

ETSI TS 129 176 V19.3.0 (2026-01)



TECHNICAL SPECIFICATION

**5G;
IP Multimedia Subsystems (IMS);
Media Function (MF) Services;
Stage 3
(3GPP TS 29.176 version 19.3.0 Release 19)**



Reference

RTS/TSGC-0429176vj30

Keywords

5G

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from the
[ETSI Search & Browse Standards application](#).

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format on [ETSI deliver repository](#).

Users should be aware that the present document may be revised or have its status changed, this information is available in the [Milestones listing](#).

If you find errors in the present document, please send your comments to the relevant service listed under [Committee Support Staff](#).

If you find a security vulnerability in the present document, please report it through our [Coordinated Vulnerability Disclosure \(CVD\)](#) program.

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2026.
All rights reserved.

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the [ETSI IPR online database](#).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found at [3GPP to ETSI numbering cross-referencing](#).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Intellectual Property Rights	2
Legal Notice	2
Modal verbs terminology.....	2
Foreword.....	5
1 Scope	7
2 References	7
3 Definitions, symbols and abbreviations	8
3.1 Definitions	8
3.2 Symbols.....	8
3.3 Abbreviations	8
4 Overview	8
5 Services offered by the MF	9
5.1 Introduction	9
5.2 Nmf_MediaResourceManagement (MRM) Service.....	9
5.2.1 Service Description.....	9
5.2.2 Service Operations	9
5.2.2.1 Introduction.....	9
5.2.2.2 Nmf_MRM_Create Service Operation	10
5.2.2.2.1 General	10
5.2.2.2.2 Creation of a new media context	10
5.2.2.3 Nmf_MRM_Update Service Operation	11
5.2.2.3.1 General	11
5.2.2.3.2 Updating an existing media context	11
5.2.2.4 Nmf_MRM_Delete Service Operation	12
5.2.2.4.1 General	12
5.2.2.4.2 Deleting an existing media context.....	12
6 API Definitions	13
6.1 Nmf_MRM Service API	13
6.1.1 Introduction.....	13
6.1.2 Usage of HTTP	13
6.1.2.1 General	13
6.1.2.2 HTTP standard headers	13
6.1.2.2.1 General	13
6.1.2.2.2 Content type	14
6.1.2.3 HTTP custom headers	14
6.1.3 Resources	14
6.1.3.1 Overview	14
6.1.3.2 Resource: Contexts Collection	15
6.1.3.2.1 Description	15
6.1.3.2.2 Resource Definition.....	15
6.1.3.2.3 Resource Standard Methods	15
6.1.3.2.4 Resource Custom Operations	16
6.1.3.3 Resource: Individual Context.....	17
6.1.3.3.1 Description	17
6.1.3.3.2 Resource Definition.....	17
6.1.3.3.3 Resource Standard Methods	17
6.1.3.3.4 Resource Custom Operations	20
6.1.4 Custom Operations without associated resources	20
6.1.5 Notifications	20
6.1.6 Data Model	20
6.1.6.1 General	20
6.1.6.2 Structured data types	21

6.1.6.2.1	Introduction	21
6.1.6.2.2	Type: MediaContext	21
6.1.6.2.3	Type: TerminationInfo	22
6.1.6.2.4	Type: MediaInfo	23
6.1.6.2.5	Type: DcMedia	26
6.1.6.2.6	Type: ArMedia	28
6.1.6.2.7	Type: Mdc1Info	28
6.1.6.2.8	Type: Mdc2Info	28
6.1.6.2.9	Type: NonDcMedia	29
6.1.6.2.10	Type: AvatarMedia	29
6.1.6.3	Simple data types and enumerations	29
6.1.6.3.1	Introduction	29
6.1.6.3.2	Simple data types	29
6.1.6.3.3	Enumeration: Mdc2Protocol	29
6.1.7	Error Handling	30
6.1.7.1	General	30
6.1.7.2	Protocol Errors	30
6.1.7.3	Application Errors	30
6.1.8	Feature negotiation	30
6.1.9	Security	30
6.1.10	HTTP redirection	31
Annex A (normative):	OpenAPI specification	32
A.1	General	32
A.2	Nmf_MRM API	32
Annex B (informative):	Change history	39
History		40

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

- shall** indicates a mandatory requirement to do something
- shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

- should** indicates a recommendation to do something
- should not** indicates a recommendation not to do something
- may** indicates permission to do something
- need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

- can** indicates that something is possible
- cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

- will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document
- will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document
- might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

might not indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

is (or any other verb in the indicative mood) indicates a statement of fact

is not (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

1 Scope

The present document specifies the stage 3 protocol and data model for the Nmf Service Based Interface. It provides stage 3 protocol definitions and message flows, and specifies the API for each service offered by the Media Function (MF).

The 5G System stage 2 architecture and procedures are specified in 3GPP TS 23.501 [2] and 3GPP TS 23.502 [3].

The IP Multimedia Subsystem (IMS) Data Channel (DC) architecture and procedures are specified in annex AC of 3GPP TS 23.228 [14].

The Technical Realization of the Service Based Architecture and the Principles and Guidelines for Services Definition are specified in 3GPP TS 29.500 [4] and 3GPP TS 29.501 [5].

2 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".
- [2] 3GPP TS 23.501: "System Architecture for the 5G System; Stage 2".
- [3] 3GPP TS 23.502: "Procedures for the 5G System; Stage 2".
- [4] 3GPP TS 29.500: "5G System; Technical Realization of Service Based Architecture; Stage 3".
- [5] 3GPP TS 29.501: "5G System; Principles and Guidelines for Services Definition; Stage 3".
- [6] OpenAPI: "OpenAPI Specification Version 3.0.0", <https://spec.openapis.org/oas/v3.0.0>.
- [7] 3GPP TR 21.900: "Technical Specification Group working methods".
- [8] 3GPP TS 33.501: "Security architecture and procedures for 5G system".
- [9] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".
- [10] 3GPP TS 29.510: "5G System; Network Function Repository Services; Stage 3".
- [11] IETF RFC 7540: "Hypertext Transfer Protocol Version 2 (HTTP/2)".
- [12] IETF RFC 8259: "The JavaScript Object Notation (JSON) Data Interchange Format".
- [13] IETF RFC 7807: "Problem Details for HTTP APIs".
- [14] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".
- [15] IETF RFC 6902: "JavaScript Object Notation (JSON) Patch".
- [16] 3GPP TS 29.571: "5G System; Common Data Types for Service Based Interfaces Stage 3".
- [17] 3GPP TS 29.175: "IP Multimedia Subsystem (IMS) Application Server (AS) Services Stage 3".

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [1], 3GPP TS 23.228 [14] and the following apply. A term defined in the present document or in 3GPP TS 23.228 [14] takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

3.2 Symbols

For the purposes of the present document, the following symbols given in 3GPP TS 23.228 [14] apply:

DC2	Reference point between MF and IMS AS
MDC1	Reference point between MF and DCSF
MDC2	Reference point either MF and DC Application Server or AR Application Server

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1], 3GPP TS 23.228 [14] and the following apply. An abbreviation defined in the present document or 3GPP TS 23.228 [14] takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

AR	Augmented Reality
DTLS	Datagram Transport Layer Security
MF	Media Function
IMS AS	IP Multimedia Subsystem Application Server

4 Overview

Within the IMS DC architecture, the MF offers services to the IMS AS via the Nmf service based interface (see 3GPP TS 23.228 [14]).

Figure 4.1-1 provides the reference model (in service-based interface representation and in reference point representation), with focus on the MF services.

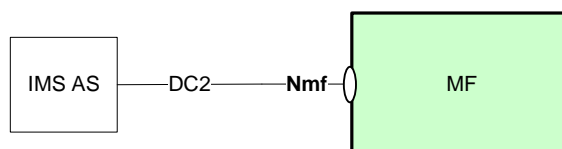


Figure 4.1-1: Reference model – MF

DC2 is the reference point between MF and IMS AS.

5 Services offered by the MF

5.1 Introduction

The MF offers the following services via the Nmf interface:

- Nmf_MediaResourceManagement (MRM) Service

Table 5.1-1 summarizes the corresponding APIs defined for this specification.

Table 5.1-1: API Descriptions

Service Name	Clause	Description	OpenAPI Specification File	apiName	Annex
Nmf_MRM	6.1	Nmf Media Resource Management Service	TS29176_Nmf_MRM.yaml	nmf-mrm	A.2

5.2 Nmf_MediaResourceManagement (MRM) Service

5.2.1 Service Description

The Nmf_MRM service as defined in 3GPP TS 23.228 [14] is provided by the Media Function (MF). This service enables the consumer to create, update and delete media resources. Data Channel (DC) and Augmented Reality (AR) are two capabilities supported by MF.

