# 129th MPEG Brussels, Belgium, 13 - 17 January 2020, Meeting Report

**Panos Kudumakis**
**qMedia, Queen Mary University of London**

## Contents



## 1 Draft Call for Proposals for MPEG-I Immersive Audio

The MPEG-I suite of standards, "Coded Representation of Immersive Media," is intended to support emerging virtual and augmented reality applications. This Call for Proposals (CfP) is for technology to be standardized in Part 4, "Immersive Audio." Along with other parts, Part 3, "Immersive Video" and Part 2, "Systems Support," the suite of standards will support a Virtual Reality (VR) or an Augmented Reality (AR) presentation in which the user can navigate and interact with the environment using 6 degrees of freedom (6 DoF), that being spatial navigation (x, y, z) and user head orientation (yaw, pitch, roll).

The goal in MPEG-I presentations is to impart the feeling that the user is actually present in the virtual world. Audio in the world (or scene) is perceived as in the real world, with sounds coming from an associated visual figure. That is, perceived with the correct location and distance. Physical movement of the user in the real world is perceived as having matching movement in the virtual world. Furthermore, and importantly, the user can interact with the virtual scene and cause sounds that are perceived as realistic and matching the user's experience in the real world.

The architecture for Part 4, Immersive Audio is shown in Figure 1 (N18158) "MPEG-I Audio Architecture and Requirements". A key stipulation of the architecture is that the audio compression engine used in MPEG-I Immersive Audio is MPEG-H 3D Audio (ISO/IEC 23008-3), specifically LC Profile. This is shown in the tan shaded box of Figure 1. The other audio decoder, "Low-Delay Decoder", is out of scope for MPEG-I Immersive Audio, and is shown in the purple shaded box of Figure 1. The new technology that is to be standardized in MPEG-I Immersive Audio is shown in the green shaded boxes of Figure 1, and primarily consists of:

- Technology for rendering the audio presentation while permitting the user to have 6 DoF.
- Metadata to support this rendering.
- A bitstream syntax that enables storage and streaming of the MPEG-I Immersive Audio content. Since the coding of all audio signals is done using MPEG-H 3D Audio LC Profile, the MPEG-I bitstream must convey an MPEG-H 3D Audio LC Profile bitstream.

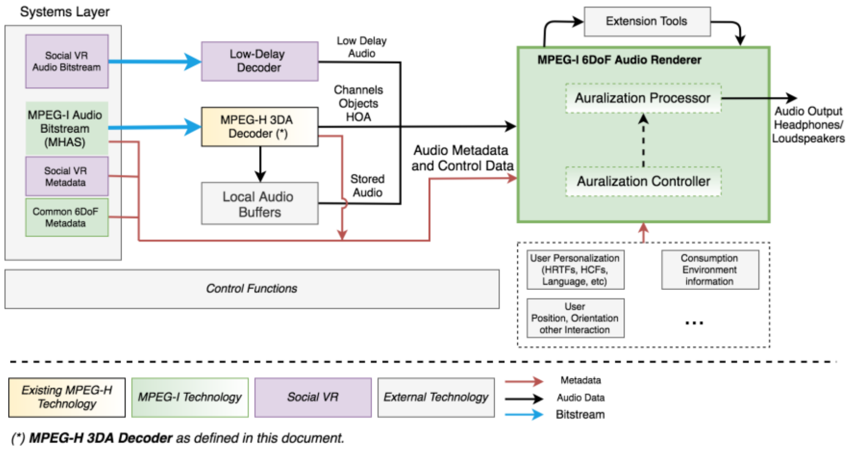



**Figure 1 - MPEG-I Audio Reference Architecture**

The framework for evaluating the CfP submitted technology is shown in Figure 2 (N18980). This figure shows that the audio media signals are encoded and decoded off-line and then made available to all proponents. This coding is done using MPEG-H 3D Audio LC Profile, and in this way audio signal compression is “out of scope” when responding to this CfP.

