Ref. Ares(2017)2745723 - 31/05/2017

<table>
<tr><td>Project acronym: LADIO

Project number: 731970

Work package:

Deliverable number and name:

D2.1: Documentation of extensions for EBU CCDM, EBUCore and a newly defined REST API</td><td>Title: Data Model and API

Work Package: WP2

Version: 1
Date: May 31, 2017

Author:
Carsten Griwodz</td></tr>
<tr><td>Type:
[X] Report
[ ] Demonstrator, pilot, prototype
[ ] Website, patent filings, videos, etc.
[ ] Other</td><td>Co-Author(s):
Thomas Eskénazi
Benoit Maujean

To:
Albert Gauthier, Project Officer</td></tr>
<tr><td>Status:
[ ] Draft
[ ] To be reviewed
[ ] Proposal
[X] Final / Released to EC</td><td>Confidentiality:
[X] PU – Public
[ ] CO – Confidential
[ ] CL - Classified</td></tr>
<tr><td colspan="2">Revision:
Final</td></tr>
<tr><td colspan="2">Contents:
Deliverable 2.1. Documentation of extensions for EBU CCDM, EBUCore and a newly defined REST API.</td></tr>
</table>

Table of contents

Introduction

Work package 2 must ensure that the structure and relations of the data collected during the work on set is maintained, along with information about position on a common timeline, spatial position and orientation of recording devices. To reduce the manual effort that is involved in re-discovering information in post-production that is available but lost during on-set recording, information that can be extracted and recorded on set must be stored in well-known formats, preferably such that are already standardized or that can be proposed for standardization.

While the immediacy of sharing information is handled by work package 1, work package 2 discovers how information should be stored, based on the expected patterns of access and use. This includes both data and metadata, including data that is generated in offline steps, and maintenance of all information that is required to align different media in time and space.

Task 2.1 is meant to formalize the needs of LADIO for data collection and exchange. Due to the known wide topical coverage of the EBU Class Conceptual Data Model (CCDM), which is a de-facto and de-jure standard forming the basis for a multitude of interchange formats that exist in the broadcasting world, we take this as a starting point for our considerations. In spite of the multitude of meta-information that is already covered by the EBUCore schema derived from the CCDM, we find that it is still limited as a meta-information interchange format between on-set recordings and post-processing needs. We identify modeling limitations of the EBU CCDM that prevent it from satisfying LADIO's needs, starting with limitations that were already known during proposal writing (Section 1.3.4 (3) of the LADIO proposal).

We develop adequate extensions of the model to remedy this, working informally with other EBU contributors and exerting influence on standardization. We propose extension to the EBUCore schema to capture the new model elements and develop a REST API to interact with a server that implements this schema.

Known limitations

To enhance the competitiveness of European SMEs in the filmmaking arena, it is important to overcome the hurdles to the exchange of information between pre-production, on-set and post-production. LADIO will adopt models and formats that are publicly available and free of charge. The EBU develops and maintains the EBU Class Conceptual Data Model (CCDM) [**EBU3351**], an ontology that captures film production processes, and EBUCore [**EBU3293**], an XML schema for encoding CCDM information. EBUCore is not self-contained, but refers to a variety of other schemas, including such published by W3C and the BBC. Although CCDM does already encode a large amount of information, it does not encode all information that is available on and about the set and that LADIO intends to retain for post-production stages, including videos from several cameras, microphones and other kinds of sensors (DSLR camera, 360° camera, LiDAR). As CCDM is under constant development, LADIO will extend it to cover its need. LADIO has identified a set of areas where enhancements to or clarifications of CCDM will be necessary:

Generated data

Although there are concepts for continuous-time data (audio and video) as well as for processing jobs, the ontology appears to miss the concept of adding generated content back to the model, although there is no limitation in the EBUCore schema. Due to trade-offs between availability and accuracy, it is important to include also versioning and accuracy information for data.

Temporal dependencies

CCDM has a rather coarse understanding of time. It expresses time through W3C MediaFragments [W3C25], which have similar expressive power as SMIL, HyTime and MHEG. However, LADIO requires timelines to be associated with other kinds of data, such as virtual elements and data that is derived in a processing step. It requires also the encoding of accuracy information and may benefit from the ability of re-use. MPEG-7 may provide inspiration for extending CCDM to achieve this. Its Part provides media tracks with a time reference without having own media content, which supports the required functionality.

Position and orientation

CCDM does not model accurate locality information for recording devices or for virtual elements with respect to their position on set. The EBU Audio Definition Model [EBU3364] does encode a static location and orientation of audio objects, but similar data is not maintained for other media, and even this cannot represent motion of the recording device. Although such data can be encoded inside media tracks, it is desirable to extend the CCDM to enable the selection of media tracks based on recording devices' position and orientation. Inspiration for encoding may be gained in the GIS community, like KML [OGC2015], but the data changes over time and should be encoded as a datastream with an associated time base that is related to a media track. A candidate may be found in scene description, e.g. MPEG-4 BIFS, or in motion tracking, e.g. C3D.

Reconstructed set

CCDM does currently not encode the static 3D model of a set, which LADIO will reconstruct by combining LiDAR information with visible light cameras (DSLR cameras). The reconstructed set is an example of generated data that is required for generating position information. LADIO will extend EBUCore to capture the extended CCDM.

Accessibility

To access information that is stored in accordance with this schema, LADIO will use a protocol approach based on JSON/REST, since an abundance of tools is available for editing, code generation and data inspection.

Terminology

The terminology used throughout this document tries to use terms that are reasonable for people working in film production. This may at times conflict with the established terminology in computer vision research. We are therefore providing a brief list.

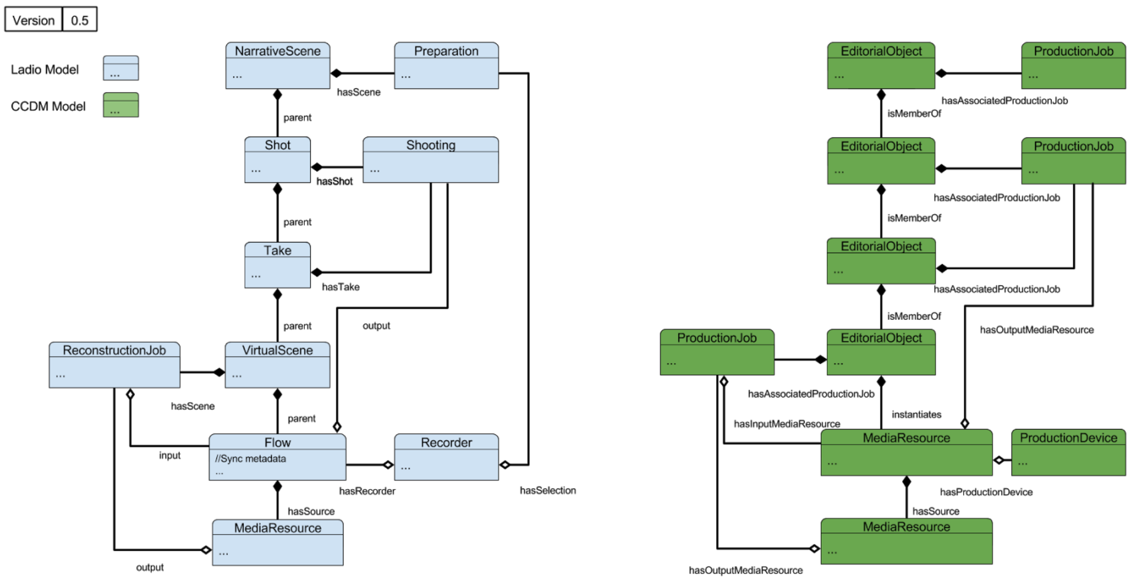


Figure 1: LADIO Model v 0.5 (see Appendix B for a bigger view)

Narrative Scene

In contrast to the computer vision understanding of the term "Scene", scene in this context refers to the place where the Shot takes place. The two are identical from an editorial viewpoint, however, the "computer vision scene" may be recomputed several times because important structural elements on the set have been modified, or because the accuracy of a reconstruction may be increased by re-computing, leading to a new instance of the "computer vision scene". The film team's understanding would consider all of this the same "scene". When we need to make a clear distinction, we will use the terms "3D Scene" for the computer vision understanding and "Narrative Scene" for the film production understanding.

Shot

In the most basic understanding, the Shot is a temporal segment of a narrative that is filmed at once (Live Action). This understanding translates directly to animated sequences that are computationally generated (Computer Graphics). The intuition changes somewhat when we refer to a reconstructed scene, because it doesn't actually have a temporal progression. Since we are bound to reconstruction scenes several times (to trade speed for quality, for example), we still have the same understanding that a Shot can have several Takes.

Take

A Take is an instance of a Shot, in the sense that the narrative segment is filmed or generated several times. For final production, at most one take is selected to represent a Shot.

Virtual Scene

A Virtual Scene is the set of all Computer Graphics elements used to render a 3D reconstruction for a Take. Since several reconstructions may occur for a given Take, there may be several Virtual Scenes by Take. If the Take may be considered as a narrative version of a Shot, defined by the Director decision, the Virtual Scene may then be considered as a technical version of a Narrative Scene.

Flow

We decided to add a new MediaResource called Flow as output of the Recorder. This level allows to handle in a intelligible way the time references to the global wallclock and local recorder drift. The actual media will be a child MediaResource of the Flow. The output of ReconstructionsJob is still a MediaResource.

For an ongoing recording, we should prefer to maintain a single Flow, even if only temporal sub-segments of the the Flow are later on useful. This applies to support Recorders (a 360 camera, a GoPro, …) that runs all day, but it also applies to a main camera rig that is not stopped between several Shots of the same Take or even several Takes (just as a matter of principle, no matter whether that ever happens on a real set). The benefit of this approach is that the shorter timelines can later been extracted using a ProductionJob as many times as necessary. The decision is delayed for as long as necessary. If we split the Flow immediately, we would have to recover the recording in case something goes wrong. If we take the decision late, we retain the link between “useful segment of the recording” and “everything that was recorded”.

Model

From these concepts, we built the LADIO model shown in Figure 1. This model is a concrete implementation of the EBU CCDM, with a strict focus on the recording aspects that are relevant for LADIO work.

Note

Whereas we developed the understanding that media attributes should be recorded in a class that is separate from the media reference itself, we had earlier distinguished between MediaResources for attribute storage and Essences for media reference. In a meeting with Tormod Værvågen (NRK, chair of the EBU’s Metadata Developer Network), we understood that Essences should only be used for the final product of a production.

Understanding of time and synchronicity

Skew and drift

Clock skew refers to a system in which the same clock signal reaches different components of a system at different times. **Clock drift** refers to a clock that does not tick at exactly the same rate as a given reference clock.

Our assumption in LADIO is that we use one reference clock on a film set, whose signal is broadcast to other components on the set. Usually there is a master timecode from the audio-department and all devices are either manually synchronized to that with their own separate sync boxes (like Lockit Sync Box from Ambient), or these Lockit boxes are on a WiFi network, and the sync gets refreshed frequently (but not during a shot) to get minimal drift. This only will work for devices that can accept external clock from Lockit devices, though. In LADIO, we extend this to devices that cannot accept SMPTE external sync like the Theta cam (60 HZ, no external clock) or the Velodyne LiDAR (20 HZ, no external sync), by pairing them with a MiniBox to translate between their local timeline and the synchronized timeline of the reference clock.

We make the assumption that it is sufficient to estimate a single multiplier value to estimate clock drift, ie. that the drift is linear. Although we know that changes in temperature and voltage can change drift factors over time, we do currently assume that our assumption is sufficient for the post-production pipelines. We encode skew as a double value according to W3C's XML Schema (http://www.w3.org/TR/xmlschema-2/).

Devices that we have available on a film set cannot necessarily estimate its own clock drift accurately, although it should be possible to estimate the minimal round-trip-time between the device and the reference clock source, and to make the assumption that the observed delay is symmetrical. We encode drift as in NTPTime [**RFC5905**], which is a 64 bit fixed point format in network byte order, where the upper 32 bits denote seconds, the lower 32 bits denote fractions of a second.

However, we know that recording devices on set are moving, meaning that an initial drift estimate will not remain entirely constant. During recording, we are not able to distinguish changes in one-way latency to the movement of the recording unit and queueing delays in wireless transmissions. As a consequence, we may have a drift estimate that is sufficient for camera movements, but there may be inaccuracies in audio timeline information due to inaccurate drift estimation. Solving this requires additional knowledge either of the signal or the physical location of the device.

Wallclock time

This should be the global time that can be interpreted as following a universal clock. It should be the basis for time alignment of all devices and recordings on set, and it should be sufficiently well aligned with the actual local clock as possible. Yet, we do not recommend that wallclock time that is recorded on set is drawn from a global time server such as a remote NTP (network time protocol) server. The reason for this is that NTP clients that follow the NTP protocol may speed up or slow down, or even jump. Contrary to this, we propose for LADIO that a single device on set generates the wallclock time reference for all devices on set, and acts as a root-level NTP server on the set.

NTP specifies how multiple servers can synchronize with each other, and how machine of a lower hierarchy level of a time-distribution system, all the way down to client machine, should together agree on a reference time. The accuracy of this time is limited by the accuracy of round-trip-time estimates between neighbouring machines in the hierarchy, and it does furthermore rely on symmetrical networking delays between nodes. If wirelessly connected nodes are moving, as we expect them to do on a film set, round-trip-time measurements vary for two reasons: the quality and load on the network that delays the transmission of packets on the network link layer, and the changing distance of the wireless node from the time reference. Given these limitations, we know already that the clocks of all the devices on set will not be perfectly synchronized by the NTP approach.

Instead, we use NTP to learn the clock skew for every individual machine, meaning that every client machine is able to estimate it locally by receiving NTP updates how much slower or faster its own local clock is ticking than the reference machine's. This skew is recorded by the individual machines, and it is stored in the metadata associated with every single Flow along with the recording device's own timestamps. We do not expect that this skew is changing in a relevant way during the recording of a single flow. Even if a flow has an extremely long lifetime, we expect that a single skew value can be computed after recording is complete.

Due to simple observation of the reference wallclock time, every machine can derive its own skew. It can not derive its absolute offset from the reference wallclock. This relies on content is discussed further in Wallclock time synchronization.

Narrative time

Narrative time is associated with a Narrative Scene, and determines the temporal placement of every Shot on the timeline that turns individual shots into a semantically meaningful sequence, forming a scene within the overall story.

There is no need for narrative time to be unique within a film, it must only make sense within a narrative scene. The advancement of narrative time is usually identical to wallclock time, unless the story demands inconsistent temporal advancement (think of superhero TV series *The Flash*, or Bullet Time like used in *The Matrix* - both have their own time reference for multiple recordings, which must be combined in post-production for a single combined Narrative Time).

We do not consider it possible that a Shot owns two narrative time attributes. If a Shot is re-used two or more times within the narrative scenes of the film, we require that a copy of the Shot is made (at least a logical copy), connected to an original Shot by means of a ProductionJob.

Recorder timeline

Recorders deliver time in the form of Timecodes, where SMTPE timecodes are the standard SMPTE-12M-2/ITU-R BT.1366-2 [**BT.1366**]. In some scenarios (namely TV broadcast productions), it is possible to synchronize the Timecodes of all recording devices, but this is not typical for larger film productions.

Recorders such as cameras are usually not programmable, and can therefore not deliver timestamps, skew information and drift values in a way that is consistent with components running LADIO software. However, in LADIO, every recorder is represented by a CamBox or MiniBox (for simplicity called a *Box* in the rest of the document, as a recording interface for various capture devices), which may be present as a pure software component, or a combination of hardware and software. It is the responsibility of the connected Box to estimate and record the skew and drift of the recorder's time with its own clock.

There are cameras whose drift from a wallclock time is changed every time a recording is started. Consequently, we store the recorder's skew and drift with every individual Flow, although this may lead to a minor overhead in case of camera rigs that deliver hardware-synchronized individual flows.

