|  |
| --- |
| 3GPP TS 26.567 V0.3.0 (2024-08) |
| Technical Specification |
| 3rd Generation Partnership Project;Technical Specification Group Services and System Aspects;Split Rendering over IMS;(Release 19) |
|   |
|  |  |
|  |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP.The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented.This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification.Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. |

|  |
| --- |
|  |
| ***3GPP***Postal address3GPP support office address650 Route des Lucioles - Sophia AntipolisValbonne - FRANCETel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16Internethttps://www.3gpp.org |
| ***Copyright Notification***No part may be reproduced except as authorized by written permission.The copyright and the foregoing restriction extend to reproduction in all media.© 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).All rights reserved.UMTS™ is a Trade Mark of ETSI registered for the benefit of its members3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational PartnersLTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational PartnersGSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 5

Introduction 6

1 Scope 7

2 References 7

3 Definitions of terms, symbols and abbreviations 7

3.1 Terms 7

3.2 Symbols 8

3.3 Abbreviations 8

4 System description 8

4.1 Overview 8

4.2 Reference Architecture 10

4.3 Reference Points 11

4.4 Split Rendering DCMTSI Client (SR-DCMTSI) 11

4.5 Media Function (MF) 11

4.6 DC Application Server (DC AS) 12

5 Media codecs, configuration, and data transport 12

5.1 General 12

5.2 Media codecs 12

5.3 Media configuration 12

5.4 Data transport 12

5.4.1 General 12

5.4.2 Metadata Formats 12

5.4.2.1 General 12

5.4.2.2 Pose Format 12

5.4.2.3 Action Format 12

6 Split Rendering Metrics 13

7 Procedures 13

7.1 General procedures for session establishment 13

7.2 General procedures for session modification 15

7.3 Network support procedures 17

Annex <A> (normative): <Normative annex for a Technical Specification> 19

Annex <B> (informative): <Informative annex for a Technical Specification> 20

B.1 Heading levels in an annex 20

Annex <F> (informative): 21

Change history 21

For definitive guidance on drafting 3GPP TSs and TRs, see [3GPP TS 21.801](https://www.3gpp.org/DynaReport/21801.htm).

Ensure all blue guidance text is removed before submitting the TS/TR to the TSG for approval.

# Foreword

This clause is mandatory; do not alter the text in any way other than to choose between "Specification" and "Report".

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 drafting the TS/TR, pay particular attention to the use of modal auxiliary verbs! TRs shall not contain any normative provisions.

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.

# Introduction

This clause is optional. If it exists, it shall be the second unnumbered clause.

# 1 Scope

The present document specifies functional entities, reference points and protocols for IMS-based split rendering. It also provides codecs for delivery of split-render media and metadata of split rendered content for both uplink and downlink. The present document also contains procedures of split rendering session establishment, session management, adaption, as well as other procedures to support split rendering process based on network support functions.

Key use cases enabled by this specification include XR services incorporating real-time and non-real-time media in industrial (e.g., for monitoring, maintenance, collaboration and tele-operation), enterprise and educational environments; entertainment use-cases, including cloud-gaming, and shared and collaborative entertainment and productivity XR services.

# 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 23.228: "IP Multimedia Subsystem (IMS); Stage 2".

[3] 3GPP TS 26.264: "IMS-based AR Real-Time Communication".

[4] 3GPP TS 23.501: "System architecture for the 5G System (5GS); Stage 2".

[5] 3GPP TS 26.565: "Split Rendering Media Service Enabler".

[6] 3GPP TS 26.119: "Device Media Capabilities for Augmented Reality Services".

…

[x] <doctype> <#>[ ([up to and including]{yyyy[-mm]|V<a[.b[.c]]>}[onwards])]: "<Title>".

It is preferred that the reference to TR 21.905 be the first in the list.

# 3 Definitions of terms, symbols and abbreviations

This clause and its three (sub) clauses are mandatory. The contents shall be shown as "void" if the TS/TR does not define any terms, symbols, or abbreviations.

