**3GPP TSG SA WG4 Meeting # 127 *S4-240196rev1***

**Sophia Antipolis, France, 29th Jan - 2nd Feb, 2024**

|  |
| --- |
| *CR-Form-v12.0* |
| **PSEUDO CHANGE REQUEST** |
|  |
|  | **26**.**813** | **CR** |  | **rev** |  | **Current version:** | **0.2.1** |  |
|  |
| *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 |  | Radio Access Network |  | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  | **pCR on Generic Avatar AR Call Flows** |
|  |  |
| ***Source to WG:*** | Nokia, Huawei |
| ***Source to TSG:*** | SA4 |
|  |  |
| ***Work item code:*** | FS\_AVATAR |  | ***Date:*** | 23rd Jan 2023 |
|  |  |  |  |  |
| ***Category:*** | **B** |  | ***Release:*** | 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-12 (Release 12)**Rel-13 (Release 13)Rel-14 (Release 14)Rel-15 (Release 15)Rel-16 (Release 16)* |
|  |  |
| ***Reason for change:*** | Proposal to document generic avatar call flows |
|  |  |
| ***Summary of change:*** | Generic call flow for avatar generation and animation to support UC1 |
|  |  |
| ***Consequences if not approved:*** | Incomplete TR |
|  |  |
| ***Clauses affected:*** | 8.1 |
|  |  |
|  | **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:*** |  |

# Introduction

At SA plenary # 100 a study item FS\_AVATAR was agreed in document SP-230544. Among the objectives of the study is to document the network procedures for avatars and to investigate integration of avatars into RTC services and impact thereof on 5G-RTC architecture. In this contribution we propose generic call flows for conversational real-time animated avatar-based communication, specifically avatar animation. While the proposed network procedures may be relevant to all the use cases in clause 5 of draft TR 26.813 of FS\_AVATAR Study Item, they are especially relevant to avatar loading and animation requirements of UC1 (clause 5.1) and UC4 (clause5.4).

It should be noted that the proposed network procedures are broadly in alignment with AR call procedures for Avatars agreed as basis of future work in the Permanent Document (v0.61) of IBACS work item.

# 2 AR call with avatars

Avatars are defined in TR 22.856 [1] as a digital representation specific to media that encodes facial (possibly body) position, motions and expressions of a person or some software generated entity. As digital representations of a user’s face, avatars may range from photorealistic to cartoonish in their likeness to the user. Users may participate in AR calls as 3D or 2D avatars. This scenario is captured in clauses 5.11 and 5.26 of TR 22.856 and consequently in clause 5.1 (UC1: Avatar Communication) of FS\_AVATAR draft TR 26.813. 3D avatars may be represented, rendered, and animated using 3D models. 2D avatars are usually represented in the form of 2D videos or images that may be generated from deep neural networks.

3D modelling of real-time objects may be done from a series of images of the object captured from different angles. The same approach may be used to generate 3D models of a human, these images may be provided by the user by capturing while moving the camera or by capturing with multiple cameras. This can be done offline or online depending on the use case and implementation. Online avatar generation is becoming faster day by day due to advances in computer vision and machine learning, for example, [Vid2Avatar](https://github.com/MoyGcc/vid2avatar) , [I M Avatar](https://github.com/zhengyuf/IMavatar), [Hi4D](https://yifeiyin04.github.io/Hi4D/) and [Realtime reconstruction of an animating human body from a single depth camera](https://orca.cardiff.ac.uk/id/eprint/87637/).

A 3D model of a humanoid may be represented using a 3D representation format, e.g., glTF2.0. After appropriate rigging and skinning procedures the model may be animated. To animate a 3D model, animation data in the form of motion signals may be generated from the user’s movements and facial expressions as captured by one or more cameras. The method for generating motion signals depends on the implementation, and may be carried out at a UE or by a media function in the 5G network.

A 2D digital human (i.e. avatar) whose figure is a planar image with graphic content containing information in horizontal and vertical dimensions. Unlike 3D avatars, 2D avatars are mainly used in scenarios where the avatar’s position and posture are fixed so the service may be provided in a much simpler solution with and less computational cost (e.g. rendering).

Before the call, a personalized 2D avatar model may be generated by some offline training process on Deep Neural Networks (DNNs). The input data for model training is usually recorded user videos. The trained DNN model may work as the base model of the 2D avatar for online usage. Unlike 3D models, such 2D avatar base models themselves don’t have visualizable representations directly, but may generate visible animated avatars in the form of 2D videos.