Box timeline

A Box must maintain skew and drift values for every device that is attached to it. For practical reasons, we don't expect that we are able to update time information in cameras and other Recorders in all cases, and maintaining the values on the attached device is the generic solution. We believe we can mostly ignore distance variations between the Box and the Recorder because distances are likely to stay within a radius of less than 30 m.

As mentioned in Wallclock time, each Box computes its own skew and drift from the broadcast wallclock time, as well as the skew and drift of every Recorder that it controls. It combines these two estimates to compute the total drift and skew of a recorder's time from wallclock time.

Wallclock time synchronization

As each Box makes its own drift and skew estimates, and accuracy of the NTP protocol under ideal conditions is supposed to be on the range of 10 ms, synchronization of clocks may not be sufficient for the requirements of some post-production steps.

In such cases, it is necessary to use one or more external means that make it possible to derive more accurate drift and skew estimates from the Flow's content using content analysis. Such a recomputation is a computation step that modifies the information that is carried in the metadata belonging to the Flow. Consequently, we consider the recomputation a ProductionJob that creates a new Take or a new Virtual Scene.

Conceptual challenges

Post-production use of wallclock time

We assume that all devices on set derive their mapping to wallclock time, and consequently, the translation between recorder time and narrative time, from a single source on the filmset. This single source is probably not available during post-production, and it does probably also have its own skew and drift relative to an abstract universal wallclock time.

As a consequence of this, drift will exist between clocks on computers creating content in post-production and the material that is recorded on set. Whereas pure digital effects can be added based on the recorded timeline, drift compensation is once more necessary when real-world recordings are merged during post-production. We assume that all real-world recordings are to be normalized to a single timeline as in Wallclock time synchronization.

Terminology

The 'Shot' terminology above appears limited. When there is a scene where the director wants several cameras to choose the view during editing, several Shots are concurrently defined for the same Narrative Scene on the set level.

Assuming that we have several "main" camera rigs, whose timeline (Recorder timeline) is not pre-determined (ie. where we don't create the metadata informing us about the start time and end time of a shot immediately while shooting), then we can take the same approach as in the proposed approach for using free-running support recorders, which could basically be running all day.

The approach is that the rigs' Recorder timeline is aligned with the set's Wallclock timeline (from the SetBox or another unit), each Recorder knows its own reference to the wallclock time and its own drift. Consequently, if we select the shot timeline by any means (this may be as primitive as watching the video and marking frame), we can use a ProductionJob to extract it and create a new Shot with set of MediaResources that are referring to the content of the original media resource, but cover only the new timeline.

Creation of Narrative time

The composition of VirtualScenes from a variety of real and virtual elements, which are input to its creation, requires that individual recordings are sufficiently time-aligned. The main camera (or cameras) tends to be time-aligned to a recording start, which creates the Recorder timeline, which is positioned on the wallclock timeline by mappings created by the Box (and possible later corrections).

There is an intuitive assumption that all supporting devices (witness cameras, 360 cameras, various microphones) should have Recorder timelines whose time origins are very close to the main camera's Recorder time origin for every Take.

However, this is not a reasonable expectation in practice. It is rather likely that many Recorders with supporting tasks (360 cameras, secondary microphones), which are recording data to increase the knowledge that exist in post-production about the conditions on set during the live action shoot, will start recording at the beginning of a shooting day and stop at the end of the day. Consequently, the major of the content referred to by their MediaResource will be irrelevant for a particular Take. They are also forming their own hierarchy of Shot and Take within a Narrative Scene, since they cannot be aligned with a specific Take.

To make use of these resources in a VirtualScene and create the final production, we must relate all content to the same Narrative timeline. A live action Take inherits its Narrative time from its parent Shot, while it has its own Recording timeline and placement on the Wallclock timeline.

We use Wallclock time alignment of both, the live action Take's Recording time and the supporting resource's Recording time, to identify how both are positioned on the Wallclock timeline. Then, we can use a ReconstructionJob to extract a MediaResource that refers only to the data that was recorded during the time interval that is relevant for the Take. The resulting MediaResource inherits the Narrative time of the live action Take and is added to the live action Take as an additional Flow. On most modern media formats, the extraction can be performed without copying the media data itself, meaning that the output creates only additional metadata information.

The resulting relationships between temporal information and the LADIO model is illustrated in Figure 2.

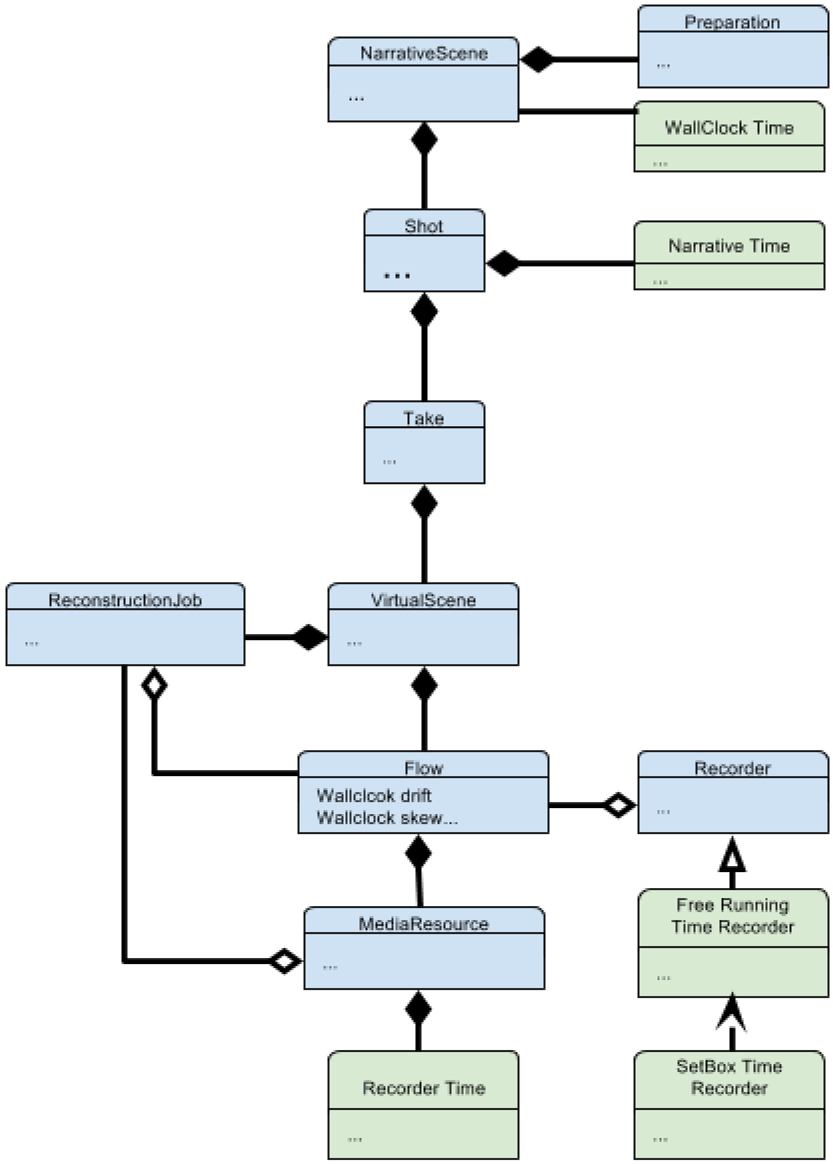


Figure 2: Time reference sketch

Changing intrinsic camera parameters

The LADIO project uses image-based reconstruction to create high-quality 3D models of a film set, and merge VFX elements into this set before final movie scenario are rendered from a combination of real recordings and the virtual elements. Obviously, the extrinsic parameters of the main camera, i.e. position, rotation and focal length, must be known perfectly at the time when VFX elements are rendered into the views. However, this is not sufficient. Also intrinsic parameters such as lens displacement and distortion, color lookup tables, sensor size, pixel aspect ratio, etc., must be known for perfect rendering. Furthermore, this information is not only essential during rendering, in spite of the successful implementation of automatic extraction of lens distortion information in Task 3.5, LADIO benefits also greatly from having it available as an input to 3D reconstruction.

All of this camera information can be stored in an EBUCore extension, as proposed by EBU in collaboration with camera manufacturer Sony and the MediaInfo framework. It is meant to result in an extension to EBU3349 [**EBU3349**], and LADIO has voiced its opinion on the extension and influenced the direction of the ongoing discussion. However, position and orientation can change continuously, and also changes in zoom and exposure occur rather frequently during the filming of a shot. Consequently, we should support recording and extraction of up-to-date Parameter information for every video frame of a camera's recording. Since LADIO is not the only project that has discovered this need, Jerome Martinez of MediaInfo provided us kindly with a set of proposals for implementing dynamic camera metadata in EBUCore. There are four proposals, all of which allow applications to retrieve parameters for every frame. For your information, we include the proposals in the Appendix A of this document.

Per frame

In this proposal, a complete set of metadata is stored for every video frame. Each frame of the video and its metadata provide a reading application with a complete knowledge about the camera's setting during the recording of this frame. No information has to be carried over from previous or later frames. All parameters are characterized a XML tags that may either be boolean or hold a single value. The XML metadata description does not contain any lists. The option alleviates interpretation, but requires a considerable amount of storage space.

Assessment

This approach is very space-consuming. It allows for a very simple relation between frames and their metadata, which can be very helpful as a format that is exposed during editing steps in post-processing. It can therefore be a very good basis for formulating a REST interface between an application in frame-oriented post-processing steps and a middleware that stored frames and metadata. Due to its space requirement, it is neither suited for storage nor for live streaming on set.

Per frame except when unique value

This is a somewhat more compact variation of the "Per frame" approach. Either in advance of recording or in a post-processing step, it is determined whether there are parameters that cannot or did not change during the recording. These parameters are encoded as metadata for the very first recorded segment (which is understood to be a single frame or, in an alternative interpretation, to hold a single frame). Metadata blocks are subsequently created for all other frames of the video, including a completely self-contained set of parameters for every frame, but restricted to those parameters that have changed during recording. Compared to the "per frame" approach, a lot of storage space or communication cost is saved, while using application have very limited additional overhead in extracting camera parameters. Instead of receiving all information from the processed frame itself, the using application requires the meta information from the initial frame and the currently processed frame.

Assessment

This approach provides a compromise between the easy accessibility of "Per frame" and the very compact representations "Segment then parameter and "Parameter then segment". In principle, it is very well suited for streaming operation, because parameter changes can be stored or transmitted in a metadata stream immediately for every frame.

It could serve as a standardized method for storing metadata for every frame locally on the Box as it becomes available. The primary reason for storing metadata locally inside a proprietary box in the non-compact XML that is proposed here is that the storage medium could be a removable flash medium, allowing for offline transfer of the metadata without conversions.

The “Per frame except when unique value” could serve as a streaming format because parameter changes are sent per frame. The challenge is the high bandwidth consumption. A real implementation should therefore expect a domain-specific compression approach that eliminates the overhead. As long as this domain-specific compression is not standardized, the SetBox-side of the streaming communication needs driver code that matches each proprietary recorder.

For offline transfer, it is unnecessarily large. As a foundation for a REST API to access per-frame metadata, it is not as well-suited as the “Per frame” proposal.

Segment then parameter

This approach is more compact than the previous approaches because it deals with every parameters separately. A block of type acquisitionDataFormat encapsulates a sequence of parameter changes, where every change is associated with a range of frames called a segment. This has several benefits for compactness of the representation, and eases readability for a reader who is only interested in isolated parameter.

In the approach, a parameter (such as a zoom factor) may be unchanged for several frames. In this case, a segment of frames is encoded with a single, constant acquisitionParameter that holds this value. Then, the parameter may be changing in a linear manner. In this case, the acquisitionParameter holds two values, the start value and end value of the parameter during the linear change. As a third option, the parameter may change in a non-linear manner, but in a way that can be approximated sufficiently well by equidistant linear pieces. In this case, the acquisitionParameter holds a sequence of values that represent the corner values of the linear intervals. The encoder must choose whether a representation is more efficient when the representation is provided as a dense approximation by equidistant linear pieces (which may mean a large number of redundant values for long linear or long constant stretches), or whether these should be divided into several segments.

This decision is made separately for every changing parameter, whereas unchanged parameters are encoded with the initial description of a media format. Like in the “Per frame except when unique value” approach, it is easier to make this decision in a post-processing step after the video has been encoded completely.

An interesting property of this flexible approach is that parameter changes can either be ordered by parameter or by time. Ordering by time will probably result in less frequent uses of the equidistant linear piece approach, and therefore be less compact, but a reading application can process the recording parameter changes in sequence from top to bottom. The slightly more compact representation that ignores time groups the segments together by parameter, which makes it easy to extract separate timelines of change for each parameter. Obviously, an arbitrary permutation of entries is also legal. The reordering can be performed at any time.

Assessment

The proposed layout is very suitable as a compact data format for storage and offline transfer, although not as compact as “Parameter then segment”.

The proposed layout is not suitable for live streaming of metadata. Using the approach that sorts parameter changes by acquisition parameter is obviously unsuited for streaming because new <segment> nodes would have to be inserted into the metadata tree at random locations. However, this is only one sorting option, and sorting <segment> nodes by the time when the validity of a value ends provides a streamable layout. However, since the format proposes segments that encode parameters for frame ranges, rather than proposing one segment per frame, it is affected by unexpected interruptions of the network. Value changes that have occurred before the interruption would be lost from the metadata stream for all compact representations. If this problem is avoided and parameter values are sent for every frame, then the proposal “Segment then parameter” become identical in size and complexity to “Per frame except when unique value”.

Parameter then segment

In this format, which is the most compact of the four proposals, the hierarchy of acquisitionParameter and segment is reversed, such that the stretches of constant, linearly interpolated or equidistant linearly interpolated parameter values are described for every segment of frames in correct temporal orders within one container for each parameter. The approach removes the need of specifying the keyword acquisitionParameter multiple times, leading to higher compactness.

In contrast to the “Segment then parameter” approach, this approach isolates the changing parameters from each other and forces a separate timeline for each of them. If this is feasible for processing applications, it can lead to very compact representation that are easy to read for humans and machines. However, it does not provide an opportunity for reading parameter groups that change together over time.

Assessment

The proposed layout is highly suitable as a compact data format for storage and offline transfer.

The proposed layout is not suitable for live streaming of metadata. The first reason is that sequences of parameter value changes are grouped by acquisition parameter, and a streamed representation would require the ability to insert new <segment> node into the <ebucore:coreMetadata> at fairly arbitrary locations. While this is technically possible using XPath requests, this comprises a sequence of commands rather than a simple stream of data. The second reason is that the format proposes segments that encode frame ranges, rather than proposing one segment per frame. An unexpected interruption of the network can thus loose parameter updates for a potential large number of frames. This could be fixed easily by promoting one update per frame.

Assessment in the context of LADIO

In LADIO, we can clearly distinguish between two phases of recording, during which we will acquire parameters from recording devices. First, there is recording by the Box, from which data should be streamed to the SetBox. Second, there is the SetBox, which serves as a collection and organization hub that prepares data for use in post-processing steps, but which does have a secondary function to provide data for on-set reviews, live previews, and on-set post-processing.

Clearly, LADIO must design for live wireless streaming from the Box to the SetBox, although local recording on Box and offline transfer to the SetBox is a more essential requirement. Although LADIO puts a priority on streaming of the data from recording devices, we accept that data connection on set will be interrupted intermittently. Furthermore, network conditions on set may require that live streaming from

Box to SetBox uses a low-quality encoding. In both cases, full quality data remains stored on the Box and is delivered after the shoot, either by transfer over the network or using a flash disk.

Within the LADIO project, we must consider whether the transfer, which is internal to the project, should be implemented in a standards-compliant manner. It is well-known that XML data representations are bandwidth-inefficient, with a large number of engineering activities focussed on domain-specific, schema-aware compression methods to reduce this overhead. On many film sets targeted by LADIO, bandwidth is precious, such that uncompressed XML representations are not a reasonable choice.