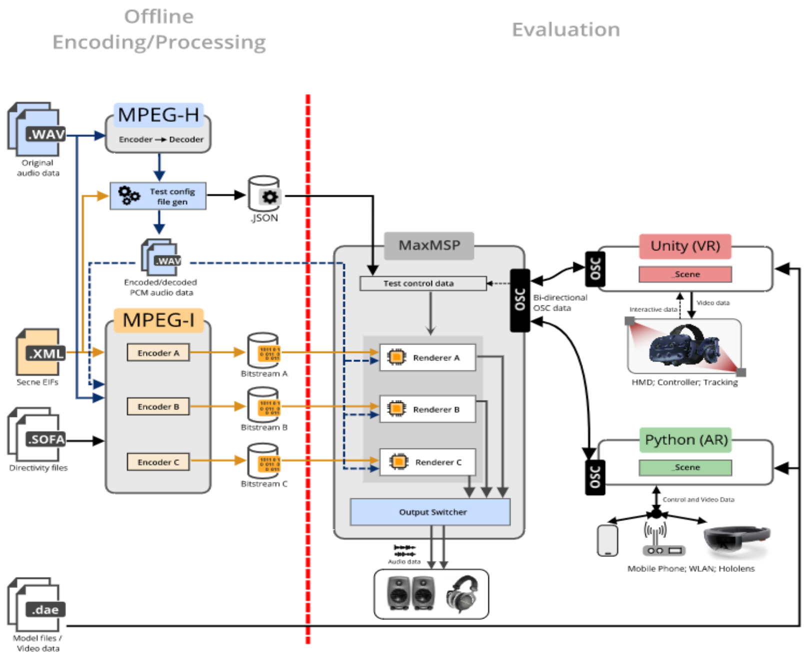



**Figure 2 - Overview of Audio Evaluation Platform and Process**

Furthermore, an important component of the evaluation of proponent technology is assessing subjective quality of the MPEG-I Immersive Audio presentation, which is done via subjective listening tests in which all proponent MPEG-I Immersive Audio rendering technology must run in real-time on the MPEG-I Audio Evaluation Platform (described in N18980).

Additional components of the evaluation of proponent technology will be done via proponent-supplied description of the functionality of the submitted technology and also via proponent objective measurements of aspects of the technology (e.g. motion-to-sound latency), which must be included in the submitted proponent documentation.

**Time Line Table**
Each entry in the table is described in a section below.

| Meeting | Date | Who | Action |
|---|---|---|---|
| 130 | Apr 2020 | WG11 | Issue Call for Proposals package |
| | 19 Jun 2020 | Proponent | Register |
| | Jun 2020 | Proponent, Test labs | Get Audio Evaluation Platform |
| 131 | Jun 2020 | Audio subgroup | Prepare Test Material |
| | 6 Jul 2020 | Proponent | Get test material. |
| | 24 Aug 2020 | Proponent | Submit real-time Max External, bitstream files, and other required documentation. (e.g. latency, loudness) |
| | 31 Aug 2020 | Test Lab | Conduct subjective tests. |
| | 25 Sep 2020 | Test Lab | Submit subjective test data |
| | | Test Administrator | Performs analysis of subjective test data |
| | | Proponent | Submit proponent documentation |
| 132 | Oct 2020 | Audio subgroup | Evaluate Call for Proposals submissions and select technology |
| 133 | Jan 2021 | Proponent | Submits Working Draft (WD) and Reference Model source code (RM) for selected technology. |

**Table 1 - Time Line for Call for Proposals**

**Fulfilment of Audio Requirements**
Proponent must submit a description of how their technology fulfils certain MPEG-I Immersive Audio requirements, indicated in Table 2, below, by the entry "Description" under the heading "Methodology." The table lists all requirements found in N18158 by their number and gives a short description of the requirement. The additional columns indicate an assessment methodology for the requirement and any relevant comments concerning the methodology.

The Methodologies are:

| | |
|---|---|
| Subjective | Assess via subjective test, described in Section **Error! Reference source not found.** of N18982. |
| Objective | Proponent-conducted objective test, described in Section 3.8 of N18982. |
| Description | Proponent description of how requirement is satisfied, as described below. |

For requirements whose assessment methodology is *Description*, the proponent submitted documentation must contain a description of how the proposed technology satisfies the requirement. This description must be sufficient for Audio subgroup experts to check on their own that the requirement is met.

