



**ATSC**

ADVANCED TELEVISION  
SYSTEMS COMMITTEE

**ATSC Standard:  
A/331:2023-08 Amendment No. 1,  
“FDT Enhancement”**

---

Doc. A/331:2023-08 Amend. No. 1  
4 October 2023

**Advanced Television Systems Committee**  
1300 I Street, N.W., Suite 400E  
Washington, D.C. 20005  
202-872-9160

The Advanced Television Systems Committee, Inc. is an international, non-profit organization developing voluntary standards and recommended practices for broadcast television and multimedia data distribution. ATSC member organizations represent the broadcast, professional equipment, motion picture, consumer electronics, computer, cable, satellite, and semiconductor industries. ATSC also develops 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 Consumer Technology Association (CTA), the Institute of Electrical and Electronics Engineers (IEEE), the National Association of Broadcasters (NAB), the Internet & Television Association (NCTA), and the Society of Motion Picture and Television Engineers (SMPTE). For more information visit [www.atsc.org](http://www.atsc.org).

---

*Note:* The user's attention is called to the possibility that compliance with this standard may require use of an invention covered by patent rights. By publication of this 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

Version	Date
Amendment approved	4 October 2023

# ATSC Standard: A/331:2023-08 Amendment No. 1, “FDT Enhancement”

## 1. OVERVIEW

### 1.1 Definition

An Amendment is generated to document an enhancement, an addition or a deletion of functionality to previously agreed technical provisions in an existing ATSC document. Amendments shall be published as attachments to the original ATSC document. Distribution by ATSC of existing documents shall include any approved Amendments.

### 1.2 Scope

This document specifies updated signaling in order to enable Repair Flow operation on memory constrained platforms and repair large files in smaller pieces. This amendment also adds Source Block interleaving to increase time diversity of the NRT files. This amendment is in response to New Project Proposal N-063r0, “A331-AMD-ALFEC-Enhancements”.

### 1.3 Rationale for Changes

The changes described in this document are being proposed in order to enable Repair Flow in memory constrained platforms and increase time diversity of Source Blocks to increase length of packet dropout toleration.

### 1.4 Compatibility Considerations

The changes described in this document are backward-compatible relative to the currently published version of the standard to which this Amendment pertains and any previously approved Amendments for that standard; however currently deployed receivers would not have the capability of repairing source block symbols on-the-go with smaller memory sizes for file partitions. Such receivers would need updates if they are to repair file partitions and tolerate large packet loss.

## 2. 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. **Yellow highlights** indicate intended cross-references and other information that is expected to be updated prior to publication within a new revision.

### 2.1 Normative References

*No changes.*

### 2.2 Informative References

*No changes.*

2.3 Acronyms and Abbreviations

No changes.

2.4 Terms

No changes.

2.5 Change Instructions

Update text to A/331 Section A.3.3.2.2 with the following redline changes.

A.3.3.2.2. Semantics Overview

The FDT extensions shall be represented as an XML data structure as elements and attributes that conform to the definitions in the XML schema that has namespace:

tag:atsc.org,2016:XMLSchemas/ATSC3/Delivery/ATSC-FDT/1.0/

The definition of this schema is in an XML schema file, *ATSC-FDT-1.0-2021042830714.xsd*, accompanying this Standard, as described in Section 3.6 above. The XML schema xmlns short name should be "afdt".

The indicated XML schema specifies the normative syntax of the FDT extensions.

Update text to A/331 Section 7.1.4 with the following redline changes.

...

The S-TSID shall be represented as XML documents and other dependent data type definitions containing an **S-TSID** root element that conforms to the definitions in the XML schema files that have the namespace:

tag:atsc.org,2016:XMLSchemas/ATSC3/Delivery/S-TSID/1.0/

The definition of this schema is in the file, *S-TSID-1.0-2021041130714.xsd*, accompanying this Standard, as described in Section 3.6 above. The XML schema xmlns short name should be "stsid".

Add text to A/331 Section A.3.3.2.3 with the following redline changes.

A.3.3.2.3 ATSC Extensions to the FDT-Instance Element

Informative Table A.3.3 lists the ATSC-defined extensions to the **FDT-Instance** element. The semantics of the attributes described in Table A.3.3 shall be as given below the table.

**Table A.3.1** ATSC-Defined Extensions to **FDT-Instance** Element

Element or Attribute Name	Use	Data Type	Description
@efdtVersion	0..1	unsignedByte	The version of this Extended FDT Instance descriptor.
@maxExpiresDelta	0..1	unsignedInt	Time interval for use in deriving the expiration time of the associated Extended FDT Instance.
@maxTransportSize	0..1	unsignedInt	The maximum transport size of any object described by this Extended FDT Instance.

