**3GPP TSG-SA5 Meeting #141eS5-221234**

**17 January to 26 January 2022, E-meeting**

**Source: China Mobile**

**Title: pCR 28.312 Update the clause 4.5, 6.2.1 and annex A to align with intent definition**

**Document for: approval**

**Agenda Item: 6.4.9**

# 1 Decision/action requested

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

# 2 References

[1] 3GPP draft TS 28.312: “Management and orchestration; Intent driven management services for mobile networks v0.7.0”.

# 3 Rationale

This pCR aimes to resolve the editor’s notes by updating the term alignment based on intent definition.

There is no statement to clarify why a new term is introduced in clause 4.5. The difference and relation between “context” and “constraint” are not clear. In the first sentence of clause 4.5.4, context and constraint look like equally. Therefore, a solution is introduced that regarding “context” as a complement component to be used in intent modelling, and needn’t to change the definition of intent.

Similiarly, the clause 6.2.1 should be updated either.

# 4 Detailed proposal

|  |
| --- |
| **1st Modified Section** |

## 4.5 General concept of Intent Content

### 4.5.1 Intent Expectation

In the most basic form, a consumer may use an intent to express to the producer the need for:

"an object O with characteristics S".

Where the characteristics S reflect the requirements, goals, constraints and contexts for the object.

The object may be a 3GPP managed object like a slice, subnetwork (e.g. radio network) or other objects like a service. The consumer may desire the same requirements, goals, constraints and contexts for multiple objects with the same properties, in which case the intent may be stated for a list of objects as

"objects {O1,O2, …ON} with characteristics S"

However, the consumer may wish to state different requirements, goals, constraints and contexts for objects of different properties. It is in that case necessary to distinguish the requirements, goals, constraints and contexts to be achieved for each set of objects with the same properties. Correspondingly, the combination of requirements, goals and constraints for each set of objects with the same properties is the IntentExpectation. Also, the consumer may wish to distinguish the requirements, goals, constraints and contexts for different objects with the same properties, in this case, the combination of requirements, goals, constraints and contexts for each instance may be contained in a separate IntentExpectations or requirements, goals, constraints and contexts for the multiple instances may be combined in a single IntentExpectation.

### 4.5.2 Expectation Targets

For a given intent expectation, the desired characteristics of the object(s) are the expectation targets to be achieved. The targets may include the metrics that characterize the performance of the object(s) or some abstract index that expresses the behavior of the object(s)). A given intent expectation may include multiple targets on the same object or on objects with the same properties. A consumer may for example require for the Slice object(s) that User throughput > 5Mbps and latency < 1ms.The expectation targets may also be context specific, i.e. the intent may require a specific targets given a specific context. As such with the characteristics as a combination of intent targets and contexts, the intent expectation may be stated as

"ensure that for

Applicable Object O,

Target\_1 is T\_1, Conext\_1 is C\_1

….,

Target\_m is T\_m, Context\_k is C\_k;

Each Target expresses an aspect of the characteristics of the object under consideration, i.e. it expresses a desired outcome on a specific object State attribute. Each of the object state attributes may be set to be equivalent to a specific value or constrained to a value or a range of values, e.g. as listed in Table 1. The combination of the name of state attribute (or simply the targetName), the condition constraining the attribute and the value or value range for the attribute is the target, i.e. the target is the tuple

target = [targetName, condition, value range]

Table 1: Example intent targets for different Objects

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **example of target** | **ApplicableObject** | **targetName** | **Condition** | **Value range** |
| example 1 | Slice | Coverage area | Is at least | 40km radius |
| example 2 | Communication Service | User throughput | Is greater than | 2Mbps |

### 4.5.3 IntentObjects

The object (s) for which a given expectation is addressed can be expressed with the object's identifier. This may, however, not always be adequate (e.g., if the consumer does not have or know the identifiers of the object) or may be cumbersome for some intents. For example, it may be easier to state "all slices in city ABC" as opposed to listing the individual slices. As such it may be easier to identify the objects by stating the object context information that filters and identifies the desired objects. The objectContext is in form of a context list whose entries are each a tuple (attribute, condition, value range). For example, in the case of "all slices in a city" there is a object context, which is the tuple "location, =, city\_ABC" and "objectType=slice".

