**3GPP TSG-SA WG4 Meeting #124S4-231009**

**Belrin,** **Germany, 22 – 26 May 2023**

**Source: NTT**

**Title: FS\_eiRTCW Permanent Document**

**Version: 5.0.0**

**Document for: Agreement**

**Agenda item 15.3**

# 1 Scope

The present document extends immersive Real-time Communication for WebRTC (iRTCW) and introduces a new concept called native WebRTC signalling.

This study includes following aspects:

1. Analyze gaps and identify required enhancements of terminal device and network architectures including additional functional entities (e.g., WebRTC Signalling Server, ICE-STUN Server, IMS Interworking Gateway, NNI Gateway).
2. Identify impacts on and possible enhancements for the WebRTC-based U-plane components in terms of adaptation, media handling, and cross-layer optimizations over 5G systems.
3. Identify signalling protocol details (e.g., based on JSON) for the common WebRTC-based immersive RTC session management.
4. Identify information elements in the C/U-Plane signal (including NNI) to enhance connectivity of media sessions with carrier assistance for WebRTC-based applications (including OTT applications).
5. Identify the minimal functional capabilities needed to support the enhancements identified in Objectives 2, 3 and 4 (including transport, NAT-traversal, and XR conferencing), state transitions, and typical call flows.
6. Identify collaboration formation with other WGs in 3GPP and SDOs including IETF and W3C.
7. Identify enhancements for E2E QoS realizations over 5G systems for communications between MNOs and WebRTC clients operating over non-5G links (e.g., Wi-Fi) using WebRTC-based transport. This also includes communication between WebRTC clients operating on tethering/tethered devices.
8. Study security, QoE reporting, and rate adaptation in tethered use cases (including coordination of Uu and non-3GPP access).

The study should consider as a principle that the third party access to the operator network need to be controlled with SLAs and with secure access to protect the underlying network resources.

# 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*.

[TR21.905] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[TS23.228] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".

[TS23.501] 3GPP TS 23.501: "System architecture for the 5G System (5GS); Stage 2".

[TS26.506] 3GPP TS 26.506: "5G Real-time Media Communication Architecture (Stage 2)"

[TS33.501] 3GPP TS 33.501: "Security architecture and procedures for 5G system".

[TS24.371] 3GPP TS 24.371: "Web Real-Time Communications (WebRTC) access to the IP Multimedia (IM) Core Network (CN) subsystem (IMS); Stage 3; Protocol specification".

[W3C.WD-webrtc] W3C Proposed Recommendation, "WebRTC 1.0: Real-time Communication Between Browsers", <https://www.w3.org/TR/webrtc/>.

[RFC791] IETF RFC 791: "Internet Protocol".

[RFC793] IETF RFC 793: "Transmission Control Protocol".

[RFC3261] IETF RFC 3261: "SIP: Session Initiation Protocol".

[RFC3489] IETF RFC 3489: "STUN – Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)".

[RFC6120] IETF RFC 6120: "Extensible Messaging and Presence Protocol (XMPP): Core".

[RFC6455] IETF RFC 6455: "The WebSocket Protocol".

[RFC6598] IETF RFC 6598: "IANA-Reserved IPv4 Prefix for Shared Address Space".

[RFC6749] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

[RFC7230] IETF RFC 7230: "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing".

[RFC7231] IETF RFC 7231: "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content"

[RFC7232] IETF RFC 7232: "Hypertext Transfer Protocol (HTTP/1.1): Conditional Requests"

[RFC7233] IETF RFC 7233: "Hypertext Transfer Protocol (HTTP/1.1): Range Requests"

[RFC7234] IETF RFC 7234: "Hypertext Transfer Protocol (HTTP/1.1): Caching"

[RFC7235] IETF RFC 7235: "Hypertext Transfer Protocol (HTTP/1.1): Authentication"

[RFC7362] IETF RFC 7362: "Latching: Hosted NAT Traversal (HNT) for Media in Real-Time Communication"

[RFC7540] IETF RFC 7540: "Hypertext Transfer Protocol Version 2 (HTTP/2)"

[RFC7635] IETF RFC 7635: "Session Traversal Utilities for NAT (STUN) Extension for Third-Party Authorization"

[RFC8200] IETF RFC 8200: "Internet Protocol, Version 6 (IPv6) Specification"

[RFC8259] IETF RFC 8259: "The JavaScript Object Notation (JSON) Data Interchange Format"

[RFC8441] IETF RFC 8441: "Bootstrapping WebSockets with HTTP/2"

[RFC8446] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3"

[RFC8825] IETF RFC 8825: "Overview: Real-Time Protocols for Browser-Based Applications"

[RFC8829] IETF RFC 8829: "JavaScript Session Establishment Protocol (JSEP)"

[RFC8445] IETF RFC 8445: “Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal”

[RFC8656] IETF RFC 8656: "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)"

[RFC9114] IETF RFC 9114: "HTTP/3"

[RFC9220] IETF RFC 9220: "Bootstrapping WebSockets with HTTP/3"

[OpenAPI] OpenAPI Initiative "OpenAPI Specification v3.0.0" https://spec.openapis.org/oas/v3.0.0

[AsyncAPI] AsyncAPI Initiative "AsyncAPI Specification v2.4.0" https://asyncapi.com/docs/specifications/v2/4/0

# 3 Definitions of terms, symbols and 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.

## 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].

Abbreviation format (EW)

<ABBREVIATION> <Expansion>

# 4 Motivations for Native WebRTC Signalling and assumptions

## 4.1 General

In 3GPP, the use of WebRTC technology has been investigated since Rel-12 (around 2014). They are a network-based architecture for WebRTC access to IMS specified in Annex U to TS 23.228 and its stage 3 protocols specified in TS 24.371. They define functional entities including WIC (WebRTC IMS Client) and eP-CSCF (P-CSCF enhanced for WebRTC). The eP-CSCF is assumed to be located in the Home IMS domain and communicates with other IMS entities using the existing interfaces. For the C-plane signalling between WIC and eP-CSCF, those specifications specify an option to use SIP over WebSocket, whose information model can be used for options other than SIP over WebSocket. Although SIP satisfies almost all conversational applications, it is somewhat over-engineered or too strict to extend. Another method which is flexible, extensible, and can be optimized for new XR conversational applications, therefore, should be investigated. These requirements remind us of the original design principle of WebRTC. WebRTC, by its inherent characteristics, does not regulate C-plane signalling and allow a wide range of C-plane signalling. This study looks over this design principle again and investigates a new SIP-decoupled C-plane signalling, called native WebRTC.

Regarding the level of signalling details, TS 24.371 specifies a signalling transport mechanism using SIP over WebSocket, but it is not a mandatory mechanism for eP-SCSF. Even though there are other options such as XMPP or other application protocols over WebSocket, a RESTful based interface, etc., TS 24.371 does not specify any details of C-plane signalling using other options. Each service provider (e.g., operator) develops its own application by following the guidelines in TS 24.371. Its subscriber downloads the application and connects to the service and other subscribers only within the same service. Detailed C-plane signalling is left open to each operator’s design. In contrast, this study tries to identify a new C-plane signalling in detail (as an interface specification) to the extent that client implementations based on it have enough interoperability. This realizes connectivity to any operators or roaming services for new XR real-time communications. Operators can provide the interface common to them according to well-defined C-plane signalling specifications. Clients can connect to any operators via the interface (see Figure 4.1-1).

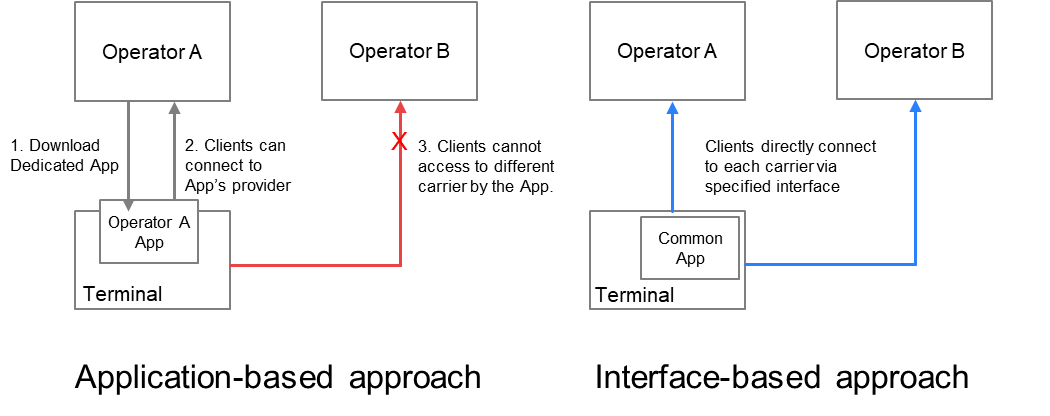


Figure 4.1-1: Two approaches for defining specifications and their application connectivity

## 4.2. High-level network model and target interfaces

The eiRTCW signalling protocol studied in this study is intended for various media session control on the following interfaces:

- UNI: The interface between operator network and UE (e.g., smart phone, content server of the Content Provider).

- NNI: The interface between the two different operator networks, or that between operator network and service provider network.

A UE and a Content Provider can set up a media session by using eiRTCW signalling protocol for session control on the UNI. Figure 4.2-1 shows the high-level network model indicating above interfaces and media sessions established via eiRTCW functional entities (which described in clause 6.2) by using eiRTCW signalling protocol.

There are following benefits to using eiRTCW signalling protocol.

- A UE (including the equipment of Content Provider) which is compliant with the eiRTCW signalling protocol can connect to any Operator Network which complies with the eiRTCW signalling protocol and set up a media session with the media resources (including UEs) in the Operator Network, based on the same signalling requirement.

- A UE (including the equipment of Content Provider) which is compliant with the eiRTCW signalling protocol can connect to services provided by other Operator Network or service provider network via NNI, based on the same signalling requirement.

- Content Providers can set up an operator assisted media session (e.g., media session with QoS) with UEs connected to the Operator Network via the NNI, by connecting to the operator network via the NNI.

- Service Providers can set up an operator assisted media session (e.g., media session with QoS) with UEs connected to the Operator Network via the NNI, by connecting to the operator network via the NNI.

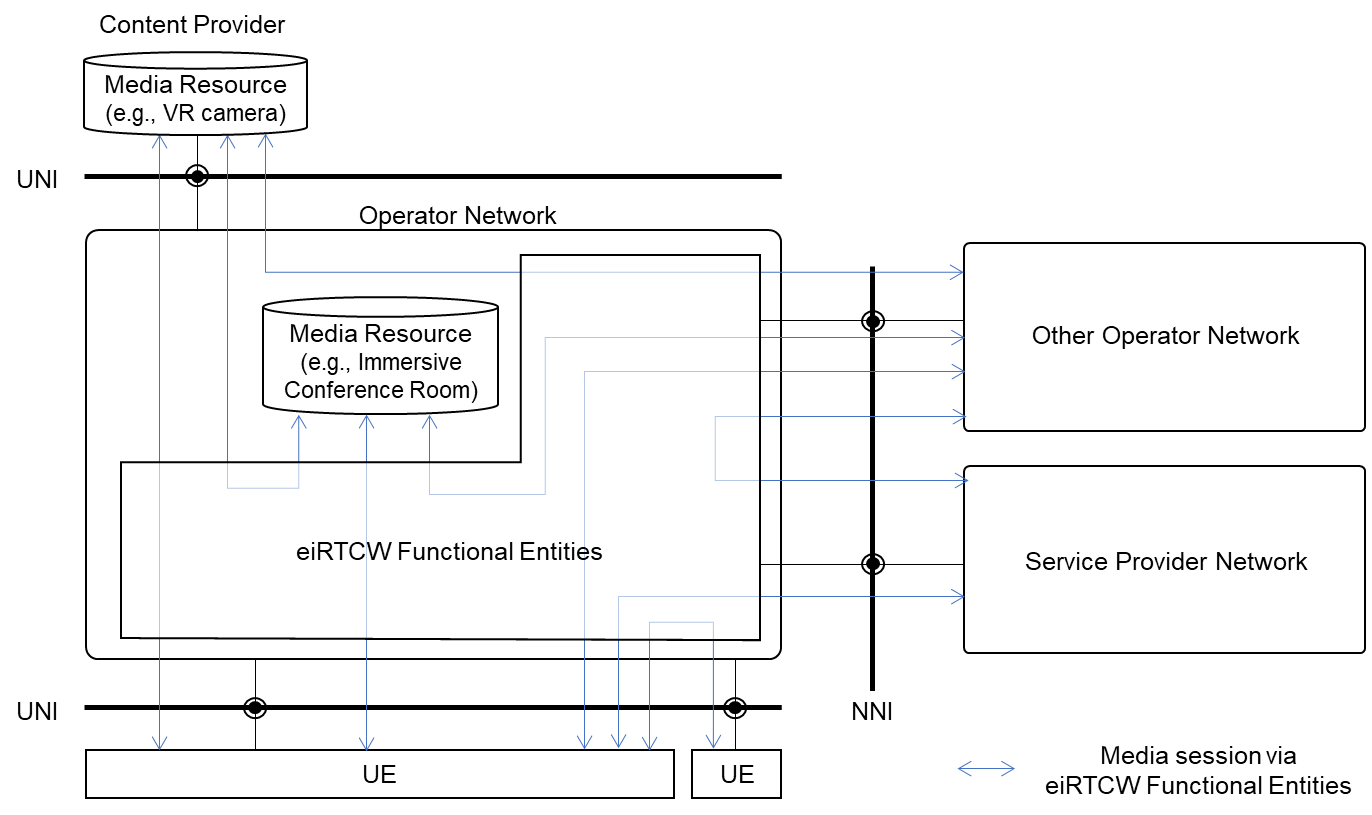


Figure 4.2-1: High-level network model and interfaces

**<Terminology>**

**User Equipment (UE)**: It indicates the user equipment and servers acting as user equipment such as a content server of a content provider. User equipment includes an WebRTC endpoint supporting eiRTCW signalling protocol.

**Operator**: Mobile and Fixed network operator who provides telecommunication services.

**Service Provider (SP)**: 3rd party service provider who connects its service to operator network via NNI. OTT service is one of the typical services provided by service provider. Network Operator is excluded from the definition of this terminology in this document.

**Content Provider (CP)**: 3rd party service provider who connects its service to operator network via UNI. Network Operator is excluded from the definition of this terminology in this document.

**UNI**: User-to-Network Interface. The interface between UE and Network.

**NNI**: Network-to-Network Interface. The interface between two different Networks.

## 4.3 C-plane Signalling comparison

The C-plane signalling can be expressed as follows. Now, there are roughly four possible methods, classified in terms of their protocol stacks (see Figure 4.3-1).

