**3GPP TSG-SA5 Meeting #143-e *S5-223373***

**, , -**

**Source: Nokia, Nokia Shanghai Bell**

**Title: Rel-18 pCR 28.831 Improve description of current situation for key issue 2 (Targeted notification subscription)**

**Document for: Approval**

**Agenda Item: 6.5.9.4**

# 1 Decision/action requested

The group is requested to discuss and approve the pCR below

# 2 References

[1] 3GPP TS 28.831: " Management and orchestration; Study on basic Service-Based Management Architecture (SBMA) enabler enhancements"

# 3 Rationale

None.

# 4 Detailed proposal

The following changes are proposed for TR 28.831[1].

|  |
| --- |
| **Begin of modifications** |

## 4.2 Issue #2: Targeted notification subscriptions

### 4.2.1 Issue description

SA5 is moving to a fully model driven approach. In this architecture all aspects that are managed are represented in the information model. It is possible to observe all changes in the network by observing the changes in the information model. Note the information model covers all fragments: configuration management, alarm management and performance management.

Many MnS consumers are not interested in all changes in the network. For that reason, it is important that MnS consumers can subscribe to specific portions of the information model only. These portions may be described in terms of object instances, attributes or attribute fields.

The currently specified "NtfScubscriptionControl" does not allow to scope subscriptions with that granularity. Some enhancements seem to be required to "NtfScubscriptionControl". This clause analyses the current situation and proposes a solution.

### 4.2.2 Current situation

#### 4.2.2.1 Scoping objects based on object classes

For scoping managed objects, the attribute "scope" is provided in "NtfSubscriptionControl". It has two attribute fields: "scopeType" and "scopeLevel". This scoping mechanism works only on the hierarchical levels of the information model. For example, you can select the level below "SubNetwork".

Typically, instances of more than one object class can be located on a level. For example, below "SubNetwork" there can be instances of "ManagedElement", "PerfMetricJob", "TraceJob" and "AlarmList". It is not possible to scope only instances of one or more specific object classes with the current scoping mechanism.

To reduce the scoped set of object instances to those with a specific object class, the "notificationFilter" attribute can be used. The IS level parameters "objectClass" and "objectInstance" are typically mapped to one stage 3 parameter ("href") only. There is no dedicated parameter for the "objectClass" in stage 3. Filtering out notifications related to certain object classes involves therefore the application of string functions on "href".

The described mechanism applies to CM notifications and non-CM notifications such as alarm notifications.

Note on all examples:

The following examples are for the RESTful HTTP-based solution only. For this solution the "notificationFilter" format is XPath 1.0 in Rel-17. In addition, the JSON instance document sent over the wire is expressed in XML. A <notification> element is added as root element to produce a valid XML document, a necessary manipulation not described yet in Rel-17.

**Example 1 (alarm notifications):**

Assume the NRM specified in TS 28.622 is supported on a MnS producer. A subscription scoping (with "scopeType" and "scopeLevel") the level below "SubNetwork" has been created. No notification filter is specified. The subscription is for alarm notifications only. On the level below "SubNetwork" there are instances of "ManagedElement", "PerfMetricJob" and "NtfSubscriptionControl".

The described subscription forwards alarm notifications related to all manged object classes. For example, all following notifications are forwarded:

<notification>

<href>example.com/SubNetwork=SN1/ManagedElement=ME1</href>

...

</notification>

<notification>

<href>example.com/SubNetwork=SN1/ManagedElement=ME2</href>

...

</notification>

<notification>

<href>example.com/SubNetwork=SN1/PerfMetricJob=PMJ1</href>

...

</notification>

<notification>

<href>example.com/SubNetwork=SN1/NtfSubscriptionControl=NSC1</href>

...

</notification>

The following notification is not related to the set of scoped objects and hence not forwarded:

<notification>

<href>example.com/SubNetwork=SN1/ManagedElement=ME1/XyzFunction=XYZF1</href>

...

</notification>

The subscription shall be modified now to include only "ManagedElement" objects. This is realized with an appropriate XPath expression as value of "notificationFilter". Possible XPath expressions include:

**XPath expression 1:**

The following XPath expression returns the "notification" node since the string "ManagedElement" is contained in the "href" value.

"/notification[contains(href,"ManagedElement")]"

**XPath expression 2:**

The following expression checks for the presence of "ManagedElement" after "example.com/SubNetwork=SN1".

"/notification[starts-with\

(substring-after(path,"example.com/SubNetwork=SN1/"),\

"ManagedElement")]"

**Example 2 (alarm notifications):**

Assume again the NRM specified in TS 28.622 is supported on a MnS producer. "PerfMetricJob" instances have been created below "SubNetwork" and below some "ManagedElement" instances. A MnS consumer wishes to create a subscription related to all "PerfMetricJob" instances.

The notification scope needs to include the complete object tree starting at "SubNetwork". In case a "PerfMetricJob" instance is always a leaf object, then checking for the presence of the sub-string "PerfMetricJob" in "href" does the job. In case a "PerfMetricJob" instance is not always a leaf object, then it is necessary to check, if the last path segment of "href" identifies a "PerfMetricJob". A simple test on if the "href" value includes the sub-string "PerfMetricJob" is not sufficient. There is no obvious Xpath expression to solve this problem, at least not with XPath1.0.

**Example 3 ("notifyMOICreation", "notifyMOIDeletion", "notifyMOIAttributeValueChanges")**

The created, dfeleted or updated objerct is specified with the "href" parameter of the notification header. For that reason the same considerations as for alarm notifications apply.