| Requirement Number (from N18158) | Requirement | Methodology | Description |
|---|---|---|---|
| 1 | Perceived experience | Subjective | Performance is assed via listening tests. See Section **Error! Reference source not found.** of N18982. |
| 2 | Efficient representation | Objective | State metadata bit-rate (see Section **Error! Reference source not found.** of N18982). |
| 2.1 | MPEG-H 3D Audio coding | Description | Does the proponent MPEG-I bitstream contain a conformant MPEG-H 3D Audio LC profile bitstream?<br>Does proponent technology support any combination of channels, objects and HOA signals? |
| 2.2 | DRC | Description | Proponent describes how this is done. |
| 2.3 | MPEG-I metadata | Description | Proponent describes metadata. |
| 2.4 | Multiple streams | Description | Proponent describes how this is done. |
| 3 | Consistent experience | Subjective | Performance is assed via subjective tests with HMD with headphone listening. |
| 3.1 | Consistent visual and audio | Subjective | Performance is assed via subjective tests with HMD with headphone listening. |
| 3.2 | Non-diegetic audio elements | Description | Proponent describes how this is done. |
| 3.3 | Earcons | Description | Proponent describes how this is done. |
| 4, 4.1 | Division into sub-scenes | Description | Proponent describes how this is done.<br>Describe sub-setting process and algorithm (if any).<br>Describe any associated interfaces and metadata. |
| 5 | Metadata describing scene | Description | Describe MPEG-I metadata for scene. |
| 6, 7 | Controlling or restricting audio scene or parameters | Description | Describe MPEG-I metadata or interfaces to support restrictions and how the restrictions are performed. |
| 8 | Random access | Objective and Description | Describe the bitstream structure needed to support random access, and the overhead required for a range of time and space random access intervals. |
| 9, 10 | Rendering for jumps, accelerated movement and zoom | Description | Describe the metadata that supports this functionality.<br>Describe how this is done and what the user would perceive. |
| 11-13, 15 | Rendering requirements | Subjective | Describe the metadata that supports this functionality.<br>Performance is assed via HMD and headphone listening tests that stress the requirements of the renderer, including the perceptual effects listed in **Error! Reference source not found.** of N18982. |
| 14 | Locally captured audio | Description | Describe how local voice is captured and how side tone is rendered. |
| 16 - 18 | Interfaces | Description | Describe specified interfaces and how they support this functionality. |
| 18, 19 | Personalization of HRTF, HP EQ | Subjective<br><br>Description | Testing with personalized HRTF and headphone compensation filter.<br>Describe the specified HRTF and HP EQ interfaces and how they would support the desired personalization. |
| 20 | 6DoF with headphones | Subjective | Performance is assed via subjective tests with HMD and headphone listening. See Section **Error! Reference source not found.** of N18982. |
| 21 | 6DoF with loudspeakers | Description | Describe algorithms used with head tracker and loudspeaker listening. |
| 22 | Static user with joystick | Description | Describe specified interfaces and how they support this functionality. |
| 23 | 6DoF with both headphones and loudspeakers | Description | Describe how technology fulfilling requirements 20 and 21 can be combined into a single presentation |
| 24, 24.4 | Social VR | Description | Describe specified interfaces for remote user media (speech/audio) and metadata (6DoF position and orientation) and how this is supported in proponent architecture and technology. |
| 24.1, 24.2 | Social VR | Objective and Description | Measure algorithmic latency subject to stated assumptions |
| 24.3 | Synchronization of A and V | Description | Describe how proponent metadata and technology would assure synchronization. |
| 25-27 | Interoperability | Description | Describe how this would occur, with reference to envisioned bitstream syntax or decoder functionality.<br>Describe the envisioned user experience. |

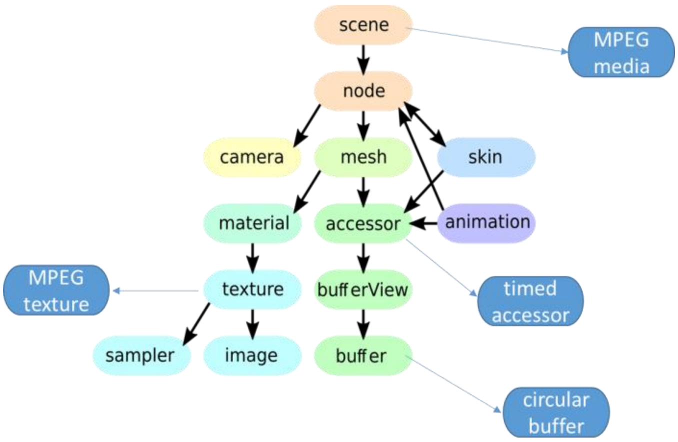
**Table 2 – Template for Reporting Fulfilment of Requirements**

**Output Documents**
**N18978 - Workplan on MPEG-I Audio**
**N18979 - MPEG-I 6DoF Audio Encoder Input Format**
**N18980 - Documentation for the MPEG-I Audio Evaluation Platform**
**N18981 - MPEG-I Audio Subjective Testing Instructions**
**N18982 - Draft Call for Proposals for MPEG-I Immersive Audio**

## 2 MPEG-I: Scene Description for MPEG Media

Extensions are provided to existing scene description formats in order to support MPEG media, in particular immersive media. MPEG media includes but is not limited to media encoded with MPEG codecs, media stored in MPEG containers, MPEG media and applications formats as well as media provided through MPEG delivery mechanisms. Extensions include scene description format syntax and semantics and the processing model when using these extensions in combination with a presentation engine. It also defines Media Access APIs for communication between the presentation engine and the Media Access Functions for these extensions. While the extensions defined in this part may be applicable to other scene description formats, a specific instantiation is provided for "The GL Transmission Format (glTF) 2.0" as defined by Khronos.