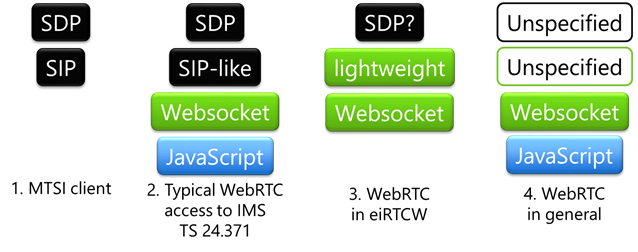


Figure 4.3-1: Comparison of protocol stacks

The first method is MTSI-based, using SIP and SDP. General C-plane signalling requirements for conversational services can be covered by SIP. Interoperability is fine with the existing 5G core network. It is to be treated in IMS-based AR Conversational Services (IBACS).

The second is the method specified in TS 24.371. It enables the WebRTC clients to communicate over an IMS-based core network; only the interfaces for downloading dedicated applications and the signalling path using WebSocket are specified for C-plane signalling. Ordinary implementations adopt SIP-like protocols over WebSocket. In most cases, it is partially SIP-compliant or tightly coupled with SIP to adapt WebRTC clients in IMS domain.

The third method is an alternative to the second method that uses SIP-like protocol over WebSocket. The third method uses another signalling protocol over WebSocket, but SIP-decoupled approaches are investigated. It can be more lightweight, omitting features that is not used in XR conversational. Some constraints on SDP are necessary for interoperability. Non-browser based implementations are also in the scope. This method is the main subject of this study, FS\_eiRTCW.

The other is a general WebRTC protocol stack that is not specified and left open to the users (i.e., service providers). C-plane may be SIP, XMPP, http, etc. A general WebRTC application uses SDP syntax compliant to RFC 4566 for its internal representation, when setting the local and remote descriptions. C-plane protocol may have its own on-the-wire format for SDP, which can be constructed from SDP and be serialized out to SDP.

Editor’s Note: The reason why WebRTC signalling is necessary

Editor’s Note: Comparison interworking between WebRTC signalling and existing SIP

# 5 Key Issue

## 5.1 General

This clause describes the key issues of eiRTCW.

## 5.2 Key Issue #1: Architecture for eiRTCW

Editor’s note: Key issue need to be described. The title of this key issue is re-considered corresponding to concluded issues.

## 5.3 Key Issue #2: Requirements for C-Plane Signalling

Editor’s note: Key issue need to be described. The title of this key issue is re-considered corresponding to concluded issues.

## 5.4 Key Issue #3: Requirements for U-plane Signalling

Editor’s note: Key issue need to be described. The title of this key issue is re-considered corresponding to concluded issues.

## 5.5 Key Issue #4: Interworking with IMS Network

Editor’s note: Key issue need to be described. The title of this key issue is re-considered corresponding to concluded issues.

## 5.6 Key Issue #5: Tethered Cases

Editor’s note: Key issue need to be described. The title of this key issue is re-considered corresponding to concluded issues.

Editor’s Note: SmarTAR-related clause;  
Identify enhancements for E2E QoS realizations over 5G systems for communications between MNOs and WebRTC clients operating over non-5G links (e.g., Wi-Fi) using WebRTC-based transport. This also includes communication between WebRTC clients operating on tethering/tethered devices.

## 5.7 Key Issue #6: Security Considerations

Editor’s note: Key issue need to be described. The title of this key issue is re-considered corresponding to concluded issues.

Editor’s Note: Considerations that the third-party access to the operator network need to be controlled with SLAs and with secure access to protect the underlying network resources.  
- Rate limiting  
- Abuse protection  
- Security measures

## 5.8 Key Issue #7: Related Groups Considerations

Editor’s note: Key issue need to be described. The title of this key issue is re-considered corresponding to concluded issues.

Editor’s Note: Identify collaboration formation with other WGs in 3GPP and SDOs including IETF and W3C.

# 6 Solutions

## 6.1 General

This clause describes the solutions for key issues in clause 5.

Editor’s note: The title of following clauses in clause 6 are tentative for the current description. These can be re-considered corresponding to the modification of the solution.

Table 6.1-1: Mapping of Solutions to Key Issues

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Solutions | Key Issues | | | | | | |
| #1 | #2 | #3 | #4 | #5 | #6 | #7 |
| **#1:** **Architecture for eiRTC** | X |  |  |  |  |  |  |
| **#2: Requirements for C-Plane Signalling** |  | X |  |  |  |  |  |
| **#3: Requirements for U-plane Signalling** |  |  | X |  |  |  |  |
| **#4: Interworking with IMS Network** |  |  |  | X |  |  |  |
| **#5: Tethered Cases** |  |  |  |  | X |  |  |
| **#6: Security Considerations** |  |  |  |  |  | X |  |
| **#7: Related Groups Considerations** |  |  |  |  |  |  | X |

## 6.2 Solution #1: Possible Architecture for eiRTCW

### 6.2.1 Solution description

This solution addresses key issue #1.

This clause identifies what functional entities and reference points are needed for WebRTC, and proposes a possible architecture integrated with 5GC (5G Core Network) defined in 3GPP TS 23.501 [TS23.501].

Editor’s note: Terminology of common functional entities should be aligned with the architecture in GA4RTAR.

### 6.2.2 Functional Entities for WebRTC

#### 6.2.2.1 General

This clause enumerates functional entities in terms of 1) WebRTC specifications, 2) WebRTC implementations, and 3) providing inter-operator services.

1) functional entities that are essential for this study and already defined in IETF RFCs or 3GPP specifications concerning WebRTC (see clause 6.2.2.2).

2) functional entities that are not directly specified in WebRTC-related specifications in IETF RFCs or 3GPP specifications but are considered to be widely implemented for realizing WebRTC services; they are essential for this study (see clause 6.2.2.3).

3) functional entities that may be specifically required for inter-operator or 3rd-party collaboration services if modification of signalling and termination of media on network boundaries are needed (see clause 6.2.2.4).

#### 6.2.2.2 Functional Entities defined in WebRTC specifications

##### 6.2.2.2.1 UE (User Equipment)

User Equipment (UE) contains a user agent function for WebRTC. The user agent function is equivalent to "WebRTC Endpoint" as described below.

For the purposes of the present document, the following terms and definitions given in IETF RFC 8825 [RFC8825] apply:

**WebRTC Endpoint**: Either a WebRTC browser or a WebRTC non-browser. It conforms to the protocol specification.

**WebRTC Browser (also called a "WebRTC User Agent" or "WebRTC UA")**: Something that conforms to both the protocol specification and the JavaScript API[W3C.WD-webrtc].

**WebRTC Non-Browser**: Something that conforms to the protocol specification but does not claim to implement the JavaScript API. This can also be called a "WebRTC device" or "WebRTC native application".

##### 6.2.2.2.2 WSF (WebRTC Signalling Function)

The WebRTC Signalling Function (WSF) is a functional entity that is responsible for WebRTC signalling mechanism including capability exchange and management of call sessions between UEs and the network. This functional entity is described as "Servers" or "Web Server" in IETF RFC 8825 [RFC8825] clause 3. Each operator or 3rd-party in this study is assumed to have their own WSF in their network.

#### 6.2.2.3 Functional Entities widely implemented for WebRTC

##### 6.2.2.3.1 WMCF (WebRTC Media Centre Function)

The WebRTC Media Centre Function (WMCF) is a functional entity that performs media signal processing. WMCF terminates media signals and performs media processing (e.g., mixing, selective forwarding, transcoding) which are required for conferencing applications. It may also perform decryption and encryption of media packets if DTLS, SRTP, or TLS is used for a transport layer. It also has the function of storing contents (including text or other static material as well as audio and video) and providing them to the UE.

In cases, WMCF performs as a simple media relay function. It simply relays media data packets and supports IP packet connectivity. When UE behave as ICE Agents defined in IETF RFC 8445 [RFC8445], WMCF may be either STUN servers defined in IETF RFC 8489 [RFC8489] for connectivity check or TURN servers defined in IETF RFC 8656 [RFC8656] for relaying media data packets. This functional entity facilitates NAT traversal of UE and the connectivity between UE and other network functions.

This functional entity is generally implemented in WebRTC Multipoint Control Unit (MCU) or Selective Forwarding Unit (SFU).

##### 6.2.2.3.2 CSF (Conference Supporting Function)

The Conference Supporting Function (CSF) provides the following functionality:

- Conference session management, i.e., "CRUD" operation – create, read, update, delete of conference instances.

- Providing supplementary files (e.g., icon images of participants, and shared documents) via best-effort transport different from the channels for real-time media.

- Storage of user subscription data specific to MNO’s WebRTC services.

NOTE: In this study, it is assumed that a single user (i.e., identity) and its subscription data (associated with the identity) are assigned, owned, and managed by both MNO and service provider independently. The two identities have a link with each other via some technique. User subscription data specific to Service Provider’s services are stored in their networks.

- Authorization Endpoint and Token Endpoint of OAuth 2.0 described in IETF RFC 6749[RFC6749] for establishing authentication linkage between MNO’s ID and Service Provider’s ID.

Editor’s note: OAuth token will be used to C-Plane authentication at WSF and U-Plane authentication at WMCF in the Service Provider model. STUN/TURN authentication with OAuth token is defined in IETF RFC 7635[RFC7635]. Portal http(s) servers of WebRTC services provide this function in general implementations.

#### 6.2.2.4 Functional Entities needed for inter-operator services

##### 6.2.2.4.1 WNSGF (WebRTC NNI Signalling Gateway Function)

The WebRTC NNI Signalling Gateway Function (WNSGF) is located at the boundary of the networks where different operators or third-party network connects.

Each operator or 3rd-party has its own WebRTC Signalling Functions (WSF) so that WSFs are connected to each other with border control functions such as security, policy management, charging, etc. WNSGF is inserted into "Signaling Path" in Figure 2 of RFC 8825 and responsible for border control functions and supports session establishment between disparate address realms' networks.

##### 6.2.2.4.2 WNMGF (WebRTC NNI Media Gateway Function)

The WebRTC NNI Media Gateway Function (WNMGF) is a media relay located at the boundary of the networks where different operators or third-party network connects. It is the function responsible for the border control and transport of media data packets between different networks. It may also transcode media data packets.

### 6.2.3 Possible Architecture

#### 6.2.3.1 Overview

Figure 6.2.3.1-1 depicts a possible network architecture of this study. It contains the functional entities described in clause 6.2.2 and reference points between the entities.



Figure 6.2.3.1-1: Possible Architecture (from WebRTC’s viewpoint)

WSF and CSF may co-locate in a physical node. WNSGF and WNMGF are also optional when gateway functions are not needed at the network boundary. Terminal authentication needs further study.

#### 6.2.3.2 Reference Points

The reference points shown in Figure 6.2.3.1-1 are described as follows.

Reference points for signalling are called as “Control Plane” or “C-Plane” in this study. Reference points for Media are similarly called as “User Plane” or “U-Plane” in this study.

Editor’s note: The interface names here are tentative. They are used only for the study. The final interface names may be updated based on the outcome of 5G\_AREA.

Reference Points for Signalling:

**Rs-u**: Reference Point between a WSF and a UE.

**Rs-i**: Reference Point between a WSF and another WSF in the same network (DN) or between a WSF and a WNSGF.

**Rs-a**: Reference Point between a WSF and a CSF.

**Rs-n**: Reference Point between a WNSGF and another WNSGF in an external network.

Reference Points for Media:

**Rm-u**: Reference Point between a WMCF and a UE.

**Rm-i**: Reference Point between a WMCF and another WMCF in the same network (DN) or between a WMCF and a WNMGF.

**Rm-n**: Reference Point between a WNMGF and another WNMGF in an external network.

Reference Points for signalling nodes to control media nodes:

**Mc-i**: Reference Point between a WSF and a WMCF.

**Mc-r**: Reference Point between a WNSGF and a WNMGF.

Other Reference Points:

**Rh-u**: Reference Point between a CSF and UE.

Detailed protocol for each reference point will be discussed in clause 6.

## 6.2.4. Target use cases from network view

eiRTCW signalling protocol supports the following use cases of media session set up from network view.

<Media session set up with media resource which connected to the same Operator>

a. UE - Media Resource (served by the same Operator)

b. UE - Media Resource (served by the same Operator) - UE (CP)

<Media session set up with media resource via NNI>

c. UE - Media Resource (served by other Operator)

d UE - Media Resource (served by an SP)

e. UE (served by other Operator) – Media Resource - UE (CP)

f. UE - Transit entity (served by other Operator) - Media Resource (served by an SP)

<Media session set up between UEs>

g. UE - UE (served by the same Operator) without media gateway

h. UE - UE (served by other Operator) without media gateway

i. UE - UE (CP) without media gateway

j. UE (connected to other Operator) - UE (CP) without media gateway

The overviews of these use cases are described below.

NOTE: Media Resource of Content Provider is treated as UE.

a. **UE - Media Resource (served by the same Operator)**:  
UE establishes a media session with a media resource (e.g., Immersive conference room) served by the same operator. Figure 6.2.4-1 shows an example that UE\_A and UE\_B establish media sessions with the media resource to an immersive conference room to communicate with each other.

