**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

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

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG11 MPEG2018/N17503**

**April 2018, San Diego, US**

**Source Requirements**

**Status Approved**

**Title Call for Proposals on Network-Based Media Processing**

**Authors** Kyungmo Park, Rufael Mekuria, Imed Bouazizi

# Introduction

Network-Based Media Processing (NBMP) is a framework that allows service providers and end users to describe media processing operations that are to be performed by the network. NBMP describes the composition of network-based media processing services out of a set of network-based media processing functions and makes these network-based media processing services accessible through Application Programming Interfaces (APIs).

An NBMP media processing entity performs media processing tasks on the input media data and the related metadata. NBMP also provides Control Functions that are used to compose and configure the media processing.

This Call for Proposal is soliciting proposals to define a network-based media processing framework that addresses the identified use cases and satisfies the NBMP requirements [1].

# Scope

The NBMP framework will define external interfaces between the Media Source/Media Sink and the Media Processing Entity, that will allow users of the framework to access the framework, configure the media processing, upload/stream media data to the network for media processing, and access the processed media and the resulting metadata in a scalable fashion in real-time or in a deferred way.

The media and metadata formats that are used between Media Processing Entities in a media processing pipeline are also within scope. Workflow management that is used to instantiate new media processing entities and to compose media processing services into a pipeline of media processing entities is also in scope.

The following diagram depicts the NBMP architecture that will be used as a reference architecture to scope the NBMP work:



Figure NBMP Reference Architecture

In this document, a network-based media processing system is defined as follows:

1. **NBMP system:** A system for processing that is performed across processing entities in the network. It consists of Media Source Entity, Media Processing Entity and Media Sink Entity.
2. **Media Source Entity:** the capture entity which provides the raw media content to be processed, for example, a digital camera, a microphone, an encoder, or persistent storage.
3. **Media Processing Entity:** performs a task(s) on the input media, which may be the output of a prior Media Processing Entity.
4. **Media Sink Entity:** consumes the output of Media Processing Entity through existing delivery methods, for example through download, DASH, MMT, or other means. This entity is not standardized through this work.
5. **Control Function:** provides functionalities for control and management of task(s) and workflows.
6. **Processing Function:** provides functionalities for media processing in the Media Processing Entity given instructions from the Control Function.
7. **Task**: is the running instance of Processing Function that gets executed inside the Media Processing Entity.
8. **NBMP workflow:** provides a chain of one or more task(s) to achieve a specific media processing. Chaining of task(s) can be sequential, parallel, or both at any level of the workflow.
9. **NBMP Format:** the media format that is exchanged in between the Media Source Entity and the Media Processing Entities of NBMP system, and between individual Media Processing Entities inside the NBMP system
10. **NBMP Publish Format:** the media format of the content that is sent from Media Processing Entity to the Media Sink Entity
11. **Media Resources:** the media data that is captured by the Media Source Entity and is sent to the Media Processing Entities of the NBMP system
12. **Supplementary Information:** the metadata or the auxiliary information for the NBMP system.
13. **Workflow Description:** describes the information that enables the NBNP workflow

# Timeline – To be revised

|  |  |
| --- | --- |
| 2018/04/20 | Final Call for Proposals issued |
| 2018/06/20 | Registration deadline |
| 2018/07/11 | Deadline for submission of description of the proposals |
| 2018/07/16-20 | Evaluation of the proposals |
| 2019/10/11 | Final draft international standard may be anticipated |

# Submission Requirements

The valid proposal should include all listed items in this section as part of the submission.

The following items are required as part of the submission:

* A description of the network-based media processing framework, including the workflow management function
* The specification of the interfaces in an interface design language
* A description of the data formats for media data and metadata
* Example workflows that depict the usage of the framework, e.g., to realize a selected set of use cases

Submissions that are incomplete or delivered late will not be considered.

## Input contribution

Each proposal must be described in an input contribution to the 123rd MPEG meeting and should at least include the following elements:

1. A detailed technical description of the proposed technology.
2. A description of how the proposal fulfills the requirements as listed in the requirement document [1].
3. A description of which parts of the proposed technology would be standardized by MPEG, preferably accompanied by draft specification text.
4. It is desirable, but not necessary, for the proposals to include additional information on the realization of the use cases of network-based media processing based on the provided proposal.

