**3GPP TSG-WG SA4 #127 meeting *-240321***

**Sophia-Antipolis, France, Jan. 29 – Feb. 2, 2024 Revision of S4-240148**

|  |
| --- |
| *CR-Form-v12.2* |
| **Pseudo CHANGE REQUEST** |
|  |
|  | **26.113** | **CR** |  | **rev** | **1** | **Current version:** | **1.0.0** |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **X** | Radio Access Network |  | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  | [iRTCW] pCR on 26113: Scope & Annex |
|  |  |
| ***Source to WG:*** | Samsung Electronics, Co., LTD |
| ***Source to TSG:*** | S4 |
|  |  |
| ***Work item code:*** | iRTCW |  | ***Date:*** | 23 Jan. 2024 |
|  |  |  |  |  |
| ***Category:*** | B |  | ***Release:*** | Rel-18 |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19)* |
|  |  |
| ***Reason for change:*** |  |
|  |  |
| ***Summary of change:*** |  |
|  |  |
| ***Consequences if not approved:*** |  |
|  |  |
| ***Clauses affected:*** |  |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  |  |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  |  |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  |  |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** |  |

|  |
| --- |
| **First Change** |

# 1 Scope

The present document specifies the set of stage-3 procedures, APIs, and protocols for the reference points defined in Real-Time Media Communication (RTC) architecture. While TS 26.510 defines the common set of APIs and interactions, this document refers to TS 26.510 for the general aspects and primarily deals with RTC-specific aspects to support WebRTC-based real-time media transport over 5G.

|  |
| --- |
| **Second Change** |

Annex A (informative):
RTC client in terminal

# A.1 Overview of high-level RTC data flow

The Real-Time Communication (RTC) system is designed based on the RTC architecture specified in [2] to handle an immersive media such as AR or XR. Figure A.1-1 illustrates the high-level view of the RTC system that uses RTC AF and AS for realizing the services. RTC AF and AS provide the Control Plane (C-Plane) functionalities for setting up and controlling media and data sessions (U-Plane). The functionalities depend on supported collaboration scenarios, which are described in [2].



Figure A.1-1: High-level data flow showing two RTC endpoints in terminals.

NOTE 1: RTC AS may exist in the media/data path depending on the collaboration scenarios.

NOTE 2: RTC AF and AS are provided by MNO or 3rd party, depending on deployed collaboration scenario.

NOTE 3: Operator B is depicted for collaboration scenario 4. In other collaboration scenarios, "Operator B" is replaced with "Operator A", and the boxes representing the same functionalities are provided by an operator.

# A.2 Reference RTC endpoint model

The RTC endpoint supports a subset of WebRTC, which enables real-time communication via application programming interfaces (APIs), supporting audio, video, and generic data to be sent between peers. Functionalities of WebRTC are available as JavaScript APIs for browsers, and libraries for applications [12]. Information on use cases and requirements of WebRTC can be found in [22].

The functional components of a terminal including an RTC endpoint using 3GPP access are shown in figure A.2-1. Based on XR Baseline terminal architecture specified in TS 26.119 [23], Media Session Handler and Content delivery protocols are realized as a RTC MSH and WebRTC Framework, as specified in TS 26.506, respectively. Application may be a WebRTC application where C-plane is supported by RTC architecture or Web application (e.g., browser) where WebRTC APIs are involved for peer connection and immersive media delivery. Details of the associated APIs (RTC-6 and RTC-7) are specified in TS 26.510 [3]. The rest of functional blocks and interfaces are addressed in TS 26.119.



Figure A.2-1: Functional components of a terminal



Figure A.2-2: Functional components to handle immersive media

NOTE 1: Device information is assumed to be stored in the UE and loaded to the RTC endpoint during session setup.

NOTE 2: The RTC endpoint may exchange media and data with external devices tethered over wired/wireless links such as USB-C, 3GPP PC5 [24], or non-3GPP radio access technologies (e.g., Wi-Fi or Bluetooth).

NOTE 3: Text can be entered via user interface, typically available on display.

When a user launches a WebRTC application, a RTC MSH communicates with RTC AF to retrieve configuration information for session establishment. Note that this is exchanged via RTC-5 or alternatively, application-specific signalling function (e.g., collaboration scenario 1) as addressed in Annex A of TS 26.506 [2]. The configured information is then available to Application and Media Access Function via RTC-6 interface and the Application is ready to deliver an immersive media to the remote endpoint.

The following components are exchanged over WebRTC session.

- Video component: An RTC endpoint in terminal can be connected to one or more colour cameras, and/or to one or more depth cameras. The outputs of cameras may be pre-processed (e.g., converting data rates or representation formats) and the pre-processed media may be transmitted to the receiver of remote RTC endpoint. Then the remote client may post-process before they are input to displays (e.g., scene composition).

- Audio component: Similarly to video component, one or more microphones can be connected to an RTC endpoint. The captured audio bitstreams may perform pre-processing and/or post-processing to enhance the immersiveness (e.g., acoustically matching the perceived directions or locations of audio with those of video scenes).

- Sensor component: An RTC endpoint can utilize the information from various sensors for understanding environments, processing captured or received media, or other goals. The information may be locally utilized or transmitted with processed media.

- Signalling information: An RTC endpoint communicates to WebRTC signalling server to establish peer-to-peer connection. This signalling information is delivered through RTC-4s interface (as specified in clause 4.3.3 of TS 26.506) using WebSocket. Detailed protocol of WebRTC signalling is addressed in clause 13.2.