Source: Samsung Electronics Co. Ltd

**Title: Analysis on Use Cases for Avatar communication service**

**Agenda Item: 9.10**

**Document for: Discussion and Agreement**

# **Introduction**

In the last 125th meeting, various use cases related to FS\_AVATAR were proposed based on the TR of SA1 FS\_Metaverse [TR 22.856], and through the offline discussions, they were organized into four use cases related to SA4, then have been included in the output of this study, TR 26.813 v0.1.0 [S4-231545].

Also, a reference architecture was proposed in the meeting [S4-231189]. It was considered as an initiation document on functional blocks that can be derived from the use cases.

Based on the use cases organized in the specification, this contribution identifies the types of inputs and outputs that the Avatar communication service should support and provide. It also lists and categorizes the processes that need to be considered for generating or transcoding outputs from such inputs.

To simplify the complexity of the description, this document proposes defining the *Avatar Model,* *Animation Commands* and *Animated Avatar*.

# **Analysis on Use Cases for Avatar communication service**

### 2.1 General

This clause lists the inputs, outputs, and relevant processes of the Avatar communication service derived from the use cases in the clause 5 of TR 26.813.

In this study's SID [SP-230544], both Avatar representation and Avatar animation are mentioned. However, in the use cases, the term "representation" is used for the results that reflect user intent, such as animated avatars or digital representations. Therefore, the following new terms are proposed for clear distinction.

**Avatar Model**: is the data storing the user's actual or preferred appearance and physical characteristics but at the state of neutral regarding expressions and posture (e.g., T-pose).

**Animated Avatar**: is the result of modifying the Avatar Model either statically or dynamically to represent it in accordance with the user's explicit intent.

**Animation Command**: is a set of command data representing the user's explicit intent to make the Avatar model behave as an Animated Avatar.

Note) This clause does not propose or restrict a specific technical solution or format for Avatar Model, Animated Avatar, or Animation Command. Rather, defining Avatar Model and Animated Avatar separately should be considered as a measure to avoid such technical constraints. For instance, in the case of a 3D format, Avatar Model and Animated Avatar can be the same. In another example, in the case of an AI-based Avatar, the Avatar Model may be a trained AI model, and the Animated Avatar could be a 2D or 3D video.

The subsequent clauses classify and collect descriptions from the use cases, and organize the specified data from the collection. Inputs, outputs, and processes are each categorized into Clause 2.2, 2.3, and 2.4, respectively.

### 2.2 Input

Information provided, selected, or captured from the user to create Avatar Models and/or result Animated Avatar is collected in this clause.

Head movement, hand gestures, and other data such as 3D audio acquired from the user or their surroundings are continuously acquired and digitized by devices for Avatar services without the need for manual specification by the user. This enables synchronization with the user to the Animated Avatar. In addition, various associated devices like haptic gloves, body suits, [motion] sensors, and cameras were listed with the use cases.

- Heads movement

- Hands gestures

- Information from Haptic gloves/body suit

- Facial expression

- 3D audio

Users can explicitly indicate or select their intent to control the Avatar Model. Examples such as "A choice of using a full body or a head-only avatar" or "using a different outfit." are the explicit ways in which users control the Avatar Model. One of the possible ways of processing such an explicit intent can be the Avatar service app on the UE.

- Selection of Avatar Model

- Selection of outfit for the Avatar Model

- Selection of the display range of the Avatar Model

Furthermore, the Avatar is also considered for use in interactions between computer-generated systems and humans. The computer-generated systems can represent simulated entities as Avatars to interact with users or users' Avatars. It is unclear from the use case description whether the input generated from the computer-generated systems are sensed information (e.g., heads movement), Animation Command or Animated Avatar.

- Input from computer-generated systems

Table 2.2.1 is the collection of descriptions related to Input from the Use Cases. Figure 2.2.1 illustrates the transfer of input to the Avatar service through capturing components attached to the UE.

