**3GPP TSG-CT WG1 Meeting #133-bis-eC1-220041**

**E-meeting, 17-21 January 2022**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **24.282** | **CR** | **0279** | **rev** | **01** | **Current version:** | **17.5.0** |  |
|  | | | | | | | | |
| *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:*** | Editorial clean ups | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | AT&T | | | | | | | | | |
| ***Source to TSG:*** | C1 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | eMCData3 | | | | |  | ***Date:*** | | | 2022-01-10 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **D** |  | | | | | ***Release:*** | | | Rel-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:*** | | Editorial and clean up across different procedures within clause 21.2. in order to make the procedures within clause 21.2 consistently worded and read better. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Editorials such as:   * Adding missing words (e.g. “MCData” before message store). * Correcting Formatting issues (e.g. B1 to Normal format change), * Renumbering adjustments to certain steps of some procedures in order to make them better read and consistent with the way similar steps are sequenced/worded in other procedures within clause 21.2. * Removing repeated or unnecessary words | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | The spec may be viewed as suboptimal. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 21.2 subclauses | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | | R01 incorporates further editorials as listed below:  Globally changing "with the following clarification(s):" to "as follows:"  21.2.3.2: "and" between 2)a) and 2)b)  21.2.4.2: same comment as in 21.2.3.2. In addition to formatting correction  21.2.5A.1: "and" after 1)c)  21.2.12A.1: removing "and" after 1)a)  21.2.13A.1: "and" after 1)a) | | | | | | | | |

\* \* \* \* \* \* \* FIRST CHANGE \* \* \* \* \* \* \*

## 21.2 MCData message store functions and client procedures

### 21.2.1 Object retrieval procedure

#### 21.2.1.1 Message store client procedures

To retrieve the object from MCData message store, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP GET request as specified in clause 6.2.3 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP GET request towards the message store function.

Upon receipt of an HTTP response, the message store client shall follow the procedure as described in clause 6.2.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.1.2 Message store function procedures

Upon receipt of the HTTP GET request from the client, as per clause 21.2.1.1, with the Request-URI identifying a resource in the MCData message store, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP GET request by following the procedures described in clause 6.2.3 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation (e.g. if the object identified by the Request URI was successfully found, it is returned in the HTTP response).

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.2 Object search procedure

#### 21.2.2.1 Message store client procedures

To search for information about a selected set of objects in the MCData message store, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.8 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.8.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP POST request, which includes a "SelectionCriteria" data structure, towards the message store function.

Upon receipt of an HTTP response, the message store client shall follow the procedure as describe in clause 6.8.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.2.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.2.1, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.8.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

3) shall generate and send an HTTP response, containing the objects matching the SelectionCriteria, towards the message store client.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.3 Update object(s) procedure

#### 21.2.3.1 Message store client procedures

To update object(s) in the MCData message store, the message store client, acting as an HTTP client, shall either follow the procedure described in clause 6.3 or 6.4, for individual object update, or 6.11 for bulk update of objects, of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall either generate an HTTP PUT request as specified in clause 6.3.4, 6.4.4, for individual object update, or an HTTP POST request, as specified in clause 6.11.5, for bulk update of objects, of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66], as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP PUT request, for individual object update, or the HTTP POST request, for bulk update of objects, towards the message store function.

Upon receipt of an HTTP response, the message store client shall either follow the procedure as described in clause 6.3.2, 6.4.2 for individual object update response, or clause 6.11.2 for bulk update of objects response, of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.3.2 Message store function procedures

Upon receipt of the HTTP PUT or the HTTP POST request from the client, as per clause 21.2.3.1, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) if the received request is an HTTP PUT, shall process the HTTP PUT request for individual object update by following the procedure described in clauses 6.3.2 or 6.4.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

b) if the received request is an HTTP POST, shall process the HTTP POST request by following the procedure described in clause 6.11.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] for bulk update of objects; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the update operation.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.4 Delete stored object(s) procedure

#### 21.2.4.1 Message store client procedures

To delete object(s) in the MCData message store, the message store client, acting as an HTTP client, shall either follow the procedure described in clause 6.2, for individual object delete, or clause 6.12 for bulk delete of objects, of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall either generate an HTTP DELETE request as specified in clause 6.2.6, for individual object delete, or an HTTP POST request as specified in clause 6.12.6, for bulk delete of objects, of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66], as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP DELETE request, for individual object delete, or the HTTP POST request, for bulk delete of objects, towards the message store function.