The media resource represents a media context including one or multiple media terminations. A media termination includes media resources for one or multiple medias on the Mb interface. When a media pass through the MF, there is one termination for the input stream and one termination for the output stream.

If interworking between DC capable UE and non-DC capable UE is supported, the Nmf_MRM shall support to associate the application data channel stream and the video streams, as well as transforming data channel media to video media.

5.2.2 Service Operations

5.2.2.1 Introduction

The Nmf_MRM service supports the following service operation.

Table 5.2.2.1-1: Service operations supported by the Nmf_MRM service

Service Operations	Description	Operation Semantics	Example Consumer(s)
Create	Create a new media context including one or multiple media terminations.	Request/Response	IMS AS
Update	Update the one or multiple existing media resources within a specific media context.	Request/Response	IMS AS
Delete	Delete a specific media context including all the existing terminations and medias.	Request/Response	IMS AS

5.2.2.2 Nmf_MRM_Create Service Operation

5.2.2.2.1 General

The Nmf_MRM_Create service operation is used by an NF service consumer to create a media context including one or multiple terminations and reserve media resources for anchoring one or multiple medias of Mb interface in each termination on MF. The consumer NF may also include application function (e.g. DCSF, DC AS) specification information requested by the application layer to be applied on the media by the MF.

5.2.2.2.2 Creation of a new media context

The NF service consumer shall request a new context by using HTTP method POST with "{apiRoot}/nmf/<apiVersion>/contexts" as resource URI representing the "Contexts Collection", see clause 6.1.3.2.

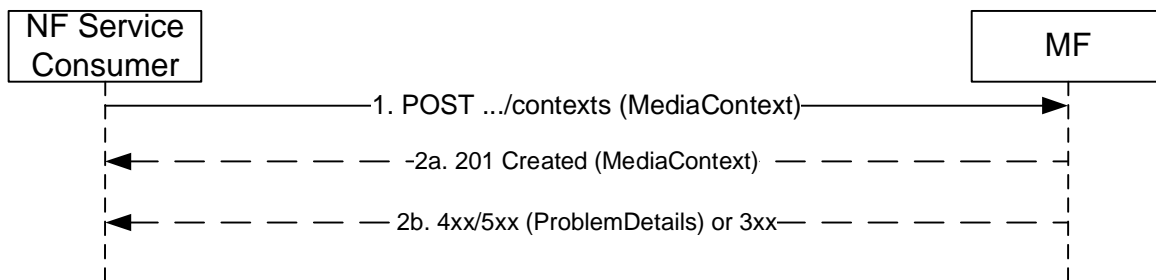


Figure 5.2.2.2.2-1 Creation of a media context

1. The NF Service Consumer shall send a POST request to create an "Individual Context" resource in the MF. The payload body of the POST request shall contain a representation of the individual context resource to be created.

The NF service consumer shall include in the HTTP message body within the MediaContext data type list of termination descriptors in the attribute "terminations". Each termination descriptor shall include list of media stream descriptors in the attribute "medias". Each media stream descriptor shall include:

- media ID in the attribute "mediaId", i.e. a unique identity of the media stream within the media context instance, which is generated by the NF service consumer (e.g. IMS AS) based on implementation;
- remote Mb specifications in the attribute "remoteMbEndpoint", i.e. the media stream IP address and port allocated at the remote endpoint, i.e. remote UE, remote network; and
- media resource description, which includes:
 - 1) media resource type, e.g. DC, AR;
 - 2) if media resource type is set to "DC", the DC media specification in the attribute "dcMedia" within the data type MediaInfo shall be included. The DC media specification shall include:
 - a) media proxy configuration applicable to the media flow in the attribute "mediaProxyConfig";
 - b) data channel mapping and configuration information in the attribute "streams" when originating or terminating data channel media flows on the Mb interface which shall include the SCTP stream Id for the DC, and may include subprotocol, order, maxRetry, maxTime and priority; and
 - c) the remote SCTP endpoint and DTLS endpoint information in the attribute "remoteDcEndpoint" which includes the SCTP port for the Data Channel, the security setup of the DTLS connection, the security certificate fingerprint and the transport layer identity;
 and may include:
 - a) maximum message size in the attribute "maxMessageSize", which represents the maximum size to be expected;

for establishing bootstrap data channel, the DC media specification shall include the remote MDC1 media specification information in the attribute "remoteMdc1Endpoint" within the data type "Mdc1Info" to be applied on the media by the MF and also the replacement HTTP URL for each streamId allocated by the application layer representing the application list (e.g. graphical user interface) offered to the IMS subscriber via the MDC1 interface;

for establishing P2A/A2P application data channel, the DC media specification shall include the remote MDC2 media specification information in the attribute "remoteMdc2Endpoint" within the data type "Mdc2Info" to be applied on the media by the MF;

- 3) if media resource type is set to "AR", the AR media resource description in the attribute "ArMedia" within the data type MediaInfo shall be included;
- 4) if media resource type is set to "VIDEO" or "AUDIO", the video or audio media resource description in the attribute "remoteNonDcMedia" within the data type MediaInfo shall be included; and

may include:

- 1) the avatar media specification in the attribute "avatarMedia" within the data type MediaInfo.

When NF service consumer indicates the need to transcode the data channel media to video media, the data channel media and the video media shall belong to different terminations. Data channel media shall be under the termination as the input stream, and the video media shall be under the termination as the output stream.

2a. Upon the reception of the HTTP POST request, if the request is accepted and no error occur, the MF shall:

- create a new media context;
- assign a media contextId;
- assign a terminationId for each termination descriptor; and
- reserve media resources for each mediaId.

The MF shall include a HTTP Location header to provide the location of a newly created resource (MediaContext) together with the status code 201 indicating the requested resource is created in the response message.

2b. On failure, one of the HTTP status code listed in Table 6.1.3.2.3.1-3 shall be returned. For a 4xx/5xx response, the message body shall contain a ProblemDetails structure with the "cause" attribute set to one of the application errors listed in Table 6.1.3.2.3.1-3.

On redirection, the MF shall include 3xx status code, which shall contain a Location header with an URI pointing to the endpoint to another MF (service) instance.

5.2.2.3 Nmf_MRM_Update Service Operation

5.2.2.3.1 General

The Nmf_MRM_Update service operation is used by an NF service consumer to update one or multiple existing media resources within a specific media context on MF. Terminations and/or Medias can be added, modified, or deleted within an Update request.

5.2.2.3.2 Updating an existing media context

The NF Service Consumer shall modify the context by using HTTP method PATCH with the URI "{apiRoot}/nmf/<apiVersion>/contexts/{contextId}" as resource URI representing the "Individual Context", see clause 6.1.3.3.

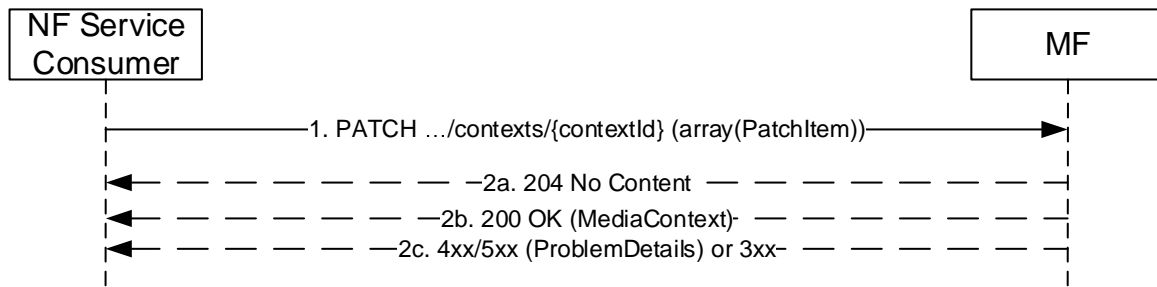


Figure 5.2.2.3.2-1 Update a media context

1. The NF Service Consumer shall send a PATCH request to modify a context resource in the MF. The modification may be for adding, updating or deleting terminations of the existing media context.

For adding, removing or updating a termination in an existing MediaContext, the payload body of the PATCH request shall contain an "add", "remove" or "replace" PATCH operation respectively, with one item of the attribute "terminations" of the MediaContext. The MF shall use the mediaId to check if the operated media exists in the MF.

