “[ref]” indicates that a cross reference to a cited referenced document should be inserted. Where possible, the reference will be to a reference existing in the current standard but highlighted in red. It is often the case that these references will not need to be changed but should be given special attention to verify correctness.

The A/344 standard maintains a “revision log” of its included APIs from revision to revision by listing the changes in Table 9.1. In addition, each revision includes an Annex which captures the API from the previous edition in unchanged form. By maintaining the previous API definition in the document, implementers may look at the history of each API.

2. SCHEMA FIXES

2.1 Scope
Testing activities of the A/344 Interactive Environment APIs has led to issues being identified between the A/344 standard document and the corresponding schemas and examples. The collected issues and the resulting changes to the document, schemas and examples are captured in this section.

2.2 Rationale for Changes
The discrepancies between the document, schemas and examples have led to confusion in the implementation community. The changes documented here are intended to make the standard clearer and easier to understand and implement. These changes should also result in a more consistent deployed base of interactive environments on receivers.
2.3 Compatibility Considerations
The table provided below captures each change and the backward compatibility consideration of each. Every attempt has been made to keep the change to the schema backward compatible with the previous implementation, however, this may not be possible in all cases.

2.4 List of Changes
The following subsections provide the list of schemas impacted and the resultant changes made to either the schema, the A/344 working draft or any relevant examples.

2.4.1 Mark Unused Request
Schema: org.atsc.cache.markUnused-request-20210127.json
Issue: All Mark Unused requests must specify an Application Context Cache address which is relative and, thus, the format in the schema should be defined as "uri-reference" while it is currently defined as "uri".
Changes:
1) Schema modified to change the format of "elementUri" from "uri" to "uri-reference".
2) Date of schema file updated to "20210405".
3) A/344 working draft updated to reflect schema file name change.
4) Updated the schema file reference in all example JSON files.
Compatibility: Since all references to the Application Context Cache were very likely relative and the format was only recently added, it is unlikely that implementations are using full URLs instead of relative URLs. Constraining the field will likely have little impact on implementations.

2.4.2 Cache Request
Schema: org.atsc.CacheRequest-request-20210127.json
Issue: CacheRequest requests can specify an Application Context Cache path as a target which is relative, "targetUrl", and a list of relative paths, "urls", which are intended to be concatenated to either the source or target, depending on the goals of the request. Thus, the format of both the target and the relative paths in the schema should be defined as "uri-reference" while they are currently defined as "uri".
Changes:
1) Schema modified to change the formats of "targetUrl" and "urls" from "uri" to "uri-reference".
2) Date of schema file updated to "20210405".
3) A/344 working draft updated to reflect schema file name change.
4) Updated the schema file reference in all example JSON files.
Compatibility: The current document refers to the element formats as relative URLs and, specifically, "uri-reference" in the semantic table. This change makes the schema consistent with the intent. In addition, the "uri" format was added when the schemas were separated into unique files so have only been publicly documented since March 2021. Constraining the field will likely have little impact on implementations.

2.4.3 Cache Request DASH
Schema: org.atsc.CacheRequestDASH-request-20210127.json
Issue: Cache Request DASH requests can specify an Application Context Cache path as a target which is relative, "targetUrl", depending on the goals of the request. Thus, the format of the target path in the schema should be defined as "uri-reference" while it is currently defined as "uri".

Changes:
1) Schema modified to change the format of "targetUrl" from "uri" to "uri-reference".
2) Date of schema file updated to "20210405".
3) A/344 working draft updated to reflect schema file name change.
4) Updated the schema file reference in all example JSON files.

Compatibility: The current document refers to the element formats as relative URLs and, specifically, "uri-reference" in the semantic table. This change makes the schema consistent with the intent. In addition, the "uri" format was added when the schemas were separated into unique files so have only been publicly documented since March 2021. Constraining the field will likely have little impact on implementations.

2.4.4 Event Stream APIs

Schemas: org.atsc.eventStream.event-notification-20210127.json
org.atsc.eventStream.subscribe-request-20210127.json
org.atsc.eventStream.unsubscribe-request-20210127.json

Issue: The semantic definition of "schemeIdUri" does not explicitly refer to A/337 where it is authoritatively specified.

Changes:
1) Added the following sentence to the semantic description of "schemeIdUri": "The syntax of the schemeIdUri shall comply with the syntax of AEI.EventStream@schemeIdUri as defined in [4]."
2) Fixed the reference to the schema file in A/344 working draft which was previously "org.atsc.eventStream.event-20210127.json".

