Office des publications
de l'Union européenne

16071971

Interinstitutional Metadata and Formats Committee

Metadata subgroup

"Testing a (preliminary) IMMC schema release"

Date:	30/11/2021
Doc. Version:	0.3

Document Control Information

Settings	Value
Document Title:	Testing a (preliminary) IMMC schema release
Basis	
Doc. Version:	0.3
Date:	30/11/2021

Document Approver(s) and Reviewer(s):

NOTE: All Approvers are required. Records of each approver must be maintained. All Reviewers in the list are considered required unless explicitly listed as Optional.

Name	Role	Action	Date
		<Approve / Review>	

Document history:

Changes to this document are summarized in the following table in reverse chronological order (latest version first).

Revision	Date	Created by	Short Description of Changes
1.1	31/08/2023	Marc VANDERPERREN	Review
1.0	14/08/2023	Marc VANDERPERREN	New version of the document following the EP's comments
0.4	07/02/2022	EP	EP review
0.3	30/11/2021	Marc VANDERPERREN Martin SCHERBAUM	Reworked sections 2 and 3
0.2	19/11/2021	Marc VANDERPERREN Martin SCHERBAUM David LASTOVICKA	Review and formatting
0.1	17/11/2021	Martin SCHERBAUM	Initial version

Contents

1. Introduction

A solution proposed for an IMMC standardisation request can include a preliminary IMMC schema release shared as an archive that can be downloaded from the OP CELLAR through a URL.

This document describes how to test an IMMC schema release.

1.1. Purpose and audience of the document

The members of the IMFC Metadata subgroup involved in the lifecycle of IMMC standardisation requests are:

- The intended audience of this document,
- Invited to comment this document, e.g. with proposal(s) for improvement, as needed on a continuous basis.

2. Definitions and purposes of testing

2.1. IMMC

2.1.1. *What is IMMC?*

IMMC was a committee to decide guidelines for the exchange of metadata between the EU institutions.

IMMC became the name of a series of XML schema definitions (namely IMMC v2 and IMMC v3) which are used to implement metadata exchanges between the EU institutions (and also their contractors or national bodies).

IMMC can be also seen as a framework for formulating metadata exchange contracts. Such contract can be described as: someone sends an IMMC message to someone else for a specific metadata exchange purpose.

2.1.2. *IMMC schema*

The purpose of an **IMMC schema** is to validate an IMMC message against constraints formulated in XML schema notation.

There are different IMMC schema versions in use, i.e. IMMC v2 and IMMC v3. IMMC messages from different schema versions (IMMCv2, IMMCv3) cannot be mixed, as they have different ways of expressing the same things. Separate releases are provided for each schema. XSD files are accessible on URLs, so that they can be directly used for validation.

2.1.3. *IMMC message, syntax & vocabulary, protocol description*

In a nutshell, an **IMMC message** is an IMMC descriptor file in XML format, and is specific to:

- A sender
- A recipient
- A transmission purpose.

Consequently, IMMC is used as a point to point type of communication where every sender knows exactly the recipient(s) of each IMMC message.

In order for every sender and recipient to understand each other, i.e. in the direction “the recipient understands the message from the sender”, they need to speak the same language, on syntax and vocabulary levels:

- The **syntax** corresponds to the IMMC data structures formulated in the IMMC XML schema format (“the IMMC schema”), and
- The **vocabulary** covers all the possible metadata items that can be used in an IMMC exchange.

The syntax and vocabulary show what can be expressed, but do not show what is actually expressed in a specific exchange. The definition of message types, used in specific exchanges within a domain including communication partners, is done by means of the **IMMC protocol description**; this lists all message types that can go from a single sender to a single recipient.

Each **message type** is a “contract” that specifies:

- The sender
- The recipient
- The purpose of the metadata exchange
- The metadata framework (root element, extensions) in use
- The metadata fields used
- The metadata values that are acceptable.

By design, two levels of metadata are defined in an IMMC schema to support any exchange:

- **Core metadata**: this is the set of metadata fields and structures which serves as basis for *all message types* (with some variance on mandatory / optional elements) between *all exchange partners*
- **Extension metadata**: this metadata is used specifically in the scope of the business domain of the sender.

From all the points above, it is important to consider that

- IMMC transmissions are not like email transmission: the sender and recipient are statically fixed by the purpose of transmission at the moment of the definition of the exchange domain by the IMMC transmission protocol description; additional recipients (“CC”) cannot be added dynamically; recipients not included in the contract most probably will not be able to understand the message they would receive.
- IMMC transmissions are not broadcast messages; they are targeted: sender/transmission/purpose; all recipients have to be aware of the specific message type in order to be able to receive and handle it properly

2.1.4. *Implementation in the form of message exchanges*

Based on this contract, i.e. the message type, two partners can do exchanges by means of message instances:

1. The sender generates messages, i.e. message instances, which are valid and compliant with a message type, and
2. The recipient receives messages, validates them against a message type and extracts the metadata information for its further processing.