- 2a. On success, if the change is to delete the existing termination and MF accepts the request change, the MF shall return the status code 204 No Content.
- 2b. On success, if the change is to add a new termination or to update the existing termination and MF accepts the request change, the MF shall return the status code 200 OK. The response shall contain the new resource representation of the resource Individual Context, which includes the added and modified resource or its sub-resource.
- 2c. On failure, one of the HTTP status code listed in Table 6.1.3.3.1-3 indicating the error shall be returned. For a 4xx/5xx response, the message body shall contain a ProblemDetails structure with the "cause" attribute set to one of the application errors listed in Table 6.1.3.3.1-3.

On redirection, the MF shall return 3xx status code, which shall contain a Location header with an URI pointing to the endpoint to another MF (service) instance.

5.2.2.4 Nmf_MRM_Delete Service Operation

5.2.2.4.1 General

The Nmf_MRM_Delete service operation is used by an NF service consumer to delete an existing media context including all existing terminations and medias on MF previously created by itself.

5.2.2.4.2 Deleting an existing media context

The NF Service Consumer shall delete an existing context by using HTTP method DELETE with the URI "{apiRoot}/nmf/<apiVersion>/contexts/{contextId}" as resource URI representing the "Individual Context", see clause 6.1.3.3.

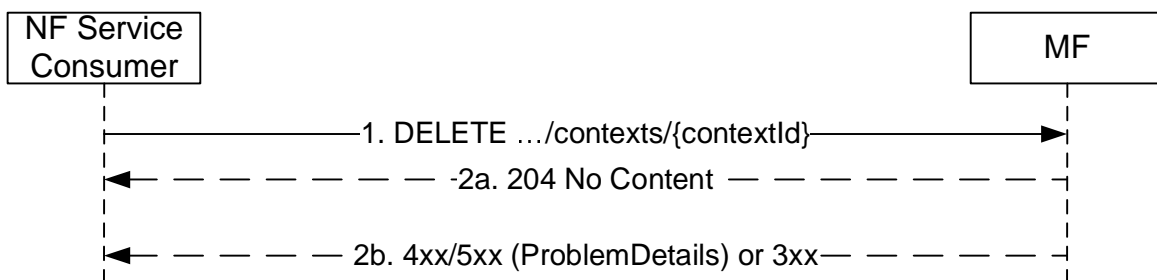


Figure 5.2.2.4.2-1 Delete a media context

1. The NF Service Consumer shall send a DELETE request to delete an existing context resource in the MF.
- 2a. On success, the request is accepted, the MF shall reply with the status code 204 No Content indicating the resource identified by context ID is successfully deleted in the response message.
- 2b. On failure, one of the HTTP status code listed in Table 6.1.3.3.2-3 indicating the error shall be returned. For a 4xx/5xx response, the message body shall contain a ProblemDetails structure with the "cause" attribute set to one of the application errors listed in Table 6.1.3.3.2-3.

On redirection, the MF shall return 3xx status code, which shall contain a Location header with an URI pointing to the endpoint to another MF (service) instance.

6 API Definitions

6.1 Nmf_MRM Service API

6.1.1 Introduction

The Nmf_MRM service shall use the Ndcmf_MRM API.

The API URI of the Nmf_MRM API shall be:

{apiRoot}/<apiName>/<apiVersion>

The request URIs used in HTTP requests from the NF service consumer towards the NF service producer shall have the Resource URI structure defined in clause 4.4.1 of 3GPP TS 29.501 [5], i.e.:

{apiRoot}/<apiName>/<apiVersion>/<apiSpecificResourceUriPart>

with the following components:

- The {apiRoot} shall be set as described in 3GPP TS 29.501 [5].
- The <apiName> shall be "nmf_mrm".
- The <apiVersion> shall be "v1".
- The <apiSpecificResourceUriPart> shall be set as described in clause 6.1.3.

6.1.2 Usage of HTTP

6.1.2.1 General

HTTP/2, IETF RFC 7540 [11], shall be used as specified in clause 5 of 3GPP TS 29.500 [4].

HTTP/2 shall be transported as specified in clause 5.3 of 3GPP TS 29.500 [4].

The OpenAPI [6] specification of HTTP messages and content bodies for the Nmf_MRM API is contained in Annex A.

6.1.2.2 HTTP standard headers

6.1.2.2.1 General

See clause 5.2.2 of 3GPP TS 29.500 [4] for the usage of HTTP standard headers.

6.1.2.2.2 Content type

JSON, IETF RFC 8259 [12], shall be used as content type of the HTTP bodies specified in the present specification as specified in clause 5.4 of 3GPP TS 29.500 [4]. The use of the JSON format shall be signalled by the content type "application/json".

"Problem Details" JSON object shall be used to indicate additional details of the error in a HTTP response body and shall be signalled by the content type "application/problem+json", as defined in IETF RFC 7807 [13].

The use of JSON Patch (IETF RFC 6902 [15]) format in a HTTP request body shall be signaled by the content type "application/json-patch+json".

6.1.2.3 HTTP custom headers

The mandatory HTTP custom header fields specified in clause 5.2.3.2 of 3GPP TS 29.500 [4] shall be supported, and the optional HTTP custom header fields specified in clause 5.2.3.3 of 3GPP TS 29.500 [4] may be supported.

6.1.3 Resources

6.1.3.1 Overview

This clause describes the structure for the Resource URIs and the resources and methods used for the service.

Figure 6.1.3.1-1 depicts the resource URIs structure for the Nmf_MRM API.

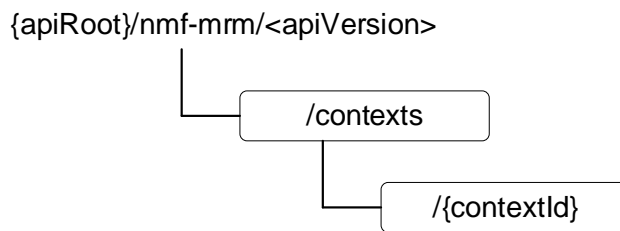


Figure 6.1.3.1-1: Resource URI structure of the Nmf_MRM API

Table 6.1.3.1-1 provides an overview of the resources and applicable HTTP methods.

Table 6.1.3.1-1: Resources and methods overview

Resource purpose/name	Resource URI (relative path after API URI)	HTTP method or custom operation	Description (service operation)
Contexts Collection	/contexts	POST	Creates a new Individual Context resource.
Individual Context	/contexts/{contextId}	PATCH	Updates an existing Individual Context resource identified by {contextId}
		DELETE	Deletes an existing Individual Context resource identified by {contextId}.

6.1.3.2 Resource: Contexts Collection

6.1.3.2.1 Description

The Contexts Collection resource represents a collection of contexts created by NF service consumers of Nmf_MRM service at a given MF. The resource is modelled as Collection resource archetype (see clause C.2 of 3GPP TS 29.501 [2]).

6.1.3.2.2 Resource Definition

Resource URI: **{apiRoot}/nmf_mrm/<apiVersion>/contexts**

This resource shall support the resource URI variables defined in table 6.1.3.2.2-1.

Table 6.1.3.2.2-1: Resource URI variables for this resource

Name	Data type	Definition
apiRoot	string	See clause 6.1.1

6.1.3.2.3 Resource Standard Methods

6.1.3.2.3.1 POST

This method shall support the URI query parameters specified in table 6.1.3.2.3.1-1.

Table 6.1.3.2.3.1-1: URI query parameters supported by the POST method on this resource

Name	Data type	P	Cardinality	Description	Applicability
n/a					

This method shall support the request data structures specified in table 6.1.3.2.3.1-2 and the response data structures and response codes specified in table 6.1.3.2.3.1-3.

Table 6.1.3.2.3.1-2: Data structures supported by the POST Request Body on this resource

Data type	P	Cardinality	Description
MediaContext	M	1	Creates a new Individual Context resource

Table 6.1.3.2.3.1-3: Data structures supported by the POST Response Body on this resource

Data type	P	Cardinality	Response codes	Description
MediaContext	M	1	201 Created	The creation of an Individual Media Context resource is confirmed and a representation of that resource is returned.
RedirectResponse	O	0..1	307 Temporary Redirect	Temporary redirection. (NOTE 2)
RedirectResponse	O	0..1	308 Permanent Redirect	Permanent redirection. (NOTE 2)
ProblemDetails	O	0..1	409 Conflict	Indicates the creation of media context has failed due to application error. The "cause" attribute may be used to indicate one of the following application errors: - MEDIA_ID_CONFLICT, if the media to be created has the identical mediaId with one already created media.
ProblemDetails	O	0..1	500 Server Internal Error	Indicates the creation of media context has failed due to application error. The "cause" attribute may be used to indicate one of the following application errors: - INSUFFICIENT_RESOURCES

NOTE 1: The mandatory HTTP error status code for the POST method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] also apply.

