**3GPP TSG-SA 4 Meeting #116-eS4-211411**

**Online, 9-19 Nov 2021**

|  |
| --- |
| *CR-Form-v12.1* |
| **DRAFT CHANGE REQUEST** |
|  |
|  | **26.238** | **CR** |   | **rev** |  | **Current version:** | **16.8.0** |  |
|  |
| *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 | **X** | Radio Access Network |  | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  | dCR on the extending support for network-based media processing in FLUS  |
|  |  |
| ***Source to WG:*** | Tencent Cloud, Sony Europe B.V., Nokia Corporation |
| ***Source to TSG:*** | S4 |
|  |  |
| ***Work item code:*** |  |  | ***Date:*** | 2021-11-3 |
|  |  |  |  |  |
| ***Category:*** | B |  | ***Release:*** | Rel-17 |
|  | *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 canbe 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-15 (Release 15)Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)* |
|  |  |
| ***Reason for change:*** | Extending the support for network-based media processing in FLUS. |
|  |  |
| ***Summary of change:*** | This dCR extends the FLUS specification to better support the network-based media processing functionality based on TR26.939 |
|  |  |
| ***Consequences if not approved:*** | Network-based media processing is not fully supported. |
|  |  |
| ***Clauses affected:*** | 5, 7 |
|  |  |
|  | **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:*** |  |

|  |
| --- |
| **First Change** |

### 2 List of FLUS Sink Configuration properties

All FLUS Sink Configuration properties, except for the resource id, are always carried in an HTTP message body. The access-token is always carried as part of HTTP headers. Except for the FLUS session creation request (where the id is not present), the resource id shall be present in the URL of all requests that relate to a specific FLUS Sink Configuration.

In the table below, the following assertions are made:

- Table header: C stands for Create FLUS session procedure, G is for Get FLUS session properties procedure, U is for Update FLUS session properties procedure and T is for Terminate FLUS session procedure. "I", and "O" respectively denote "request" (going **I**nto the FLUS sink), and response (going **O**ut of the FLUS sink).

- Optional ("O") means that the property may or may not be sent/received during a REST transaction. It does not necessarily mean that the property is optional. It is possible, for example, that a session is not yet active because the FLUS source has not set the property in any previous update transaction using the PUT or PATCH HTTP method, as opposed to representing a hint on the importance of the property for the FLUS sink.

- A property marked as optional (O) in a request message may be present in the request. When not present in the request body, the property, if present in the FLUS sink, will not be updated.

- A property marked as optional (O) in a response message is only present in the response when a value is assigned or changed by the FLUS sink.

- A property marked as mandatory (M) in a response message is always present in the response. The FLUS sink provides defaults, which may be modified subsequently by the content provider.

- A blank cell in the table means "forbidden" (the property cannot be added to the request or returned by the FLUS sink, depending on the transaction direction).

Table 5.3.6-1: List of FLUS Sink Configuration properties

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Name** | **Property Description** | **CI** | **CO** | **GI** | **GO** | **UI** | **UO** | **TI** |
| id | Identifier of the FLUS Sink Configuration resource. Note that "id" is only provided within an HTTP body during the Create FLUS session response. Otherwise, "id" should be present in the message URL to identify the resource in the FLUS sink.

|  |  |  |
| --- | --- | --- |
| Type | Unit | Default |
| Integer  | None  | N/A |

 |  | M |  |  |  |  |  |
| fu\_instantiation | Identifier of the FLUS media instantiation that is used by this FLUS session. Vendor specific enumeration values shall start with "vnd-" followed by a unique vendor name and optionally followed by additional characters.The F-U instantiation shall be provided as a globally unique URN.

|  |  |  |
| --- | --- | --- |
| Type | Unit | Default |
| URI  | None | All |

 |  |  |  | M | O |  |  |
| entrypoint\_URL | Entry point URL information (e.g., SIP URL) for establishing the F-U connection to start the Media streaming. Details on the Entrypoint URL is F-U instantiation specific. |  |  |  |  |  |  |  |
| processing\_description | This object provides a media processing description document that defines the post processing pipeline that the FLUS sink shall apply to received media components. The pipeline description may also set the distribution target (incl FLUS sink storage) for the media. The Object has the following properties:- type: the MIME type of the media processing description document- document: the media processing document may be embedded in this element. The document may be base64 encoded depending on the MIME type.- url: the URL to the media processing document.- response-code: the response code to a request in the media processing document. The syntax of this property is defined by the MIME type.The type and either the document property or the url property shall be provided.The following formats are supported:- The MPEG NBMP Workflow Resource, UTF-8 encoded,, as defined in [17], which describes the requested media processing and the desired distribution mechanism after the processing has been performed. The type field shall be set to "application/mpeg-nbmp-wdd+json" See Annex X on use of NBMP in FLUS. | O | O |  | O | O | O |  |

|  |
| --- |
| **Second Change** |

#### 7.1.1.1 Capabilities Resource

A capabilities resource provides a representation of the capabilities of a FLUS sink. Different properties of capabilities resources represent the capabilities of the corresponding FLUS sink.

Each capabilities resource has the set of properties described in Table 7.1.1.1-1.

Table 7.1.1.1-1: Properties of Sink Resource

|  |  |  |
| --- | --- | --- |
| **Property Name** | **Description** | **Example Values** |
| capabilities | List of supported features and instantiations by the FLUS sink. Each capability is to be expressed using an object element of an array. The object has the following attributes:- A scheme URN to identify the capability- An optional location URL, from which a description for the capability can be retrieved. The format of description is defined by the scheme URN- An optional location URL, through which the feature or instantiation can be directly accessed. The access protocols and the use of feature or instantiation is defined by the scheme. | { “scheme” : “urn:vnd:xzy:capability-name”, “location” : “<http://vnd.com/xzy/capability-name>”,“url”: “<http://vnd.com/xzy/capability-access>”} |

As described in Table 7.1.1.1-1 above, each capabilties resource describes the capabilities of the corresponding FLUS sink. A FLUS source can retrieve the capabilites resource description of a FLUS sink and make a decision if it wants to use the corresponding FLUS sink as described using the capability exchange procedure in section 7.3.

The attributes in Table 7.1.1.1-1 may be used for the following purposes:

* “scheme”: to identify the capability with a unique id. This id is also used to identify the format of the capability’s description document identified by the attribute “location”
* “location”: the address where a document can be found that describes the detailed properties and configuration of the capability such as supported features, the configuration parameters, and the units and ranges of parameters.
* “url”: the address for using the capability. For instance, if the capability needs set up, this address can be used to configure the capability. As an example, the url for a network-based media processing (NBMP) is the URL address of the NBMP Workflow Manager that enables creating, retrieving, getting an update, and deleting the workflow from this address.