## Source code

* Proponents are encouraged (but not required) to allow MPEG members to have access, on a temporary or permanent basis, to their source code.
* Proponents are encouraged to submit a statement about the programming language in which the software is written, e.g., PHP, Node.js, or Python, and the platform(s) on which the binaries were compiled. Note that low-level programming optimizations such as assembly code/intrinsics and external compression libraries are discouraged.

Proponents are advised that, upon acceptance for further evaluation, it will be required that certain parts of any technology proposed to be made available in source code format to participants in the core experiments process and for potential inclusion in the prospective standard as reference software. When a particular technology is a candidate for further evaluation, commitment to providing such software is a condition of participation. The software shall produce identical results to those submitted to the test. Additionally, submission of improvements (bug fixes, etc.) is encouraged.

# Evaluation Methods and Procedures

The proposals will be evaluated based on their fulfillment of the requirements. Each proposal shall provide the following table filled out to assist with the evaluation of that proposal:

|  |  |  |  |
| --- | --- | --- | --- |
| Id | Criteria | Evaluation Facts | Conclusion |
| General Requirements (G) | | | |
| G.1 | NBMP shall support NBMP format |  |  |
| G.2 | NBMP shall support media processing distributed across processing entities |  |  |
| G.3 | NBMP shall support media processing with processing entities that are interconnected with each other in a network |  |  |
| G.4 | NBMP shall have advantages in terms of compression, computation, bandwidth, or storage efficiency over existing usage of MPEG standards |  |  |
| G.5 | NBMP shall support any type of media content (be media agnostic), including the existing MPEG codecs (which may produce very low to very high data rates) and MPEG formats such as ISO/IEC 13818-1, ISO/IEC 14496-12, ISO/IEC 23008-1 and ISO/IEC 23009-1 for network-based media processing |  |  |
| G.6 | NBMP shall support content distribution over an IP-based connection |  |  |
| G.7 | NBMP shall support content distribution over HTTP and RTP |  |  |
| G.8 | NBMP shall support content distribution over UDP and TCP |  |  |
| G.9 | NBMP shall support content distribution using streaming |  |  |
| G.10 | NBMP shall support content distribution using file delivery |  |  |
| G.11 | NBMP shall support content distribution using progressive download |  |  |
| G.12 | NBMP shall support content distribution using existing MPEG protocols (e.g. ISO/IEC 23008-1). |  |  |
| G.13 | NBMP shall support content distribution of media using existing standardized delivery formats |  |  |
| G.14 | NBMP shall provide the APIs to support the network-based media processing |  |  |
| G.15 | NBMP should support push-based streaming, e.g., over unidirectional or multicast channels. |  |  |
| G.16 | NBMP should support push-based progressive download, e.g., over unidirectional or multicast channels. |  |  |
| G.17 | NBMP should support hybrid delivery environments, such as multiple transmission channels, possibly of different types |  |  |
| G.18 | NBMP should support multipath content distribution environment |  |  |
| G.19 | NBMP should support heterogeneous network environments including broadcast, unicast, multicast, storage, peer-to-peer, and mobile |  |  |
| G.20 | NBMP should support distribution of media processing workload among multiple distributed processing entities |  |  |
| G.21 | NBMP should tolerate failures of distributed media processing entities |  |  |
| G.22 | NBMP should support a creation of user-centric media streams without affecting the compliance with the existing standards, etc |  |  |
| G.23 | NBMP shall support usage of CDN deployment to enable the optimized cached content |  |  |
| Requirement for Media resource (M) | | | |
| M.1 | NBMP shall support the identification of conformance points of each content component (e.g., elementary stream). |  |  |
| M.2 | NBMP shall support to upload locally generated multimedia content to the processing entity in the network. |  |  |
| M.3 | NBMP shall support storage of multiple components for a single application. |  |  |
| M.4 | NBMP shall support storage of content that uses common components. |  |  |
| M.5 | NBMP should support media processing to ingest the media resource to the media processing entity. |  |  |
| M.6 | NBMP should support the creation of temporal and spatial overlapping sub-video streams from the main video stream |  |  |