NOTE 2: RedirectResponse may be inserted by an SCP, see clause 6.10.9.1 of 3GPP TS 29.500 [4].

Table 6.1.3.2.3.1-4: Headers supported by the 201 response code on this resource

Name	Data type	P	Cardinality	Description
Location	string	M	1	Contains the URI of the newly created resource, according to the structure: {apiRoot}/nmf-mrm/<apiVersion>/contexts/{contextId}.

Table 6.1.3.2.3.1-5: Headers supported by the 307 Response Code on this resource

Name	Data type	P	Cardinality	Description
Location	string	M	1	An alternative URI of the resource located on an alternative MF (service) instance. Or the same URI, if a request is redirected to the same target resource via a different SCP.
3gpp-Sbi-Target-Nf-Id	string	O	0..1	Identifier of the target MF instance ID towards which the request is redirected.

Table 6.1.3.2.3.1-6: Headers supported by the 308 Response Code on this resource

Name	Data type	P	Cardinality	Description
Location	string	M	1	An alternative URI of the resource located on an alternative MF (service) instance. Or the same URI, if a request is redirected to the same target resource via a different SCP.
3gpp-Sbi-Target-Nf-Id	string	O	0..1	Identifier of the target MF instance ID towards which the request is redirected.

6.1.3.2.4 Resource Custom Operations

None in the release of this specification.

6.1.3.3 Resource: Individual Context

6.1.3.3.1 Description

The Individual Context resource represents an individual context created by the NF service consumers of Nmf_MRM service at a given MF. This resource is modelled as the Document resource archetype (see clause C.1 of 3GPP TS 29.501 [2]).

6.1.3.3.2 Resource Definition

Resource URI: **{apiRoot}/nmf_mrm/<apiVersion>/contexts/{contextId}**

The <apiVersion> shall be set as described in clause 6.1.1.

This resource shall support the resource URI variables defined in table 6.1.3.3.2-1.

Table 6.1.3.3.2-1: Resource URI variables for this resource

Name	Data type	Definition
apiRoot	string	See clause 6.1.1
contextId	string	Identifies an individual context to the Nmf_MRM service.

6.1.3.3.3 Resource Standard Methods

6.1.3.3.3.1 PATCH

This method shall support the URI query parameters specified in table 6.1.3.3.3.1-1.

Table 6.1.3.3.3.1-1: URI query parameters supported by the PATCH method on this resource

Name	Data type	P	Cardinality	Description	Applicability
n/a					

This method shall support the request data structures specified in table 6.1.3.3.3.1-2 and the response data structures and response codes specified in table 6.1.3.3.3.1-3.

Table 6.1.3.3.3.1-2: Data structures supported by the PATCH Request Body on this resource

Data type	P	Cardinality	Description
array(PatchItem)	M	1..N	Document describes the modification(s) to an Individual Context resource.

Table 6.1.3.3.1-3: Data structures supported by the PATCH Response Body on this resource

Data type	P	Cardinality	Response codes	Description
MediaContext	M	1..N	200 OK	Represents a successful update on the media context.
n/a			204 No Content	
RedirectResponse	O	0..1	307 Temporary Redirect	Temporary redirection. (NOTE 2)
RedirectResponse	O	0..1	308 Permanent Redirect	Permanent redirection. (NOTE 2)
ProblemDetails	O	0..1	403 Forbidden	Indicates the modification of media context has failed due to application error. The "cause" attribute may be used to indicate one of the following application errors: - MEDIA_CONNECTION_CHANGED
ProblemDetails	O	0..1	404 Not Found	Indicates the modification of media context has failed due to application error. The "cause" attribute may be used to indicate one of the following application errors: - CONTEXT_NOT_FOUND
ProblemDetails	O	0..1	409 Conflict	Indicates the modification of media context has failed due to application error. The "cause" attribute may be used to indicate one of the following application errors: - MEDIA_ID_CONFLICT, if a new media to be newly created has the identical mediaId already assigned to one created media.
Problem Details	O	0..1	500 Server Internal Error	Indicates the modification of media context has failed due to application error. The "cause" attribute may be used to indicate one of the following application errors: - INSUFFICIENT_RESOURCES
NOTE 1: The mandatory HTTP error status code for the PATCH method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] also apply.				
NOTE 2: RedirectResponse may be inserted by an SCP, see clause 6.10.9.1 of 3GPP TS 29.500 [4].				

Table 6.1.3.3.1-4: Headers supported by the 307 Response Code on this resource

Name	Data type	P	Cardinality	Description
Location	string	M	1	An alternative URI of the resource located on an alternative MF (service) instance. Or the same URI, if a request is redirected to the same target resource via a different SCP.
3gpp-Sbi-Target-Nf-Id	string	O	0..1	Identifier of the target MF instance ID towards which the request is redirected.

Table 6.1.3.3.1-5: Headers supported by the 308 Response Code on this resource

Name	Data type	P	Cardinality	Description
Location	string	M	1	An alternative URI of the resource located on an alternative MF (service) instance. Or the same URI, if a request is redirected to the same target resource via a different SCP.
3gpp-Sbi-Target-Nf-Id	string	O	0..1	Identifier of the target MF instance ID towards which the request is redirected.

6.1.3.3.3.2 DELETE

This method shall support the URI query parameters specified in table 6.1.3.3.3.2-1.

Table 6.1.3.3.3.2-1: URI query parameters supported by the DELETE method on this resource

Name	Data type	P	Cardinality	Description	Applicability
n/a					

This method shall support the request data structures specified in table 6.1.3.3.3.2-2 and the response data structures and response codes specified in table 6.1.3.3.3.2-3.

Table 6.1.3.3.3.2-2: Data structures supported by the DELETE Request Body on this resource

Data type	P	Cardinality	Description
n/a			

Table 6.1.3.3.3.2-3: Data structures supported by the DELETE Response Body on this resource

Data type	P	Cardinality	Response codes	Description
n/a			204 No content	
RedirectResponse	O	0..1	307 Temporary Redirect	Temporary redirection. (NOTE 2)
RedirectResponse	O	0..1	308 Permanent Redirect	Permanent redirection. (NOTE 2)
Problem Details	O	0..1	404 Not Found	Indicates the deletion of the context has failed due to application error. The "cause" attribute may be used to indicate one of the following application errors: - CONTEXT_NOT_FOUND
NOTE 1: The mandatory HTTP error status code for the DELETE method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] also apply.				
NOTE 2: RedirectResponse may be inserted by an SCP, see clause 6.10.9.1 of 3GPP TS 29.500 [4].				

Table 6.1.3.3.3.2-4: Headers supported by the 307 Response Code on this resource

Name	Data type	P	Cardinality	Description
Location	string	M	1	An alternative URI of the resource located on an alternative MF (service) instance. Or the same URI, if a request is redirected to the same target resource via a different SCP.
3gpp-Sbi-Target-Nf-Id	string	O	0..1	Identifier of the target MF instance ID towards which the request is redirected.

Table 6.1.3.3.3.2-5: Headers supported by the 308 Response Code on this resource

Name	Data type	P	Cardinality	Description
Location	string	M	1	An alternative URI of the resource located on an alternative MF (service) instance. Or the same URI, if a request is redirected to the same target resource via a different SCP.
3gpp-Sbi-Target-Nf-Id	string	O	0..1	Identifier of the target MF instance ID towards which the request is redirected.

6.1.3.3.4 Resource Custom Operations

None in the release of specification.

6.1.4 Custom Operations without associated resources

None in this release of specification.

6.1.5 Notifications

None in this release of specification.

6.1.6 Data Model

6.1.6.1 General

This clause specifies the application data model supported by the API.

Table 6.1.6.1-1 specifies the data types defined for the Nmf_MRM service based interface protocol.

