**3GPP TSG-SA WG4 Meeting #129-e *S4-241509***

**Online 19th-23th August 2024**

**Source: HUAWEI**

**Title: [FS\_AVATAR] pCR on IMS architecture and call flow mapping**

**Spec: 3GPP TR 26.813 v0.5.0**

**Agenda item: 9.8**

**Document for: Discussion and agreement**

**1. Introduction**

In the SA4#128 meeting, the IMS architecture mapping to support 2D/3D avatar was introduced by S4-241309 and incorporated into the latest PD v0.4.0. During the previous discussion, it was asked to suspend the study of IMS mapping for avatar communication until the conclusion from SA2 is available. Since the conclusion from SA2 FS\_NG\_RTC\_Ph2 has been reached in clause 8.8 of TR 23.700-77 V1.0.0, this contribution proposes to move forward as follows:

1) Move the IMS architecture updates from PD to TR 26.813.

2) Add the IMS call flow to support both 2D and 3D avatar communication.

Comparing to S4aV240046, which was discussed during the post SA4#128 call, the content is further updated as:

3) Change Avatar Storage to BAR to align with the corresponding SA2 contribution.

4) Rename the interface name regarding BAR to DC6/DC7 to align with the corresponding SA2 contribution.

5) Remove the connection between BAR and UE to align with the conclusion of SA2 in TR 23.700-77 V1.0.0.

6) Separate the capability negotiation call flow into network centric and UE2 centric to make it clearer.

7) Update the call flow when ADC is used for capability negotiation.

8) Modify the animated avatar description to also include the rendering operation in the same entity due to feasibility consideration.

9) Move the Scene Management function from MF to DC AS.

**2. Reason for Change**

Update IMS architecture mapping and call flow for avatar communication.

**3. Proposal**

It is proposed to agree the following changes to 3GPP TR 26.813 v0.5.0.

\* \* \* 1st Change \* \* \* \*

## 8.6 Mapping to IMS-based Services

### 8.6.1 Architecture Mapping



**Figure 8.6.1-1: Mapping Avatar Functions to IMS Architecture**

Figure 8.6.1-1 shows an example of mapping the avatar functions to the IMS architecture, specifically the possible avatar functions which may be supported by the MF.

**BAR (Base Avatar Repository):**

- Avatar Storage: Store base avatars and associated avatar IDs.

**MF:**

- Base Avatar Generation: the MF may generate base avatar by the user input and stores the base avatar to BAR. For 3D avatar, the base avatar may be a 3D model or an INR model. For 2D avatar, the base avatar is comprised of a DNN model and a base image/video.

- Animation Data Generation: the MF generates animation data using conventional or AI/ML technologies based on the media received from the user.

- Avatar Animation: the MF generates or downloads the base avatar, and animates the base avatar using the received animation data.

**DC AS:**

- Support the subscription of avatar communication service and session control for avatar communication service.

Scene Management: supports the sdescription document management.description

Through such functions, the network may assist the UE with media processing related to the creation of avatar and animation data, as well the consumption of avatar data, in particular scene management/composition, and rendering.

For the support of avatar services based on the IMS architecture, media negotiation between the UE and network should include aspects related to:

* UE capability
* Network capability

The following media interface are used for the IMS Avatar Communication architecture.

- DC6: Reference point of Avatar representation downloading between DC AS and BAR.

- DC7: Reference point of Avatar representation downloading between MF and BAR.

\* \* \* 2nd Change \* \* \* \*

### 8.6.2 Call Flow

#### 8.6.2.1 Call Setup and Capability Negotiation

##### 8.6.2.1.1 Network Centric



**Figure 8.6.2.1.1-1: Network Centric Call Setup and Capability Negotiation Flow**

For network centric mode, the DC AS controls the capability negotiation procedure based on the avatar type (2D or 3D) and the capability information of UE and MF. The capability information includes the animation data type(s) (e.g., text, expression data and motion signals for joints) supported by UE or MF. After capability negotiation finished between UE and DC AS, the DC AS controls MF to download UE1’s base avatar from BAR, generate animation data by the source data received from UE1, and animate UE1’s base avatar by the animation data received from UE1 or generated by MF itself.