Compatibility: There is no change to the normative schema. The document semantics have been updated to clearly reference the authoritative definition of the "schemeIdUri" element.

2.4.5 Launch App Request

Schemas: org.atsc.launchApp-request-20210127.json

Issue: The semantic definition of "appId" included a reference to A/331 with Amendment 1 parenthetically. This is no longer true with A/331:2021 referenced and may soon be wrong if a new A/331:2021 amendment is passed.

Changes:
1) Removed the "with amendment 1" parenthetical from the semantic description of "appId".

Compatibility: There is no change to the normative schema. The document semantics have been updated to clearly reference the authoritative definition of the "appId" element.

2.4.6 Service Guide Query Response

Schema: org.atsc.query.serviceGuideUrls-response-20210127.json

Issue: It is not clear what the source of the content value is in the service guide response so it is not clear what the syntax or semantics should be.

Changes:
1) Add text to the content parameter semantics to indicate it should be sourced from globalContentID found in each content fragment as described in OMA BCAST service guide as referenced from A/332.

Compatibility: There is no change to any syntax or semantics only a clarification of the authoritative source of the content parameter. Implementations that used some other content ID value due to the previous vague specification will necessarily change but were not supplying helpful information anyway.

2.4.7 Notify Caption Display Prefs

Schema: org.atsc.notify-captionDisplayPrefs-20210127.json

Issue: The caption display preferences notification schema is not a valid JSON file as the properties of "imsc1" are placeholder text, not property name/value pairs. Also, there is a stray comma on line 54.

Changes:
1) Schema modified to include the format of "imsc1" from the version found in org.atsc.query.captionDisplay-response-20210127.json.
2) Date of schema file updated to "20210405".
3) A/344 working draft updated to reflect schema file name change.
4) Updated the schema file reference in all example JSON files.

Compatibility: The definition of IMSC1 was inadvertently not included in the published schema. The schema is currently illegal and could not be implemented thus there are no compatibility issues.

2.4.8 Query Dialog Enhancement Preference Request and Response

Schemas: org.atsc.query.dialogEnhancement[Pref]-request-20210127.json
org.atsc.query.dialogEnhancement[Pref]-response-20210127.json

Issue: Both the request and response schemas have the wrong method name which is missing the "Pref". Likewise, the same problem exists in the "method" property in the request schema. Also, the title and description have an unwanted double space.

Changes:
1) Schema files modified to include the correct name.
2) Date of schema file updated to "20210405".
3) A/344 working draft updated to reflect schema file name change.
4) Updated the schema file reference in all example JSON files.

Compatibility: The names of the schema files were incorrect and did not match any references. This would have precluded any automatic validation so could not have been operable prior to this change.

2.4.9 Launch App Request

Schema: org.atsc.launchApp-request-20210127.json

Issue: The schema contains the wrong method name.

Changes:
1) Schema modified to include the correct method name.
2) Date of schema file updated to "20210405".
3) A/344 working draft updated to reflect schema file name change.
4) Updated the schema file reference in all example JSON files.
Compatibility: This is a copy-paste error and inoperable in its present form.

2.4.10 Query Device Info Response

Schema: org.atsc.query-deviceInfo-response-20210127.json

Issue: The schema attempts to specify the deviceInput properties using incorrect syntax – the values were stated without including a "type" identifier.

Changes:
1) Schema corrected.
2) Date of schema file updated to "20210405".
3) A/344 working draft updated to reflect schema file name change.
4) Updated the schema file reference in all example JSON files.

Compatibility: The change is required to provide a valid schema file. The previous file had incorrect syntax and would not allow validation of any JSON files.

2.4.11 Cancel Request

Schema: cancel-request-20210127.json

Issue: The schema incorrectly omits the method parameter instead of the params in the required clause. This should be reversed.

Changes:
5) Schema corrected.
6) Date of schema file updated to "20210405".
7) A/344 working draft updated to reflect schema file name change.
8) Updated the schema file reference in all example JSON files.

Compatibility: The change is required to provide a valid schema file. The previous file had incorrect syntax and would not allow validation of any JSON files.

2.4.12 Example JSON Files Changed

Issue: Many example JSON files are not consistently named, have broken schema references or do not validate against the corresponding schema.