Table 6.1.6.1-1: Nmf_MRM specific Data Types

Data type	Clause defined	Description	Applicability
MediaContext	6.1.6.2.2	Represents an individual context resource on MF.	
TerminationInfo	6.1.6.2.3	Represents the termination information.	
MediaInfo	6.1.6.2.4	Represents the media information.	
DcMedia	6.1.6.2.5	Represents the DC media descriptor.	
ArMedia	6.1.6.2.6	Represents the AR media descriptor.	
Mdc1Info	6.1.6.2.7	Represents the MDC1 interface information.	
Mdc2Info	6.1.6.2.8	Represents the MDC2 interface information.	
NonDcMedia	6.1.6.2.9	Represents the audio and video media.	
SdpString	6.1.6.3.2	Represents the m=line and a=lines in the SDP media description.	
Mdc2Protocol	6.1.6.3.3	Represents the type of the transport layer protocols for MDC2 interface.	
AvatarMedia	6.1.6.2.10	Represents the avatar media descriptor.	
InterworkingInstruction	3GPP TS 29.175 [17]	The interworking instruction of data channel transcoding.	

Table 6.1.6.1-2 specifies data types re-used by the Nmf_MRM service based interface protocol from other specifications, including a reference to their respective specifications and when needed, a short description of their use within the Nmf_MRM service based interface.

Table 6.1.6.1-2: Nmf_MRM re-used Data Types

Data type	Reference	Comments	Applicability
ProblemDetails	3GPP TS 29.571 [16]	Problem Details	
MediaId	3GPP TS 29.571 [16]	Uniquely identifies one media flow within an IMS session.	
MaxMessageSize	3GPP TS 29.571 [16]	Max SCTP user message size.	
MediaResourceType	3GPP TS 29.571 [16]	Media resource types.	
MediaProxy	3GPP TS 29.571 [16]	Media proxy configuration applicable to the media flow	
DcEndpoint	3GPP TS 29.571 [16]	DC endpoint	
DcStream	3GPP TS 29.571 [16]	Data channel stream information.	
ReplaceHttpUrl	3GPP TS 29.571 [16]	Replacement HTTP URL allocated by the application layer.	
Uri	3GPP TS 29.571 [16]	Uri	
Endpoint	3GPP TS 29.571 [16]	Represents the IP endpoint.	
MdcEndpoint	3GPP TS 29.571 [16]	Represents the MDC1 and MDC2 endpoint information.	
RenderingMode	3GPP TS 29.571 [16]	Represents the rendering mode of avatar media.	

6.1.6.2 Structured data types

6.1.6.2.1 Introduction

This clause defines the structures to be used in resource representations.

6.1.6.2.2 Type: MediaContext

Table 6.1.6.2.2-1: Definition of type MediaContext

Attribute name	Data type	P	Cardinality	Description	Applicability
contextId	string	C	0..1	Identifies a specific context. It is assigned by the MF when creating a new context, i.e. Nmf_MRM_Create operation. It shall be contained except in the request of creating a new media context.	
terminations	array(TerminationInfo)	M	1..N	Represents the media terminations	

6.1.6.2.3 Type: TerminationInfo

Table 6.1.6.2.3-1: Definition of type TerminationInfo

Attribute name	Data type	P	Cardinality	Description	Applicability
terminationId	string	M	1	Identifies a specific media termination. It shall be set to empty string (i.e., "") when service consumer requests the MF to add a new media termination in the Nmf_MRM_Create or Nmf_MRM_Update service operation. Otherwise, it shall contain the termination Id allocated by the MF.	
medias	array(MediaInfo)	M	1..N	Represent the list of media resources for one or multiple medias on the Mb interface.	
NOTE:	When indicating transcoding via MF, the "DC" media type and the "VIDEO" media type shall belong to different terminations.				

6.1.6.2.4 Type: MediaInfo

Table 6.1.6.2.4-1: Definition of type MediaInfo

Attribute name	Data type	P	Cardinality	Description	Applicability
mediaId	MediaId	M	1	Identifies a media stream, e.g. data channel.	
associatedMediaId	MediaId	C	0..1	The identifier of the original multiplexed media stream which is de-multiplexed. It shall be included only when the mediaId is null, mediaInstruction is set to "ORIGINATE" and the media stream identified by the value of this attribute is de-multiplexed.	
mediaResourceType	MediaResourceType	M	1	Represents the media resource type, e.g. DC, AR.	
localMbEndpoint	Endpoint	C	0..1	Represents the local IP address and port number of the local endpoint of MF on the Mb interface. It will be allocated by MF. It shall be contained in the response when creating a new media stream, i.e. Nmf_MRM_Create operation or Nmf_MRM_Update operation with adding a new termination or adding a media in an existing termination. (NOTE 1, NOTE 2)	
remoteMbEndpoint	Endpoint	C	0..1	Represents the IP address and port number of the remote endpoint on the Mb interface, e.g. remote UE. It shall be contained in the request of creating a new media stream, i.e. Nmf_MRM_Create operation or Nmf_MRM_Update operation with adding a new termination or adding a media in an existing termination. It can be Null when originating a new media. (NOTE 2)	
dcMedia	DcMedia	C	0..1	Represents the data channel media descriptors. It shall be contained if the mediaResourceType is set to "DC".	
arMedia	ArMedia	C	0..1	Represents the AR media descriptors. It shall be contained if the media is used for AR communication.	
remoteNonDcMedia	NonDcMedia	C	0..1	Represents the audio or video media. It shall be contained in the request of creating a new media if the mediaResourceType is set to "AUDIO" or "VIDEO". It can be "Null" when originating a new media.	
localNonDcMedia	NonDcMedia	C	0..1	Represents the audio or video media. It shall be contained in the response of creating a new media if the mediaResourceType is set to "AUDIO" or "VIDEO".	
mediaProcessingURL	Uri	C	0..1	The mediaProcessingURL indicates the address where MF receive service-related media instructions. It shall be contained in the response when creating a new media stream, i.e. Nmf_MRM_Create operation or Nmf_MRM_Update operation with adding a new termination or adding a media in an existing termination. (NOTE 1)	

avatarMedia	AvatarMedia	O	0..1	Represents the additional media specifications needed for avatar communication services.	
NOTE 1: The IE cannot be changed once the media has been established.					
NOTE 2: The attribute "transport" inside Endpoint data type shall be set to value only "UDP".					

6.1.6.2.5 Type: DcMedia

Table 6.1.6.2.5-1: Definition of type DcMedia

Attribute name	Data type	P	Cardinality	Description	Applicability
mediaProxyConfig	MediaProxy	M	1	Represents the media proxy configuration on the MF. The value shall be set to "HTTP_PROXY" if the mediaId represents the bootstrap data channel. The value shall be set to "DC_APPLICATION_PROXY" if the mediaId represents the network initiated P2P application data channel.	
replaceHttpUrl	map(ReplaceHttpUrl)	C	0..1	A map (list of key-value pairs where streamId is used as key of the map) of ReplacementHttpUrl. Represents URL for the specific IMS subscriber when requesting the application list (e.g. graphical user interface) via the MDC1 interface. It shall be included if the mediaId represents bootstrap data channel and streamId is 0 and 100.	
mdc1Info	Mdc1Info	C	0..1	Represents the information of MDC1 interface. It shall be contained when the mediaId represents a bootstrap data channel.	
mdc2Info	Mdc2Info	C	0..1	Represents the information of MDC2 interface. It shall be contained when the mediaId represents a P2A/A2P application data channel.	
streams	map(DcStream)	M	1..N	A map (list of key-value pairs where streamId is used as key of the map) of DcStream. Represents the data channel mapping and configuration information when originating/terminating data channel media flows on the Mb interface.	
maxMessageSize	MaxMessageSize	O	0..1	Represents the maximum size of to be expected.	
localDcEndpoint	DcEndpoint	C	0..1	Represents the DC endpoint for the Data Channel. It will be allocated by MF. It shall be contained in the response when creating a new media stream, i.e. Nmf_MRM_Create operation or Nmf_MRM_Update operation with adding a new termination or adding a media in an existing termination. (NOTE)	
remoteDcEndpoint	DcEndpoint	C	0..1	Represents the DC Endpoint for the Data Channel on the Mb interface. It shall be contained in the request of creating a new media stream, i.e. Nmf_MRM_Create operation or Nmf_MRM_Update operation with adding a new termination or adding a media in an existing termination. It can be NULL when originating a data channel media. (NOTE)	
interworkingInfo	InterworkingInstruction	O	0..1	Represents instructions on how to interwork to other media types.	
NOTE: The IE cannot be changed once the media has been established.					

6.1.6.2.6 Type: ArMedia

Table 6.1.6.2.6-1: Definition of type ArMedia

Attribute name	Data type	P	Cardinality	Description	Applicability
mediaProcessingSpec	string	M	1	It specifies how the AR media stream should be processed. It indicates how the MF assists in the AR media rendering function.	