On the other hand, cameras and other recording devices are increasingly self-contained with considerable computing capabilities and wired as well as wireless connections. Especially low-end devices similar to webcams are bound to deliver data by quasi-standardized means. Since we expect this trend towards self-contained devices to expand, and with the commoditization of the broadcast market, we should also expect that standardized formats for streaming metadata as well as data directly from cameras will appear soon. Eventually, all functions that the LADIO Box performs would be built into the camera itself. There are good reasons for such cameras to implement EBUCore-compliant data streams. In such a setup, we would require a live-streamable variation of the EBUCore choice for camera parameters.

However, even when this is the case, we expect that a film set is equipped with a data collection hub that allows the DIT (Digital Imaging Technician) to coordinate and maintain control over all devices on set. One particular function that is very difficult to perform without such a central hub is time control, which is discussed in our section on Understanding of time.

For this reason, LADIO follows a combination of two options:

1. We develop a proprietary, compact data transfer mode based on “Per frame except when unique value” for streaming transfer of camera metadata from Box to SetBox as well as for storing a backup representation for offline transfer in the Box itself.
2. We propose to select the “Parameter then Segment” approach for highly efficient offline transfer of camera metadata between nodes on set as well as from the set to post-production steps. It is the SetBox that converts data from the streamed data format into the “Parameter then Segment” format, because it is the responsibility of this box to gather technical metadata as well as editorial metadata from production tracking tools.

SetBox storage

Data arrangement on the SetBox allows for 3 ways of data organization:

1. Flat files following a rigorous naming scheme
2. Relational representation that can be held in a classical relational database
3. Graph representation that can naturally express XML in-memory tree structures and be stored in graph databases

In terms of data arrangement in the processing machines (including the SetBox), there are two reasonable choices: a relational representation and a graph representation. The advantage of the relational way is that it can be stored easily in database, with well-known access methods and efficient storage and memory management. The graph representation makes it easier to export to other formats, extract subgraphs, and possibly also import from other formats. The decision is not obvious from this.

However, we assume that we are going to build a variety of temporal relationships between various assets, some of which are recorded, and others that are generated on set but treated as if they are recorded. The temporal relationship between them must be maintained.

It is probably easier to maintain these relationships in a graph representation, which can be built up in memory from a flatter representation.

At the current state of the LADIO project, we are not yet able to decide for any of the three options. We will get closer to an answer in Deliverable 2.2, but since we are still gaining experience with the trade-offs between efficient storage and transfer, and the usefulness for interaction with other applications (either through live interfaces or export modules), we will not make a solid, final decision based on experience before implementing the LADIO Application.

Encoding spatial dependencies

A central task of LADIO Task 2.1 is to develop ontology extensions that fulfill the requirements of LADIO. We already identified some limitations in the LADIO Technical Annex, specifically concerning LADIO's requirement to formulate and encode temporal and spatial dependencies.

The large number of recording devices that have already started to appear on films sets and that are expected to become even more prolific on future film sets, we must consider how their recordings can be used appropriately. These devices are naturally spread out over the set, but some of them are recording in advance of filming, some afterwards, some are only sporadically recording, and many are active during the shoot itself. Even when they are recording, they have a variety of sampling rates that require synchronization to a global clock to align them with each other. For an accurate integration of virtual audiovisual elements into a scene, it may even be important to align the time progressing of their separate streams even more accurately, and sample-accurate timing must be achieved. Within devices recording functions, accurate timing that is concerned with the time progression ("ticking") of the devices internal clocks, their relative clock skew (speed difference between different devices' clocks) and clock drift (uneven clock progression of a device, usually due to voltage variations). Even if LADIO will not be able to achieve compensation for each and every one of these challenges, because we will not be faced with every imaginable use case, task 2.1 is exploring how the appropriate spatial and temporal dependencies can be encoded and stored, and will interact with EBU contributors to raise awareness of current standards' limitations and introduce additions to the standardization process.

Relative spatial position and movement of devices

The LADIO project will use image-based 3D reconstruction and LIDAR scans, as well as combinations of these, to acquire a 3D model of Scenes. The (potentially changing) spatial position and orientation of devices that record visuals during shooting in such a Scene will be computed, based on content-based computation relative to these reconstructions; the same holds true for audio recording devices, although this will receive less attention because LADIO team members are not specialists in this topic.

An important difference between the derivation of positions and orientations from visual information and audio information is that it is generally impossible to derive spatial information from audio samples without ensuring a common time base first, and combining information from a longer sequence of samples, whereas a single video sample (ie. one frame) can frequently be sufficient to compute an accurate pose of the imaging device. Whereas the latter will not always be possible due to repetitive textures and untextured surfaces, the former is never possible because the speed of sound propagation is key to locating of audio receivers.

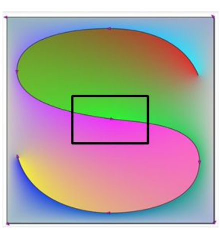
The sample rate of audio devices is so high that it is undesirable to record a stream of accurate positions, but also camera poses could be expressed in a more compact form it movement is sufficiently predictable. The current proposal to encoding camera information in EBUCore opens for partially linear segments. We are proposing a more compact way of representing movement in space, which is inspired by SVG.

We have the challenge of encoding of position and orientation information.

1. One feasible approach is the association of a camera pose relative to one or more other content elements for every individual frame that is recorded by the camera. This is consistent with LADIO's approach to perform image-based tracking based on a reconstructed scene.
    - Association of each frame with a particular version of the reconstructed static scene
    - Association of each frame with other frames of other recording devices
    - Association of different frames of the same recording device during one recording session

   This approach can be used with all of the approaches proposed in Changing intrinsic camera parameters.
2. Another feasible approach follows the previous approach but encodes only changes. This is the same refinement that has already been made for general camera parameters, by restricting encoding to changes only.
3. A third approach may use partially continuous functions to encode pose changes over time. It is very unusual for XML-based formulations and relational database content to be encoded in this way but not unheard of. The first example is implicit in the parameter encoding approach Parameter then segment. This function is a non-equicontinuous piecewise linear function that is represented by equidistant sample points.

   As another example, the image-encoding format SVG [**SVG2011**] has recently introduced functions to describe rendered images more efficiently. SVG understands the CSS expression calc() that provides one expression that evaluates differently for range of pixels and times. SVG has also adopted Diffusion Curves [**OBP+2013**] (value-crease, value-tear-crease, slope-value) that encode variants of splines following an explicitly encoded path while changing color values.



```
<defs>
  <dcPath id="dc2_boundary" d="..." boundaryColors="rgb(255,255,255)" />
  <dcPatch id="dc2" boundaryColor="black" contentUnits="objectBoundingBox" boundary="url(#dc2_boundary)">
    <dcPath d="...">
        <dcStop offset="0" left-color="redish" right-color="cyanish" />
        <dcStop offset="50%" left-color="greenish" right-color="pinkish" />
        <dcStop offset="100%" left-color="blueish" right-color="yellowish" />
    </dcPath>
  </dcPatch>
</defs>
```

For the formulation of relation recorder positions, we propose to introduce an extensible set of standard functions whose parameters can easily be derived from sample points. That means, we propose an intermediate approach following the third option. While the proposed Diffusion Curves are a feasible way of encoding color and brightness patterns in an image, we require different kinds of functions.

Examples of functional curves beyond piecewise linear paths that a recorder might follow are: circular, elliptical, accelerating or ballistic. Figure 3 shows a camera rig that is mounted on a rotating arm on a sled; the camera movement can be well-predicted or even automatized, so appropriate movements can be described in segments. The components of these functions may be encoded in independent planes instead of 3D, or directly in 3D space. If they are encoded in 3D space, the circular function is defined by 3 points, the ellipse by 5, acceleration by 3 and the ballistic curve by 4. So, instead of deriving the parameter set from the measured coordinates, LADIO proposes to store the appropriate function, which can be derived from a mechanical camera or microphone rig, record a set of coordinate samples in the same way as Parameter then segment, and impose the calculation for intermediate values on the processing application.

Figure 3: camera rig on sled, with arms resulting in rotational movement

The original proposal could approximate a solution to the challenge of movement by encoding a camera position for every position through an acceleration phase. Let’s assume an initial position of (0;0;0), linear motion (0.5;1.25;0.001) and with linear acceleration in the direction of motion of 1.0 unit^2/s. The proposed camera parameter specification inspires a piecewise linear estimate like the following.

```
&lt;segment startTime="00:00:00.000" endTime="00:00:02.000">
 <acquisitionParameter name="AbsoluteCoordinate" interval="0.033">
  <xyzPosition="(0;0;0)">
  <xyzPosition="(0.0205942;0.0514856;4.11884e-05)">
  <xyzPosition="(0.0423769;0.105942;8.47538e-05)">
  <xyzPosition="(0.065348;0.16337;0.000130696)">
  <xyzPosition="(0.0895076;0.223769;0.000179015)">
  <xyzPosition="(0.114856;0.287139;0.000229711)">
  <xyzPosition="(0.141392;0.35348;0.000282784)">
  <xyzPosition="(0.169117;0.422793;0.000338234)">
  <xyzPosition="(0.19803;0.495076;0.000396061)">
  <xyzPosition="(0.228132;0.570331;0.000456264)">
  <xyzPosition="(0.259422;0.648556;0.000518845)">
  <xyzPosition="(0.291901;0.729753;0.000583802)">
  <xyzPosition="(0.325568;0.813921;0.000651137)">
  <xyzPosition="(0.360424;0.90106;0.000720848)">
  <xyzPosition="(0.396468;0.99117;0.000792936)">
  <xyzPosition="(0.433701;1.08425;0.000867401)">
```

```
    <xyzPosition="(0.472122;1.1803;0.000944243)">
    <xyzPosition="(0.511731;1.27933;0.00102346)">
    <xyzPosition="(0.552529;1.38132;0.00110506)">
    <xyzPosition="(0.594515;1.48629;0.00118903)">
    <xyzPosition="(0.63769;1.59422;0.00127538)">
    <xyzPosition="(0.682053;1.70513;0.00136411)">
    <xyzPosition="(0.727605;1.81901;0.00145521)">
    <xyzPosition="(0.774345;1.93586;0.00154869)">
    <xyzPosition="(0.822274;2.05568;0.00164455)">
    <xyzPosition="(0.871391;2.17848;0.00174278)">
    <xyzPosition="(0.921696;2.30424;0.00184339)">
    <xyzPosition="(0.97319;2.43297;0.00194638)">
    <xyzPosition="(1.02587;2.56468;0.00205174)">
    <xyzPosition="(1.07974;2.69936;0.00215949)">
    <xyzPosition="(1.1348;2.83701;0.0022696)">
    <xyzPosition="(1.19105;2.97763;0.0023821)">
    <xyzPosition="(1.24849;3.12122;0.00249697)">
    <xyzPosition="(1.30711;3.26778;0.00261422)">
    <xyzPosition="(1.36692;3.41731;0.00273385)">
    <xyzPosition="(1.42793;3.56981;0.00285585)">
    <xyzPosition="(1.49012;3.72529;0.00298023)">
    <xyzPosition="(1.55349;3.88373;0.00310699)">
    <xyzPosition="(1.61806;4.04515;0.00323612)">
    <xyzPosition="(1.68382;4.20954;0.00336763)">
    <xyzPosition="(1.75076;4.3769;0.00350152)">
    <xyzPosition="(1.81889;4.54723;0.00363778)">
    <xyzPosition="(1.88821;4.72053;0.00377643)">
    <xyzPosition="(1.95872;4.8968;0.00391744)">
    <xyzPosition="(2.03042;5.07605;0.00406084)">
    <xyzPosition="(2.10331;5.25826;0.00420661)">
    <xyzPosition="(2.17738;5.44345;0.00435476)">
    <xyzPosition="(2.25264;5.63161;0.00450529)">
    <xyzPosition="(2.32909;5.82274;0.00465819)">
    <xyzPosition="(2.40673;6.01684;0.00481347)">
    <xyzPosition="(2.48556;6.21391;0.00497112)">
  </acquisitionParameter>
&lt;/segment&gt;
```

Our proposal enables a more compressed format, which allows a processing application to interpolate positions between frames as well. A linear acceleration as above, for example, is determined by positions at *startTime*, *endTime* and *startTime+(endTime-startTime/2)*, as follows:

```
&lt;segment startTime="00:00:00.000" endTime="00:00:02.000">
  <acquisitionParameter name="AbsoluteCoordinate" function=”LinearAcceleration”>
    <function=”LinearAcceleration” xyzPos1="(0;0;0)” xyzPos2="(0.871391;2.17848;0.00174278)" xyzPos3="(2.48556;6.21391;0.00497112)">
  </acquisitionParameter>
&lt;/segment&gt;
```

Furthermore, the ability to specify a function that is known to be correct for a recorder movement path (such a turning on an arm or accelerating on a sled along a rail), allows a much more accurate reconstruction of intermediate positions and thus, more accurate spatial synchronization with other recorders that operate at different sample rates or that are offset by a known drift value.

Association of each frame with a particular version of the reconstructed static scene

The assumption here is that the scene is reconstructed in advance of filming and can be considered a ground truth for the relative poses of those cameras that recorded the frames that have been used in the reconstruction. Both those frames and all frames that are recorded at other times can be associated with metadata that encodes the camera pose in relation to a coordinate system that is freely chosen for the reconstructed scene. Although LADIO envisions several versions of reconstruction with a variety of accuracies, it may be feasible to encode this position within the extended metadata block following every recorded frame, which also holds the intrinsic camera parameters. One feasible encoding would be the classical 3x4 extrinsic camera matrix that defines a position relative to a world coordinate system. Another feasible encoding would encode three 3-dimensional vectors for (1) camera center position, (2) camera view axis orientation that also translates the focal length into world coordinates, and (3) a unit vector encoding the up direction.

Association of each frame with other frames of other recording devices

Such an association can be created by first, time-aligning the videos streams of two cameras, then using the triangulation of matching feature points to identify the relative camera pose for each pair of frames to each other. In situations where the recorded scene can be considered static, accurate time-alignment may be avoided and replaced by coarse time-alignment sufficient to distinguish major changes in the recorded scene, for example, lighting. These derived relative poses can be associated with every frame. First, it should be noted that it is important to keep this information, because it is the basis for the scene reconstruction, which is in turn the basis for the creation of the reconstructed scene and world coordinate system assumed above. It would also create a basis for later efforts at object segmentation and change detection and tracking. Given a large number of cameras, it is impractical to associate these relative poses in the metadata associate with every frame. The relative poses should be stored in a separate (probably relational) data structure, which can uniquely identify frames across all times and recording devices.

Association of different frames of the same recording device during one recording session

Technically, this can be achieved like computing the relative pose between devices, but it allows to track the movement of a recording device. Besides the triangulation through matched features points, it is reasonable to use techniques such as optical flow computation or simultaneous localization and mapping (SLAM) approaches.

Relative temporal position

We are following the same patterns that we have explained for the intrinsic parameters in the section Changing intrinsic camera parameters. The storage of all three options for expressing spatial dependencies between frames can fit into each of the four formulations, but it can also benefit from the compression of “Parameter then Segment”.

Maintaining accuracy information

LADIO intends to provide support for the integration of virtual elements at several stages of film making. Although LADIO concentrates on retaining as much information from the live action filming as possible and make it available for post-production stages, there is a general trend towards on-set post-production. To verify the correct functioning of recording devices including temporal and spatial alignment, as well as for on-set post-processing, LADIO relies to a large degree on 3D reconstruction. The constraints of film-making, such as a brief accessibility of a film set, will sometimes force such reconstructions in a very short time period. Since computing power for doing this will be limited, one can expect a reconstruction of limited accuracy that is just sufficient for providing previews on set. These reconstruction are too inaccurate for the final product, and 3D reconstruction should later be repeated using more time and computing power.