If UE1 supports data channel, the capability negotiation is done during the application data channel establishment. The application data channel can be used to transmit the animation/source data between UE1 and MF.

If UE1 does not support data channel, the capability negotiation is done during the avatar communication session establishment. The RTP channel is used to transmit the animation/source data between UE1 and MF.

A.0: DC AS subscribes IMS session event from IMS AS through NEF.

Alternative #1: UE1 support data channel

A.1: An audio/video session is established between UE1 and UE2.

A.2: The bootstrap and application data channels are established between UE1 and IMS.

A.3: The UE1 sends a capability negotiation request using the application data channel through MF to the DC AS. The message carries parameters including an avatar id chosen by UE1 and animation data types (e.g., text, expression data and motion signals for joints) supported by UE1.

A.4: The DC AS gets the avatar type (2D or 3D, from base avatar retrieved from an BAR or to be generated by the MF) by avatar id, and confirms the capability negotiation result based on the avatar type and the capability supported by UE1 and MF. The negotiation result includes the animation method (e.g., by audio, text or expression data and motion signals for joints).

A.5: The DC AS instructs the IMS AS through NEF to anchor UE1’s audio/video media and avatar animation media resource by session management request, which includes avatar id, avatar type and animation method.

A.6: The IMS AS requests the MF to allocate audio/video media and avatar animation media resources according to the session management request from DC AS.

A.7: The MF allocates audio/video media and avatar animation media resources based on the avatar type and animation method received from IMS AS. MF should support different media processing according the avatar type, e.g., animation with no rendering for 2D avatar, animation and rendering for 3D avatar. After the resources allocated successfully, the MF sends a successful response to the IMS AS.

A.8: IMS AS sends the session management response to the DC AS through NEF according to the response from MF.

A.9: The DC AS sends the capability negotiation response carrying the capability negotiation result to UE1 through the application data channel.

Alternative #2: UE1 doesn’t support data channel

A.10: The UE1 sends an INVITE message for audio/video session establishment to the IMS AS through P/S-CSCF. The message carries an avatar id chosen by UE1 and the animation data types supported by UE1 which are used for capability negotiation.

A.11: The IMS AS sends a session event notify carrying the avatar id and animation data types to DC AS through NEF.

A.12: The DC AS gets the avatar type (2D or 3D, from base avatar retrieved from an BAR or to be generated by the MF) by avatar id, and confirms the capability negotiation result based on the capability supported by UE1 and MF. The negotiation result includes the animation method (e.g., by audio or text or expression data and motion signals for joints).

A.13: The DC AS instructs the IMS AS through NEF to anchor audio/video media and allocate avatar animation resource by the session management request, which includes avatar id, avatar type and animation method.

A.14: The IMS AS requests the MF to allocate audio/video media and avatar animation media resources according to the session management request from DC AS.

A.15: The MF allocates avatar animation media resources based on the avatar type and animation method received. MF should support different media processing according the avatar type, e.g., animation with no rendering for 2D avatar, animation and rendering for 3D avatar. After the resources allocated successfully, the MF sends a successful response to the IMS AS.

A.16: IMS AS sends the session management response to the DC AS through NEF according to the response from MF.

A.17: The DC AS sends the session event notify response carrying the negotiation result determined in step A.12 to the IMS AS.

A.18: The audio/video session between UE1, MF and UE2 is established in subsequent procedure.

A.19: The IMS AS sends 18X/200 OK carrying the capability negotiation result to UE1 through S/P-CSCF.

##### 8.6.2.1.2 UE Centric



**Figure 8.6.2.1.2-1: UE Centric Call Setup and Capability Negotiation Flow**

For UE centric mode, if UE1 centric procedure is used, there is no capability negotiation initialized by UE1, so only UE2 centric call flow is introduced. If UE2 centric procedure is used, the base avatar of UE1 is sent from MF to UE2 using data channel, and the animation data is sent from UE1 or MF to UE2 using RTP or data channel. Therefore, the UE2 centric mode requires both UE1 and UE2 supporting data channel.

