**3GPP TSG-SA5 Meeting #157S5-245401**

**14 - 18 October 2024, Hyderabad, India**

**Source: Nokia, Nokia Shanghai Bell**

**Title:** **pCR 28.914 Conclusion on negotiation and utility functions**

**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 28.914-101 “Study on intent driven management services for mobile networks phase 3”.

# 3 Rationale

TR28.867 has study several use cases on closed control loops. This pCR is to add conclusions and recommendations for normative work

# 4 Detailed proposal

|  |
| --- |
| **Start of modification** |

## 5.13 Use case #13: Utility function support

### 5.13.1 Description

For some intents, it may not be obvious to the MnS producer how to select from multiple available solutions for fulfilling an intent. In other cases, conflicts may arise for an intent, for which the MnS producer may require extra information from the MnS consumer to decide how to resolve the conflicts. Also, the MnS consumer may not be satisfied with the fulfillment achieved by the MnS producer, for which extra information provided by the MnS consumer can assist the MnS producer in providing better fulfillment. The extra information indicates the preference for the expectation targets. It may be in form of an Intent Utility Function, which enables the MnS consumer to express the relative value of their expectation targets to assist the IDMS producer(s) in fulfilling their intents in the most acceptable manner, or it may be an index that provides an indicator to the MnS Producer of how satisfied it is with the intent fulfilment, e.g. the utility corresponding to a specific target value.

Intent Utility Function defines a method by which consumers can express the relative value of an intent's expectations to assist the IDMS producer(s) in fulfilling their intents in the most acceptable manner.

Intent utility functions are mathematical expressions that quantify the satisfaction or utility derived from the various degrees of fulfilment. The basic components of which include:

- Variables: to quantify specific aspects of the fulfilment, e.g. network performance.

- Weights: to define the relative importance of each variable, e.g. for network performance a variable representing low latency might be assigned higher weight than throughput.

- Function: the mathematical functions to be applied to the variables, e.g. linear, logarithmic, polynomial applied to the variables.

- Result: the output of the function. The value of which represents the utility level achieved, i.e. the satisfaction of the current fulfilment based on the acceptability of potential outcomes defined by consumer.

Intent producer(s) can use such utility information to assess the acceptability of potential outcomes, in addition to information such as resource availability and performance targets. Utility functions may be defined by the consumer and provided as part of the intent itself, i.e. the function is defined as part of the intent. A consumer may also specify that an existing (i.e. predefined) utility function is used. Predefined utility functions may be vendor specified and/or specified by consumer.

### 5.13.2 Potential requirements

**REQ-Intent\_Util-1:** The intent driven MnS producer should have the capability to advertise its support for allowing MnS Consumers to express relative value.

**REQ-Intent\_Util-2:** The intent driven MnS producer should have the capability to advertise the methods by which an MnS Consumer can express relative value.

**REQ-Intent\_Util-3:** The intent driven MnS Producer should have the capability to allow an MnS consumer to express the relative value of its requirements within an intent.

**REQ-Intent\_Util-4:** The intent driven MnS producer should have the capability to report potential outcomes, with the impacts to the related ExpectationObject, including consideration for MnS Consumer relative value, when applicable to an intent.

**REQ-Intent\_Util-5:** The intent driven MnS producer should allow a consumer to specify the relative value as part of an intent.

**REQ-Intent\_Util-6:** The intent driven MnS producer should allow a consumer to apply an existing relative value, defined external to the intent.

### 5.13.3 Potential solutions

A key consideration for potential solutions is the extent to which utility functions must be modelled in the solution to support the above requirements. The flexibility in defining the utility function(s) themselves, the granularity of applying them to different parts of the intent model, and the level of reporting required all affect the potential solutions.

#### 5.13.3.1 Potential solution #1 - Utility function support

This solution proposes support for utility function(s) be added in a relatively simple manner. This is via new attributes added to the intent common model to express intent utility and report the impact of utility on intent fulfilment.

The goal of this solution is simple implementation and vendor flexibility. The new attributes are defined simply as strings and are left to vendor-definition.

A new attribute utilityFunction is added to allow consumers to express relative business value within intents.

