**Agenda Item:**  11.6

**Source:** Tencent (Rapporteur)

**Title:** FS\_FLUS\_NBMP: Update to Permanent Document

**Document for:** Agreement

# *Note: The document is an update to* S4-20106, *approved in S4-110, with the additions of two accepted documents in the FLUS\_NBMP adhoc calls: S4aM200493 and S4aM200600.*

# Introduction

A new work item on “Study on the use of NBMP in FLUS” as defined in [SP-200053](http://www.3gpp.org/ftp/tsg_sa/TSG_SA/TSGS_87E_Electronic/Docs/SP-200053.zip)was approved during SA#87-E in March 2020.

The objective of this study item is to identify the workflow for NBMP media processing with the TS26.238. Such workflow should start from a request for the establishment of a FLUS session and then include the instantiation, running, and monitoring of media processing application requested by FLUS source on a FLUS sink.

The concrete objectives are as follows:

* Develop a detailed workflow of the establishment of FLUS session and NBMP workflow based on TS26.238.
* Investigate if any signalling, format, or protocol is missing from the TS26.238 and NBMP specifications to successfully establish the above workflows.
* Collect relevant and exemplary use cases for the above environment.
* Map 3GPP network QoS parameters to the NBMP QoS parameters and identify the possible missing QoS parameters in the NBMP specification.
* Investigate the possible improvements and extensions of the workflow by enhancing TS26.238 and NBMP specifications

The output of this study is to extend TR26.939 to include the combined FLUS and NBMP workflow, detailed description of necessary operations, and exemplary use cases.

This permanent document addresses the requirements, the use cases, and workflow descriptions and their operational aspects which would be proposed to be added to TR26.939.

# Use cases

## Live streaming with FLUS

|  |
| --- |
| **Use Case Name** |
| Basic multi-rate live streaming of user-generated content (UGC) |
| **Description** |
| Kim is subscribed to an Application for live streaming of captured videos from her everyday life. Based on the previous number and diversity of Kim’s usual audience (e.g. close friends), Application has an “audience codecs-rates” profile which represents the typical number of stream needed based on Kim’s previous streaming sessions. Kim starts the session live stream session. While Kim is uploading a single stream using FLUS, the server Application is commanding the running of multiple transcoders based on Kim’s audience codec-rates profile. If new users join Kim’s streaming session which could not be supported with the current codecs-rates, the Application may add more transcoders to add to multirate streaming in the session.  Variations:   1. Capabilities:    1. There are sufficient resources available at FLUS Sinks, so there is no need to check whether the picked FLUS Sink has the required real-time multi-rate transcoding capabilities.    2. Available FLUS Sinks might have limited capabilities. The server application must find a Sink capable of running the transcoding session. 2. Server or Device Application    1. The UE’s Application is responsible for setting up the FLUS and NBMP session, as well as managing the audience codecs-rates profile.    2. The UE’s Application is responsible for setting up the FLUS session. The Sever Application is responsible for setting up the NBMP session. |
| **Categorization** |
| **Delivery: Live Streaming**  **Device: Any device connected to the cell network** |
| **Preconditions** |
| 1. On the device:    1. A 3GPP supported encoder is installed.    2. UE’s Application is installed which supports NBMP Source functionalities.    3. A 3GPP FLUS Source is installed. 2. On the network,    1. One or more FLUS Sinks are installed that one or more of them supports NBMP Workflow Manager functionality.    2. One or more FLUS Sinks are installed, that run various instances of decoding/encoding or transcoding, and the decoder/encoder/transcoder are described as NBMP Functions in an NBMP repository.    3. A FLUS Sink may have limited capabilities, i.e the codecs it supports and/or the number of concurrent transcoders.    4. The FLUS Sink’s load may vary dynamically due to the other parallel network processing sessions.    5. Network storage is available to store the encoded content for time-shifted streaming. |
| **Requirements in terms of Capabilities and QoS/QoE Considerations** |
| * Capabilities   + Discovering of Sink’s network processing capabilities if needed   + Setting the FLUS and NBMP sessions   + Start the Sessions   + Change the workflow by adding more transcoders if needed during the session   + Real-time transcoding and packaging to different codecs and different bitrates   + Offloading an originally selected sink to a new sink to handle additional needed processing that cannot be accommodated by the original sink * KPI   + Supporting the use-case   + Minimum extension of FLUS and NBMP Standard, preferably none. |
| **Feasibility and Industry Practices** |
|  |
| **Nominal Cost Analysis** |
| The cost of service increase linearly with the number of ingests.  The cost of service increase less than linearly with the number of download streaming clients as the encoding and caching requirement will be common with a large number of viewers. |
| **Benefits and Impact** |
| The multi-rate encoding is not possible always on the sender device or very battery consuming. Even the use of multiple FLUS sessions is not efficient.  The multi-rate encoding for a various number of devices in every FLUS sink might not be feasible depending on the availability of resources on Sink.  Even a capable Sink may not have resources available due to its load at the session start time. |
| **Potential Technical Requirements** |
|  |
| **Potential Standardization Status and Needs** |
| 1. Discovery of dynamic processing capabilities of FLUS Sink:    1. Available hardware resources    2. Support for NBMP Workflow instantiation and management    3. Built-in (hardware-assisted and/or software-optimized) encoders/transcoders 2. Start and management of FLUS and NBMP sessions by the UE’s Application via FLUS Control Source and NBMP Source. 3. Start and management of NBMP session by Server Application aligned with start and management of FLUS session by UE’s Application |

