**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC1/SC29/WG11**

**CODING OF MOVING PICTURES AND ASSOCIATED AUDIO INFORMATION**

**ISO/IEC/JTC1/SC29/WG11/N16962**

**Torino, Italy, July 2017**

|  |  |
| --- | --- |
| **Source** | **Systems** |
| **Status** | **Approved** |
| **Title** | **Proposed text of Whitepaper for ISO/IEC 23001-12:2015** |
| **Authors** | Sally Hattori (Fox), Joe McCrossan (Fox) |

**Summary**

This contribution provides proposed text of White Paper for ISO/IEC 23001-12:2015, Information technology — MPEG systems technologies – Part 12: Sample Variants in the ISO base media format.

|  |  |
| --- | --- |
| **Title** | **White Paper on Sample variants in the ISO base media file format** |
| **Author** | Sally Hattori, Joe McCrossan |

# Introduction

This part of ISO/IEC 23001 defines a framework for the carriage of Sample Variants in the ISO Base Media File Format (ISOBMFF), as specified by ISO/IEC 14496-12.

Sample Variants are typically used to provide forensic information in the rendered sample data that can, e.g. identify the Digital Rights Management (DRM) client. This variant framework is intended to be fully compatible with ISOBMFF and Common Encryption (CENC), as specified by ISO/IEC 23001-7, and agnostic to the particular forensic marking system used.

The Sample Variant framework uses three core constructs to define and carry Sample Variant data in ISOBMFF: Variant Constructors, Variant Byte Ranges and Variant Samples.

NOTE: The Variant Process Model described in the standard also assists in introducing the concepts.

Figure 1 shows a scenario where a sample (Sample 2) has a number of Sample Variants. Figure 1 shows 3 samples which are encrypted in a series left to right, the middle of which has variants. The top row is a conceptual depiction of what is encoded using ISOBMFF and the bottom row shows what is output after Sample Variant processing. Access to samples is under the control of Key Identifiers (KIDs) as depicted in the top row of in Figure 1. For Sample Variants, a hierarchy of KIDs is used to provide access to data, with the higher-level KIDs providing access to Sample Variant Metadata and the lower level KIDs providing access to media data.

