**3GPP TSG-SA WG6 Meeting #42-bis-e S6-210xxx**

**e-meeting, 12th – 20th April 2021 (revision of S6-210781, S6-210601, S6-210406)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **23.282** | **CR** | **0262** | **rev** | **3** | **Current version:** | **17.6.0** |  |
|  | | | | | | | | |
| *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 | **X** | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Corrections and enhancements to the Notification procedures | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | at&t | | | | | | | | | |
| ***Source to TSG:*** | S6 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | eMCData3 | | | | |  | ***Date:*** | | | 2021-04-06 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | B |  | | | | | ***Release:*** | | | R-17 |
|  | *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-15 (Release 15) Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | Fix some errors in the Notification procedure and add a couple of operations that are missing. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | 1. A notifiaction channel is used not just in a PULL method notification service but is also used in some of the PUSH method notification serivices (such as WebSockets for starting the PUSH notification service), thus the incorrect text is removed from 6.4.3.6 and modified in 7.13.3.17.3 procedure. 2. Correct some errors in the exisitng information flow tables with additional clarifications. 3. Add new information flow tables to support new update and delete of notification clannel and subscription procedures. 4. Clarify the procedure in 7.13.3.17.3 to align with first change that the notification channel is not only used by the PULL method of notification service. 5. Add two new procedures: Update and Delete notification channel and subscription. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | There will be errors and no guidance for stage 3 development. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.4.3.6, 7.13.3.1.32. 7.13.3.1.33, 7.13.3.1.34, 7.13.3.1.AA (new), 7.13.3.1.BB (new), 7.13.3.1.CC (new), 7.13.3.1.DD (new), 7.13.3.1.EE (new), 7.13.3.1.FF (new), 7.13.3.1.GG (new), 7.13.3.1.HH (new), 7.13.3.17.3 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | |  | | | | | | | | |

\* \* \* First Change \* \* \* \*

#### 6.4.3.6 Message notification client

The Message notification client is used to request the notification service from the MCData notification server. Once the notification service request is authorized by the MCData notification server, the Message notification client will communicate the callback endpoint, received from the MCData notification server, to the MCData message store to be used for notification message delivery.

\* \* \* Next Change \* \* \* \*

##### 7.13.3.1.32 Create notification channel request

Table 7.13.3.1.32-1 describes the information flow for the create notification channel request sent from the message notification client to the MCData notification server.

Table 7.13.3.1.32-1: Create notification channel request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Validity duration | O | How long the notification channel is intended to be used (see NOTE) |
| Channel Type | M | Indicates if PULL (e.g. long-polling method) or PUSH method will be used to deliver notification messages |
| NOTE: If this element not present, a default validity duration shall be provided by the server in response | | |

##### 7.13.3.1.33 Create notification channel response

Table 7.13.3.1.33-1 describes the information flow for the create notification channel response sent from the MCData notification server to the message notification client.

Table 7.13.3.1.33-1: Create notification channel response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Validity duration | M | How long the notification channel will last (i.e. channel lifetime) as granted by the MCData notification server |
| Notification URL | M | The URL to receive the notification message if a Pull method is requested. For some PUSH method implementation (such as WebSockets) this URL is used to start the PUSH notification service from the MCData notification server |
| Callback URL | M | The URL used by the Message notification client to subscribe to MCData message store notifications |

##### 7.13.3.1.34 Open notification channel

Table 7.13.3.1.34-1 describes the information flow for the open notification channel sent from the message notification client to the MCData notification server.

Table 7.13.3.1.34-1: Open notification channel request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Notification URL | M | The URL to receive the notification message |

##### 7.13.3.1.35 Subscribe for notification request

Table 7.13.3.1.35-1 describes the information flow for the subscribe for notification request sent from the message store client to the MCData message store.

Table 7.13.3.1.35-1: Subscribe for notification request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Callback URL | M | The URL where to send the notification message |
| Validity duration | M | How long the subscription to notification will last (i.e. subscription lifetime); this value shall be the returned value in the create notification channel response |

##### 7.13.3.1.36 Subscribe for notification response

Table 7.13.3.1.36-1 describes the information flow for the subscribe for notification response sent from the MCData message store to the message store client.

Table 7.13.3.1.36-1: Subscribe for notification response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Validity duration | M | How long the subscription of notification will last (i.e. subscription lifetime) as granted by the server |
| Result | M | Indicates if the subscription is success or failure |

\* \* \* Next Change \* \* \* \*

##### 7.13.3.1.AA         Update notification channel request

Table 7.13.3.1.AA describes the information flow for the update notification channel request sent from the message notification client to the MCData notification server.

Table 7.13.3.1.AA: Update notification channel request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Validity duration | M | How long the notification channel will last (i.e. channel lifetime) as requested by the Message notification client. |

##### 7.13.3.1.BB         Update notification channel response

Table 7.13.3.1.BB describes the information flow for the update notification channel response sent from the MCData notification server to the message notification client.

Table 7.13.3.1.BB: Update notification channel response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Validity duration | M | How long the notification channel will last (i.e. channel lifetime) as granted by the MCData notification server |
| Result | M | Indicates if the update is success or failure |

##### 7.13.3.1.CC         Update notification subscription request

Table 7.13.3.1.CC describes the information flow for the update notification subscription request sent from the message store client to the MCData message store.

Table 7.13.3.1.CC: Update notification subscription request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Validity duration | M | How long the notification channel will last (i.e. notification subscription lifetime). This value should be the returned value in the update notification channel response |

##### 7.13.3.1.DD         Update notification subscription response

Table 7.13.3.1.DD describes the information flow for the update notification subscription response sent from the MCData message store to the message store client.