## Rich on-demand video streaming

|  |
| --- |
| **Use Case Name** |
| Rich on-demand video streaming |
| **Description** |
| Khloe, Kim’s sister, is subscribed to an Application for uploading her videos, enhancing them, and adding rich features to them before publishing them as on-demand content. Example of enhancements and added rich features are:   1. Enhancing the quality of the video with e.g. noise removal, white balance correction, color correction, prebuilt filters, etc. 2. Indexing the content 3. Creating thumbnail navigation 4. Extracting subtitle from audio and add it to the video 5. Translation and adding multiple language subtitles 6. Adding interactive tags to the shops, restaurants and other service stores in the scene 7. Detecting any unintended improper shots and marking them for Khloe to review   Based on the previous number and diversity of Khloe’s usual audience (e.g. close friends), Application has an “audience codecs-rates” profile which represents the typical number of streams needed for streaming sessions. Depending on the service, the encoding can be done ahead of time, or on-fly when is requested by an audience. The on-demand content shall be available on a reasonable time set by the Application in the Khloe’s profile.  Variations:   1. Capabilities:    1. There are sufficient resources available at FLUS Sinks, so there is no need to check whether the picked FLUS Sink has the required real-time multi-rate transcoding capabilities.    2. Available FLUS Sinks might have limited capabilities. The server application must find a Sink capable of running the transcoding session.    3. Required FLUS Sink resources vary based on the number and sophistication of rich features and the processing time defined by the user’s profile. 2. Server or Device Application    1. The UE’s Application is responsible for setting up the FLUS and NBMP session, as well as managing the audience codecs-rates profile.    2. The UE’s Application is responsible for setting up the FLUS session. The Sever Application is responsible for setting up the NBMP session. |
| **Categorization** |
| **Delivery: On-demand Streaming**  **Device: Any device connected to the cell network** |
| **Preconditions** |
| 1. On the device:    1. A 3GPP supported encoder is installed.    2. UE’s Application is installed which supports NBMP Source functionalities.    3. A 3GPP FLUS Source is installed. 2. On the network,    1. One or more FLUS Sinks are installed that one or more of them supports NBMP Workflow Manager functionality.    2. One or more FLUS Sinks are installed, that run various instances of decoding/encoding or transcoding, and the decoder/encoder/transcoder are described as NBMP Functions in an NBMP repository.    3. A FLUS Sink may have limited capabilities, i.e. the codecs it supports and/or a limited number of concurrent encodings/transcodings.    4. A FLUS Sink’s load may vary dynamically due to the other parallel network processing sessions.    5. Network storage is available to store the encoded content for time-shifted streaming. |
| **Requirements in terms of Capabilities and QoS/QoE Considerations** |
| * Capabilities   + Discovering of Sink’s network processing capabilities if needed   + Setting the FLUS and NBMP sessions   + Start the Sessions   + Change the workflow by adding more transcoders if needed during the session   + Real-time transcoding and packaging to different codecs and different bit-rates   + Offloading an originally selected sink to a new sink to handle additional needed processing that cannot be accommodated by the original sink * KPI   + Supporting the use-case   + Minimum extension of FLUS and NBMP Standard, preferably none. |
| **Feasibility and Industry Practices** |
|  |
| **Nominal Cost Analysis** |
| The cost of service increase linearly with the number of ingests.  The cost of service increase less than linearly with the number of download streaming clients as the encoding and caching requirement will be common with a large number of viewers. |
| **Benefits and Impact** |
| The multi-rate encoding is not possible always on the sender device or very battery consuming. Even the use of multiple FLUS sessions is not efficient.  The multi-rate encoding for a various number of devices in every FLUS sink might not be feasible depending on the availability of resources on Sink.  Even a capable Sink may not have all its resources available due to its load at the session start time.  Many enhancement and rich features cannot be performed and/or added on the device. |
| **Potential Technical Requirements** |
|  |
| **Potential Standardization Status and Needs** |
| 1. Discovery of dynamic processing capabilities of FLUS Sink:    1. Available hardware resources    2. Support for NBMP Workflow instantiation and management    3. Built-in (hardware-assisted and/or software-optimized) encoders/transcoders 2. Start and management of FLUS and NBMP Session by the UE’s Application via FLUS Control Source and NBMP Source. 3. Start and management of NBMP session by Server Application aligned with start and management of FLUS session by UE’s Application |