Changes:
1) Rename org.atsc.eventStream.event-request-example.json to match schema (org.atsc.eventStream.event-notification) and fix schema reference.
2) Fix schema reference in org.atsc.notify-ratingChange-example.json
3) Rename org.atsc.query.recoveredComponentInfo-response-example-66023.json to "-example1.json" and fix reference.
4) Rename org.atsc.query.recoveredComponentInfo-response-example-66026.json to "-example2.json" and fix reference.
5) Rename org.atsc.query.rmpMediaTime-request-example-66024.json to "-example1.json" and fix reference.
6) Rename org.atsc.query.rmpMediaTime-request-example-66028.json to "-example2.json" and fix reference.
7) Rename org.atsc.query.rmpMediaTime-response-example-66025.json to "-example1.json" and fix reference.
8) Rename org.atsc.query.rmpMediaTime-response-example-66030.json to "-example2.json" and fix reference.
9) Rename `org.atsc.query.rmpPlaybackRate-request-example-66027.json` to "example1.json" and fix reference.
10) Rename `org.atsc.query.rmpPlaybackRate-request-example-66032.json` to "example2.json" and fix reference.
11) Rename `org.atsc.query.rmpPlaybackState-request-example-66031.json` to "example1.json" and fix reference.
12) Rename `org.atsc.query.rmpPlaybackState-request-example-66036.json` to "example2.json" and fix reference.
13) Rename `org.atsc.query.rmpPlaybackState-response-example-66033.json` to "example1.json" and fix reference.
14) Rename `org.atsc.query.rmpPlaybackState-response-example-66038.json` to "example2.json" and fix reference.
15) Rename `org.atsc.query.rmpWallClockTime-request-example-66035.json` to "example1.json" and fix reference.
16) Rename `org.atsc.query.rmpWallClockTime-request-example-66040.json` to "example2.json" and fix reference.
17) Rename `org.atsc.query.rmpWallClockTime-response-example-66037.json` to "example1.json" and fix reference.
18) Rename `org.atsc.query.rmpWallClockTime-response-example-66041.json` to "example2.json" and fix reference.
19) Rename `org.atsc.query.serviceGuideUrls-request-example-66039.json` to "example1.json" and fix reference.
20) Rename `org.atsc.query.serviceGuideUrls-request-example-66042.json` to "example2.json" and fix reference.
21) Delete `org.atsc.request-captionDisplayPrefs-example.json` as `org.atsc.notify-captionDisplayPrefs-example.json` already exists and contains exactly the same JSON.
22) `org.atsc.drmOperation-response-example1.json` was modified to have the correct error message syntax. Previously, message was defined as an object and not a string as required by the schema.
23) In both the `org.atsc.query.signaling-request-example.json` and `org.atsc.query.signaling-response-example.json`, the integer "1" is quoted making it a string instead of the integer value allowed by the corresponding schemas. The quotes were removed. Similarly, this change was also made the A/344 working draft.
24) Removed `org.atsc.query.alerting-example.json` in favor of the existing `org.atsc.query.alerting-request-example.json`. Removed `org.atsc.notify-ratingChange-example.json` since the API has been deprecated and no examples are provided for other deprecated APIs.
25) Rating level examples are incorrect throughout. "TV PG" and "TV G" should have a hyphen. Fixed three examples in A/344 and the corresponding JSON files.

Compatibility: Example files are informative only so are not relevant to compatibility concerns. These changes make sure that the examples match their corresponding schemas and are correct.
3. DIALOG ENHANCEMENT API JSON SCHEMA FILE MISMATCH

3.1 Scope
The ATSC standard A/344:2021 specifies the following in Section 9.7.5 "Dialog Enhancement API" for the Dialog Enhancement Request:

If neither this value [i.e., `dialogEnhancementReset`] nor the `dialogEnhancementGain` is specified in the request, the amount of processing is not changed by this request. This can be used to determine the current amount and limits of processing applied by the audio decoder since these values are returned in the response.

This statement conflicts with the JSON schema referenced, `org.atsc.dialogEnhancement-request-20210127.json`, which mandates that exactly one of the two parameters shall be present in the request.

3.2 Rationale for Changes
As discussed in previous S38 specialist group calls, issues with JSON schema files are treated as “software bugs” and are expected to be fixed to preserve the original intent documented in the standard text of A/344.

Section 3.4 below proposes a change to the respective JSON schema file provided as associated resource.

3.3 Compatibility Considerations
The changes described in this section are not fully backward-compatible relative to the A/344:2021 version of the standard since the updated schema allows to validate JSON-RPC requests previously considered invalid. However, the change is compatible with the intended behavior described in the standard text and removes the conflict between the two.

3.4 List of Changes
Change instructions are given below in *italics*. Unless otherwise noted, inserted text, tables, and drawings are shown in **blue**; deletions of existing text are shown in **red strikeout**. The text “[ref]” indicates that a cross reference to a cited referenced document should be inserted. The text highlighted in **yellow** are automatic numbers or references that should be validated by the document editor.