Element or Attribute Name	Use	Data Type	Description
@appContextIdList	0..1	afdt:uriList	A space-separated list of URIs representing one or more unique Application Context Identifiers.
@fileTemplate	0..1	string	Describes the means to generate the file URL, i.e. Content-Location attribute of the <b>File</b> element.
@filterCodes	0..1	afdt:listOfUnsignedInt	Filter Codes applying to all files in the Source flow.
@maxCacheMemory	0..1	unsignedint	The maximum memory required at any time by the receiver to cache the data when the file(s) are being delivered out-of-order.
@order	0..1	Boolean	Flag indicating out-of-order delivery of files if value is "false" or in-order delivery of files if value is "true".

@efdtVersion – An 8-bit unsigned integer value that shall represent the version of this Extended FDT Instance. The version shall be increased by 1 modulo 256 each time the Extended FDT Instance is updated. @version is an ATSC-defined extension of the FLUTE FDT as specified in RFC 6726 [31].

@maxExpiresDelta – A 32-bit unsigned integer value, which when present, shall represent a time interval in number of seconds, which when added to the wall clock time at the receiver when the receiver acquires the first ROUTE packet carrying data of the object described by this Extended FDT Instance, shall represent the expiration time of the associated Extended FDT Instance. When @maxExpiresDelta is not present, the expiration time of the Extended FDT Instance shall be given by the sum of a) the value of the ERT field in the EXT\_TIME LCT header extension in the first ROUTE packet carrying data of that file, and b) the current receiver time when parsing the packet header of that ROUTE packet. See Section A.3.3.2.7 on additional rules for deriving the Extended FDT Instance expiration time. @maxExpiresDelta is an ATSC-defined extension of the FLUTE FDT as specified in RFC 6726 [31].

@maxTransportSize – A 32-bit unsigned integer value that shall represent the maximum transport size in bytes of any delivery object described by this Extended FDT Instance. This attribute shall be present if a) the **FDT-Instance@fileTemplate** is present; or b) one or more **File** elements, if present in this Extended FDT Instance, do not include the @Transfer-Length attribute. When @maxTransportSize is not present, the maximum transport size is not signaled.

@appContextIdList – A list of unique URI values, separated by spaces, which when present, define one or more Application Context Identifiers associated with the containing **FDT-Instance** element. Files may be associated with multiple application contexts. The Application Context Identifier provides a mechanism of grouping files for use with Broadcaster Applications as defined in A/344, Interactive Content [47].

Any Application Context Identifiers included as an attribute of the Extended FDT element, **FDT-Instance@appContextIdList**, are associated with all files defined as children of the **FDT-Instance** data structure. Any Application Context Identifiers defined in the list attribute of the File element, **File@appContextIdList**, are associated with that file only. Thus, the total collection of Application Context Identifiers associated with a specific file consists of the union of the two lists: **File@appContextIdList** and **FDT-Instance@appContextIdList**. For the complete definition and usage, please see Section A.3.3.2.3 above.

Each Application Context Identifier shall be a globally-unique URI that allows files to be grouped together. Receivers use the Application Context Identifier to provide access to the associated files such that the group of files from one Application Context Identifier is completely separate from the group of files assigned to another Application Context Identifier.

A file may be associated with multiple Application Context Identifiers, each resulting in a different URL for accessing the file. In this way, files may be shared among applications.

For additional discussion and details of the Application Context Identifier concept, refer to A/337, Application Signaling [7] and A/344, Interactive Content [47].

**@fileTemplate** – A string value, which when present and in conjunction with parameter substitution, shall be used in deriving the file URL, i.e., the **Content-Location** attribute, for the delivery object described by this Extended FDT Instance. The mechanism, as further described in Section A.3.3.2.7, shall involve substituting the TOI value of the delivery object, as contained in the ROUTE packet header, for the pattern '\$TOI\$' in the string representation of **@fileTemplate**. The derived **Content-Location** shall be a relative URI conforming to the provisions of Section 8.1.1.2 for broadcast-delivered resources. By this means, a one-to-one mapping is created between the TOI and the file URL. It also implies that in the event that each delivery object of the source flow is a DASH Segment, the Segment number will be equal to the TOI value of the object. The **@fileTemplate** attribute shall be present in the **FDT-Instance** element when the LCT Channel carries a sequence of DASH Media Segment files.