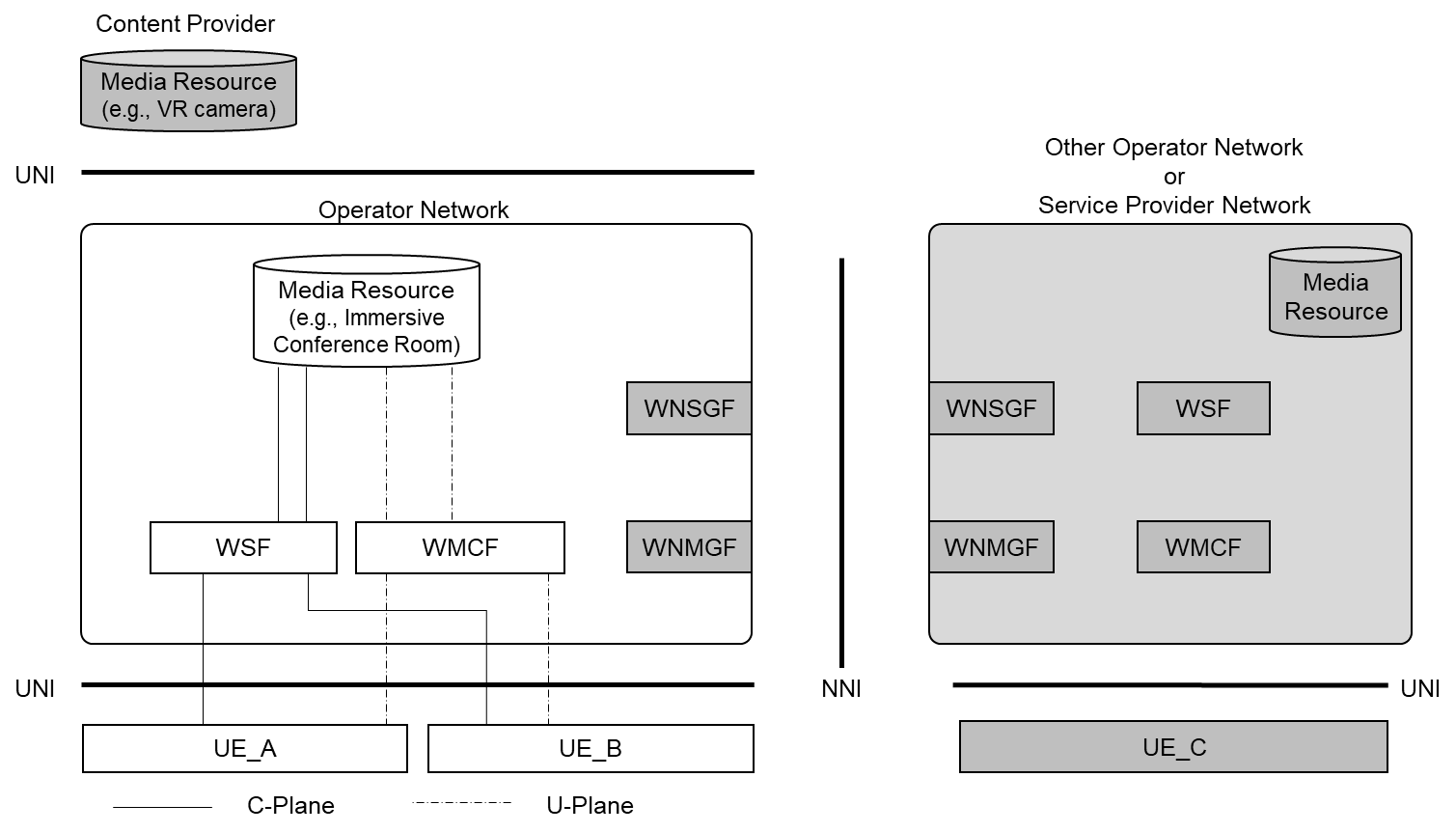


Figure 6.2.4-1: Media session: UE - Media Resource (served by the same Operator)

b. **UE - Media Resource (served by the same Operator) - UE (CP)**:  
A UE establishes a media session with a media resource (e.g., 3D video content) served by a CP which connected to the same Operator, via a media gateway (such as WMCF).

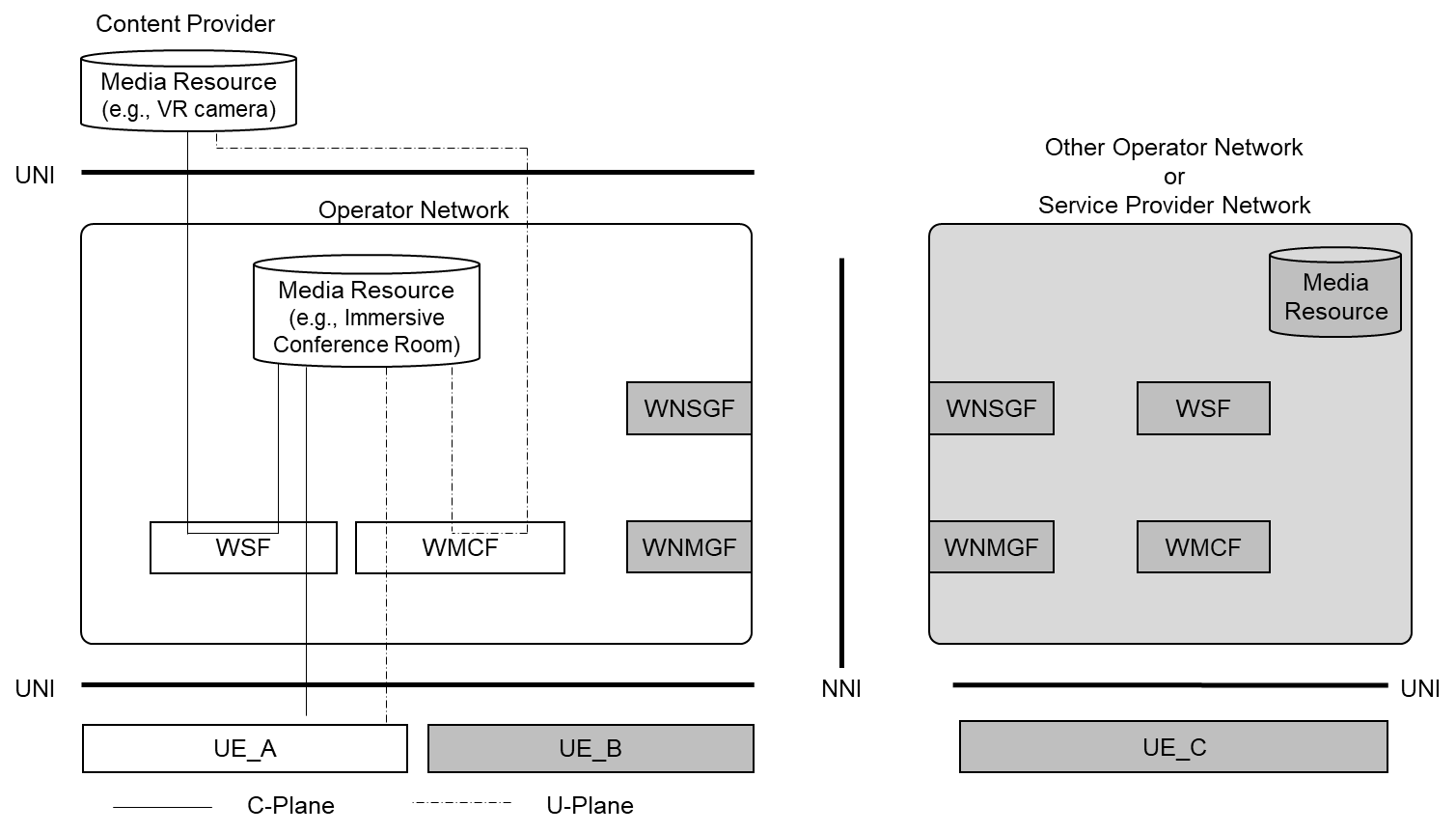


Figure 6.2.4-2: Media session: UE - UE (CP) - Media Resource (served by the same Operator)

c. **UE -** **Media Resource (served by other Operator)**:  
A UE establishes a media session with a media resource (e.g., Immersive conference room) served by the operator that different from the network which the UE is connected to. In this scenario, the C-Plane signalling message and media session stream are sent over the NNI. Other UEs can connect to the media resource as same as pattern a.

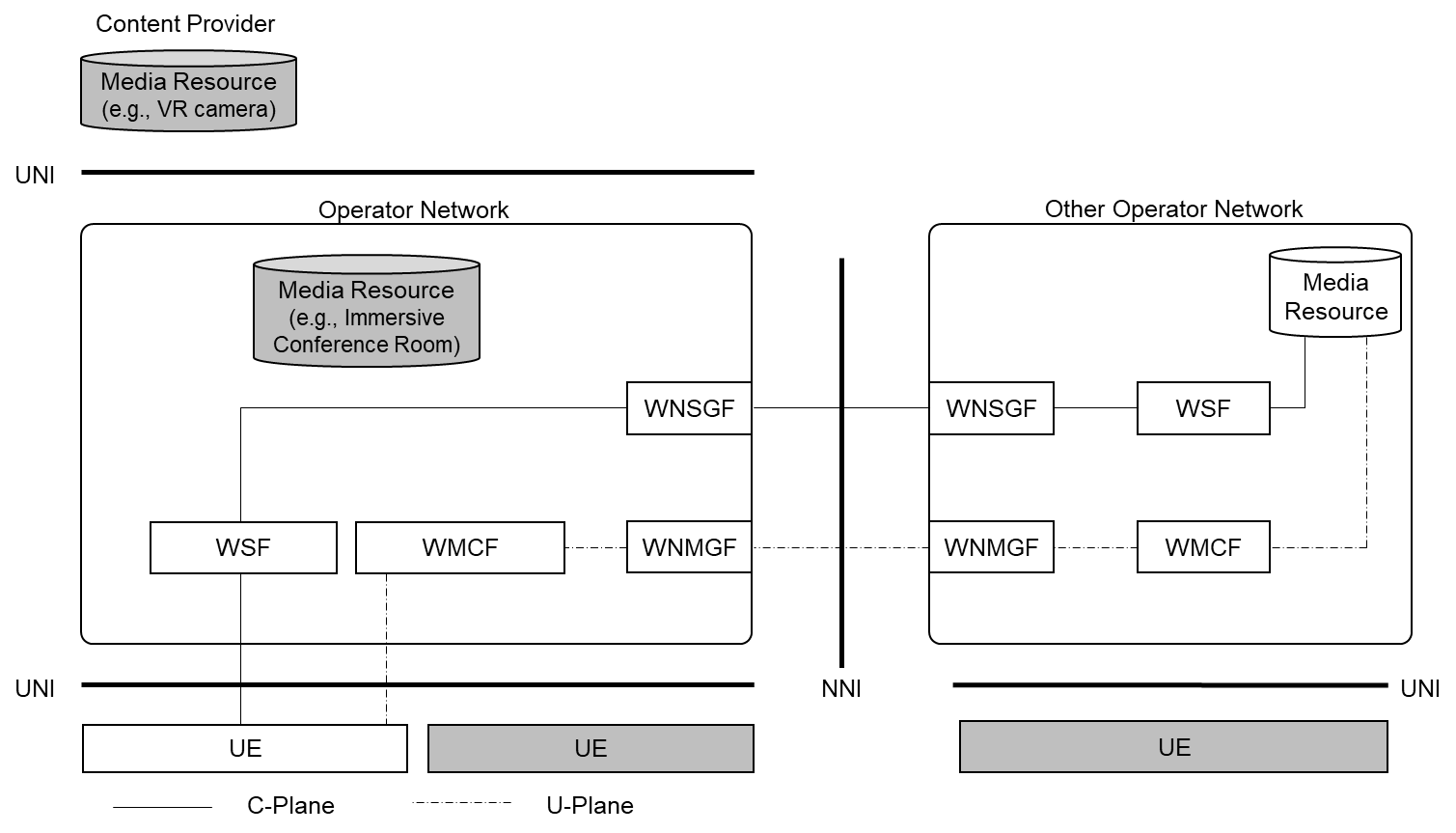


Figure 6.2.4-3: Media session: UE - Media Resource (served by other Operator)

d. **UE - Media Resource (served by an SP)**:  
A UE establishes a media session with a media resource (e.g., Immersive conference room) served by an SP. In this scenario, the C-Plane signalling message and media session stream are sent over the NNI.

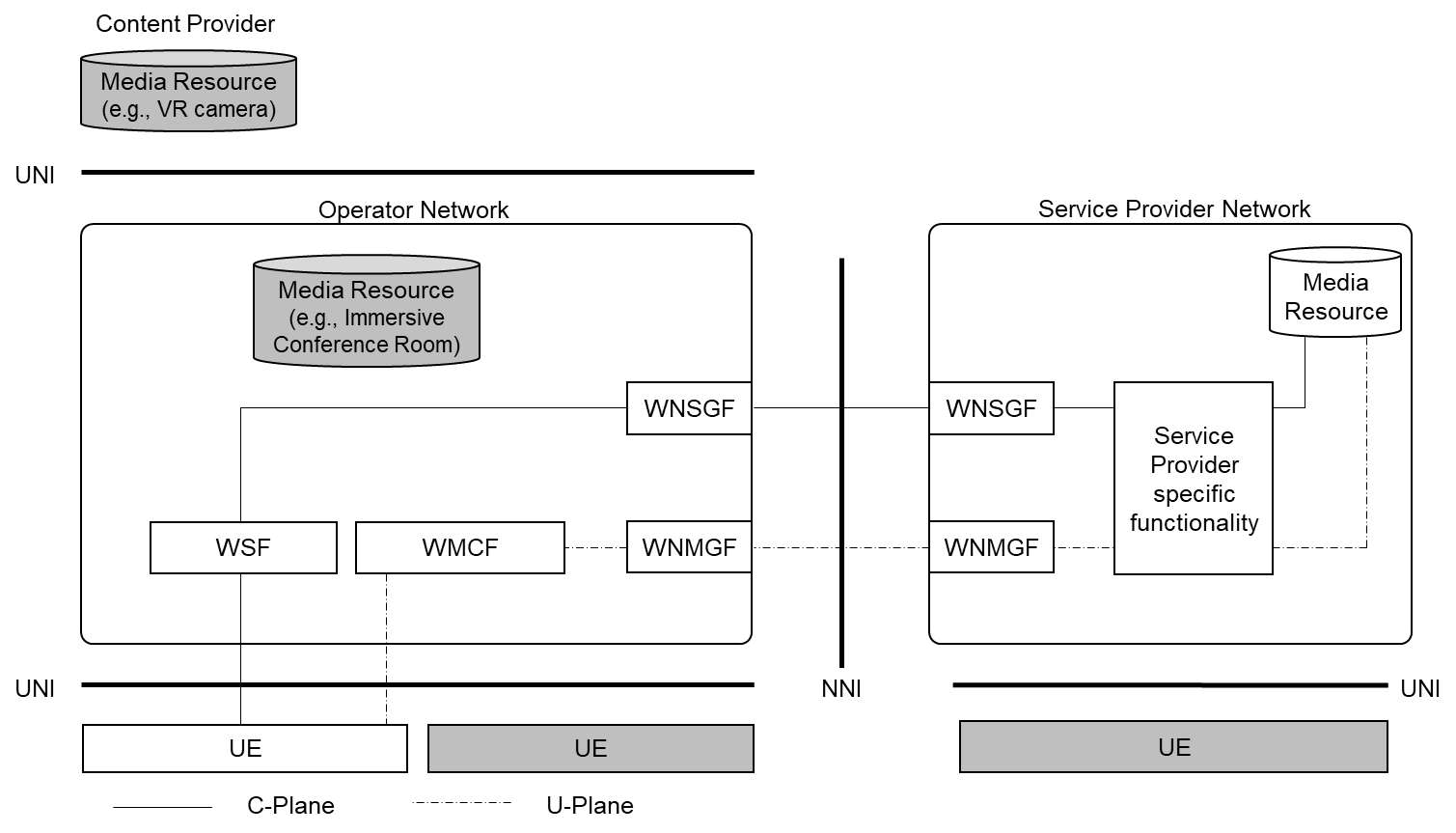


Figure 6.2.4-4: Media session: UE - Media Resource (served by an SP)

e. **UE - Media Resource (served by other Operator) - UE (CP)**:  
A UE in the other operator network and UE (CP) establishes a media session with a media resource (e.g., Immersive conference room) served by an operator network which the UE (CP) connected to. In this scenario, the C-Plane signalling message and media session stream are sent over the NNI.