3.4.1 Text of A/344
The Dialog Enhancement Request semantics shall be as defined in [Table 9.62] and the syntax defined in the schema file `org.atsc.dialogEnhancement-request-20210317.json`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Data Type</th>
<th>Short Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>jsonrpc</code></td>
<td>1</td>
<td>string</td>
<td>&quot;2.0&quot;</td>
</tr>
<tr>
<td><code>id</code></td>
<td>1</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>method</code></td>
<td>1</td>
<td>string</td>
<td>&quot;org.atsc.dialogEnhancement&quot;</td>
</tr>
<tr>
<td><code>dialogEnhancementGain</code></td>
<td>0 or</td>
<td>integer</td>
<td>If present, the requested dialog enhancement gain in dB</td>
</tr>
<tr>
<td></td>
<td>oneOfX</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 9.62 Dialog Enhancement Request Semantics
### 3.4.2 Associated Resources

Replace

```
org.atsc.dialogEnhancement-request-20210127.json
```

with

```
org.atsc.dialogEnhancement-request-20210317.json
```
as attached to this contribution.

### 3.4.3 (Informative) Additional Information

For the convenience of the reader, the proposed change is informatively shown in the following, where changes are highlighted in yellow:

Replace existing code:

```json
{
    "$schema": "http://json-schema.org/draft/2019-09/schema#",
    "title": "Dialog Enhancement API",
    "description": "Dialog Enhancement API",
    "type": "object",
    "properties": {
        "jsonrpc": {
            "type": "string",
            "enum": [ "2.0" ]
        },
        "id": {
            "type": "integer"
        },
        "method": {
            "type": "string",
            "enum": [ "org.atsc.dialogEnhancement" ]
        },
        "params": {
            "type": "object",
            "properties": {
                "dialogEnhancementGain": {
                    "type": "integer"
                },
                "dialogEnhancementReset": {
                    "type": "boolean"
                }
            }
        }
    },
    "oneOf": [
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Data Type</th>
<th>Short Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dialogEnhancementReset</td>
<td>0 or oneOf X</td>
<td>boolean</td>
<td>If present and &quot;true&quot;, resets the dialog enhancement value set by the Broadcast Application</td>
</tr>
</tbody>
</table>
With the following code:

```json
{
    "$schema": "http://json-schema.org/draft/2019-09/schema#",
    "title": "Dialog Enhancement API",
    "description": "Dialog Enhancement API",
    "type": "object",
    "properties": {
        "jsonrpc": {
            "type": "string",
            "enum": [ "2.0" ]
        },
        "id": {
            "type": "integer"
        },
        "method": {
            "type": "string",
            "enum": [ "org.atsc.dialogEnhancement" ]
        },
        "params": {
            "type": "object",
            "properties": {
                "dialogEnhancementGain": {
                    "type": "integer"
                },
                "dialogEnhancementReset": {
                    "type": "boolean"
                }
            }
        } },
        "not": {
            "required": [ "dialogEnhancementGain", "dialogEnhancementReset" ]}
}
```
4. DEVICE INPUT BAAPEAR PROPERTY DESCRIPTION

4.1 Scope
The Query Device Info API contains conflicting statements regarding whether the BAApear object is required. The last sentence indicates that the BAApear property is optional when all other information, including the schema, require it. The remaining portion of the description does not take the BAApear object into consideration aside from the last sentence.

4.2 Rationale for Changes
The conflict is causing implementor and tester confusion.

4.3 Compatibility Considerations
The sentence in the deviceInput semantic description uses language indicating the BAApear object is optional. The authoritative semantics, aka the schema, and the text describing BAApear all make it clear that the object is required. Any validation software used would have flagged the missing value if BAApear was not included so implementations should not be impacted by this change.

4.4 List of Changes
Change instructions are given below in *italics*. Unless otherwise noted, inserted text, tables, and drawings are shown in *blue*; deletions of existing text are shown in *red strikeout*. The text “[ref]” indicates that a cross reference to a cited referenced document should be inserted. The text highlighted in *yellow* are automatic numbers or references that should be validated by the document editor.

*Change the semantic definition of deviceInput as follows:*