This repetition of a stage with the goal of creation a higher accuracy and better quality, or even exploring several options for reconstruction and finally making a selection, must be encoded in LADIO.

Accuracy of spatial reconstruction

Task 3.3 ("Assessing the quality on reconstructed points and camera poses") will provide the means for generating spatial accuracy information in the considered unit, e.g., pixels. Task 2.2 ("File and database formats for data storage") chooses the file format options for encoding accuracy information in LADIO. This task and its deliverable 2.1 is concerned with the maintenance of relationships between recorded and generated data with the set of standards published by the EBU.

It must be understood that the relation between several reconstructions of different qualities is similar to the relation between several Takes of the same Shot.

In the terminology of the LADIO model (Figure 1), a 3D Reconstruction is a ProductionJob that takes one or more Flows as input and creates a MediaResource. Such input Flows may be a combination of several media. In LADIO, we envision photographs, videos and LIDAR scans.

The resulting MediaResource is a stand-alone resource, but it becomes associated with a VirtualScene. Since this VirtualScene inherits Take, which is a well-known keyword in film production for a repeated shooting of the same Scene, allowing to select one Take for inclusion in the final production, this is also the natural place for maintaining alternative versions of a 3D reconstruction. These 3D reconstructions may differ by the selection of Flows that are used as input, but they differ also in creation time, selection of reconstruction software and software parameters. This information is encoded in the ReconstructionJob object. In LADIO, we aspire to create accuracy information for a reconstruction at the same time as making the reconstruction itself; in other contexts, accuracy may have to be estimated by studying input input data and parameters, or by analysing the output itself.

A high-level accuracy estimate may be associated as an aggregate value in the ReconstructionJob object itself, but the per-pixel or per-voxel accuracy information that is generated in LADIO is an additional MediaResource that is also an output of ReconstructionJob.

Relation to the SCENE project

The FP7 project SCENE was a ground-breaking project that provided a very flexible 3D renderer to enable the compatibility between computer graphics and live action elements. The project is providing an easy-to-use, flexible programming API to use the functionality that was already developed in the SCENE project. It is offered as a C++ library that can be easily integrated into other projects. The "Scene Representation Architecture" (SRA) can be acquired at:
http://scene.cs.uni-saarland.de/index.php?content=publications

The limitation of the SCENE approach is that the work is built around a particular engine that was developed in the project. In LADIO, we must be capable of interchanging data and metadata from the production process between potentially many partners involved in a film production, who are already using a variety of established tools. Due to that, we have been inspired by SCENE, but are focussing on the exchange of information independent from a particular engine, rather than extending the SCENE project.

In the LADIO project, we are more interested in providing a REST API to allow our own as well as third party applications to interact with the model in an atomic manner. The use of a REST API provides us both with a language-independent specification, as well as an implicit understanding of distributed access.

REST API

Principles

We choose a straightforward approach regarding the REST API. Indeed, a model with a clear hierarchy where most classes compose other classes leads to an hierarchical API where resources can be accessed through other resources. Each Class of the model is transformed into one or more corresponding resources for which the four basic CRUD operations (Create Retrieve Update Delete) can be performed.

Notations

Following the common practice, REST resources are written as URIs like this : `/object/{objectID}` where text between { } indicates a variable.
For instance the NarrativeScene Class is an independent object which can exist by its own and therefore leads to a `/narrativeScene/{narrativeSceneID}` resource.

Atomic CRUD Resources

The pattern `/object/{objectID}` is used as follows, in what we call the atomic API :

- POST method on `/object/` with body to Create an object
- GET method on `/object/{objectID}` to Retrieve an object
- GET method on `/object/` to Retrieve a collection (with optional filters)
- PUT method on `/object/{objectID}` with body to Update an object
- DELETE method on `/object/{objectID}` to Delete an object

Resources Hierarchy

The Shot Class is composing the NarrativeScene Class and then is depending on it. It will therefore lead to a `/narrativeScene/{narrativeSceneID}/shot/{ShotID}` resource. And so on for other Classes. Note that with this hierarchy, it is easy to take care of classes that compose several classes. Namely the Shooting Class - which composes either a Shot or a Take - leads to two resources `/narrativeScene/{narrativeSceneID}/shot/{shotID}/shooting/{shootingID}` and `/narrativeScene/{narrativeSceneID}/shot/{shotID}/take/{takeID}/shooting/{shootingID}`.

With this structure of the API we can guarantee that the dependency of the Shooting relies either on one Shot or on one Take, without the need to control it at the application level. The full graph is shown in Figure 4.

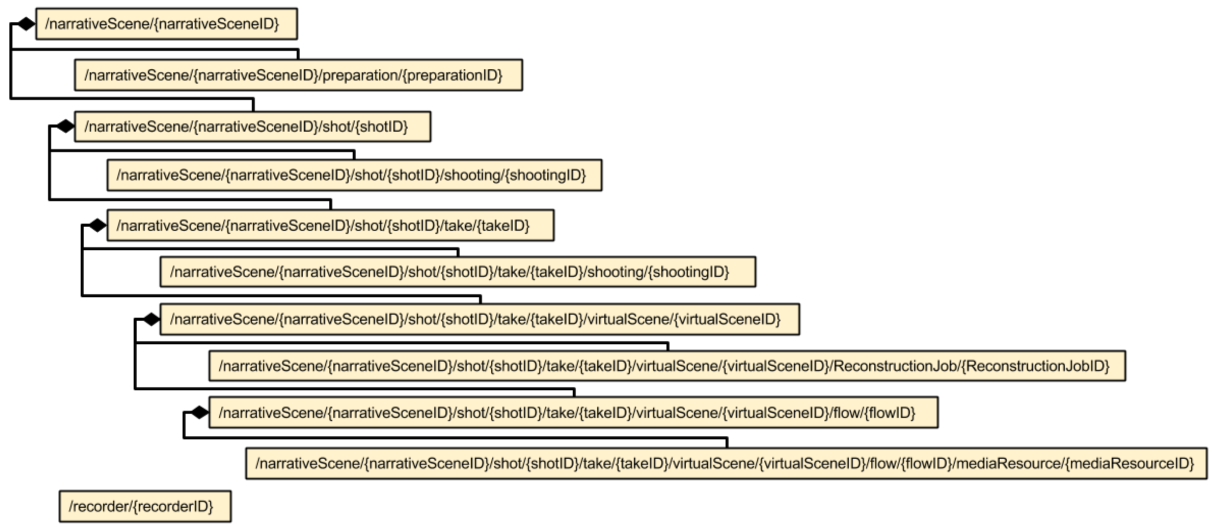


Figure 4: Full graph of the atomic REST API
(without forthcoming business resources)

Business Resources

The previous basic calls can easily be extended for specific business needs.
For example, if we want to retrieve the list of all Recorders used for a given Shot, we can access the resource with the following method :

- GET method on `/narrativeScene/{narrativeSceneID}/shot/{shotID}/recorders` to retrieve a collection.

At the time we will develop the applications (WP5) we will know the specific business resources that will need to be defined in the API. Precisely, it allows us to add business resources regarding synchronization and time afterwards, if there is a need for them.

IDs and Locators

In a RESTful world, IDs are locators that allow resources to be directly accessible. Although it fits perfectly for web applications on the internet that performs a lot of micro calls to resources, it is not optimal for our environment. Indeed, post production developers do not need to know the whole onset hierarchy of an object, and moreover it is unintuitive for them since the shot and take words do not have the same meaning onset and during post production. We then decided to go for a RESTless way, where IDs and locators are two separated fields.

References

[EBU3293] EBU. Tech 3293 v1.7 EBU CORE METADATA SET (EBUCore) (2016).

[EBU3351] EBU. Tech 3351 v1.1 EBU CLASS CONCEPTUAL DATA MODEL (CCDM) (2016).

[EBU3349] EBU. Tech 3349 v1.0 EBU Acquisition Technical Metadata Set. (2012)

[EBU3364] EBU. Tech 3364 v1.0 EBU Audio Definition Model Ver. 1.0 (2014)

[RFC5905] D. Mills, J. Martin, J. Burbank, W. Kasch (2010). Network Time Protocol Version 4: Protocol and Algorithms Specification. IETF. URL http://www.ietf.org/rfc/rfc5905.txt

[SVG2011] E. Dahlström, P. Dengler, A. Grasso, C. Lilley, C. McCormack, D. Schepers, J. Watt, J. Ferraiolo, J. Fujisawa, D. Jackson (edtrs, 2011). Scalable Vector Graphics (SVG) 1.1. W3C Recommendation 16. URL https://www.w3.org/TR/2011/REC-SVG11-20110816/

[W3C25] R. Troncy, E Mannens, S. Pfeiffer, D. Van Deursen (2012). Media Fragments URI 1.0 (basic). W3C Recommendation 25. URL https://www.w3.org/TR/media-frags/

[OGC2015] Open Geospatail Consortium (2015). OGC KML v 2.3. URL http://docs.opengeospatial.org/is/12-007r2/12-007r2.html

[BT.1366] ITU-R (2009). Recommendation BT.1366-2 “Transmission of time code and control code in the ancillary data space of a digital television stream according to Recommendations ITU-R BT.656, ITU-R BT.799 and ITU-R BT.112”. https://www.itu.int/rec/R-REC-BT.1366-2-200901-I/en (better known as SMPTE 12M-2)

[OBP+2013] A. Orzan, A. Bousseau, P. Barla, H. Winnemöller, J. Thollot, D. Salesin (2013). Diffusion curves. *Communications of the ACM*, *56*(7), 101. http://doi.org/10.1145/2483852.2483873

Appendix A

Per frame


```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- Generated by MediaInfoLib - v0.7.91 -->
<ebucore:ebuCoreMain xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:ebucore="urn:ebu:metadata-schema:ebuCore_2015"
  xmlns:xalan="http://xml.apache.org/xalan" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ebu:metadata-schema:ebuCore_2015
  https://www.ebu.ch/metadata/schemas/EBUCore/20150522/ebucore_20150522.xsd"
  version="1.6" dateLastModified="2016-12-15" timeLastModified="08:54:38Z">
 <ebucore:coreMetadata>
  <ebucore:format>
   <ebucore:videoFormat videoFormatName="AVC">
    <ebucore:width unit="pixel">1920</ebucore:width>
    <ebucore:height unit="pixel">1080</ebucore:height>
    <ebucore:frameRate factorNumerator="30000" factorDenominator="1001">30</ebucore:frameRate>
    <ebucore:aspectRatio typeLabel="display">
     <ebucore:factorNumerator>16</ebucore:factorNumerator>
     <ebucore:factorDenominator>9</ebucore:factorDenominator>
    </ebucore:aspectRatio>
    <ebucore:videoEncoding typeLabel="High 4:2:2@L4.1"/>
    <ebucore:codec>
     <ebucore:codecIdentifier>
      <dc:identifier>0D01030102106001-0401020201316001</dc:identifier>
     </ebucore:codecIdentifier>
    </ebucore:codec>
    <ebucore:bitRate>27896896</ebucore:bitRate>
    <ebucore:bitRateMax>24999936</ebucore:bitRateMax>
    <ebucore:bitRateMode>variable</ebucore:bitRateMode>
    <ebucore:scanningFormat>progressive</ebucore:scanningFormat>
    <ebucore:videoTrack trackId="2"/>
    <ebucore:technicalAttributeString typeLabel="Standard">Component</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="ColorSpace">YUV</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="ChromaSubsampling">4:2:2</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="colour_primaries">BT.709</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="transfer_characteristics">BT.709</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="matrix_coefficients">BT.709</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="colour_range">Limited</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">37465801</ebucore:technicalAttributeInteger>
    <ebucore:technicalAttributeInteger typeLabel="BitDepth" unit="bit">10</ebucore:technicalAttributeInteger>
    <ebucore:technicalAttributeBoolean typeLabel="CABAC">true</ebucore:technicalAttributeBoolean>
    <ebucore:technicalAttributeBoolean typeLabel="MBAFF">false</ebucore:technicalAttributeBoolean>
   </ebucore:videoFormat>
   <ebucore:audioFormat audioFormatName="PCM">
    <ebucore:audioEncoding typeLabel="PCM" typeLink="http://www.ebu.ch/metadata/cs/ebu_AudioCompressionCodeCS.xml#11"/>
    <ebucore:codec>
     <ebucore:codecIdentifier>
      <dc:identifier>0D01030102060300-0402020101000000</dc:identifier>
     </ebucore:codecIdentifier>
    </ebucore:codec>
    <ebucore:samplingRate>48000</ebucore:samplingRate>
    <ebucore:sampleSize>24</ebucore:sampleSize>
    <ebucore:bitRate>1152000</ebucore:bitRate>
    <ebucore:bitRateMode>constant</ebucore:bitRateMode>
    <ebucore:audioTrack trackId="3"/>
```

```xml
    <ebucore:channels>1</ebucore:channels>
    <ebucore:technicalAttributeString typeLabel="Endianness">Little</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="Wrapping">Frame (AES)</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">1547136</ebucore:technicalAttributeInteger>
   </ebucore:audioFormat>
   <ebucore:audioFormat audioFormatName="PCM">
     [...]
   </ebucore:audioFormat>
   <ebucore:containerFormat containerFormatName="MXF">
    <ebucore:containerEncoding formatLabel="MXF"/>
    <ebucore:technicalAttributeString typeLabel="FormatProfile">OP-1a</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="FormatSettings">Closed / Complete</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="WritingLibrary">Sony MXF Development Kit (Win32) 4.10.0.119.1</ebucore:technicalAttributeString>
   </ebucore:containerFormat>
   <ebucore:timecodeFormat timecodeFormatName="MXF TC">
    <ebucore:timecodeStart>
     <ebucore:timecode>01:44:11;24</ebucore:timecode>
    </ebucore:timecodeStart>
    <ebucore:timecodeTrack trackId="1" typeLabel="Material"/>
    <ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
   </ebucore:timecodeFormat>
   <ebucore:timecodeFormat timecodeFormatName="MXF TC">
    <ebucore:timecodeStart>
     <ebucore:timecode>01:44:11;24</ebucore:timecode>
    </ebucore:timecodeStart>
    <ebucore:timecodeTrack trackId="1" typeLabel="Source"/>
    <ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
   </ebucore:timecodeFormat>
   <ebucore:timecodeFormat timecodeFormatName="SMPTE TC">
    <ebucore:timecodeStart>
     <ebucore:timecode>01:44:11;24</ebucore:timecode>
    </ebucore:timecodeStart>
   </ebucore:timecodeFormat>
   <acquisitionDataFormat>
    <duration>00:00:10.744</duration>
    &lt;segment startTime="00:00:00.000" endTime="00:00:00.033">
     <acquisitionParameter name="CaptureGammaEquation">BT.709</acquisitionParameter>
     <acquisitionParameter name="IrisFNumber">2.813628</acquisitionParameter>
     <acquisitionParameter name="FocusPositionFromImagePlane">1.129</acquisitionParameter>
     <acquisitionParameter name="FocusPositionFromFrontLensVertex">0.979</acquisitionParameter>
     <acquisitionParameter name="MacroSetting">Off</acquisitionParameter>
     <acquisitionParameter name="LensZoom35mmStillCameraEquivalent">437.000</acquisitionParameter>
     <acquisitionParameter name="LensZoomActualFocalLength">80.800</acquisitionParameter>
     <acquisitionParameter name="OpticalExtenderMagnification" unit="percentage">100</acquisitionParameter>
     <acquisitionParameter name="LensAttributes">Hoshinon HD17x5.5AF f1.4</acquisitionParameter>
     <acquisitionParameter name="IrisTNumber">2.884010</acquisitionParameter>
     <acquisitionParameter name="IrisRingPosition">24.8108</acquisitionParameter>
     <acquisitionParameter name="FocusRingPosition">29.1397</acquisitionParameter>
     <acquisitionParameter name="ZoomRingPosition">99.8123</acquisitionParameter>
     <acquisitionParameter name="AutoExposureMode">Full Auto</acquisitionParameter>
     <acquisitionParameter name="AutoFocusSensingAreaSetting">Manual</acquisitionParameter>
     <acquisitionParameter name="ColorCorrectionFilterWheelSetting">Color Compensation 3200 K</acquisitionParameter>
     <acquisitionParameter name="NeutralDensityFilterWheelSetting">1/8</acquisitionParameter>
     <acquisitionParameter name="ImageSensorDimensionEffectiveWidth">7.190</acquisitionParameter>
     <acquisitionParameter name="ImageSensorDimensionEffectiveHeight">4.045</acquisitionParameter>
     <acquisitionParameter name="CaptureFrameRate" unit="fps">29.970</acquisitionParameter>
```

```
    <acquisitionParameter name="ImageSensorReadoutMode">Interlaced field</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Angle" unit="degree">360.0</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Time" unit="second">1001/60000</acquisitionParameter>
    <acquisitionParameter name="CameraMasterGainAdjustment" unit="dB">0.00</acquisitionParameter>
    <acquisitionParameter name="ISOSensitivity">430</acquisitionParameter>
    <acquisitionParameter name="ElectricalExtenderMagnification" unit="percentage">100</acquisitionParameter>
    <acquisitionParameter name="AutoWhiteBalanceMode">Hold</acquisitionParameter>
    <acquisitionParameter name="WhiteBalance" unit="kelvin">5600</acquisitionParameter>
    <acquisitionParameter name="CameraMasterBlackLevel" unit="percentage">3.0</acquisitionParameter>
    <acquisitionParameter name="CameraKneePoint" unit="percentage">85.0</acquisitionParameter>
    <acquisitionParameter name="CameraKneeSlope">0.100</acquisitionParameter>
    <acquisitionParameter name="CameraLuminanceDynamicRange" unit="percentage">450.0</acquisitionParameter>
    <acquisitionParameter name="CameraSettingFileURI">file:///Camera/Setting001.dat</acquisitionParameter>
    <acquisitionParameter name="CameraAttributes">pseudo</acquisitionParameter>
    <acquisitionParameter name="ExposureIndexofPhotoMeter">430</acquisitionParameter>
    <acquisitionParameter name="GammaForCDL">Same as Capture Gamma</acquisitionParameter>
    <acquisitionParameter name="ASC_CDL_V12">sR=1.0 sG=1.0 sB=0.5 oR=0.0 oG=0.0 oB=-262144.0 pR=1.0 pG=1.0 pB=0.5 sat=2.0</acquisitionParameter>
    <acquisitionParameter name="ColorMatrix">RR=1.000 GR=0.000 BR=0.000 RG=0.000 GG=1.000 BG=0.000 RB=0.000 GB=0.000 BB=1.000</acquisitionParameter>
   &lt;/segment&gt;
   &lt;segment startTime="00:00:00.033" endTime="00:00:00.067"&gt;
    <acquisitionParameter name="CaptureGammaEquation">BT.709</acquisitionParameter>
    <acquisitionParameter name="IrisFNumber">2.821734</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromImagePlane">1.129</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromFrontLensVertex">0.979</acquisitionParameter>
    <acquisitionParameter name="MacroSetting">Off</acquisitionParameter>
    <acquisitionParameter name="LensZoom35mmStillCameraEquivalent">437.000</acquisitionParameter>
    <acquisitionParameter name="LensZoomActualFocalLength">80.800</acquisitionParameter>
    <acquisitionParameter name="OpticalExtenderMagnification" unit="percentage">100</acquisitionParameter>
    <acquisitionParameter name="LensAttributes">Hoshinon HD17x5.5AF f1.4</acquisitionParameter>
    <acquisitionParameter name="IrisTNumber">2.892319</acquisitionParameter>
    <acquisitionParameter name="IrisRingPosition">24.9146</acquisitionParameter>
    <acquisitionParameter name="FocusRingPosition">29.1397</acquisitionParameter>
    <acquisitionParameter name="ZoomRingPosition">99.8123</acquisitionParameter>
    <acquisitionParameter name="AutoExposureMode">Full Auto</acquisitionParameter>
    <acquisitionParameter name="AutoFocusSensingAreaSetting">Manual</acquisitionParameter>
    <acquisitionParameter name="ColorCorrectionFilterWheelSetting">Color Compensation 3200 K</acquisitionParameter>
    <acquisitionParameter name="NeutralDensityFilterWheelSetting">1/8</acquisitionParameter>
    <acquisitionParameter name="ImageSensorDimensionEffectiveWidth">7.190</acquisitionParameter>
    <acquisitionParameter name="ImageSensorDimensionEffectiveHeight">4.045</acquisitionParameter>
    <acquisitionParameter name="CaptureFrameRate" unit="fps">29.970</acquisitionParameter>
    <acquisitionParameter name="ImageSensorReadoutMode">Interlaced field</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Angle" unit="degree">360.0</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Time" unit="second">1001/60000</acquisitionParameter>
    <acquisitionParameter name="CameraMasterGainAdjustment" unit="dB">0.00</acquisitionParameter>
    <acquisitionParameter name="ISOSensitivity">430</acquisitionParameter>
    <acquisitionParameter name="ElectricalExtenderMagnification" unit="percentage">100</acquisitionParameter>
    <acquisitionParameter name="AutoWhiteBalanceMode">Hold</acquisitionParameter>
    <acquisitionParameter name="WhiteBalance" unit="kelvin">5600</acquisitionParameter>
    <acquisitionParameter name="CameraMasterBlackLevel" unit="percentage">3.0</acquisitionParameter>
    <acquisitionParameter name="CameraKneePoint" unit="percentage">85.0</acquisitionParameter>
    <acquisitionParameter name="CameraKneeSlope">0.100</acquisitionParameter>
    <acquisitionParameter name="CameraLuminanceDynamicRange" unit="percentage">450.0</acquisitionParameter>
    <acquisitionParameter name="CameraSettingFileURI">file:///Camera/Setting001.dat</acquisitionParameter>
    <acquisitionParameter name="CameraAttributes">pseudo</acquisitionParameter>
    <acquisitionParameter name="ExposureIndexofPhotoMeter">430</acquisitionParameter>
```