Table 7.13.3.1.DD: Update notification subscription response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Validity duration | M | How long the notification channel will last (i.e. notification subscription lifetime) as granted by the MCData message store |
| Result | M | Indicates if the update is success or failure |

##### 7.13.3.1.EE         Delete notification channel request

Table 7.13.3.1.EE describes the information flow for the delete notification channel request sent from the message notification client to the MCData notification server.

Table 7.13.3.1.EE: Delete notification channel request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |

##### 7.13.3.1.FF         Delete notification channel response

Table 7.13.3.1.FF describes the information flow for the delete notification channel response sent from the MCData notification server to the message notification client.

Table 7.13.3.1.FF: Delete notification channel response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Result | M | Indicates if deletion of notification channel is success or failure |

##### 7.13.3.1.GG         Delete notification subscription request

Table 7.13.3.1.GG describes the information flow for the delete notification subscription request sent from the message store client to the MCData message store.

Table 7.13.3.1.GG: Delete notification subscription request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |

##### 7.13.3.1.HH         Delete notification subscription response

Table 7.13.3.1.HH describes the information flow for the delete notification subscription response sent from the MCData message store to the message store client.

Table 7.13.3.1.HH: Delete notification subscription response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| MCData ID | M | The identity of the MCData client initiating the request |
| Result | M | Indicates if deletion of notification subscription is success or failure |

\* \* \* Next Change \* \* \* \*

##### 7.13.3.17.3 Procedure using MCData notification server

The procedure in figure 7.13.3.17.3-1 describes how the MCData message store notifies the message notification client, using a MCData notification server, that there are new objects in the MCData message store needing to be synchronized. This procedure uses a web base notification mechanism in wide deployment today. The Message notification client requests the notification service from the MCData notification server and the MCData notification server returns with two URLs; one used by the service client to inform the service server where to send notification messages and the other one to use by the service client to PULL notification messages from the MCData notification server.

Pre-conditions:

1. The MCData user has an account with the MCData message store.

2. A successful authentication and authorization have been performed between the message store client and the MCData message store.

3. The Message store client doesn't have an ongoing session with the MCData message store.

4. The trust relationship between the MCData notification server and the MCData message store has been established.

5. The MCData notification server has a trust relationship and connection with the PUSH Enabler server.



Figure 7.13.3.17.3-1: Notify client to synchronize through MCData notification server

1. The Message notification client wants to create notification channels (i.e. endpoint URLs) to be used by the MCData message store to send notification messages and sends a Create notification channel request to the MCData notification server. The desired validity duration for the channels to be used and the notification channel type (PUSH or PULL) are included in the request.

2. The MCData notification server authenticates the Message notification client and authorizes its request.

3. The MCData notification server sends the Message notification client the Create notification channel response with the endpoint URLs that will be used by the MCData message store to send the notification messages and the Message notification client to receive the notification messages. The MCData notification server also includes what is the valid duration for these endpoint URLs to be used in the response.

4. If the notification type is PULL method, the message notification client sends the Open notification channel to the MCData notification server to start receiving the notification message. For certain PUSH method notification type (such as WebSockets) the message notification client requests the MCData notification server to start the PUSH notification service with its specific protocol that is outside the scope of this specification.

5. The message store client sends the Subscribe for notification request to the MCData message store asking to be notified if there are changes to its message store account. The callback URL returned from the MCData notification server in step 3 is included in the request for the MCData message store to use to send notification messages.

6. The MCData message store sends the Subscribe for notification response to the message store client to acknowledge the request.

7. The MCData user's message store account has changed and the MCData message store generates a notification message.

8. Using the callback URL, the MCData message store sends the notification message to the MCData notification server.

9. If the delivery method is PULL, the MCData notification server sends the notification message to the message notification client over the opened notification channel. If the delivery method is PUSH, the MCData notification server sends the notification message to the PUSH Enabler server (not shown in the figure) to deliver to the message notification client.

NOTE: The PUSH Enabler server is implementation specific and outside the scope of this specification.

The procedure in figure 7.13.3.17.3-2 describes how the message notification client updates the validity duration of a notification channel and subscription to avoid its expiration, i.e. to extend its lifetime.

Pre-conditions:

1. A notification channel has already been requested and established between the message notification client and MCData notification server.

2. The message store client has a successful notification subscription with the MCData message store.

3. The validity duration of the notification channel is about to expire.



Figure 7.13.3.17.3-2: Update a notification channel

1. The message notification client sends the Update notification channel request, including the desired new validity duration, to the MCData notification server.
2. The MCData notification server grants the request and sends the Update notification channel response to the message notification client. The new validity duration is included in the response.
3. The message store client sends the Update notification subscription request to the MCData message store with the new validity duration received from the MCData notification server in step 2.
4. The MCData message store sends the Update notification subscription response to the message store client and confirms the new validity duration.

The procedure in figure 7.13.3.17.3-3 describes how the message notification client delete a notification channel and subscription that is no longer needed.

Pre-conditions:

1. A notification channel has already been requested and established between the message notification client and MCData notification server.
2. The message store client has a successful notification subscription with the MCData message store.
3. The MCData user no longer wants to receive notifications from the MCData message store.

Figure 7.13.3.17.3-3: Delete a notification channel

1. The message store client decides to stop receiving notifications from the MCData message store and sends the Delete notification subscription request to the MCData message store.
2. The MCData message store acknowledges the request and sends the Delete notification subscription response to the message store client.
3. The message notification client sends the Delete notification channel request to the MCData notification server.
4. The MCData notification server acknowledges the request and sends the Delete notification channel response to the message notification client.