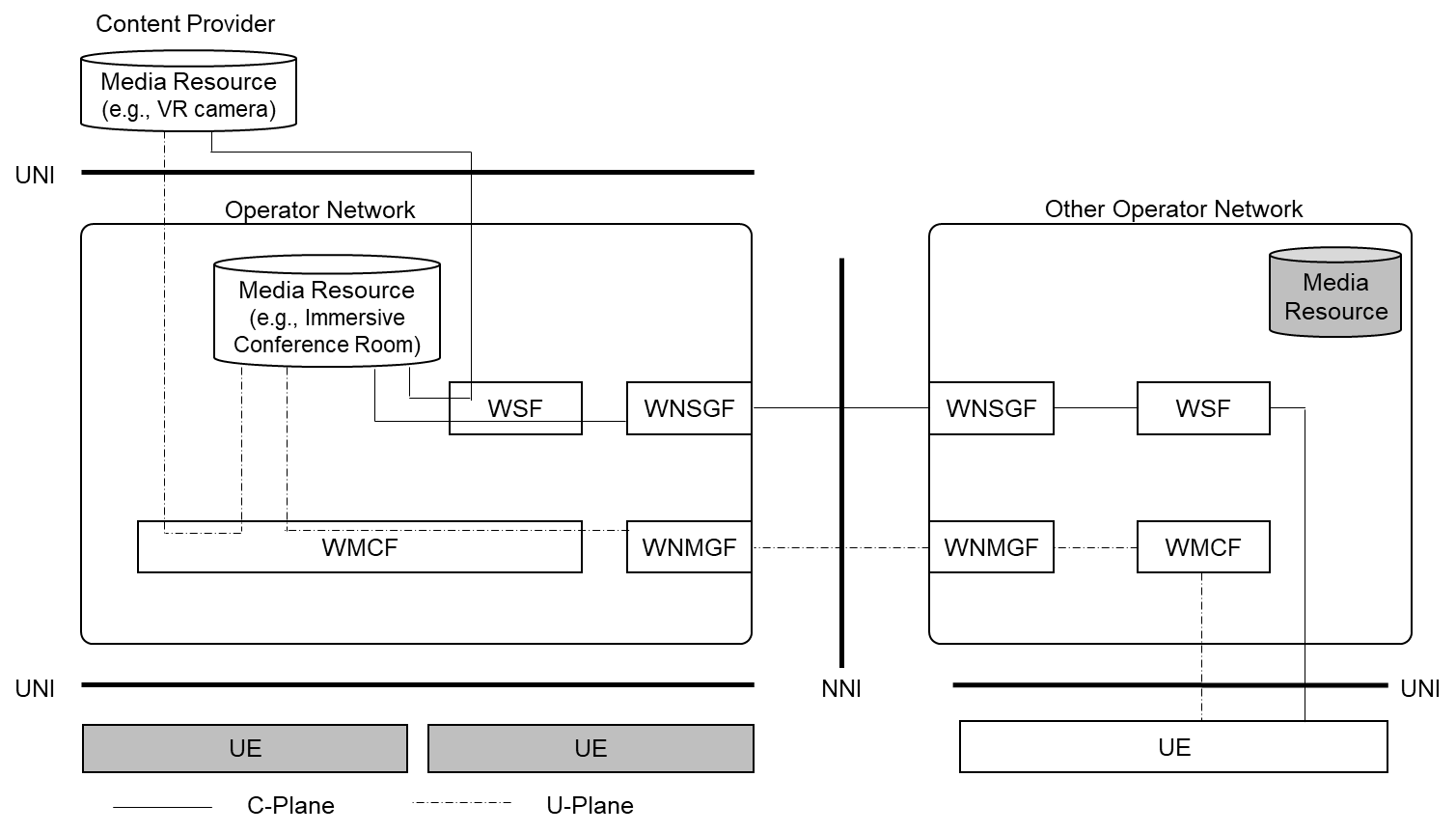


Figure 6.2.4-5: Media session: UE – Media Resource (served by other Operator) - UE (CP)

f. **UE - Transit NW (other Operator) - Media Resource (served by an SP)**:  
A UE establishes a media session with a media resource (e.g., Immersive conference room) served by an SP via transit NW (other operator). In this scenario, the C-Plane signalling message and media session stream are sent over the two different NNIs.

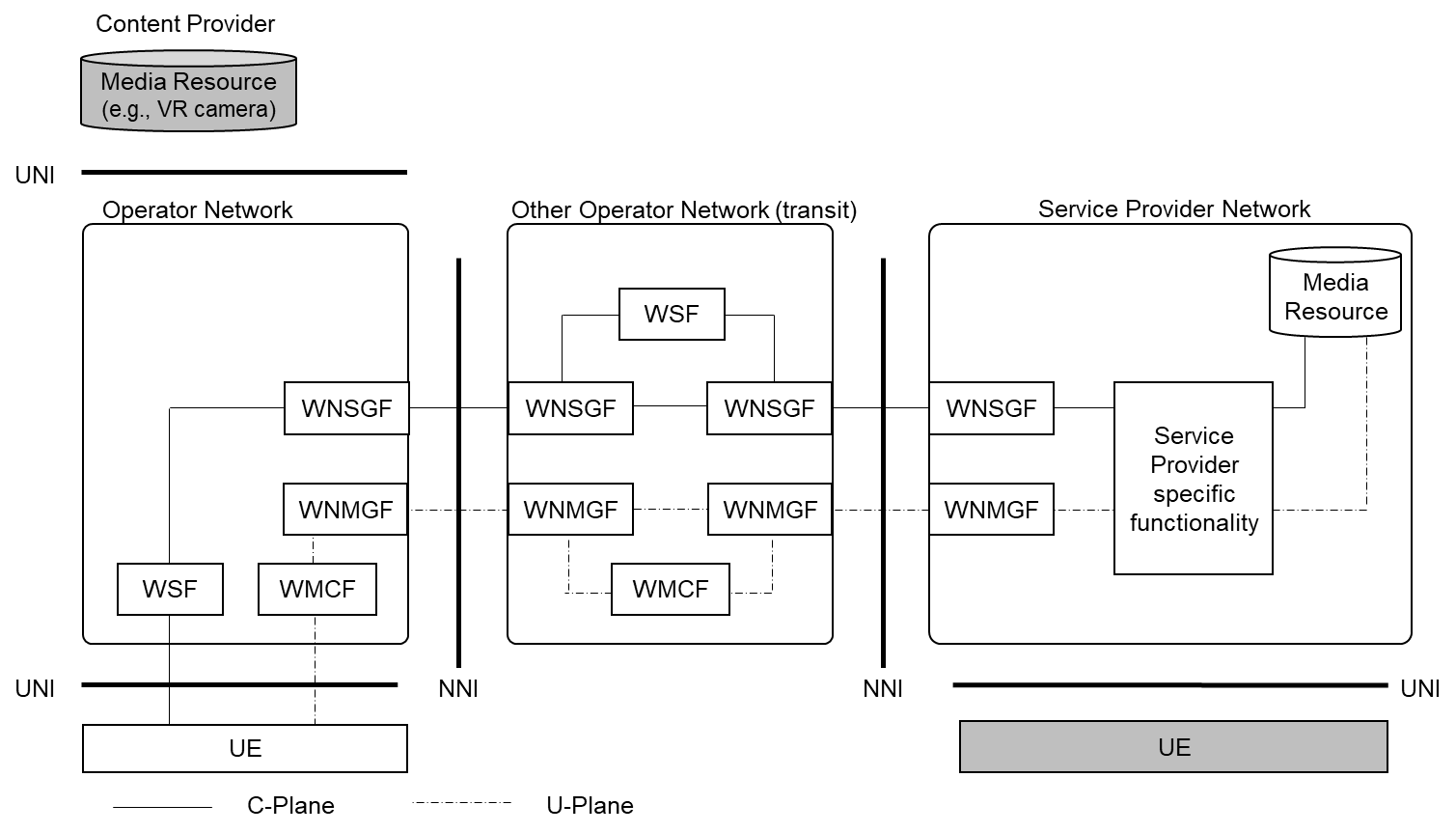


Figure 6.2.4-6: Media session: UE - Transit NW (other Operator) - Media Resource (served by an SP)

g. **UE - UE (served by the same Operator) without WMCF**:  
A UE establishes a media session (e.g., voice chat) with another UE served by the same operator, without using WMCF.

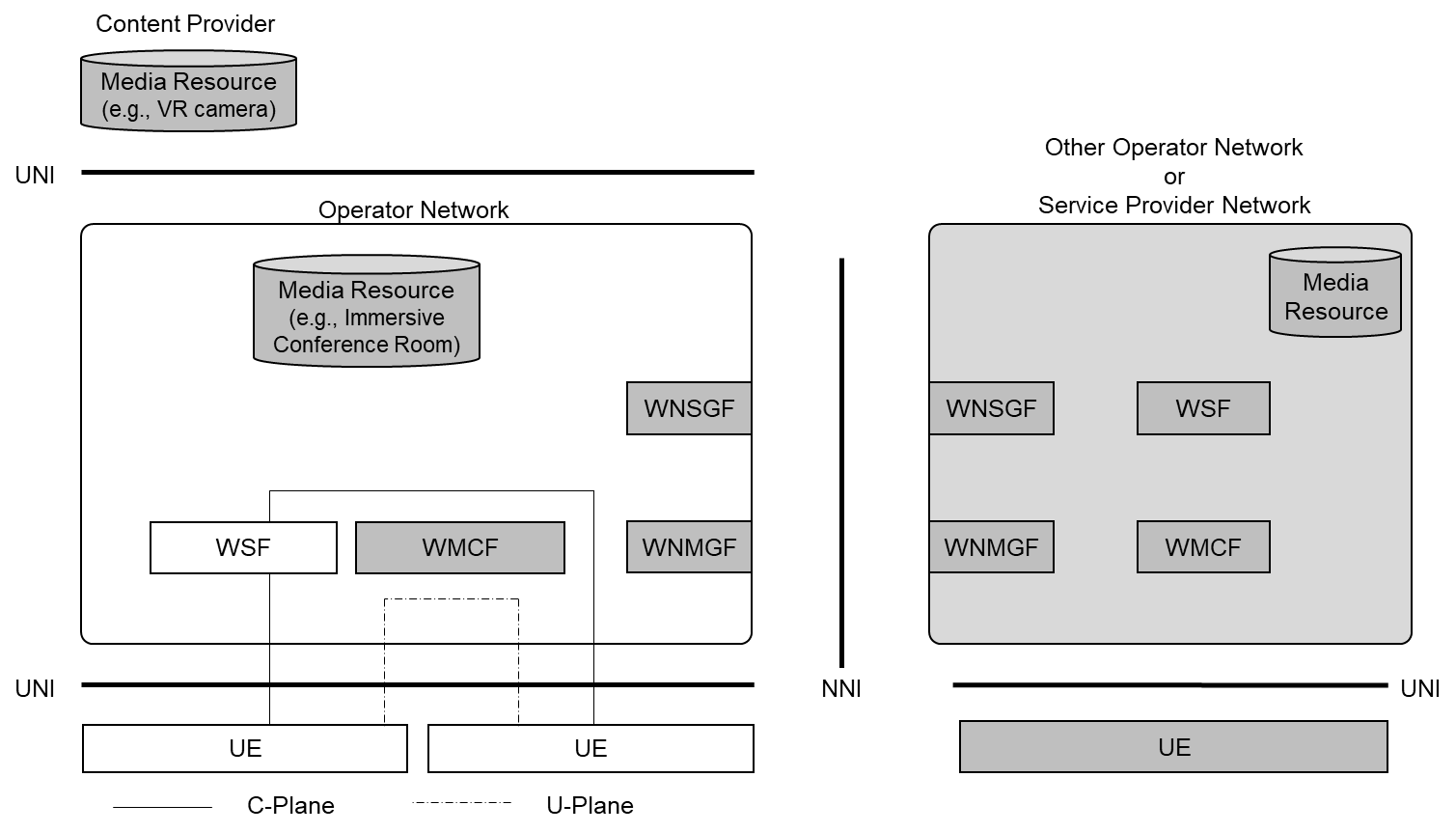


Figure 6.2.4-7: Media session: UE - UE (served by the same operator) without WMCF

h. **UE - UE (served by other Operator) without WMCF**:  
A UE establishes a media session (e.g., voice chat) with another UE served by the different operator, without using WMCF. In this scenario, the C-Plane signalling messages and media session stream are sent over the NNI.