| M.7 | NBMP should support pre-transcoded video content for efficient media caching in the network. |  |  |
| M.8 | NBMP should support media aware cache by media processing of cached NBMP format. |  |  |
| M.9 | NBMP should support media processing of non-video contents, such as cloud gaming, point-clouds, 3D graphics, and multiple sensory data. |  |  |
| M.10 | NBMP should support the media processing of video streams with overlay graphics. |  |  |
| Requirement for supplementary information (S) | | | |
| S.1 | NBMP shall support content component identification. |  |  |
| S.2 | NBMP shall support clock recovery (e.g., PCR). |  |  |
| S.3 | NBMP shall enable content synchronization at capture (from different sources) and playback (on different screens). |  |  |
| S.4 | NBMP shall provide means to signal NBMP metadata (e.g., side information). |  |  |
| S.5 | NBMP shall support in-band or out-of-band carriage of format information such as bit-rate, resolution, codec type, frame rate, dynamic range, color space, sub-sampling type, segment duration. |  |  |
| S.6 | NBMP shall support interfaces to receive congestion notification from other network elements and enable appropriate congestion control. |  |  |
| S.7 | NBMP should support functionalities for adaptive media processing based on awareness of context, service, application, user’s history, location, client device, and processing resources. |  |  |
| S.8 | NBMP should support exchange of messages among media processing entity. |  |  |
| S.9 | NBMP should support signaling messages of media service for optimal use of network resources in distributed network entities. |  |  |
| S.10 | NBMP should support generating metadata as opposed to only the processed media. |  |  |
| S.11 | NBMP should support signaling of metadata about the capturing information such as camera parameters, geometric relationship among the multiple cameras to improve the processing quality. |  |  |
| S.12 | NBMP should support the metadata and auxiliary information for dynamic configuration of media delivery. |  |  |
| S.13 | NBMP should support signaling of media distribution information such as identification of streams, required bitrate and URI assignment to the media processing entity. |  |  |
| S.14 | NBMP should support the feedback information based on the user preference and device capability to a media processing entity. |  |  |
| S.15 | NBMP should provide a way to use the existing feedback mechanism to allow a Media Sink to exchange an information with a Media Processing Entity, e.g. to indicate the current viewport. |  |  |
| S.16 | NBMP should provide a method to transfer the information of full or partial structure of media. |  |  |
| Requirement for workflow description (W) and API | | | |
| W.1 | NBMP shall support the workflow description to set-up the NBMP workflow. |  |  |
| W.2 | NBMP shall support the identification of workflow |  |  |
| W.3 | NBMP shall support the seamless update of workflow during the media processing. |  |  |
| W.4 | NBMP shall support the identification of media processing task(s). |  |  |
| W.5 | NBMP shall support transcoding for adaptive streaming in the network. |  |  |
| W.6 | NBMP shall support low latency content distribution (e.g., to support conversational applications, live content, etc.). |  |  |
| W.7 | NBMP shall support use of different QoS types and levels. |  |  |
| W.8 | NBMP shall support low-complexity format conversion for content distribution and storage. |  |  |
| W.9 | NBMP shall support dynamic media processing during content distribution. |  |  |
| W.10 | NBMP shall support to configure and execute processing of content to change encodings and frame packing variants |  |  |
| W.11 | NBMP shall provide the APIs to enable the configuration and execution of media processing in distributed networks |  |  |
| W.12 | NBMP shall provide the APIs to enable the feedback and analyze of media processing |  |  |
| W.13 | NBMP shall support distribution of pre-transcoding of video content |  |  |
| W.14 | NBMP shall support low-latency real-time stitching fully synchronized media data distribution for real-time stitching (to reduce buffering requirements) |  |  |
| W.15 | NBMP shall support QoE reporting to enable optimized rate adaptation |  |  |
| W.16 | NBMP should support the dynamic update of workflow without interrupting the existing NBMP workflow. |  |  |
| W.17 | NBMP shall support the required information of quality of media processing to configure the NBMP workflow |  |  |
| W.18 | NBMP should support to collect user preferences to configure the media creation procedures. |  |  |
