**3GPP TSG-SA4 Meeting # S4-231795**

**, US,** *revision S4aV230061*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.2* | | | | | | | | |
| **PSEUDO CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** | pseudo | **rev** | **1** | **Current version:** |  |  |
|  | | | | | | | | |
| *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 | **X** | Radio Access Network |  | Core Network |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | [MeCAR] pCR on latency metrics definition | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | InterDigital Finland Oy | | | | | | | | | |
| ***Source to TSG:*** | S4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | MeCAR | | | | |  | ***Date:*** | | | 13-11-2023 |
|  |  | | | |  | |  | | |  |
| ***Category:*** |  |  | | | | | ***Release:*** | | | Rel-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 can be 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-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18) Rel-19 (Release 19)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | Add a list of information observed at the Ops and populating the metrics definitions section with latency metrics | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Remove split rendering related text following the comments received during the Video AH telco.   * In sections 9.1.2 to 9.1.5, defining the observed information that is later used in the metrics definition. * Adding section 9.2 “Metrics Definition”. * Adding section 9.2.1 “Latency metrics” that define latency metrics and the time information used to compute the latencies. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | WID objectives on QoE are not fulfilled. Information is missing and editor’s note are not addressed in OPs. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 9.1.2, 9.1.3, 9.1.5, 9.2 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | | Rev 1: remove the change in section 6.2.2 pose prediction format. Remove split rendering related metrics. | | | | | | | | |

|  |
| --- |
| 1. Change |

# 9 QoE metrics

[Editor’s note: related WID objectives

Identify which QoE metrics from VR QoE metrics can be reused or enhanced for AR media (e.g., resolution per eye, Field of view (FOV), round-trip interaction delay, etc.) and define relevant KPIs that are dedicated to AR/MR

Specify additional relevant KPIs and simple QoE Metrics for AR media]

## 9.1 Metrics and Observation Points

### 9.1.1 Overview

The Observation Points (OPs) are defined to support the definition of the corresponding metrics. This specification defines four observation points as shown in Figure 9.1.1-1. The metrics collection function, as part of the Media Session Handler, is responsible of collecting specific information observed at each OP in order to generate the metrics. This function has also access to the 5G System such that the metrics can be reported to an external entity.



Figure 9.1.1-1 - Observation Points in the XR Baseline Client

9.1.2 Observation Point 1: XR Runtime information

Observation point 1 (OP-1) is derived from the XR Runtime API. The OP-1 observes information exchanged between the XR Runtime on one side and the XR Source Management, the Presentation Engine and the application on the other side, i.e. on IF-1.

On observation point 1, the following observed information is defined:

[Editor’s note: define the observed information that is later used in the metrics definition]

- XR runtime clock

- Actual presentation/display time

- Actual playout frame rate

- Viewer pose prediction and pose prediction parameters

- Projection parameters

- Tracking pose prediction parameters

- user input actions and the time when the action is made

- Rendering loop status

- Camera information

9.1.3 Observation Point 2

Observation point 2 (OP-2) observes information at the input of the Scene Manager, i.e. on IF-9 for data received from the Media Access Function and the IF-10 for information exchanged between the Scene Manager and the application.

On observation point 2, the following observed information is defined:

[Editor’s note: define the observed information that is later used in the metrics definition]

- Media resolution

- Media codec

- Media frame rate

- Media decoding time

9.1.4 Observation Point 3

Observation point 3 (OP-3) is derived from the API which exchanges information between the XR Source Management and the Media Access Functions. It corresponds to the IF-3 interface.

On observation point 3, the following observed information is defined:

[Editor’s note: define the observed information that is later used in the metrics definition]

9.1.5 Observation Point 4

Observation point 4 (OP-4) observes information between the Media Access Function and the 5G System, i.e. on IF-4 interface.

On observation point , the following observed information is defined:

[Editor’s note: define the observed information that is later used in the metrics definition]

- The media type

|  |
| --- |
| 1. Change |

|  |
| --- |
| 1. Change |

9.2 Metrics Definitions

9.2.1 Latency metrics

To enable good XR experiences, it is relevant to monitor latencies such as the motion-to-photon and the pose-to-render-to-photon.

Beyond the sense of presence and immersiveness, the age of the content and user interaction delay are of the uttermost importance for immersive and non-immersive interactive experiences, i.e. experiences for which the user interaction with the scene impacts the content of scene (such as online gaming).

Table 9.2.1-X1 provides time information that may be gathered to compute the latency metrics. The observation points to collect the time information are indicated per device type.

**Table 9.2.1-X1: Time information for latency metrics**

|  |  |  |
| --- | --- | --- |
| **Observation Point** | **Time information** | **Definition** |
| OP-1 | estimatedAtTime (ref. T1) | The time, in units of milliseconds, when the viewer pose prediction is made. It corresponds to the time when the predicted viewer pose is collected using the XR runtime API-1 by the application or the XR Source Manager. |
| OP-1 | lastChangeTime | The time, in units of milliseconds, when the user action is made. It corresponds to the lastChangeTime field defined in the action format in Table 5.1.3-1. |
| OP-2 | sceneUpdateTime (ref. T6) | The time, in units of milliseconds, when the Scene Manager starts to update the 3D scene graph according to the viewer pose and the user actions. |
| OP-1 | startToRenderAtTime (ref. T3) | The time, in units of milliseconds, when the renderer starts to render the scene according to the viewer pose. |
| OP-1 | actualDisplayTime (ref. T2.actual) | The actual display time, in units of milliseconds, of the rendered frame in the swapchain. The estimation of the actual display time is available through the XR runtime. |

The latency metrics are specified in Table 9.2.1-X2. The formula to compute the latencies are defined using the collected time information.

**Table 9.2.1-X2: Latency metrics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Key** | | | **Type** | **Description** |
| latency | | | List | List of latencies |
|  | *Entry* | | Object |  |
|  |  | motionToRenderToPhoton | Integer | The time duration, in units of milliseconds, between the time to query the pose information from the XR runtime to the renderer (the renderer uses this pose to generate the rendered frame) and the display time of the rendered frame. It can be computed as follows: actualDisplayTime – estimatedAtTime |
|  |  | renderToPhoton | Integer | The time duration, in units of milliseconds, between the start of the rendering by the Presentation Engine and the display time of the rendered frame. It can be computed as follows: actualDisplayTime – startToRenderAtTime |
|  |  | roundtripInteractionDelay | Integer | The time duration, in units of milliseconds, between the time a user action is initiated and the time the action is presented to the user. It can be computed as follows: actualDisplayTime – lastChangeTime |
|  |  | userInteractionDelay | Integer | The time duration, in units of milliseconds, between the time a user action is initiated and the time the action is taken into account by the content creation engine in the scene manager. It can be computed as follows: sceneUpdateTime – lastChangeTime |
|  |  | ageOfContent | Integer | The time duration, in units of milliseconds, between the time the content is created in the scene by the Scene Manager and the time it is presented to the user. It can be computed as follows: actualDisplayTime – sceneUpdateTime |
|  |  | sceneUpdateDelay | Integer | The time duration, in units of milliseconds, spent by the Scene Manager to update the scene graph. It can be computed as follows: startToRenderAtTime – sceneUpdateTime |

|  |
| --- |
| 1. Change |