# Assumptions and requirements

## NBMP in the current FLUS architecture

This clause describes the use of NBMP in the current FLUS architecture.

* + 1. Mapping between system components

Based on the functional definitions of different NBMP system components, the following mapping between NBMP components and FLUS system components can be made, as shown in Table 1.

**Table 1. Mapping from NBMP components to FLUS components**

|  |  |
| --- | --- |
| System Component | Description |
| NBMP Source | * FLUS control source inside FLUS source: The FLUS source uses F-C interface to setup workflows at the FLUS sink as described in TS 26.238 and TR 26.939. F-C uses Workflow API as defined in ISO/IEC 23090-8 for this procedure * Non-colocated Control Source: The Non-colocated control source outside the FLUS source, described in clause A.1.3 of 3GPP TS 26.238, can take the role of NBMP Source and use F-C interface to configure workflow at the FLUS sink. * Control Source inside Remote Control Device: The control source inside the control device described in TS 26.238 clause A.2.2 can take the role of NBMP Source and use the F-C interface to configure workflow at the FLUS sink. * Control Source inside a Remote Controller co-located with a Control Sink sub-function: The control source inside the remote controller co-located with a control sink sub-function can act as an NBMP source and use F-C interface to configure workflow at the FLUS sink. |
| NBMP Media Source | * Media source inside FLUS source: The media source inside FLUS source assumes the role of NBMP Media Source. |
| NBMP Workflow Manager | * Control Sink inside FLUS Sink: The control sink inside FLUS sink can take the role of NBMP workflow manager.   The control sink sets up post-processing and distribution functions as described in TR 26.939 clause A.1 and A.2 in one or more NBMP media processing entities   * AF in operator core: An AF inside operator core can assume the role of NBMP workflow manager and receive a workflow description from an NBMP source (e.g., a FLUS sink) * 3rd party server outside the operator domain: A server outside the operator domain can assume the role of NBMP workflow manager and receive a workflow description from an NBMP Source (e.g., a FLUS sink) |
| NBMP Task | * Media Sink inside FLUS Sink: The media sinks inside FLUS sinks assumes the role of NBMP Task to ingest content from FLUS source using the F-U interface.   The ingested content can then be sent to post-processing and distribution functions in other NBMP Tasks in the workflow setup by the workflow manager. |
| NBMP Function Repository | None |

* + 1. API considerations

Table 2 shows the mapping between different NBMP API and FLUS API.

**Table 2. Mapping NBMP API to FLUS API**

|  |  |
| --- | --- |
| API | Description |
| Workflow API | * Uplink Streaming Control Interface as defined in TS 26.238 clause 7 is to be used for NBMP Workflow API |
| Task API | * Currently out of the scope of FLUS specification |
| Function Discovery API | * Currently out of the scope of FLUS specification |

* + 1. Procedures

In the case in which NBMP sessions are managed through FLUS control plane, the following procedures defined in TS 26238 can be used or updated.

* Workflow Manager Discovery: Clause 7.2 of TS 26.238 describes the discovery procedure of FLUS sink. This procedure can be used to discover a FLUS sink that can act as an NBMP workflow manager as described in clause 4 of this contribution.
* Workflow Manager Capability Retrieval: Clause 7.3 of TS 26.238 describes capability retrieval of a FLUS sink. This procedure can be used to retrieve workflow management capabilities at the FLUS sink as described in clause 4 of this contribution.
* Workflow Establishment: Clause 7.5 of TS 26.238 is used for setting FLUS sessions between FLUS source and FLUS sink. This procedure can be used for setting up a workflow session at the FLUS sink.
* Workflow Termination: Clause 7.6 of TS 26.238 is used for terminating FLUS sessions between FLUS source and FLUS sink. This procedure can be used for terminating a workflow session at the FLUS sink.
* Workflow Modification: Clauses 7.4.2 of TS 26.238 is used for modification of FLUS sessions between FLUS source and FLUS sink. This procedure can be used for modifying a workflow at the FLUS sink.
* Workflow Retrieval: Clauses 7.4.1 of TS 26.238 is used for retrieval of a FLUS session between FLUS source and FLUS sink. This procedure can be used for retrieving a workflow at the FLUS sink.