**Overview**
Immersive media applications, for example those that aim to provide true AR and 6DoF experiences, require a scene description format that describes a rich 3D scene that enables physically-based rendering (PBR) of the audio-visual content. Instead of specifying a new Scene Graph format for this purpose, this specification builds on the well-established glTF 2.0 format that is standardized by the Khronos Group. The following diagram depicts the glTF 2.0 format hierarchy and shows the extensions defined in this specification:




In addition to the extensions, which provide a tight integration of MPEG media with the Scene Description, the interface between the Presentation Engine and the Media Retrieval Engine is defined. Finally, a processing model as well as conformance and validation definitions of scene descriptions according to this specification are provided.

**glTF 2.0 Overview and Extension Mechanisms**
glTF defines an extension mechanism (section "Specifying Extensions") that allows the base format to be extended with new capabilities. Any glTF object can have an optional extensions property that lists the extensions that are used by that object. All extensions that are used in a glTF scene must be listed in the top-level `extensionsUsed` array object. Extensions that are required to correctly render the scene must also be listed in the `extensionsRequired` array.

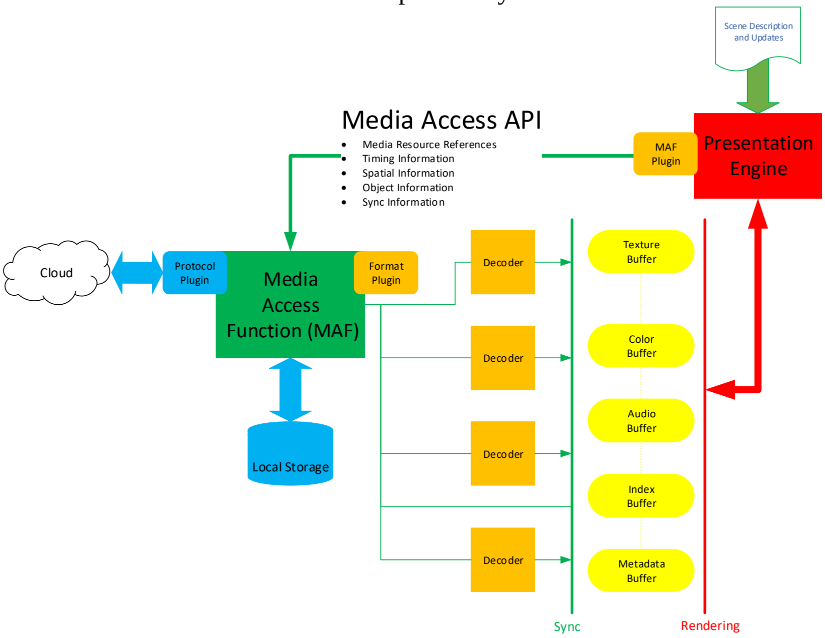
As an example of possible extension, MPEG is currently defining a 6DoF audio encoder input format to address the MPEG-I requirements on 6DoF scene audio. glTF does not provide any support for audio scenes. To address this gap, a new node type and new material extension should be defined. Similar to Javascript for HTML documents, an active processing may be supported in order to update a glTF scene description. This allows to update the description object model in an asynchronous manner (based on events such as interactivity or server events) as well as in a synchronous manner with a media source. In the latter case, a model as defined for Web Resource Track model for which updates are timed using an ISO BMFF track format aligned with ISO/IEC 29001-15 should be defined. glTF 2.0 can be extended beyond the core specifications by basically 4 means:

- Vendor extensions:
- EXT Extensions:
- KHR Extensions
- KHX Extensions

This specification defines extensions to glTF under the vendor-specific extension framework with an MPEG namespace.