Upon receipt of an HTTP response, the message store client shall either follow the procedure as described in clause 6.2.2, for individual object delete response, or clause 6.12.2, for bulk delete of objects response, of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.4.2 Message store function procedures

Upon receipt of the HTTP DELETE or the HTTP POST request from the client, as per clause 21.2.4.1, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) if the received request is an HTTP DELETE, shall process the HTTP DELETE request for individual object delete by following the procedure described in clause 6.2.6 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

b) if the received request is an HTTP POST, shall process the HTTP POST request by following the procedure specified in clause 6.12.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] for bulk delete of objects; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the delete procedure.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.5 Void

### 21.2.5A Deposit an object procedure

#### 21.2.5A.1 MCData server procedures

To deposit an object of an MCData user in the MCData message store, the MCData server acting as an HTTP client shall follow the procedure described in clause 6.1 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.1.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall set the boxId of the resource URL as specified in clause 6.1.1 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] to MCData ID which is the identity of the MCData user;

c) shall include a valid MCData access token in the HTTP Authorization header; and

d) may include the query parameter "retrieveFile" in the Request URI with its value set to:

i) "No" if the MCData store is not required to retrieve the file from MCData content server; or

ii) "Yes" if the MCData store is required to retrieve the file from MCData content server and to store it locally in the MCData message store; and

NOTE: Including the retrieveFile query parameter with the value "Yes" is the same as if the retrieveFile query parameter is absent.

2) shall send the HTTP POST request towards the message store function.

Upon receipt of an HTTP response, the MCData server shall follow the procedure described in clause 6.1.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.5A.2 Message store function procedures

Upon receipt of the HTTP POST request from MCData server, as per clause 21.2.5A.1, with a Request-URI identifying a resource on the MCData message store, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.1.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; with the following clarification:

i) if the query parameter "retrieveFile" is set to "Yes" or if it is absent from the request URI, the message store function shall retrieve the file pointed to by the object's payloadPart URL(carried within the HTTP POST request body), store the file in the user's message storage area and update the object's payloadPart URL accordingly; and

3) shall generate and send the HTTP response towards the MCData server indicating the result of the deposit an object operation as per clause 6.1.2 of the OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.6 Object and folder copy procedure

#### 21.2.6.1 Message store client procedures

To copy object(s) and/or folder(s) to a destination folder in the MCData message store, the message store client, acting as an HTTP client, shall follow the procedure described in clause 6.18 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.18.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP POST request identifying the target folder and the source objects(s) and/or folder(s) for copying operation towards the message store function.

Upon receipt of an HTTP response, the message store client should follow the procedure as described in clause 6.18.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.6.2 Message store function procedures

Upon receipt of the HTTP POST from the client, as per clause 21.2.6.1, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.18.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] and copy to the target folder the requested source object(s) and/or folders(s); and

3) shall generate and send a HTTP response towards the message store client indicating the result of the operation.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.7 Deleting a folder procedure

#### 21.2.7.1 Message store client procedures

To delete a folder in the MCData message store using the message store function, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.14 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP DELETE request as specified in clause 6.14.6 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP DELETE request identifying the folder to be deleted towards the message store function.

Upon receipt of an HTTP response, the message store client should follow the procedure as described in clause 6.14.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.7.2 Message store function procedures

Upon receipt of the HTTP DELETE request from the client, as per clause 21.2.7.1, with the Request-URI identifying the folder in the message store to be deleted, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP DELETE request by following the procedures described in clause 6.14.6 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.8 Create a folder procedure

#### 21.2.8.1 Message store client procedures

To create a folder in the MCData message store using the message store function, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.13 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.13.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send towards the message store function the HTTP POST request identifying the target folder where the new folder is to be created.