```json
{   "required": [   "jsonrpc", "id", "method", "params" ]
}
```

**deviceInput** – This required object *indicates defines* the *available* user input key name and codes of the Receiver user interface as well as a required "launch key" for the Broadcaster Application. *It* The deviceInput object contains is a collection of input key/value pairs where the key is the user input name, and the value is the associated integer code. The Receiver shall provide all input keys and their associated values that the Broadcaster Application may request. The Broadcaster Application may use the Key APIs defined in Section [9.12] to request any or all of the keys defined in the deviceInput object.

In addition to the collection of input keys, the deviceInput object shall *may also contain* a BAApear object as described below.
5. **SECTION 7 REWRITE**

5.1 **Scope**
Section 7.2 describes Push and Pull modes for delivery of content to an Application Media Player (AMP). This is not supported.

5.2 **Rationale for Changes**
There is no mechanism defined in a Receiver to provide the broadcast "live" content in a way that can be "pulled" by the AMP. The discussion in Section 7.2 is misleading and not supported by any other text in the document. Further, the entire section is written around the push and pull concepts so needs to be rewritten.

5.3 **Compatibility Considerations**
The changes described in this section are not fully backward-compatible relative to the A/344:2021 version of the standard since the updated schema allows to validate JSON-RPC requests previously considered invalid. However, the change is compatible with the intended behavior described in the standard text and removes the conflict between the two.

5.4 **List of Changes**
Change instructions are given below in *italics*. Unless otherwise noted, inserted text, tables, and drawings are shown in *blue*; deletions of existing text are shown in *red strikeout*. The text “[ref]” indicates that a cross reference to a cited referenced document should be inserted. The text highlighted in *yellow* are automatic numbers or references that should be validated by the document editor.

7. **Utilizing AMP**

7.2.1 **Broadcast or Hybrid Broadband and Broadcast Live Streaming**
Although broadcast or hybrid live media streaming is typically played out by the RMP, it is possible for the AMP to request playback of the content. A flag in the HELD indicates whether the RMP can immediately play out the media content, or whether the service expects the AMP to play out the live media streaming. The Receiver can ignore this signaled expectation, in which case the RMP can immediately play out the live media streaming. There are *two* possible methods *on* how *by which* an AMP can play out a *live** media streaming, Pull and Push, and they *are* as described in the following sections Sections 7.2.4 and 7.2.5.

7.2.2 **Broadband Media Streaming**
There is no special consideration for playing media streams delivered over broadband-only other than what is provided in the DASH client specification [36]. Other media types may be supported by a Receiver as reported in the device capabilities provided by the Query Device Info API (see Section 9.13).

7.2.3 **Downloaded Media Content**
The AMP can play out *broadcast* or *broadcast downloaded media content* from either broadcast, using NRT, or broadband, using the Cache Request APIs (see Section 9.4). The Broadcaster Application discovers the MPD URL of the downloaded media content, and initiates play-out using one of the two mechanisms described here and described in Section 9.7.3. In either case, the
resultant media content is placed in the Application Context Cache. The Broadcaster Application may then use the HTML5 <video> tag in conjunction with MSE or EME to initiate playout.

7.2.4 AMP Utilizing Pull Model
The Pull model behaves the same as if the Receiver was a remote DASH Server. The DASH specification describes the details on how a DASH server should be implemented.

7.2.5 AMP Utilizing the Push Model
Pushed Media WebSocket Interface
In the Push model, this mechanism allows the AMP to play content delivered through the broadcast. The AMP opens the binary WebSocket connections specified in Section 8.2.1. Opening these WebSocket connections is an implicit request that the Receiver retrieve the Initialization and Media Segments of the media content and pass them to the AMP via these connections. The AMP then uses an HTML5 <video> tag in conjunction with MSE or EME to pass the media content to the Receiver's decoders for decoding and presentation. In the broadcast case, the media are retrieved from the broadcast via the ROUTE Client and pushed to the AMP via the WebSocket server. In the broadband case, the media are retrieved from a remote HTTP server via the HTTP Client and pushed to the AMP via the WebSocket server. In both cases, a DASH Streaming Server acts as an intermediary to retrieve the media segments from the ROUTE Client or HTTP Client and stream them to the Broadcaster App via the WebSocket connections.

