Doc# TST-2016-xxxx-Group-FanoutPoint-Tests

|  |
| --- |
| Input Contribution |
| Meeting ID\* | TST\_24 |
| Title:\* | TS-0018-Test Purposes contribution |
| Source:\* | Bob Flynn, InterDigital, Bob.Flynn@interdigital.com |
| Uploaded Date:\* | 2016-07-07 |
| Document(s)Impacted\* | TS-0018 |
| Intended purpose ofdocument:\* | [x]  Decision[x]  Discussion[ ]  Information[ ]  Other <specify>  |
| Decision requested or recommendation:\* | The contribution proposes some test purposes to be added to TS-0018 (Test Suite Structure and Test Purposes) |
| Template Version:23 February 2015 (Dot not modify) |

**oneM2M Notice**

The document to which this cover statement is attached is submitted to oneM2M. Participation in, or attendance at, any activity of oneM2M, constitutes acceptance of and agreement to be bound by terms of the Working Procedures and the Partnership Agreement, including the Intellectual Property Rights (IPR) Principles Governing oneM2M Work found in Annex 1 of the Partnership Agreement.

1. **Introduction**

This contribution document consist of test purposes for group fanoutPoint requests for create, retrieve, update and delete operations of the <fanoutPoint>.

FROM TS-0001

#### 10.2.7.6 *<fanOutPoint>* Management Procedures

Figure 10.2.7.6-1 illustrates how the *<fanOutPoint>* virtual resource works on the group Hosting CSE. The procedures in the figure apply to clauses 10.2.7.6 to 10.2.7.9.



**Figure 10.2.7.6-1: Group content management procedures**

#### 10.2.7.7 Create *<fanOutPoint>*

This procedure shall be used for creating the content of all members resources belonging to an existing <group> resource.

**Table 10.2.7.7-1: <fanOutPoint> CREATE**

|  |
| --- |
| *<fanOutPoint>* CREATE |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | ***From:*** Identifier of the AE or the CSE that initiates the Request***To:*** The address of the *<fanOutPoint>* virtual resource***Content:*** The representation of the resource the Originator intends to create***Group Request Identifier:*** The group request identifier |
| Processing at Originator before sending Request | The Originator shall request to create the resource that have the same content in allmembers resources belonging to an existing *<group>* resource by using a CREATEoperation. The Request may address the virtual child resource *<fanOutPoint>* of thespecific *<group>* resource of a group Hosting CSE. The request may also address theaddress that results from appending a relative address to the *<fanOutPoint>* address inorder to create the resources that have the same content under the correspondingchild resources represented by the relative address with respect to all membersresources. The Originator may be an AE or CSE |
| Processing at GroupHosting CSE | For the CREATE procedure, the Group Hosting CSE shall:* Check if the Originator has CREATE privilege in the *<accessControlPolicy>*

 resource referenced by the members *AccessControlPolicyIDs* in the *<group>* resource. In the case members *membersAccessControlPolicyIDs* is not provided the access control policy defined for the *<group>* resource shall be used* Upon successful validation, obtain the IDs of all members resources from the

 attribute *membersIDs* of the addressed *<group>* resource* Generate fan out requests addressing the obtained address (appended with

 the relative address if any) to the member hosting CSEs as indicated in figure 10.2.7.6-1.The ***From*** parameter in the request is set to ID of the Originator from the request from the original Originator* In the case that a member resource is a *<group>* resource and the request to

 be fanned out does not contain a group request identifier already, generate a unique group request identifier, include the group request identifier in all the requests to be fanned out and locally store the group request identifier* If the group Hosting CSE determines that multiple members resources belong

 to one CSE according to the IDs of the members resources, it may converge the requests accordingly before sending out. This may be accomplished by the group Hosting CSE creating a *<group>* resource on the members Hosting CSE to collect all the members on that members Hosting CSE* After receiving the responses from the members hosting CSEs, respond to

 the Originator with the aggregated results and the associated members list |
| Processing at MemberHosting CSE | For the CREATE procedure, the Member Hosting CSE shall:* Check if the request has a group request identifier. Check if the group request

identifier is contained in the requested identifiers stored locally. If match isfound, ignore the current request and respond an error. If no match is found,locally store the group request identifier* Check if the original Originator has the CREATE permission on the addressed

resource. Upon successful validation, perform the create procedures for thecorresponding type of addressed resource as described in other sub-clausesof clause 10.2* Send the corresponding response to the Group Hosting CSE
 |
| Information in Response message | Converged responses from members hosting CSEs |
| Processing at Originator after receiving Response | None |
| Exceptions | * Same request with identical group request identifier received
* Originator does not have the CREATE permission to access the

*<fanOutPoint>* resource |

#### 10.2.7.8 Retrieve *<fanOutPoint>*

This procedure shall be used for retrieving the content of all member resources belonging to an existing <group> resource.

 **Table 10.2.7.8-1: <fanOutPoint> RETRIEVE**

|  |
| --- |
| *<fanOutPoint>* RETRIEVE |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | ***From:*** Identifier of the AE or the CSE that initiates the Request***To:*** The address of the *<fanOutPoint>* virtual resource***Content:*** The representation of the resource the Originator intends to retrieve***Group Request Identifier:*** The group request identifier  |
| Processing at Originator before sending Request | The Originator shall request to obtain the resource or specific attributes of all memberresources belonging to an existing *<group>* resource by using a RETRIEVE operation.The request may address the virtual child resource *<fanOutPoint>* of the specific*<group>* resource of a group Hosting CSE. The request may also address the addressthat results from appending a relative address to the *<fanOutPoint>* address in order toretrieve the corresponding attributes or child resources represented by the relativeaddress with respect to all members resources. The Originator may be an AE or CSE |
| Processing at GroupHosting CSE | For the RETRIEVE procedure, the Group Hosting CSE shall:* Check if the Originator has RETRIEVE permission in the