# Combined FLUS and NBMP Architecture

## General Architecture

Figure 1 shows the general architecture of FLUS with Application on UE (UA) as well as Application of the Application Server (EA):

UE

Sink

FLUS Control Source

FLUS

Media Source

Application (UA)

FLUS Control Sink

FLUS

Media Sink

External Application Server

F-C

F-U

F5

F8

F2

Application (EA)

F3

F1

F7

**Figure 1: Genral Flus architecture including user application and external application server**

We consider various deployment scenarios, starting with a case that requires the least number of standard-compliant APIs to cases requiring more standard-compliant APIs.

## NBMP in the Application Server (All-AP)

This scenario is shown in Figure 2. In this case, NBMP Source, Workflow Manager, and MPEs are located in the Application Server.

UE

Sink

FLUS Control Source

FLUS

Media Source

Application (UA)

FLUS Control Sink

FLUS

Media Sink

External Application Server

F-C

F-U

F5

F8

F2

NBMP Source

NBMP Workflow Manager

Application (EA)

Application Server (MPE)

Origin Server

(NBMP Media Sink)

N3

N4

F3

F1

N1

N2

F7

Figure 2: NBMP in Application Server

* + 1. Call flow

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:

1. UE Application (UA) makes a request through F8 to Application (EA) to start a live session.
2. EA requests the list of FLUS Sinks from a Sink Discovery Server (not shown).
3. Sink Discovery Server responds to EA’s request.
4. EA picks a Sink and finds its FLUS Media Sink address.
5. EA retrieves the user profile and identifies the resources needed to run the service.
6. EA requests NBMP Source to start an NBMP Workflow.
7. NBMP Source builds the WDD, and requests NBMP Workflow Manager to instantiate the Workflow.
8. NBMP Workflow Manager discovers various MPEs and finds enough number of MPEs to run the workflow (Note: SA6 discovery may be considered to address this functionality)
9. NBMP Workflow Manager instantiates the workflow.
10. NBMP Workflow responds to NBMP Source with updated WDD.
11. NBMP Source acknowledge workflow instantiation to EA.
12. EA responds to UA with Control Sink and Media Sink information.
13. UA requests FLUS Control Source to establish the FLUS session.
14. FLUS Control Source establishes the FLUS session and acknowledges UA.
15. UA start ingesting the content.
16. The session runs
17. UA requests EA to end the session.
18. EA request NBMP Source to stop the NBMP workflow.
19. NBMP Source acknowledges the stopping of the NBMP session.
20. EA acknowledges UA the stopping of the workflow.
21. UA requests FLUS Control Sink to stop the FLUS session.
    * 1. Interfaces

Table 1 shows the required standard interfaces in this scenario:

Table 1: Required Standard APIs for NBMP in Application Server

|  |  |  |
| --- | --- | --- |
| Standard | FLUS | F-C, F-U, F1 |
| NBMP | N4, F2\* |

* Note: The internal APIs inside green boxes are out of scope of this document.

[\*The FLUS specification currently does not define setting up an output for FLUS Media Sink. To support this scenario, the FLUS specification needs to be extended to either support setting up an output address for FLUS Media Sink, or provide an address for FLUS Media Sink’s output for data retrieval.]

## NBMP in the Application Server, MPE in Sink (MPE-Sink)

This scenario is shown in Figure 3. In this case, NBMP Source and Workflow Manager are located in the Application Server, and MPEs are located in Sinks.

Sink

UE

FLUS Control Source

NBMP/FLUS

Media Source

Application (UA)

FLUS Control Sink

FLUS

Media Sink

External Application Server

F-C

F-U

F5

F8

F2

NBMP Source

NBMP Workflow Manager

Application (EA)

Application Server (MPE)

Origin Server

(NBMP Media Sink)

N3

N4

F3

F1

N1

N2

F7

F11

Figure 3: NBMP in Application Server, MPE in Sink

* + 1. Call flow

There are two possibilities of discovering MPE capabilities:

1. EA discovers MPE capabilities through FLUS Control Sink (F1)
2. EA discovers MPE’s location through FLUS Control Sink (F1) and discover the MPE capabilities through N3.

The call flows for both cases are shown below.

