EBU

OPERATING EUROVISION AND EURORADIO

TECH 3352

THE CARRIAGE OF IDENTIFIERS IN THE BROADCAST WAVE FORMAT (BWF)

SOURCE: MIM, FAR-BWF

Geneva
August 2012

* Page intentionally left blank. This document is paginated for two sided printing

Summary

The Broadcast Wave Format (BWF) is a file format for audio data and metadata which is specified by EBU Tech 3285. This format adds a number of "chunks" to the Microsoft WAVE audio file format to allow users and systems to store information relevant to an audio file, and thus enable the file's discovery and use in a workstation, Media Asset Management system or archive.

In 2003, the EBU ratified the <axml> chunk to carry XML data so as to prevent the need to define further chunks. The <axml> chunk was designed to carry any (and any number) of XML fragments and documents. However, because information may be conveyed from one organisation to another using a variety of expressions of XML, the EBU has published this Recommendation to encourage a uniform method to convey identifiers by means of a consistent representation of XML. The expected benefit of using this Recommendation is that there will be a minimum amount of effort involved in both the creation and the correct interpretation of the XML instance documents which are exchanged between organisations which use BWF files.

Conformance Notation

This document contains both normative text and informative text.

All text is normative except for that in the Introduction, any section explicitly labelled as 'Informative' or individual paragraphs which start with 'Note:'.

Normative text describes indispensable or mandatory elements. It contains the conformance keywords 'shall', 'should' or 'may', defined as follows:

'Shall' and 'shall not':	Indicate requirements to be followed strictly and from which no deviation is permitted in order to conform to the document.
'Should' and 'should not':	Indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others.
	OR indicate that a certain course of action is preferred but not necessarily required.
	OR indicate that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.
'May' and 'need not':	Indicate a course of action permissible within the limits of the document.

Default identifies mandatory (in phrases containing "shall") or recommended (in phrases containing "should") presets that can, optionally, be overwritten by user action or supplemented with other options in advanced applications. Mandatory defaults must be supported. The support of recommended defaults is preferred, but not necessarily required.

Informative text is potentially helpful to the user, but it is not indispensable and it does not affect the normative text. Informative text does not contain any conformance keywords.

A conformant implementation is one which includes all mandatory provisions ('shall') and, if implemented, all recommended provisions ('should') as described. A conformant implementation need not implement optional provisions ('may') and need not implement them as described.

Page intentionally left blank. This document is paginated for two sided printing

Contents

Page intentionally left blank. This document is paginated for two sided printing

The Carriage of Identifiers in the Broadcast Wave Format (BWF)

EBU Committee	*First Issued*	*Revised*	*Re-issued*
TC	2012		

Keywords: Metadata, XML, Broadcast Wave Format, Reference Data, Identifiers, axml.

1. Introduction

The Broadcast Wave Format (BWF) is a file format for audio data. Its specification was published by the EBU as Tech 3285 [1] in 1997 and it has since been updated to support loudness metadata.

The BWF was created to contain audio data and the minimum metadata considered necessary for all broadcast production applications. It was based on the Microsoft RIFF WAV audio file format, to which the EBU added a "Broadcast Audio Extension" chunk.

It can be used for the seamless exchange of audio material between different broadcast environments and between equipment based on different computer platforms. The WAV file format has become the standard for distribution of audio by the major record companies. Although producers may supply AIF files, they will be converted to WAV / BWF during the mastering process.

As use of the BWF has grown, so has the information that users wish to include with the payload. Initially, further chunks were created to meet particular requirements, but in 2003, the EBU created a chunk to carry Extensible Markup Language ("XML") data so as to prevent the need to define further chunks for each new requirement that arose and to encourage the integration of IT and Broadcast applications.

The XML chunk for the BWF file (<axml>) [3] was designed to carry any number of XML fragments and documents. Recognising that this information may be conveyed from one organisation to another using a variety of XML expressions, the EBU has developed this recommended practice to encourage a uniform representation of identifiers in XML. In this way there will be a minimum amount of effort involved in the creation and in the correct interpretation of XML documents and fragments that are carried in the BWF, for the sake of interoperability.

2. Rationale for this Document

The EBU encourages the use of a uniform method of expression of identifiers carried as XML in WAV / BWF files.

Unique identifiers such as the ISRC (International Standard Recording Code) provides the most appropriate method of ensuring that the file at hand is the expected resource (e.g. a clean version of the label copy supplied by the record company), which may share the same artist, title and timing with other undesired versions (e.g. an instrumental version, or a version with adult lyrics). With the right ISRC identifier embedded in a BWF file, a user can check that he has the correct file.