## 3.1 Terms

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

Definition format (Normal)

**<defined term>:** <definition>.

**example:** text used to clarify abstract rules by applying them literally.

SR-DCMTSI Client A Split-Rendering capable DCMTSI Client

## 3.2 Symbols

For the purposes of the present document, the following symbols apply:

Symbol format (EW)

<symbol> <Explanation>

## 3.3 Abbreviations

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

DC Data Channel

DCMTSI Data Channel Multimedia Telephony Service for IMS

DCSF Data Channel Signalling Function

IMS IP Multimedia Core Network Subsystem

MF Media Function

MTSI Multimedia Telephony Service for IMS

P2A Person to Application

P2P Person to Person

# 4 System description

The main text of the document should start here, after the above clauses have been added.

The following styles and editing techniques are aimed to help in the formatting of the document using the 3GPP Template: 3GPP\_70.dot, available from the 3GPP FTP site (<https://www.3gpp.org/ftp/Information/All_Templates>).

## 4.1 Overview

*Editor’s note: document the background and existing TS/TR related with IMS-based split rendering, including potential requirements in terms of enhancement on functional entities/APIs to enable split rendering.*

The IMS stage-2 service description is specified in TS 23.228. The IMS architecture enhancements to support data channel services can be found in TS 23.228 Annex AC [2], which supports separation of signalling function and media function supporting data channel services.



Figure 4.1.1: Architecture of IMS supporting DC usage with MF

*Editor’s note: above content will be updated to align with TS 23.228 as well as SA2 R19 IMS related work and study items.*

Based on the architecture in Figure 4.1.1, TS 26.264 [3] provides the IMS-based conversational AR (Augmented reality) services, including the generalized end-to-end architecture to support AR communication over IMS DC, as shown in Figure 4.1.2.



Figure 4.1.2: Generalized IMS DC Architecture to support AR communication (TS 26.264)

Accordingly, AR Application Server (AR AS) is responsible for AR service control related to AR communication, including AR session media control and AR media capability negotiation with the UE.

The DCSF receives event reports from the IMS AS, and decides whether AR communication service is allowed to be provided during the IMS session. Additionally, the DCSF interacts with the AR AS for DC resource control.

MF/MRF supports AR conversational service by providing transcoding for terminals with limited capabilities. Additionally, the MF/MRF may collect spatial and media descriptions from UEs and create scene descriptions for symmetrical AR call experiences. MF/MRF also provide remote rendering for AR-MTSI clients in terminals with limited capabilities based on rendering negotiation. For remote rendering the AR-MTSI client provides AR metadata.

The IMS AS receives the media control instructions from the DCSF and accordingly interacts with the UE for connecting the UE's audio/video media termination to the MF/MRF and interacts with MF/MRF for data channel media resource management for AR media processing.

*Editor’s note: above content will be updated to align with TS 26.264 when it completes.*

According to the architectures above, the present document introduces a mapping to the IMS architecture for IMS-based split rendering.

## 4.2 Reference Architecture

*Editor’s note: identify function entities for IMS-based split rendering, introducing a mapping to the IMS architecture.*

The generalized IMS DC architecture to support split rendering is shown in Figure 4.2.1.



Figure 4.2.1: Generalized IMS DC Architecture to support split rendering

Split-Rendering DCMTSI Client (SR-DCMTSI Client): A DCMTSI Client, that is responsible for acquiring the UE media capabilities and interacting with the MF during the split-rendering process.

Media Function (MF): A Media Function, that is responsible for interacting with SR-DCMTSI Client during split-rendering process, monitoring resource usage, and managing/running the split rendering process, etc.

Data Channel Application Server (DC AS): A DC Application Server, that is responsible for service control related to split-rendering, including session media control and media capability negotiation with the UE via MF, etc.

*Editor’s note: the detailed definition of SR-DCMTSI Client and MF will be provided at section 4.4 and 4.5 of TS 26.567. Above text will be updated to align with those detailed functions. Furthermore, the definition of SR-DCMTSI Client will be revisited (legacy device or new device) according to the detailed functions defined in this specification.*

## 4.3 Reference Points

As shown in Figure 4.2.1, the following reference points are used to enable split rendering over IMS:

**Mb:** Reference point to enable split rendering between UE and MF, as specified in TS 23.228 [2].

**Gm**: Reference point to support communication between UE and IMS, as specified in TS 23.228 [2].

**MDC1**: Reference point for transport of data channel media between data channel media function and DCSF. MF terminates the bootstrap data channel from the UE and forward HTTP traffic between UE and DCSF via MDC1, as specified in TS 23.228 [2].