A.0: DC AS subscribes IMS session event from IMS AS through NEF.

A.1: An audio/video session is established between UE1 and UE2.

A.2: The bootstrap and application data channels are established between UE1 and IMS, UE2 and IMS.

A.3: The UE1 sends a capability negotiation request using the application data channel through MF to the DC AS. The message carries an avatar id chosen by UE1 and the animation data types supported by UE1.

A.4: The DC AS check if UE2 centric mode is used, then the DC AS transfers the capability negotiation request to UE2 through MF. The request carries the animation data types supported by MF in addition to the capability negotiation parameters.

A.5: The terminating network/UE2 finishes the capability negotiation.

A.6: The terminating network/UE2 returns the capability negotiation response carrying the negotiation result. The negotiation result includes the animation method (e.g., by audio, text or expression data and motion signals for joints).

A.7: The DC AS checks the animation method in the negotiation result, if the animation method doesn’t match the animation data types that supported by UE1, the DC AS determines that animation data generation should be done on MF.

A.8: If animation data generation should be done on MF, the DC AS instructs the IMS AS through NEF to allocate avatar animation resource by the session management request, the request includes animation data input and output type (e.g., “audio and text” means MF should generate animation data (text) by UE1’s audio input stream).

A.9: The IMS AS requests the MF to allocate avatar animation media resources according to the session management request from DC AS.

A.10: The MF allocates avatar animation data generation media resources based on the animation data input and output type received. Then the MF sends the response to the IMS AS.

A.11: IMS AS sends the session management response to the DC AS through NEF according to the response from MF.

A.12: The DC AS sends the capability negotiation response carrying the negotiation result to UE1 through S/P-CSCF.

A.13: The subsequent procedure continues.

#### 8.6.2.2 Avatar Delivery and Animation

 

**Figure 8.6.2.2-1: IMS Avatar Delivery and Animation Flow**

**A. Call Setup and Capability Negotiation**

An audio/video session is established between UE1 and UE2 and parameters of the session are negotiated as described in clause 8.6.2.1.

**B. Scene Description Retrieval**

 The MF and the participating UEs retrieve scene descriptions, the scene description may be shared by the MF with the UEs, 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. Avatar Loading**

D.1: The MF loads the base avatar (for 2D avatar, the base avatar is comprised of a DNN model and a base image/video; for 3D avatar, the base avatar can be a 3D model or an INR model) for UE1 from BAR.

**E. Avatar Delivery**

Alternative #1: UE1 centric

E.1: The MF delivers the base avatar to UE1 through data channel.

Alternative #1: UE2 centric

E.2: The MF delivers the base avatar to UE2 through data channel.

**F. Animation 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

F.1: The UE1 generates the animation data based on the source data (e.g., audio, video, text). The animation data may be transformed from the source data (e.g., from audio to text), or the same as the source data.

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

Alternative #2: Network centric animation data generation

F.3: UE1 sends source data (e.g., audio, video, text) to the MF through RTP or data channel.

F.4: 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), or the same as the source data.

F.5: The MF delivers animation data through RTP or 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.

**G. Avatar Animation:**

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

Alternative #1: UE centric avatar animation

Alternative #1a: UE1 does avatar animation

G1: UE1 animates and renders the base avatar using animation data. The animation data may be generated by the MF, following steps F.3 and F.4 or it may be generated by UE1 in step F.1.

G.2: UE1 delivers the animated and rendered avatar to UE2. The animated and rendered avatar (e.g., 3D or 2D video) may be delivered through RTP.

Alternative #1b: UE2 does avatar animation

G.3: UE2 animates and renders the base avatar using animation data. The animation data may be generated by the MF, following steps F.3 to F.4 and received by UE2 in step F.5 or it may be generated by UE1 in step F.1 and received by UE2 in step F.2.

Alternative #2: Network centric avatar animation

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

G.5: 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 (e.g., 3D or 2D video) may be delivered through RTP.

NOTE2: Rendering is not needed for 2D avatar.

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