Utility functions can be defined at various levels within the intent:

- For the entire intent or a set of expectations in the intent based on criteria. The Intent IOC is updated to support such utility functions.

- For a specific expectation, the IntentExpectation datatype is updated to support such utility functions.

A method to report the impact of utility function(s) on the fulfilment is required. The ExpectationFulfillmentResult is updated to allowing reporting of the evaluation of the utility function(s).

Potential updates to 3GPP TS 28.312 [2] are show below, using clause numbers and headers from it with modified text in **bold**:

*"6.2.1.2* Class definition

*6.2.1.2.1 Intent <<InformationObjectClass>>*

*6.2.1.2.1.1 Definition*

*This IOC represents the properties of an Intent driven management information between MnS consumer and MnS producer.*

*The* *Intent IOC contains one or multiple* *IntentExpectation(s) which includes MnS consumer's requirements, goals and contexts given to a 3GPP system.*

*The Intent IOC also contains intentAdminState to support intent lifecycle management. In case MnS consumer wants to suspend an intent, MnS consumer can request MnS producer to configure attribute intentAdminState with the value "DEACTIVATED". A suspended intent means this intent is not considered for fulfillment. In case MnS consumer wants to resume an intent on the MnS producer side when the intent is suspended, MnS consumer can request MnS producer to configure attribute intentAdminState with the value "ACTIVATED".*

*The attribute "observationPeriod" indicates the time period for which the fulfilment process is observed and at the end of which the fulfilmentInfo for corresponding ExpectationTargets, IntentExpectations and Intent is updated. The observation period can be set by the MnS consumer or by the MnS producer if the MnS consumer does not provide a value.*

*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.*

*The Intent IOC includes contextSelectivity respectively used to define how to select among the stated intentContexts*

***The utilityFunction is used to specify a utility function for the Intent.***

***The utilityFunctionRef is used to reference a utility function for the Intent, which is specified elsewhere.***

*6.2.1.2.1.2 Attributes*

*The* *Intent IOC includes attributes inherited from* *Top IOC (defined in 3GPP TS 28.622) and the following attributes.*

*Table 6.2.1.2.1.2-1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Attribute Name* | *Support Qualifier* | *isReadable* | *isWritable* | *isInvariant* | *isNotifyable* |
| *intentExpectations* | *M* | *T* | *T* | *F* | *F* |
| *userLabel* | *M* | *T* | *T* | *F* | *F* |
| *contextSelectivity* | *O* | *T* | *T* | *F* | *F* |
| *intentContexts* | *O* | *T* | *T* | *F* | *F* |
| *observationPeriod* | *O* | *T* | *T* | *F* | *F* |
| *intentPriority* | *O* | *T* | *T* | *F* | *T* |
| *intentAdminState* | *CM* | *T* | *T* | *F* | *F* |
| *intentPreemptionCapability* | *CM* | *T* | *T* | *F* | *F* |
| ***utilityFunction*** | ***CM*** | ***T*** | ***T*** | ***F*** | ***F*** |
| ***utlityFunctionRef*** | ***CM*** | ***T*** | ***T*** | ***F*** | ***F*** |
| ***Attribute related roles*** | | | | | |
| *intentReportReference* | *M* | *T* | *F* | *F* | *F* |

*6.2.1.2.1.3 Attribute constraints*

|  |  |
| --- | --- |
| *Name* | *Definition* |
| *intentAdminState*  *Support Qualifier* | *Condition: MnS consumer-suspension mechanism is supported.* |
| *intentPreemptionCapability*  *Support Qualifier* | *Condition: The pre-emption mechanism is supported.* |
| ***utilityFunction*** | ***Condition: Intent utility function definition capability is supported.*** |
| ***utilityFunctionRef*** | ***Condition: Intent utility function reference capability is supported.*** |

*6.2.1.2.1.4 Notifications*

*The common notifications defined in clause 6.2.1.5 are valid for this IOC. In addition, the following set of notifications is also valid.*

| *Name* | *S* | *Notes* |
| --- | --- | --- |
| *notifyMOIChanges* | *M* | *--* |