 *<accessControlPolicy>* resource referenced by the *membersAccessControlPolicyIDs* in the addressed *<group>* resource. In the case *membersAccessControlPolicyIDs* is not provided, the access control policy defined for the group resource shall be used* Upon successful validation, obtain the IDs of all members resources from the

 *membersIDs* attribute of the addressed *<group>* resource* Generate fan out requests addressing the obtained address (appended with

 the relative address if any) to the members hosting CSEs as indicated in figure 10.2.7.6-1.The ***From*** parameter in the request is set to ID of the Originator from the request from the original Originator* In the case that a member resource is a *<group>* resource, generate a unique

 group request identifier and the request to be fanned out does not contain a group request identifier already, include the group request identifier in all the requests to be fanned out and locally store the group request identifier* If the group hosting CSE determines that multiple members resources belong

 to one CSE according to the IDs of the members resources, it may converge the requests accordingly before sending out. This may be accomplished by the group Hosting CSE creating a *<group>* resource on the members Hosting CSE to collect all the members on that members Hosting CSE* After receiving the responses from the members hosting CSEs, respond to

 the Originator with the aggregated results and the associated member list |
| Processing at MemberHosting CSE | For the RETRIEVE procedure, the Member Hosting CSE shall:* Check if the request has a group request identifier. Check if the group request

 identifier is contained in the requested identifier stored locally. If match is found, ignore the current request and respond an error. If no match is found, locally store the request identifier* Check if the original Originator has the RETRIEVE permission on the

 addressed resource. Upon successful validation, perform the retrieve procedures for the corresponding type of addressed resource as described in other sub-clauses of clause 10.2* Send the corresponding response to the group Hosting CSE
 |
| Information in Response message | Converged responses from members hosting CSEs |
| Processing at Originator after receiving Response | None |
| Exceptions | * Same request with identical group request identifier received
* Originator does not have RETRIEVE permission to access the *<fanOutPoint>*

 resource |

 **Table 10.2.7.9-1: <fanOutPoint> UPDATE**

|  |
| --- |
| *<fanOutPoint>* UPDATE |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | ***From:*** Identifier of the AE or the CSE that initiates the Request***To:*** The address of the *<group>* resource***Content:*** The representation of the resource the Originator intend to Update***Group Request Identifier:*** The group request identifier |
| Processing at Originator before sending Request | The Originator shall request to update all member resources belonging to an existing*<group>* resource with the same data by using a UPDATE operation. The request mayaddress the virtual child resource *<fanOutPoint>* of the specific *<group>* resource of agroup Hosting CSE. The request may also address the address that results fromappending a relative address to the *<fanOutPoint>* in order to update thecorresponding child resources represented by the relative address with respect to all*<members>* resources. The Originator may be an AE or CSE |
| Processing at GroupHosting CSE | For the UPDATE procedure, the Group Hosting CSE shall:* Check if the Originator has UPDATE permission in the *<accessControlPolicy>*

resource referenced by the *membersAccessControlPolicyIDs* in the groupresource. In the case members *membersAccessControlPolicyIDs* is notprovided the access control policy defined for the group resource shall beused* Upon successful validation, obtain the IDs of all member resources from the

attribute *membersIDs* of the addressed *<group>* resource* Generate fan out requests addressing the obtained address (appended with

the relative address if any) to the members hosting CSEs as indicated infigure 10.2.7.6-1.The ***From*** parameter in the request is set to ID of theOriginator from the request from the original Originator* In the case that a member resource is a *<group>* resource and the request to

be fanned out does not contain a group request identifier already, generate aunique group request identifier, include it in all the requests to be fanned outand locally store the group request identifier* If the group Hosting CSE determines that multiple members resources belong

to one CSE according to the IDs of the member resources, it may convergethe requests accordingly before sending out. This may be accomplished bythe group Hosting CSE creating a *<group>* resource on the member HostingCSE to collect all the members on that members Hosting CSE* After receiving the responses from the member hosting CSEs, respond to the Originator with the aggregated results and the associated members list
 |
| Processing at MemberHosting CSE | For the UPDATE procedure, the Member Hosting CSE shall:* Check if the request has a group request identifier. Check if the request

identifier is contained in the requested identifier stored locally. If match isfound, ignore the current request and respond an error. If no match is found,locally store the request identifier* Check if the original Originator has the UPDATE permission on the addressed

resource. Upon successful validation, perform the update procedures for thecorresponding type of addressed resource as described in other sub-clausesof clause 10.2* Send the corresponding response to the group Hosting CSE
 |
| Information in Response message | Converged responses from members hosting CSEs |
| Processing at Originator after receiving Response | None |
| Exceptions | * Same request with identical group request identifier received
* Originator does not have UPDATE permission to access the *<fanOutPoint>*

 resource |

 **Table 10.2.7.10-1: <fanOutPoint> DELETE**

|  |
| --- |
| *<fanOutPoint>* DELETE |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | ***From:*** Identifier of the AE or the CSE that initiates the Request***To:*** The address of the *<fanOutPoint>* virtual resource***Content:*** The representation of the resource the Originator intends to delete***Group Request Identifier:*** The group request identifier |
| Processing at Originator before sending Request | The Originator shall request to delete all members resources belonging to an existing*<gro*u*p>* resource by using a DELETE operation. The request may address the virtualchild resource *<fanOutPoint>* of the specific *<group>* resource of a group HostingCSE. The request may also address the address that results from appending a relativeaddress to the *<fanOutPoint>* in order to delete the corresponding child resourcesrepresented by the relative address with respect to all member resources. TheOriginator may be an AE or a CSE |
| Processing at GroupHosting CSE | For the DELETE procedure, the *<group>* Hosting CSE shall:* Check if the Originator has DELETE permission in the *<accessControlPolicy>*

resource referenced by the *membersAccessControlPoliciIDs* in the *<group>*resource. In the case *membersAccessControlPolicyIDs* is not provided theaccess control policy defined for the group resource shall be used* Upon successful validation, obtain the IDs of all member resources from the