During the call, a user may use voice or text to drive the DNN model to generate the animated avatar as the lip-sync video stream, then send it to the other party. The method for generating the animated avatar video is implementation dependent, and may be carried out at a UE or by a media function in the 5G network.

|  |
| --- |
| **Start of Change 1** |

8.1.1 Avatar AR call

Figure 8.1.1 shows a generic call flow for an avatar based call over the 5G Network. In addition to data transmission, when supported, the network provides network assistance for avatar generation, animation and retrieval. In the call flows the network entities providing avatar generation, animation and retrieval functions are generalized as a network media function, which may be for example, an RTC-AS an MRF or an MCU. Further, an avatar storage entity is illustrated which may be an avatar repository, UE storage or a cloud storage.

The figure shows alternative flows which correspond to different possible mappings of functional blocks identified in the reference architecture of clause 7. The mappings are highlighted with a call out box when applicable.



Figure 8.1.1 Avatar AR call flow

1. **Call Setup**
	1. A session is established between UE1, Media Function and UE2 and parameters of the session are negotiated. This may include exchanging capability information, media and metadata descriptions and formats, resource discovery etc. The involved entities agree on assignment of avatar generation, animation tasks and media requirements.
2. **Scene Description Retrieval**

 The media function and the participating UEs retrieve scene descriptions, the scene description may be shared by the Media function with the UEs, or the UEs may have their own scene descriptions

1. **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 Media Function itself or the UEs. The UEs may update their scene descriptions independently or the MRF may generate an updated scene description and share it with the UEs

1. **Media and Metadata exchange**
	1. *Avatar Acquisition: In this step a base avatar is acquired by the media function.*

*Alternative #1: Network centric avatar generation*

* + 1. UE1 sends data for avatar generation to the media function. The data may be images (RGB or RGB-D), streamed to the media function as image or video stream(s). The media description of the streams may contain the camera configuration as well.
		2. The media function processes the received data to create a base avatar, for example, a rigged and/or skinned 3D model or a 2D model.

*Alternative #2: Network centric avatar loading*

* + 1. The media function loads an avatar for UE1 from an avatar storage which may be an avatar repository or storage on UE1.
	1. The media function delivers the base avatar to UE2
	2. *Animation Data Generation*

*Alternative #1 Network centric animation data generation*

* + 1. UE1 sends source data to the media function. The source data may include images, video stream or voice stream, text. The media description of the streams may contain the camera configuration as well.
		2. The media function processes the received data to create animation data during the session. The animation data may include text, expression data and motion signals for joints.
		3. The media function delivers animation data to the UE animating the base avatar. In the diagram UE2 is shown as the recipient for clarity. The animation data may be delivered to UE1 as well.

*Alternative #2 UE centric animation data generation*

* + 1. UE1 creates animation data based on data like images, video, audio or text. The animation data may include text, expression data and motion signals for joints.
		2. UE1 delivers the animation data to the entity actuating avatar animation. The animating entity may be the media function or UE2.
	1. Avatar Animation

*Alternative #1 Network centric avatar animation*

* + 1. The media function animates the base avatar using animation data. The animation data may be generated by the media function, following step D.3.1 and D.3.2 or it may be received from UE1 following steps D.3.4 and D.3.5
		2. The media function delivers the animated avatar to the UEs. In the figure, delivery to UE1 is shown as example. The animated avatar may be delivered, for example, as 3D (e.g. video with depth and multi-view information) or 2D video.

*Alternative #2 UE centric avatar animation*

*Alternative #2a UE1 does avatar animation*

* + 1. UE1 animates the base avatar using animation data. The animation data may be generated by the media function, following steps D.3.1 and D.3.2 or it may be generated by UE1 in step D.3.4.
		2. UE1 delivers the animated avatar to UE2. The animated avatar may be delivered, for example, as 3D (e.g. video with depth and multi-view information) or 2D video.

*Alternative #2b UE2 does avatar animation*

* + 1. UE2 animates the base avatar using animation data. The animation data may be generated by the media function, following steps D.3.1 and D.3.2 and received by UE2 in step D.3.3 or it may be generated by UE1 in step D.3.4 and received by UE2 in step D.3.5.
	1. Avatar Rendering
		1. UE2 renders the animated avatar, using for example, viewport and pose of the user. In case of 2D avatars, only decoding and display of the received 2d video may be needed.

|  |
| --- |
| **End of Change 1** |

# 3 Proposal

We propose to implement the above change in TR26.813.