|  |  |  |
| --- | --- | --- |
| category | from | description |
| input | uc1 | **3D audio** can be captured and transmitted |
| input | uc1 | her glasses detect her **heads movement** and her avatar is animated accordingly |
| input | uc1 | **Sensor information** from the terminals of the participants is sent to the server to generate their animated avatars |
| input | uc3 | She is using the **controllers** connected wirelessly to the HMD to play the game and she is wearing a haptics body suit |
| input | uc2 | She puts on her HMD and **haptic gloves** / Philippe puts on his HMD and gloves |
| input | uc3 | The captured **sensing data** are sent to the cloud or the edge server |
| input | uc1 | the information captured by the **sensors and cameras** on User B's terminal |
| input | uc3 | The terminals of the participants are equipped with **sensors and cameras** |
| input | uc3 | They opt for using **hand gestures** to play the game |
| input | uc1 | User B's avatar is now animated through the **movement sensors** on her phone |
| input | uc4 | User A confirmed that the digital representation could perfectly represent User **A's facial expressions and body gestures**. |
| input/CG | uc1 | a **computer-generated system** where an avatar is used to generate an appearance for a simulated entity |
| input/CG | uc3 | Some avatars may not be user controlled but they are **software agents** controlled by the game engine |
| input/user decision | uc1 | a **choice** of using a full body or a head-only avatar |
| input/user decision | uc1 | a **pre-recorded 3D representation** (avatar) of her using a different outfit |

Table 2.2.1 The collection of descriptions related to the Input from the Use Cases



Figure 2.2.1 The inputs transferred to the Avatar service

### 2.3 Output

Input to the Avatar service can be applied to the Avatar Model to transform it into an Animated Avatar. The Animated Avatar includes not only visual outputs but also auditory and sensory (haptic) outputs. 2D video or 3D avatar were the examples of the representation format for the said devices and the visual quality of the avatar is expected to be a photo-realistic.

- 2D video or 3D avatar

- Photo-realistic visual quality

- 3D audio

- Haptic feedback

From a perspective of spatial representation, the Animated Avatar is represented as an object within the VR space (e.g., virtual meeting room) or as an AR object augmented into the user's real-world space. In some use cases, there are scenarios where VR and AR users are included in the same Avatar call. The common scene (if occupied) that accommodates the Avatars of these users is represented differently for VR users and AR users. When one side of the users adds an object (e.g., VR users sharing his laptop display), it can be implemented to affect the other side of the users as well (e.g., the shared display is overlaid on the AR user's display). The Scene Manager would identify object (such as table) of the virtual meeting room scene whether they are for VR user or AR user, then corresponding interactions on those objects should be handled correspondingly.

- VR (avatars within a virtual meeting room/virtual office)

- AR overlay

- AR or VR depends on user's device form factor

For the form-factor of the device perspectives, HMD, AR glasses and phone can be occupied.

- HMD

- AR glasses

- Phone

Table 2.3.1 is the collection of descriptions related to Output from the Use Cases. Figure 2.3.1 illustrates the avatar service output and the processing blocks derived from MeCAR discussions, such as the Scene Manager, Presentation Engine, and XR Runtime. Depending on the Avatar use case, the Scene Manager manages both VR and AR sessions, enabling interactions between inputs from one side with the other. As a result, the received Scene includes nodes for representing Animated Avatars, audio, and haptic feedback.

Note) Nodes and buffers in the diagram are presented to show the relationship between the scene and resources. It should not be understood based on specific technologies, as the figure is intended to be technology-agnostic. They can be further simplified, or specified in a form common to at least multiple different technologies agreed, if necessary.

|  |  |  |
| --- | --- | --- |
| category | from | desc |
| output/audio | uc1 | **avatars and audio** of each participant in avatar call are transmitted |
| output/format | uc1 | the encoded representation is sent to User A's terminal for rendering either as a **2D video or a 3D avatar** |
| output/haptics | uc3 | User B senses a **vibration** in her body suit and she realizes that her avatar was hit in the game |
| output/haptics | uc3 | This interaction may also wearables that the users have on their hands and bodies which provide **haptic feedback** in response to actions and interactions |
| output/AR | uc1 | the avatars are then sent to User A and Frank's terminals for **overlaying** |
| output/AR | uc3 | avatar **augmented** to their surroundings |
| output/AR | uc4 | both of whom also accepted and joined the service, and soon **appeared in User A's living room** |
| output/AR | uc2 | Layla is only able to join the call her **AR glasses** and headphones / While she is unable to see the complete virtual office |
| output/ARVR | uc1 | **spatially rendered** |
| output/ARVR | uc1 | While User A and Frank see an **augmented** avatar, Mike can see his colleagues' avatars **within a virtual meeting room.** |
| output/ARVR | uc1 | **The screen contents are projected in the virtual meeting room** that Mike sees on his HMD and **are also augmented as an overlay** on Frank's glasses. |
| output/VR | uc2 | Both Sarah and Philippe can now see **each other's avatar in the virtual office** |
| output/quality | uc1 | a **photo-realistic** digital representation (avatar) of the other two rendered on his display |

Table 2.3.1 The collection of descriptions related to the Output from the Use Cases



Figure 2.3.1 The outputs transferred from the Avatar service

### 2.4 Process

From the use cases, processing blocks to generate the Animated Avatar from the inputs were collected as Table 2.4.1. The processing blocks are intended to align with the use case descriptions, regardless of any specific format or technology for the Avatar Model and the Animation Command.