attribute *membersIDs* of the addressed *<group>* resource* Generate fan out requests addressing the obtained address (appended with

the relative address if any) to the member hosting CSEs as indicated in figure10.2.7.6-1. ***From*** parameter in the request is set to ID of the Originator fromthe request from the original Originator* In the case that the members resources is a *<group>* resource and the

request to be fanned out does not contain a group request identifier already,generate a unique group request identifier, include the group requestidentifier in all the requests to be fanned out and locally store the grouprequest identifier* If the *<group>* Hosting CSE determines that multiple members resources

belong to one CSE according to the IDs of the members resources, it mayconverge the requests accordingly before sending out. This may beaccomplished by the group Hosting CSE creating a *<group>* resource on themember Hosting CSE to collect all the members on that member HostingCSE* After receiving the responses from the members hosting CSEs, respond to

the Originator with the aggregated results and the associated member list |
| Processing at MemberHosting CSE | For the DELETE procedure, the Members Hosting CSE shall:* Check if the request has a group request identifier. Check if the group request

identifier is contained in the requested identifier stored locally. If match isfound, ignore the current request and respond an error. If no match is found,locally store the group request identifier* Check if the original Originator has the DELETE permission on the addressed

resource. Upon successful validation, perform the delete procedures for thecorresponding type of addressed resource as described in other sub-clausesof clause 10.2* Send the corresponding response to the Group Hosting CSE
 |
| Information in Response message | Converged responses from members hosting CSEs |
| Processing at Originator after receiving Response | None |
| Exceptions | * Same request with identical group request identifier received
* Originator does not have DELETE permission to access the *<fanOutPoint>*

 resource |

 **Table 10.2.7.11-1: <fanOutPoint> Subscribe**

|  |
| --- |
|  *<fanOutPoint>* Subscribe |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | ***From:*** Identifier of the AE or CSE that initiates the request***To:*** The address of the <fanOutPoint> resource appended with the ID of the*<subscription>* resource to be created***Group Request Identifier:*** The group request identifier |
| Processing at Originator before sending Request | The Originator shall request to create a subscription resource under all memberresources belonging to an existing *<group>* resource by using a CREATE operation.The request may address the virtualchild resource *<fanOutPoint>* of the specific *<group>* resource of a group HostingCSE. The request may also address the address that results from appending a relativeaddress to the *<fanOutPoint>* in order to create the corresponding subscription to the resource represented by the relative address with respect to all member resources. The requestshall include *notificationForwardingURI* attribute if the Originator wants the groupHosting CSE to aggregate the notifications. The request shall include the requiredinformation and may include the optional information as described in subscriptionmanagement clause 10.2.11. The Originator may be an AE or a CSE |
| Processing at GroupHosting CSE | The *<group>* Hosting CSE shall:* Check if the Originator has CREATE privilege in the *<accessControlPolicy>*

resource referenced by the *membersAccessControlPolicyIDs* in the groupresource. In the case *membersAccessControlPolicyIDs* is not provided theaccess control policy defined for the group resource shall be used* If the subscription resource in the request contains an

*notificationForwardingURI* attribute, assign a URI to replace the*notificationURI* of the subscription resource which will be used to receivenotifications from member hosting CSEs. The ID of the *<group>* resourceshall be set to the *groupID* attribute of the *<subscription>* resource. The groupHosting CSE shall maintain the mapping of the generated *notificationURI* andthe former *notificationURI** Upon successful validation, obtain the IDs of all member resources from the

*membersIDs* attribute of the addressed *<group>* resource and fan outrequests to the members hosting CSEs addressing the obtained IDsappended with the ID of the *<subscription>* resource to be created* If the group Hosting CSE determines that multiple members resources belong

to one CSE according to the IDs of the member resources, it may convergethe requests accordingly before sending out. This may be accomplished bythe *<group>* Hosting CSE creating a *<group>* resource on the membersHosting CSE to collect all the members on that members Hosting CSE* After receiving the responses from the members hosting CSEs, respond to

the Originator with the aggregated results and the associated *memberIDs* |
| Processing at MemberHosting CSE | For the subscribe/un-subscribe procedure, the Members Hosting CSE shall treat therequest received from the group Hosting CSE as a normal SUBSCRIBE request on theaddressed member resource as if it comes from the original Originator. Therefore themembers Hosting CSE shall:* Check if the original Originator has the READ permission on the members

Resource* Upon successful validation, perform the subscribe procedures for the

corresponding type of member resource as described in clause 10.2.12* Send the corresponding response to the group Hosting CSE
 |
| Information in Response message | Converged responses from members hosting CSEs |
| Processing at Originator after receiving Response | None |
| Exceptions | * Same request with identical group request identifier received
* Originator does not have the access control privilege to access the

*<fanOutPoint>* resource |

FROM TS-0004

### 7.4.15 Resource Type <fanOutPoint>

#### Introduction

The <fanOutPoint> resource is a virtual resource because it does not have a representation. It is the child resource of a <group> resource. Whenever the request is sent to the <fanOutPoint> resource, the request is fanned out to each of the members of the <group> resource indicated by the memberIDs attribute of the <group> resource. The responses (to the request) from each member are then aggregated and returned to the Originator. The detailed description can be found in clause 9.6.14 in TS-0001 [**Error! Reference source not found.**].

There are no common attributes, resource specific attributes or xsd file to <fanOutPoint> resource because it''s a virtual resource.

A <fanOutPoint> can be addressed in one of two ways:

* Using the URI retrieved from its parent <group> resource; or
* Using a hierarchical URI formed by taking the hierarchical URI of the parent <group> and appending the string /fanOutPoint to that URI

This hierarchical URI can be extended by appending further path elements beyond the place where /fanOutPoint/ occurs. A request sent to such a URI is not fanned out to the group members, but instead it is fanned out to the resources located by taking the hierarchical URI of each group member in turn and then appending the additional path elements to that URI.