Upon receipt of a HTTP response, the message store client should follow the procedure as described in clause 6.13.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.8.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.8.1, identifying the new folder to be created, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.13.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] and create the requested folder; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.9 void

### 21.2.10 Moving object(s) and folder(s) procedure

#### 21.2.10.1 Message store client procedures

To move object(s) and/or folder(s) to a destination folder in the MCData message store, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.19 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.19.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP POST request, identifying source objects and/or folder(s) to be moved to the designated destination folder, towards the message store function.

Upon receipt of a HTTP response, the message store client shall follow the procedure as described in clause 6.19.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.10.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.10.1, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.19.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] and perform the move operation; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.11 Folder search procedure

#### 21.2.11.1 Message store client procedures

To search for information about a selected set of folder(s) in the MCData message store, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.16 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.16.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP POST request, which includes a "SelectionCriteria" data structure, towards the message store function.

Upon receipt of a HTTP response, the message store client should follow the procedure as described in clause 6.16.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.11.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.11.1, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.16.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

3) shall generate and send an HTTP response, containing the folders matching the SelectionCriteria, towards the message store client.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.12 Void

### 21.2.12A Create a subscription to notifications procedure

#### 21.2.12A.1 Message store client procedures

In order for the message store client to keep its local store in sync with the MCData message store, it needs to receive notifications about changes in the message store. For this purpose, the message store client would need to subscribe to notification from the message store. Synchronization using subscriptions and notifications is described in clause 5.1.5.1 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

To create a subscription to notifications about changes in the message store using the message store function, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.20 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.20.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function; and

b) shall include a valid MCData access token in the HTTP Authorization header; and

2) shall send the HTTP POST request towards the message store function.

Upon receipt of an HTTP response, the message store client should follow the procedure as described in clause 6.20.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.12A.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.12.1, with a Request-URI identifying a resource on the message store, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.20.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] and create the requested subscription; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation as per clause 6.20.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.13 Void

### 21.2.13A Delete a subscription to notifications procedure

#### 21.2.13A.1 Message store client procedures

To delete / cancel a subscription and stop corresponding notifications about changes in the MCData message store using the message store function, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.21 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP DELETE request as specified in clause 6.21.6 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function; and

b) shall include a valid MCData access token in the HTTP Authorization header; and

2) shall send the HTTP DELETE request identifying the subscription to be deleted towards the message store function.

Upon receipt of an HTTP response, the message store client should follow the procedure as described in clause 6.21.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.13A.2 Message store function procedures

Upon receipt of the HTTP DELETE request from the client, as per clause 21.2.13.1, with a Request-URI identifying the subscription resource on the message store, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP DELETE request by following the procedures described in clause 6.21.6 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] and delete the requested subscription; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation as per clause 6.21.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.14 Void

### 21.2.14A Update a subscription to notifications procedure

#### 21.2.14A.1 Message store client procedures

A client may update its subscription to notification in order to:

1) extend the life of the subscription;

2) restart the notification stream from where it left off.

Synchronization using subscriptions and notifications is described in clause 5.1.5.1 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]

To update a subscription to notifications about changes in the MCData message store using the message store function, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.21 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.21.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

2) shall send the HTTP POST request towards the message store function.

Upon receipt of an HTTP response, the message store client should follow the procedure described in clause 6.21.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.14A.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per bclause 21.2.14A.1, with a Request-URI identifying a subscription resource on the message store, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.21.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] and update the requested subscription; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation as per clause 6.21.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.15 Object(s) upload procedure

#### 21.2.15.1 Message store client procedures

To upload the object(s) to the MCData message store, the message store client acting as an HTTP client, shall either follow the procedure described in clause 6.1 for single upload or clause 6.10 for bulk upload of objects as specified in the OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in either clause 6.1.5 or 6.10.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] depending on a single object upload or bulk upload of objects as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) shall send the HTTP POST request towards the message store function.