*6.2.1.3 DataType definition*

*6.2.1.3.1 IntentExpectation <<dataType>>*

*6.2.1.3.1.1 Definition*

*IntentExpectation <<dataType>>represents MnS consumer's requirements, goals and contexts given to a 3GPP system.*

*The IntentExpectation <<dataType>> includes contextSelectivity used to define how to select among the stated expectationContexts.*

***The IntentExpectation <<dataType>> includes utilityFunction used to optionally define the business value of the stated*** *expectationTargets.*

*6.2.1.3.1.2 Attributes*

*The* *IntentExpectation includes the following attributes.*

*Table 6.2.1.3.1.2-1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Attribute Name* | *Support Qualifier* | *isReadable* | *isWritable* | *isInvariant* | *isNotifyable* |
| *expectationId* | *M* | *T* | *T* | *T* | *T* |
| *expectationVerb* | *O* | *T* | *T* | *T* | *F* |
| *expectationObject* | *M* | *T* | *T* | *F* | *F* |
| *expectationTargets* | *M* | *T* | *T* | *F* | *F* |
| *contextSelectivity* | *O* | *T* | *T* | *F* | *F* |
| ***utilityFunction*** | ***O*** | ***T*** | ***T*** | ***F*** | ***F*** |
| *expectationContexts* | *O* | *T* | *T* | *F* | *F* |
| *NOTE: The scenariospecific IntentExpectations in clause 6.2.2 are defined utilizing the constructs of this generic IntentExpectation <<dataType>>.* | | | | | |

*6.2.1.3.1.3 Attribute constraints*

*None.*

*6.2.1.3.7 ExpectationFulfilmentResult <<dataType>>*

*6.2.1.3.7.1 Definition*

*ExpectationFulfilmentResult <<dataType>> includes the expectationFulfilmentInfo and targetFulfilmentResults for each IntentExpectation.*

*The expectationFulfilmentInfo describes status of fulfilment of an intentExpectation and the related reasons for the infeasible status.*

***The utilityFunctionResult describes the impact of the utility function on the fulfilment.***

*6.2.1.3.7.2 Attributes*

*The ExpectationFulfilmentResult includes the following attributes.*

*Table 6.2.1.3.7.2-1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Attribute Name*** | *Support Qualifier* | *isReadable* | *isWritable* | *isInvariant* | *isNotifyable* |
| *expectationId* | *M* | *T* | *F* | *T* | *T* |
| *expectationFulfilmentInfo* | *M* | *T* | *F* | *F* | *T* |
| *targetFulfilmentResults* | *O* | *T* | *F* | *F* | *T* |
| ***utilityFunctionResult*** | ***O*** | ***T*** | ***T*** | ***F*** | ***F*** |

*6.2.1.3.7.3 Attribute constraints*

*None.*

*6.2.1.4 Attribute definition*

*Table 6.2.1.4-1*

| *Attribute Name* | *Documentation and Allowed Values* | *Properties* |
| --- | --- | --- |
| *utilityFunction* | *Logical expression of a utility function.*  *The syntax and capabilities of utilityFunction are vendor specified.*  *An empty string is not allowed.*  *allowedValues: N/A* | *type: String*  *multiplicity: 0..1*  *isOrdered: N/A*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *utilityFunctionResult* | *The result of the evaluation of a utility function.*  *The syntax and values are vendor specified.*  *allowedValues: Not Applicable* | *type: String*  *multiplicity: 0..1*  *isOrdered: N/A*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |

*"*

#### 5.13.3.2 Potential solution #2 - Utility function support

The proposal enhances solution #1 by defining specific aspects of the utility function which the consumer can express as input.

The goals with this approach are to better support multi-vendor interoperability and interaction with intent handling functions in other non-3gpp management domains.

NOTE: For this solution the detailed definitions of the arguments, operations, and functions are not included in the input.