For example, if /IN-CSE-0001/myGroup were a group with members

* /IN-CSE-0001/m1 and
* /IN-CSE-0001/m2

then a request sent to /IN-CSE-0001/myGroup/fanOutPoint/x/y would be fanned out to

* /IN-CSE-0001/m1/x/y and
* /IN-CSE-0001/m2/x/y

The additional path elements can reference virtual resources, for example if m1 and m2 were both <container> resources then a request sent to /IN-CSE-0001/myGroup/fanOutPoint/latest would be fanned out to the most recent <contentInstance> child resource of both m1 and m2.

If the members m1 and m2 are themselves also <group> resources, a request sent to /IN-CSE-0001/myGroup/fanOutPoint will be fanned out to all the members of m1 and all members of m2.

.

#### <fanOutPoint> operations

##### Validate the type of resource to be created

If this is a CREATE request and the memberType attribute of the addressed parent group resource is not "MIXED", the group hosting CSE may check whether the type of resource to be created is a valid and compatible child resource type of the group members. If they are not consistent, the request shall be rejected with a ***Response Status Code*** indicating "MEMBER\_TYPE\_INCONSISTENT" error.

If the ***To*** parameter includes …/fanOutPoint without any additional appended relative address, then the type of resource specified by the memberType attribute of the parent group resource shall be checked to ensure that it is compatible with the type of child resource to be created.

If the ***To*** parameter includes an additional appended relative address after the fanOutPoint element and the Hosting CSE is able to determine the corresponding resource type (e.g. relative address corresponds to a virtual resource having a fixed name and known type), then this type shall be checked to ensure that is compatible with the type of child resource to be created.

Otherwise if the hosting CSE is not able to determine the type of the resource targeted by the relative address it shall not perform the validation.

##### Sub-group creation for members residing on the same CSE

The group hosting CSE shall obtain URIs of addressed resources from the attribute ***memberIDs*** of the parent <group> resource. The group hosting CSE may determine that multiple member resources belong to the same remote member hosting CSE, and may perform as an Originator to request to create a sub-group containing the specific multiple member resources in that member hosting CSE. This sub-group is created in the member hosting CSE as described in clause **Error! Reference source not found.**. The ***To*** parameter of this group Create request may be <memberHosting cseBase>/ <groupHosting remoteCse>/ or <memberHosting cseBase>/ etc. The group hosting CSE shall also provide ***From*** parameter (i.e. group hosting CSE) and sub-group resource representation that contains a ***memberIDs*** attribute with all the members residing on the addressed member Hosting CSE. The sub-group representation may include the attribute ***accessControlPolicyID***s, so that the group hosting CSE has the access right to this sub-group. The ID of the sub-group may be proposed by the group hosting CSE and determined by the member hosting CSE or it may be given by the member hosting CSE.
If there is already a sub-group resource defined in the remote member hosting CSE, then the group hosting CSE may utilize the existing sub-group resource.

##### Assign URI for aggregation of notification

If the request is a request to create a <subscription> resource, the group hosting CSE shall validate the request to check whether it contains a notificationForwardingURI attribute or not. If it does not, the group hosting CSE shall forward it to the group memebers. If it does, the group hosting CSE shall assign a new URI to the notificationURI attribute of the <subscription> in the requests before forwarding it to the group members. This new URI shall address the group hosting CSE so that it can receive and aggregate Notifications from those subscriptions.

##### Fanout Request to each member

For each member, the group hosting CSE shall perform the following steps:

a) The primitive parameters ***From*** and ***To*** shall be mapped to the primitive parameters of the corresponding Request to be sent out to each member of the group. The primitive parameter ***From*** shall be directly used. The prefix of primitive parameter ***To*** i.e. <URI of group resource>/fanOutPoint shall be replaced by hierarchical URIs derived from the attribute ***memberIDs*** of the group resource, but excluding the member resources which construct a sub-group if the sub-group was created by the group hosting CSE. In addition, any additional relative address that was appended to .../fanOutPoint in the original Request shall be appended to each ***To*** URI. For the member resources contained in a sub-group that was created by the group hosting CSE, the primitive ***To*** of the composed Request shall be <URI of sub-group resource>/fanOutPoint plus any additional appended relative address including in the original Request. The group hosting CSE shall execute "Compose Request primitives". In addition, the group hosting CSE shall generate a unique group request identifier, add it as a primitive parameter to the Request and locally store the group request identifier as per the local policy.

b) "Send the Request to the receiver CSE".

c) "Wait for Response primitives".

The procedures between group hosting CSE and member hosting CSEs shall comply with the corresponding creation procedures as described in clause **Error! Reference source not found.**. The detailed procedures are according to the type of Resource provided in the Request primitive. During fanOutPoint manipulation, the member hosting CSE receiving a Request send from the group hosting CSE shall check if the Request contains a ***Group Request Identifier*** parameter. If the Request contains a ***Group Request Identifier*** parameter, the member hosting CSE shall compare the ***Group Request Identifier*** parameter to the ***Group Request Identifier*** locally stored. If a match is found, the member hosting CSE shall reject the request with the ***Response Status Code*** indicating "GROUP\_REQUEST\_IDENTIFIER\_EXISTS" error in the Response primitive. Otherwise, the member hosting CSE shall continue with the operations according to the Request and locally store the ***Group Request Identifier*** parameter.

#### <fanOutPoint> resource specific procedure on CRUD operations

##### Introduction

This clause describes <fanOutPoint> resource specific behaviour for CRUD operations.

##### Create

A Create operation sent to a <fanOutPoint> is fanned out to the members (if any) of the parent <group>. It is equivalent to sending a Create to each member and therefore results in new resources being created as children of these existing members.

If the Create is sent to a hierarchical URI containing a fanOutPoint and an additional path relative to that fanOutPoint then the new resources are not created as immediate children of the members, rather they are created as children of descendents of those members (as determined by the relative path).