**MDC2:** Reference point for transport of data channel media between data channel media function and DC Application Server, for split rendering functions between application server to MF. MF relay traffic on A2P/P2A application data channels between the UE and the DC Application Server via MDC2, as specified in TS 23.228 [2].

The following reference points are also used to support split rendering related procedures:

**DC1**: Reference point between the DCSF and the IMS AS, as specified in TS 23.228 [2].

**DC2**: Reference point between the IMS AS and MF, for split rendering related data channel media resource management, as specified in TS 23.228 [2].

**DC3**: Reference point between the DCSF and NEF, as specified in TS 23.228 [2].

**DC4**: Reference point between the DCSF and DC Application Server, as specified in TS 23.228 [2].

**N33**: Reference point between NEF and DC Application server, network exposure to enable split rendering related applications, as specified in TS 23.501 [x].

Note: DC5, ISC, and N5 are out of the scope of this specification.

## 4.4 Split Rendering DCMTSI Client (SR-DCMTSI)

*Editor’s note: Additional details of SR DCMTSI client are FFS.*

An SR-DCMTSI is a DCMTSI client in terminal defined as per clause 3.1 of TS 26.114 which supports split rendering. It is responsible for acquiring media capabilities and interacting with IMS functions during a split rendering session. It is responsible for the signalling with DCSF to set-up a split rendering session and negotiating with the MF and DCAS about the parameters of the split rendering session, for example, the scene to be rendered and the split of rendering operations.

## 4.5 Media Function (MF)

*Editor’s note: Additional details of MF are FFS.*

MF is a Media Function which supports split rendering. It is responsible for interacting with SR-DCMTSI client and the DC AS during split-rendering process, monitoring resource usage, managing and running the split rendering process, etc.

## 4.6 DC Application Server (DC AS)

*Editor’s note: Additional details of DC AS are FFS.*

DC Application Server (DC AS) is responsible for service control related to split-rendering, including session media control and, session setup and media capability negotiation with the SR-DCMTSI client via MF and DCSF where applicable. The DC AS may be in the media path via MDC2, for example in P2A, A2P or P2A2P scenarios as defined in clause xx of TS 23.228.

# 5 Media codecs, configuration, and data transport

## 5.1 General

*Editor’s note: this clause provides media formats, signalling protocols to support split rendering, etc.*

## 5.2 Media codecs

*Editor’s note: media format, where possible, references to TS 26.114, TS 26.264, TS 26.119, TS 26.522 and TS 26.565.*

## 5.3 Media configuration

## 5.4 Data transport

## 5.4.1 General

This clause defines media and metadata formats that are used for IMS-based split rendering.

## 5.4.2 Metadata Formats

#### 5.4.2.1 General

SR-DCMTSI client and Media Function shall support the usage of the IMS data channel for the exchange of split rendering metadata with the MF. The data channel shall declare “3gpp-sr” as the data channel sub-protocol. The message content format depends on the type of the message. The data channel sub-protocol is defined in clause 8.3.3 of TS 26.565 [5].

#### 5.4.2.2 Pose Format

For XR services, the pose information format that is used for IMS-based split rendering [shall] comply with the format defined in clause 12.2 of TS 26.119 [6]. The pose information [shall] be carried as part of the data channel messaging mechanism. The metadata data channel message format is as defined in clause 8.3.3 of TS 26.565 [5]. The message type [shall] be “urn:3gpp:split-rendering:v1:pose”.

#### 5.4.2.3 Action Format

The action information format that is used for IMS-based split rendering [shall] comply with the format defined in clause 12.3 of TS 26.119 [6]. The action information [shall] be carried as part of the data channel messaging mechanism. The metadata data channel message format is as defined in clause 8.3.3 of TS 26.565 [5]. The message type [shall] be “urn:3gpp:split-rendering:v1:action”.

##

# 6 Split Rendering Metrics

*Editor’s note: this clause defines a set of metrics that are relevant to the operation of a split rendering session.*

# 7 Procedures

## 7.1 General procedures for session establishment

*Editor’s note: session establishment procedures*



**Figure7.1-1: high level call flows for split rendering over IMS**

The steps are as follows:

Step 1: The UE1 initiates a media session and establishes audio and video session connections with the UE2.