Upon receipt of an HTTP response, the message store client shall follow the procedure as described in clause 6.1.2 for single upload or 6.10.2 for bulk upload as specified in the OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.15.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.15.1, with a Request-URI identifying a resource on the MCData message store, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in either clause 6.1.5 or 6.10.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] depending on a single object upload or bulk upload of objects; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the upload operation.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.16 Synchronization notifications procedure

#### 21.2.16.1 Message store function procedures

To send notifications about changes in the MCData message store using the message store function, the MCData message store, acting as an HTTP client shall follow the procedure described in clause 6.22 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP POST request as specified in clause 6.22.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field using the callback URL which was previously provided by the message store client in its corresponding subscription creation request as specified in clause 21.2.12A; and

b) shall send the HTTP POST request towards the callback URL provided by the client.

Upon receipt of an HTTP response, the message store function should follow the procedure as described in clause 6.22.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.16.2 Message store client procedures

If the callback URL in the HTTP POST request (clause 21.2.16.1) points to the message store client then upon receipt of the HTTP POST request from the MCData message store, as per clause 21.2.16.1, the message store client acting as an HTTP server (for such an in-band connection as described in clause 7.13.3.17.2 of 3GPP TS 23.282[2]):

1) shall process the HTTP POST request by following the procedures described in clause 6.22.5 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

a) either use the notification content and the reported "restartToken" and "index" as specified in clause 5.1.5.1 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] to have the client’s local message store updated accordingly; or

b) use the notification as a trigger to subsequently search the MCData message store for the list of changes as specified in clause 21.2.17; and

2) shall generate and send an HTTP response towards the message store function indicating the result of the operation as per clause 6.22.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

NOTE: The notifications about changes in the MCData message store can be used by the message store client to synchronize its local message store with the MCData message store in two distinguished ways which are listed in sub-bullets "a" and "b" above.

If however, the Message store client is not using an in-band connection with the MCData message store to receive notifications and has instead created a notification channel with the MCData notification server (see clause 7.13.3.17.3 of 3GPP TS 23.282[2]) as described in clause 21.2.19, then the message store client shall not follow the procedure in this clause and instead follow the procedure described in clause 21.2.22 "Open notification channel" in order to start receiving the notifications (about changes in the message store).

#### 21.2.16.3 MCData Notification server procedures

If the callback URL in the HTTP POST request, as described in clause 21.2.16.1, points to the MCData Notification server then upon receipt of the request from the MCData message store, the MCData notification server acting as an HTTP server as per clause 7.13.3.17.3 of 3GPP TS 23.282[2]:

1) shall process the HTTP POST request; and

2) shall make the notifications available to the message notification client (and hence the message store client) through the associated channel which was previously created and as need be opened (see clause 21.2.19 and clause 21.2.22).

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.17 Search-based synchronization procedure

#### 21.2.17.1 Message store client procedures

To search for changes (e.g. newly created objects, recently deleted objects, etc) in the MCData message store using the message store function, the message store client, acting as an HTTP client shall follow the procedure described in clause 21.2.2.1 with the following clarification:

1) shall use the search criterion of "CreatedObjects", "VanishedObjects" or "Flag" in the HTTP POST request as specified in clause 5.1.5.2 and 5.4.2.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] in order to retrieve from the MCData message store the list of the newly created object, recently deleted object and/or changes to flags respectively.

