**3GPP TSG-WG SA4 Meeting #124**

**Berlin, DE, 22nd – 26th May 2023 *(revision of S4-230798)***

|  |
| --- |
| *CR-Form-v12.2* |
| **CHANGE REQUEST** |
|  |
|  | **26.565** | **CR** |  | **rev** |  | **Current version:** | **0.4.0** |  |
|  |
| *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 |  |

|  |
| --- |
|  |
| ***Title:***  | SR Rendering API |
|  |  |
| ***Source to WG:*** | Qualcomm Inc. |
| ***Source to TSG:*** | S4 |
|  |  |
| ***Work item code:*** | SR\_MSE |  | ***Date:*** | 16th May 2023 |
|  |  |  |  |  |
| ***Category:*** | B |  | ***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 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-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19)* |
|  |  |
| ***Reason for change:*** | This pCR introduces the SR API that is exposed by the SRC to the application. |
|  |  |
| ***Summary of change:*** |  |
|  |  |
| ***Consequences if not approved:*** |  |
|  |  |
| ***Clauses affected:*** |  |
|  |  |
|  | **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:*** |  |

|  |
| --- |
| **First Change** |

## 9.1 Functionality

The Split Rendering Client (SRC) is a function that runs on the UE to provide split rendering functionality to applications. The SRC is designed to be offered as an SDK to application developers. The SRC abstracts the details of the split rendering operation and provides a simple to use API to the application to facilitate the usage of split rendering.

The SRC performs the following functions:

* Creates and manages the XR session
* Discovers and selects a split rendering server (SRS) in the network
* Establishes a split rendering session with the SRS
* Communicates the necessary information about the session to the MSH/AF to benefit from dynamic policy, network assistance, consumption reporting, etc.
* Operates the rendering loop on the UE

## 9.2 Client API

As described in clause 5.1.3, the SRC exposes an API over RTC-7 interface to the application. The SRC defines the following interface:

| **Method** | **State after Success** | **Description** |
| --- | --- | --- |
| SplitRenderer() | STATE\_READY | Creates a SplitRenderer object, which can subsequently be used to connect to an SRS and perform split rendering. |
| connect() | STATE\_CONNECTED | Instructs the SRC to discover and connect to an SRS. |
| disconnect() | STATE\_DISCONNECTED | Terminates the connection to the SRS. |
| getMetrics() | - | Retrieves a set of metric reports for the split rendering session that describe the quality of experience of the session. |

The SplitRenderer interface is defined using the IDL syntax (according to ISO/IEC 19516) as follows:

|  |
| --- |
| interface SplitRenderer { readonly attribute SRState state; attribute EventHandler onstatechange; attribute EventHandler onerror; attribute EventHandler onqualitychange; void SplitRenderer(String application\_id); void connect(); void disconnect(); Metrics getMetrics();}; |

The application is able to subscribe to events related to the split rendering session by setting the corresponding event handler. The supported events are:

* State change: the state of the SR session has changed
* Error: an error has occurred during the split rendering session. The error is not severe enough to cause a state change to the STATE\_ERROR state.
* Quality change: the SRC has observed a change in the quality of the split rendering session. This may involve one or more SR metrics.

## 9.3 Split Rendering Metrics

Editor’s Note: TBD.