 ISO/IEC JTC 1/SC 29/WG 03 N0714

**ISO/IEC JTC 1/SC 29/WG 03  
MPEG Systems   
Convenorship: KATS (Korea, Republic of)**

**Document type:** Output Document

**Title:** **NBMP reference software and conformance framework**

**Status:** Approved

**Date of document:** 2021-10-28

**Source:** ISO/IEC JTC 1/SC 29/WG 03

**Expected action:** None

**Action due date:** None

**No. of pages:** 6 (with cover page)

**Email of Convenor:** young.L @ samsung . com

**Committee URL:** <https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg3>

**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 03 MPEG SYSTEMS**

**ISO/IEC JTC 1/SC 29/WG 03 N0714**

**October 2022, Mainz**

|  |  |
| --- | --- |
| **Title** | **NBMP reference software and conformance framework** |
| **Source** | **WG 03, MPEG Systems** |
| **Status** | **Approved** |
| **Serial Number** | **21993** |
| **Editor** | **Iraj Sodagar** |

Introduction

This document defines a framework for NBMP conformance software and test vectors.

Functional conformance

A quick review of the “shall”s of the spec for various entities and APIs are summarized in Table 1:

Table 1 — Various entities’ conformance test

|  |  |  |
| --- | --- | --- |
| Test Case | Testing subject | Conformance requirements |
| 1 | Documents format:  FD, WD, TD, and MD | Schema validation |
| 2 | Documents restrictions including the descriptor semantic requirements i.e. specific values or dependent conditions to other parameters. | Absence of some parameters in Descriptions or each descriptor restrictions. |
| 3 | Workflow lifecycle on task lifecycle | Changing the state of the corresponding tasks by the workflow manager |
| 4 | Workflow API  Task API  Function API  MPE API | Workflow id, HTTP status codes, the inclusion of the body in response |
| 5 | NBMP client  Workflow Manager  Task  Function Repository  MPE  ‘shall’s | The specific behavior of each entity depending on the interaction with others and the internal state of the entity. |

NBMP documents conformance tool

Figure 2 shows the conformance tool for checking the conformance of documents (test cases 1 and 2).

Semantic Verifier

*Schema*

*Input Document*

*Report*



*Rules*



*Report*

*Optional: Web service*

Schema Verifier

Figure 2 — NBMP document conformance tool architecture

As shown in Figure 2, an input document conformance is verified in two steps:

1. Using a relevant schema, its schema is verified.
2. Using a set of rule tables, the semantics are verified.

Each step produces a report outlining which item in the document has failed.

While there are online and off-the-shelf software and services for schema verifiers, the semantic verifier may need a custom implementation.

The following tasks may need to be performed to develop the conformance tool:

1. Derive a more constrained schema from the schema-definition.json for each of WD, TD, FD, and MD.
2. Implement additional rules and possible semantic constraints as part of the derived schema.
3. Develop the rule tables and
   1. Investigate the use of out-of-box software for the semantic verifier.
   2. Otherwise, develop a simple code for the semantic verifier.
4. Optionally develop a web-based framework to offer the conformance tool as a web service.

Further investigation is the available of-the-self software packages that can be used for either verifier is needed.

## Current conformance schemas

The schemas for WD, TD, FD, and MD that apply the constraints defined in the 2nd edition NBMP specification tables 1,3, 5, and 7 are included.

In the tables above, the parameters that are not allowed in the corresponding document are described. We have used the JSON schema and derived variations that apply the same restrictions in these schema file.

## List of conformance schemas

1. nbmp-conformance-function-schema.json
2. nbmp-conformance-tast-schema.json
3. nbmp-conformance-workflow-schema.json
4. nbmp-conformance-mpecapabilities-schema.json

NBMP entities conformance tool

Figure 3 shows the conformance tool for checking the conformance of NBMP entities (Test cases 3, 4, and 5).