Potential updates to 3GPP TS 28.312 [2] are show below, using clause numbers and headers from it with modified text in **bold**:

*"6.2.1.3.x UtilityFunction <<dataType>>*

*6.2.1.3.x.1 Definition*

*The* *UtilityFunction <<IOC>> represents a utility function instance.*

*This representation includes attributes to support a utility function, result, and/or error information. The function is a series of input arguments and operations defined as ordered lists.*

*The attribute functionDefinitionInput allows the consumer to select the utility function to be used (e.g. function name, function definition) as supported by the vendor. The semantics of functionDefinition are vendor defined.*

*The attributes argumentName and argumentWeight can be used by the consumer to indicate relative weights of specified arguments to be used in the utility function calculation.*

*The attribute function is used by the producer to provide the definition of the utility function based on the inputs. E.g. the function defined/selected by consumer.*

*The attribute error is used by the producer to indicate error(s) with the definition of the utility function. E.g. invalid specification in the functionDefinitionInput.*

*6.2.1.3.x.2 Attributes*

*UtilityFunction includes the following attributes:*

*Table 6.2.1.3.4.2-1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Attribute Name* | *Support Qualifier* | *isReadable* | *isWritable* | *isInvariant* | *isNotifyable* |
| *argumentName* | *O* | *T* | *T* | *F* | *F* |
| *argumentWeight* | *O* | *T* | *T* | *F* | *F* |
| *function* | *O* | *T* | *F* | *F* | *F* |
| *error* | *O* | *T* | *F* | *F* | *F* |
| *functionDefinitionInput* | *O* | *T* | *T* | *F* | *F* |

Editor's Note: additional attributes to allow consumer to provide producer additional utility information (e.g. target utility function result) may be added.

*6.2.1.3.x.3 Attribute constraints*

*None.*

*6.2.1.4 Attribute definition*

*Table 6.2.1.4-1*

| *Attribute Name* | *Documentation and Allowed Values* | *Properties* |
| --- | --- | --- |
| *argumentName* | *An ordered list which contains one entry per argument.*  *allowedValues: a defined expectationName.targetName in the intent*  *Editor's Note: This definition likely has a dependency on the note in 3GPP TR 21.905 [1], clause 6.2.2.1.3.3 as measurements/KPI will need to be referenced in the functions.* | *type: String*  *multiplicity: 1..\**  *isOrdered: True*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *argumentWeight* | *Relative weight of the associated argument.*  *Default value is 1.*  *allowedValues: value between 0 and 1.* | *type: Real*  *multiplicity: 1..\**  *isOrdered: True*  *isUnique: N/A*  *defaultValue: 1*  *isNullable: False* |
| *function* | *The function to be used to calculate the utility of the intent fulfilment.*  *E.g. may comprise the combination of the specified utility function input, plus any addition arguments (e.g. weights).expressed as a mathematical function.*  *E.g. may indicate the name of the specified utility function to be used* | *type: String*  *multiplicity: 1..\**  *isOrdered: True*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *error* | *Error string.*  *allowedValues: N/A* | *type: String*  *multiplicity: 0..1*  *isOrdered: False*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *functionDefinitionInput* | *String representation of a utility function.*  *The syntax and evaluation of the string are vendor defined.*  *An empty string is not allowed.*  *allowedValues: N/A* | *type: String*  *multiplicity: 0..1*  *isOrdered: N/A*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |

*"*

#### 5.13.3.3 Potential solution #3 - Utility function support

Intent Handlers, including an SA5 IDMS Producer, deployed in a multi-domain deployment with domain specific Intent Handlers would benefit from a consistent format for expressing utility functions. For example, TMF defined intents available in the deployment defined in 3GPP TS 28.312 [2], F.3 "Management interactions for Intent-CSC between CSC and CSP" would define utility function in RDF.

This proposed solution emphasizes close alignment with the functional ontology defined in TM Forum TR291I [13].

This could be achieved by modifying potential solution #2 to align with functional definitions in TM Forum TR292H [14], clause 11. Some attributes (e.g. arityMin, artiyMax) are excluded as those are not required.

In addition to the attributes defining the function, the attribute sourceDefinition maintains the original input in its native format.

NOTE: This approach could be further defined to support additional functions such as those defined in TM Forum TR292H [14] and TM Forum TR292C [15]. The mapping of expectations and targets current defined in 3GPP TS 28.312 [2], Annex C may also require update to better align with latest TMF spec versions and to ensure it is sufficient for the argument/property mappings.