Globally unique identifiers can take two main forms:

- Organisations can develop globally unique identifiers within their own namespce or domain;
- Organisations can register and obtain identifiers from third-party organisations, which role is to ensure that one globally unique identifier is attributed only once to a work or a version of a work.

This specification particularly defines how to use ISRC identifiers, but others can alternatively be used.

It should be noted that additional EBUCore compliant metadata can be provided in complement to the identifier element in the same <axml> chunk, which gives the potential for considerably extending the domain of application of the present recommended practice.

3. Recommended Practice

The EBU recommends that implementers of systems that employ the BWF:

- use XML to exchange relevant information which is not specified elsewhere in the BWF;
- use the <axml> chunk of the BWF to carry that XML metadata;
- use EBUCore (Tech 3293) [4] as the preferred metadata set unless you are in a specific context where the use of an internal in-house metadata format is sufficient;
- if EBUCore is used the recommended form of expression is as set out in Section 4 of this document;
- if EBUCore is used then the EBU's Classification Scheme for Identifiers should be used – see Section 5;
- put processes in place to ensure that only data relevant for public use is retained in BWF files when those files are released into the public domain – see Section 6.

4. Recommended Form of Expression

To assist implementers in achieving efficient, consistent and reliable transmission of identifiers, the EBU's recommended form of expressing identifiers as XML is set out below.

The reference schema can be found in the *EBUCore* (Tech 3293) specification. *EBUCore* defines the XML structure to define an identifier element of identifierType, as shown below.

Table 1: Definitions applying to EBUCore's identifierType

Name	Definition
identifierType	A complexType defining the structure and minimum information required to provide accurate and reliable identification information.
typeLabel	An optional element to define as a chain of characters the type associated with the identifier e.g. a GUID (Globally Unique Identifier).
typeDefinition	An optional element to provide as a chain of characters a definition of the type of identifier defined in the typeLabel or typeLink.
typeLink	An optional element to provide a URI (Unique resource Identifier) pointing to a term in a classification scheme. If available, the classification term name and definition can be used to populate the typeLabel and typeDefinition attributes.
formatLabel	An optional element to define as a chain of characters the format used to express the identifier e.g. a ISRC (International Standard recording Code),

	ISAN (International Standard Audiovisual Number) or V-ISAN (Versionned-ISAN).
formatDefinition	An optional element to provide as a chain of characters a definition of the format of identifier defined in the formatLabel or formatLink.
formatLink	An optional element to provide a URI (Unique resource Identifier) pointing to a term in a classification scheme. If available, the classification term name and definition can be used to populate the formatLabel and formatDefinition attributes.
note	An optional attribute to provide additional contextual information related to the use of the identifier (as a chain of characters).
dc:identifer	A mandatory element to provide the value of the identifier.
Attributor	An optional element to provide information on the person or organisation who has attributed /attached the identifier to the BWF file. The Attributor is of entityType as defined in EBUCore (Tech 3293) .

