**3GPP TSG- Meeting #**

**, , -**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.2* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** |  | **rev** |  | **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 | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** |  | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** |  | | | | | | | | | |
| ***Source to TSG:*** | S4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** |  | | | | |  | ***Date:*** | | |  |
|  |  | | | |  | |  | | |  |
| ***Category:*** |  |  | | | | | ***Release:*** | | |  |
|  | *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:*** | | Annex Y.6.9.3 describes the SDP signaling for scene description-based overlays via bootstrap data channels using sub-protocol “mpeg-sd” with stream IDs below 1000. However, it is stated in cluase 6.2.10.1 that data channel stream IDs below 1000 must be reserved for using HTTP protocol. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | It is proposed to update the text in clause 6.2.10.1 to allow the use of sub-protocol “mpeg-sd” for bootstrap data channels. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | There will be an inconsistency on sub-protocol for bootstrap data channels. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.2.10.1, Annex Y.6.9 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | |  | | | | | | | | |

## ================= Start of the 1st change ================

#### 6.2.10.1 General

Support of data channel media is optional for an MTSI client and an MTSI client in terminal. For brevity, an MTSI client supporting data channel is henceforth denoted as a DCMTSI client or DCMTSI client in terminal, respectively.

To indicate support for the procedures in this clause, a DCMTSI client shall when including media feature tags as specified in TS 24.229 [7] include a +sip.app-subtype media feature tag, as specified by RFC 5688 [177], with a value of "webrtc-datachannel" (the application media format used by [172]), regardless of data channel media being part of the SDP or not.

One or more data channel SDP media descriptions formatted according to [172] may be added to the SDP, alongside other SDP media descriptions such as e.g. speech, video, and text. A data channel SDP media description shall not be placed before the first SDP speech media description. SDP examples are provided in Annex A.17.

If data channels are used in a session, the session setup shall determine the applicable bandwidth limit(s) as defined in clause 6.2.5.

Multiple data channels may be mapped to a single data channel SDP media description, each with a corresponding "a=dcmap" SDP attribute and stream IDs that are unique within that media description. There is no limit to the number of data channels in an SDP media description, but the aggregate of all defined data channels shall keep within the set bandwidth limit and care should be taken to avoid excessive SDP size. If the session is re-negotiated to include a changed number of data channels in an SDP media description, the bandwith limit may either be kept constant, changing the share of bandwidth available to each individual data channel, or the bandwidth limit may be changed to accommodate the changed number of data channels, keeping individual data channel bandwidth shares. Regardless of what approach is used when changing number of used data channels in a media description, the aggregate of all defined data channels shall keep within the re-negotiated bandwidth limit.

If there is a need to use data channels with either different transport IP addresses, different UDP ports, or different SCTP ports, separate data channel SDP media descriptions shall be used, as IP address, UDP port and SCTP port are all constant per SDP media description. Multiple SCTP associations for a single channel, commonly denoted as "multi-homing", defined in IETF RFC 4960 [173] for reasons of redundancy and basically using one destination transport address at a time, is not described for use with WebRTC data channel and shall therefore not be used in this specification.

NOTE 1: The main reasons to not specify multi-homing are because it cannot use the needed separation of signalling paths for redundancy purposes in the applicable usage scenarios, and it is also not considered feasible when using SCTP on top of DTLS.

To ease data channel media implementation and ease interworking with WebRTC data channels, DCMTSI clients shall support ICE Lite and may support full ICE [184], for data channel media. DCMTSI clients supporting full ICE shall only use host candidate addresses. SDP "a=candidate" line host address information shall match corresponding SDP "c=" and "m=" line information.

NOTE 2: In typical IMS deployments, it is expected that DCMTSI clients have no need to use STUN or TURN servers with ICE. This is in line with what constitutes an ICE Lite agent.

A "data channel application" consists of an HTML web page including JavaScripts(s), and optionally image(s) and style sheet(s). A "bootstrap data channel" is henceforth defined as a data channel used to retrieve data channel application(s) for a DCMTSI client in terminal, with a data channel stream ID below 1000, and using the HTTP [73] protocol as data channel subprotocol. The data channel application accessible at the HTTP root ("/") URL through a bootstrap data channel describes the graphical user interface and the logic needed to handle any further data channel usage beyond the bootstrap data channel itself. The meaning of the "authority" (host) part of the URL and consequently the "Host" HTTP header are not defined, shall be ignored on reception, and shall be set to the empty value by a DCMTSI client in terminal.