Step 2: UE1 sends a request to create a split rendering session leveraging the IMS network entities:

 When the UE1 discovers that its media capabilities cannot meet the related media rendering requirements, the UE1 decides to start split rendering call flow. Then the UE1 calculates which objects can be rendered by itself based on its status and decides which part of the objects to be rendered in the UE1 and the others to be rendered in the IMS network.

 The UE1 initiates the application data channels between the UE1 and the IMS AS, for the split rendering request and metadata transmission.

Step 3: The IMS AS interacts with the DCSF via DC1 for event notifications.

Step 4: The DCSF receives event reports from the IMS AS and decides whether data channel service is allowed to be provided during the IMS session. The DCSF manages bootstrap data channel and (if applicable) application data channel resources at the MF via the IMS AS;

Step 5 and 6: The IMS AS receives the data channel control instructions from the DCSF and accordingly interacts with the MF via DC2

Step 7: The IMS AS sends a Split Rendering Request to the DC AS via the MF through the established application data channel, the request includes the information of the objects to be rendered in IMS network.

Step 8: The DC AS sends a description of the split rendering output to the IMS AS via the MF.

Step 9: The IMS AS sends the media resource allocation request to the DCSF, to reserve XR media rendering resource for the UE1.

Step 10: When the resources are allocated successfully, the DCSF returns a successful response to the IMS AS.

Step 11: The IMS AS returns a successful response to the UE1.

Step 12: Successful SR session is established between UE1 and MF through the application data channel.

Step 13: Subsequent procedures continue for the UE2.

## 7.2 General procedures for session modification

*Editor’s note: session modification/update procedures*

The following below procedures indicate the session modification procedures during an established SR session over a standalone IMS DC channel, when the UE requires to download a dedicated application via a standalone IMS DC session. The session negotiation for split rendering is indicated as below:



Figure7.2-1: General procedures for session modification

Step 1-10: established IMS session and split rendering session establishment.

Step 11: When the UE1 discovers that its media capabilities cannot meet the related media rendering requirements, the UE1 decides to modify the existing split rendering call flow; UE sends a request to modify the split rendering session to the IMS AS.

Step 12: The IMS AS interacts with the DCSF via DC1 to send updated event notifications to modify the split.

Step 13: The DCSF receives event reports from the IMS AS and decides whether the requested data channel service is allowed to be provided during the IMS session.

Step 14: The IMS AS receives the updated data channel control instructions from the DCSF and accordingly interacts with the MF. Note: here it is assumed that there are more than one MF. It is also assumed that the serving MF does not satisfy the new request (due to unavailability).

Step 15: The (new) MF is discovered (if the existing serving MF does not satisfy the UE1 request for session modification).

Step 16: UE-1 sends application request to the newly discovered MF to request an application list.

Step 17: (new) MF response with application list include an identifier that if the application is capable to be split rendered. DCSF received the list and forward to the UE-1.

Step 18: If step 15 is yes, the IMS AS sends a Split Rendering Request to the (new) MF through the established application data channel, the request includes the information of the objects to be rendered in IMS network.

Step 19: The (new) MF sends a description of the split rendering output to the IMS AS.

Step 20: The IMS AS sends the media resource allocation request to the MF, to reserve media rendering resource for the UE1.

Step 21: When the resources are allocated successfully, the MF returns a successful response to the IMS AS.

Step 22: The IMS AS returns a successful response to the UE1.

Step 23: Successful SR session is established between UE1 and MF through the data channel.

Step 24: Similar session also can be established between UE-1 and UE-2 for remote UE.

## 7.3 Network support procedures

*Editor’s note:* *procedures for adaption of split rendering client and server based on network support.*

An SR-DCMTSI client or an MF may trigger further procedures during a split rendering, One such procedure may be to adapt the split of rendering operations between the SR-DCMTSI client and the MF during a split rendering session. This split may be due to change in operating conditions of the split rendering session, for example operating conditions of the UE, the MF or changes in the application or scene being rendered, for example changes in the scene description. Split adaptation may include data exchange, for example, exchange of adaptation messages, application state information and assets needed for the split rendering of an DC application. The following generic procedure shall apply, while the exact details may depend on the DC-application being rendered.