#### 21.2.17.2 Message store function procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.17.1, the message store function acting as an HTTP server shall follow the procedure described in clause 21.2.2.2 with the following clarification:

1) if search criterion in the HTTP POST request is set to "CreatedObjects", then the HTTP POST, response shall include a "creationCursor" as specified in clause 5.3.2.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.18 Retrieve content of a given folder procedure

#### 21.2.18.1 Message store client procedures

To retrieve the content of a given folder identified by its folder ID in the MCData message store using the message store function, the message store client, acting as an HTTP client shall follow the procedure described in clause 6.14 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

1) shall generate an HTTP GET request as specified in clause 6.14.3 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] as follows:

a) shall set the Host header field to a hostname identifying the message store function;

b) shall include a valid MCData access token in the HTTP Authorization header; and

c) may include URI query parameter(s) necessary to control the extent of the folder's information returned in the response; and

2) shall send the HTTP GET request towards the message store function.

NOTE: in order for the message store client to retrieve the content of the root folder, it first needs to discover its folder ID as described in clause 5.1.6 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] using Folder search procedure specified in clause 21.2.11 of the present document.

Upon receipt of an HTTP response, the message store client should follow the procedure as described in clause 6.14.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

#### 21.2.18.2 Message store function procedures

Upon receipt of the HTTP GET request from the client, as per clause 21.2.18.1, the message store function acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP GET request by following the procedures described in clause 6.14.3 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66]; and

3) shall generate and send an HTTP response towards the message store client indicating the result of the operation as per clause 6.14.2 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66].

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.19 Create notification channel procedure

#### 21.2.19.1 Message notification client procedures

To create a notification channel, the Message notification client, acting as an HTTP client shall follow the procedure described in clause 6.1 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] as follows:

1) shall generate an HTTP POST request as specified in clause 6.1.5 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] as follows:

a) shall set the Host header field to a hostname identifying the MCData Notification server;

b) shall include a valid MCData access token in the Authorization header; and

c) shall send the HTTP POST request towards the MCData Notification server.

Upon receipt of an HTTP response, the Message notification client should follow the procedure as described in clause 6.1.2 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76].

#### 21.2.19.2 MCData Notification server procedures

Upon receipt of the HTTP POST request from the client, as per clause 21.2.19.1, with the Request-URI identifying a resource in the MCData Notification server, the MCData Notification server acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.1.5 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76]; and

3) shall generate and send an HTTP response towards the Message notification client indicating the result of the operation.

NOTE 1: A successful HTTP response includes a Callback URL and can also include a Channel URL depending on the "channelType" requested (see clause 5 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76]).

NOTE 2: The Callback URL is used by the message store client in its request for creation of subscription to notifications sent towards the Message store function as described in clause 21.2.12A.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.20 Delete notification channel procedure

#### 21.2.20.1 Message notification client procedures

To delete a notification channel, the Message notification client, acting as an HTTP client shall follow the procedure described in clause 6.2 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] as follows:

1) shall generate an HTTP DELETE request as specified in clause 6.2.6 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] with following the clarifications:

a) shall set the Host header field to a hostname identifying the MCData Notification server;

b) shall include a valid MCData access token in the Authorization header; and

c) shall send the HTTP DELETE request towards the MCData Notification server.

Upon receipt of a HTTP response, the Message notification client should follow the procedure as described in clause 6.2.2 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76].

NOTE: When the notification channel is deleted, the Message store client normally removes the notification subscription in the MCData Message store function as described in clause 21.2.13A.

#### 21.2.20.2 MCData Notification server procedures

Upon receipt of the HTTP DELETE request from the client, as per clause 21.2.20.1, with the Request-URI identifying a resource in the MCData Notification server, the MCData Notification server acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP DELETE request by following the procedures described in clause 6.2.6 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76]; and

3) shall generate and send an HTTP response towards the Message notification client indicating the result of the operation.

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.21 Update notification channel procedure

#### 21.2.21.1 Message notification client procedures

To update a notification channel's lifetime, the Message notification client, acting as an HTTP client shall follow the procedure described in clause 6.4 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] as follows:

1) shall generate an HTTP PUT request as specified in clause 6.4.4 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] as follows:

a) shall set the Host header field to a hostname identifying the MCData Notification server;

b) shall include a valid MCData access token in the Authorization header; and

c) shall send the HTTP PUT request towards the MCData Notification server.

