**3GPP TSG-SA WG4 Meeting #133-eS4-251345\_r01**

**Online, 18-25 July 2025**

**Source: Samsung Electronics Co., Ltd.**

**Title: [AvCall-MED] Updates to General Avatar Call Flow and Timing Information**

**Agenda item: 10.7**

**Document for: Agreement**

**1. Introduction**

This contribution provides updates to the general avatar call flow in the latest base CR (S4-251145).

**2. Discussion**

During SA4 #132 the general avatar call flow as updated to include aspects related to pose information, pose correction and pose related feedback for the sender-centric rendering mode. During the meeting it was mentioned that similar aspects should be included for the receiver-centric and network-centric modes. This contribution brings these aforementioned updates.

**3. Proposal**

It is proposed to agree the following changes to the latest version of the CR for 3GPP TR 26.264.

\* \* \* First Change \* \* \* \*

A.2 Avatar Communication Call Flows

### A.2.1 General Avatar Call Flow



Figure A.2.1-1: IMS Avatar Delivery and Animation Flow

**Z. Base Avatar Generation Before Call Setup**

The base avatar is generated before step A. Call Setup and Capability Negotiation. It may be uploaded to the BAR by using the Avatar management interface defined in annex B.

**A. Call Setup and Capability Negotiation**

A.1 An audio/video session is established between UE1 and UE2 and parameters of the session are negotiated.

The list of Avatar ID(s) and/or Avatar Representations is downloaded to the UE by the following options:

* Pre-configured in the UE: The Avatar ID List and/or Avatar Representations is provisioned or downloaded to the UE before a data channel for avatar call is setup.,
* Through bootstrap data channel: The Avatar ID List is fetched by the DC AS from the BAR when the associated Avatar communication application is downloaded and transferred from the DC AS to the DCSF and downloaded to UE through bootstrap data channel (see details in Annex AC 11.3.1 in TS 23.228[2].
* Through application data channel: The Avatar ID List is fetched by the DC AS from the BAR and downloaded to the UE through application data channel).

Note: further details of avatar selection and negotiation are defined in clause A.2.3.

**B. Scene Description Retrieval**

The MF and the participating UEs retrieve the scene description. The scene description may be shared by the MF with the UEs, in case of a shared experience, or the UEs may have their own scene descriptions.

**C. Scene Description Update**

A scene update trigger occurs, e.g., if an object is added to or removed from a scene or if spatial information is updated. The update trigger may originate from the MF itself or the UEs. The UEs may update their scene descriptions independently or the MF may generate an updated scene description and share it with the UEs.

NOTE1: The step B and C are not needed for 2D avatar.

**D.1. Avatar Acquisition**

D.1.1: The MF loads the base avatar for UE1 from BAR.

**D.2. Avatar Delivery**

Alternative #1: Sender-centric

D.2a.1: The MF delivers the base avatar of UE 2 to UE1 through data channel.

Alternative #2: Receiver-centric

D.2b.1: The MF delivers the base avatar of UE1 to UE2 through data channel.

**D.3. Animation Data Generation**

Based on the capability negotiation result in step A, the UE or network may generate animation data.

Alternative #1: UE centric animation data generation

D.3a.1: The UE1 generates the animation data based on the source data (e.g., audio, video, text) or using an XR runtime. The animation data may be transformed from the source data (e.g., from audio to text), or the same as the source data.

D.3a.2: UE1 delivers the animation data to the entity actuating avatar animation through data channel. The animating entity may be the MF or UE2.

Alternative #2: Network centric animation data generation

D.3b.1: UE1 sends source data for animation data generation to the MF over RTP (audio, video, text) or data channel (text).

D.3b.2: The MF processes the received source data to generate animation data during the session. The animation data may be transformed from the source data (e.g., from audio to text, video to motion data), or the same as the source data.

D.3b.3: The MF delivers animation data over data channel to the UE2 animating the base avatar. If network centric avatar animation is used, this step will be skipped. The animation data may be delivered to UE1 as well.

**D.4. Avatar Animation**

Based on the capability negotiation result in step A, the UE or network may animate the avatar.

Alternative #1a: Sender-centric avatar animation

[Optional] D.4a.1: UE2 delivers its pose information to UE1 for viewer-dependent avatar animation and rendering.

D.4a.2: UE1 animates and renders the base avatar using animation data. The animation data is generated by UE1 in step D.3a.1.1

D.4a.3: UE1 delivers the animated and rendered avatar to UE2. The animated and rendered avatar may be delivered as a 2D video through RTP.

D.4a.4: UE2 corrects the rendered video (for latency compensation) from UE1 before displaying as rendered avatar.

D.4a.5: UE2 delivers a report of timing information, including its actual display time to UE1 for the monitoring of the UE1 centric rendering service.

Alternative #1b: Receiver-centric avatar animation

D.4b.1: UE2 animates and renders the base avatar using animation data. The animation data may be generated by the MF, following steps D.3b.1 to D.3b.2 and received by UE2 in step D.3b.3 or it may be generated by UE1 in step D.3a.1 and received by UE2 in step D.3a.2.

D.4b.2: UE2 delivers a report of timing information, including its actual display time to UE1 (and MF) for the monitoring of the UE1 centric rendering service.

Alternative #1c: Network-centric avatar animation

D.4c.1: The MF animates and renders the UE1’s base avatar using animation data. The animation data may be generated by the MF, following step D.3b.1 and D.3b.2 or it may be received from UE1 following steps D.3a.1 and D.3a.2.

D.4c.2: The MF delivers the animated and rendered avatar to the UEs. In the figure, delivery to UE2 is shown as example. The animated and rendered avatar may be delivered as a 2D video through RTP.

D.4c.3: UE2 corrects the rendered video (for latency compensation) from the MF before displaying as rendered avatar.

D.4c.4: UE2 delivers a report of timing information, including its actual display time to UE1 (and MF) for the monitoring of the network-centric rendering service.

NOTE2: Rendering is not needed for 2D avatar.

\* \* \* Second Change \* \* \* \*

### 6.5.4 Timing Information for Avatar Animation and Rendering

In avatar communication, compared to traditional remote or split rendering for AR, one important parameter to estimate the user quality of experience is the *posture to render to photon time delay,* defined as the time duration between the *posture-capture-time* or *animation-data-generation-time*, the *avatar-animation-time*, and the *actual-display-time*. The calculation or measurement of the timestamps related to this delay is dependent on the entity performing the avatar animation and rendering, which is also influenced by the latencies involved in generating and delivering the animation data required at the corresponding entity.

The *posture-capture-time* is measured in the sending UE, as the time when the sender UE user’s pose is captured in order to generate the animation data.

The *animation-data-generation-time* is measured either in the sending UE or the MF, as the time when the animation data is generated from the source data.

The *avatar-animation-time* is measured, depending on the animation mode, either in the sending UE, MF or receiving UE, as the time when the base avatar is animated and rendered.

The *actual-display-time* is measured in the receiving UE, as the time when then rendered avatar is displayed to the user.

These timestamps may be delivered to each corresponding entity via feedback messages in order to facilitate better quality of experience. Better QoE may be provided to the user either through an adjustment in pose correction (in the receiving UE), or by other means such as the re-negotiation of a more suitable entity for animation data generation and/or avatar animation and rendering.

\* \* \* End of Changes \* \* \* \*