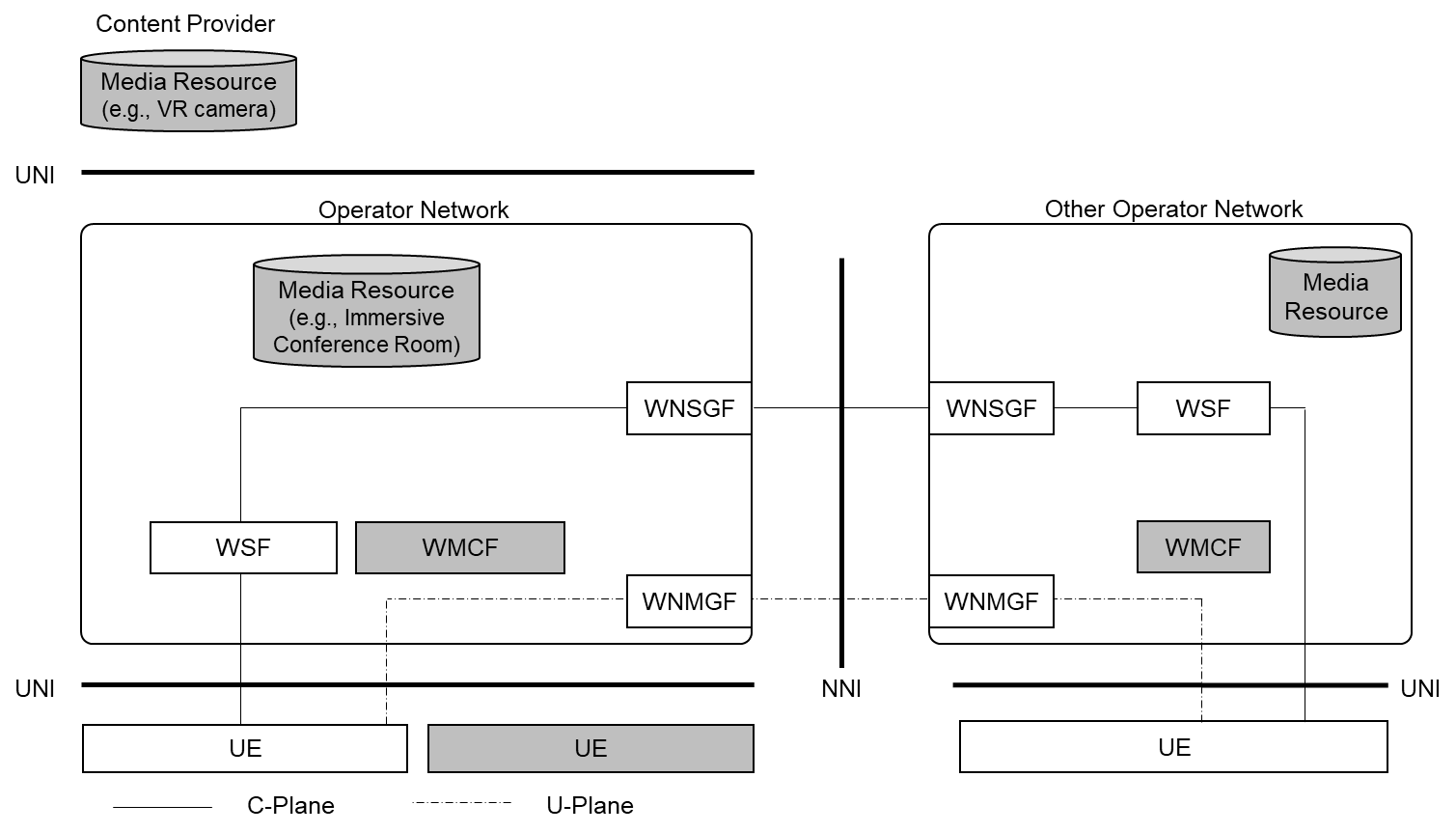


Figure 6.2.4-8: Media session: UE - UE (served by other Operator) without WMCF

i. **UE - UE (CP) without WMCF**:  
A UE establishes a media session with a UE (e.g., 3D video content) served by a CP which connected to the same operator, without using WMCF.

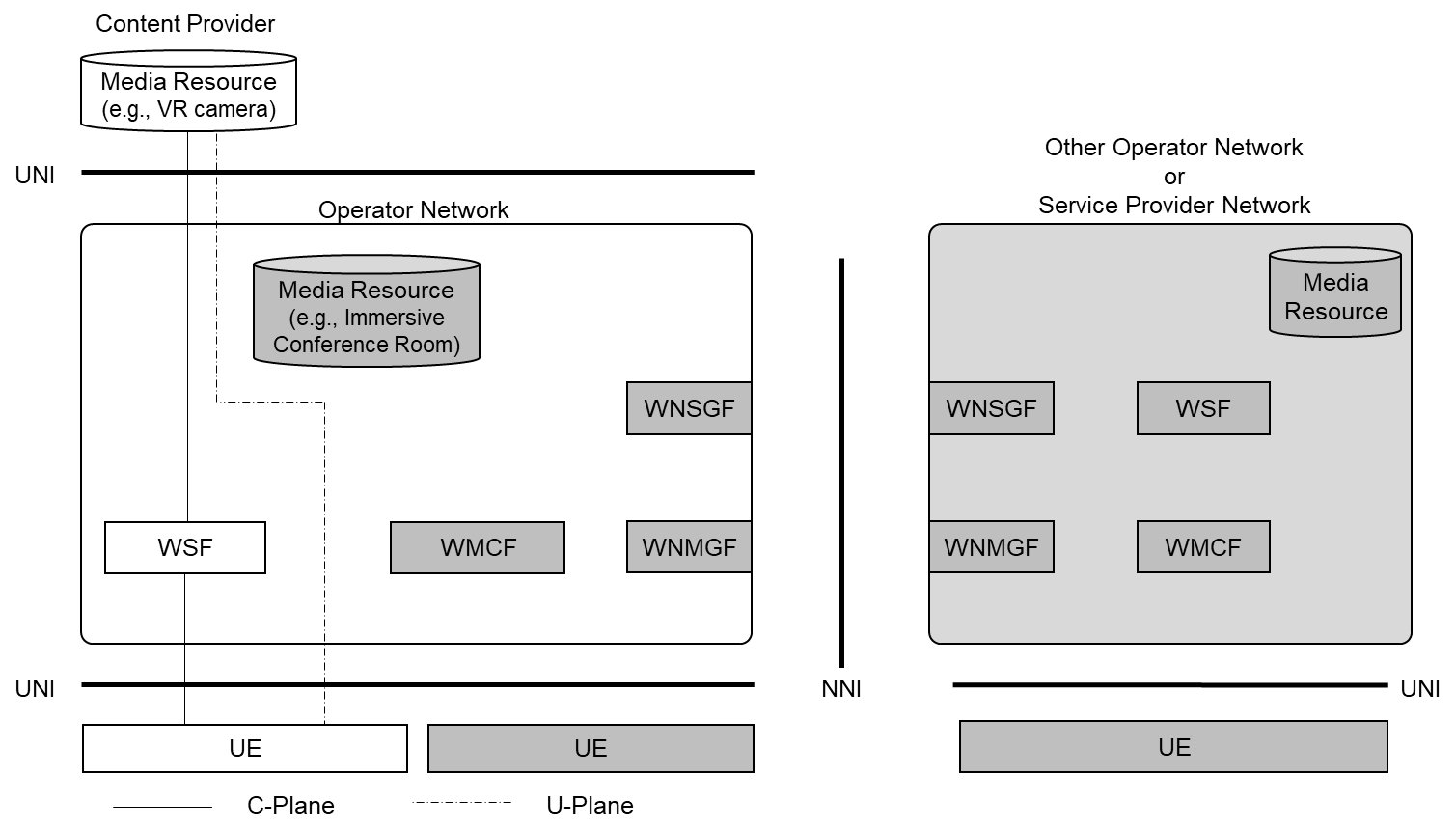


Figure 6.2.4-9: Media session: UE - UE (CP) without WMCF

j. **UE (connected to other Operator) - UE (CP) without WMCF**:  
A UE establishes a media session with a UE (e.g., 3D video content) served by a CP which connected to the different operator, without using WMCF. In this scenario, the C-Plane signalling messages and media session stream are sent over the NNI.

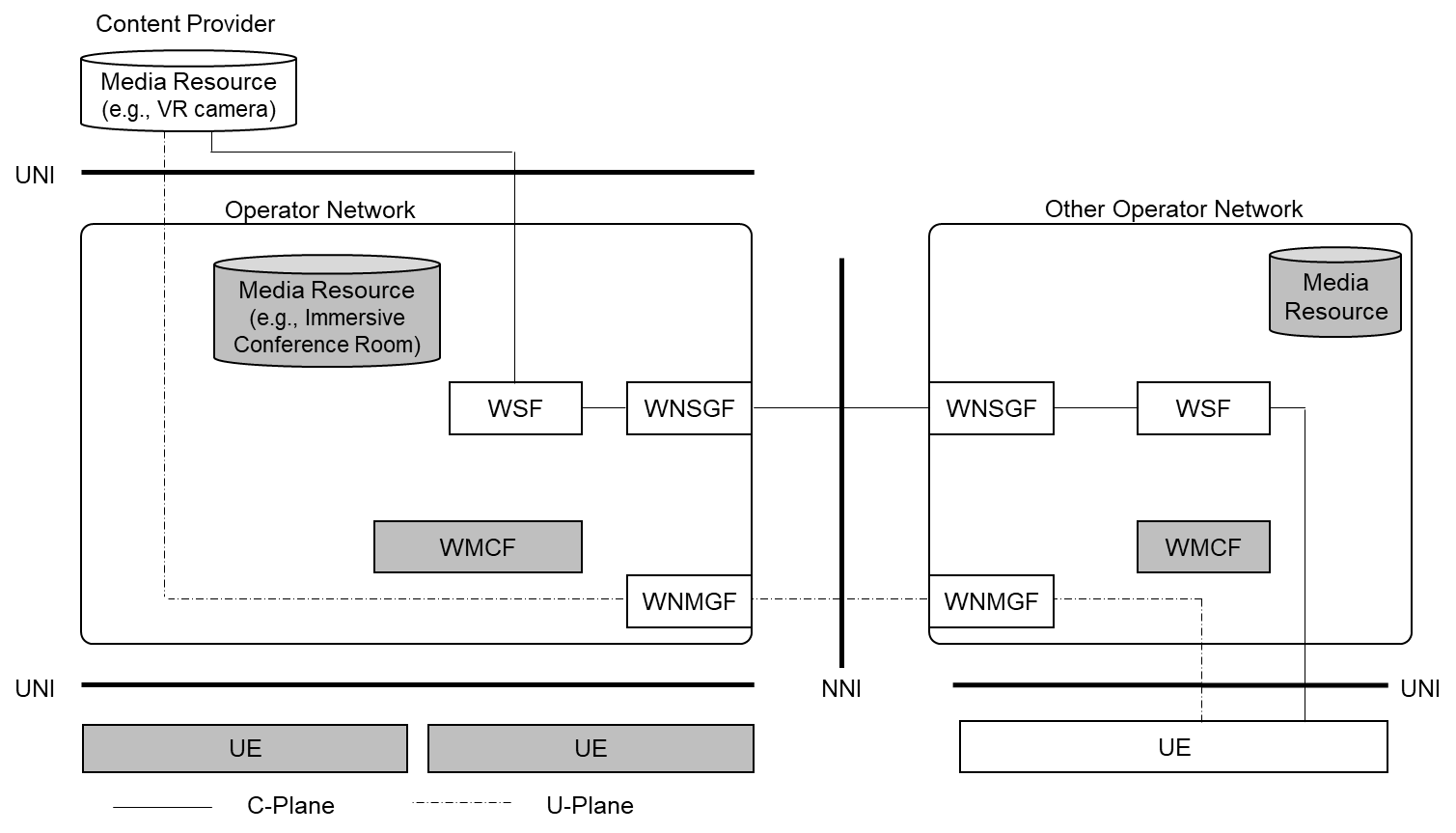


Figure 6.2.4-10: UE (connected to other Operator) - UE (CP) without media gateway

### 6.2.5 Possible Architecture with 5GC

#### 6.2.5.1 Overview

A possible architecture in terms of WebRTC view is described in clause 6.2.3. This clause shows a solution for integrating the pure WebRTC architecture with 5GC (5G core).

NOTE: "pure WebRTC" means the original WebRTC described in IETF work, which basically does not take into account domain specific functions or features (e.g., mobile networks).

#### 6.2.5.2 Mapping of Functional Entities

##### 6.2.5.2.1 General

This clause identifies the mapping of functional entities shown in Figure 6.2.3.1-1 into 5GC functional entities defined in 3GPP TS 23.501 [TS23.501].

Currently, the mapping of two WebRTC functional entities are discussed. The mapping of other WebRTC functional entities needs further study.

##### 6.2.5.2.2 WSF and AF

WSF is interconnected with UE and is expected to process the following:

1) authenticate a UE.

2) setup a WebRTC session required by a UE, which may be in another network.

3) manage QoS for the media path of a WebRTC session.

WSF interacts with 5GC functions and UE to perform 1) and 3) as the following:

1) WSF retrieves the identity of a UE via EDGEAPP framework, then authenticates the UE.

3) WSF requests PCF to manage QoS for the media path of a WebRTC session retrieved from WebRTC C-Plane signals.

Editor’s note: The detail description of EDGEAPP framework is for further study.

Additionally, these processes are close to the processes of IMS functions such as P-CSCF and S-CSCF defined in 3GPP TS 23.228 [TS23.228]. The process of 1) is performed by S-CSCF and UDM, and 3) is performed by P-CSCF and PCF.

WSF can be mapped into “AF (Application Function)” of 5GC according to the definition of AF in 3GPP TS 23.501 clause 5.2.10 due to the following reasons:

- WSF interacts with the 3GPP Core Network to provide services.

- The interaction between WSF and PCF/UDM is close to IMS interactions with 5GC.

NOTE: Process 2) is discussed in clause 6.2.5.2.3.

Editor’s note: A possible idea is as follows:  
User subscription data for MNO specific to WebRTC are stored in CSF’s backend DB. User subscription data for OTT are placed in OTT data network. The linkage between MNO’s subscription data for WebRTC and UE’s identity as a 5G terminal is realized using EDGEAPP features. The linkage between MNO’s subscription data and OTT’s subscription data is realized with OAuth or OIDC (This part needs further study)

##### 6.2.5.2.3 WNSGGF

###### 6.2.5.2.3.1 Overview

This clause identifies the mapping of WNSGF to a 5GC functional entity. There are a couple of possibilities currently identified. The following two 5GC functional entities can be mapped from WNSGF:

- NEF (see clause 6.2.5.2.3.2)

- SEPP (see clause 6.2.5.2.3.3)

As another possibility, it may be appropriate that WNSGF is mapped to a brand new functional entity (e.g., Interconnection Border Control Function: IBCF). The exact mapping of WNSGF needs further study.

###### 6.2.5.2.3.2 WNSGF and NEF

When WSF is mapped into an AF, WNSGF can be mapped into an NEF (Network Exposure Function) due to the following reasons:

- When WSF processes 2) of clause 6.2.5.2.2 and the session relates to other operator’s network, WSF (mapped to an AF) of operator A is requested to interact with WNSGF on the boundary of operator B to communicate with WSF (mapped into an AF) in operator B due to operator B’s policy. In this model, the relationship between WSF (in A) and WNSGF (in B) is close to the relationship between AF and NEF described in 3GPP TS 23.501 clause 6.2.10.

- The major function of WNSGF is close to the former three functionalities described in 3GPP TS 23.501 clause 6.2.5.0; WNSGF exposes WSF’s WebRTC signalling capability and events. WNSGF interworks with WebRTC signalling between Rs-i and Rs-n reference points in terms of security and translation of internal-external information.

When WNSGF is mapped into an NEF, the definition of the NEF function may need to be modified as follows:

- Descriptions for the exposure of WSF’s WebRTC signalling capability and the events by WNSGF are added to 3GPP TS 23.501 clause 7.2.8.

- Descriptions for the event exposure details are added to 3GPP TS 23.502 clause 4.15.3.

- Descriptions for the capability exposure details are added to 3GPP TS 23.502 clause 5.2.6.

###### 6.2.5.2.3.3 WNSGF and SEPP

Security Edge Protection Proxy (SEPP) is defined in 3GPP TS 33.501 [TS33.501] and 3GPP TS 23.501. The SEPP is an entity sitting at the perimeter of the PLMN for protecting control plane messages, hiding network topology. The SEPP enforces inter-PLMN security on the N32 interface that is a reference point between a SEPP in one PLMN and a SEPP in another PLMN.