***Originator***:

Primitive specific operation after Orig-1.0 "Compose Request primitive" and before Orig-2.0 "Send the Request to the receiver CSE": In the case the Originator wants to subscribe to all the member resources of the group and the originator wants the group hosting CSE to aggregate all the notifications come from its member hosting CSEs, the Originator shall include ***notificationForwardingURI*** attribute in the <subscription> resource.

***Receiver:***

Primitive specific operation after Recv-6.2 "Check existence of the addressed resource" and before Recv-6.3 "Check authorization of the Originator".

Primitive specific operation additional to Recv-6.3 "Check authorization of the Originator": The Group Hosting CSE shall check the authorization of the Originator based on the ***membersAccessControlPolicyIDs***of the parent <group> resource. In the case the ***membersAccessControlPolicyIDs***is not provided, the ***accessControlPolicyIDs***of the parent <group> resource shall be used.

Primitive specific operation to replace Recv-6.5"Create/Update/Retrieve/Delete/Notify operation is performed" and Recv-6.6"Announce/De-announce the resource" in the generic procedure:

1. Validate the type of of resource to be created, refer to clause 7.4.15.2.1.
2. Sub-group creation for members residing on the same CSE, refer to clause 7.4.15.2.2.
3. Assign URI for aggregation of notification, refer to clause 7.4.15.2.3.
4. Fanout Request to each member, refer to clause 7.4.15.2.4.

5) The group hosting CSE shall aggregate the Responses after receiving responses from its member resources and sub-groups and aggregate the Responses:

Primitive specific operation additional to Recv-6.7 "Create a success response", the Response shall include the aggregated Responses.

#####  Retrieve

Originator:

No primitive specific operations.

Receiver:

Primitive specific operation after Recv-6.2 "Check existence of the addressed resource" and before Recv-6.3 "Check authorization of the Originator".

Primitive specific operation additional to Recv-6.3 "Check authorization of the Originator": The Group Hosting CSE shall check the authorization of the Originator based on the ***membersAccessControlPolicyIDs***of the parent group resource. In the case the ***membersAccessControlPolicyIDs***is not provided, the ***accessControlPolicyIDs***of the parent group resource shall be used.

Primitive specific operation to replace Recv-6.5"Create/Update/Retrieve/Delete/Notify operation is performed" and Recv-6.6 "Announce/De-announce the resource" in the generic procedure:

1. Sub-group creation for members residing on the same CSE, refer to clause 7.4.15.2.2.
2. Fanout Request to each member, refer to clause 7.4.15.2.4.

3) The group hosting CSE shall aggregate the Responses after receiving responses from its member resources and sub-groups and aggregate the Responses:

Primitive specific operation additional to Recv-6.7 "Create a success response", the Response shall include the aggregated Responses.

##### Update

Originator:

No primitive specific operations.

Receiver:

Primitive specific operation after Recv-6.2 "Check existence of the addressed resource" and before Recv-6.3 "Check authorization of the Originator".

Primitive specific operation additional to Recv-6.3 "Check authorization of the Originator": The Group Hosting CSE shall check the authorization of the Originator based on the ***membersAccessControlPolicyIDs***of the parent group resource. In the case the ***membersAccessControlPolicyIDs***is not provided, the ***accessControlPolicyIDs***of the parent group resource shall be used.

Primitive specific operation to replace Recv-6.5 "Create/Update/Retrieve/Delete/Notify operation is performed" and Recv-6.6"Announce/De-announce the resource" in the generic procedure:

1. Sub-group creation for members residing on the same CSE , refer to clause 7.4.15.2.2.
2. Fanout Request to each member. See Clause 7.4.15.2.4.
3. The group hosting CSE shall aggregate the Responses after receiving responses from its <member> resources and sub-groups and aggregate the Responses:

Primitive specific operation additional to Recv-6.7 "Create a success response", the Response shall include the aggregated Responses.

##### Delete

The primitive deletes the member resources and their child resources belonging to an existing <group> resource.

Originator:

No primitive specific operations.

Receiver:

Primitive specific operation after Recv-6.2 "Check existence of the addressed resource" and Recv-6.3 "Check authorization of the Originator": The ***To*** parameter consists of the URI of the group resource plus a suffix consisting of /fanOutPoint or /fanOutPoint/plus any additional appended relative address.

Primitive specific operation additional to Recv-6.3 "Check authorization of the Originator": The Group Hosting CSE shall check the authorization of the Originator based on the ***membersAccessControlPolicyIDs***of the parent group resource. In the case the ***membersAccessControlPolicyIDs***is not provided, the ***accessControlPolicyIDs***of the parent group resource shall be used.

Primitive specific operation to replace Recv-6.5"Create/Update/Retrieve/Delete/Notify operation is performed" and Recv-6.6"Announce/De-announce the resource" in the generic procedure:

1. Sub-group creation for members residing on the same CSE , refer to clause 7.4.15.2.2
2. Fanout Request to each member. See clause 7.4.15.2.4
3. The group hosting CSE shall aggregate the Responses after receiving responses from its <member> resources and sub-groups and aggregate the Responses:

Primitive specific operation additional to Recv-6.7 "Create a success response", the Response shall include the aggregated Responses.

**Valid Behaviour (BV):** test group that handles valid exchanges of messages, which are properly structured and correctly

**Invalid Behaviour (BI):** test group that handles valid exchanges of messages, which are either not properly structured or incorrectly encoded

**Inopportune Behaviour (BO):** test group that handles invalid exchanges of messages, which are properly structured and correctly encoded

Summary of test requirements and test objectives (the TP defined in this document are highlighted):

REQ-0001-10xxx-1; TP/oneM2M/CSE/GMG/BV/00X1

1. (BV) Allow OPERATION when the Originator has OPERATION\_PERMISSION specified in *membersAccessControlPolicyIDs* in the group resource.
2. (BO) Deny OPERATION when the Originator does NOT have OPERATION\_PERMISSION specified in membersAccessControlPolicyIDs in the group resource.