**Example 3 ("notifyMOIChanges")**

The objects are identifierd with both the "href" and "path" parameter. Filtering on these parameters to reduce the scoped set of objects to the sub-set of interest is not a feasible solution.

#### 4.2.2.2 Scoping attributes

The scoping mechanism allows to select a set of objects. This is good enough for notification types related to the complete object, such as alarm notifications, or when a MnS consumer is interested in receiving attribute value change notifications for all attributes of an object. It is not possible to target specific attributes of an object only using the scoping mechanism. When a MnS consumer is interested only in value changes of one or more specific attributes of an object, then the MnS consumer needs to configure into the subscription an appropriate notification filter.

Note the following examples are for the RESTful HTTP-based solution only.

**Example 1 ("notifyMOIAttributeValueChanges"):**

Assume a subscription for "notifyMOIAttributeValueChanges" includes the instance of "XyzFunction" identified by "XYZF1". A notification reporting the value change of "attrA" and "attrB" looks like

<notification>

<href>example.com/SubNetwork=SN1/ManagedElement=ME1/XyzFunction=XYZF1</href>

...

<attributeListValueChanges>

<attrA>123</attrA>

<attrB>abc</attrA>

</attributeListValueChanges>

<attributeListValueChanges>

<attrA>456</attrA>

<attrB>def</attrA>

</attributeListValueChanges>

</notification>

If a MnS consumer is interested only in value changes of "attrA", then he needs to specify a notification filter testing on the presence of "attrA", for example:

"/notification/attributeListValueChanges[attrA]"

Note that the complete notification including also "attrB" is forwarded in case the test evaluates to true. The "attrB" cannot be removed from the notification. With filtering either the complete notification is forwarded, or the complete notification is not forwarded.

Note also that for "notifyMOIChanges" the situation is more complicated. Changes of multiple managed object instances can be reported using a single "notifyMOIChanges" notification. Even worse, the object instance in the notification header may not even be an instance that has changed, but only a common ancestor of the instances that have changed. This means that not only the "href" property of the notification header needs to be checked but also the "path" properties of the notification body.

#### 4.2.2.3 Scoping objects or attributes based on conditions

It is currently not possible to specify a conditional scope. A conditional scope could be based for example on the presence of an attribute or on an attribute with a specific value. Note that scoping based on object classes can be considered as conditional scope.

For example, a MnS consumer might be interested only in changes of "AlarmInformation" instances (reported with alarm notifications) with a perceived severity equal to "CRITICAL".

Editor's note:

The following issuers are agreed for further work:

1. The examples are all in XML while examples in 28.532 are in YAML. The relation beween the two must be specified.
2. This needs to be updated to follow 223359, 223360, 223388
3. To use Xpath a number of things MUST to be clarified:
   1. The notification must have a conceptual XML representation. Is it a single string as encoded by the solution set? Is it a set of XML elements per parameter? Or in case of notifyMOIChanges subparameters? Are XML attributes used at all?
   2. Specify on stage 2 that the notificationFilter is an Xpath expression.
   3. Xpath 1.0 or 2.0 ? 2.0 seems complicated? Do we force a service provider to support the full Xpath ? That's a tall order. Any limitations, options? All Xpath axes ? E.g. preceding axes ?
   4. Is document order considered, used as a concept?
   5. What do we do with Xpath namespaces ? How do we consider them?
   6. What is the Xpath function library available ?
   7. Are there any variable bindings ?
   8. What is the accessible tree for Xpath? E.g. Are the Http headers included? VES headers? Are only notifyable attributes included  or all attributes?
   9. What is the root node of the accesible subtree ?
   10. What is the context node in Xpath ? E.g. if I call the function current() which node is returned
   11. Specify what is the canonical representation of the attribute values. If we have a comparision statement will should it compare to True, TRUE or true? 4 or 4.0 or +4.0 ?
   12. Specify that the Xpath is evealuate against the relevant stage 3 data model. Stage 2 evaluation may be extra tricky and not consistent.
   13. How do we address individual values/elements in a multivalue attribute? Indexes, keys?
   14. Can Xpath work on the new value or also on the oldValue? What if oldValue is not supported? Should such a subscription be rejected ?
   15. Numbers in XPath 1.0 are IEEE 754 [IEEE754-2008] double-precision floating-point values;  This means that    some values of int64, uint64, and decimal64 types cannot be exactly represented in XPath expressions.
4. An alternative is to define some own Xpath functions for specific use cases.
5. In YANG-Push which is a better solution for this topic, the notificationFilter for data change notification is evaluate against the conseptual datastore (the instantiated NRM) not against notification body. We propose to use that here too.
   1. This avoids complex string manipulattion on href and path, but needs the SW to build up a conceptual model, which it probably already has.
   2. This would mean we have to define conceptual XML representation, accessible subtree, context node for the datastore.
   3. If we follow the stage 3 concepts for the XML representation this is possible. All IOCs, attribute, attribute fields are elements. The attribute container is an element. Actually the YANG mappings XML representation following RFC 7950 is nearly ready made for this.
6. In 28.622 notificationFilter is simply described as string. In the OpenAPI solution set, filter is indicated as having Xpath format, But the YANG solution set currently doesn’t make such additional statements for notificationFilter in NtfSubscriptionControl.
7. It is not clear why the scenario in example 2 in clause 4.2.2.1 cannot be supported.

Regarding scoping based on condition in clause 4.2.2.3, there is no AlarmInformation entity in the generic NRM (28.622). Additionally, alarm records are not notifyable, meaning that they are not reported using MOI change notifications. So the example appears to be incorrect.

|  |
| --- |
| **End of modifications** |