### 4.5.4 Context

Each target may be restricted to only be achieved for a very specific set of contexts. For example, the consumer may state that: *"ensure that handoverFailureRate < 2% if Load > 80%"*, where the target *"HandoverFailureRate < 2%"* is only to be achieved only in the context *"Load > 80%"*.

Similar to the target, the context is also a tuple of < attribute, condition, value range > but which the values having a different semantics.

Although contexts and targets have the same structure, to distinguish between what must be achieved and the context which is only to be considered as required conditions, the Context has to be explicitly stated separate from the target. For example, if the consumer may wish that the Radio Link Failure rate (RLF) is less than 2% when the load is more than 50%. If the context (i.e. load > 50%) is not explicitly stated/modeled as context, the producer could interpret the request to mean (RLF<2% and load > 50%). Generally, contexts can be used to describe the scenarios and/or applicable conditions.

For a given expectation, the specific list of targets may be desired to be achieved for given combined contexts, i.e., besides the Target, an expectation may state a list of contexts which apply to all targets within the intent expectation. Similarly, there may be contexts that apply to all expectations within a given intent. Correspondingly, both Intent expectations and intents should be modeled to contain aggregate contexts that apply to all the contained sub elements.

|  |
| --- |
| **2nd Modified Section** |

## 6.2 Information model definition for Intent (MnS component typeB)

### 6.2.1 Information model definition for Intent

Editor’s Note: The following information model needs to be revisited based on the further discussion, and the alignment/coordination work with other SDO needs to be considered, which may impact the following information model.

#### 6.2.1.1 Class diagram

##### 6.2.1.1.1 Relationship

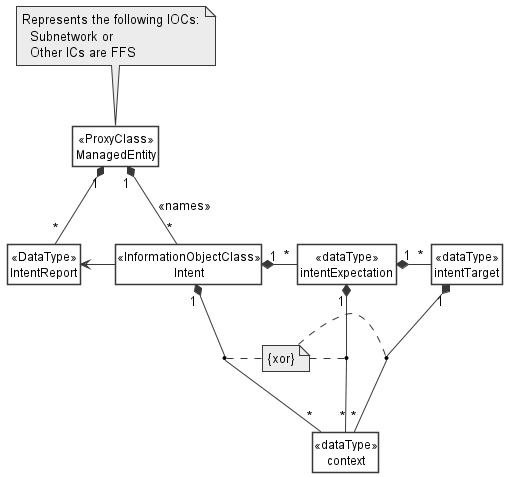


Figure 6.2.1.1.1-1 Relationship UML diagram for intent

Note: Above XOR means an instance of Context can only be name contained by either the Intent, IntentExpectation or IntentTarget.

Editor’s Note: The detailed model for Intent, IntentReport and IntentExpectation objects (e.g. is it <<IOC>>, <<DataType>>, or string) is FFS as their relationship needs to be decided later based on the content of these three objects 6.2.1.1.2 Inheritance

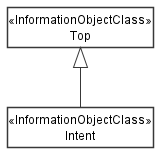


Figure 6.2.1.1.2-1 Inheritance UML diagram for intent

#### 6.2.1.2 Class definition

##### 6.2.1.2.1 Intent <<IOC>>

###### 6.2.1.2.1.1 Definition

This IOC represents the properties of an Intent. The Intent IOC contains one or multiple intentExpectation(s) which includes MnS consumer’s requirements, goals and constraints given to a 3GPP system*.*

The Intent IOC includes the attribute objectClass and objectInstance from the TOP IOC. The value of attribute objectClass is “Intent” and the value of attribute objectInstance is the DN of the instance of Intent IOC.