Data channel stream IDs below 1000 may use a well-defined subprotocol for other features than retrieving data channel application(s). For example, tstream ID below 1000

The data channel application is created prior to the DCMTSI call where it is intended to be used, by means left out of scope for this specification. The data channel application workflow is depicted by Figure 6.2.10.1-1 below.



Figure 6.2.10.1-1: Data Channel Workflow

The data channel application is, referring to the numbered arrows in Figure 6.2.10.1-1:

1. Uploaded to the network, by the UE user or some other authorized party.

2. Stored in a data channel application repository in the network.

3. During the DCMTSI call where it should be used, retrieved from the repository.

4. Sent through a bootstrap data channel to the local UE A.

5. Sent through a bootstrap data channel to the remote UE B. This may happen in parallel with and rather independent of step 4.

6. Any additional data channels created and used by the data channel application itself are established (logically) between UE A and UE B. Data transmission on data channels shall not start until there is confirmation that both peers have instantiated the data channel, using the same procedures as described for WebRTC in section 6.5 of [172]. The traffic may effectively go through the Data Channel Server, e.g., when the bootstrap and end-to-end data channels have the same anchoring point. This traffic may pass across an inter-operator border if UE A and UE B belong to different operators’ networks.

The bootstrap data channel is not intended for use directly between DCMTSI clients in terminal. DCMTSI clients in terminal that receive HTTP requests on a bootstrap data channel shall ignore such request and shall update the session by removing the SDP "a=dcmap" line with the stream ID where such HTTP request was received, and closing that stream ID.

The data channel application sent in a bootstrap data channel may be updated at any time, automatically or interactively, using normal HTTP procedures.

A bootstrap data channel shall be configured as ordered, reliable, with normal SCTP multiplexing priority. The sub-protocol for a bootstrap data channel shall be HTTP (not encapsulating HTTP in TCP), represented by the following, example SDP "a=dcmap" line, which therefore shall be present in each data channel media description in an SDP offer from a DCMTSI client in terminal:

a=dcmap:0 subprotocol="http"

When the HTTP subprotocol is used, any other data channels used by the data channel application JavaScript(s) sent in the bootstrap data channel shall be represented in an updated SDP as additional "a=dcmap" lines with stream ID values starting from 1000, using stream ID numbers from the JavaScript(s).

There are multiple, possible providers of data channel applications. In Figure 6.2.10.1-1, assume that UE A is local to the operator hosting the data channel server. Further assume that UE B belongs to a different operator (remote). The user of UE A can create and use data channel applications (steps 1-4), which can also be sent to UE B (step 5). Similarly, some other authorized part associated with UE A’s operator can create data channel applications for use by UE A (steps 1-4), which can also be sent to UE B (step 5). For simplicity, there’s no data channel server and data channel application repository depicted for UE B in Figure 6.2.10.1-1, but those could be present in a more general case. Seen from the perspective of a single UE, there are then at least four possible data channel application providers:

1. The local UE user.

2. Other authorized parties associated with the local network (e.g. the local operator).

3. The remote UE user.

4. Other authorized parties associated with the remote network (e.g. the remote operator).

The HTML web content making up a data channel application in each bootstrap data channel represents a different context of user interaction and should open in a separate tab, or some corresponding user interface construct, but the details are out of scope for this specification and left open for individual implementations. It shall be possible to use and navigate between different data channel applications from different bootstrap data channels with different stream IDs that are open simultaneously.

Table 6.2.10.1-2 describes a mandatory mapping between stream ID and bootstrap channel data channel application content sources, as seen from a single (local) DCMTSI client in terminal, each of which shall be listed as separate "a=dcmap" lines with "http" subprotocol in SDP when the DCMTSI client in terminal supports receiving data channel application content from that source.

Table 6.2.10.1-2: Bootstrap Data Channel Content Sources

|  |  |
| --- | --- |
| **Stream ID** | **Content Source** |
| 0 | Local network provider |
| 10 | Local user |
| 100 | Remote network provider |
| 110 | Remote user |

NOTE 4: When the local user has defined and stored multiple, different data channel applications in the local data channel application repository, the local network provider may provide functionality in the stream ID 0 data channel application that enables a dynamic choice of which user-defined data channel application to use with stream ID 10 in the DCMTSI call.