<acquisitionParameter name="GammaForCDL">Same as Capture Gamma</acquisitionParameter>
<acquisitionParameter name="ASC_CDL_V12">sR=1.0 sG=1.0 sB=0.5 oR=0.0 oG=0.0 oB=-262144.0 pR=1.0 pG=1.0 pB=0.5 sat=2.0</acquisitionParameter>
<acquisitionParameter name="ColorMatrix">RR=1.000 GR=0.000 BR=0.000 RG=0.000 GG=1.000 BG=0.000 RB=0.000 GB=0.000 BB=1.000</acquisitionParameter>
</segment>
<segment startTime="00:00:00.067" endTime="00:00:00.100">
<acquisitionParameter name="CaptureGammaEquation">BT.709</acquisitionParameter>
<acquisitionParameter name="IrisFNumber">2.829863</acquisitionParameter>
<acquisitionParameter name="FocusPositionFromImagePlane">1.129</acquisitionParameter>
<acquisitionParameter name="FocusPositionFromFrontLensVertex">0.979</acquisitionParameter>
<acquisitionParameter name="MacroSetting">Off</acquisitionParameter>
<acquisitionParameter name="LensZoom35mmStillCameraEquivalent">437.000</acquisitionParameter>
<acquisitionParameter name="LensZoomActualFocalLength">80.800</acquisitionParameter>
<acquisitionParameter name="OpticalExtenderMagnification" unit="percentage">100</acquisitionParameter>
<acquisitionParameter name="LensAttributes">Hoshinon HD17x5.5AF f1.4</acquisitionParameter>
<acquisitionParameter name="IrisTNumber">2.900651</acquisitionParameter>
<acquisitionParameter name="IrisRingPosition">25.0183</acquisitionParameter>
<acquisitionParameter name="FocusRingPosition">29.1397</acquisitionParameter>
<acquisitionParameter name="ZoomRingPosition">99.8123</acquisitionParameter>
<acquisitionParameter name="AutoExposureMode">Full Auto</acquisitionParameter>
<acquisitionParameter name="AutoFocusSensingAreaSetting">Manual</acquisitionParameter>
<acquisitionParameter name="ColorCorrectionFilterWheelSetting">Color Compensation 3200 K</acquisitionParameter>
<acquisitionParameter name="NeutralDensityFilterWheelSetting">1/8</acquisitionParameter>
<acquisitionParameter name="ImageSensorDimensionEffectiveWidth">7.190</acquisitionParameter>
<acquisitionParameter name="ImageSensorDimensionEffectiveHeight">4.045</acquisitionParameter>
<acquisitionParameter name="CaptureFrameRate" unit="fps">29.970</acquisitionParameter>
<acquisitionParameter name="ImageSensorReadoutMode">Interlaced field</acquisitionParameter>
<acquisitionParameter name="ShutterSpeed_Angle" unit="degree">360.0</acquisitionParameter>
<acquisitionParameter name="ShutterSpeed_Time" unit="second">1001/60000</acquisitionParameter>
<acquisitionParameter name="CameraMasterGainAdjustment" unit="dB">0.00</acquisitionParameter>
<acquisitionParameter name="ISOSensitivity">430</acquisitionParameter>
<acquisitionParameter name="ElectricalExtenderMagnification" unit="percentage">100</acquisitionParameter>
<acquisitionParameter name="AutoWhiteBalanceMode">Hold</acquisitionParameter>
<acquisitionParameter name="WhiteBalance" unit="kelvin">5600</acquisitionParameter>
<acquisitionParameter name="CameraMasterBlackLevel" unit="percentage">3.0</acquisitionParameter>
<acquisitionParameter name="CameraKneePoint" unit="percentage">85.0</acquisitionParameter>
<acquisitionParameter name="CameraKneeSlope">0.100</acquisitionParameter>
<acquisitionParameter name="CameraLuminanceDynamicRange" unit="percentage">450.0</acquisitionParameter>
<acquisitionParameter name="CameraSettingFileURI">file:///Camera/Setting001.dat</acquisitionParameter>
<acquisitionParameter name="CameraAttributes">pseudo</acquisitionParameter>
<acquisitionParameter name="ExposureIndexofPhotoMeter">430</acquisitionParameter>
<acquisitionParameter name="GammaForCDL">Same as Capture Gamma</acquisitionParameter>
<acquisitionParameter name="ASC_CDL_V12">sR=1.0 sG=1.0 sB=0.5 oR=0.0 oG=0.0 oB=-262144.0 pR=1.0 pG=1.0 pB=0.5 sat=2.0</acquisitionParameter>
<acquisitionParameter name="ColorMatrix">RR=1.000 GR=0.000 BR=0.000 RG=0.000 GG=1.000 BG=0.000 RB=0.000 GB=0.000 BB=1.000</acquisitionParameter>
</segment>
[...]
</acquisitionDataFormat>
<ebucore:duration>
<ebucore:normalPlayTime>PT10.744S</ebucore:normalPlayTime>
</ebucore:duration>
<ebucore:fileSize>49856048</ebucore:fileSize>
<ebucore:fileName>RDD18_fullitem_sample.MXF</ebucore:fileName>
<ebucore:locator>E:\20161009_Download\RDD18_fullitem_sample\RDD18_fullitem_sample.MXF</ebucore:locator>
<ebucore:technicalAttributeInteger typeLabel="OverallBitRate" unit="bps">37122896</ebucore:technicalAttributeInteger>

```
      <ebucore:dateCreated startDate="2016-04-19" startTime="07:33:33.000"/>
    </ebucore:format>
  </ebucore:coreMetadata>
</ebucore:ebuCoreMain>
```