#### Through F1 (MPE-Sink-F1)

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:



The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:

1. UE Application (UA) makes a request through F8 to Application (EA) to start a live session.
2. EA retrieves the user profile and identifies the resources needed to run the service.
3. EA requests the list of FLUS Sinks and their capabilities from Sink Discovery Server (not shown).
4. EA picks a Sink that can run the workflow in its MPE and find its MPE address and MPE APIs in its capabilities.
5. EA requests NBMP Source to start an NBMP Workflow with FLUS Media Sink Address.
6. NBMP Source builds the WDD, and requests NBMP Workflow Manager to instantiate the Workflow, with the assigned MPE.
7. NBMP Workflow Manager instantiates the workflow in the assigned MPE.
8. NBMP Workflow responds to NBMP Source with updated WDD.
9. NBMP Source acknowledges workflow instantiation to EA.
10. EA responds to UA with Control Sink and Media Sink information.
11. UA requests FLUS Control Source to establish the FLUS session
12. FLUS Control Source establishes the FLUS session and acknowledges UA
13. UA start ingesting the content.
14. The session runs
15. UA requests EA to end the session.
16. EA request NBMP Source to stop the NBMP workflow.
17. NBMP Source acknowledges the stopping of the NBMP session.
18. EA acknowledges UA the stopping of the workflow.
19. UA requests FLUS Control Sink to stop the FLUS session.

#### Through N3 (MPE-Sink-N3)

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:



* + 1. Interfaces

Table 2 shows the required standard interfaces in this scenario:

Table 2: Required Standard APIs for NBMP in Application Server, MPE in Sink

|  |  |  |
| --- | --- | --- |
| Standard | FLUS | F-C, F-U, F1 |
| NBMP | N4, N3\* |

* \*May be a closed API implemented by Application provider-operator agreement.
* Note: The internal APIs inside green boxes are out of scope of this document.

## NBMP Source in the Application Server, NBMP Workflow Manager, and MPE in Sink (WM-MPE-Sink)

This scenario is shown in Figure 4.

Sink

UE

FLUS Control Source

NBMP/FLUS

Media Source

Application (UA)

FLUS Control Sink

FLUS

Media Sink

External Application Server

F-C

F-U

F5

F8

F11

F2

NBMP Source

NBMP Workflow Manager

Application (EA)

Application Server (MPE)

Origin Server

(NBMP Media Sink)

N3

N4

F3

F1

N1

N2

F7

Figure 4: NBMP Source in Application Server, NBMP Workflow Manager and MPE in Sink

* + 1. Workflow

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:



Same variations (discovering the entire MPE capabilities through FLUS Control Sink vs discovering MPE location through FLUS Control Sink) are possible here.

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:

1. UE Application (UA) makes a request through F8 to Application (EA) to start a live session.
2. EA retrieves the user profile and identifies the resources needed to run the service.
3. EA requests the list of FLUS Sinks and their capabilities from Sink Discovery Server (not shown).
4. EA picks a Sink that can run the workflow in its MPE and find its NBMP Workflow Manager and Media Sink address in the Sink capabilities.
5. EA requests NBMP Source to start an NBMP Workflow with FLUS Media Sink Address.
6. NBMP Source builds the WDD, and requests NBMP Workflow Manager to instantiate the Workflow, with the assigned MPE.
7. NBMP Workflow Manager instantiates the workflow in the assigned MPE.
8. NBMP Workflow responds to NBMP Source with updated WDD.
9. NBMP Source acknowledges workflow instantiation to EA.
10. EA responds to UA with Control Sink and Media Sink information.
11. UA requests FLUS Control Source to establish the FLUS session
12. FLUS Control Source establishes the FLUS session and acknowledges UA
13. UA start ingesting the content.
14. The session runs
15. UA requests EA to end the session.
16. EA request NBMP Source to stop the NBMP workflow.
17. NBMP Source acknowledges the stopping of the NBMP session.
18. EA acknowledges the stopping of the workflow to the UA.
19. UA requests FLUS Control Sink to stop the FLUS session.
    * 1. Interfaces

Table 3 shows the required standard interfaces in this scenario:

Table 3: NBMP Source in Application Server, NBMP Workflow Manager and MPE in Sink

|  |  |  |
| --- | --- | --- |
| Standard | FLUS | F-C, F-U, F1 |
| NBMP | N2, N4 |

* Note: The internal APIs inside green boxes are out of scope of this document.

## NBMP Source in the FLUS Control Source, NBMP Workflow Manager and MPE in Sink (NBMPSource-FLUSSource)

This scenario is shown in Figure 5.