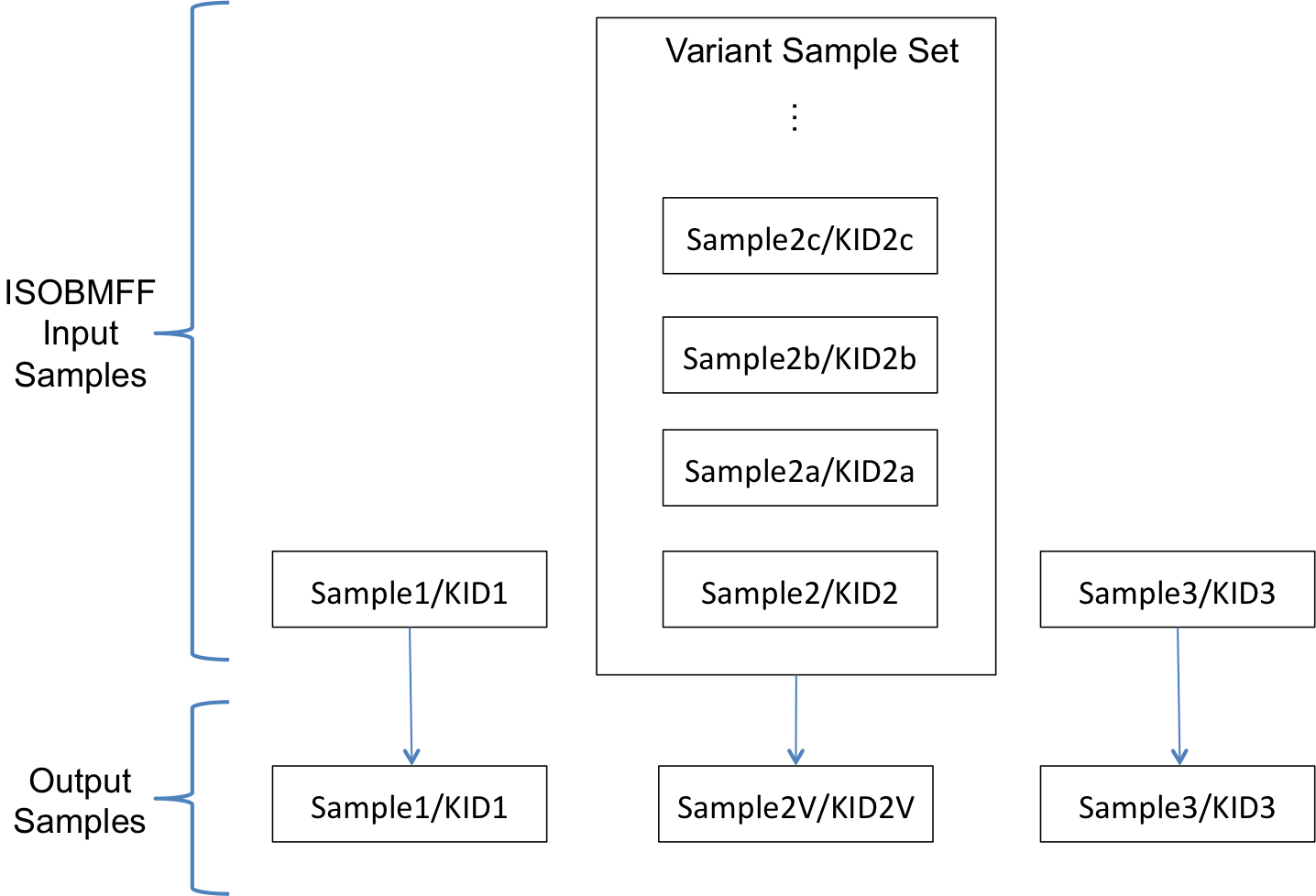


Figure 1 Sample Variant structure

The control point for the use of the proposed framework is the content publisher:

* the content publisher will encode encrypted, compressed Sample Variant data into the ISOBMFF file and ensure that each set of Sample Variant data for a given sample time is encrypted with a key and signaled with a KID.
* the content publisher will work with the DRM to manage the release of KIDs/keys such that the playback path (the actual sample data used during playback) is controlled and the player can only decrypt and render the data that it has been authorized to render.

The decoder model for the processing of the file is shown in Figure 2. Control over if and how the Sample Variants are processed is critical to the Sample Variant decoding.

Note: the decrypt and decode steps are standard operations as they would be for any CENC-enabled decoder.

By operating in the encrypted/compressed domain, secure baseband link operation (e.g. dedicated, secure video pathways) is preserved and is intended to be fully compatible with CENC.