REQ-0001-10xxx-2

1. (BV) Allow OPERATION when membersAccessControlPolicyIDs in the group resource is empty AND the Originator has OPERATION\_PERMISSION specified in accessControlPolicyIDs in the group resource.
2. (BO) Deny OPERATION when membersAccessControlPolicyIDs in the group resource is empty AND the Originator does NOT have OPERATION\_PERMISSION specified in accessControlPolicyIDs in the group resource.

REQ-0001-10xxx-3 (does not apply to subscription resource)

1. (BV) Generate a request primitive for each resource in *memberIDs* with no appended relative address*.* Send original request payload.
2. (BV) Generate a request primitive for each resource in *memberIDs* with relative address appended to it. Send original request payload.
3. (BV) Generate a request primitive for each resource in *memberIDs* with relative address that includes a virtual resource appended to it. Send original request payload.

REQ-0001-10xxx-4

1. (BV) Generate request primitives with the **From** primitive parameter set to the original Originator.

REQ-0001-10xxx-7

1. (BV) Generate request primitives with the **GroupRequestID** primitive parameter set to a “unique” value.

REQ-0001-10xxx-6

1. (BV) Respond to the originator with a valid aggregated response primitive

REQ-0001-10xxx-7

1. (BV) Respond to a request where the **GroupRequestID** from the received primitive IS NOT stored locally normally.
2. (BO) Respond to a request where the **GroupRequestID** from the received primitive IS stored locally with an error ??? (How long to store the gid? Until ***Result Expiration Timestamp***)

REQ-0001-10xxx-8

1. (BV) Allow OPERATION when the Originator has OPERATION\_PERMISSION specified in accessControlPolicyIDs in the target member resource when the member hosting CSE is the same as the group hosting CSE.
2. (BO) Deny OPERATION when the Originator does NOT have OPERATION\_PERMISSION specified in accessControlPolicyIDs in the target member resource when the member hosting CSE is the same as the group hosting CSE.
3. (BV) Allow OPERATION when the Originator has OPERATION\_PERMISSION specified in accessControlPolicyIDs in the target member resource when the member hosting CSE is NOT the same as the group hosting CSE.
4. (BO) Deny OPERATION when the Originator does NOT have OPERATION\_PERMISSION specified in accessControlPolicyIDs in the target member resource when the member hosting CSE is NOT the same as the group hosting CSE.

The previous tests apply to resources other than <subscription> resources. The following apply to CREATE <subscription> fanoutPoint operations.

REQ-0001-10xxx-9

1. (BV) if the *notificationForwardingURI* attribute is present change the *notificationURI* attribute request payload to target the group hosting CSE.

REQ-0001-10xxx-10

1. (BV) Set the *groupId* attribute of the <subscription> resource to the *resourceID* of the targeted <group> resource.

???

1. (BV) if the *resourceName* attribute is not present assign a value to the <subscription> resource *resourceName* attribute

REQ-0004-10xxx-1

1. (BV) if the *notificationForwardingURI* attribute is present change the *notificationURI* attribute request payload to target the group hosting CSE

REQ-0004-10xxx-1

1. When a member of a <group> is another <group> the fanoutPoint request /myGroup/fanoutPoint should be fanned out to all members of myGroup and all members of the “member <group>”.

REQ-0004-10xxx-2

1. (BI) When the *memberType* attribute is not “MIXED” and the OPERATION is CREATE and there is no additional appended relative address, then the CSE should return "MEMBER\_TYPE\_INCONSISTENT" when the resource type being created is not a valid child of the resource type specified in *memberType*.

REQ-0004-10xxx-3#This is an optional procedure. However, if the procedure is done, it must be done in this manner. Therefore, when running this test, a valid response is to not see this procedure.

1. (BV) Verify that during a <fanoutPoint> request, where 2 or more members reside on the same remoteCSE, the SUT creates a <group> resource on the remoteCSE at either <memberHosting cseBase>/ <groupHosting remoteCse>/ or <memberHosting cseBase>/.

REQ-0004-10xxx-4#This is an optional procedure. However, if the procedure is done, it must be done in this manner. Therefore, when running this test, a valid response is to not see this procedure.

1. (BV) Verify that during a <fanoutPoint> request, where 2 or more members reside on the same remoteCSE, the SUT creates a <group> resource on the remoteCSE with the ***From*** parameter set to the group hosting CSE-ID.

REQ-0004-10xxx-5#This is an optional procedure. However, if the procedure is done, it must be done in this manner. Therefore, when running this test, a valid response is to not see this procedure.

1. (BV) Verify that during a <fanoutPoint> request, where 2 or more members reside on the same remoteCSE, the SUT creates a <group> resource on the remoteCSE with the <group> resource containing the ***memberIDs*** hosted on that remoteCSE.

REQ-0004-10xxx-6#This is an optional procedure. However, if the procedure is done, it must be done in this manner. Therefore, when running this test, a valid response is to not see this procedure.

1. (BV) Verify that during a <fanoutPoint> request, where 2 or more members reside on the same remoteCSE, the SUT creates a <group> resource on the remoteCSE with ***accessControlPolicyID***s that grant the same access permissions as the original <group> resource plus permissions for the original group hosting CSE to manage this new <group> resource.