Sink

UE

FLUS Control Source

NBMP/FLUS

Media Source

Application (UA)

FLUS Control Sink

FLUS

Media Sink

External Application Server

F-C

F-U

F5

F8

F11

F2

NBMP Source

NBMP Workflow Manager

Application (EA)

Application Server (MPE)

Origin Server

(NBMP Media Sink)

N3

N4

F3

F1

F7

Figure 5: NBMP Source in FLUS Control Source, NBMP Workflow Manager and MPE in Sink

* + 1. Workflow

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:



In this case, the UE can either

1. provide NBMP Workflow’s location, so that NBMP Source can discover the MPE capabilities through that API as well as establishing the workflow, or
2. discover capabilities of Sink during the sink discovery process, so that NBMP Source doesn’t need to discover capabilities of a sink’s MPE in a different step.

The steps of establishing, operating, and tearing down a FLUS-NBMP session are as the following:

1. UE Application (UA) makes a request through F8 to Application (EA) to start a live session.
2. EA retrieves the user profile and identifies the resources needed to run the service.
3. EA requests the list of FLUS Sinks and their capabilities from Sink Discovery Server (not shown).
4. EA picks a Sink that can run the workflow in its MPE and find its NBMP Workflow Manager and Media Sink address in the Sink capabilities.
5. EA responds to UE with the NBMP Workflow Manager's full URL or a relative URL through FLUS Control Sink.
6. UA requests FLUS Control Source to establish the FLUS session
7. FLUS Control Source establishes the FLUS session and acknowledges UA
8. EA requests NBMP Source start the workflow.
9. NBMP Source builds WDD, and requests NBMP Workflow Manager (directly or through FLUS Control Sink) to instantiate the Workflow
10. NBMP Workflow Manager instantiates the workflow in the MPE.
11. NBMP Workflow responds to NBMP Source with updated WDD.
12. NBMP Source acknowledges workflow instantiation to UA.
13. UA start ingesting the content.
14. The session runs
15. UA requests FLUS Control Source to end the session.
16. NBMP Source request NBMP Workflow Manager to stop the workflow
17. FLUS Control Source request to end the FLUS Session.
    * 1. Interfaces

Table 4 shows the required standard interfaces in this scenario:

Table 4: NBMP Source in FLUS Control Source, NBMP Workflow Manager and MPE in Sink

|  |  |  |
| --- | --- | --- |
| Standard | FLUS | F-C\*, F-U, F1 |
| NBMP | N4 |

* \*With the support of NBMP Workflow Manager APIs
* Note: The internal APIs inside green boxes are out of scope of this document.

## Summary of the deployment scenarios

Table 5 shows a summary of deployment scenarios.

Table 5: Summary of the deployment scenarios

|  |  |  |
| --- | --- | --- |
| Scenario | Standard | API |
| NBMP in Application Server | FLUS | F-C, F-U, F1 |
| NBMP | N4, F2\* |
| NBMP in Application Server, MPE in Sink | FLUS | F-C, F-U, F1 |
| NBMP | N4, N3\*\* |
| NBMP Source in Application Server, NBMP Workflow Manager and MPE in Sink | FLUS | F-C, F-U, F1 |
| NBMP | N2, N4 |
| NBMP Source in FLUS Control Source, NBMP Workflow Manager and MPE in Sink | FLUS | F-C\*\*\*, F-U, F1 |
| NBMP | N4 |

* \*Need an extension to FLUS spec to define output mechanism for FLUS Media Sink
* \*\*May be a closed API implemented by Application provider-operator agreement.
* \*\*\*With the support of NBMP Workflow Manager APIs

## Common Standard APIs requirement

Among all cases, to select a Sink among various sinks, it is required to discover and retrieve the Sink’s MPE capabilities through F1 or F-C in scenarios 2,3, and 4.

## Architecture variation between two use-cases

The above deployment scenarios address the live streaming use-case. For the on-demand use case, a FIFO buffer (such as storage) can be added between FLUS Media Sink and MPE.

# Gap Analysis

## WM-MPE-Sink

This section provide a gap analysis for the deployment scenario of section ‎4.4, in which NBMP Workflow Manager and MPE are implemented in FLUS Sink.