Figure7.3-1: General procedures for adaption of split rendering

The steps are as follows:

Step 1: The IMS session is established between the SR-DCMTSI client in terminal and a terminating SR-DCMTSI Client which may be in a terminal. For Person to Person calls, procedures in clause 7.1 are followed.

Step 2: A split rendering session is set up between the SR-DCMTSI client and a serving MF.

Step 3: Assets related to the application being split rendered may be delivered to participants of the split rendering session. The asset delivery may include javascript assets, scene descriptions and graphical objects needed for the session.

Rendering Loop:

The rendering loop is executed continuously during the duration of the split rendering session, for each frame.

Step 4: The SR-DCMTSI client in terminal sends metadata required for rendering to the MF. The metadata may include pose, pose predictions, user inputs etc.

Step 5 and 6: The SR-DCMTSI client in terminal and the MF render the frame.

Step 7: The frame rendered by the MF is transmitted to the SR-DCMTSI client in terminal as well as possible metadata.

Step 8: The SR-DCMTSI client in terminal composes a display frame from the received rendered media and media rendered locally.

NOTE: Steps 5,6,7 although ordered above, may occur in any order. Step 8 may include pose-correction. Step 8 and 6 may be executed as a single step.

Further Procedures:

Split Adaptation:

NOTE: Split Adaptation refers to adaptation of split rendering operations in an ongoing split rendering session between the SR-DCMTSI client and MF, without impacting the MF resources provisioned by IMS AS in step 9 of clause 7.1

Step 9: A trigger to adapt the split occurs at the SR-DCMTSI client in terminal; the trigger may be, for example, a change in available UE resources (e.g. battery, compute), changes in QoE of the SR session, changes in the scene/application being rendered.

Step 10: The SR-DCMTSI client in terminal decides if a new split of the rendering operations is needed and determines the new split.

Step 11: The SR-DCMTSI client in terminal sends a request to the MF to adapt the split to the new split.

Step 12: The MF actuates the new split of the rendering operations.

Step 13: The MF sends an acknowledgment of the new split to the SR-DCMTSI client in terminal.

Step 14: The MF and UE may exchange messages and data to support the new split of operations. This may include exchange of messages, for example, for synchronization of the state of the scene being split rendered or exchange of assets, for example, those in Step 3.

Step 15: The rendering loop (steps 4 through 12) continues.

Note:  Split adaptation is shown to be initiated by the SR-DCMTSI client in terminal for clarity, the procedure may be triggered by the MF. Further, other procedures to actuate the new split may be executed during the split rendering session.

Annex <A> (normative):
<Normative annex for a Technical Specification>

Start each annex on a new page.

Annexes are labelled A, B, C, etc. and designated either "normative" or "informative" depending on their content.

Normative annexes only to appear in Technical Specifications. Use style "Heading 8".

Annex <B> (informative):
<Informative annex for a Technical Specification>

Informative annexes may appear in both Technical Specifications and Technical Reports. Use style "Heading 8" for use in TSs.

Informative annexes shall not contain requirements for the implementation of the Technical Specification.

# B.1 Heading levels in an annex

Heading levels within an annex are used as in the main document, but for Heading level selection, the "A.", "B.", etc. are ignored. e.g. **B.1.2** is formatted using ***Heading 2*** style.

Annex <F> (informative):

Change history

|  |
| --- |
| Change history |
| Date | Meeting | TDoc | CR | Rev | Cat | Subject/Comment | New version |
| 04-2024 | 127b-e | S4-240645 |  |  |  | [SR\_IMS]TS 26.567 Skeleton | 0.0.1 |
| 04-2024 | 127b-e | S4-240790 |  |  |  | [SR\_IMS] version agreed during SA4#127bis-e (including S4-240583 and S4-240651)  | 0.1.0 |
| 05-2024 | 128 | S4-241272 |  |  |  | [SR\_IMS] version agreed during SA4#128 (including S4-241160, 241161, 241274 and S4-241213)  | 0.2.0 |
| 08-2024 | 129-e | S4-241739 |  |  |  | [SR\_IMS] version agreed during SA4#128 (including S4-241550, 241731, 241735 and S4-241728)  | 0.3.0  |
|  |  |  |  |  |  |  |  |