Figure 6.2.10.1-3, referring to Figure 6.2.10.1-1 and Table 6.2.10.1-2, is depicting the stream IDs used for distribution of a data channel application owned by UE A from its local data channel repository to both UE A (stream ID 10) and its remote UE B (stream ID 110).



Figure 6.2.10.1-3: Distribution of local data channel application to both UE

================= End of the 1st change ================

## ================ Start of the 2nd change ================

================= End of the 2nd change ================

## Y.6.9 Scene Description-Based Overlays

### Y.6.9.1 General

ITT4RT clients that support the “Overlay” feature may support the scene description as defined in [183] for signaling the overlay configuration.

If scene description-based overlays are supported, the following subset of the MPEG-I scene description extensions and features shall be supported:

- The MPEG\_media extension: used to reference the media streams.

- The MPEG\_accessor\_timed and the MPEG\_buffer\_circular: used to bind timed media.

- The MPEG\_texture\_video: used to define video textures for the overlay and the 360 video.

- The scene description update mechanism as defined in clause 5.2.4 of [183].

If scene description-based overlays are used in an ITT4RT session with multiple participants, then the ITT4RT MRF shall be used for the session and shall own the scene description.

If scene description-based overlays are used, then the ITT4RT-TX client in the ITT4RT MRF shall:

- Create a sphere or cubemap mesh node (depending on the selected projection) in the scene description for each 360 video stream in the ITT4RT session. The source of the node's texture shall reference the ITT4RT media stream of the corresponding 360 video as signaled by the SDP.

- Create a rectangular or spherical mesh node in the scene description for each overlay stream in the ITT4RT session. The source of the node's texture shall reference the media stream of the corresponding overlay stream as signaled by the SDP.

- The location of the overlay shall be indicated by the transformation of the corresponding overlay node in the scene description.

NOTE: In a scene description-based overlay solution, the scene camera corresponds the viewer’s position and it tracks the user’s 3DoF movements. The camera’s projection determines the field of view of the user.

The URL format as specified in 23090-14 Annex C shall be used to reference media streams in the ITT4RT session.

For participants that support scene description, the overlay information and positioning that is provided as part of the scene description shall take precedence over any information provided as part of the 3gpp\_overlay attribute.

An ITT4RT-Tx client in terminal that offers overlays may select to signal the overlay either through the 3gpp\_overlay attribute or through a scene update that adds the overlay node. The scene update mechanism is described in [183]. In case the ITT4RT-Tx uses the 3gpp\_overlay attribute to describe its overlays, the ITT4RT-Tx client in the ITT4RT MRF shall generate the scene description or scene description update document that signals the presence and position of that overlay.

### Y.6.9.2 Offer/Answer Negotiation

An ITT4RT-Tx client that desires to use scene description-based overlays, shall offer a data channel with a data channel indicating the “mpeg-sd” sub-protocol. The ITT4RT-Rx client in the MRF that supports scene-based overlays may answer by accepting the scene description data channel.

If the offer is accepted, the ITT4RT MRF shall generate and send the scene description to the offerer upon establishment of the data channel.

If the ITT4RT MRF receives an offer that does not contain a data channel with the “mpeg-sd” sub-protocol, it shall assume that the offering ITT4RT client does not support scene description-based overlays. In such case, the answering ITT4RT MRF shall not add a data channel with the “mpeg-sd” sub-protocol and may describe any overlays using the 3gpp\_overlay attribute.

### Y.6.9.3 SDP Signaling

An ITT4RT-Tx in the ITT4RT MRF that supports scene description-based overlays, shall support MTSI data channel media and act as a DCMTSI client. The stream id of the data channel with the sub-protocol “mpeg-sd” shall be in the range allocated for bootstrap data channels, i.e. below 1000, excluding values in Table 6.2.10.1-2. A single data channel with sub-protocol “mpeg-sd” shall be present in the offer/answer SDP. If multiple data channels with the “mpeg-sd” sub-protocol are detected, the one with the lowest stream ID shall be used. The scene description data channel shall be configured as ordered, reliable, with normal SCTP multiplexing priority.

When scene description-based overlays are offered, the ITT4RT-Tx in the ITT4RT MRF shall offer a data channel with a stream id that indicates the “mpeg-sd” subprotocol in the dcmap attribute. The “mpeg-sd” messages shall be JSON formatted in UTF-8 coding without BOM.

Scene description-based overlay descriptions, including complete scene descriptions and scene updates, shall be delivered through the same data channel.

## ================= End of the changes ================