**3GPP TSG- Meeting # *R02***

**, , -**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.3* | | | | | | | | |
| **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:*** | Clarification of “req-app-id" parameter | | | | | | | | | |
|  |  | | | | | | | | | |
| ***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-17 (Release 17) Rel-18 (Release 18) Rel-19 (Release 19)  Rel-20 (Release 20)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | 3GPP SA6 has defined in TS 23.392, “The APPID is a globally unique value that identifies a Data Channel Application. It is assigned by the MMTel Enabler Server. The value of this IE is also used to set the "req-app-id" parameter in "a=3gpp-req-app" media-level SDP attribute as specified in 3GPP TS 26.114 [4] when the application requests application data channel(s).”  This means that the “req-app-id” specified in TS 26.114 may no longer be sufficient to address the defination in TS 23.392, same APP ID may not be sufficient to guarantee full interoperability. In particular, this holds with the standalone DC introduced by SA2 in Rel-19. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Adding further clarifications in 6.2.10 and 6.2.13 in “req-app-id” related parts. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Risk for misinterpretation that could cause interoperability problems. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.2.10, 6.2.13 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | |  | | | | | | | | |

\* \* \* 1st Change \* \* \* \*

### 6.2.10 Data channel

#### 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 IETF RFC 5688 [177], with a value of "webrtc-datachannel" (the application media format used by IETF RFC 8864 [172]), regardless of data channel media being part of the SDP or not.

To indicate support for bootstrap data channel multiplexing procedures and application data channel multiplexing procedures as described by clauses 6.2.10.2 and 6.2.10.3, a DCMTSI client shall when including media feature tags as specified in TS 24.229 [7] include a g.3gpp.dc-mux media feature tag (see Annex AB for IANA registration information), regardless of data channel media being part of the SDP or not.

One or more data channel SDP media descriptions formatted according to IETF RFC 8864 [172] may be added to the SDP, alongside other SDP media descriptions such as e.g. speech, video, and text. The protocol identifier (proto value) and media format (fmt value) of a data channel SDP media description shall be set to "UDP/DTLS/SCTP" defined in IETF RFC 8864 [172] and "webrtc-datachannel" defined in IETF RFC 8841 [194], respectively.

A data channel SDP media description shall not be placed before the first SDP speech media description. SDP examples are provided in clause 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 bandwidth 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 JavaScript(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.

NOTE 3: Data channel stream IDs below 1000 may use a well-defined subprotocol for other features than retrieving data channel application(s). For example, the “mpeg-sd” subprotocol can be used for a data channel stream ID below 1000 for scene description-based overlays as specified in clause Y.6.9.

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

NOTE 4: A Data Channel Server in this specification can be further decomposed into a number of functional entities including DC Signalling Function, Media Function (or MRF) and DC Application Server as specified in Annex AC of [167].

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 as a response of its request.

5. Sent through a bootstrap data channel to the remote UE B as a response of its request. 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 including its resources retrieved via a bootstrap data channel may be updated at any time, automatically or interactively, using normal HTTP procedures over the bootstrap data channel.

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"

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 5: 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.

NOTE 6: To help the SDP answerer's network to distinguish the two media descriptions (m= lines) containing bootstrap data channels with the same stream ID values transferred between two networks, the SDP offerer's network adds an "a=3gpp-bdc-used-by:sender" attribute in the media description of the bootstrap data channel(s) established between the originating UE and the terminating network, and optionally adds "a=3gpp-bdc-used-by:receiver" attribute in the media description of the bootstrap data channel(s) established between the originating network and the terminating UE, before it sends the SDP offer to the remote network. Even when there is only one bootstrap data channel contained in the media descriptions, the SDP offerer's network also needs to add an "a=3gpp-bdc-used-by:sender" attribute in the media description of the bootstrap data channel established between the originating UE and the terminating network, or add an "a=3gpp-bdc-used-by:receiver" attribute in the media description of the bootstrap data channel established between the originating network and the terminating UE.

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

When the user in UE A in a call with UE B selects data channel application(s) for retrieval and use, and after the new application(s) are launched, the application(s) may make use of additional data channel(s) (see step 6 of 6.2.10.1-1). In this case, UE A initiates a call upgrade to add new data channel(s) to the call for the new application(s). The SDP offer the UE A generates shall include an "a=3gpp-req-app" attribute with a "req-app-id" parameter, as defined by clause 6.2.13, to identify the requesting application as part of the media description creating application data channels for that application. The application should be configured with that identification and the network deployment should ensure that identification to be sufficiently, potentially globally, unique to avoid ambiguity. The "a=3gpp-req-app" attribute may also include an "app-dc-info" parameter to allow the application to identify a different end point when creating multiple application data channels used for communication to a network server or to the remote UE.

The combination of "req-app-id" and "app-dc-info" parameters allows the communicating UEs to bind the SDP offers and answers for each data channel and stream IDs being negotiated for the respective applications using these data channel stream IDs.