* + 1. Mapping call flow to the standard APIs

The call flow presented in section ‎4.4 is mapped to the FLUS and NBMP APIs in the following table:

Table 1 Mapping call flow to FLUS and NBMP APIs

|  |  |
| --- | --- |
| Call flow step | Support in FLUS or NBMP |
| 1. UE Application (UA) makes a request through F8 to Application (EA) to start a live session. | Out of scope (optional and application dependent.) |
| 1. EA retrieves the user profile and identifies the resources needed to run the service. | Out of scope (optional and application dependent.) |
| 1. EA requests the list of FLUS Sinks and their capabilities from Sink Discovery Server (not shown). | Supported by FLUS discovery API from the FLUS discovery server. |
| 1. EA picks a Sink that can run the workflow in its MPE and find its NBMP Workflow Manager and Media Sink address in the Sink capabilities. | Partially supported by FLUS.  The EA discovers locations and optionally the capabilities of each sink. The sink can list the NBMP Workflow Manager identifier (URI) and optionally the location (URL) in the Sink capabilities. |
| 1. EA requests NBMP Source to start an NBMP Workflow with FLUS Media Sink Address. | Out of scope (Internal to application). |
| 1. NBMP Source builds the WDD, and requests NBMP Workflow Manager to instantiate the Workflow, with the assigned MPE. | Supported by NBMP spec. |
| 1. NBMP Workflow Manager instantiates the workflow in the assigned MPE. | Supported by NBMP spec/ the exact API is MNO specific. |
| 1. NBMP Workflow responds to NBMP Source with updated WDD. | Supported by NBMP spec. |
| 1. NBMP Source acknowledges workflow instantiation to EA. | Out of scope (Internal to application). |
| 1. EA responds to UA with Control Sink and Media Sink information. | Out of scope (Internal to application). |
| 1. UA requests FLUS Control Source to establish the FLUS session | Out of scope (Internal to application). |
| 1. FLUS Control Source establishes the FLUS session and acknowledges UA | Supported by FLUS. |
| 1. UA start ingesting the content. |  |
| 1. The session runs |  |
| 1. UA requests EA to end the session. |  |
| 1. EA request NBMP Source to stop the NBMP workflow. |  |
| 1. NBMP Source acknowledges the stopping of the NBMP session. |  |
| 1. EA acknowledges the stopping of the workflow to the UA. |  |
| 1. UA requests FLUS Control Sink to stop the FLUS session. |  |

* + 1. Shortcomings

The capability signaling of FLUS Sink is defined by the following table (TS26.236):

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.. | { “scheme” : urn:vnd:xzy:capability-name, “location” : “<http://vnd.com/xzy/capability-name>”  } |

As indicated in the above table, the support of the NMBP can be signalled using the “scheme” item. The optional “location” provides the description for the capability to be retrieved. However, to directly access the NBMP Workflow Manager (WM), the location of WM must be signaled. Therefore, we need to either:

1. Add the URL location of WM to the description
2. Add another item in the capabilities array item for the actual address:

An Example for adding the support for NBMP Workflow Manager, with a description of NBMP Workflow Manager:

{ “scheme” : “urn:mpeg:mpegi:nbmp:workflowmanager: 2020”,

“location”: “http://vnd.com/xzy/nbmpwm\_description.json”

}

The document nbmpwm\_description shall contain the description information about the WM. Currently, the ISO/IEC 23090-8 doesn’t define a description for WM. The MPEG NBMP TuC defines a description of MPE, i.e. MPE capabilities. Therefore, either

1. Solution 1: we need to create a standard object for NBMP Workflow description that have the WM address such as:

{

“url”: “<http://10.20.30.40>”,

“name”: “vnd-workflow-manager-1”,

“description”: “the workflow manager provided by VND, version 1, 2021”,

…..

}

1. Or, solution 2: we need to add a new item in the capabilities:

{ “scheme” : “urn:mpeg:mpegi:nbmp:workflowmanager: 2020”,

“location”: “http://vnd.com/xzy/nbmpwm\_description.json”,

“url”: “http://10.20.30.40”

}

## NBMPSource-FLUSSource

This section provide a gap analysis for the deployment scenario of section ‎4.5, in which NBMP Workflow Manager and MPE are implemented in FLUS Sink.