- Animation Command Generator generates the Animation Command.

- Avatar Model Generator generates the Avatar Model from the inputs such as captured video from camera and other sensors information.

- Avatar Model Animator generates the Animated Avatar from the Avatar Model according to instructions from the Animation Command.

- Storage stores the Avatar Model, authenticates the requesting party, and transfers the Avatar Model. It can be located in the Avatar Model owner's UE, the network's Media Function, and the counterpart's UE.

|  |  |  |
| --- | --- | --- |
| category | from | desc |
| process/animation | uc3 | the cloud or the edge server which use this information [The captured sensing data] for **coding and rendering** to generate the digital representations (avatars) |
| process/animation | uc1 | the information captured by the sensors and cameras on User B's terminal is **processed locally to encode and animate** her avatar |
| process/model | uc4 | Digital representations may be **generated** by the service provider, and can be created using an app installed on the user's device or by functions provided by the 5G network |
| process/model | uc4 | Users may **update** the digital representation they have stored in the network or transmitted to others |
| process/model | uc4 | The app captured the hand again, and the network **updated** the digital representation in real-time, providing the updated digital representation to User A, B, and C simultaneously. |
| process/scene | uc1 | (the avatars) are also **included as part of the scene description** for the virtual environment |
| process/split computing | uc1 | **depending on the terminal's capabilities**, the captured information may be transmitted over an uplink to an AS associated with the established session |
| process/storage | uc4 | it may be **pre-uploaded and stored in the network** or **transmitted to the other party** via the network before the start of avatar service |

Table 2.4.1 The collection of descriptions related to the process from the Use Cases

As a result of the analysis of the use cases in this clause, inputs, outputs, and the processing between them have been identified. Figure 2.4.1 illustrates a reference Avatar architecture and delineates the end-to-end processes for Avatar processing as the Avatar Processing Block.

1) The inputs from user's selection such as choosing a specific Avatar Model and selecting an outfit for it, are linked to the Avatar Model Animator.

2) The inputs from user's selection on controlling the display range of Avatar presentation is linked to the Scene Manager. It can be linked to the Avatar Model Animator to generate the only portion of the Animated Avatar Model required for the display range.

3) The Animation Command from the Animation Command Generator includes instructions on actuating the Avatar Model according to the user's expressions and gestures.

4) The Animation Command may include instructions on moving the Avatar in a spatial space for the use cases such as VR virtual meeting room.

5) The type or format of the Animated Avatar Model can be considered as 3D.

6) The Scene Manager may require pose (or position) of participants to map the location of their Avatar in the VR and AR space.

7) The Scene Manager may require user's surrounding information (e.g., geometry) to map and track the shared objects in each participants' AR space differently.

8) The output from Presentation Engine can be considered as 2D. The 2D may not require additional post-processing such as projection.

9) The Avatar Model can be pre-recorded or updated in real-time.

10) The Avatar Model can be stored in the local storage of UE1 (the owner of the Avatar Model), pre-uploaded in the network storage, and pre-downloaded to the local storage of UE2 (the counterpart's). As delivering the Avatar Model is expected to be time-consuming task, it is assumed that the model will be pre-uploaded to the network and pre-downloaded to the counterpart's local storage before establishing the Avatar call.

11) Each process within the Avatar Processing Block can be executed by either UE1, UE2, or network Media Functions. The decision regarding the execution location may vary for each session and each participant's connection to the Avatar call service. The instantiation of the required processes on UE or network and the establishment of necessary media/data delivery connections are negotiated then determined based on the factors such as device type, performance, and the user's eligibility for subscriptions on the network services.

Note) The decision on where to link the input from the Computer-generated system to the processing block is FFS (indicated by dashed lines).



Figure 2.4.1 Consolidated end-to-end processes from Avatar use cases

Figure 2.4.2 is an avatar reference architecture that focusing processing blocks in Figure 2.4.1. The processing blocks highlighted in gray designed to illustrate the overall flow of avatar processes demonstrating adaptability to changes in use cases. The inputs and outputs of each processing block serve as points where standardized interfaces can specify data formats that are interoperable, even if different technologies are employed within the processing blocks.



Figure 2.4.2 Avatar reference architecture

# **3. Proposal**

It is proposed to add clause 2 into clause 5 and 7 of TR 26.813. Separate contribution S4-231766 provides pCR to incorporate the proposed changes to the TR.