**Architecture**

The scene description is consumed by a Presentation Engine to render a 3D scene to the viewer. The extensions defined in this specification allow for the creation of immersive experiences using MPEG media. The scene description extensions are designed with the goal of decoupling the Presentation Engine from the Media Retrieval Engine. Presentation Engine and Media Retrieval Engine communicate through the i-m interface, which allows the Presentation Engine to request media data required for the rendering of the scene. The Media Retrieval Engine will retrieve the requested media and make it available in a timely manner and in a format that can be immediately processed by the Presentation Engine. For instance, a requested media asset may be compressed and residing in the network, so the Media Retrieval Engine will retrieve and decode the asset and pass the resulting media data to the Presentation Engine for rendering. The media data is passed in form of buffers from the Media Retrieval Engine to the Presentation Engine. The requests for media data are passed through the Media Retrieval API from the Presentation Engine to the Media Retrieval Engine. The following diagram depicts the reference architecture as described previously:




The interfaces and components in green are in scope for this specification. The following assumptions apply:

- The format of the buffers is dictated by the scene description and is passed to the MAF through the Media Access API
- Decoder might need to perform necessary transformations to match the buffer format and layout declared in the scene description for that buffer
- The fetching of scene description and scene description updates may be triggered by the MAF.

NOTE: Upon definition of the scene update mechanism, the architecture might need to be adjusted.

**Output Documents**

**N19046 - Request for subdivision of ISO/IEC 23090-14 Scene Description for MPEG Media**
**N19070 - WD on Scene Description for MPEG Media**
**N19071 - Technologies under Considerations on Scene Description for MPEG Media**
**N19072 - Procedures for standard development and reference software of ISO/IEC 23090-14**

# 3 Explorations: Video Coding for Machines

Traditional coding methods aim for the best video/image under certain bit-rate constraint for human consumption. However, with the rise of machine learning applications, along with the abundance of sensors, many intelligent platforms have been implemented with massive data requirements including scenarios such as connected vehicles, video surveillance, and smart city.

The sheer quantity of data being produced constantly leads previous methods with a human in the pipeline to be inefficient, and unrealistic in terms of latency and scale. There are additional concerns in transmission and archive systems which require a more compact data representation and low latency solution. This led to the introduction of Video Coding for Machines.

In some cases, machines will communicate amongst themselves to perform tasks without a human in the mix, while in others there will be a need for additional human consumption of the specific decompressed stream. This specific scenario is possible in surveillance use cases, where a human "supervisor" may occasionally search for a specific person, or scene in video. In other cases, the corresponding bitstream may be used for both human and machine consumption. In the case of connected cars, the features may be used for image enhancement functionality for humans and object segmentation and detection for machines.

Any use cases in which video features need to be transmitted for additional processing which may potentially be used for machine or human end users could benefit from a standard in the coded features (shared backbone). Interoperability is crucial where different manufacturers and platforms need communication to achieve a common goal. Additionally, the feature stream must be efficient for both transmission and archive concerns for both latency and space. A standard for the compressed coding of this feature stream will establish an efficient protocol for machines to communicate.

**Scope**

MPEG aims to define a compressed bitstream for extracted features in video for further processing on different nodes. This bitstream should be able to

- Enable efficient and high-performing solutions to multiple tasks
- Compares favorably in comparison to the original video file after compression with regards to performance and bitrate

**System Overview**

The generic system architecture contains: a feature extractor that returns either processed or unprocessed video, and a compression scheme optimized for features. The feature extractor and compression scheme can be optimized for either a single task or multiple, and the size of the compressed stream should compare favorably to traditional coding techniques on the unprocessed video. The features may take different forms as described in below.

The decompressed output of the feature maps may then be used for post-processing tasks, which may include machine consumption tasks and human consumption tasks. The output of the Feature Extraction Network is a standardized feature stream, which could include a stream of features, low bitrate video, and other information such as a list of key points. The decompressed bitstream can be used for both key tasks for machine vision and human vision. There is an optional profile from NNR for the Feature Extraction Network and the Task Specific Networks. The VCM decoder may be updated by sending appropriate components of the decoder, typically compressed using NNR. The MPEG activity on Video Coding for Machines (VCM) aims to standardize a bitstream format generated by compressing a previously extracted feature stream and an optional video stream.

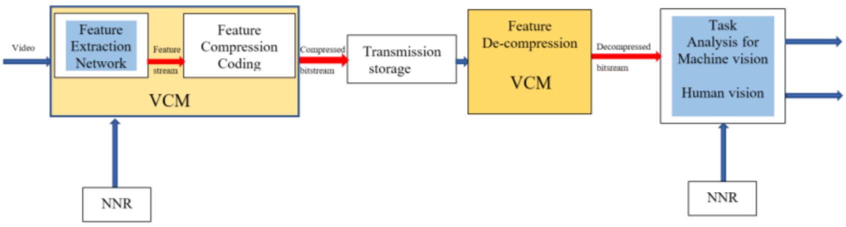



**Figure 3 - Pipeline for Video Coding for Machines (VCM)**

**Use Cases**

- **Surveillance**
  Recently, surveillance systems have incorporated the use of neural networks for different tasks such as object detection and tracking. However, current surveillance systems often take up large amounts of data for storage due to the number of sensors and length of video to be recorded.