###### 6.2.1.2.1.2 Attributes

The Intent includes attributes inherited fromTOP IOC (defined in TS 28.622) and the following attributes:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifyable |
| intentExpectation | M | T | T | F | T |
| userLabel | M | T | T | F | T |
| intentContexts | O | T | T | F | T |
| intentFulfilStatus | M | T | F | F | T |

Editor’s Note: whether other the attributes are needed for the Intent IOC needs further discussion.

###### 6.2.1.2.1.3 Attribute constraints

None

##### 6.2.1.2.2 IntentExpectation

###### 6.2.1.2.2.1 Definition

IntentExpectation class represent MnS consumer’s requirements, goals and constraints given to a 3GPP system*.*

Editor’s Note: more description for IntentExpectation will be added later based on the further discussion.

###### 6.2.1.2.2.2 Attributes

The IntentExpectation includes the following attributes:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifyable |
| expectationId | M | T | T | T | T |
| expectationObject | O | T | T | F | T |
| expectationObjectContexts | O | T | T | F | T |
| expectationTargets | O | T | T | F | T |
| expectationContexts | O | T | T | F | T |

Note: The scenario/requirements-specific IntentExpectations are defined utilizing the constructs of this generic IntentExpectation model.

Editor’s Note: whether other the attributes are needed for the IntentExpectation IOC needs further discussion.

Editor’s Note: The naming of the terms may need further discussion.

Editor’s Note: whether both expectionObjectContexts and ExpetationTargets are mandatory is FFS.

###### 6.2.1.2.2.3 Attribute constraints

TBD

##### 6.2.1.2.3 IntentReport

###### 6.2.1.2.3.1 Definition

IntentReport class represent intent fulfilment feedback information that MnS consumer can obtained from a 3gpp system.

Editor’s Note: more description for IntentReport will be added later based on the further discussion.

###### 6.2.1.2.3.2 Attributes

TBD

###### 6.2.1.2.3.3 Attribute constraints

TBD

##### 6.2.1.2.4 IntentExpectation

###### 6.2.1.2.4.1 Definition

Following Content for MnS’s expectation on a radio network is used as example to discuss the concrete model for IntentExpectation:

Editor’s Note: following content needs to be revisited based on further discussion for intent model, how to model IntentExpectation is FFS.

Following described the information/attributes for MnS Consumer’ s expectation to deliver/ensure on a radio network,

- ManagedObject, it describes the managed object (i.e. RadioNetwork) expressed by MnS’s expectation on a radio network.

- ExpectedAreas, it describes a list of coverage areas for radio network. This can be implemented by multiple choices, e.g. CoverageAreaPolygon, TrackingAreaCode.

- RANCoverageTarget, it describes the properties of coverage quality for the radio network, which includes:

- targetWeakRSRPRatioInfo, it describes the list of target Weak RSRP Ratio for the radio network. Each TargetWeakRSRPRatioInfo includes WeakRSRPRatio and WeakRSRPThreshold.

- targetLowSINRRatioInfos, it describes the list of target low SINR Ratio for the radio network. Each TargetLowSINRRatioInfo includes LowSINRThreshold and LowSINRRatio.

- RANCapacityTarget, it describes the properties of network capacity for the radio network, which includes：

- targetMaximumUENumber, it describes the maximum number of should be supported by the radio network

- targetActivityFactor, it describes the maximum number of should be supported by the radio network

- RANUEThptTarget, it describes the properties of the RAN UE throughput for the radio network, which includes:

- targetAveULRANUEThpt, it describes the average UL RAN UE throughput for the radio network

- targetAveDLRANUEthpt, it describes the average UL RAN UE throughput for the radio network

- targetLowULRANUEThptRatioInfos, it describes the list of target low uplink RAN UE throughput ratio for the network, which includes LowULRANUEThptRatio and LowULRANUEThptThreshold

- targetLowDLRANUEThptRatioInfos, it describes the list of target low downlink RAN UE throughput ratio for the radio network, which includes LowDLRANUEThptThreshold and LowDLRANUEThptRatio

- RANConstraint, it describes the properties of a set of constraints for the radio network, which includes:

- radioConstraints, it describes the properties of a set of radio constraints for the radio network, which includes cellIdRanges, gNBIdRanges, tACRanges, pLMNIdList and nRPCIRanges.

