**3GPPSA4 #120-e S4-** **220993**

**E-meeting, 17-26 Aug 2022**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.0* | | | | | | | | |
| **PSEUDO CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **26**.**857** | **CR** | pseudo | **rev** | **-** | **Current version:** | **0.3.0** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | **[FS\_5G\_MSE] Proposed update to 4.3** | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Tencent | | | | | | | | | |
| ***Source to TSG:*** |  | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | FS\_5G\_MSE | | | | |  | ***Date:*** | | | 8/03/2022 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **B** |  | | | | | ***Release:*** | | | 18 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) Rel-12 (Release 12)* *Rel-13 (Release 13) Rel-14 (Release 14) Rel-15 (Release 15) Rel-16 (Release 16)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | This document provides an update to 4.3. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | * W3C media capabilities API scope and features * OpenXR extension query * NBMP function description * 3GPP FLUS sink capabilities discovery | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  |  | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  |  | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  |  | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

CHANGE

# References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[ISO-23090-8] ISO/IEC 23090-8:2020: "Information technology — Coded representation of immersive media — Part 8: Network based media processing".

[W3CMC] W3C: "Media Capabilities", Working Draft, 3 August 2022,  
<https://www.w3.org/TR/media-capabilities/>

[OpenXR] Khronos: "The OpenXR Specification",  
<https://registry.khronos.org/OpenXR/specs/1.0/html/xrspec.html>

[NBMP] ISO/IEC 23090-8: "Information technology — Coded representation of immersive media — Part 8: Network-based media processing".

[FLUS] 3GPP TS 26.238: "Uplink Streaming".

[x] <doctype> <#>[ ([up to and including]{yyyy[-mm]|V<a[.b[.c]]>}[onwards])]: "<Title>".

It is preferred that the reference to 21.905 be the first in the list.

2. CHANGE

## 4.3 Existing specifications

### 4.3.1 General

This clause provides an overview of existing specifications that provide discovery tools that can be potentially used as a tool in Media Service Enabler specifications.

### 4.3.2 W3C Media Capabilities framework

The W3C Media Capabilities API [W3CMC] is designed to query a W3C user agent (such as a web browser) with regard to its media decoding and encoding capabilities. The intent is to provide a replacement for the HTML5 canPlayType() method and the Media Source Extension isTypeSupported() method that more accurately reflects the user agent’s media encoding and decoding capabilities.

The API supports decoding from a file, from Media Source Extension (MSE), or from WebRTC, and also encoding to media chunks that can be stored in afile or transmitted as a WebRTC stream.

- The video and audio configuration objects define the general characteristics of the video and audio configuration.

- The key system configuration object defines the key configuration for the encrypted media (EME).

For a given encoding or decoding configuration, the API returns a binary flag for each of the following values: supported, smooth, or power-efficient. It seems the logic is very similar to canPlayType(), but the configuration can be described in more detail and the result is more accurate.

The specification is characterised as follows:

1. It limits the expression of capabilities to video and audio encoding/decoding and key configuration for decryption.

2. It doesn’t define a general language to define the capabilities of various media functions other than video/audio decoding.

3. It seems various encoding configurations (e.g. motion estimation, bit rate control modes, number of reference frames, etc) cannot be described or set as the desired configuration.

3.

4. The API for querying the functionality i still based on a binary pass/fail result (similar to canPlayType), presumably to reduce the risk of fingerprinting the device, and therefore, it is not possible to get an expressive list of supported features.

### 4.3.3 Open XR extensions query

OpenXR is an API for XR applications. The application can use this API to interface with a runtime to render one or more XR scenes using one or more runtimes.

OpenXR is an extensible API and the optional functions are call extensions. Since the extensions are options, only a subset of runtimes may implement a particular extension. Therefore, an application should first query which extensions are available from its underlying runtime.

The function [xrEnumerateInstanceExtensionProperties](https://registry.khronos.org/OpenXR/specs/1.0/html/xrspec.html#xrEnumerateInstanceExtensionProperties) [OpenXR] provides the list of available extensions in the current runtime. Each extension is identified by a unique name and an extension version, and by the type of the extension.

OpenXR query functionality has the clear advantage of providing the supported extensions with one single query. However, since the supported extensions are only identified by their name and version, it is not clear whether and how the support of optional features in an extension in an implementation is explicitly signaled.

### 4.3.4 MPEG Network-Based Media Processing function description

ISO/IEC 23090-8 [NBMP] is a specification for describing media functions and workflows as microservices running in the cloud. The specification defines the data structures for describing workflows and methods for managing them. The framework enables dynamic creation of media processing pipelines, as well as access to processed media data and metadata in real time or in a deferred way. The media and metadata formats used between the media source, workflow manager, and media processing entities in a media processing pipeline are also specified.

Each function in a media processing pipeline is specified using a JSON object called a *function description* that describes the functionality, the inputs, outputs, processing characteristics, and configuration parameters, as well as the processing requirements. The function description can also describe events, and even how to set up notifications, reporting, and monitoring for that function if it is deployed as a microservice. More complex functions can be described also as a composite of simpler functions. Finally, the function description JSON object can also be used programmatically to instantiate a task or microservice that performs the function it describes.

The NBMP function description has the advantage of describing being able to describe a function with identifiers as well as a detailed description of a function. The optional parameters of an implementation of a generic function can be described using this approach.

### 4.3.5 3GPP TS 26.238 Flus sink capability discovery

3GPP TS 26.238 [FLUS] defines a set of protocols for uplink media streaming. This specification includes a method for describing the processing capabilities of the entity (known as FLUS sink) that receives the uplink stream. In this specification, these capabilities are described as a list. Each entry in the list includes a scheme identifier, the location for the description of the scheme, and a URL where the specific capability can be accessed. The FLUS sink capabilities description can be retrieved from the sink or it can be found in a sink directory.

The advantage of the FLUS sink capabilities description is its simplicity. However, since each item in the capabilities list has its own scheme, it does not provide much interoperability for describing the available functions and their detailed features, since each function defines its own scheme for describing its capabilities.

END OF CHANGES