6. REMOVE JSON RPC BATCH MODE REQUIREMENT

6.1 Scope
The second paragraph of Section 8.3 requires that JSON-RPC 2.0 Specification, duplicated in Annex D, in its entirety including Section 6, Batch Mode.

6.2 Rationale for Changes
The inclusion of batch mode would force a large development effort without significant benefit. In addition, the usage would need to be clarified to make it useful since it is not at all certain the behavior of some A/344 APIs if allowed to be "batched".

6.3 Compatibility Considerations
It is not clear if any current implementations have included batch mode. The language introduced does not require that it shall not be implemented only that it need not be implemented, thereby not impacting any past implementations.

6.4 List of Changes
Change instructions are given below in italics. Unless otherwise noted, inserted text, tables, and drawings are shown in blue; deletions of existing text are shown in red strikeout. The text “[ref]” indicates that a cross reference to a cited referenced document should be inserted. The text highlighted in yellow are automatic numbers or references that should be validated by the document editor.

In the second paragraph of section 8.3, add the following text:

The WebSocket interface for command and control shall be the JSON-RPC 2.0 Specification in Annex D, except that the WebSocket interface need not include the features described in Annex D,
Section 6 (batch mode operation). JSON-RPC provides RPC (remote procedure call) style messaging, including unidirectional notifications and well-defined error handling using the JavaScript Object Notation (JSON) data structure [20].

7. URI CLARIFICATION

7.1 Scope
This document clarifies the usage of two JSON RPC APIs that specify "uri" should always prepend the baseURI when referencing file content in the Application Context Cache. Also, add clarification to describe references to Periods. This amendment reflects work done as part of the CS Schemas reconciliation project N-016r0. https://members.atsc.org/apps/org/workgroup/tg3/download.php/68186/N-016r0-S38-293r1--NWIP-A344-CS-Schemas.pdf

7.2 Rationale for Changes
The current schemas for the Set RMP URL and XLink Resolution APIs constrain the rmpurl and mpdURL properties to type "string" with a "uri" format. The "uri" format requires the use of a scheme portion of the URI (e.g., "http:" or "urn:")) to successfully pass schema validation. This requires a Broadcaster Application to use a full URI by including the Base URI of the Application Context Cache whenever using these two APIs to access content within that cache. While it might be convenient for BAs to provide either a full URI or a relative path to Application Context Cache files, there are presently implementations that only work with full URIs and, in fact, fail silently (i.e., show a black screen without any error to the BA) when relative paths are submitted.

Therefore, BAs must necessarily support the full URI submission in these APIs to be certain that legacy receiver implementations will operate correctly. Since the BA will have code to support this case, there is little benefit in providing a relative path submission. However, adding clarifying text to indicate that a full URI is required and providing a suggestion for how to form that full URI for the Application Context Cache would be helpful to both receiver and BA developers.

7.3 Compatibility Considerations
The changes do not modify any normative requirements on the receiver and so are completely compatible with current implementations. The clarifying text is intended to make certain only full URIs are submitted as is currently required by the syntax.

7.4 List of Changes
Change instructions are given below in italics. Unless otherwise noted, inserted text, tables, and drawings are shown in blue; deletions of existing text are shown in red strikeout. The highlighted text "[ref]" indicates that a cross reference to a cited referenced document should be inserted. Other internal references are also highlighted to call attention to the need to update them as part of the editing process.

Change Section 9.7.3 as follows:

The Set RMP URL Request semantics shall be as defined in Table 9.58 and the syntax defined in the schema file org.atsc.setRMPURL-request-20210127.json. Additional semantic definitions of parameters follow the table.
Table 9.58  Set RMP URL Request Semantics

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Data Type</th>
<th>Short Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>jsonrpc</td>
<td>1</td>
<td>string</td>
<td>&quot;2.0&quot;</td>
</tr>
<tr>
<td>id</td>
<td>1</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td>method</td>
<td>1</td>
<td>string</td>
<td>&quot;org.atsc.setRMPURL&quot;</td>
</tr>
<tr>