*Function Discovery*

*API*

*Sample functions*

Function Repository

NBMP Entity Verifier



Media Processing Entity (MPE)

*NBMP*

*MPE*

*API*

NBMP Entity Verifier

*Reports/Notifications*

*Task*

*API*

Task

NBMP Entity Verifier



*Task configuration*

*Reports/Notifications*



*MPE configuration*

Workflow

Manager

*Workflow API*

*Task* *API*

NBMP

Entity Verifier

*Function Rep API*

*MPE API*



*Workflow Manager configuration*

**(a)**

**(b)**

**(c)**

**(d)**

*Reports/Notifications*

Figure 3 — NBMP entities verifier: a. Function repository b. MPE c. Task d. Workflow Manager

The conformance of the following features is verified in the conformance scenarios of Figure 3:

1. Create, update, retrieve and delete operations for each APIs
2. Monitoring, reporting, and notifications
3. Change of state between Workflow and tasks
4. The interaction of Workflow Manager with Function Repository, Task, and MPE, due to NBMP client calls

Note that a Workflow Manager implementation may not support all defined APIs. In those cases, the conformance tests would be limited to those which are defined. For instance, if an implementation only supports Workflow API, then the test would be very limited.

In this case, the tests include a set of test cases, each testing one or more aspects of the APIs, functional requirements, or state machine of the entity under test. For each test case, one or more document needs to be created, and the verifier needs to use those documents. The verifier should be able to receive notifications and reports from the entities and cross-check them with the correct results in each test case.

Summary of test cases and test tools

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case | Testing subject | Conformance requirements | Possible conformance tool | Priority |
| 1 | Documents format:  FD, WD, TD, and MD | Schema validation | Any JSON schema validation tools. Online scheme validators are already available | High |
| 2 | Documents restrictions including the descriptor semantic requirements i.e. specific values or dependent conditions to other parameters. | Absence of some parameters in Descriptions or each descriptor restrictions. | A custom software (JSON parser + custom logic) and/or derived schemas | Moderate |
| 3 | Workflow lifecycle on task lifecycle | Changing the state of the corresponding tasks by the workflow manager | Does not exist.  Difficult to check since it depends on the internal state of the workflow manager | Very low |
| 4 | Workflow API  Task API  Function API  MPE API | Workflow id, HTTP status codes, the inclusion of the body in response | Basic HTTP tests are commonly available.  The basic HTTP tests are commonly available and do not need to be provided. The specific requirements depend on the workflow manager's behavior. | Low |
| 5 | NBMP client  Workflow Manager  Task  Function Repository  MPE  ‘shall’s | The specific behavior of each entity depending on the interaction with others and the internal state of the entity. | Does not exist.  The specific requirements often depend on the internal entity's behavior. | Low |

Test vectors

The conformance spec may contain a set of test vectors demonstrating the major features of the spec. The test vectors may include the following list:

1. Examples of FD, WD, TD, and MD. Each example should demonstrate one or more NBMP key features and/or descriptors. A coverage map of the descriptors and parameters should be provided.

1. Examples of function templates. Each function template shall include JSON examples for its configuration and other JSON files.

## Current Test vectors

The candidate test vectors are located at: <http://mpegx.int-evry.fr/software/MPEG/Systems/NBMP/nbmp-test-vectors>

The current test vectors are:

Table 4 — Test vectors

|  |  |  |
| --- | --- | --- |
| Category | Name | Scope |
| FD | FunctionDescription-FIFO.json | FIFO function |
| FD | FunctionDescription-merge.json | Merger function |
| FD | FunctionDescription-split.json | Splitter function |

Coverage Tables

Every ‘shall’ in the spec as well as every descriptor and parameter ‘shall’ must be included in the two coverage tables below. The first coverage table indicates which tools cover testing the shall. The second coverage table indicates which test vectors cover a descriptor and/or parameter. Examples of such tables are shown in Table 4 and Table 5.

Table 4 — Shall coverage

|  |  |  |
| --- | --- | --- |
| clause | topic | Conformance tool |
| 6.1.2 | Priority shall not be present in FD | FD schema |
| 6.1.2 | nonessential shall not be present in FD | FD schema |
| … | … | … |

Table 5 — Descriptors and parameters coverage

|  |  |  |
| --- | --- | --- |
| Clause | Descriptor, object, or parameter | Conformance tool |
| 8.2 | General Descriptor | V1, V2, ….. |
| 8.8 | Startup Descriptor | V28 |
| … | … | … |

Scope

At this stage, we propose that test cases (test cases 1 and 2, section 2) and test vectors (section 7) are considered in the first phase of the NBMP conformance development.