Per frame except when unique value


```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Generated by MediaInfoLib - v0.7.91 -->
<ebucore:ebuCoreMain xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:ebucore="urn:ebu:metadata-schema:ebuCore_2015"
    xmlns:xalan="http://xml.apache.org/xalan" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="urn:ebu:metadata-schema:ebuCore_2015
    https://www.ebu.ch/metadata/schemas/EBUCore/20150522/ebucore_20150522.xsd"
    version="1.6" dateLastModified="2016-12-15" timeLastModified="08:49:50Z">
  <ebucore:coreMetadata>
    <ebucore:format>
      <ebucore:videoFormat videoFormatName="AVC">
        <ebucore:width unit="pixel">1920</ebucore:width>
        <ebucore:height unit="pixel">1080</ebucore:height>
        <ebucore:frameRate factorNumerator="30000" factorDenominator="1001">30</ebucore:frameRate>
        <ebucore:aspectRatio typeLabel="display">
          <ebucore:factorNumerator>16</ebucore:factorNumerator>
          <ebucore:factorDenominator>9</ebucore:factorDenominator>
        </ebucore:aspectRatio>
        <ebucore:videoEncoding typeLabel="High 4:2:2@L4.1"/>
        <ebucore:codec>
          <ebucore:codecIdentifier>
            <dc:identifier>0D01030102106001-0401020201316001</dc:identifier>
          </ebucore:codecIdentifier>
        </ebucore:codec>
        <ebucore:bitRate>27896896</ebucore:bitRate>
        <ebucore:bitRateMax>24999936</ebucore:bitRateMax>
        <ebucore:bitRateMode>variable</ebucore:bitRateMode>
        <ebucore:scanningFormat>progressive</ebucore:scanningFormat>
        <ebucore:videoTrack trackId="2"/>
        <ebucore:technicalAttributeString typeLabel="Standard">Component</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="ColorSpace">YUV</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="ChromaSubsampling">4:2:2</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="colour_primaries">BT.709</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="transfer_characteristics">BT.709</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="matrix_coefficients">BT.709</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="colour_range">Limited</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">37465801</ebucore:technicalAttributeInteger>
        <ebucore:technicalAttributeInteger typeLabel="BitDepth" unit="bit">10</ebucore:technicalAttributeInteger>
        <ebucore:technicalAttributeBoolean typeLabel="CABAC">true</ebucore:technicalAttributeBoolean>
        <ebucore:technicalAttributeBoolean typeLabel="MBAFF">false</ebucore:technicalAttributeBoolean>
      </ebucore:videoFormat>
      <ebucore:audioFormat audioFormatName="PCM">
        <ebucore:audioEncoding typeLabel="PCM" typeLink="http://www.ebu.ch/metadata/cs/ebu_AudioCompressionCodeCS.xml#11"/>
        <ebucore:codec>
          <ebucore:codecIdentifier>
            <dc:identifier>0D01030102060300-0402020101000000</dc:identifier>
          </ebucore:codecIdentifier>
        </ebucore:codec>
        <ebucore:samplingRate>48000</ebucore:samplingRate>
        <ebucore:sampleSize>24</ebucore:sampleSize>
        <ebucore:bitRate>1152000</ebucore:bitRate>
```

```
  <ebucore:bitRateMode>constant</ebucore:bitRateMode>
  <ebucore:audioTrack trackId="3"/>
  <ebucore:channels>1</ebucore:channels>
  <ebucore:technicalAttributeString typeLabel="Endianness">Little</ebucore:technicalAttributeString>
  <ebucore:technicalAttributeString typeLabel="Wrapping">Frame (AES)</ebucore:technicalAttributeString>
  <ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">1547136</ebucore:technicalAttributeInteger>
 </ebucore:audioFormat>
 <ebucore:audioFormat audioFormatName="PCM">
   [...]
 </ebucore:audioFormat>
 <ebucore:containerFormat containerFormatName="MXF">
  <ebucore:containerEncoding formatLabel="MXF"/>
  <ebucore:technicalAttributeString typeLabel="FormatProfile">OP-1a</ebucore:technicalAttributeString>
  <ebucore:technicalAttributeString typeLabel="FormatSettings">Closed / Complete</ebucore:technicalAttributeString>
  <ebucore:technicalAttributeString typeLabel="WritingLibrary">Sony MXF Development Kit (Win32) 4.10.0.119.1</ebucore:technicalAttributeString>
 </ebucore:containerFormat>
 <ebucore:timecodeFormat timecodeFormatName="MXF TC">
  <ebucore:timecodeStart>
   <ebucore:timecode>01:44:11;24</ebucore:timecode>
  </ebucore:timecodeStart>
  <ebucore:timecodeTrack trackId="1" typeLabel="Material"/>
  <ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
 </ebucore:timecodeFormat>
 <ebucore:timecodeFormat timecodeFormatName="MXF TC">
  <ebucore:timecodeStart>
   <ebucore:timecode>01:44:11;24</ebucore:timecode>
  </ebucore:timecodeStart>
  <ebucore:timecodeTrack trackId="1" typeLabel="Source"/>
  <ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
 </ebucore:timecodeFormat>
 <ebucore:timecodeFormat timecodeFormatName="SMPTE TC">
  <ebucore:timecodeStart>
   <ebucore:timecode>01:44:11;24</ebucore:timecode>
  </ebucore:timecodeStart>
 </ebucore:timecodeFormat>
 <acquisitionDataFormat>
  <duration>00:00:10.744</duration>
  &lt;segment startTime="00:00:00.000" endTime="00:00:10.744"&gt;
   <acquisitionParameter name="CaptureGammaEquation">BT.709</acquisitionParameter>
   <acquisitionParameter name="MacroSetting">Off</acquisitionParameter>
   <acquisitionParameter name="OpticalExtenderMagnification" unit="percentage">100</acquisitionParameter>
   <acquisitionParameter name="LensAttributes">HoshinonHD17x5.5AFf1.4</acquisitionParameter>
   <acquisitionParameter name="AutoExposureMode">FullAuto</acquisitionParameter>
   <acquisitionParameter name="AutoFocusSensingAreaSetting">Manual</acquisitionParameter>
   <acquisitionParameter name="ColorCorrectionFilterWheelSetting">ColorCompensation3200K</acquisitionParameter>
   <acquisitionParameter name="NeutralDensityFilterWheelSetting">1/8</acquisitionParameter>
   <acquisitionParameter name="ImageSensorDimensionEffectiveWidth">7.190</acquisitionParameter>
   <acquisitionParameter name="ImageSensorDimensionEffectiveHeight">4.045</acquisitionParameter>
   <acquisitionParameter name="CaptureFrameRate" unit="fps">29.970</acquisitionParameter>
   <acquisitionParameter name="ImageSensorReadoutMode">Interlacedfield</acquisitionParameter>
   <acquisitionParameter name="CameraMasterGainAdjustment" unit="dB">0.00</acquisitionParameter>
   <acquisitionParameter name="ISOSensitivity">430</acquisitionParameter>
   <acquisitionParameter name="ElectricalExtenderMagnification" unit="percentage">100</acquisitionParameter>
   <acquisitionParameter name="AutoWhiteBalanceMode">Hold</acquisitionParameter>
   <acquisitionParameter name="WhiteBalance" unit="kelvin">5600</acquisitionParameter>
   <acquisitionParameter name="CameraMasterBlackLevel" unit="percentage">3.0</acquisitionParameter>
```

```
    <acquisitionParameter name="CameraKneePoint" unit="percentage">85.0</acquisitionParameter>
    <acquisitionParameter name="CameraKneeSlope">0.100</acquisitionParameter>
    <acquisitionParameter name="CameraLuminanceDynamicRange" unit="percentage">450.0</acquisitionParameter>
    <acquisitionParameter name="CameraSettingFileURI">file:///Camera/Setting001.dat</acquisitionParameter>
    <acquisitionParameter name="CameraAttributes">pseudo</acquisitionParameter>
    <acquisitionParameter name="ExposureIndexofPhotoMeter">430</acquisitionParameter>
    <acquisitionParameter name="GammaForCDL">SameasCaptureGamma</acquisitionParameter>
    <acquisitionParameter name="ASC_CDL_V12">sR=1.0sG=1.0sB=0.5oR=0.0oG=0.0oB=-262144.0pR=1.0pG=1.0pB=0.5sat=2.0</acquisitionParameter>
    <acquisitionParameter name="ColorMatrix">RR=1.000GR=0.000BR=0.000RG=0.000GG=1.000BG=0.000RB=0.000GB=0.000BB=1.000</acquisitionParameter>
   &lt;/segment&gt;
   &lt;segment startTime="00:00:00.000" endTime="00:00:00.033"&gt;
    <acquisitionParameter name="IrisFNumber">2.813628</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromImagePlane">1.129</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromFrontLensVertex">0.979</acquisitionParameter>
    <acquisitionParameter name="LensZoom35mmStillCameraEquivalent">437.000</acquisitionParameter>
    <acquisitionParameter name="LensZoomActualFocalLength">80.800</acquisitionParameter>
    <acquisitionParameter name="IrisTNumber">2.884010</acquisitionParameter>
    <acquisitionParameter name="IrisRingPosition">24.8108</acquisitionParameter>
    <acquisitionParameter name="FocusRingPosition">29.1397</acquisitionParameter>
    <acquisitionParameter name="ZoomRingPosition">99.8123</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Angle" unit="degree">360.0</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Time" unit="second">1001/60000</acquisitionParameter>
   &lt;/segment&gt;
   &lt;segment startTime="00:00:00.033" endTime="00:00:00.067"&gt;
    <acquisitionParameter name="IrisFNumber">2.821734</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromImagePlane">1.129</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromFrontLensVertex">0.979</acquisitionParameter>
    <acquisitionParameter name="LensZoom35mmStillCameraEquivalent">437.000</acquisitionParameter>
    <acquisitionParameter name="LensZoomActualFocalLength">80.800</acquisitionParameter>
    <acquisitionParameter name="IrisTNumber">2.892319</acquisitionParameter>
    <acquisitionParameter name="IrisRingPosition">24.9146</acquisitionParameter>
    <acquisitionParameter name="FocusRingPosition">29.1397</acquisitionParameter>
    <acquisitionParameter name="ZoomRingPosition">99.8123</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Angle" unit="degree">360.0</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Time" unit="second">1001/60000</acquisitionParameter>
   &lt;/segment&gt;
   &lt;segment startTime="00:00:00.067" endTime="00:00:00.100"&gt;
    <acquisitionParameter name="IrisFNumber">2.829863</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromImagePlane">1.129</acquisitionParameter>
    <acquisitionParameter name="FocusPositionFromFrontLensVertex">0.979</acquisitionParameter>
    <acquisitionParameter name="LensZoom35mmStillCameraEquivalent">437.000</acquisitionParameter>
    <acquisitionParameter name="LensZoomActualFocalLength">80.800</acquisitionParameter>
    <acquisitionParameter name="IrisTNumber">2.900651</acquisitionParameter>
    <acquisitionParameter name="IrisRingPosition">25.0183</acquisitionParameter>
    <acquisitionParameter name="FocusRingPosition">29.1397</acquisitionParameter>
    <acquisitionParameter name="ZoomRingPosition">99.8123</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Angle" unit="degree">360.0</acquisitionParameter>
    <acquisitionParameter name="ShutterSpeed_Time" unit="second">1001/60000</acquisitionParameter>
   &lt;/segment&gt;
    [...]
  </acquisitionDataFormat>
  <ebucore:duration>
   <ebucore:normalPlayTime>PT10.744S</ebucore:normalPlayTime>
  </ebucore:duration>
  <ebucore:fileSize>49856048</ebucore:fileSize>
```

```
    <ebucore:fileName>RDD18_fullitem_sample.MXF</ebucore:fileName>
    <ebucore:locator>E:\20161009_Download\RDD18_fullitem_sample\RDD18_fullitem_sample.MXF</ebucore:locator>
    <ebucore:technicalAttributeInteger typeLabel="OverallBitRate" unit="bps">37122896</ebucore:technicalAttributeInteger>
    <ebucore:dateCreated startDate="2016-04-19" startTime="07:33:33.000"/>
  </ebucore:format>
 </ebucore:coreMetadata>
</ebucore:ebuCoreMain>
```


Segment then parameter


```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Generated by MediaInfoLib - v0.7.91 -->
<ebucore:ebuCoreMain xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:ebucore="urn:ebu:metadata-schema:ebuCore_2015"
  xmlns:xalan="http://xml.apache.org/xalan" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ebu:metadata-schema:ebuCore_2015
  https://www.ebu.ch/metadata/schemas/EBUCore/20150522/ebucore_20150522.xsd"
  version="1.6" dateLastModified="2016-12-15" timeLastModified="09:35:20Z">
 <ebucore:coreMetadata>
  <ebucore:format>
   <ebucore:videoFormat videoFormatName="AVC">
    <ebucore:width unit="pixel">1920</ebucore:width>
    <ebucore:height unit="pixel">1080</ebucore:height>
    <ebucore:frameRate factorNumerator="30000" factorDenominator="1001">30</ebucore:frameRate>
    <ebucore:aspectRatio typeLabel="display">
     <ebucore:factorNumerator>16</ebucore:factorNumerator>
     <ebucore:factorDenominator>9</ebucore:factorDenominator>
    </ebucore:aspectRatio>
    <ebucore:videoEncoding typeLabel="High 4:2:2@L4.1"/>
    <ebucore:codec>
     <ebucore:codecIdentifier>
      <dc:identifier>0D01030102106001-0401020201316001</dc:identifier>
     </ebucore:codecIdentifier>
    </ebucore:codec>
    <ebucore:bitRate>27896896</ebucore:bitRate>
    <ebucore:bitRateMax>24999936</ebucore:bitRateMax>
    <ebucore:bitRateMode>variable</ebucore:bitRateMode>
    <ebucore:scanningFormat>progressive</ebucore:scanningFormat>
    <ebucore:videoTrack trackId="2"/>
    <ebucore:technicalAttributeString typeLabel="Standard">Component</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="ColorSpace">YUV</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="ChromaSubsampling">4:2:2</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="colour_primaries">BT.709</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="transfer_characteristics">BT.709</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="matrix_coefficients">BT.709</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeString typeLabel="colour_range">Limited</ebucore:technicalAttributeString>
    <ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">37465801</ebucore:technicalAttributeInteger>
    <ebucore:technicalAttributeInteger typeLabel="BitDepth" unit="bit">10</ebucore:technicalAttributeInteger>
    <ebucore:technicalAttributeBoolean typeLabel="CABAC">true</ebucore:technicalAttributeBoolean>
    <ebucore:technicalAttributeBoolean typeLabel="MBAFF">false</ebucore:technicalAttributeBoolean>
   </ebucore:videoFormat>
   <ebucore:audioFormat audioFormatName="PCM">
    <ebucore:audioEncoding typeLabel="PCM" typeLink="http://www.ebu.ch/metadata/cs/ebu_AudioCompressionCodeCS.xml#11"/>
    <ebucore:codec>
     <ebucore:codecIdentifier>
      <dc:identifier>0D01030102060300-0402020101000000</dc:identifier>
     </ebucore:codecIdentifier>
    </ebucore:codec>
    <ebucore:samplingRate>48000</ebucore:samplingRate>
```

<ebucore:sampleSize>24</ebucore:sampleSize>
<ebucore:bitRate>1152000</ebucore:bitRate>
<ebucore:bitRateMode>constant</ebucore:bitRateMode>
<ebucore:audioTrack trackId="3"/>
<ebucore:channels>1</ebucore:channels>
<ebucore:technicalAttributeString typeLabel="Endianness">Little</ebucore:technicalAttributeString>
<ebucore:technicalAttributeString typeLabel="Wrapping">Frame (AES)</ebucore:technicalAttributeString>
<ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">1547136</ebucore:technicalAttributeInteger>
</ebucore:audioFormat>
<ebucore:audioFormat audioFormatName="PCM">
[...]
</ebucore:audioFormat>
<ebucore:containerFormat containerFormatName="MXF">
<ebucore:containerEncoding formatLabel="MXF"/>
<ebucore:technicalAttributeString typeLabel="FormatProfile">OP-1a</ebucore:technicalAttributeString>
<ebucore:technicalAttributeString typeLabel="FormatSettings">Closed / Complete</ebucore:technicalAttributeString>
<ebucore:technicalAttributeString typeLabel="WritingLibrary">Sony MXF Development Kit (Win32) 4.10.0.119.1</ebucore:technicalAttributeString>
</ebucore:containerFormat>
<ebucore:timecodeFormat timecodeFormatName="MXF TC">
<ebucore:timecodeStart>
<ebucore:timecode>01:44:11;24</ebucore:timecode>
</ebucore:timecodeStart>
<ebucore:timecodeTrack trackId="1" typeLabel="Material"/>
<ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
</ebucore:timecodeFormat>
<ebucore:timecodeFormat timecodeFormatName="MXF TC">
<ebucore:timecodeStart>
<ebucore:timecode>01:44:11;24</ebucore:timecode>
</ebucore:timecodeStart>
<ebucore:timecodeTrack trackId="1" typeLabel="Source"/>
<ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
</ebucore:timecodeFormat>
<ebucore:timecodeFormat timecodeFormatName="SMPTE TC">
<ebucore:timecodeStart>
<ebucore:timecode>01:44:11;24</ebucore:timecode>
</ebucore:timecodeStart>
</ebucore:timecodeFormat>
<acquisitionDataFormat>
<duration>00:00:10.744</duration>
<segment startTime="00:00:00.000" endTime="00:00:10.744">
<acquisitionParameter name="CaptureGammaEquation">BT.709</acquisitionParameter>
</segment>
<segment startTime="00:00:00.000" endTime="00:00:00.067">
<acquisitionParameter name="IrisFNumber" interval="0.033">2.813628 2.821734</acquisitionParameter>
</segment>
<segment startTime="00:00:00.067" endTime="00:00:00.133">
<acquisitionParameter name="IrisFNumber">2.829863</acquisitionParameter>
</segment>
<segment startTime="00:00:00.133" endTime="00:00:00.200">
<acquisitionParameter name="IrisFNumber" interval="0.033">2.838016 2.829863</acquisitionParameter>
</segment>
<segment startTime="00:00:00.200" endTime="00:00:00.334">
<acquisitionParameter name="IrisFNumber">2.838016</acquisitionParameter>
</segment>
<segment startTime="00:00:00.334" endTime="00:00:00.534">
<acquisitionParameter name="IrisFNumber">2.854392</acquisitionParameter>