<td>operation</td>
<td>0..1</td>
<td>enum</td>
<td>&quot;startRmp&quot;, &quot;stopRmp&quot;, &quot;pauseRmp&quot;, &quot;resumeRmp&quot;, &quot;fastForwardRmp&quot;, &quot;rewindRmp&quot;, &quot;skipForwardRmp&quot;, &quot;skipBackwardsRmp&quot;, &quot;resumeService&quot;</td>
</tr>
<tr>
<td>rmpUrl</td>
<td>0..1</td>
<td>string (uri)</td>
<td>Provides the URI of the MPD URL to be played by the RMP if &quot;operation&quot; = &quot;startRmp&quot;</td>
</tr>
<tr>
<td>rmpSyncTime</td>
<td>0..1</td>
<td>number</td>
<td>Indicates the time offset when the operation specified should occur</td>
</tr>
<tr>
<td>endOperation</td>
<td>0..1</td>
<td>enum</td>
<td>&quot;stopRmp&quot;, &quot;pauseRmp&quot;, &quot;resumeService&quot;</td>
</tr>
</tbody>
</table>

operation — This optional string shall define the operation to be performed by the RMP. When operation is absent, the Receiver shall cancel any pending Set RMP URL API request. The meaning of the enumerated values shall be defined as follows:

"startRmp" indicates that the RMP shall start playing the URL provided by the rmpurl property as described in the property description below.

"stopRmp" indicates that the RMP shall cease playback. For this operation, the rmpurl property is not required and shall be ignored if present.

"pauseRmp" indicates that the RMP shall suspend playback, freeze-frame, and mark the current position in the content.

"resumeRmp" indicates that the RMP shall continue playing from a previous pauseRmp operation.

"fastForwardRmp" indicates that the RMP shall speed up playback. The initial speed and sequential calls for this operation are Receiver-dependent.

"rewindRmp" indicates that the RMP shall playback in reverse. The initial speed and sequential calls for this operation are Receiver-dependent.

"skipForwardRmp" indicates that the RMP shall skip forward and resume playback. The time skipped is Receiver-dependent.

"skipBackwardsRmp" indicates that the RMP shall skip backward and resume playback. The time skipped is Receiver-dependent.

"resumeService" indicates that the RMP shall resume normal playback of the current Service.

For this operation, the "rmpurl" property is not required and shall be ignored if present.

rmpurl — When the operation value is set to startRmp, this string shall be specified and provide a fully qualified URL referencing an MPD to be played by the RMP, whether referencing an MPD over broadband or in the Application Context Cache. The URL shall be accessible to the Receiver. The appropriate error code (see below) shall be returned if the URL cannot be accessed. Note that for an MPD in the Application Context Cache, the full URL can be constructed using the Base URI provided using the Query Receiver Web Server URI API as described in Section 9.2.7.

Change Section 9.16.2 as follows:
The XLink Resolved Request semantics shall be as defined in Table 9.105 and the syntax defined in the schema file org.atsc.xlinkResolution-request-20210127.json. Additional semantic definitions of parameters follow the table.

### Table 9.105 XLink Resolved Request Semantics

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Use</th>
<th>Data Type</th>
<th>Short Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>jsonrpc</td>
<td>1</td>
<td>string</td>
<td>&quot;2.0&quot;</td>
</tr>
<tr>
<td>id</td>
<td>1</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td>method</td>
<td>1</td>
<td>string</td>
<td>&quot;org.atsc.xlinkResolution&quot;</td>
</tr>
<tr>
<td>xlink</td>
<td>1</td>
<td>string</td>
<td>The XLink value from the xlink:href attribute in the MPD Period element</td>
</tr>
<tr>
<td>mpdURL</td>
<td>oneOf X</td>
<td>string (uri)</td>
<td>The URI of the MPD URL Period XML fragment intended to replace the Period containing the xlink:href attribute</td>
</tr>
<tr>
<td>period</td>
<td>oneOf X</td>
<td>string</td>
<td>The Period text to replace the Period containing the xlink:href attribute</td>
</tr>
</tbody>
</table>

**xlink** – This required string shall be the XLink value from the xlink:href attribute in the MPD **Period** element. This string corresponds to the value received in the XLink Resolution Notification parameters (see Section 9.16.1).

**mpdURL** – If provided, this string shall provide the **MPD** URL of a Period XML fragment to replace the **Period** containing the xlink:href attribute. One of either the mpdURL or period properties shall be provided. If this property is provided, it shall be a fully qualified URI, whether referencing the Period XML fragment over broadband or in the Application Context Cache. Note that for a Period XML fragment in the Application Context Cache, the full URI can be constructed using the Base URI provided using the Query Receiver Web Server URI API as described in Section 9.2.7.

... As an example, the Broadcaster Application submits the following request, presuming that the **Base URI** is "http://192.168.32.117:8182/s02gPkwZx14iO":