**@filterCodes** – A space-separated list of 32-bit unsigned integers that shall represent Filter Codes that are associated with all files identified in this Extended FDT Instance except for those files in which a **File@filterCodes** attribute is present, in which case the Filter Codes in the **File@filterCodes** attribute shall take precedence.

**@maxCacheMemory** – When **FDT-Instance.order** flag is false, this optional 32-bit unsigned integer attribute represents the maximum memory required at any time to keep the received file data in cache. This allows the receiver to assess its capability to receive the data in non-persistent storage prior to block transfer of a completed block of contiguous data to persistent storage or for the receiver to repair a block of data in the case when Repair Flow is utilized. When not present the **FDT-Instance.maxTransportSize**, the LCT Header Transfer Length or the Entity Mode Content-Length should be used by the receiver. If **FDT-Instance.order** is true, this attribute has no meaning. When Repair Flows are in use, the value of **@maxCacheMemory** shall be the summation of the source flow symbol payload bytes across the maximum concurrently interleaved source blocks. See RFC 6726 [31] Section 3.4.2 for guidance when **FDT-Instance** element contains common attribute(s) with File element.

**@order** – This Boolean attribute shall indicate whether the files with File element(s) within **FDT-Instance** element are delivered out of the order of their generation. When the value is “false”, arbitrary delivery order is indicated. When the value is “true”, in-order delivery is indicated. When this attribute is absent, the default value shall be “true”. See RFC 6726 [31] Section 3.4.2 for guidance when **FDT-Instance** element contains common attribute(s) with File element.

*Add text to A/331 Section A.4.3.2 with the following redline changes.*

The **RepairFlow** element is a child element of **S-TSID.RS.LS** (see Section 7.1.4).

While the XML schema specifies the normative syntax of the **RepairFlow** element, informative Table A.4.1 describes the structure of the **RepairFlow** element in a more illustrative way.

The semantics of the elements and attributes of **RepairFlow** shall be as given in Table A.4.1.

**Table A.4.2** Semantics of **RepairFlow** Element

Element or Attribute Name	Use	Data type	Description
<b>RepairFlow</b>		stsid:rprFlowType	Repair flow carried in the LCT channel.
<b>FECParameters</b>	0..1		FEC Parameters corresponding to the repair flow.
@maximumDelay	0..1	unsignedInt	Maximum delivery delay between any source packet in the source flow and the repair flow.
@overhead	0..1	unsignedShort (percentage)	FEC overhead in a ROUTE packet represented as a percentage.
@minBuffSize	0..1	unsignedInt	Required buffer size to handle all associated objects that are assigned to a super-object.
@fecOTI	1	hexBinary	FEC-related information associated with an object as well as FEC information associated with the encoding symbols of the object and is to be included within this declaration and applies to all repair packets with the repair flow.
@percentRepair	0..1	unsignedShort	The maximum ratio of repair symbols to source symbols as a percentage with values between zero and 200 inclusive.
@checksumList	0..1	listOfString	List of CRC32 hexadecimal formatted checksums for each source block in the order of the source blocks.
<b>ProtectedObject</b>	0..N		List of the source flow(s) protected by this repair flow and the details on how the protection is done. It also defines how certain delivery objects of a collection of objects are included in the repair flow.
@sessionDescription	0..1	string	Specifies the session description information for the source flow.
@tsi	1	unsignedInt	The Transport Session Identifier (TSI) for the source flow to be protected.
<b>SourceTOI</b>	1..N		The Transport Object Identifier (TOI) value(s) of the delivery object(s) corresponding to the TOI of a given FEC (super-)object delivered by the repair flow.
@x	0..1	unsignedShort	Value of the constant X for use in deriving the TOI of the delivery object from the TOI of the FEC (super-)object.
@y	0..1	unsignedShort	Value of the constant Y for use in deriving the TOI of the delivery object from the TOI of the FEC (super-)object.

**RepairFlow** – A complex element whose subordinate elements and attributes shall contain information about each of the one or more repair flows carried in the LCT channel, which may be associated with Source Flows, and referenced by other Service signaling metadata.

**FECParameters** – A complex element whose subordinate elements and attributes contain information pertaining to AL-FEC parameters that apply to this repair flow.

@maximumDelay – A 32-bit unsigned integer whose value, when present, shall represent the maximum delivery delay, in milliseconds, between any source packet in the source flow and the repair packet, associated with that source packet, in the repair flow. Default semantics of this attribute, when absent, is not defined.