</segment>
<segment startTime="00:00:00.534" endTime="00:00:00.801">
<acquisitionParameter name="IrisFNumber" interval="0.067">2.862858 2.871106 2.887673 2.904335</acquisitionParameter>
</segment>
<segment startTime="00:00:00.801" endTime="00:00:00.834">
<acquisitionParameter name="IrisFNumber">2.912949</acquisitionParameter>
</segment>
<segment startTime="00:00:00.834" endTime="00:00:00.934">
<acquisitionParameter name="IrisFNumber">2.921341</acquisitionParameter>
</segment>
<segment startTime="00:00:00.934" endTime="00:00:01.001">
<acquisitionParameter name="IrisFNumber" interval="0.033">2.929757 2.938198</acquisitionParameter>
</segment>
<segment startTime="00:00:01.001" endTime="00:00:01.134">
<acquisitionParameter name="IrisFNumber" interval="0.067">2.955152 2.963666</acquisitionParameter>
</segment>
<segment startTime="00:00:01.134" endTime="00:00:01.235">
<acquisitionParameter name="IrisFNumber" interval="0.033">2.972456 2.981019 2.989607</acquisitionParameter>
</segment>
<segment startTime="00:00:01.235" endTime="00:00:01.301">
<acquisitionParameter name="IrisFNumber">3.006858</acquisitionParameter>
</segment>
<segment startTime="00:00:01.301" endTime="00:00:01.435">
<acquisitionParameter name="IrisFNumber" interval="0.033">3.015521 3.024209 3.041916 3.059469</acquisitionParameter>
</segment>
<segment startTime="00:00:01.435" endTime="00:00:01.635">
<acquisitionParameter name="IrisFNumber" interval="0.067">3.068283 3.086249 3.113000</acquisitionParameter>
</segment>
<segment startTime="00:00:01.635" endTime="00:00:01.969">
<acquisitionParameter name="IrisFNumber" interval="0.033">3.121969 3.130963 3.139983 3.167468 3.185745 3.167468 3.185745 3.194923 3.222889 3.232174</acquisitionParameter>
</segment>
<segment startTime="00:00:01.969" endTime="00:00:02.035">
<acquisitionParameter name="IrisFNumber">3.260190</acquisitionParameter>
</segment>
<segment startTime="00:00:02.035" endTime="00:00:02.402">
<acquisitionParameter name="IrisFNumber" interval="0.033">3.279280 3.298202 3.317233 3.336657 3.355910 3.365578 3.384999 3.404819 3.414628 3.424466 3.434331</acquisitionParameter>
</segment>
<segment startTime="00:00:02.402" endTime="00:00:02.469">
<acquisitionParameter name="IrisFNumber">3.474374</acquisitionParameter>
</segment>
<segment startTime="00:00:02.469" endTime="00:00:06.473">
<acquisitionParameter name="IrisFNumber" interval="0.033">3.494422 3.535164 3.545349 3.576080 3.607382 3.628197 3.702315 3.712981 3.777947 3.788831 3.844050 3.866231 3.922577 3.933878 4.025805 4.049378 4.108045 4.132100 4.179924 4.240842 4.290287 4.365354 4.415878 4.454531 4.506086 4.584929 4.638387 4.692070 4.733141 4.787921 4.871695 4.914338 4.971215 5.029177 5.072769 5.161527 5.176397 5.266968 5.297808 5.390047 5.452892 5.532362 5.564285 5.661643 5.694312 5.810637 5.878385 5.929338 5.964057 6.103425 6.138644 6.246051 6.282092 6.392009 6.429437 6.560779 6.598637 6.714092 6.733435 6.851250 6.931131 7.051808 7.093099 7.217206 7.300736 7.407137 7.471341 7.580228 7.646579 7.735079 7.802786 7.939310 7.985122 8.124837 8.171719 8.362677 8.386770 8.558097 8.583479 8.733663 8.834745 8.963502 8.989326 9.172962 9.225892 9.387317 9.442283 9.635173 9.690770 9.860329 9.945797 10.148972 10.178211 10.326548 10.387013 10.567860 10.660362 10.878135 10.909475 11.068469 11.262133 11.294579 11.492199 11.625208 11.794633 11.932151 12.105025 12.246162 12.353355 12.532332 12.642030 12.714978 12.863227 12.974723 13.126000 13.201741 13.239774 13.316171 13.355665 13.471430</acquisitionParameter>
</segment>
<segment startTime="00:00:06.473" endTime="00:00:06.540">
<acquisitionParameter name="IrisFNumber">13.549163</acquisitionParameter>
</segment>

<segment startTime="00:00:06.540" endTime="00:00:06.874">
<acquisitionParameter name="IrisFNumber" interval="0.033">13.628498 13.707138 13.786232 13.825949 13.906905 13.987151 14.027448 14.067861 14.150233 14.190999</acquisitionParameter>
</segment>
<segment startTime="00:00:06.874" endTime="00:00:06.940">
<acquisitionParameter name="IrisFNumber">14.314005</acquisitionParameter>
</segment>
<segment startTime="00:00:06.940" endTime="00:00:07.207">
<acquisitionParameter name="IrisFNumber" interval="0.033">14.355243 14.396600 14.439298 14.522616 14.564456 14.606416 14.648496 14.691941</acquisitionParameter>
</segment>
<segment startTime="00:00:07.207" endTime="00:00:07.274">
<acquisitionParameter name="IrisFNumber">14.776717</acquisitionParameter>
</segment>
<segment startTime="00:00:07.274" endTime="00:00:07.474">
<acquisitionParameter name="IrisFNumber" interval="0.033">14.861983 14.904800 14.947740 15.035264 15.078580 15.122021</acquisitionParameter>
</segment>
<segment startTime="00:00:07.474" endTime="00:00:07.541">
<acquisitionParameter name="IrisFNumber">15.165588</acquisitionParameter>
</segment>
<segment startTime="00:00:07.541" endTime="00:00:07.574">
<acquisitionParameter name="IrisFNumber">15.209279</acquisitionParameter>
</segment>
<segment startTime="00:00:07.574" endTime="00:00:07.674">
<acquisitionParameter name="IrisFNumber">15.298335</acquisitionParameter>
</segment>
<segment startTime="00:00:07.674" endTime="00:00:07.741">
<acquisitionParameter name="IrisFNumber">15.342409</acquisitionParameter>
</segment>
<segment startTime="00:00:07.741" endTime="00:00:07.875">
<acquisitionParameter name="IrisFNumber" interval="0.033">15.386610 15.430939 15.475395 15.519979</acquisitionParameter>
</segment>
<segment startTime="00:00:07.875" endTime="00:00:08.008">
<acquisitionParameter name="IrisFNumber" interval="0.067">15.566009 15.610854</acquisitionParameter>
</segment>
<segment startTime="00:00:08.008" endTime="00:00:08.041">
<acquisitionParameter name="IrisFNumber">15.655828</acquisitionParameter>
</segment>
<segment startTime="00:00:08.041" endTime="00:00:08.108">
<acquisitionParameter name="IrisFNumber">15.700932</acquisitionParameter>
</segment>
<segment startTime="00:00:08.108" endTime="00:00:08.141">
<acquisitionParameter name="IrisFNumber">15.746166</acquisitionParameter>
</segment>
<segment startTime="00:00:08.141" endTime="00:00:08.408">
<acquisitionParameter name="IrisFNumber">15.837026</acquisitionParameter>
</segment>
<segment startTime="00:00:08.408" endTime="00:00:08.542">
<acquisitionParameter name="IrisFNumber">15.883996</acquisitionParameter>
</segment>
<segment startTime="00:00:08.542" endTime="00:00:08.642">
<acquisitionParameter name="IrisFNumber">15.929757</acquisitionParameter>
</segment>
<segment startTime="00:00:08.642" endTime="00:00:08.709">
<acquisitionParameter name="IrisFNumber" interval="0.033">15.975650 15.929757</acquisitionParameter>
</segment>

<segment startTime="00:00:08.709" endTime="00:00:10.744">
  <acquisitionParameter name="IrisFNumber">15.975650</acquisitionParameter>
</segment>
<segment startTime="00:00:00.000" endTime="00:00:00.100">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.129</acquisitionParameter>
</segment>
<segment startTime="00:00:00.100" endTime="00:00:00.234">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.133</acquisitionParameter>
</segment>
<segment startTime="00:00:00.234" endTime="00:00:00.534">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.137</acquisitionParameter>
</segment>
<segment startTime="00:00:00.534" endTime="00:00:00.601">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.141</acquisitionParameter>
</segment>
<segment startTime="00:00:00.601" endTime="00:00:00.701">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">1.144 1.148 1.152</acquisitionParameter>
</segment>
<segment startTime="00:00:00.701" endTime="00:00:00.767">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.156</acquisitionParameter>
</segment>
<segment startTime="00:00:00.767" endTime="00:00:00.834">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">1.152 1.148</acquisitionParameter>
</segment>
<segment startTime="00:00:00.834" endTime="00:00:00.901">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.156</acquisitionParameter>
</segment>
<segment startTime="00:00:00.901" endTime="00:00:01.001">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.164</acquisitionParameter>
</segment>
<segment startTime="00:00:01.001" endTime="00:00:01.201">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.067">1.168 1.171 1.175</acquisitionParameter>
</segment>
<segment startTime="00:00:01.201" endTime="00:00:01.268">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">1.179 1.183</acquisitionParameter>
</segment>
<segment startTime="00:00:01.268" endTime="00:00:01.335">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.187</acquisitionParameter>
</segment>
<segment startTime="00:00:01.335" endTime="00:00:01.435">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">1.191 1.196 1.200</acquisitionParameter>
</segment>
<segment startTime="00:00:01.435" endTime="00:00:01.535">
  <acquisitionParameter name="FocusPositionFromImagePlane">1.204</acquisitionParameter>
</segment>
<segment startTime="00:00:01.535" endTime="00:00:01.802">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.067">1.216 1.220 1.225 1.233</acquisitionParameter>
</segment>
<segment startTime="00:00:01.802" endTime="00:00:02.035">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">1.237 1.242 1.251 1.255 1.259 1.264 1.268</acquisitionParameter>
</segment>
<segment startTime="00:00:02.035" endTime="00:00:02.236">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.067">1.277 1.286 1.296</acquisitionParameter>
</segment>
<segment startTime="00:00:02.236" endTime="00:00:02.302">
  <acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">1.305 1.310</acquisitionParameter>

</segment>
<segment startTime="00:00:02.302" endTime="00:00:02.369">
<acquisitionParameter name="FocusPositionFromImagePlane">1.315</acquisitionParameter>
</segment>
<segment startTime="00:00:02.369" endTime="00:00:03.737">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">1.324 1.334 1.339 1.344 1.354 1.369 1.379 1.395 1.411 1.416 1.444 1.449 1.472 1.483 1.501 1.507 1.531 1.550 1.569 1.582 1.602 1.615 1.636 1.664 1.678 1.700 1.714 1.752 1.760 1.792 1.800 1.841 1.850 1.876 1.893 1.929 1.938 1.948 1.985 2.014 2.034</acquisitionParameter>
</segment>
<segment startTime="00:00:03.737" endTime="00:00:03.804">
<acquisitionParameter name="FocusPositionFromImagePlane">2.085</acquisitionParameter>
</segment>
<segment startTime="00:00:03.804" endTime="00:00:03.937">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">2.139 2.150 2.195 2.206</acquisitionParameter>
</segment>
<segment startTime="00:00:03.937" endTime="00:00:04.004">
<acquisitionParameter name="FocusPositionFromImagePlane">2.265</acquisitionParameter>
</segment>
<segment startTime="00:00:04.004" endTime="00:00:04.104">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">2.314 2.366 2.419</acquisitionParameter>
</segment>
<segment startTime="00:00:04.104" endTime="00:00:04.171">
<acquisitionParameter name="FocusPositionFromImagePlane">2.446</acquisitionParameter>
</segment>
<segment startTime="00:00:04.171" endTime="00:00:04.638">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">2.532 2.592 2.561 2.687 2.703 2.720 2.806 2.878 2.859 2.953 3.053 3.013 3.137 3.181</acquisitionParameter>
</segment>
<segment startTime="00:00:04.638" endTime="00:00:04.705">
<acquisitionParameter name="FocusPositionFromImagePlane">3.319</acquisitionParameter>
</segment>
<segment startTime="00:00:04.705" endTime="00:00:06.840">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">3.367 3.547 3.575 3.630 3.521 3.837 4.100 3.999 4.400 4.320 4.560 4.700 4.980 4.840 4.980 5.250 5.030 4.740 5.090 6.230 5.730 5.600 6.470 6.550 6.310 5.480 6.550 7.980 6.150 8.940 8.370 6.730 8.370 6.550 12.260 13.840 9.770 11.990 8.500 18.590 22.390 70.800 62.900 24.370 142.200 16.860 113.600 29.580 900.000 9.260 23.340 31.250 7.520 15.880 1300.000 29.580 40.300 17.980 8.940 1100.000 1900.000 70.800 500.000 20.700</acquisitionParameter>
</segment>
<segment startTime="00:00:06.840" endTime="00:00:06.907">
<acquisitionParameter name="FocusPositionFromImagePlane">500.000</acquisitionParameter>
</segment>
<segment startTime="00:00:06.907" endTime="00:00:07.040">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">2500.000 6.550 40.300 56.500</acquisitionParameter>
</segment>
<segment startTime="00:00:07.040" endTime="00:00:07.241">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.067">2500.000 1100.000 19.240</acquisitionParameter>
</segment>
<segment startTime="00:00:07.241" endTime="00:00:07.307">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">2500.000 900.000</acquisitionParameter>
</segment>
<segment startTime="00:00:07.307" endTime="00:00:07.374">
<acquisitionParameter name="FocusPositionFromImagePlane">2500.000</acquisitionParameter>
</segment>
<segment startTime="00:00:07.374" endTime="00:00:07.474">
<acquisitionParameter name="FocusPositionFromImagePlane" interval="0.033">11.990 2500.000 900.000</acquisitionParameter>
</segment>
[...]
</acquisitionDataFormat>

```xml
      <ebucore:duration>
        <ebucore:normalPlayTime>PT10.744S</ebucore:normalPlayTime>
      </ebucore:duration>
      <ebucore:fileSize>49856048</ebucore:fileSize>
      <ebucore:fileName>RDD18_fullitem_sample.MXF</ebucore:fileName>
      <ebucore:locator>E:\20161009_Download\RDD18_fullitem_sample\RDD18_fullitem_sample.MXF</ebucore:locator>
      <ebucore:technicalAttributeInteger typeLabel="OverallBitRate" unit="bps">37122896</ebucore:technicalAttributeInteger>
      <ebucore:dateCreated startDate="2016-04-19" startTime="07:33:33.000"/>
    </ebucore:format>
  </ebucore:coreMetadata>
</ebucore:ebuCoreMain>
```


Parameter then segment


```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- Generated by MediaInfoLib - v0.7.91 -->
<ebucore:ebuCoreMain xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:ebucore="urn:ebu:metadata-schema:ebuCore_2015"
  xmlns:xalan="http://xml.apache.org/xalan" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:ebu:metadata-schema:ebuCore_2015
  https://www.ebu.ch/metadata/schemas/EBUCore/20150522/ebucore_20150522.xsd"
  version="1.6" dateLastModified="2016-12-15" timeLastModified="09:37:15Z">
  <ebucore:coreMetadata>
    <ebucore:format>
      <ebucore:videoFormat videoFormatName="AVC">
        <ebucore:width unit="pixel">1920</ebucore:width>
        <ebucore:height unit="pixel">1080</ebucore:height>
        <ebucore:frameRate factorNumerator="30000" factorDenominator="1001">30</ebucore:frameRate>
        <ebucore:aspectRatio typeLabel="display">
          <ebucore:factorNumerator>16</ebucore:factorNumerator>
          <ebucore:factorDenominator>9</ebucore:factorDenominator>
        </ebucore:aspectRatio>
        <ebucore:videoEncoding typeLabel="High 4:2:2@L4.1"/>
        <ebucore:codec>
          <ebucore:codecIdentifier>
            <dc:identifier>0D01030102106001-0401020201316001</dc:identifier>
          </ebucore:codecIdentifier>
        </ebucore:codec>
        <ebucore:bitRate>27896896</ebucore:bitRate>
        <ebucore:bitRateMax>24999936</ebucore:bitRateMax>
        <ebucore:bitRateMode>variable</ebucore:bitRateMode>
        <ebucore:scanningFormat>progressive</ebucore:scanningFormat>
        <ebucore:videoTrack trackId="2"/>
        <ebucore:technicalAttributeString typeLabel="Standard">Component</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="ColorSpace">YUV</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="ChromaSubsampling">4:2:2</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="colour_primaries">BT.709</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="transfer_characteristics">BT.709</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="matrix_coefficients">BT.709</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeString typeLabel="colour_range">Limited</ebucore:technicalAttributeString>
        <ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">37465801</ebucore:technicalAttributeInteger>
        <ebucore:technicalAttributeInteger typeLabel="BitDepth" unit="bit">10</ebucore:technicalAttributeInteger>
        <ebucore:technicalAttributeBoolean typeLabel="CABAC">true</ebucore:technicalAttributeBoolean>
        <ebucore:technicalAttributeBoolean typeLabel="MBAFF">false</ebucore:technicalAttributeBoolean>
      </ebucore:videoFormat>
      <ebucore:audioFormat audioFormatName="PCM">
        <ebucore:audioEncoding typeLabel="PCM" typeLink="http://www.ebu.ch/metadata/cs/ebu_AudioCompressionCodeCS.xml#11"/>
        <ebucore:codec>
          <ebucore:codecIdentifier>
```

