**Agenda item:** 10.7

**Source:** Qualcomm Inc., Nokia, Samsung, InterDigital

**Title: [AvCall-MED] Avatar Communication Call Flows**

**Document for** Discussion andAgreement

# Introduction

In this contribution, we propose content for the base CR on Avatar Communication that introduces the main call flow on Avatar communication as described in TR 26.813 [1].

# Proposed Changes

## A.1.6 Avatar Call Flows

A.1.6.1 General Avatar Call Flow



Figure 24: 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.

The list of Avatar ID(s) and/or Avatar Representations is downloaded to the UE by 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 (see details in step A.2).

**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.

Note: details of the avatar selection and negotiation are defined in A.1.6.3.

**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 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.

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.

NOTE2: Rendering is not needed for 2D avatar.

A.1.6.2 Avatar Management Call Flow

TBD.

A.1.6.3 Avatar Selection and Negotiation Call Flow

TBD.

# Proposal

We propose to add the content of section 2 to the base CR for AvCall-MED.

# References

[1] 3GPP TR 26.813, Avatar Representation and Communication (Release 19)

[2] 3GPP TS 23.228, IP Multimedia Subsystem (IMS); Stage 2 (Release 19)