6.1.6.2.7 Type: Mdc1Info

Table 6.1.6.2.7-1: Definition of type Mdc1Info

Attribute name	Data type	P	Cardinality	Description	Applicability
remoteMdc1Endpoint	MdcEndpoint	C	0..1	Represents the remote MDC1 media endpoint information of DCSF. It shall be contained in the request when creating a new media.	
localMdc1Endpoint	MdcEndpoint	C	0..1	Represents the local MF MDC1 media endpoint information. It will be allocated by MF and contained in the response when creating a new media.	
NOTE: The MDC1 interface shall use "TCP/TLS" transport layer protocol and the attributes "tlsId", "fingerprint" within the data type MdcEndpoint shall be included.					

6.1.6.2.8 Type: Mdc2Info

Table 6.1.6.2.8-1: Definition of type Mdc2Info

Attribute name	Data type	P	Cardinality	Description	Applicability
mdc2Protocol	Mdc2Protocol	C	0..1	Represents the transport layer protocol or protocol stack for MDC2 interface. It shall be included when the mediaProxyConfig is set to "HTTP".	
remoteMdc2Endpoint	MdcEndpoint	C	0..1	Represents the remote MDC2 media endpoint information on DC AS. It shall be included in the request for creating a new media. (NOTE 1) (NOTE 2)	
localMdc2Endpoint	MdcEndpoint	C	0..1	Represents the local MDC2 media endpoint information on MF. It will be allocated by MF and contained in the response when creating a new media. (NOTE 1) (NOTE 2)	
NOTE 1: When the mdc2Protocol is set to "UDP/DTLS/SCTP", "TCP/TLS" or "SCTP/DTLS", the attributes "tlsId" and "fingerprint" within the data type MdcEndpoint shall be included, and the "securitySetup" may be included. When the mdc2Protocol is set to "UDP/DTLS/SCTP", the attributes "sctpPort" within the data type MdcEndpoint shall be included.					
NOTE 2: When the mediaProxyConfig is set to "UDP_PROXY", only UDP/IP connections should be established between the MF and the DC AS, the attributes "tlsId", "sctpPort" and "fingerprint" within the MdcEndpoint shall not present.					

6.1.6.2.9 Type: NonDcMedia

Table 6.1.6.2.9-1: Definition of type NonDcMedia

Attribute name	Data type	P	Cardinality	Description	Applicability
sdpmLine	SdpString	M	1	Represents the m=line in the SDP media description.	
sdpaLines	array(SdpString)	M	0..N	Represents the a=lines in the SDP media description.	
associatedDcMediaId	MediaId	O	0..1	Indicates the data channel media identifier this media associates.	

6.1.6.2.10 Type: AvatarMedia

Table 6.1.6.2.10-1: Definition of type AvatarMedia

Attribute name	Data type	P	Cardinality	Description	Applicability
resourceUrl	Uri	M	1	Represents the URL that MF to retrieve the avatar representation to perform network rendering.	
mediaProcessSpec	string	M	1	It specifies how media should be processed.	
renderingMode	RenderingMode	M	1	It specifies the rendering mode of avatar media.	

6.1.6.3 Simple data types and enumerations

6.1.6.3.1 Introduction

This clause defines simple data types and enumerations that can be referenced from data structures defined in the previous clauses.

6.1.6.3.2 Simple data types

The simple data types defined in table 6.1.6.3.2-1 shall be supported.

Table 6.1.6.3.2-1: Simple data types

Type Name	Type Definition	Description	Applicability
SdpString	string	Represents the content after "m=" of the "m=" line and the content after "a=" of the associated "a=" lines.	

6.1.6.3.3 Enumeration: Mdc2Protocol

The enumeration Mdc2Protocol represents the type of the transport layer protocols for MDC2 interface. It shall comply with the provisions defined in table 6.1.6.3.3-1.

Table 6.1.6.3.3-1: Mdc2Protocol

Enumeration value	Description
"UDP/DTLS/SCTP"	UDP/DTLS/SCTP for MDC2 interface.
"TCP"	TCP for MDC2 interface.
"UDP"	UDP for MDC2 interface.
"SCTP"	SCTP for MDC2 interface.
"TCP/TLS"	TCP/TLS for MDC2 interface.
"SCTP/DTLS"	SCTP/DTLS for MDC2 interface.

6.1.7 Error Handling

6.1.7.1 General

For the Nmf_MRM API, HTTP error responses shall be supported as specified in clause 4.8 of 3GPP TS 29.501 [5]. Protocol errors and application errors specified in table 5.2.7.2-1 of 3GPP TS 29.500 [4] shall be supported for an HTTP method if the corresponding HTTP status codes are specified as mandatory for that HTTP method in table 5.2.7.1-1 of 3GPP TS 29.500 [4].

In addition, the requirements in the following clauses are applicable for the Nmf_MRM API.

6.1.7.2 Protocol Errors

No specific procedures for the Nmf_MRM service are specified.

6.1.7.3 Application Errors

The application errors defined for the Nmf_MRM service are listed in Table 6.1.7.3-1.

Table 6.1.7.3-1: Application errors

Application Error	HTTP status code	Description
CONTEXT_NOT_FOUND	404 Not Found	Indicates that the requested context is not found in the MF.
MEDIA_CONNECTION_CHANGED	403 Forbidden	Indicates that the connection information which has been marked that cannot be changed once the media has established is changed in the request. including the remoteMbEndpoint, localMbEndpoint etc.
MEDIA_ID_CONFLICT	409 Conflict	Indicates that a new media to be created has the identical mediaId already assigned to one created media.
INSUFFICIENT_RESOURCES	500 Server Internal Error	Indicates that the modification of media context has failed due to insufficient resources.

6.1.8 Feature negotiation

The optional features in table 6.1.8-1 are defined for the Nmf_MRM API. They shall be negotiated using the extensibility mechanism defined in clause 6.6 of 3GPP TS 29.500 [4].

Table 6.1.8-1: Supported Features

Feature number	Feature Name	Description

6.1.9 Security

As indicated in 3GPP TS 33.501 [8] and 3GPP TS 29.500 [4], the access to the Nmf_MRM API may be authorized by means of the OAuth2 protocol (see IETF RFC 6749 [9]), based on local configuration, using the "Client Credentials" authorization grant, where the NRF (see 3GPP TS 29.510 [10]) plays the role of the authorization server.

If OAuth2 is used, an NF Service Consumer, prior to consuming services offered by the Nmf_MRM API, shall obtain a "token" from the authorization server, by invoking the Access Token Request service, as described in 3GPP TS 29.510 [10], clause 5.4.2.2.

NOTE: When multiple NRFs are deployed in a network, the NRF used as authorization server is the same NRF that the NF Service Consumer used for discovering the Nmf_MRM service.

The Nmf_MRM API defines a single scope "nmf-mrm" for the entire service, and it does not define any additional scopes at resource or operation level.

6.1.10 HTTP redirection

An HTTP request may be redirected to a different MF service instance when using direct or indirect communications (see 3GPP TS 29.500 [4]).

An SCP that reselects a different MF producer instance will return the NF Instance ID of the new MF producer instance in the 3gpp-Sbi-Producer-Id header, as specified in clause 6.10.3.4 of 3GPP TS 29.500 [4].

If an MF redirects a service request to a different MF using an 307 Temporary Redirect or 308 Permanent Redirect status code, the identity of the new MF towards which the service request is redirected shall be indicated in the 3gpp-Sbi-Target-Nf-Id header of the 307 Temporary Redirect or 308 Permanent Redirect response as specified in clause 6.10.9.1 of 3GPP TS 29.500 [4].

Annex A (normative): OpenAPI specification

A.1 General

This Annex specifies the formal definition of the API(s) defined in the present specification. It consists of OpenAPI specifications in YAML format.

This Annex takes precedence when being discrepant to other parts of the specification with respect to the encoding of information elements and methods within the API(s).

NOTE 1: The semantics and procedures, as well as conditions, e.g. for the applicability and allowed combinations of attributes or values, not expressed in the OpenAPI definitions but defined in other parts of the specification also apply.

Informative copies of the OpenAPI specification files contained in this 3GPP Technical Specification are available on a Git-based repository that uses the GitLab software version control system (see clause 5.3.1 of 3GPP TS 29.501 [5] and clause 5B of 3GPP TR 21.900 [7]).