#### 6.2.10.2 Generating SDP offer

A DCMTSI client in terminal may include a data channel media description for the bootstrap data channels in the initial SDP offer, as described above and according to [172] [184]. A DCMTSI client in terminal may add or disable (by setting port 0, as for RTP media) additional data channel media descriptions as needed in subsequent SDP offers.

A DCMTSI client in terminal that desires to use application data channels for a data channel application retrieved from any of its bootstrap data channels, shall initiate a subsequent SDP offer after the initial SDP offer, opening those application data channels by adding or updating a data channel media description describing application data channels for the retrieved data channel application, unless it received (and potentially already answered to) an SDP offer opening those application data channels. The added or updated data channel media description shall include corresponding "a=dcmap", "a=3gpp-req-app", and (optionally) "a=dcsa" attributes.

The "a=3gpp-req-app" attribute may also include an "adc-stream-id-endpoint" parameter as part of the "app-dc-info" parameter to differentiate what the SDP offerer intends the remote, answering "endpoint" to be. It is the application responsibility to know which data flows is to use which data channels created for the application, as appropriate for the remote endpoint-type.

NOTE: The combination of the "req-app-id" and "adc-stream-id-endpoint" parameters can be used to learn which server and what network is meant by the "Server" endpoint-type value in "adc-stream-id-endpoint" parameter.

The retrieved applications are to be configured with an appropriate value for the "a=3gpp-req-app" attribute. The offering DCMTSI client uses the value in this attribute to bind the media lines in the SDP describing application data channels to the corresponding application. The application also assigns the optional "app-dc-info" parameter values and uses them to differentiate the data channels to use for communication to the respective endpoints.

A DCMTSI client that supports bootstrap data channel multiplexing may include bootstrap data channels for both local and remote bootstrap data channel content sources (see Table 6.2.10.1-2) in a single SDP m= line media description in the SDP offer.

A DCMTSI client that supports application data channel multiplexing may include multiple "a=3gpp-req-app" attributes for different data channel applications with compatible QoS requirements in a single SDP m= line media description in the SDP offer.

The "a=3gpp-req-app" line shall not be included in a media description describing bootstrap data channels.

A data channel media description with specific loss or latency requirements should use "a=3gpp-qos-hint" in the SDP offer, as detailed in section 6.2.7.4. If subsequent SDP offers or answers adds data channels with more strict loss or latency requirements that cannot be met by keeping current "a=3gpp-qos-hint" and providing suitable SCTP "a=dcmap" parameters, the existing "a=3gpp-qos-hint" should be modified accordingly. Similarly, if subsequent SDP offers or answers closes (removes) data channels that are known to be the limiting factor for choosing the existing "a=3gpp-qos-hint", a more relaxed "a=3gpp-qos-hint" should be chosen to better fit the remaining data channels.

A DCMTSI client that intends to suspend the data channel media stream while other media streams are put on hold shall offer an updated SDP and the corresponding data channel media description shall include the SDP direction attribute "a=inactive". To resume a data channel when other media streams are resumed from hold, it shall offer an updated SDP and the corresponding data channel media description shall change the SDP direction attribute to "a=sendrecv" (or, equivalently, omit the SDP direction attribute).

#### 6.2.10.3 Generating SDP answer

An answering DCMTSI client in terminal may accept an SDP offer with data channel as described by [172] [184].

A DCMTSI client that received an SDP offer including application data channel media descriptions, uses the "req-app-id" parameter with the "a=3gpp-req-app" attribute to identify the application for which the data channels are added/updated, and formulates a corresponding SDP answer (especially the SCTP/DTLS transport parameters).

An answering DCMTSI client that desires to accept an offer for the application data channel media description shall include the same values for the "a=3gpp-req-app" from the offer. The application on the answering DCMTSI client should already be configured with the same identification as is present in the "req-app-id" parameter value for the "a=3gpp-req-app". The answering DCMTSI client can use the received "adc-stream-id-endpoint" parameter variant of the "app-dc-info" parameter to know which application data channels to use for the media directed to the respective end points.

An answering DCMTSI client that supports application data channel multiplexing and that desires to accept such offer shall do so by keeping all accepted "a=3gpp-req-app" attributes for different data channel applications in the SDP answer.

An answering DCMTSI client that does not support application data channel multiplexing shall reject the SDP m= line media description containing multiple "a=3gpp-req-app" attributes for different data channel applications, by setting port 0 on that m= line.

An answering DCMTSI client that does not support bootstrap data channel multiplexing shall reject the SDP m= line media description containing both local and remote bootstrap data channels, by setting port 0 on that m= line.

An answering DCMTSI client in terminal that desires to reject the entire SCTP association for all offered data channels shall set the port to 0 (zero) on the corresponding "m=application" line in SDP, as described in [172]. An SCTP association that initially, or as a result of session modification, has no open data channels ("a=dcmap" lines) should be rejected or closed by modifying the session, setting port number to 0 (zero).