| W.19 | NBMP should support a format for requesting media processing of time segments of media data for adaptive bit-rate streaming. |  |  |
| W.20 | NBMP should support dynamic configuration (e.g., merging) of content components during content distribution. |  |  |
| W.21 | NBMP should support relaying received content stored on storage devices. |  |  |
| W.22 | NBMP should support control of content relay and retransmission. |  |  |
| W.23 | NBMP should support signaling, content distribution, and utilization of content using multiple protection and rights management tools. |  |  |
| W.24 | NBMP should support content encryption using encryption methods |  |  |
| W.25 | NBMP should support seamless change between content rights management schemes. |  |  |
| W.26 | NBMP should support load balancing among media processing entities. |  |  |
| W.27 | NBMP should support adaptive scheduling mechanisms for dynamic scheduling of media processing tasks on distributed entities |  |  |
| W.28 | NBMP should support migration of the media processing task(s) among distribution entities during the processing |  |  |
| W.29 | NBMP should support a logically centralized entity to manage the whole media processing distribution environment |  |  |
| W.30 | NBMP should support real-time metrics for available resources on distributed media processing entities, such as number of clusters, available memory, available processors, etc. |  |  |
| W.31 | NBMP should support metadata and/or API accounting for total queuing delay until the start of media processing |  |  |
| W.32 | NBMP should support media processing that is either batch, or data/event-driven |  |  |
| W.33 | NBMP should support media processing that is either delay-sensitive or delays tolerant |  |  |
| W.34 | NBMP should support video analysis techniques on the video stream, e.g., to perform object recognition and tracking |  |  |
| W.35 | NBMP should support targeting caching which related to user preference. |  |  |
| W.36 | NBMP should support large-scale caching for considering large-scale media distribution. |  |  |
| W.37 | NBMP should be able to make use of mobile’s capability to use Multiple Radio Access Technologies and shall support multi-path delivery |  |  |
| W.38 | NBMP should support dynamic content offloading to different channels and to broadcast channels |  |  |
| W.39 | NBMP should support low-latency streaming with high bitrate allocation – even higher for low-latency compressed or raw media data streaming |  |  |
| W.40 | NBMP should support functionalities for filtering, modifying and mixing of media content. |  |  |
| W.41 | NBMP should support adaptive QoS-aware media processing to differentiate delay-sensitive and less delay-sensitive services |  |  |
| W.42 | NBMP should provide task completion metadata and/or accounting API (usage and duration). |  |  |
| W.43 | NBMP should support the APIs for dynamic configuration of media delivery (e.g., unicast to multicast). |  |  |
| W.44 | NBMP should support the description for multicast streaming with ABR to enable network assisted media distributions. |  |  |
| W.45 | NBMP should support pre-rendering function in the media processing entity. |  |  |
| W.46 | NBMP should support configuration information about the required processing resources (e.g., required GPU acceleration). |  |  |
| W.47 | NBMP should provide the model of workflow for providing information about a certain functional behavior of media processing pipelines. |  |  |
| W.48 | NBMP should provide abstract model to specify the workflow that characterize the desired output of a workflow given an input. |  |  |

# Participation fee

Participation in the CfP will not be associated with any fee.

# IPR

Proponents are advised that this call is being made subject to the patent policy of ISO/IEC and other established policies of the standardization organization. The persons named below as contacts can assist potential submitters in identifying the relevant policy information.

# Contacts

Prof. Dr.-Ing. Jörn Ostermann

MPEG Requirements Chair

Leibniz Universität Hannover, Institut für Informationsverarbeitung

Appelstr. 9A 30167 Hannover, Germany

Email: [ostermann@tnt.uni-hannover.de](mailto:ostermann@tnt.uni-hannover.de)

Dr. Kyungmo Park

Chair Ad hoc Group for Network-Based Media Processing

6, Hwangsaeulro 258, Bundang, Seongnam

Gyeonggi-do, Rep. of Korea 13595

Email: [kyungmo.park@sk.com](mailto:a.hinds@cablelabs.com)

# References

1. w17502, “Use cases and requirements for network-based media processing(v4)”, ISO/IEC JTC1/SC29/WG11 MPEG2018/N17502, April 2018, San Diego, US