# ===================Start of change 1 ===============

|  |  |
| --- | --- |
| **TP Id** | TP/oneM2M/CSE/GMG/BV/0023 |
| **Test objective** | Allow a <group>/fanoutPoint OPERATION when the Originator has OPERATION\_PERMISSION specified in *membersAccessControlPolicyIDs* in the group resource. |
| **Reference** | TS-0001 10.2.7.7 |
| **Config Id** | CF01 |
| **PICS Selection** | PICS\_CSE |
| **Initial conditions** | **with {** the IUT **being** in the "initial state"  **and** the IUT **having registered** the AE **and** the IUT **having** a <group> resource at TARGET\_RESOURCE\_ADDRESS  **containing** *membersAccessControlPolicyIDs* ***set to*** allow the AE privileges to  perform OPERATION **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1 **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2 **}** |
| **Expected behaviour** | **Test events** | **Direction** |
| **when {** the IUT **receives** a valid OPERATION request **from** AE **containing**  To **set to** TARGET\_RESOURCE\_ADDRESS/fanOutPoint**and** From **set to** AE-ID**}** | IUT 🡨 AE |
| **then {** the IUT **sends** a Response message **containing**  Response Status Code **set** RESPONSE\_STATUS\_CODE Content **set to** aggregated\_response **containing** Response for MEMBER\_RESOURCE\_ADDRESS1 **and**Response for MEMBER\_RESOURCE\_ADDRESS2 **}** | IUT 🡪 AE |

|  |  |
| --- | --- |
| **TP Id** | **OPERATION** |
|  TP/oneM2M/CSE/GMG/BV/0023\_1 | CREATE |
| TP/oneM2M/CSE/GMG/BV/0023\_2 | RETRIEVE |
| TP/oneM2M/CSE/GMG/BV/0023\_3 | UPDATE |
| TP/oneM2M/CSE/GMG/BV/0023\_4 | DELETE |

|  |  |
| --- | --- |
| **TP Id** | TP/oneM2M/CSE/GMG/BO/0024  |
| **Test objective** | Deny a <group>/fanOutPointOPERATION when the Originator does not have OPERATION\_PERMISSION specified in *membersAccessControlPolicyIDs* in the group resource. |
| **Reference** | TS-0001 10.2.7.7 |
| **Config Id** | CF01 |
| **PICS Selection** | PICS\_CSE |
| **Initial conditions** | **with {** the IUT **being** in the "initial state"  **and** the IUT **having registered** the AE **and** the IUT **having** a <group> resource at TARGET\_RESOURCE\_ADDRESS  **containing** *membersAccessControlPolicyIDs* ***set to*** allow the AE privileges to  perform all operations except OPERATION **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1 **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2 **}** |
| **Expected behaviour** | **Test events** | **Direction** |
| **when {** the IUT **receives** a valid OPERATION request **from** AE **containing**  To **set to** TARGET\_RESOURCE\_ADDRESS/fanOutPoint**and** From **set to** AE-ID**}** | IUT 🡨 AE |
| **then {** the IUT **sends** a Response message **containing**  Response Status Code **set to** ORIGINATOR\_HAS\_NO\_PRIVILEGE**}** | IUT 🡪 AE |

|  |  |
| --- | --- |
| **TP Id** | **OPERATION** |
| TP/oneM2M/CSE/GMG/BV/0024\_1 | CREATE |
| TP/oneM2M/CSE/GMG/BV/0024\_2 | RETRIEVE |
| TP/oneM2M/CSE/GMG/BV/0024\_3 | UPDATE |
| TP/oneM2M/CSE/GMG/BV/0024\_4 | DELETE |

|  |  |
| --- | --- |
| **TP Id** | TP/oneM2M/CSE/GMG/BV/0025 |
| **Test objective** | Allow a <group>/fanoutPoint OPERATION when the Originator has OPERATION\_PERMISSION specified in *accessControlPolicyIDs* and the *membersAccessControlPolicyIDs* is empty in the <group> resource. |
| **Reference** | TS-0001 10.2.7.7 |
| **Config Id** | CF01 |
| **PICS Selection** | PICS\_CSE |
| **Initial conditions** | **with {** the IUT **being** in the "initial state"  **and** the IUT **having registered** the AE **and** the IUT **having** a <group> resource at TARGET\_RESOURCE\_ADDRESS  **containing** *membersAccessControlPolicyIDs* ***set to*** empty  **containing** *accessControlPolicyIDs* ***set to*** allow the AE privileges to  perform OPERATION **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1 **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2 **}** |
| **Expected behaviour** | **Test events** | **Direction** |
| **when {** the IUT **receives** a valid OPERATION request **from** AE **containing**  To **set to** TARGET\_RESOURCE\_ADDRESS/<group>/fanOutPoint**and** From **set to** AE-ID**}** | IUT 🡨 AE |
| **then {** the IUT **sends** a Response message **containing**  Response Status Code **set** RESPONSE\_STATUS\_CODE Content **set to** aggregated\_response **containing** Response for MEMBER\_RESOURCE\_ADDRESS1 **and**Response for MEMBER\_RESOURCE\_ADDRESS2 **}** | IUT 🡪 AE |

|  |  |
| --- | --- |
| **TP Id** | **OPERATION** |
| TP/oneM2M/CSE/GMG/BV/0025\_1 | CREATE |
| TP/oneM2M/CSE/GMG/BV/0025\_2 | RETRIEVE |
| TP/oneM2M/CSE/GMG/BV/0025\_3 | UPDATE |
| TP/oneM2M/CSE/GMG/BV/0025\_4 | DELETE |

|  |  |
| --- | --- |
| **TP Id** | TP/oneM2M/CSE/GMG/BO/0026  |
| **Test objective** | Deny a <group>/fanOutPoint OPERATION when the Originator does not have OPERATION\_PERMISSION specified in *accessControlPolicyIDs* and the *membersAccessControlPolicyIDs* is empty in the <group> resource. |
| **Reference** | TS-0001 10.2.7.7 |
| **Config Id** | CF01 |
| **PICS Selection** | PICS\_CSE |
| **Initial conditions** | **with {** the IUT **being** in the "initial state"  **and** the IUT **having registered** the AE **and** the IUT **having** a <group> resource at TARGET\_RESOURCE\_ADDRESS **containing** *membersAccessControlPolicyIDs* ***set to*** empty  **containing** *accessControlPolicyIDs* ***set to*** allow the AE privileges to  perform all operations except OPERATION **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1 **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2 **}** |
| **Expected behaviour** | **Test events** | **Direction** |
| **when {** the IUT **receives** a valid OPERATION request **from** AE **containing**  To **set to** TARGET\_RESOURCE\_ADDRESS/fanOutPoint**and** From **set to** AE-ID**}** | IUT 🡨 AE |
| **then {** the IUT **sends** a Response message **containing**  Response Status Code **set to** ORIGINATOR\_HAS\_NO\_PRIVILEGE**}** | IUT 🡪 AE |