In a larger understanding (abstracting from the recipient data record in the IMMC messages) a message type “template” could also be sent to recipients. Such template has to be included in the contract.

There are hundreds of message types, i.e. contracts; the handling of the different contracts is basically a handling of dependencies on sets of metadata in given business domains. The **IMMC dataflow inventory** is essential to manage those dependencies. The inventory:

- Is the register of all (known) contracts for the exchange of metadata
- Lists the sender, the recipient(s), and the purpose of a specific message type
- Describes implicitly the domain specific extension in use, and metadata items to be used.

2.1.5. *The types of changes and the impacts of changes*

2.1.5.1. Changes to message types

Two exchange partners may wish to exchange more information; a sender may have more data to transmit.

In this case, there is a need to “enrich” one or more existing business domains and to update the message types accordingly; ultimately contract(s) has(have) to be amended and upon agreement the implementation of the change(s) has(have) to be coordinated between the pair of exchange partners i.e. the sender and the recipient.

An institution can issue an IMMC standardisation request in order to change the relevant metadata, and release a new IMMC schema version.

2.1.5.2. Changes to metadata definitions

Changes at each level of metadata have different impacts:

- **Changes to core metadata** (i.e. the XML Schema for Core Metadata and the XML Schema for Core Metadata Common Extensions) can impact all exchange partners using an IMMC schema, in a specific version and release
- **Changes to extension metadata** impact only partners using the specific extension for exchanging messages.

2.1.5.3. New versions of IMMC schemas

Changes to metadata definitions result in a new IMMC schema version, as a consequence of an IMMC standardisation request issued by an institution using the IMMC protocol. These changes result in an IMMC schema release.

Changes to an IMMC schema can be done in a backward compatible way; this means that message instances which were validated by means of a previous IMMC schema release remain valid for the new IMMC schema release which includes the changes. Backward compatibility is the general baseline of changes.

2.1.5.4. Messages based on new message types

In case:

1. A new feature is added to an IMMC schema version (e.g. for the COMMISSION),
2. The institution sends the message to the desired recipient (e.g. COMMISSION to PARLIAMENT).
3. However the exchange partner (PARLIAMENT) is still relying on an old IMMC schema. They are not able to check for all of the rules.

then this may cause a situation where a message from a sender may not be properly validated and used by a recipient. This situation would be a violation of an established “contract”.

2.2. IMMC schema release testing

"IMMC schema release testing" means in the context of this document[1] the set of activities performed by an institution using the IMMC protocol to inspect the IMMC schema release with the aim to:

- **Test the compatibility** of the (new) IMMC schema release against a previous IMMC schema release,
- **Test the new features**, i.e. changes, of the new IMMC schema release.

The tests shall be performed with a representative sample of IMMC messages, as used in usual operation on IT systems in production.

3. TESTING SCOPE AND TYPES

3.1. Determining the scope of testing and who should test

3.1.1. *Release note*

Upon availability of a preliminary IMMC schema release for testing purposes, check first the release note, which shall indicate the:

- Reference to the standardisation request(s) to which the schema release corresponds
- Changes
- Indication of estimated impacted business stakeholders

3.1.2. *Level 1: IMMC schema version*

When a schema version is changed, only institutions using that schema version may have to test. An institution using another – unmodified - schema version does not have to test.

3.1.3. *Level 2: Core metadata*

Within an IMMC schema version, changes to the:

- Core metadata,
- Core metadata extensions,

impact all users of that IMMC schema version; all business stakeholders shall test as all message types in that series might be affected by the change.

3.1.4. *Level 3: Extension metadata*

Changes on a domain specific extension metadata level impact only message types using this specific extension. Only senders and recipients of such messages shall test. More specifically:

- Change(s) in (an) extension(s) within a folder of an institution, i.e. in a XML schema for the institution 1 extensions to the Core Metadata, require the institution 1 to test the IMMC schema release. Also, any other institution (institution 2 to N) sending to or receiving IMMC messages from institution 1, based on such extension(s), shall test the IMMC schema release.
- In principle, every IMMC message is only allowed to use extensions corresponding to the root element of the message. For an exchange domain, there is usually one specific root element i.e. the transmission request element. Extensions are in use within a transmission request. In case there is a change to this/these extension(s), institutions using the specific transmission request shall test.

