**3GPP TSG-SA5 Meeting #158 *S5-247083***

Orlando, USA, 18 - 22 November 2024

**Source: Huawei**

**Title: pCR TR 28.867 Clarification of escalationRecipient attribute and escalation recipient CCL**

**Document for: Approval**

**Agenda Item: 6.19.4**

# 1 Decision/action requested

***The group is asked to discuss and agree on the proposal.***

# 2 References

[1] 3GPP TR 28.867: "Management and orchestration; Closed control loop management".

# 3 Rationale

The escalationRecipient is an attribute added to identify the CCL acted as an escalation recipient, while escalation recipient CCL is an entity that can performs the actions, e.g., can send an output representing the outcome of the escalation. Therefore, these two terms need to be distinguished in clause 5.10.3. Besides, there are some redundant sentences and wording issues which should be addressed.

# 4 Detailed proposal

This document proposes the following changes in TR 28.867[1].

|  |
| --- |
| **1st Change** |

### 5.10.3 Possible solutions



Figure 5.10.3-1: End-to-end flow of CCL decision escalation

1. Introduce an attribute on the CCL for identifying an entity to which the decision is escalated to. This attribute is called escalationRecipient.
2. Introduce an attribute on the CCL for defining the condition that triggers the escalation. For example, the CCL may trigger escalation when its level of confidence in the derived decision is below some threshold, say called a confidence threshold. The confidence threshold attribute enables the CCL to autonomously make decisions for each situation and context based on its computed confidence level in the given situation. If the confidence level is lower than the confidence threshold the decision is escalated otherwise the decision is executed.
3. Introduce an IOC representing the request for escalation, say called EscalationRequest, that holds all information related to the request for escalation. Introduce a corresponding attribute on the escalation recipient. The EscalationRequest may include:

- A proposedCMChange attribute which describes the configuration management changes that have been proposed by the escalator CCL.

- A decisionConstraints attribute indicating the constraints observed by the escalating CCL in making the decision(s). The constraints may be the type context as defined in 3GPP TS 28.312 [5] or conditions expressed using JEX/XPATH

- The EscalationReason attribute can provide an optional description of the reason behind the escalation to provide more context and further clarification.

1. The escalation recipient may send an output representing the outcome of the escalation. Introduce in the AssuranceReport an escalationOutcome data type that holds the outcomes of the escalation. When the escalationOutcome attribute of the report is updated, the escalation recipient CCL sends a notification to the escalator CCL that the report is available:

- The escalation outcome indicates whether the CCL should take any action and what that action is. Introduce an attribute to indicate what should be done by the CCL. It is an ENUM with the values:

1. "DONOTHING"- indicating that the CCL does not need to take any action, i.e., the escalation recipient is addressing the scenario.
2. "APPLYACTION"- indicating that the CCL should apply a specific set of actions proposed by the escalation recipient. The action may be placed in a new attribute, say called proposedActions, that list as et of CM changes.
3. "APPLYGUIDANCE"- indicating that the CCL should compute a new CM change based on the guidance from the escalation recipient. The guidance may be placed in the proposedActions attribute.

|  |
| --- |
| **Next Change** |

## 5.11 Use case 11: Performance Evaluation of a Closed Control Loop

### 5.11.1 Description

The advanced monitoring functionalities of a CCL can provide real-time insights into the performance and outcomes of a CCL. The monitoring activity for a Closed Control Loop may result in further actions that happen in the operation phase, e.g., evaluate and update, in order to change the closed control loop settings and improve its performance. So, there is a need to evaluate the performance of a Closed Control Loop itself. Such metrics are important to understand and change a CCL's behaviour and to improve its performance to pursue the assigned goal(s).

For example, certain performance aspects of a CCL can be very crucial to know in order to evaluate and decide upon a CCL's performance, such as the number of breached goals, time taken to meet a breached goal, number of conflicts occurred by a CCL etc. With the knowledge of such performance aspects of an existing CCL, a MnS consumer can more effectively update or create a new CCL.