WNSGF is also located at the perimeter of the PLMN and its function is protecting control plane messages and hiding network topology. The function of WNSGF is close to that of SEPP.

The difference is the type of PLMN. WNSGF is located at the edge of inter-HPLMN. On the other hand, SEPP is expected to be used for N32 that lies between HPLMN and VPLMN.

#### 6.2.5.3 Possible Architecture integrated with 5GC

The functional entities shown in Figure 6.2.3.1-1 are connected to 5GC (5G Core) as described in Figure 6.2.5.3-1.



Figure 6.2.5.3-1: Possible Architecture (integrated with 5GC)

WSF is mapped into an AF, and WNSGF is mapped into an NEF.

WSF is interconnected with PCF via N5 interface. WSF manages QoS of real-time media packets and signalling packets via N5 interface. WSF may interact with UDM to authenticate the UE.

Both signalling packets and media packets between UE and the network are transmitted via N6 interface. Signalling packets (C-Plane packets) from UE are transmitted to WSF, and real-time media packets (U-Plane packets) from UE are transmitted to WMCF. C-Plane signals may travel to WNSGF via RS-i, and may travel further to other operator’s WNSGF via RS-n. U-Plane signals may travel to WNMGF via Rm-i, and may travel further to other operator’s WNMGF. (see Figure 6.2.5.3-2)



Figure 6.2.5.3-2: Possible Architecture (from 5GC view, with data flows of C/U-Planes)

#### 6.2.5.4 Mapping to iRTCW Collaboration Scenarios

The following table shows the mapping of functional entities in this study into iRTCW collaboration scenarios. Each box shows the condition (required or not) for MNO. The targets of this study are Scenarios 3 and 4.

Table 6.2.5.4-1: Mapping to iRTCW collaboration scenarios

|  |  |  |  |
| --- | --- | --- | --- |
| Functional Entity | Collaboration Scenario 3 | | Collaboration Scenario 4 |
| 3A / Service Provider provides WebRTC services and MNO assists the services. | 3B / MNO provides WebRTC services only in the MNO’s network | MNO’s WebRTC service interconnects with other MNO’s or Service Provider’s service |
| WSF | Required | Required | Required |
| WMCF | Required (Note 1) | Required | Required |
| CSF | Required | Required | Required |
| WNSGF | N/A (Note 2) | N/A | Required |
| WNMGF | N/A (Note 2) | N/A | Required |
| NOTE 1: Scenario 3A in this table assumes WMCF acts as a TURN relay server at the network boundary between the MNO and the service provider. Further Operator-Assistance models may be introduced.  NOTE 2: Scenario 3A in this table assumes WSF does not communicate with Service Provider’s WebRTC functions and WMCF works as a gateway by itself. Further Operator-Assistance models may be introduced. | | | |

### 6.2.6 Possible Architecture in Software View

#### 6.2.6.1 Overview

There are two types of WebRTC Endpoint as described in clause 6.2.2.2.1; One is “WebRTC Browser”, and the other is “WebRTC Non-Browser”. This clause shows possible architectures for each type of endpoints in terms of software development.

#### 6.2.6.2 WebRTC Browser

A JavaScript (JS) application runs on “WebRTC Browser” type UE. The application runs on a web browser that has capabilities of JS APIs including WebRTC API defined by W3C (see Figure 6.2.6.2-1).



Figure 6.2.6.2-1: Software view of “WebRTC Browser” type endpoint

According to the concept of WebRTC described in IETF RFC 8829 [RFC8829], the procedures and protocols stated in this study are expected to be fully writable only with JS.

All of the functions needed for realizing immersive RTC must be provided via JS API because the only way for JS applications to access devices or networks is utilizing JS API. Functions provided by enablers for immersive RTC must be accessible via web browser’s JS API.

#### 6.2.6.3 WebRTC Non-Browser

An application written in a programming language specific to the UE platform runs on “WebRTC Non-Browser” type UE. (see Figure 6.2.6.3-1)



Figure 6.2.6.3-1: Software view of “WebRTC Non-Browser” type endpoint

NOTE: The programming language and programming APIs used to write applications depend on the UE platform. For example, Java and Android API (SDK) will be selected for Android platform UEs, Swift and its libraries will be selected for iOS platform UEs, and C++ and Win64 API will be selected for Windows platform UEs.

The application is realized in a way other than JS running on a web browser. This study does not state details of the application’s implementation; this study mainly discusses the network interface. The network interface is the same for both “Browser” and “Non-Browser” type UEs.

### 6.2.7 Media connection model

In the original WebRTC design, the communication between UEs is thought to be peer-to-peer (P2P). In most of the existing WebRTC implementations, however, the media connection is not P2P. An intermediate server (or servers) between UEs is used. In the multi-party call, the intermediate server which performs media processing is helpful for a UE because, for a UE, decoding all media from other UEs is a heavy load. Direct full-mesh connections among multiple UEs consumes a lot of network resources. Additionally, such an intermediate server is useful even for a one-to-one communication for offloading immersive media processing which needs more computation power than conventional media. This leads to the discussion about split rendering.

This study mainly focuses on the media connection model with intermediate servers.

P2P connection has some benefit for one-to-one communication (i.e., no need for an intermediate server and less server-relayed delay). For that reason, P2P connection is also considered for some special cases.

### 6.2.8 IP Addressing

#### 6.2.8.1 Overview

IP addressing for UE has some options: assigning IPv4 address only, IPv6 address only, or both.

In the operator deployment, the number of available IPv4 addresses would be insufficient for its subscribers. Generally, operators use IPv4 private address (and ISP shared address defined in IETF RFC 6598[RFC6598]) with network address translation (NAT).

In clause 6.2.8, appropriate IP addressing is identified, discussing NAT-traversal in the WebRTC user plane and network verified ID retrieval.

#### 6.2.8.2 NAT

##### 6.2.8.2.1 Overview

NAT, including port translation as NAPT (Network Address and Port Translation), is a method of mapping an IP address space into another, which is mainly used to translate a private IP address into a global IP address, and vice versa, for communicating with external networks.

Generally, UE can be assigned with an IP address through a PDU session in operator networks. When an IPv4 address is allocated, as mentioned in clause 6.2.8.1, a private IP address or an ISP shared address is used. On the contrary, when an IPv6 address is allocated, a global unicast address is assigned.

NAT is essential for carrier-grade network deployment. Subscribers can be much more than usually available IPv4 global address space, and they are treated by using IPv4 private address and NAT. The same private address can be reused in each different domain behind NAT. Although NAT deployments have a wide variety, NAT is generally installed in a DN (data network) and often put in the middle between the UPF and other functional entities (see Figure 6.2.8.2-1).

On the other hand, IPv6 global unicast addresses basically do not require NAT, except for special security reasons or some transition method between IPv6 and IPv4 domains.



Figure 6.2.8.2-1: Possible NAT location

##### 6.2.8.2.2 NAT Variation

NAT is classified into some types by its address translation and packet filtering behavior.

The first version of STUN in IETF RFC 3489 [RFC3489] defines three types:

- Full Cone NAT,

- Restricted NAT (Restricted Cone NAT or Restricted Port Cone NAT), and

- Symmetric NAT.

Full cone NAT does not limit access to an internal UE from external network entities, which have not communicated with the internal UE. Any external entities can re-use the external IP address and port number mapped to a specific internal UE and can access to it (Figure 6.2.8.2.2-1). Full cone NAT is less restrictive than other NATs. Restricted NAT only permits external entities to access the internal UE if the NAT have received any packets from the internal UE directed to the external UE (Figure 6.2.8.2.2-2). Symmetric NAT uses a different pair of an external IP address and port, which are specific to each external entity and only the external entity can access to the internal UE through the IP address and port pair.



Figure 6.2.8.2.2-1: Full Cone NAT behaviour



Figure 6.2.8.2.2-2: Restricted or Symmetric NAT behaviour

##### 6.2.8.2.3 Existing NAT-traversal

###### 6.2.8.2.3.1 General

An effective NAT-traversal method is different depending on the NAT type described in clause 6.2.8.2.2.

In the original WebRTC design, STUN and TURN are listed, included as ICE, for major NAT-traversal methods. In addition, Hosted NAT Traversal (HNT, described in IETF RFC 7326 [RFC7326]) and its similar mechanism are frequently used in real implementations for conversational applications.

###### 6.2.8.2.3.2 STUN

STUN is the method for UE behind the NAT to discover its external IP address observed by external networks. This method supports P2P communications and only works for full-cone NAT.

This study excludes STUN because the main communication model is not P2P but with intermediate servers (as described in clause 6.6), and general NATs deployed in operator networks are not limited to full-cone type.

###### 6.2.8.2.3.3 TURN

TURN is the method for UE behind the NAT to communicate with external nodes via an intermediate server. TURN is a protocol for the session management and requires an intermediate server.

Generally, this method is regarded as the last resort for NAT-traversal for UDP-based conversational services. This method does not require the alignment with other control plane signalling, but is equipped as its own user plane connection management mechanism. This method needs additional message exchanges and has a protocol overhead.

The TURN server has its authentication mechanism for UEs and can be used for the purpose of traffic steering for an inter-operator communication scenario detailed in clause 6.9.

###### 6.2.8.2.2.4 HNT

HNT (Hosted NAT Traversal) is the mechanism that a session border controller (SBC) placed at the edge of networks intermediates the communication between UEs behind NAT.

The problem tackled by HNT is that a UE behind a NAT tries to set up a session with its private address and port number for media, which have no clue to the SBC for the real media which comes later.

Regarding the control plane signalling, the signalling part of the SBC modifies media-related information represented by the private IP address and port number set in the SDP offered by an originating node into a global IP address and a new port number. This modification enables a terminating node to target the accessible IP address and port pair provided by the SBC. In the signalling return path, the SBC also modifies the terminating node’s IP address and port number set in the SDP answered by the terminating node into new ones, and forwards it to the originating node. This is to solicit the originating node to send media to the SBC. Once the SBC receives the first media packet from the originating node targeting at the solicitation, the SBC recognizes the real NAT-ed IP address and port pair of the originating node. The SBC captures that information and uses it for relaying packets from the terminating node to the originating node. This is called “latching”.

This method is embedded in the control plane signalling and does not require extra message exchange. For that reason, it has no additional protocol overhead. It is a better feature than TURN in the same condition requiring an intermediate server.

Since this study focuses on the connection model with an intermediate server, the NAT issues can be argued differently. Let’s assume that all communication services are provided by the intermediate server as a conference. UEs can just join the open channel provided by the server and receives media from the server. UEs can also send their media to the intermediate server and the server mixes the media and distributes to other UEs. In this model, the first join packet from a UE to the NAT and the NAT to the server creates an address mapping at the NAT. The server simply sends packets to the source address of the join packet from the UE.

This mechanism does not need the dedicated protocol and there is no additional protocol overhead for NAT-traversal by sending media to the specific IP address and port pair exposed by WMCF. That points are analogous to HNT (Figure 6.2.8.2.3-1).



Figure 6.2.8.2.3-1: HNT like NAT-traversal

##### 6.2.8.2.4 Conclusion of NAT handling

NAT-traversal problems have been discussed and several solutions have been proposed as described above. However, if equipment for NAT-traversal is not required, certainly less server resources would be needed.

In short, it is preferable that only IPv6 global unicast address be assigned to UE and no dedicated NAT-traversal equipment be used. Intermediate servers are used mainly for media processing and for the media relay when there is no direct IP reachability (e.g., across inter-operator connection).

#### 6.2.8.3 IP Address and Trustable Subscriber Identifier

The operator uses subscription identifiers (e.g., GPSI (Generic Public Subscription Identifier) in 5GC) for managing its customer’s service subscription and charging. In WebRTC support, the operator needs to check customer’s service requests by checking against operator’s subscriber database organized with the subscription identifier. An OTT-specific ID and password may be insufficient even in the collaboration scenario with external service providers because they cannot be securely linked with subscriber information in the viewpoint of the operator. The issue is how the MNO deduces (or retrieves) the trustable subscriber identifier from customer’s requests, which are carried by IP packets.

Trustable subscriber identifiers in the MNO network are required for certain validity check, since a UE’s self-claimed GPSI and source IP address are untrusted.

The EDGEAPP architecture specifies the method how the EAS function block retrieves the GPSI from terminal’s source IP address. The AF regarded as an EAS can retrieve the GPSI bound to the UE by Eees\_UEIdentifier API in EDGEAPP. This mechanism and its flow contain authentications conducted at the related network functions (i.e., EES and NEF), which enable the EAS to acquire the valid GPSI in the operator network as a trustable subscriber identifier.

Validity of the terminal’s source IP address needs consideration. UE’s self-claimed IP address, especially presented in an application level, is not trustable. The source IP address presented in an IP header can be relatively trustable when the IP packet is transmitted through a connection with some handshake procedures.

The IP address linkage with a subscriber identifier also has an issue when NAT is deployed. In Release 18, the method with which the AF can identify the trustable subscriber identifier (e.g., GPSI) to invoke the 3GPP network service API for the UE (Application client) remains to be investigated in eEDGEAPP. In VoLTE, this linkage with NAT can be achieved with the help of additional operator-specific information (e.g., PDN session related value). In the AF for WebRTC, it depends on which additional information element can be acquired by the AF. There is no clear answer for the ID linkage between the NAT-ed IP address and the subscriber identifier.

Contrarily, the UE IP address without translated by NAT can be linked with GPSI by Eees\_UEIdentifier API (though detailed specification is needed).

In terms of ID linkage, using IPv6 global unicast address for UE is reasonable.

Using IPv4 private address will be studied further when NAT-ed ID linkage issue is solved.

#### 6.2.8.4 Conclusion of IP Addressing

In terms of the required server resources for NAT-traversal and unclear retrieval of the trustable subscriber identifier, using IPv6 global unicast address for UE is reasonable. NAT deployments have a wide variety of behaviors and cannot be treated straightforward (refer to clause 6.2.8.2). Using media relay servers that act as either TURN or HNT covers most cases with NAT-traversal. However, there are still issues using IPv4 private address with NAT, such as ID linkage (refer to clause 6.2.8.3). For the sake of simplicity and to concentrate on identifying signalling requirements, this study considers IPv6-only use.

Also, IPv6-only use is acceptable for future services because IPv6 address allocation to UEs is now widely available among operators. Also, IPv6-only deployment (or not using limited IPv4 address resource) leads to efficient system development and equipment utilization.

### 6.2.9 QoS Enabled End-to-End Path

This study covers two collaboration scenarios as is described in the previous clause. In the collaboration scenario where the WebRTC functions in an MNO assist an external service provider (OTT or another MNO), setting up a QoS-enabled media path across different networks needs to be studied .