- @overhead** – A 16-bit unsigned integer whose value shall represent the sum of the AL-FEC related fields in the ROUTE repair packet relative to the size of the repair packet size as a percentage up to 1000%. The AL-FEC related parameters shall comprise the attributes **@maximumDelay**, **@overhead**, and **@minBufferSize**, and the **@fecOTI** attribute under the element **RepairFlow.FECParameters**. Absence of this attribute means that the AL-FEC related overhead information is not provided.
- @minBuffSize** – A 32-bit unsigned integer whose value shall represent a required size of the receiver transport buffer for AL-FEC decoding processing. When present, this attribute shall indicate the minimum buffer size that is required to handle all associated objects that are assigned to a super-object i.e. a delivery object formed by the concatenation of multiple FEC transport objects in order to bundle these FEC transport objects for AL-FEC protection. Absence of this attribute means that the required minimum buffer size is unknown.
- @fecOTI** – A parameter formatted as 12 octets in **xs:hexBinary** format consisting of the concatenation of Common and Scheme-Specific FEC Object Transmission Information (FEC OTI) as defined in Sections 3.3.2 and 3.3.3 of RFC 6330 [29], and which corresponds to the delivery objects carried in the source flow to which this repair flow is associated, with the following qualification. The 40-bit Transfer Length (F) field may either represent the actual size of the object, or it is encoded as all zeroes. In the latter case, it means that the FEC transport object size is either unknown, or cannot be represented by this attribute. In other words, for the all-zeroes format, the delivery objects in the Repair flow correspond to streaming content – either a live Service whereby content encoding has not yet occurred at the time this S-TSID instance was generated, or pre-recorded streaming content whose delivery object sizes, albeit known at the time of S-TSID generation, are variable and cannot be represented as a single value by the **@fecOTI** attribute.
- @percentRepair** – An optional integer that equates to the maximum ratio of repair symbols to source symbols as a percentage with range of 0 to 200 inclusive. Absence of this attribute means the AL-FEC repair data size is not provided.
- @checksumList** – An optional space separated list of 32bit unsigned integers in hexadecimal format (e.g., values are ‘hexadecimal’ not ‘0xhexadecimal’) that equates to the calculated CRC32 checksums of each source block. This list may be used by the receiver to determine the accuracy of received and/or repaired data in a source block. In its absence the integrity of the complete received and/or repaired file may be available through the optional FDT-Instance MD5 Checksum, however this CRC32 checksum list, when present, allows the receiver to more finely assess the need to replace data at a source block level of resolution.
- ProtectedObject** – A complex element whose attributes contain information about the source flow(s) protected by this repair flow, and the details on how the protection is performed. It also defines how certain delivery objects of a collection of objects are included in the repair flow.
- @sessionDescription** – A string whose value, in the form of comma-separated fully-qualified attributes, shall represent session description information for the source flow, if applicable, protected by the repair flow. The set of fully-qualified attributes contained in this attribute shall be identical to the corresponding session description information as specified in the S-TSID fragment. These might include, for example: **S-TSID.RS@sIpAddr**, **S-TSID.RS@dIpAddr**, **S-TSID.RS@dPort**, **S-TSID.RS.LS@tsi**, **S-TSID.RS.LS@bw**, **S-TSID.RS.LS@startTime**, **S-TSID.RS.LS@endTime** (list is not intended to be comprehensive). When **@sessionDescription** is not present, it shall imply that the source flow is carried in the same ROUTE session/LCT channel as the repair flow.



**@tsi** – A 32-bit unsigned integer whose value shall represent the Transport Session Identifier (TSI) of the source flow protected by this repair flow.

**SourceTOI** – This element shall convey the TOI value(s) of the one or more source objects protected by a given FEC transport object or FEC super-object. Its attributes **@x** and **@y**, when present, shall represent the constants X and Y, respectively, for use in deriving each constituent **SourceTOI** value from the TOI of a corresponding FEC (super-)object according to the equation as shown in Section A.4.3.3, i.e., [ $\text{SourceTOI} = X * \text{TOI} + Y$ ].

**@x** – The value of the constant X, when present, for use in deriving the parent **SourceTOI** value from the above equation. The default value shall be "1".

**@y** – The value(s) of the constant Y, when present, for use in deriving the parent **SourceTOI** value from the above equation. The default value shall be "0".

Note that when neither **@x** nor **@y** is present there is a 1:1 relationship between each delivery object carried in the source flow as identified by **ProtectedObject@tsi** to an FEC object carried in this repair flow. In this case the TOI of each of those delivery objects shall be identical to the TOI of the corresponding FEC object.

Note that the TSI of the LCT channel carrying the repair flow is not included as an attribute of **RepairFlow** element because that parameter is already provided by the **S-TSID.RS.LS@tsi** associated with the repair flow.

– End of Document –