An operator can also compare different CCLs based on these performances and choose the best one for its network deployment.

|  |
| --- |
| **Next Change** |

## 5.12 Coordinating CCLs with other management functions

### 5.12.1 Description

A CCL makes and executes decisions in different network contexts and for different network functions and parameters. Yet within the network, there may be other management functions or features including MDA functions, SON functions, and AIML Functions, which also make decisions that affect the same network functions and parameters as the CCL. The operation of CCLs needs to be coordinated with the other management functions.

**NOTE 1:** This use-case only focuses on coordinating CCLs with other management functions for executing decisions.

### 5.12.2 Potential requirements

**REQ- CCL-ESC-1:** The CCL MnS producer should have a capability to indicate to a coordination functionality the set of network functions including their parameters which it is interested in changing.

**REQ- CCL-ESC-2:** The CCL coordination MnS producer should have a capability to enabling an authorized MnS consumer to receive information on the latest changes to a network function parameter and an identifier of a management entity/function responsible for the change to the parameter.

**Note 2:** an example MnS consumer may be a CCL instance.

**Note 3:** an example of management entity/function responsible for the change to the parameter may include an instance of a MDA Function, a SON Function or an AI/ML inference Function. These functions may not necessarily execute or be responsible for executing the change.

**REQ-CCL-ESC-3:** The CCL coordination MnS producer should have a capability to enabling an authorized MnS consumer to receive the history of previous values of the parameter, e.g., from the coordination functionality.

**Note 4:** The history includes, for each previous value, the identifier of a respective management entity/function responsible for that change to the parameter.

### 5.12.3 Possible solutions

#### 5.12.3.1 Required capabilities and interactions.

For a given context, the CCL should indicate to a coordination functionality the set of network functions and corresponding parameters which it is interested in changing. Accordingly, the coordination functionality may subscribe to changes to the network functions and corresponding parameters. Then in case of changes to subscribed network functions and parameters, the network functions will notify the coordination functionality of these changes. The coordination functionality subsequently notifies the CCL of the information on the latest changes to a network function parameter and an identifier of a management entity/function (e.g., CCL, MDA, SON, AI/ML inference Function) responsible for the change to the parameter,

The CCL determines the history of previous values of the parameter, e.g., from the coordination functionality. The history includes, for each previous value, the identifier of a respective management entity/function responsible for that change to the parameter. The CCL defines a favourable range of values of the parameter based on the received information on the latest change and the history of previous changes to the parameter, The CCL calculates a new value of the parameter considering the favourable range as a constraint for the new value. The CCL can then update the value of the parameter of the network function to the new value.

#### 5.12.3.2 Information objects to realize required capabilities and interactions

- update the sourceIndicator attribute of the notifyMOIcreation to include a CCL’s control scope. With this, the CCL coordination entity is notified of the specific control scopes when CCL is instantiated.

- Introduce a data type and corresponding attribute on the CCL coordination entity representing information about the control scopes of different CCLs. The data type may be called cCLsControlScope and includes an attribute representing the history of changes to that control scope by any other management entity which is not the CCL. The attribute may be called cCLsControlScope.

Note: the logging of history may be based on a generic logging data functionality that is not specific to CCLs

- introduce on the cCLsControlScope an attribute representing a change to that control scope by another management entity which is not the CCL. The attribute may be called ControlScopeChange. The attribute should be notifiable, so that in case of a change, the CCL coordination entity can write the change into the ControlScopeChange which is then notified to the CCL.

- Given the notification, the CCL may read the ControlScopeChange to obtain the history of changes.

5.12.4 Evaluation of solutions

The solution in clause 5.12.3 provides the procedures and information objects to enable coordinating CCLs with other management functions including MDA functions, SON functions, and AIML Functions. No conflicting solution for coordinating CCLs with other management functions is provided. it is recommended to apply the solution in in clause 5.12.3 as the solution for coordinating CCLs with other management functions.

|  |
| --- |
| **End of Change** |