* + 1. Mapping call flow to the standard APIs

The call flow presented in section ‎4.5 is mapped to the FLUS and NBMP APIs in the following table:

Table 2 Mapping call flow to FLUS and NBMP APIs

|  |  |
| --- | --- |
| Call flow step | Support in FLUS or NBMP |
| 1. UE Application (UA) makes a request through F8 to Application (EA) to start a live session. | Out of scope (optional and application dependent.) |
| 1. EA retrieves the user profile and identifies the resources needed to run the service. | Out of scope (optional and application dependent.) |
| 1. EA requests the list of FLUS Sinks and their capabilities from Sink Discovery Server (not shown). | Supported by FLUS discovery API from the FLUS discovery server. |
| 1. EA picks a Sink that can run the workflow in its MPE and find its NBMP Workflow Manager and Media Sink address in the Sink capabilities. | Partially supported by FLUS.  The EA discovers locations and optionally the capabilities of each sink. The sink can list the NBMP Workflow Manager identifier (URI) and optionally the location (URL) in the Sink capabilities. |
| 1. EA responds to UE with the NBMP Workflow Manager's full URL or a relative URL through FLUS Control Sink. | Out of scope (Internal to application). |
| 1. UA requests FLUS Control Source to establish the FLUS session | Out of scope (Internal to application). |
| 1. FLUS Control Source establishes the FLUS session and acknowledges UA | Supported by FLUS spec. |
| 1. EA requests NBMP Source start the workflow. | Out of scope (Internal to application). |
| 1. NBMP Source builds WDD, and requests NBMP Workflow Manager (directly or through FLUS Control Sink) to instantiate the Workflow | Directly: Supported by NBMP Spec  Through FLUS sink: Supported by FLUS spec |
| 1. NBMP Workflow Manager instantiates the workflow in the MPE. | Out of scope (Internal to application). |
| 1. NBMP Workflow responds to NBMP Source with updated WDD. | Directly: Supported by NBMP Spec  Through FLUS sink: Partially supported by FLUS Spec. |
| 1. NBMP Source acknowledges workflow instantiation to UA. | Out of scope (Internal to application). |
| 1. UA start ingesting the content. |  |
| 1. The session runs |  |
| 1. UA requests FLUS Control Source to end the session. |  |
| 1. NBMP Source request NBMP Workflow Manager to stop the workflow |  |
| 1. FLUS Control Source request to end the FLUS Session. |  |

The sink capability discovery of the FLUS spec is adequate for the discovery of NBMP WM support by FLUS Sink.

The communication of the WDD or the URL is achieved through the Sink resource:

Table 5.3.6-1: List of FLUS Sink Configuration properties

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Name** | **Property Description** | **C I** | **C O** | **G I** | **G O** | **U I** | **U O** | **T I** |
| 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.  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 |  |  |

The ‘url’ item provides the location of NBMP WM. Therefore, the NBMP Source can interact with NBMP WM directly. However, if NBMP Source is expected to interact with NBMP WM through FLUS F-C link, then sending WDD and receiving it possible in the NBMP Workflow API’s synchronous mode only. The asynchronous mode may suboptimally work if the response includes the NBMP WM URL, which means that the retrieval of the WDD must be direct. Also, the wait-time for retrieving the WDD is not provided in this case.

For complete support of NBMP Workflow API asynchronous mode, the Sink configuration needs to be extended to include HTTP headers in the response in addition to the already included resource (WDD).

# QoS mapping

## General principles and considerations

*TBD*

## QoS mapping example 1

*TBD*

## QoS mapping example 2

*TBD*

## Gap analysis

*TBD*

# Potential solutions

*TBD*

## NBMP extensions

*TBD*

## Flus extensions

In order to address the above short commings the following extensions should be added to TS26.238.

* + 1. Location signaling in Sink Resource

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 capability URL, from which the capability can be accessed. The API for the resource is defined by the scheme URN. | { “scheme” : urn:vnd:xzy:capability-name, “location” : “<http://vnd.com/xzy/capability-name>”,  “url”: “<http://vnd.com/xzy/capability-url>”  } |

* + 1. Header response in FLUS Sink properties

Table 5.3.6-1: List of FLUS Sink Configuration properties

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Name** | **Property Description** | **C I** | **C O** | **G I** | **G O** | **U I** | **U O** | **T I** |
| 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.  - header-responses: the response headers associated with the media processing document, when an update to the media processing document is provided in a request’s response. The format of header resonses are defined by the media processing description document format.  - url: the URL to the media processing document.  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. The header-responses include the header-responses of MPEG NBMP Workflow API as defined in [17]. | **O** | **O** |  | O | O |  |  |

Note: We will investigate combining the header responses of both calls (FLUS SINK Control respose and NBMP Workflow Manager response) in a single set of headers (i.e. FLUS Sink Control response headers include the NBMP workflow manager response header) as an alternative solution to the above addition.

# Areas of further standardization

*TBD*

# References

[1] 3GPP TS 26.238, 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Uplink Streaming (Release 16), V16.4.2

[2] 3GPP TR 26.939, 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Guidelines on the Framework for Live Uplink Streaming (FLUS); (Release 16), V16.1.0