|  |  |
| --- | --- |
| **TP Id** | **OPERATION** |
| TP/oneM2M/CSE/GMG/BV/0026\_1 | CREATE |
| TP/oneM2M/CSE/GMG/BV/0026\_2 | RETRIEVE |
| TP/oneM2M/CSE/GMG/BV/0026\_3 | UPDATE |
| TP/oneM2M/CSE/GMG/BV/0026\_4 | DELETE |

|  |  |
| --- | --- |
| **TP Id** | TP/oneM2M/CSE/GMG/BV/0027 |
| **Test objective** | Generate a request primitive for each resource in *memberIDs* with no relative address appended to it. |
| **Reference** | TS-0001 10.2.7.7 |
| **Config Id** | CF01 |
| **PICS Selection** | PICS\_CSE |
| **Initial conditions** | **with {** the IUT **being** in the "initial state"  **and** the IUT **having registered** the AE **and** the IUT **having** a <group> resource at TARGET\_RESOURCE\_ADDRESS  **containing** *membersAccessControlPolicyIDs* ***set to*** allow the AE privileges to  perform CREATE  **containing** *memberType* **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1  **containing** resourceType **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2 **containing** resourceType **set to** AE **}** |
| **Expected behaviour** | **Test events** | **Direction** |
| **when {** the IUT **receives** a valid CREATE request **from** AE **containing**  To **set to** TARGET\_RESOURCE\_ADDRESS/fanOutPoint**and** From **set to** AE-ID Content **set to** *<*container*>* resource **containing** resourceName attribute **set to** “myContainer” resourceType **set to**  <container>**}** | IUT 🡨 AE |
| **then {** the IUT **sends** a Response message **containing**  Response Status Code **set** 2000 (OK) Content **set to** aggregated\_response **containing** Response for MEMBER\_RESOURCE\_ADDRESS1 **and**Response for MEMBER\_RESOURCE\_ADDRESS2 **}** | IUT 🡪 AE |

|  |  |
| --- | --- |
| **TP Id** | **OPERATION** |
| TP/oneM2M/CSE/GMG/BV/0027\_1 | CREATE |
| TP/oneM2M/CSE/GMG/BV/0027\_2 | RETRIEVE |
| TP/oneM2M/CSE/GMG/BV/0027\_3 | UPDATE |
| TP/oneM2M/CSE/GMG/BV/0027\_4 | DELETE |

|  |  |
| --- | --- |
| **TP Id** | TP/oneM2M/CSE/GMG/BV/0028 |
| **Test objective** | Generate a request primitive for each resource in *memberIDs* with a relative address appended to it. |
| **Reference** | TS-0001 10.2.7.7 |
| **Config Id** | CF01 |
| **PICS Selection** | PICS\_CSE |
| **Initial conditions** | **with {** the IUT **being** in the "initial state"  **and** the IUT **having registered** the AE **and** the IUT **having** a <group> resource at TARGET\_RESOURCE\_ADDRESS  **containing** *membersAccessControlPolicyIDs* ***set to*** allow the AE privileges to  perform CREATE  **containing** *memberType* **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1  **containing** resourceType **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2 **containing** resourceType **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1/myCon  **containing** resourceType **set to** container **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2/myCon **containing** resourceType **set to** container **}** |
| **Expected behaviour** | **Test events** | **Direction** |
| **when {** the IUT **receives** a valid CREATE request **from** AE **containing**  To **set to** TARGET\_RESOURCE\_ADDRESS/fanOutPoint/myCon**and** From **set to** AE-ID Content **set to** *<*contentInstance*>* resource**}** | IUT 🡨 AE |
| **then {** the IUT **sends** a Response message **containing**  Response Status Code **set** 2000 (OK) Content **set to** aggregated\_response **containing** Response for MEMBER\_RESOURCE\_ADDRESS1 **and**Response for MEMBER\_RESOURCE\_ADDRESS2 **}** | IUT 🡪 AE |

|  |  |
| --- | --- |
| **TP Id** | TP/oneM2M/CSE/GMG/BV/0029 |
| **Test objective** | Generate a request primitive for each resource in *memberIDs* with a relative address appended to fanoOutPoint that includes a virtual resource. |
| **Reference** | TS-0001 10.2.7.7 |
| **Config Id** | CF01 |
| **PICS Selection** | PICS\_CSE |
| **Initial conditions** | **with {** the IUT **being** in the "initial state"  **and** the IUT **having registered** the AE **and** the IUT **having** a <group> resource at TARGET\_RESOURCE\_ADDRESS  **containing** *membersAccessControlPolicyIDs* ***set to*** allow the AE privileges to  perform RETRIEVE  **containing** *memberType* **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1  **containing** resourceType **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2 **containing** resourceType **set to** AE **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS1/myCon  **containing** resourceType **set to** container **and** the IUT **having** a resource at MEMBER\_RESOURCE\_ADDRESS2/myCon **containing** resourceType **set to** container **}** |
| **Expected behaviour** | **Test events** | **Direction** |
| **when {** the IUT **receives** a valid RETRIEVE request **from** AE **containing**  To **set to** TARGET\_RESOURCE\_ADDRESS/fanOutPoint/myCon/latest**and** From **set to** AE-ID**}** | IUT 🡨 AE |
| **then {** the IUT **sends** a Response message **containing**  Response Status Code **set** 2000 (OK) Content **set to** aggregated\_response **containing** Response for MEMBER\_RESOURCE\_ADDRESS1/myCon/latest **and**Response for MEMBER\_RESOURCE\_ADDRESS2/myCon/latest **}** | IUT 🡪 AE |

===============End of change 1============