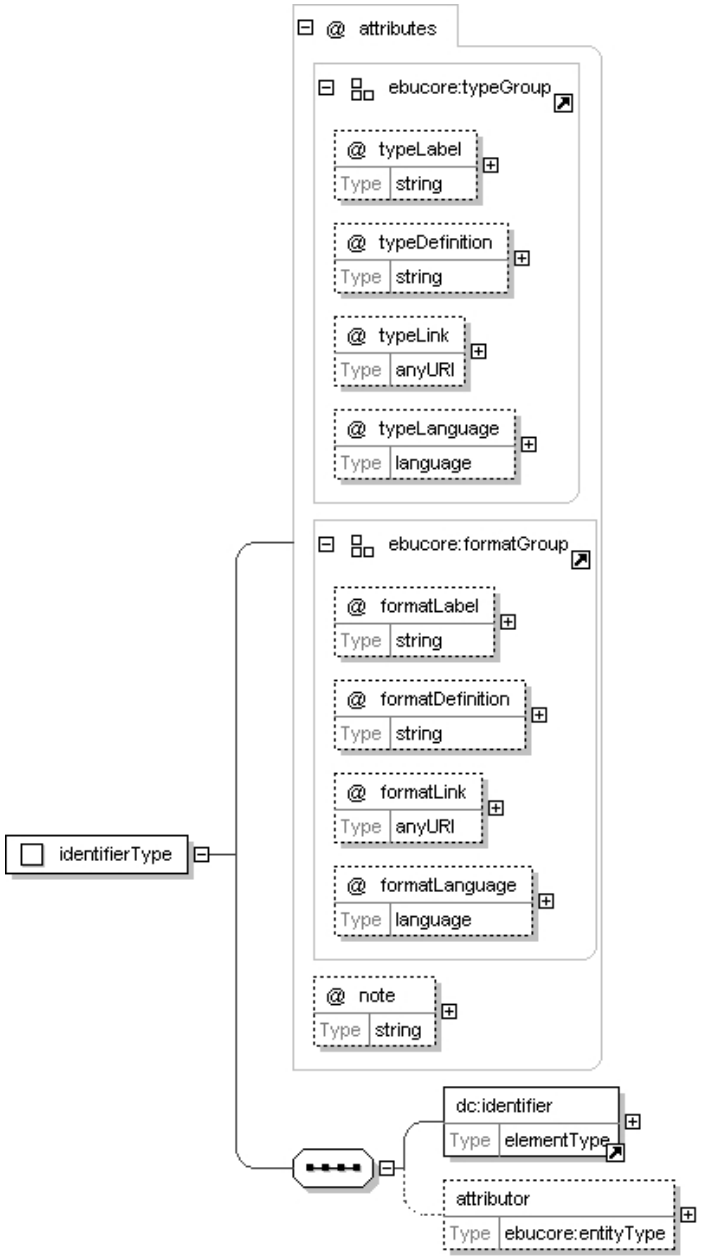

Figure 1: IdentifierType as defined in EBUCore (Tech 3293)

5. Examples of Usage

Example 1: International Standard Recording Code expressed in XML

The following fragment shows how to convey the International Standard Recording Code (ISRC) of the audio in a BWF file.

```
<ebucore:ebuCoreMain xmlns:dc=" http://purl.org/dc/elements/1.1/"
xmlns:ebucore="urn:ebu:metadata-schema:ebuCore_2012">
      <ebucore:coreMetadata>
            <ebucore:identifier typeLabel="GUID" typeDefinition="Globally Unique Identifier"
            formatLabel="ISRC" formatDefinition="International Standard Recording Code"
            formatLink="http://www.ebu.ch/metadata/cs/ebu_IdentifierTypeCodeCS.xml#3.7">
                  <dc:identifier>ISRC:NOX001212345</dc:identifier>
            </ebucore:identifier>
            <!-- More optional EBUCore: titles, descriptions, contributors, rights - ->
      </ebucore:coreMetadata>
</ebucore:ebuCoreMain>
```

These few lines of XML represent an instance of the EBUCore's identifierType schema applied to ISRC.

- It contains the necessary declaration of the namespaces associated with the dc: and ebucore: prefixes
- An instance of the identifier element where:
  - The typeLabel is set as a GUID, or Globally Unique Identifier as defined in typeDefinition (optional), which distinctly differentiates works and versions of a work. The typeLink is not used as EBU doesn't have a classification scheme for identifierTypeTypes.
  - The formatLabel is set to ISRC, or International Standard Recordiong Code as defined in formatDefinition (optional), which defines the structure of the identifier. The formatLink provides a URI to the EBU classification scheme and the termID associated with the ISRC (optional)
  - It is recommended to express the required value of the identifier in the form of a URN starting with the ISRC: prefix followed by the formally registered ISRC value.

IMPORTANT NOTE: This metadata structure can be used to carry one or more identifiers and in a format other than ISRC.

Example 2: A mix stem identifier expressed in XML

The following shows how the same technique can be used to carry a custom identifier (in this case, a possible implementation of a mix stem identifier).

```
<ebucore:ebuCoreMain xmlns:dc=" http://purl.org/dc/elements/1.1/"
xmlns:ebucore="urn:ebu:metadata-schema:ebuCore_2012">
      <ebucore:coreMetadata>
            <ebucore:identifier typeLabel="MIXID" typeDefinition="mix stem identifier"
            formatLabel="URN" formatDefinition="A custom urn compliant identifier to
            identifiy a mix stem" >
                  <dc:identifier> MIXID:NOX001212345 </dc:identifier>
            </ebucore:identifier>
            <!-- More optional EBUCore: titles, descriptions, contributors, rights - ->
      </ebucore:coreMetadata>
</ebucore:ebuCoreMain>
```