An answering DCMTSI client in terminal that desires to accept some offered data channels and reject others shall indicate this by removing the non-desired data channel "a=dcmap" and "a=dcsa" lines from the SDP answer, as described in [172]. The DCMTSI client in terminal accepting a data channel must also accept the corresponding, supported bootstrap data channels with stream ID <1000 (e.g. a=dcmap:0 …).

An answering DCMTSI client that intends to accept an offer for the data channel media description including the SDP direction attribute as either "a=sendrecv" (or, equivalently, omitted) or "a=inactive" shall include the same attribute value for the corresponding data channel media description from the offer.

NOTE: The meaning of "a=sendonly" and "a=recvonly" SDP direction attributes are undefined for data channel media descriptions.

#### 6.2.10.4 Receiving SDP answer

An offering DCMTSI client in terminal receiving an SDP answer where the data channel SCTP association is accepted (port is not 0) may use any offered stream ID that has a corresponding "a=dcmap" line in the SDP answer, as described by section 6.5 in [172]. Data channels with "a=dcmap" lines in the SDP offer that are not included in the SDP answer must be considered as rejected and shall not be used, as described by section 6.5 in [172]. The "req-app-id" parameter of the "a=3gpp-req-app" attribute is used to identify the application for which the application data channels are added/updated.

An offering DCMTSI client in terminal that offered bootstrap and/or application data channel multiplexing and that receives an SDP answer where a data channel SDP media description that used data channel multiplexing in the corresponding offer is rejected (port 0), and where the g.3gpp.dc-mux media feature tag is not included as part of the SIP message carrying the answer, should re-attempt sending an SDP offer that is not using data channel multiplexing. Any m= lines that are modified due to removing multiplexing shall follow SDP session modification procedures [172][184]. Any additional m= lines containing the non-multiplexed "a=3gpp-req-app" lines in such re-attempted SDP offer shall be constructed as described in clauses 6.2.10.1 and 6.2.10.2.

\* \* \* 2nd Change \* \* \* \*

### 6.2.13 The a=3gpp-req-app SDP attribute

#### 6.2.13.1 General

When a DCMTSI client initiates the addition of application data channel(s) to a call with a peer DCMTSI client for different applications, the SDP offer/answer shall identify the applications and which data channels are to be created for them by signalling the identification for that application via the media level "a=3gpp-req-app" attribute added to the media lines describing application data channels for the application as discussed in section 6.2.10.1. The application identification can be included in the data channel application.

Both DCMTSI clients negotiating application data channels for a call between them shall identify the application requesting application data channel(s) to be established via the "req-app-id" parameter of an "a=3gpp-req-app" media-level SDP attribute that conveys a value set by the application launched on a UE, and configured on that application when it is made available for retreival via a bootstrap data channel. The "a=3gpp-req-app" attribute may also include an "adc-stream-id-endpoint" variant of the "app-dc-info" parameter to allow the UEs to identify the endpoints for the application data channels used for communication to a network server or to the remote UE. The combination of "req-app-id" and "app-dc-info" parameters allows the communicating UEs to bind the offers and answers for each data channel being negotiated for the identified application.

#### 6.2.13.2 3gpp-req-app ABNF syntax and semantics

3gpp-req-app-value = req-app-id [";" plmn-id] [";" app-dc-info] \*(";" 3gpp-req-app-opt)

req-app-id = quoted-string

plmn-id = mcc "/" mncmcc = 3(DIGIT)

mnc = 2\*3(DIGIT)

app-dc-info = adc-info \*[ SP adc-info]

adc-info = adc-stream-id / adc-stream-id-endpoint

adc-stream-id = 4\*5DIGIT

adc-stream-id-endpoint = adc-stream-id "-" endpoint-type

endpoint-type = "UE" / "Server"

3gpp-req-app-opt = token

quoted-string = DQUOTE \*(quoted-char / escaped-char) DQUOTE

quoted-char = SP / quoted-visible

quoted-visible = %x21 / %x23-24 / %x26-7E ; VCHAR without " or %

escaped-char = "%" HEXDIG HEXDIG

DQUOTE = <from RFC 5234>

token = <from RFC 4566>

By using the "req-app-id" and the “plmn-id" (as defined in sections 12.1 and 2.2 in [80]) parameters together, the application is identified globally unique.

The "app-dc-info" parameter indicates the application data channels used by the application identified by "req-app-id" parameter and the type of the remote endpoint of every application data channel. By including "adc-stream-id-endpoint" parameter, the SDP sender indicates to the network if the corresponding data channel stream is intended to be established towards a server or the remote UE. The application is responsible for distinguishing which data channel stream is towards a server or remote UE.

\* \* \* End of Changes \* \* \* \*