A.2 Nmf_MRM API

```
openapi: 3.0.0

info:
  title: 'MF Media Resource Management (MRM) Service'
  version: 1.1.0-alpha.3
  description: |
    MF Media Resource Management (MRM) Service.
    © 2025, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).
    All rights reserved.

externalDocs:
  description: >
    3GPP TS 29.176 V19.3.0; IP Multimedia Subsystem (IMS); Media Function (MF) Services; Stage 3.
  url: https://www.3gpp.org/ftp/Specs/archive/29_series/29.176/

servers:
  - url: '{apiRoot}/nmf-mrm/v1'
    variables:
      apiRoot:
        default: https://example.com
        description: apiRoot as defined in clause 4.4 of 3GPP TS 29.501

security:
  - {}
  - oAuth2ClientCredentials:
      - nmf-mrm

paths:
  /contexts:
    post:
      summary: Create a new media context.
      operationId: CreateMediaContext
      tags:
        - MediaContext(Collection)
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/MediaContext'
      responses:
        '201':
          description: Success
          content:
            application/json:
```

```

    schema:
      $ref: '#/components/schemas/MediaContext'
  headers:
    Location:
      description: >
        Contains the URI of the newly created resource, according to the structure:
        {apiRoot}/nmf-mrm/<version>/contexts/{contextId}
      required: true
      schema:
        type: string
'307':
  description: Temporary Redirect
  content:
    application/json:
      schema:
        $ref: 'TS29571_CommonData.yaml#/components/schemas/RedirectResponse'
  headers:
    Location:
      description: The URI pointing to the resource located on the redirect target MF.
      required: true
      schema:
        type: string
'308':
  description: Permanent Redirect
  content:
    application/json:
      schema:
        $ref: 'TS29571_CommonData.yaml#/components/schemas/RedirectResponse'
  headers:
    Location:
      description: The URI pointing to the resource located on the redirect target MF.
      required: true
      schema:
        type: string
'400':
  $ref: 'TS29571_CommonData.yaml#/components/responses/400'
'401':
  $ref: 'TS29571_CommonData.yaml#/components/responses/401'
'403':
  $ref: 'TS29571_CommonData.yaml#/components/responses/403'
'404':
  $ref: 'TS29571_CommonData.yaml#/components/responses/404'
'409':
  $ref: 'TS29571_CommonData.yaml#/components/responses/409'
'411':
  $ref: 'TS29571_CommonData.yaml#/components/responses/411'
'413':
  $ref: 'TS29571_CommonData.yaml#/components/responses/413'
'415':
  $ref: 'TS29571_CommonData.yaml#/components/responses/415'
'429':
  $ref: 'TS29571_CommonData.yaml#/components/responses/429'
'500':
  $ref: 'TS29571_CommonData.yaml#/components/responses/500'
'503':
  $ref: 'TS29571_CommonData.yaml#/components/responses/503'
default:
  $ref: 'TS29571_CommonData.yaml#/components/responses/default'

/contexts/{contextId}:
  patch:
    summary: Update Media Context
    operationId: UpdateMediaContext
    tags:
      - Media Context ID (Document)
    parameters:
      - name: contextId
        in: path
        required: true
        description: Unique ID of the Media Context to update.
        schema:
          type: string
    requestBody:
      content:
        application/json-patch+json:
          schema:
            type: array

```

```

    items:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/PatchItem'
    minItems: 1
  required: true
  responses:
    '200':
      description: Expected response to a valid request.
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/MediaContext'
    '204':
      description: Expected response with empty body
    '307':
      description: Temporary Redirect
      content:
        application/json:
          schema:
            $ref: 'TS29571_CommonData.yaml#/components/schemas/RedirectResponse'
      headers:
        Location:
          description: The URI pointing to the resource located on the redirect target MF.
          required: true
          schema:
            type: string
    '308':
      description: Permanent Redirect
      content:
        application/json:
          schema:
            $ref: 'TS29571_CommonData.yaml#/components/schemas/RedirectResponse'
      headers:
        Location:
          description: The URI pointing to the resource located on the redirect target MF.
          required: true
          schema:
            type: string
    '400':
      $ref: 'TS29571_CommonData.yaml#/components/responses/400'
    '403':
      $ref: 'TS29571_CommonData.yaml#/components/responses/403'
    '404':
      $ref: 'TS29571_CommonData.yaml#/components/responses/404'
    '409':
      $ref: 'TS29571_CommonData.yaml#/components/responses/409'
    '411':
      $ref: 'TS29571_CommonData.yaml#/components/responses/411'
    '412':
      $ref: 'TS29571_CommonData.yaml#/components/responses/412'
    '413':
      $ref: 'TS29571_CommonData.yaml#/components/responses/413'
    '415':
      $ref: 'TS29571_CommonData.yaml#/components/responses/415'
    '429':
      $ref: 'TS29571_CommonData.yaml#/components/responses/429'
    '500':
      $ref: 'TS29571_CommonData.yaml#/components/responses/500'
    '501':
      $ref: 'TS29571_CommonData.yaml#/components/responses/501'
    '503':
      $ref: 'TS29571_CommonData.yaml#/components/responses/503'
  default:
    $ref: 'TS29571_CommonData.yaml#/components/responses/default'

delete:
  summary: Deletes a given Media Context
  operationId: DeleteMediaContext
  tags:
    - Media Context ID (Document)
  parameters:
    - name: contextId
      in: path
      required: true
      description: Unique ID of the Media Context to delete.
      schema:
        type: string
  responses:
    '204':

```

```

description: Expected response to a successful deletion.
'307':
description: Temporary Redirect
content:
  application/json:
    schema:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/RedirectResponse'
headers:
  Location:
    description: The URI pointing to the resource located on the redirect target MF.
    required: true
    schema:
      type: string
'308':
description: Permanent Redirect
content:
  application/json:
    schema:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/RedirectResponse'
headers:
  Location:
    description: The URI pointing to the resource located on the redirect target MF.
    required: true
    schema:
      type: string
'400':
  $ref: 'TS29571_CommonData.yaml#/components/responses/400'
'401':
  $ref: 'TS29571_CommonData.yaml#/components/responses/401'
'403':
  $ref: 'TS29571_CommonData.yaml#/components/responses/403'
'404':
  $ref: 'TS29571_CommonData.yaml#/components/responses/404'
'411':
  $ref: 'TS29571_CommonData.yaml#/components/responses/411'
'429':
  $ref: 'TS29571_CommonData.yaml#/components/responses/429'
'500':
  $ref: 'TS29571_CommonData.yaml#/components/responses/500'
'501':
  $ref: 'TS29571_CommonData.yaml#/components/responses/501'
'503':
  $ref: 'TS29571_CommonData.yaml#/components/responses/503'
default:
  $ref: 'TS29571_CommonData.yaml#/components/responses/default'

```

components:

```

securitySchemes:
  oAuth2ClientCredentials:
    type: oauth2
    flows:
      clientCredentials:
        tokenUrl: '{nrfApiRoot}/oauth2/token'
        scopes:
          nmf-mrm: Access to the Nmf_MRM API.

```

schemas:

```

MediaContext:
description: Information of a Media Context in a MF.
type: object
required:
  - terminations
properties:
  contextId:
    type: string
  terminations:
    type: array
    items:
      $ref: '#/components/schemas/TerminationInfo'
    minItems: 1

```

```

TerminationInfo:
description: Represents the termination information.
type: object
required:
  - terminationId

```

```

- medias
properties:
  terminationId:
    type: string
  medias:
    type: array
    items:
      $ref: '#/components/schemas/MediaInfo'
    minItems: 1

MediaInfo:
description: Represents the media information.
type: object
required:
- mediaId
- mediaResourceType
properties:
  mediaId:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/MediaId'
  associatedMediaId:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/MediaId'
  mediaResourceType:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/MediaResourceType'
  localMbEndpoint:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/Endpoint'
  remoteMbEndpoint:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/Endpoint'
  dcMedia:
    $ref: '#/components/schemas/DcMedia'
  arMedia:
    $ref: '#/components/schemas/ArMedia'
  localNonDcMedia:
    $ref: '#/components/schemas/NonDcMedia'
  remoteNonDcMedia:
    $ref: '#/components/schemas/NonDcMedia'
  mediaProcessingUri:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/Uri'
  avatarMedia:
    $ref: '#/components/schemas/AvatarMedia'

DcMedia:
description: Represents the DC media descriptor.
type: object
required:
- streams
- mediaProxyConfig
properties:
  mediaProxyConfig:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/MediaProxy'
  replaceHttpUrl:
    type: object
    description: >
      Contains a list of replacement HTTP URLs. The streamId attribute
      within the ReplaceHttpUrl data type is the key of the map.
    additionalProperties:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/ReplaceHttpUrl'
    minProperties: 1
  mdclInfo:
    $ref: '#/components/schemas/MdclInfo'
  mdc2Info:
    $ref: '#/components/schemas/Mdc2Info'
  streams:
    type: object
    description: >
      Contains a data channel mapping and configuration information. The streamId
      attribute within the DcStream data type is the key of the map.
    additionalProperties:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/DcStream'
    minProperties: 1
  maxMessageSize:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/MaxMessageSize'
  localDcEndpoint:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/DcEndpoint'
  remoteDcEndpoint:
    $ref: 'TS29571_CommonData.yaml#/components/schemas/DcEndpoint'
  interworkingInfo:
    $ref: 'TS29175_Nimsas_MediaControl.yaml#/components/schemas/InterworkingInstruction'