<dc:identifier>0D01030102060300-0402020101000000</dc:identifier>
</ebucore:codecIdentifier>
</ebucore:codec>
<ebucore:samplingRate>48000</ebucore:samplingRate>
<ebucore:sampleSize>24</ebucore:sampleSize>
<ebucore:bitRate>1152000</ebucore:bitRate>
<ebucore:bitRateMode>constant</ebucore:bitRateMode>
<ebucore:audioTrack trackId="3"/>
<ebucore:channels>1</ebucore:channels>
<ebucore:technicalAttributeString typeLabel="Endianness">Little</ebucore:technicalAttributeString>
<ebucore:technicalAttributeString typeLabel="Wrapping">Frame (AES)</ebucore:technicalAttributeString>
<ebucore:technicalAttributeInteger typeLabel="StreamSize" unit="byte">1547136</ebucore:technicalAttributeInteger>
</ebucore:audioFormat>
<ebucore:audioFormat audioFormatName="PCM">
[...]
</ebucore:audioFormat>
<ebucore:containerFormat containerFormatName="MXF">
<ebucore:containerEncoding formatLabel="MXF"/>
<ebucore:technicalAttributeString typeLabel="FormatProfile">OP-1a</ebucore:technicalAttributeString>
<ebucore:technicalAttributeString typeLabel="FormatSettings">Closed / Complete</ebucore:technicalAttributeString>
<ebucore:technicalAttributeString typeLabel="WritingLibrary">Sony MXF Development Kit (Win32) 4.10.0.119.1</ebucore:technicalAttributeString>
</ebucore:containerFormat>
<ebucore:timecodeFormat timecodeFormatName="MXF TC">
<ebucore:timecodeStart>
<ebucore:timecode>01:44:11;24</ebucore:timecode>
</ebucore:timecodeStart>
<ebucore:timecodeTrack trackId="1" typeLabel="Material"/>
<ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
</ebucore:timecodeFormat>
<ebucore:timecodeFormat timecodeFormatName="MXF TC">
<ebucore:timecodeStart>
<ebucore:timecode>01:44:11;24</ebucore:timecode>
</ebucore:timecodeStart>
<ebucore:timecodeTrack trackId="1" typeLabel="Source"/>
<ebucore:technicalAttributeBoolean typeLabel="Stripped">true</ebucore:technicalAttributeBoolean>
</ebucore:timecodeFormat>
<ebucore:timecodeFormat timecodeFormatName="SMPTE TC">
<ebucore:timecodeStart>
<ebucore:timecode>01:44:11;24</ebucore:timecode>
</ebucore:timecodeStart>
</ebucore:timecodeFormat>
<acquisitionDataFormat>
<duration>00:00:10.744</duration>
<acquisitionParameter name="CaptureGammaEquation">
<segment startTime="00:00:00.000" endTime="00:00:10.744">BT.709</segment>
</acquisitionParameter>
<acquisitionParameter name="IrisFNumber">
<segment startTime="00:00:00.000" endTime="00:00:00.067" interval="0.033">2.813628 2.821734</segment>
<segment startTime="00:00:00.067" endTime="00:00:00.133">2.829863</segment>
<segment startTime="00:00:00.133" endTime="00:00:00.200" interval="0.033">2.838016 2.829863</segment>
<segment startTime="00:00:00.200" endTime="00:00:00.334">2.838016</segment>
<segment startTime="00:00:00.334" endTime="00:00:00.534">2.854392</segment>
<segment startTime="00:00:00.534" endTime="00:00:00.801" interval="0.067">2.862858 2.871106 2.887673 2.904335</segment>
<segment startTime="00:00:00.801" endTime="00:00:00.834">2.912949</segment>
<segment startTime="00:00:00.834" endTime="00:00:00.934">2.921341</segment>
<segment startTime="00:00:00.934" endTime="00:00:01.001" interval="0.033">2.929757 2.938198</segment>

<segment startTime="00:00:01.001" endTime="00:00:01.134" interval="0.067">2.955152 2.963666</segment>
<segment startTime="00:00:01.134" endTime="00:00:01.235" interval="0.033">2.972456 2.981019 2.989607</segment>
<segment startTime="00:00:01.235" endTime="00:00:01.301">3.006858</segment>
<segment startTime="00:00:01.301" endTime="00:00:01.435" interval="0.033">3.015521 3.024209 3.041916 3.059469</segment>
<segment startTime="00:00:01.435" endTime="00:00:01.635" interval="0.067">3.068283 3.086249 3.113000</segment>
<segment startTime="00:00:01.635" endTime="00:00:01.969" interval="0.033">3.121969 3.130963 3.139983 3.167468 3.185745 3.167468 3.185745 3.194923 3.222889 3.232174</segment>
<segment startTime="00:00:01.969" endTime="00:00:02.035">3.260190</segment>
<segment startTime="00:00:02.035" endTime="00:00:02.402" interval="0.033">3.279280 3.298202 3.317233 3.336657 3.355910 3.365578 3.384999 3.404819 3.414628 3.424466 3.434331</segment>
<segment startTime="00:00:02.402" endTime="00:00:02.469">3.474374</segment>
<segment startTime="00:00:02.469" endTime="00:00:06.473" interval="0.033">3.494422 3.535164 3.545349 3.576080 3.607382 3.628197 3.702315 3.712981 3.777947 3.788831 3.844050 3.866231 3.922577 3.933878 4.025805 4.049378 4.108045 4.132100 4.179924 4.240842 4.290287 4.365354 4.415878 4.454531 4.506086 4.584929 4.638387 4.692070 4.733141 4.787921 4.871695 4.914338 4.971215 5.029177 5.072769 5.161527 5.176397 5.266968 5.297808 5.390047 5.452892 5.532362 5.564285 5.661643 5.694312 5.810637 5.878385 5.929338 5.964057 6.103425 6.138644 6.246051 6.282092 6.392009 6.429437 6.560779 6.598637 6.714092 6.733435 6.851250 6.931131 7.051808 7.093099 7.217206 7.300736 7.407137 7.471341 7.580228 7.646579 7.735079 7.802786 7.939310 7.985122 8.124837 8.171719 8.362677 8.386770 8.558097 8.583479 8.733663 8.834745 8.963502 8.989326 9.172962 9.225892 9.387317 9.442283 9.635173 9.690770 9.860329 9.945797 10.148972 10.178211 10.326548 10.387013 10.567860 10.660362 10.878135 10.909475 11.068469 11.262133 11.294579 11.492199 11.625208 11.794633 11.932151 12.105025 12.246162 12.353355 12.532332 12.642030 12.714978 12.863227 12.974723 13.126000 13.201741 13.239774 13.316171 13.355665 13.471430</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="FocusPositionFromImagePlane">
<segment startTime="00:00:00.000" endTime="00:00:00.100">1.129</segment>
<segment startTime="00:00:00.100" endTime="00:00:00.234">1.133</segment>
<segment startTime="00:00:00.234" endTime="00:00:00.534">1.137</segment>
<segment startTime="00:00:00.534" endTime="00:00:00.601">1.141</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="FocusPositionFromFrontLensVertex">
<segment startTime="00:00:00.000" endTime="00:00:00.100">0.979</segment>
<segment startTime="00:00:00.100" endTime="00:00:00.234">0.983</segment>
<segment startTime="00:00:00.234" endTime="00:00:00.534">0.987</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="MacroSetting">
<segment startTime="00:00:00.000" endTime="00:00:10.744">Off</segment>
</acquisitionParameter>
<acquisitionParameter name="LensZoom35mmStillCameraEquivalent">
<segment startTime="00:00:00.000" endTime="00:00:00.100">437.000</segment>
<segment startTime="00:00:00.100" endTime="00:00:00.267">435.000</segment>
<segment startTime="00:00:00.267" endTime="00:00:00.567" interval="0.100">433.000 431.000 429.000</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="LensZoomActualFocalLength">
<segment startTime="00:00:00.000" endTime="00:00:00.100">80.800</segment>
<segment startTime="00:00:00.100" endTime="00:00:00.267">80.400</segment>
<segment startTime="00:00:00.267" endTime="00:00:00.567" interval="0.100">80.100 79.700 79.300</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="OpticalExtenderMagnification" unit="percentage">
<segment startTime="00:00:00.000" endTime="00:00:10.744">100</segment>
</acquisitionParameter>
<acquisitionParameter name="LensAttributes">

<segment startTime="00:00:00.000" endTime="00:00:10.744">HoshinonHD17x5.5AFf1.4</segment>
</acquisitionParameter>
<acquisitionParameter name="IrisTNumber">
<segment startTime="00:00:00.000" endTime="00:00:00.067" interval="0.033">2.884010 2.892319</segment>
<segment startTime="00:00:00.067" endTime="00:00:00.133">2.900651</segment>
<segment startTime="00:00:00.133" endTime="00:00:00.200" interval="0.033">2.909008 2.900651</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="IrisRingPosition">
<segment startTime="00:00:00.000" endTime="00:00:00.067" interval="0.033">24.8108 24.9146</segment>
<segment startTime="00:00:00.067" endTime="00:00:00.133">25.0183</segment>
<segment startTime="00:00:00.133" endTime="00:00:00.200" interval="0.033">25.1221 25.0183</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="FocusRingPosition">
<segment startTime="00:00:00.000" endTime="00:00:00.100">29.1397</segment>
<segment startTime="00:00:00.100" endTime="00:00:00.234">29.3900</segment>
<segment startTime="00:00:00.234" endTime="00:00:00.534">29.6387</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="ZoomRingPosition">
<segment startTime="00:00:00.000" endTime="00:00:00.100">99.8123</segment>
<segment startTime="00:00:00.100" endTime="00:00:00.267">99.6384</segment>
<segment startTime="00:00:00.267" endTime="00:00:00.567" interval="0.100">99.4629 99.2874 99.1119</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="AutoExposureMode">
<segment startTime="00:00:00.000" endTime="00:00:10.744">FullAuto</segment>
</acquisitionParameter>
<acquisitionParameter name="AutoFocusSensingAreaSetting">
<segment startTime="00:00:00.000" endTime="00:00:10.744">Manual</segment>
</acquisitionParameter>
<acquisitionParameter name="ColorCorrectionFilterWheelSetting">
<segment startTime="00:00:00.000" endTime="00:00:10.744">ColorCompensation3200K</segment>
</acquisitionParameter>
<acquisitionParameter name="NeutralDensityFilterWheelSetting">
<segment startTime="00:00:00.000" endTime="00:00:10.744">1/8</segment>
</acquisitionParameter>
<acquisitionParameter name="ImageSensorDimensionEffectiveWidth">
<segment startTime="00:00:00.000" endTime="00:00:10.744">7.190</segment>
</acquisitionParameter>
<acquisitionParameter name="ImageSensorDimensionEffectiveHeight">
<segment startTime="00:00:00.000" endTime="00:00:10.744">4.045</segment>
</acquisitionParameter>
<acquisitionParameter name="CaptureFrameRate" unit="fps">
<segment startTime="00:00:00.000" endTime="00:00:10.744">29.970</segment>
</acquisitionParameter>
<acquisitionParameter name="ImageSensorReadoutMode">
<segment startTime="00:00:00.000" endTime="00:00:10.744">Interlacedfield</segment>
</acquisitionParameter>
<acquisitionParameter name="ShutterSpeed_Angle" unit="degree">
<segment startTime="00:00:00.000" endTime="00:00:00.133">360.0</segment>
<segment startTime="00:00:00.133" endTime="00:00:00.467">359.5</segment>
<segment startTime="00:00:00.467" endTime="00:00:00.567">359.0</segment>
[...]

</acquisitionParameter>
<acquisitionParameter name="ShutterSpeed_Time" unit="second">
<segment startTime="00:00:00.000" endTime="00:00:00.133">1001/60000</segment>
<segment startTime="00:00:00.133" endTime="00:00:00.467">1000/60000</segment>
<segment startTime="00:00:00.467" endTime="00:00:00.567">998/60000</segment>
[...]
</acquisitionParameter>
<acquisitionParameter name="CameraMasterGainAdjustment" unit="dB">
<segment startTime="00:00:00.000" endTime="00:00:10.744">0.00</segment>
</acquisitionParameter>
<acquisitionParameter name="ISOSensitivity">
<segment startTime="00:00:00.000" endTime="00:00:10.744">430</segment>
</acquisitionParameter>
<acquisitionParameter name="ElectricalExtenderMagnification" unit="percentage">
<segment startTime="00:00:00.000" endTime="00:00:10.744">100</segment>
</acquisitionParameter>
<acquisitionParameter name="AutoWhiteBalanceMode">
<segment startTime="00:00:00.000" endTime="00:00:10.744">Hold</segment>
</acquisitionParameter>
<acquisitionParameter name="WhiteBalance" unit="kelvin">
<segment startTime="00:00:00.000" endTime="00:00:10.744">5600</segment>
</acquisitionParameter>
<acquisitionParameter name="CameraMasterBlackLevel" unit="percentage">
<segment startTime="00:00:00.000" endTime="00:00:10.744">3.0</segment>
</acquisitionParameter>
<acquisitionParameter name="CameraKneePoint" unit="percentage">
<segment startTime="00:00:00.000" endTime="00:00:10.744">85.0</segment>
</acquisitionParameter>
<acquisitionParameter name="CameraKneeSlope">
<segment startTime="00:00:00.000" endTime="00:00:10.744">0.100</segment>
</acquisitionParameter>
<acquisitionParameter name="CameraLuminanceDynamicRange" unit="percentage">
<segment startTime="00:00:00.000" endTime="00:00:10.744">450.0</segment>
</acquisitionParameter>
<acquisitionParameter name="CameraSettingFileURI">
<segment startTime="00:00:00.000" endTime="00:00:10.744">file:///Camera/Setting001.dat</segment>
</acquisitionParameter>
<acquisitionParameter name="CameraAttributes">
<segment startTime="00:00:00.000" endTime="00:00:10.744">pseudo</segment>
</acquisitionParameter>
<acquisitionParameter name="ExposureIndexofPhotoMeter">
<segment startTime="00:00:00.000" endTime="00:00:10.744">430</segment>
</acquisitionParameter>
<acquisitionParameter name="GammaForCDL">
<segment startTime="00:00:00.000" endTime="00:00:10.744">SameasCaptureGamma</segment>
</acquisitionParameter>
<acquisitionParameter name="ASC_CDL_V12">
<segment startTime="00:00:00.000" endTime="00:00:10.744">sR=1.0sG=1.0sB=0.5oR=0.0oG=0.0oB=-262144.0pR=1.0pG=1.0pB=0.5sat=2.0</segment>
</acquisitionParameter>
<acquisitionParameter name="ColorMatrix">
<segment startTime="00:00:00.000" endTime="00:00:10.744">RR=1.000GR=0.000BR=0.000RG=0.000GG=1.000BG=0.000RB=0.000GB=0.000BB=1.000</segment>
</acquisitionParameter>
</acquisitionDataFormat>
<ebucore:duration>
<ebucore:normalPlayTime>PT10.744S</ebucore:normalPlayTime>

```
    </ebucore:duration>
    <ebucore:fileSize>49856048</ebucore:fileSize>
    <ebucore:fileName>RDD18_fullitem_sample.MXF</ebucore:fileName>
    <ebucore:locator>E:\20161009_Download\RDD18_fullitem_sample\RDD18_fullitem_sample.MXF</ebucore:locator>
    <ebucore:technicalAttributeInteger typeLabel="OverallBitRate" unit="bps">37122896</ebucore:technicalAttributeInteger>
    <ebucore:dateCreated startDate="2016-04-19" startTime="07:33:33.000"/>
  </ebucore:format>
 </ebucore:coreMetadata>
</ebucore:ebuCoreMain>
```

Appendix B

Ladio Model (V 0.5)