[1] OP performs tests as part of the development of the IMMC schema release

3.2. Test type - Schema compatibility test

The IMMC new schema release has usually to be tested to ensure that existing data flows and the systems running them will continue operating without restriction, both on the side of the sender and on the side of the message recipient.

Test type	Schema compatibility test
Test description	Validate against the new IMMC schema a set of representative IMMC messages for an existing dataflow. Validating the IMMC descriptor file suffices.
Expected outcome	All messages shall be validated successfully. If there is one message that does not validate then the test is failed.
Attention point	Although the schema covers all possible cases and combinations of data in its validation scope, real data instances might have additional needs that cannot be tackled on this level of testing, e.g. combinations of values or exclusiveness of business information.
Note about good practice of exchanging messages	At daily operations level, good practices to prevent issues (with transmission / messages that were not validated by the sender) are to: • Perform validation on reception • Reject non-compliant transmissions • Not to handle IMMC messages that cannot be understood.

3.3. Test type - Feature tests

Business stakeholders impacted by new feature(s) of an IMMC schema will want to assess the actual change in the IMMC messages.

Test type	New feature test
Test description	Validate against the new IMMC schema new sample messages provided with the new schema need to be validated. These samples serve as well the purpose of documenting the change i.e. new feature
Expected outcome	All messages shall be validated successfully. If there is one message that does not validate then the test is failed.

Test type	Descriptor change test
Test description	Test the resilience of the IT system to changes in descriptors (messages). Changes to the descriptor(s) to leverage the new features from the new schema require that the concerned IT systems are tested. E.g. in case a new *optional* element is introduced in the schema, sending the element allows to evaluate the resilience of the system.
Expected outcome	Validated resilience of IT system, otherwise assessment of any needed adaptation(s) of IT system(s)

4. Testing: validation approaches

This section describes:

- A list of validation approaches (this list may be extended as needed)
- With steps and means applicable for the test types: schema compatibility test, Feature test.

The descriptions below assume that the IMMC schema files are provided in a ZIP package following the directory structure established for IMMC schema extensions and files

4.1. Prerequisites

These steps are a necessary prerequisite for starting with test activities:

1. Download the new schema version: a new schema version is usually announced together with a URL where the schema package can be obtained through download.
2. Unpack the schema package into a separate file system location using appropriate ZIP tools.

4.2. Message validation using an XML editor

Validate a message against the schema **using an XML editor** that provides an XML validator facility, e.g. XMLspy, Oxygen, …

1. Load the IMMC descriptor to be tested into the XML editor
2. Adapt the xsi:schemaLocation[2] attribute to the location of the unpacked IMMC schema or use the editor's schema assignment functionality to announce the IMMC schema to be used for the validation of this message
3. Validate using the XML editor's validation functionality
4. Assess the outcome

The effort for this test is depending on the number of IMMC messages to be validated, but has to be considered low per message.

4.3. Message validation using a stand-alone validation tool

Validate a message against the schema using a **stand-alone validation tool**, e.g. OP's XMLparser module:

1. Modify the configuration of the validation tool to point to the right schema directory and/or
2. Call the validation tool with an appropriate parametrization, indicating the IMMC message as validation subject and the IMMC schema to be used for validation
3. Assess the outcome

The effort for this test is depending on the number of IMMC messages to be validated, but has to be considered low per message.

[2] The xsi:schemaLocation attribute on the IMMC message's root element has to contain (in that order) the namespace URI of the message's root element and the related IMMC extension transmission XSD file from the location of the unpacked IMMC schema directory, e.g. <schema-directory>/generic/gen_transmission.xsd, separated by a space character.

4.4. Message validation in the concerned IT system

Validate a message against the schema **in the concerned IT system**, usually a testing environment:

1. Use the IT system's upload or configuration mechanism to announce the new IMMC schema version
2. Handle IMMC transmission to be tested in the way the IT system usually does (this procedure should be covered by the IT system's operations manual)
3. Assess the outcome

The effort for this test depends on the complexity and configurability of the IT system and possibly on the number of messages to be tested. The IT system processes an IMMC transmission, thus does **more** than just validating the IMMC message.

4.5. Message validation using an online XML validator

Validate an IMMC message against the schema using an **online XML validator**

1. Announce the IMMC schema to the online validator in the appropriate way
2. Upload the IMMC message to be validated, indicating the schema to be used
3. Request validation
4. Assess the outcome

The effort for this test is depending on the number of IMMC messages to be validated, but has to be considered low per message, depending on the complexity of the online tool in use.