```

```

Mdc1Info:
  description: Represents the information of MDC1 interface.
  type: object
  properties:
    remoteMdc1Endpoint:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/MdcEndpoint'
    localMdc1Endpoint:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/MdcEndpoint'

Mdc2Info:
  description: Represents the information of MDC2 interface.
  type: object
  properties:
    remoteMdc2Endpoint:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/MdcEndpoint'
    localMdc2Endpoint:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/MdcEndpoint'
    mdc2Protocol:
      $ref: '#/components/schemas/Mdc2Protocol'

ArMedia:
  description: Represents the AR media descriptor.
  type: object
  required:
    - mediaProcessingSpec
  properties:
    mediaProcessingSpec:
      type: string

AvatarMedia:
  description: Represents the Avatar media descriptor.
  type: object
  required:
    - resourceUrl
    - mediaProcessSpec
    - renderingMode
  properties:
    resourceUrl:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/Uri'
    mediaProcessSpec:
      type: string
    renderingMode:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/RenderingMode'

NonDcMedia:
  description: Represents the audio and video media
  type: object
  required:
    - sdpmLine
    - sdpaLines
  properties:
    sdpmLine:
      $ref: '#/components/schemas/SdpString'
    sdpaLines:
      type: array
      items:
        $ref: '#/components/schemas/SdpString'
    associatedDcMediaId:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/MediaId'

SdpString:
  description: The content after "m=" of the "m=" line and the content after "a=" of the
  associated "a=" lines.
  type: string

Mdc2Protocol:
  description: The type of the transport layer protocols for MDC2 interface.
  anyOf:
    - type: string
      enum:
        - UDP/DTLS/SCTP
        - TCP
        - UDP
        - SCTP
        - TCP/TLS
        - SCTP/DTLS

```

- type: string
- description: >
 - This string provides forward-compatibility with future extensions to the enumeration and is not used to encode content defined in the present version of this API.

Annex B (informative): Change history

Change history							
Date	Meeting	Tdoc	CR	Rev	Cat	Subject/Comment	New version
2023-04	CT4#115e	C4-231552				Version 0.1.0	0.1.0
2023-05	CT4#116	C4-232178				Clause 4 Overview	0.2.0
2023-05	CT4#116	C4-232224				Clause 5.1 Introduction of DCMF services	0.2.0
2023-05	CT4#116	C4-232506				Service description and service operation introduction of MRM service	0.2.0
2023-05	CT4#116	C4-232607				MRM API definition	0.2.0
2023-05	CT4#116	C4-232637				MRM service operations	0.2.0
2023-08	CT4#117	C4-233707				Change DCMF to MF	0.3.0
2023-08	CT4#117	C4-233307				Update the data channel related data types	0.3.0
2023-08	CT4#117	C4-233306				Update MF service to support AR	0.3.0
2023-10	CT4#118	C4-234539				Update the scope	0.4.0
2023-10	CT4#118	C4-234540				Support of Redirection	0.4.0
2023-10	CT4#118	C4-234575				Clarification on the DC media	0.4.0
2023-10	CT4#118	C4-234543				Editorial Corrections	0.4.0
2023-10	CT4#118	C4-234544				Update the data model for AR remote cooperation	0.4.0
2023-10	CT4#118	C4-234341				Usage of the apiVersion placeholder	0.4.0
2023-11	CT4#119	C4-235576				Move the Endpoint to Common Data	0.5.0
2023-11	CT4#119	C4-235473				OpenAPI of Nmf_MRM Service	0.5.0
2023-12	CT#102	CP-233022				Presented for information and approval	1.0.0
2023-12	CT#102	CP-233022				Approved in TSC CT#102	18.0.0
2024-03	CT#103	CP-240043	0002		F	Support of media renegotiation	18.1.0
2024-03	CT#103	CP-240043	0001	1	B	EN resolution on AR media	18.1.0
2024-03	CT#103	CP-240043	0004	1	F	Change the map data type and correct the editorial errors	18.1.0
2024-03	CT#103	CP-240043	0007	1	F	Clarification of definitions, symbols and abbreviations	18.1.0
2024-03	CT#103	CP-240043	0008	1	F	Correction IE of mdc2Protocol	18.1.0
2024-03	CT#103	CP-240043	0003	2	F	Add the error insufficient resource	18.1.0
2024-03	CT#103	CP-240043	0006	1	F	Update the Nmf_MRM_Update service operation	18.1.0
2024-03	CT#103	CP-240043	0005	3	F	Update the Nmf_MRM_Create service operation	18.1.0
2024-03	CT#103	CP-240056	0009		F	29.176 Rel-18 API version and External doc update	18.1.0
2024-06	CT#104	CP-241044	0010		F	Clarification on the Media Context ID and termination ID	18.2.0
2024-06	CT#104	CP-241044	0011	1	F	Update the MDC1 and MDC2 interface information	18.2.0
2024-06	CT#104	CP-241305	0012	3	F	Inclusion of the video and audio media	18.2.0
2024-06	CT#104	CP-241044	0013	2	F	Update the presence condition and cardinality for mediaProxyConfig	18.2.0
2024-06	CT#104	CP-241052	0014		F	29.176 Rel-18 API version and External doc update	18.2.0
2024-09	CT#105	CP-242048	0017		F	Remove the redundant securitySetup in DcMedia	18.3.0
2024-09	CT#105	CP-242048	0015	1	F	Cardinality correction	18.3.0
2024-09	CT#105	CP-242048	0018	1	F	Update the procedure of creation a new media context	18.3.0
2024-09	CT#105	CP-242048	0016	1	F	Updates to MF Media Resource Management Service Operations	18.3.0
2024-09	CT#105	CP-242054	0019		F	29.176 Rel-18 API version and External doc update	18.3.0
2024-12	CT#106	CP-243027	0020		F	Correction on Nmf_MRM service	18.4.0
2024-12	CT#106	CP-243068	0022		F	29.176 Rel-18 API version and External doc update	18.4.0
2024-12	CT#106	CP-243045	0021	1	B	Support of interworking with MTSI client	19.0.0
2025-02	CT#107	CP-250047	0024	2	B	Support of avatar communication	19.1.0
2025-02	CT#107	CP-250137	0025		F	29.176 Rel-19 API version and External doc update	19.1.0
2025-06	CT#108	CP-251049	0028	1	A	Correction on the MediaInfo data type	19.2.0
2025-06	CT#108	CP-251049	0030	1	A	Correction on Nmf_MRM OpenAPI	19.2.0
2025-06	CT#108	CP-251068	0034	1	B	Define the rendering mode for avatar media specification data type	19.2.0
2025-06	CT#108	CP-251068	0031	1	B	Updates the Nmf_MRM service to support DC multiplexing	19.2.0
2025-06	CT#108	CP-251068	0032	1	B	Update the MediaProxy value	19.2.0
2025-06	CT#108	CP-251049	0036	1	A	HTTP Response Code for Application Error MEDIA_ID_CONFLICT	19.2.0
2025-06	CT#108	CP-251079	0039		F	29.176 Rel-19 API version and External doc update	19.2.0
2025-09	CT#109	CP-252050	0041	1	F	Correction on the data channel multiplexing	19.3.0
2025-09	CT#109	CP-252050	0040	2	B	Instruction on transcoding for interworking	19.3.0
2025-09	CT#109	CP-252176	0042		F	29.176 Rel-19 API version and External doc update	19.3.0

History

Document history		
V19.3.0	January 2026	Publication