- frequencyConstraints, it describes the properties of a set of frequency constraints for radio network, which includes arfcnUL/DL and bSChannelBwUL/DL

- transportConstraints, it describes the properties of a set of transport constraints to be used in the radio network, including ngcLocalIpAddressList, nguRemoteIpAddressList, logicInterfaceId and nextHopInfo.

#### 6.2.1.3 DataType definition

##### 6.2.1.3.1 expectationTarget <<dataType>>

###### 6.2.1.3.1.1 Definition

This <<dataType>> represents the Targets of the IntentExpectation that are required to be achieved .

###### 6.2.1.3.1.2 Attributes

The expectationTarget includes the following attributes:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifyable |
| targetName | M | T | T | T | T |
| targetCondition | M | T | T | F | T |
| targetValueRange | M | T | T | F | T |
| targetContexts | O | T | T | F | T |

###### 6.2.1.3.1.3 Attribute constraints

None

##### 6.2.1.3.2 context << dataType >>

###### 6.2.1.3.2.1 Definition

This IOC represents the properties of a context. A context describes the list of constraints and conditions that should evaluate to True when the targets are fulfilled but are themselves not to be enforced. The context may apply to the intent, the intent expectation, the intent targets or to the object as filter information used to identify the manged objects to which the targets are intended.

###### 6.2.1.3.2.2 Attributes

The context includes the following attributes:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifyable |
| contextAttribute | M | T | F | T | T |
| contextCondition | M | T | T | F | T |
| contextValueRange | M | T | T | F | T |