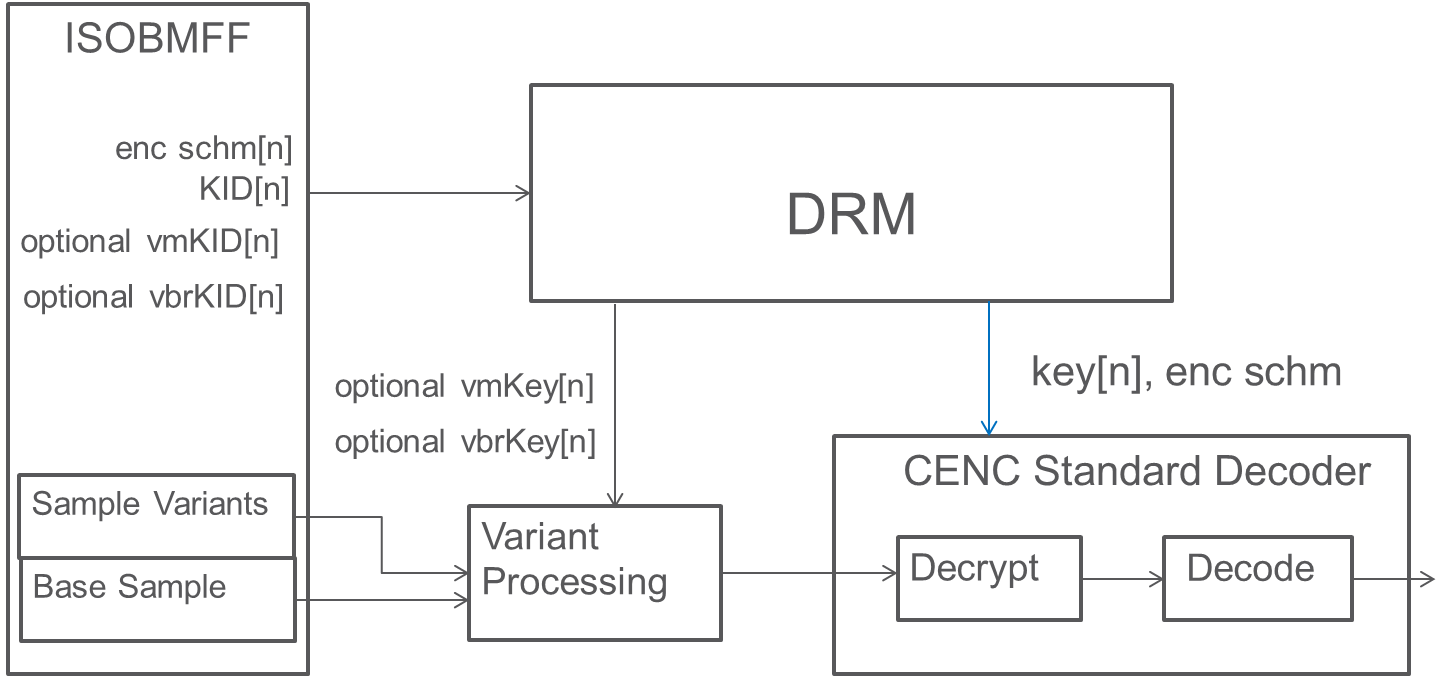


Figure 2 Variant Decoder Model

# Variant Constructors

A Variant Constructor defines which bytes are used to assemble a Sample Variant. There may be one or more Variant Constructors defined for a given ISOBMFF sample.

The Variant Processor may use a Variant Constructor if the Variant Processor has access to the Variant Constructor. In addition to the presence of the Variant Constructor, “access” includes cryptographic access. A Variant Constructor defines which data is used to assemble a Sample Variant and the associated Media KID and initialization vector for decrypting the Sample Variant.

# Variant Byte Ranges

Each Variant Constructor defines a sequence of one or more Variant Byte Ranges. Each Variant Byte Range defines the location of a sequence of bytes that might constitute bytes in a Sample Variant. Variant Byte Ranges can contain unused data.

The sequence of Variant Byte Ranges defined in a Variant Constructor are grouped into one or more Variant Byte Range groups. Each Variant Byte Range group shall define one or more Variant Byte Ranges. An individual Variant Byte Range within a Variant Byte Range group:

* May reference bytes of data that constitute bytes in a Sample Variant that is made available to certain Variant Processors (“real Variant Byte Range”).
* May reference bytes of data that are not made available to any Variant Processor (“fake Variant Byte Range”).

A “fake Variant Byte Range” can be used to hide the amount of actual “real Variant Byte Ranges” defined within a Variant Constructor. The Variant Processor uses all Variant Byte Ranges that it has access to. In addition to the presence of the Variant Byte Range, “access” includes cryptographic access.

Data for different Sample Variants can be stored non-contiguously as referenced by different Variant Constructors. Data for a particular Sample Variant can also be stored non-contiguously using a sequence of two or more Variant Byte Ranges.

# Sample Variants

The data used for rendering a sample is defined by either a Variant Constructor (if the Variant Processor has access to the Variant Constructor for the sample), or by the media data defined by ISOBMFF. When Variant Constructors are used, the actual data used for reconstructing the sample is obtained by assembling, in the order of appearance in the Variant Constructor, the byte data referenced by the Variant Byte Ranges made available to the Variant Processor and this construction shall result in a valid encrypted sample for the signaled underlying encryption system – this sample is a Sample Variant.

# ISO Storage

Variant data is stored in an ISOBMFF metadata tracks (variant track). An ISOBMFF media track (media track) or variant track may be associated with one or more variant tracks as defined in the standard.

* When an association is established between a media track and a variant track, Sample Variant processing will occur whenever a decoder (or Sample Variants Media Track Extractor) does not have access to the KID/key and the protection scheme defined for a sample in the media track as defined in the standard.
* When an association is established from a variant track (original variant track) to another variant track (other variant track), variant data contained in the other variant track can be utilized by the original variant track.
* Samples within associated tracks are associated if they are time-parallel as defined in the standard.

# Variant Processor Model

The rendering of a sample is expected to satisfy the observable behavior defined by the Variant Processor Model defined in the standard.

# Example Application

The Sample Variant framework can be used in an application in which the content providers assigns different playback path of Sample Variants for each device as additional layer of content protection. By assigning the pre-defined playback path for each device, the content providers can detect the specific device which the content was played from.

# References

[1] ISO/IEC 14496-12, Information technology — Coding of audio-visual objects — Part 12: ISO base media file format

[2] ISO/IEC 23001-7:2014, Information technology — MPEG systems technologies – Part 7: Common encryption in ISO base media file format files

[3] ISO/IEC 23001-12:2015, Information technology — MPEG systems technologies – Part 12: Sample Variants in the ISO base media format