- **Intelligent transport**
  In smart traffic system, cars may need to communicate features between each other and other sensors in order to perform different tasks. Sensors in the infrastructure may communicate features towards different vehicles, which then use these features to do object detection, lane tracking, etc. Final processing of these features is done on the individual vehicles.

- **Smart City**
  With the rise of IoT, there is a high degree of interconnectivity between different node sensors and devices. It is important for these devices to communicate with each other to optimize and efficiently solve tasks. Different vendors may develop part of the VCM pipeline, and there is a need for interoperability between devices and systems. Smart City applications encompass use cases such as traffic monitoring, density detection and prediction, traffic flow prediction and resource allocation.

**Summary of Proposed Sub-tasks**

| Task | Description | Surveillance / Smart City | Intelligent Transportation |
|---|---|---|---|
| Object Detection | Determine a bounding box for an object that may be in the input image / video along with object id | x | x |
| Object Segmentation | Determine which pixels belong to which objects by defining binary masks for each image | x | x |
| Image/Video Reconstruction | Given the compressed feature stream with an additional bit-stream return the reconstructed image/video | x | |
| Image/Video Enhancement | With an additional bit-stream return the reconstructed image/video enhanced for human consumption such as super resolution, low light | x | |
| Object Tracking | Determine the location of an object throughout video along with object id | x | x |
| Event Recognition | Determine which event has occurred in the video | x | x |
| Event Prediction | Predict which event will occur | x | x |
| Anomaly Detection | Determine whether or not a nonstandard deviation has occurred such as malfunctions | x | x |
| Density Estimation | Estimation of population density within a certain bounding box | x | |
| Event Search | Provide a time stamp for when an event has occurred given an input image or video | x | |

**Output Documents**
**N18966 - Use cases and draft requirements for Video Coding for Machines**
**N19076 - Video coding for machines: Table of collected evidence**
**N19077 - Draft Call for Evidence for Video Coding for Machines**
**N19080 - Evaluation Framework for Video Coding for Machines**
**N19138 - Call for Test Data for Video Coding for Machines**

# 4 Smart Contracts

**Benefits of MPEG IPR Ontologies**

- MPEG IPR ontologies can be used by music and media value chain stakeholders to **share and exchange all metadata and contractual information connected to creative works**, in a standardised and interoperable way, leading to transparent payment of royalties and reduced time spend searching for the right data.
- The latter is due to **inference and reasoning capabilities** inherently associated with ontologies. That is, **knowledge and data** can be derived by **evidence** (true facts) and **logic** based on rich **semantic copyright models** expressed by MPEG IPR ontologies. In such way, the data derived are **unambiguously interpretable** facilitating **efficient processing in B2C and B2B** music and media value chains.

**The Challenge: From MPEG IPR Ontologies to Smart Contracts and Blockchains**

How MPEG IPR standardised ontologies can be converted to smart contracts being executable on existing blockchain environments, thus

1. **enriching blockchain** environments with **inference and reasoning** capabilities inherently associated with ontologies, while
2. **increasing the trust level** among music and media value chain stakeholders **for sharing data** in the ecosystem, since the data will be cryptographically secured, and **its truth is verified by a blockchain**?

**Towards a Semantic Music and Media Blockchain**

- While lots of research literature deals with ontologies' semantic-level interoperability **(linking different ontologies)** and blockchains' protocol-level interoperability **(transferring verified data from one to another)**, the **interoperability gap** between them has not yet been sufficiently addressed.
- Towards this direction, MPEG is not going to develop any blockchain based technology or any new language for smart contracts.
- **MPEG aim is to develop the means (e.g., protocols and APIs**) for converting MPEG IPR ontologies to smart contracts being executable on existing blockchain environments.
- Such developments towards a ***semantic music and media blockchain*** have the potential to unlock both the semantic web and the creative economy.

### Description of the Challenge & Hints for Solution

**Challenge**

- Panos Kudumakis, Thomas Wilmering, Mark Sandler, Víctor Rodríguez-Doncel, Laurent Boch, Jaime Delgado, '[The Challenge: From MPEG Intellectual Property Rights Ontologies to Smart Contracts and Blockchains]', IEEE Signal Processing Magazine, pp. 89-95, Vol. 37, Issue 2, March 2020.