###### 6.2.1.3.2.3 Attribute constraints

None

#### 6.2.1.4 Attribute definition

| Attribute Name | | Documentation and Allowed Values | Properties | |
| --- | --- | --- | --- | --- |
| userLabel | | A user-friendly (and user assignable) name of the intent.  allowedValues: Not Applicable | type: String  multiplicity: 1  isOrdered: F  isUnique: F  defaultValue: None  isNullable: False | |
| intentExpectation | | It indicates the expectations including requirements, goals and constraint given to a 3GPP system. It states the list of specific outcomes desired to be realized for a certain object | type: FFS  multiplicity: \*  isOrdered: F  isUnique: F  defaultValue: None  isNullable: False | |
| intentFulfilStatus | | It describes the current status of the intent fulfilment result, which is configured by MnS producer and can be read by MnS consumer.  allowedValues: "FULFILLED", “NOT\_FULFILLED  Editor’s Note: whether other allowed values should be supported is FFS, and the name for the attribute intentFulfilStatus is FFS. | type: ENUM  multiplicity: 1  isOrdered: N/A  isUnique: N/A  defaultValue: None  isNullable: False | |
| intentContext | | It describes the list of constraints and conditions that should apply for the entire intent even if there may be specific constraints and conditions defined for specific parts of the intent.  allowedValues: triple of (attribute, condition, value range) | type: Context  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: None  isNullable: False | |
| expectationId | | A user-friendly (and user assignable) name of the intentExpectation.  allowedValues: Not Applicable | type: String  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: None  isNullable: False | |
| expectionObject | | expectation carries requirements (expectations, goals and constrains) on an expectationObject.  expectionObject refers to an object (e.g. instance of managed object) to which the intentExpectation should apply. This means, which object the requirements specified by the expectation are meant for. expectionObject is optional.  allowedValues: NA | type: DN  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: None  isNullable: False | |
| expectionobjectContext | | It describes the list of constraints and conditions to be used as filter information to identify the specific object to which a given intentExpectation should apply. Note there may be other constraints and conditions defined either for the entire intent, for the specific intentExpectation or for the expectationTarget of the considered intentExpectation.  allowedValues: depends on Object in the IntentExpectation | type: Context  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: None  isNullable: False | |
| expectionTargets | | It describes the list of specific outcomes on metrics and observables related to the Object (e.g. the metrics that characterize the performance of the object(s) or some abstract index that expresses the behavior of the object(s)) that are desired to be realized for a given intentExpectation.  allowedValues: depends on Object in the IntentExpectation | type: expectationTarget  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: None  isNullable: False | |
| expectationContext | | It describes the list of constraints and conditions that should apply for a specific intentExpectation. Note there may be other constraints and conditions defined for the entire intent or for specific parts of the intentExpectation.  allowedValues: depends on Object in the IntentExpectation | type: Context  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: None  isNullable: False | |
| targetName | | It describes a specific attribute of the object on which the outcomes are stated, either a configuration or observable of that object. The attributes may be a parameter, gauge, counter, KPI, weighted metric, etc. related to that object  allowedValues: depends on Object in the IntentExpectation | type: string  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: Null  isNullable: True | |
| targetCondition | | It expresses the limits within which the targetName is allowed/supposed to be  allowedValues: is equal to; is less than; is greater than;  Note: Others conditions like "is within the range" or "is outside the range" can be expressed in terms of these basic conditions | type: enum  multiplicity: upto 2  isOrdered: False  isUnique: False  defaultValue: "is equal to"  isNullable: False | |
| targetValueRange | | It describes the range of values that applicable to the targetName and the TargetCondition. | type: FFS  multiplicity: upto 2  isOrdered: False  isUnique: False  defaultValue: Null  isNullable: True | |
| targetContext | | It describes the list of constraints and conditions that should apply for a specific expectationTarget. Note there may be other constraints and conditions defined for the entire intent or the intentExpectation.  allowedValues: triple of (attribute, condition, value range) | type: Context  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: None  isNullable: False | |
| contextAttribute | | It describes a specific attribute of or related to the object or to characteristics thereof (e.g. its control parameter, gauge, counter, KPI, weighted metric, etc) to which the expectation should apply or an attribute related to the operating conditions of the object (such as weather conditions, load conditions, etc). | type: DN  multiplicity: 1  isOrdered: False  isUnique: False  defaultValue: Null  isNullable: True | |
| contextCondition | | It expresses the limits within which the ContextAttribute is allowed/supposed to be  allowedValues: is equal to; is less than; is greater than;  Note: Others conditions like "is within the range" or "is outside the range" can be expressed in terms of these basic conditions | type: enum  multiplicity: upto 2  isOrdered: False  isUnique: False  defaultValue: "is equal to"  isNullable: False | |
| contextValueRange | | It describes the range of values that explicatable to the ContextAttribute and the ContextCondition. | type: FFS  multiplicity: upto 2  isOrdered: False  isUnique: False  defaultValue: Null  isNullable: True | |
| **3rd Modified Section** | | |

# Annex C (informative): Mapping the 3GPP and the TM Forum intentExpectation Models

The TM forum defines the structure of an intent as a list of expectations with each expectation containing the requirements goals and constraints to be achieved. The expectation is defined to contain 2 attributes - the imm:target and the imm:params. On the hand, the intentExpectation defined in 3GPP (see clause 6.2.1.2.2) contain more attributes some of which (the expectionObject, expectionObjectContexts, expectationTargets and expectationContexts)can be mapped to the TM Forum model.

Following the table to illustrate the attributes mapping between 3GPP Intent Expectation and TM Forum IntentExpectation

|  |  |
| --- | --- |
| **3GPP Intent Expectation** | **TM Forum Intent Expectation** |
| **Class Property** | **Attribute** |
| expectionObject | imm:target |
| expectionObjectContexts | imm:params |
| expectationTargets |
| expectationContexts |

|  |
| --- |
| **End of Modified Sections** |