Upon receipt of an HTTP response, the Message notification client should follow the procedure as described in clause 6.4.2 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76].

NOTE: A successful HTTP response includes the new Channel's lifetime duration which can be used by the Message store client to update the lifetime of the notification subscription in the MCData message store function as described in clause 21.2.14A.

#### 21.2.21.2 MCData Notification server procedures

Upon receipt of the HTTP PUT request from the client, as per clause 21.2.21.1, with the Request-URI identifying a resource in the MCData Notification server, the MCData Notification server acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP PUT request by following the procedures described in clause 6.4.4 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76]; and

3) shall generate and send an HTTP response towards the Message notification client indicating the result of the operation;

\* \* \* \* \* \* \* NEXT CHANGE \* \* \* \* \* \* \*

### 21.2.22 Open notification channel procedure

#### 21.2.22.1 Message notification client procedures

Based on the channel type created as part of the notification channel creation procedure (see clause 21.2.19. "Create notification channel"), the Message notification client would determine if and how it needs to open (interact with) the created channel for notification flow (i.e. using PULL or PUSH).

To open the notification channel for a PULL notification delivery method (i.e. created channel is of type LongPolling), the Message notification client, acting as an HTTP client shall follow the procedure described in clauses 6.3 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] as follows:

1) shall generate an HTTP POST request as specified in clause 6.3.5 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] as follows:

a) shall set the Host header field to a hostname identifying the Notification server extracted from the channelURL received from the Notification server during channel creation (see clause 21.2.19. "Create notification channel");

b) shall include a valid MCData access token in the Authorization header; and

c) shall send the HTTP POST request towards the MCData Notification server using the channelURL received from the MCData Notification server during channel creation procedure (see clause 21.2.19. "Create notification channel").

Upon receipt of a HTTP response, the Message notification client should follow the procedure as described in clause 6.3.2 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76]; and

1) either use the notification content and the reported "restartToken" and "index" as specified in clause 5.1.5.1 of OMA-TS-REST\_NetAPI\_NMS-V1\_0-20190528-C [66] to have the client’s local message store updated accordingly; or

2) use the notification as a trigger to subsequently search the MCData message store for the list of changes as specified in clause 21.2.17;

NOTE: The notifications about changes in the MCData message store can be used by the message store client to synchronize its local message store with the MCData message store in two distinguished ways which are listed in bullets "1" and "2" above.

To open the notification channel for a PUSH notification delivery method over WebSocket (i.e. created channel is of type WebSocket), the Message notification client shall follow the procedure described in Appendix I of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] and use the channelURL received from the MCData Notification server during the channel creation procedure (see clauses 21.2.19 ) to create a WebSocket connection with the MCData Notification server. The process of creating a WebSokect connection between the Message notification client and the MCData Notification server through which the MCData Notification server can send notifications to the Message notification client is not RESTful.

If the created channel is of type NativeChannel, the Message notification client, is not required to invoke the "Open notification channel" procedure as defined in this clause. See clauses 5, 5.3.13, 5.3.14 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76] for description on receiving notification over a NativeChannel.

#### 21.2.22.2 MCData Notification server procedures

Upon receipt of the HTTP POST request (i.e. PULL notification delivery method) from the client, as per clause 21.2.22.1, with the Request-URI (i.e. channelURL) identifying a resource in the MCData Notification server, the MCData Notification server acting as an HTTP server:

1) shall validate the MCData access token (with "Bearer" authentication scheme) received in the Authorization header of the request as specified in 3GPP TS 24.482 [24];

2) if validation is successful then

a) shall process the HTTP POST request by following the procedures described in clause 6.3.5 of OMA-TS-REST\_NetAPI\_NotificationChannel-V1\_0-20200319-C [76]; and

3) shall generate and send an HTTP response towards the Message notification client indicating the result of the operation. If the response is successful, it shall contain the notifications (i.e. MCData message store change events).

\* \* \* \* \* \* \* END CHANGES \* \* \* \* \* \* \*