```json
{}
"jsonrpc": "2.0",
"method": "org.atsc.xlinkResolution",
"params": {
  "xlink": "urn:xbc4399FB77-3939EA47",
  "mpdURL": "http://192.168.32.117:8182/s02gPkwZx14iO/local/ads/ReplacementAd.xml",
  "id": 104
}
```

*Change the org.atsc.xlinkResolution-request-example.json example file to match the example shown above.*

Further, the A/344 CS WD may not be consistent when referring to a "Period" XML fragment but may instead use "MPD" incorrectly. Replace all inconsistent uses of "MPD" with "Period XML
fragment". Note that "MPD" should continue to be used when referring to an actual MPD, e.g., in the Set RMP URL API.

8. S38-332R0: DATE-TIME FORMAT UPDATES

8.1 Scope
This document changes all API JSON schema time and date field formats from W3C XML-standard "xs:dateTime" to JSON-standard "date-time" in A/344:2021 pursuant to New Project Proposal (NPP) N-028r1-S38-325r2-NPP-A344-Time-Date-Schema-Format-Change.pdf.

8.2 Rationale for Changes
The current A/344:2021 JSON schemas use "xs:dateTime" to be consistent with other standard signaling schemas (e.g., A/331). This causes a problem because "xs:dateTime" is in an XML schema name space and does not readily apply to JSON schema syntax. The JSON "date-time" format is consistent with "xs:dateTime" but is known to validation tools whereas "xs:dateTime" is not.

The changes described below allow JSON validation tools to correctly validate date and time values in JSON messages. Currently, the option to validate formats must be disabled to avoid errors or, more likely, the validation tool simply skips over the unknown format giving the illusion that the message is correct when, in fact, it has not been validated completely. Specifying a format that is ignored or causes a parsing error is, at best, not helpful and, at worst, causes validation errors with the schema or well-formed JSON messages validated against the schema.

Further, JSON "date-time" format is the default serialization format for JavaScript and is likely the mechanism used by Broadcaster Applications to populate string fields with this format.

8.3 Compatibility Considerations
Presently, validation tools must have the format validation disabled or some may simply skip the validation of the unknown format syntax. This change will allow those tools to validate the date and time values provided in JSON API messages. This change may cause some messages to fail to validate, however, this is because they were originally not correct and were not being completely validated.

It is assumed that receiver implementations will not be impacted by this change since Broadcaster Application-created JSON messages encode date and time in RFC 3339 format, namely, JSON "date-time" format. RFC 3339 is a constrained subset of ISO 8601 which in turn is the basis for the XML "xs:dateTime" format so date and time values generated using JavaScript will be compatible with "xs:dateTime".

Broadcaster Applications that copy "xs:dateTime"-formatted strings directly from XML data structures into JSON API will need to change when these changes are implemented. "xs:dateTime" is a superset of ISO 8601 so potentially strings may be provided that do not translate directly into "date-time". Broadcaster Applications must convert the "xs:dateTime"-formatted string to native binary and then serialize to "date-time" format. Text describing this potential problem is provided for the appropriate API changes below. Note this is only an issue for request APIs since receivers implementing these changes will not respond with "xs:dateTime"-formatted strings.
8.4 List of Schema Change Instructions

Change schema file org.atsc.query.alerting-response-20210127.json as follows:
1. Change the file name date to 20210827 both in the actual file name and internally
2. Change the format of receiveTime from "xs:dateTime" to "date-time"
3. Update all example JSON files as appropriate

Change schema file org.atsc.notify-alertingChange-20210127.json as follows:
1. Change the file name date to 20210827 both in the actual file name and internally
2. Change the format of receiveTime from "xs:dateTime" to "date-time"
3. Update all example JSON files as appropriate

Change schema file org.atsc.setFilterCodes-request-20210127.json as follows:
1. Change the file name date to 20210827 both in the actual file name and internally
2. Change the format of expires from "xs:dateTime" to "date-time"
3. Update all example JSON files as appropriate

Change schema file org.atsc.query.rmpMediaTime-response-20210127.json as follows:
1. Change the file name date to 20210827 both in the actual file name and internally
2. Change the format of startDate from "xs:dateTime" to "date-time"
3. Update all example JSON files as appropriate

Change schema file org.atsc.xlinkResolution-response-20210127.json as follows:
1. Change the file name date to 20210827 both in the actual file name and internally
2. Change the format of periodStart from "xs:dateTime" to "date-time"
3. Update all example JSON files as appropriate

9. S38-335R0: CONTENT RECOVERY STATE API SCHEMA FIX

9.1 Scope
This document provides a fix to the Query Content Recovery State API response schema and examples as it does not match the semantic description in the A/344 document. This work is covered under New Project Proposal (NPP) N-016r0-S38-293r1--NWIP-A344-CS-Schemas.pdf.

9.2 Rationale for Changes
The response schema omits the required result properties and does not match the published document.

9.3 Compatibility Considerations
These changes fix errors in the schema response definition. Any implementations are based on erroneous schemas and are not correct or have not been validated.

9.4 List of Schema Change Instructions
Change schema file org.atsc.query.contentRecoveryState-response-20210127.json as follows:
1. Change the file name date to 20210902 both in the actual file name and internally
2. Add the four properties described in A/344 Section 9.10.1
3. Update the example JSON file to match the schema date (the file already provided the missing properties)

— End of Document —