The media path from a UE to the external service provider is roughly divided into four sections:

Section 1) Between a UE and the UPF (Operator’s CN section)

Section 2) Between the UPF and the operator’s network edge (Operator’s DN section)

Section 3) Between the operator’s network edge and the external service provider network edge

Section 4) A network in the external service provider

Section 4) is a matter of a service provider and out of scope of this study.

Regarding Section 1), this section includes the operator’s core network. In this section, QoS is controlled by the PCF. In the collaboration scenario with an external service provider, the main signalling server is placed in the service provider’s domain. While UE exchanges control plane signalling messages with the signalling server placed in the service provider’s domain, UE sends a QoS-related request separately to the WSF placed in the operator network. The WSF receives and interprets the UE’s request and requests the PCF to prioritize the UE’s specific session.

Regarding Section 2), operator’s DN may have sufficient bandwidth and other QoS mechanism may be adopted.

Regarding Section 3), this section’s QoS control needs a bandwidth guaranteed path (i.e., a dedicated line) and forwarding control. Such a forwarding path selection cannot be realized by simple IP routing. A couple of options are possible. The first option is that only media packets to be prioritized are transmitted to WMCF placed in the operator’s network (which acts as TURN server) and the WMCF relays the media to the main media server in the OTT network via guaranteed path (Path 1 in Figure 6.2.9-1). The second option is that a non-prioritized forwarding path can be selected using a different destination IP address other than WMCF’s transport address within the same PDU session (Path 2 in Figure 6.2.9-1). The third option is that such a non-prioritized path can be selected through a different PDU session which goes through a different gateway (Path 3 in Figure 6.2.9-1). Such a PDU session can be selected by Data Network Name (DNN).



Figure 6.2.9-1: Forwarding Path Selection

Editor’s Note: Analyze gaps and identify required enhancements of terminal device and network architectures including additional functional entities (e.g., WebRTC Signalling Server, ICE-STUN Server, IMS Interworking Gateway, NNI Gateway).  
- Stage2 work requirements  
- Necessary functional blocks  
- Architectural comparison (details are in annex)

### 6.2.10 Possible 5G Real-time Communication Architecture for collaboration scenario 4

This study identifies the possible architecture for collaboration scenario 4 of 5G Real-time Communication specified in 3GPP TS 26.506 [TS26.506]. Figure 6.2.10-1 shows the derivative 5G-RTC architecture for collaboration scenario 4.



Figure 6.2.10-1: Possible derivative 5G-RTC architecture for collaboration scenario 4

NOTE 1: Other Network includes Trusted 5G-RTC ASs in a different MNO, 5G-RTC ASs in a service provider and functions of an IMS network.

NOTE 2: Trusted 5G-RTC AF and Trusted 5G-RTC AS are controlled by a single operator and located in the same operator network. Inter-working Function and Transport Gateway Function act as a border controller function at the boundary of the network.

The following interfaces is expected to be introduced for collaboration scenario 4.

- **RTC-9s**: This interface is for C-Plane signalling between Inter-working Function and other network(s).

- **RTC-9m**: This interface is for U-Plane media transport between Transport Gateway Function and other network(s).

- **RTC-2a**: This interface exposes 5G-RTC AS functionalities to 5G RTC Application Provider. (e.g., subscription of media resource in 5G RTC Application Provider network to enable media session setup by WebRTC Signalling Function.)

The functions described in this study correspond to the functions in the architecture for collaboration scenario#4 of 5G Real-time Communication Architecture as follows.

- **WSF (WebRTC Signalling Function)**: WebRTC Signalling Function

- **WMCF (WebRTC Media Centre Function)**: Media Function

- **CSF (Conference Supporting Function)**: Application Supporting Web Function

- **WNSGF (WebRTC NNI Signalling Gateway Function)**: Inter-working Function

- **WNMGF (WebRTC NNI Media Gateway Function)**: Transport Gateway Function

NOTE 3: As described in 3GPP TS 26.506 [TS26.506], the integration/collocation of 5G-RTC AF and WebRTC signalling server is possible. Co-located WebRTC signalling server is able to act as a 5G-RTC AF which is accessible to 5GC, and replace some of this 5G-RTC AF’s interfaces and APIs with WebRTC signalling. For example, interfaces and APIs between this 5G-RTC AF and UE will be replaced to avoid concurrent/redundant requests from UE.

The interfaces described in this study correspond to the interfaces in the architecture for collaboration scenario#4 of 5G Real-time Communication Architecture as follows.

- **Rs-u**: RTC-4s

- **Rs-n**: RTC-Xs

- **Rm-u**: RTC-4u

- **Rm-n**: RTC-Xm

For the study of C-Plane signalling aspects, this study focuses on RTC-4 based solutions as shown in Figure 6.2.10-2 to support the collaboration scenario 4 and the case for the application which is not able to use MSH (e.g., Web App).

- Trusted 5G-RTC AF functionalities are integrated in WebRTC Signalling Function, since MSH is not used. Then, MSH related interfaces are omitted in Figure 6.2.10-2.

- Functions of Trusted 5G-RTC AF are integrated within WebRTC Signalling Function, then RTC-3 is out of the scope.

- The representation of RTC-4s and RTC-4m are simplified. Web App and Native WebRTC App are expected to use these interfaces as follows.

- Web App utilizes the web browser’s JS API (including WebRTC API) to send/receive signalling message on RTC-4s and media on RTC-4m.

- Native WebRTC App utilizes the SDK provided by the OS of the UE to send/receive signalling message on RTC-4s and media on RTC-4m.



Figure 6.2.10-2: The focused interface of eiRTCW C-Plane signalling protocol

NOTE 4: RTC-4m is connected to ICE Function when TURN server needs to be used. Otherwise, RTC-4m is connected to Media Function.

NOTE 5: The interfaces and the functionalities related to MSH, NS-AF, Configuration Function and Provisioning Function are not in the focus.

### 6.2.x Solution evaluation

Editor's note: This clause provides an evaluation of the solution.

## 6.3 Solution #2: Requirements for C-Plane Signalling

### 6.3.1 Solution description

This solution addresses key issue #2.

This clause identifies requirements for control plane signalling for each reference point. In this study, reference points for control plane are Rs-u, Rs-i, Rs-n, Rs-a, as described in clause 6.

### 6.3.2 Rs-u

#### 6.3.2.1 Base Protocol

HTTP[RFC7230-7235][RFC7540]/HTTPS and WebSocket[RFC6455] are available options for signalling between UE and WSF so that connection setup procedure could be invoked by JavaScript API as described in IETF RFC 8825 [RFC8825] clause 3. Nevertheless, HTTP/HTTPS is less appropriate for two reasons described in RFC 6455 clause 1.1:

- Server load caused by http transactions (based on request-response)

- A connection has two sessions each for sending and receiving signalling packets

In addition, when a notification from the network to the UE is required, for such as an incoming call, an HTTP(S) connection is originated from the network side, but this case has some problem. Generally, NAT box is placed between UE and network entities, therefore NAT-traversal problem should be resolved. Besides, in terms of security configuration, UEs often deny incoming TCP[RFC793] connections.

For that reasons mentioned above, only WebSocket is utilized for the base protocol of signalling in this study. WebSocket can solve the three problems, server load, number of sessions and the NAT-traversal.

#### 6.3.2.2 Upper Layer Protocol over WebSocket

In IETF RFC 8825 [RFC8825], upper layer protocols over the base protocol of control plane are not specified and are thought to be application specific. In the RFC, SIP[RFC3261] and XMPP[RFC6120] are listed as candidate protocols for control plane.

##### 6.3.2.2.1 SIP

Utilizing SIP for control plane signalling for WebRTC is already described in 3GPP TS 24.371 [TS24.371] clause 5. One of the main advantages of using SIP is the ease of interwork between WebRTC-aware network and IMS network. On the other hand, disadvantages of using SIP are as follows:

- UE and network must be able to understand both WebRTC and SIP. SIP is not widely used outside of telephony. If SIP must be used in conjunction with WebRTC, the advantage of WebRTC, friendliness to web-based development environments and developers, is to be spoiled.

- SIP has a strictly managed communication model as SIP dialogue. In principle, the originated signalling is transparently relayed through the network and the terminals manage the dialogue with each other. These characteristics are not compatible with the UE-network relation model, which is the scope of this study.

- UE and network must be able to understand both WebRTC and SIP. SIP is not widely used outside of telephony. If SIP must be used in conjunction with WebRTC, the advantage of WebRTC, friendliness to web-based development environments and developers, is to be spoiled.

- SIP specifies methods divided by each signalling characteristic (i.e., INVITE, ACK, BYE, CANCEL, PRACK, UPDATE, SUBSCRIBE, NOTIFY, REFER, PUBLISH, INFO). Adding control for a new characteristic may need to start from the method definition.

- Less affinity with cloud environment which mainly uses HTTP. For example, raw values of the IP addresses related to the SIP dialog (consisting a communication path of SIP trapezoid) are in the protocol header or message body, therefore changing communication elements is difficult once the call session is established.

For those reasons above, SIP is not appropriate except for the applications where the interwork to IMS is expected.

##### 6.3.2.2.2 XMPP

There is no specification using XMPP for the upper layer protocol of the control plane signalling in 3GPP and no major commercial implementations of WebRTC either. The reason seems that XMPP can be used on its own and does not need to be combined with other protocols. WebSocket encapsulation of XMPP has little benefit except the case that an application using XMPP is implemented using JavaScript.

Therefore, this study will not utilize SIP and XMPP. More optimal (or WebRTC native) signalling protocols for the upper layer of control plane is to be identified in this study.

##### 6.3.2.2.3 Other Existing Implementations

Among the existing implementations of WebRTC communication, JSON[RFC8259] format is mainly used for the upper layer of control plane. This is because JSON format is easy to handle in JavaScript. In this study, the potential of JSON for the upper layer protocol of control plane signalling is investigated.

In 3GPP specifications, RESTful APIs (such as service-based interface and Northbound APIs) are often defined using OpenAPI 3 [OpenAPI] and the message-body of the APIs are based on JSON. However, OpenAPI is mainly suitable for RESTful APIs and not suitable for message-driven APIs such as control plane signalling over WebSocket. For this reason, AsyncAPI [AsyncAPI] as well as OpenAPI are used for identifying API schemas in this study. Those two APIs are managed by Linux Foundation.

#### 6.3.2.3 Protocol Stack

As described above, JSON based protocol over WebSocket is a solution for the protocol of Rs-u. (see Figure 6.3.2.3-1)

WebSocket can be deployed over several versions of HTTP.

- WebSocket with HTTP/1.1 is specified in IETF RFC 6455 [RFC6455] and used in this study. HTTP/1.1 is not, however, shown in the protocol stack because HTTP/1.1 does not remain after upgrading into WebSocket.

- WebSocket with HTTP/2 is specified in IETF RFC 8441 [RFC8441] and used in this study. HTTP/2 is shown in the protocol stack because HTTP/2 framing remains after a stream in HTTP/2 connection is upgraded into WebSocket.

- WebSocket with HTTP/3[RFC9114] is specified in IETF RFC 9220 [RFC9220] but not used in the current version of this study. The transport protocol used over HTTP/3 needs to be selected in alignment with IETF/W3C discussions.

The sub layers of each protocol are according to the existing specifications.

- TLS under HTTP/1.1 and HTTP/2 is specified in IETF RFC 8446 [RFC8466].

- TCP under TLS is specified in IETF RFC 793[RFC793].

- IPv4 and v6 under TCP are specified in IETF RFC 791[RFC791](IPv4) and IETF RFC 8200 [RFC8200](IPv6).



**Figure 6.3.2.3-1: Protocol Stack of Rs-u**

Editor’s Note: Identify signalling protocol details (e.g., based on JSON) for the common WebRTC-based immersive RTC session management. Identify information elements in the C/U-Plane signal (including NNI) to enhance connectivity of media sessions with carrier assistance for WebRTC-based applications (including OTT applications). Identify the minimal functional capabilities needed to support the enhancements identified in above.  
- Stage3 requirements  
- C-plane signalling requirements  
- Signalling methods comparison (details are in annex)

### 6.3.x Solution evaluation

Editor's note: This clause provides an evaluation of the solution.

## 6.4 Solution #3: Requirements for U-Plane Signalling

### 6.4.1 Solution description

This solution addresses key issue #3.

### 6.4.x Solution evaluation

Editor's note: This clause provides an evaluation of the solution.

## 6.5 Solution #4: Interworking with IMS Network

### 6.5.1 Solution description

This solution addresses key issue #4.

### 6.5.x Solution evaluation

Editor's note: This clause provides an evaluation of the solution.

## 6.6 Solution #5: Tethered Cases

### 6.6.1 Solution description

This solution addresses key issue #5.

Editor’s Note: SmarTAR-related clause;  
Identify enhancements for E2E QoS realizations over 5G systems for communications between MNOs and WebRTC clients operating over non-5G links (e.g., Wi-Fi) using WebRTC-based transport. This also includes communication between WebRTC clients operating on tethering/tethered devices.

### 6.6.x Solution evaluation

Editor's note: This clause provides an evaluation of the solution.

## 6.7 Solution #6: Security Considerations

### 6.7.1 Solution description

This solution addresses key issue #6.

Editor’s Note: Considerations that the third-party access to the operator network need to be controlled with SLAs and with secure access to protect the underlying network resources.  
- Rate limiting  
- Abuse protection  
- Security measures

### 6.7.x Solution evaluation

Editor's note: This clause provides an evaluation of the solution.

## 6.8 Solution #7: Related Groups Considerations

### 6.8.1 Solution description

This solution addresses key issue #7.

Editor’s Note: Identify collaboration formation with other WGs in 3GPP and SDOs including IETF and W3C.

### 6.8.x Solution evaluation

Editor's note: This clause provides an evaluation of the solution.

# 7 Overall evaluation

Editor's note: This clause provides overall evaluation of the solutions in clause 6.

# 8 Conclusions and Recommendations

Annex A: Use Cases

Editor’s Note: Use cases and communication methods (P2P, SFU, MCU)

Annex B: Examples

# B.1 Architectural WebRTC Entity Examples

Editor’s Note: Architectural example of integration of WebRTC with 5G network

# B.2 Protocol Stack Examples

Editor’s Note: Definite example of C-plane protocol stack. Reference of U-plane (other TS/TRs) and supplemental explanation.

# B.3 WebRTC Signalling Protocol Examples

Editor’s Note: Expected signalling regulation examples (Async API)

# B.4 WebRTC Signalling Flow Examples

## B.4.1 General

This clause shows example flows. This flow can be applied to the collaboration scenario 3A described in FS\_eiRTCW where a service provider (i.e., OTT) provides WebRTC services and an MNO assists the services. The WebRTC Signalling Server provided by the external service provider acts as the main signalling server, and the WSF deployed in MNO is complementary, only used for ID authentication and QoS control.