**Hints for Solution**

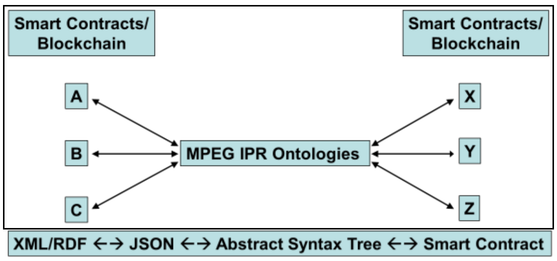
- Olivia Choudhury, Nolan Rudolph, Issa Sylla, Noor Fairoza, Amar Das, '[Auto-Generation of Smart Contracts from Domain-Specific Ontologies and Semantic Rules]', IEEE Blockchain, Halifax, Canada, 30 July-3 Aug. 2018.

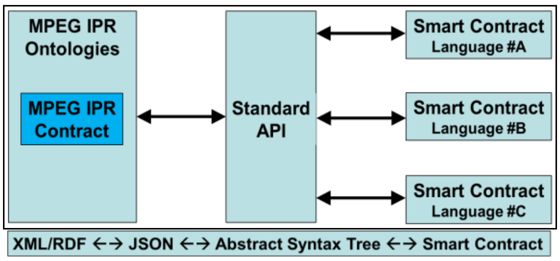
### MPEG IPR Ontologies: Interlingua for Smart Contracts Conversion

Electronic contracts are implemented in blockchains as smart contracts. One shortcoming is that there is no way to deduce from a smart contract the clauses that the smart contract contains. Publishing the human readable contract does not ensure that the clauses of the human readable contract correspond to the clauses of the smart contract. There should be a way that allows the other party of the smart contract to know beyond doubt what the clauses of the smart contract express. However, MPEG IPR ontologies facilitate the one-to-one expression and linking of the clauses of the human readable contract to the clauses in the MPEG IPR ontology-based contract (electronic/digital contract).

A standard way to further translate MPEG IPR ontology-based contracts to smart contracts will ensure users that the clauses of the smart contract executed by a blockchain correspond to the clauses of the MPEG IPR ontology-based contract and, thus to the clauses of the human readable contract. By doing this conversion in a standard way for several smart contract languages would ensure MPEG IPR ontologies prevail as the interlingua (Esperanto) for transferring verified contractual data from one blockchain to another.

The API to be standardised and the methodology for achieving the conversion from MPEG IPR ontology-based contracts to smart contracts is shown in the following Figures. Such a conversion has already been proved for the Liquidity smart contracts language supported by Tezos blockchain.




### AHG Activities Overview

1. **Solicit industry participation in the area of smart contracts**

   26 members subscribed (covering both ontologies & blockchain communities)

2. **Dissemination of MPEG IPR Ontologies**
   - Panos Kudumakis, Thomas Wilmering, Mark Sandler, Víctor Rodríguez-Doncel, Laurent Boch, Jaime Delgado, "[The Challenge: From MPEG Intellectual Property Rights Ontologies to Smart Contracts and Blockchains]", IEEE Signal Processing Magazine, pp. 89-95, Vol. 37, Issue 2, March 2020.
   - Panos Kudumakis, Thomas Wilmering, Mark Sandler and Jeremy Foss, "[MPEG IPR Ontologies for Media Trading and Personalization]", 1st International Workshop on Data-driven Personalization of Television (DataTV'19) at ACM International Conference on Interactive Experiences for Television and Online Video (TVX'19), Manchester, UK, 5-7 Jun. 2019.
   - Panos Kudumakis, Thomas Wilmering, Mark Sandler, Víctor Rodríguez-Doncel, Laurent Boch and Jaime Delgado, "[MPEG IPR Ontologies]", ISO/IEC JTC1/SC29/WG11/N18500, Geneva, CH, Mar. 2019.

3. **Input Documents**

   [M52815] - AHG on MPEG-21 Contracts to Smart Contracts

   [M51587] - On the coupling of visual fingerprinting to blockchain

   This contribution presents a technical solution ensuring the on-chain/off-chain load balancing of the tasks requiring computational/storage resources that exceed the current capabilities of blockchains. It allows for a straightforward coupling of the MPEG-21 CEL contracts to a blockchain, in the sense that the MPEG-21 CEL translation to smart contracts can be delegated from the on-chain to the off-chain blockchain processing and subsequently authenticated by the blockchain.