Note: “Web Service Entry Point” (in the figures) is a general entry point for the service, which acts as a portal website or an API endpoint of the service.

## B.4.2 Part.1 UE connects to OTT’s server



The UE connects to the OTT’s server (i.e., Service Provider’s server). This procedure and its protocols are proprietary to the OTT.

## B.4.3 Part.2 UE discovers CSF with EDGEAPP API



The UE discovers the CSF in the MNO with EDGEAPP APIs.

## B.4.4 Part.3 UE discovers OAuth Endpoints in CSF



The UE discovers OAuth Endpoints in the CSF with RFC8414 procedures.

## B.4.5 Part.4 OAuth



The UE starts OAuth 2.0 authorization procedures and gets an Access Token from the CSF (acting as an Authorization Endpoint and Token Endpoint of OAuth 2.0). Now the UE can use MNO’s QoS assistance with the access token. This authorization process does not have to be done each time until the token expires. Additionally, the process can be independently done in advance.

## B.4.6 Part.5 UE discovers WSF and WMCF with EDGEAPP API



The UE discovers the WSF and the WMCF with EDGEAPP API.

The CSF, WSF and WMCF are registered with each subtype of an EAS (Edge Application Server).

## B.4.7 Part.6 UE establishes QoS media flows



The UE requests TURN allocation and channel binding from the WMCF with OAuth access token (from 6-1 to 6-6), and builds a local SDP with the relaying address and port (6-7). Web browser’s WebRTC API will build an SDP, but “SDP munging” may be needed later because some browser does not include b= lines in SDP that is essential to controlling QoS.

With the local SDP, the UE requests MNO’s WSF (6-8) to control QoS, and the WSF requests the PCF to control QoS via N5 interface (6-9 and 6-10). The interworking procedure on the SDP at N5 interface will be the same as IMS P-CSCF’s procedure described in 3GPP TS 29.513.

The UE also sends the local SDP to OTT’s signalling server and receives the remote SDP (from 6-12 to 6-14). The UE tries hole punching and updates the local and remote SDPs (6-15 and 6-16).

With the updated local and remote SDPs, the UE requests MNO’s WSF to update QoS control (6-17). The WSF requests PCF to update QoS with N5 interface (6-18 and 6-19) and responds to the UE with success (6-20).

Finally, the UE can communicate with QoS (6-21).

Editor’s Note: Sequence and message examples using clause  B.3

# B.5 Conference Management Protocol Examples

Editor’s Note: Examples of conference session management (OpenAPI)

# B.6 Conference Management Flow Examples

Editor’s Note: Sequence and message examples using Annex E

Annex C: Open Issues

Editor’s note: This annex describes possible future work.

# C.1 Materials for Further Study

a) The C-plane signalling protocol should support basic WebRTC service operations such as client registration, authentication and authorization; call control; and data channel management that are relevant to the new architecture.

c) Security considerations for interoperable WebRTC services such as authentication, authorization, and key management

d) Deployment options of traditional WebRTC functions in 5G network, and mapping of those functions to 5G media architecture

NOTE: Mapping of WebRTC functions to 5GMS functions to be confirmed in 5GAREA study

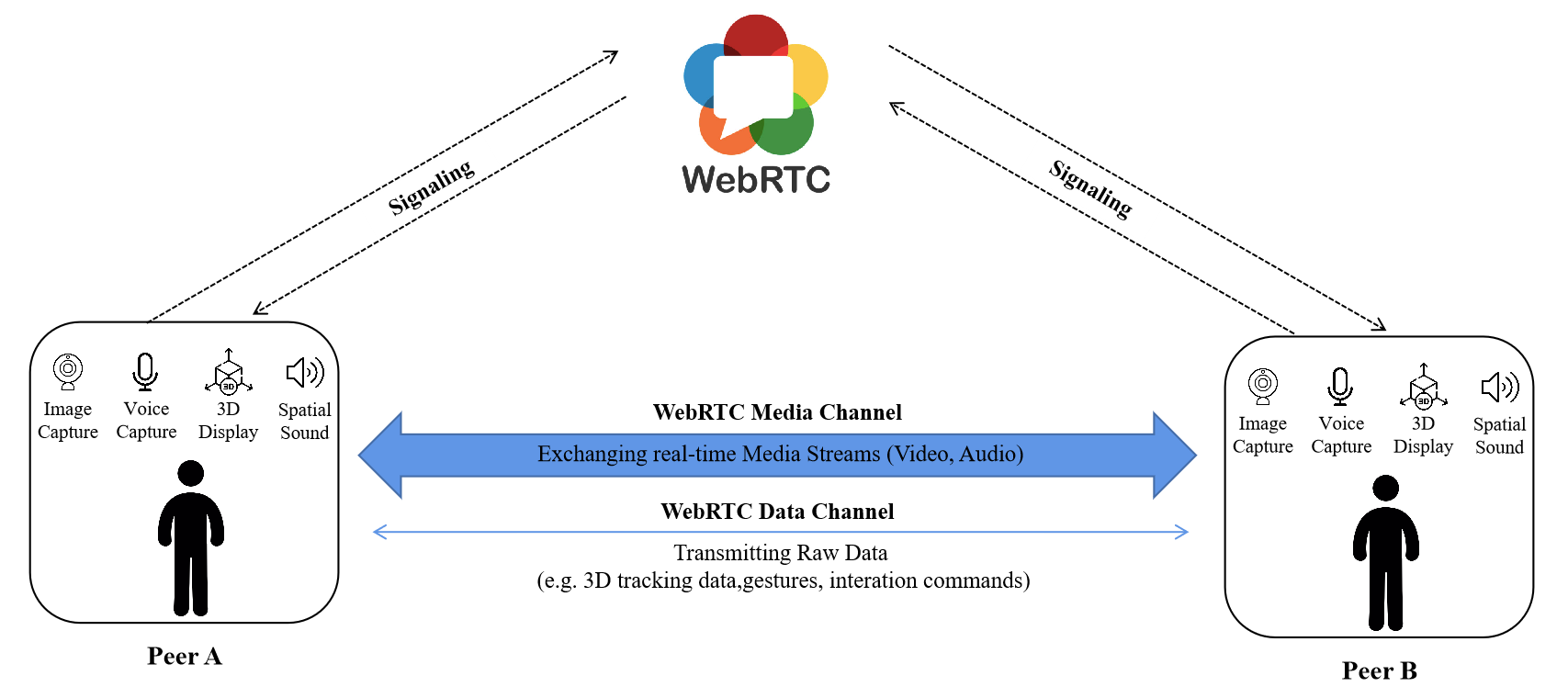
e) Feasibility to use existing 5GMS architecture enablers for betterment of WebRTC services.

# C.2 Candidate Use Case

Editor’s note: Media handling remains unidentified, but the use case of one-to-one communication is to be included in clause 3 by some form. The following description is suggested by CMCC.

3D Telepresence is a real-time bidirectional video communication system that enables two people to communicate at distance as if they were physically together. Using the benefits of WebRTC, we can combine 3D Telepresence with a regular video call service. It offers users a more natural and intuitive way for remote connection.

3D Video Call establishes a one-on-one communication between users to guarantee low latency and high security holographic communication experience. The 3D video stream and the audio stream are being compressed and real-time transmitted over WebRTC protocol. Both ends support incoming and outgoing calls.



For example, Alice sits in a conference room in Midtown Manhattan, there are cameras, microphones, and sensors to capture her image, voice and movements from multiple vantage points. The captured imagery and voice get transmitted in real-time to the remote peer Bob, who is working in his studio in Boston. Both of them can see each other’s imagery in three dimensions and also hear the voice. They can talk naturally with the full range of communication cues, such as eye contact, hand gestures, and body languages. The whole experience makes them feel like they’re actually sitting in the same room.

# C.3 Solution Design Concepts

It is proposed to construct solutions in this study along the concepts described as follows:

1. Complying with the WebRTC standards and following the practical implementations in real deployments.
2. Using IETF and 3GPP protocols where the WebRTC standards do not specify.
3. Using general-purpose 5G functions (e.g., EDGEAPP and Network Slice) and the associated protocols if available rather than newly creating RTC specific ones.
4. Using the same protocols as much as possible for the collaboration scenarios: MNO’s Operator-Assistance for OTT’s WebRTC services and MNO’s WebRTC services as an operator service.

# C.4 Major features concerned with WebRTC

## C.4.1 Overview

These are the major features in each step of the communications with Operator-Assistance for OTT’s WebRTC services:

- Discovery - procedures to discover the relevant functional entities

- NAT traversal - ICE functions for communication through NAT/NAPT

- Identifiers - Linkage between OTT ID and MNO ID

- QoS enablement - information notified from the UE to the MNO for QoS control

## C.4.2 Feature: Discovery of Functional Entities

A solution with the existing 5G functions is using EDGEAPP (Architecture for enabling Edge Applications) specified in SA6.

WebRTC applications on a UE can discover WebRTC functional entities (e.g., CSF, WSF, and WMCF) in an MNO network with the EDGEAPP enabler. CSF, WSF and WMCF are registered on EES (Edge Enabler Server) beforehand as EASs (Edge Application Server) with different service specific features (easFeats).

Editor’s note: Server discovery procedure is essential. Various methods including static and/or dynamic way should be studied further. As a possible solution for dynamic server discovery, the example call flow using EDGEAPP API is provided.

## C.4.3 Feature: NAT traversal and ICE Functions

For NAT traversal, ICE is integrated in the WebRTC framework and JavaScript API so that ICE must be considered. However, ICE requires the network to support additional features. In this study, it is proposed to consider not using ICE functions if possible.

Editor’s note: A possible solution is described in clause 5.7.2 of FS\_eiRTCW permanent document.

## C.4.4 Feature: Linkage of OTT ID and MNO ID

It is essential to make a linkage between OTT’s ID and MNO’s ID (e.g., GPSI or IMSI) for providing operator-assistance for OTT. The linkage must be verified by the MNO for charging and service conditions/restrictions management of MNO.

The CSF or WSF can obtain verified MNO’s ID of UEs from the EES with EDGEAPP APIs as an EAS. (API specifications of the function is under discussion in SA6 and CT3.)

Editor’s note: A solution for retrieving verified MNO’s ID is described in clause 5.7.3 of FS\_eiRTCW permanent document.

A solution for making the linkage between OTT’s ID and MNO’s ID (obtained via EDGEAPP APIs) is OAuth that is widely used in OTT applications.

## C.4.5 Feature: QoS enablement - Information notified from the UE to the MNO for QoS control

In the WebRTC framework, the UE generates and accepts SDPs to exchange IP addresses/port numbers and QoS related information (e.g., bandwidth) for transmitting media. The SDPs are sent to or received from the signalling server of OTTs with their proprietary protocols.

A potential solution can implement an extra mechanism for QoS control. The UE notifies the SDPs to the MNO’s WSF in addition to the OTT’s signalling server. The MNO’s WSF can control the PCF based on the information in the SDP. In this mechanism, the role of MNO’s WSF is similar to the role of P-CSCF in IMS.

NOTE 1: Application’s QoS requirements may not be met if the application runs on the default PDU session. The use of Network Slice may be needed and for further study.

NOTE 2: Prior to the SDP generation, suggestions about available bandwidth from the MNO to the UE is an open issue.

## C.5 Multiple Video Sources and zone allocations

Figure C.5-1 illustrates a common use case when a UE has multiple video sources (e.g. 2D/3D-capable). Each camera may have a fixed Field-of-View (FoV) or varied FoV. Each camera may support the same set of video capabilities such as codec support.

A camera ID or zone ID is assigned for each camera. By assigning a zone number, the UE has the flexibility to signal each source by its source ID (e.g. SSRC in case of RTP) or zone/camera ID.

The zone ID may be assigned with a priority based on the areas it covers and may consist of one or more cameras. For example, the area covered by cameras in zone-1 may be more important than the ones located in zone-2 and zone-3 since it covers the front FoV of the UE. This is important information since it enables UE to signal the essential zone areas or high-priority zones. In some cases, all of the zones have to be treated equally, then all the zone will have the same priority assignment.



**Figure C.5-1 Multiple Video Sources With different zone allocations in UE**

NOTE: the number of cameras and the size of UE, i.e., the entire coverage of all cameras, can vary.

In this use case, there are a couple of possible scenarios.

For example, UE-A has multiple media sources under its control. UE-A is communicating with UE-B:

1. Each media source belonging to UE-A is able to produce an individual media stream. To set up the media stream with UE-B, UE-A identifies the source of the media (e.g., by camera ID) and exchanges information about the media.
2. In each zone, media produced by the sources in the zone may be processed and combined into a new media. UE-A identifies (the source of) the combined media (e.g., by zone ID) and exchanges information about the media.
3. Media produced by the sources in all zones may be combined into a new media stream. UE-A identifies (the source of) the combined media and exchanges information about the media.
4. The zone IDs or camera IDs may be associated to particular pose information of UE-B when the UE-A is creating or sending immersive content. Streams from individual camera or cameras in certain zones can be paused/resumed depending on the viewing orientation of the receiver UE-B (i.e., for viewport-dependent media). In this case the streams do not have to be combined.

Annex <X> (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **TSG #** | **TSG Doc.** | **CR** | **Rev** | **Subject/Comment** | **Old** | **New** |
| 2022-04 | SA4#118 |  |  |  | Initial version |  | 1.0.0 |
| 2022-05 | SA4#119 | S4-220516 |  |  | This version contains the changes of agreed SA4#118 |  | 2.0.0 |
| 2022-06-01 | SA4#119 | S4-220777 |  |  | This version contains the changes of agreed SA4#119  (S4-2207585, S4-220778)  Concepts included or merged  (S4-220780, S4-220781) |  | 2.0.1 |
| 2022-06-01 | SA4#119 post |  |  |  | Editorial change in reference clause |  | 2.0.2 |
| 2022-08-03 | SA4#120 | S4-220925 |  |  | This version contains the changes of agreed during SA4#119  (S4a-220006, S4a-220007, S4a-220019)  Updated references to RFC 9114 and 9220 |  | 2.1.0 |
| 2022-08-25 | SA4#120 | S4-221211 |  |  | This version contains the changes of agreed during SA4#120  (S4-221210) |  | 3.0.0 |
| 2023-02-24 | SA4#122 | S4-230340 |  |  | This version contains the changes of agreed during SA4#122  (S4-230182, S4-230334) |  | 4.0.0 |
| 2023-05-26 | SA4#124 | S4-231009 |  |  | This version contains the changes of agreed during SA4#124  (S4-230850, S4-231000) |  | 5.0.0 |