M51588 - MPEG-21 CEL to TEZOS blockchain smart contracts conversion for visual content tracking

This contribution advances a processing-pipeline and establishes a proof-of-concept for the automatic conversion of the MPEG IPR ontologies based specific contracts (i.e., clauses in the MPEG-21 CEL sense) to blockchain smart contracts (with illustration for the Liquidity language considered for the Tezos blockchain). It also identifies the scope of the current standardization efforts, namely a unique API (abstract class) allowing for different blockchain technologies to fetch data from MPEG IPR ontologies.

M52742 - IPNG project: Blockchain solution for local authorities' collaborative contractual signatures

This contribution reports on the IPNG project, a French national project related to a blockchain solution for local authorities' collaborative contractual signatures.

## 4. Output Documents

N18947 - AHGs Established at 129th Meeting

N18974 - Background and use cases for MPEG-21 smart contracts

This document approved by the Requirements group is based on the aforementioned contributions and, in particular, M51376, M51030 and M51588. It presents the challenge the MPEG AHG on 'MPEG-21 Contracts to Smart Contracts' is mandated to address, that is, converting MPEG IPR ontologies to smart contracts being executable on existing blockchain environments. It also includes related use cases (e.g., on-demand streaming, digital sale and radio broadcast), a walkthrough, the API identified for standardisation and related resources for further experimentation. It should be noted that for the conversion of MPEG IPR ontologies to smart contracts, the following methodology has been adopted: XML/RDF ←→ JSON ←→ Abstract Syntax Tree ←→ Smart Contract. In that way, not only the interoperability gap between MPEG IPR ontologies and blockchains is bridged, but the MPEG IPR ontologies may also become the Esperanto for transferring verified data (contractual media information) from one blockchain to another.

## AHG Recommendations

1. Implement the chain XML/RDF ←→ JSON ←→ Abstract Syntax Tree ←→ Smart Contract using Liquidity for the last conversion
2. Solicit further industry participation and contributions in the area of smart contracts (e.g., use cases & requirements)
3. Identify / create tools for converting MPEG IPR Ontology based contracts to smart contracts (e.g., Golang, Michelson, Solidity, Move)
4. Explore the use of MPEG IPR Ontologies as smart contracts in IM AF (ISO/IEC 23000-12) & possibly CMAF (ISO/IEC 23000-19)

| | |
|---|---|
| **Chairmen** | Panos Kudumakis (QMUL), Xin Wang (MediaTek) |
| **Duration** | Until next meeting |
| **Reflector(s)** | smart-contracts@lists.aau.at |
| **Subscribe** | https://lists.aau.at/mailman/listinfo/smart-contracts |

## Resources

- **Standards and software**

| Acronym | Standard | MPEG Document | Reference Software |
|---|---|---|---|
| **MVCO** | ISO/IEC 21000-19, 'Information technology -- Multimedia framework (MPEG-21) -- Part 19: Media value chain ontology', June 2010. | N11146<br>91st Kyoto | N/A |
| | ISO/IEC 21000-8/AMD2, 'Information Technology -- Multimedia Framework (MPEG-21) -- Part 8: Reference software / AMD2 Reference software for media value chain ontology', Nov. 2011. | N12135<br>97th Torino | https://tinyurl.com/y6tsr9as |

| | | | |
|---|---|---|---|
| **AVCO** | ISO/IEC 21000-19:2010/AMD1, 'Information Technology -- Multimedia Framework (MPEG-21) -- Part 19: Media Value Chain Ontology / AMD 1 Extensions on Time-Segments and Multi-Track Audio', June 2018. | N17170<br>120th Macau | N/A |
| | ISO/IEC 21000-8:2008/AMD4, 'Information Technology -- Multimedia Framework (MPEG-21) -- Part 8: Reference Software / AMD 4 Media Value Chain Ontology Extensions on Time-Segments and Multi-Track Audio', Oct. 2018. | N17404<br>121th Gwangju | https://standards.iso.org/iso-iec/21000/-8/ed-2/en/amd/4 |
| **MCO** | ISO/IEC 21000-21 **(2nd Ed.)**, 'Information technology -- Multimedia framework (MPEG-21) -- Part 21: Media Contract Ontology', May 2017. | N15940<br>114th San Diego | https://standards.iso.org/iso-iec/21000/-21/ed-2 |
| **CEL** | ISO/IEC 21000-20 **(2nd Ed.)**, 'Information technology -- Multimedia framework (MPEG-21) -- Part 20: Contract Expression Language', Dec. 2016. | N15994<br>114th San Diego | Included in N15994 |

**Output Documents**

**N18974 - Background and use cases for MPEG-21 smart contracts**