Potential updates to 3GPP TS 28.312 [2] are show below, using clause numbers and headers from it with modified text in **bold**:

"*6.2.1.3.x UtilityFunction <<dataType>>*

*6.2.1.3.x.1 Definition*

*The UtilityFunction <<dataType>> represents a utility function.*

*This representation includes attributes to support a utility function, result, and/or error information. The attribute nativeRepresentation provides the utility function in its native format.*

*6.2.1.3.x.2 Attributes*

*UtilityFunction includes the following attributes:*

*Table 6.2.1.3.4.2-1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Attribute Name* | *Support Qualifier* | *isReadable* | *isWritable* | *isInvariant* | *isNotifyable* |
| *argumentNames* | *O* | *T* | *T* | *F* | *F* |
| *argumentTypes* | *O* | *T* | *T* | *F* | *F* |
| *Error* | *O* | *T* | *F* | *F* | *F* |
| *operation* | *O* | *T* | *F* | *F* | *F* |
| *function* | *O* | *T* | *F* | *F* | *F* |
| *Result* | *O* | *T* | *F* | *F* | *F* |
| *resultType* | *O* | *T* | *F* | *F* | *F* |
| *nativeRepresentation* | *M* | *T* | *T* | *F* | *F* |
| *nativeFormat* | *M* | *T* | *T* | *F* | *F* |

*6.2.1.3.x.3 Attribute constraints*

*None.*

*6.2.1.4 Attribute definition*

*Table 6.2.1.4-1*

| *Attribute Name* | *Documentation and Allowed Values* | *Properties* |
| --- | --- | --- |
| *utilityFunction* | *Logical expression of a utility function.*  *allowedValues: N/A* | *type: UtilityFunction*  *multiplicity: 0..\**  *isOrdered: N/A*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *argumentNames* | *An ordered list which contains the function arguments.*  *allowedValues: a defined expectationName.targetName in the intent*  *Editor's Note: This definition likely has a dependency on the note in 3GPP TR 21.905 [1], clause 6.2.2.1.3.3 as measurements/KPI will need to be referenced in the functions.* | *type: String*  *multiplicity: 1..\**  *isOrdered: True*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *argumentTypes* | *An ordered list which contains one entry per associated argumentName.*  *allowedValues: objectType of the argumentName* | *type: String*  *multiplicity: 1..\**  *isOrdered: True*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *operation* | *An ordered list which contains the function operations.*  *allowedValues: PLUS, MINUS, MULTIPLY\_BY, DIVIDE\_BY, LOG, MIN, MAX, MEAN* | *type: Enum*  *multiplicity: 1..\**  *isOrdered: True*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *function* | *The mathematical function. Comprises the combination of the list of arguments (\* by their weight) and list of operations defined for the utility function.* | *type: String*  *multiplicity: 1..\**  *isOrdered: True*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *result* | *Result of the function evaluation.*  *allowedValues: N/A* | *type: Real*  *multiplicity: 0..1*  *isOrdered: False*  *isUnique: N/A*  *defaultValue: N/A*  *isNullable: False* |
| *resultType* | *Type of the result.*  *allowedValues: <Whether this attribute is needed, and what values it would allow depends on which functions are supported>* | *type: Enum*  *multiplicity: 0..1*  *isOrdered: False*  *isUnique: N/A*  *defaultValue: N/A*  *isNullable: False* |
| *error* | *Error string.*  *allowedValues: N/A* | *type: String*  *multiplicity: 0..1*  *isOrdered: False*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *nativeRepresentation* | *String representation of a utility function in its native format.*  *An empty string is not allowed.*  *allowedValues: N/A* | *type: String*  *multiplicity: 0..1*  *isOrdered: N/A*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |
| *nativeFormt* | *The format of the natively defined utility function.*  *An empty string is not allowed.*  *allowedValues: N/A* | *type: String*  *multiplicity: 0..1*  *isOrdered: N/A*  *isUnique: N/A*  *defaultValue: None*  *isNullable: False* |

*"*

#### 5.13.3.4 Potential solution #4 - Utility function capability support

An Intent MnS Producer should advertise support for utility functionality (REQ-Intent\_Util-1) and the available support for expressing utility functions (REQ-Intent\_Util-2).

This can be achieved using existing solutions by adding the utility function capabilities to the 'intentHandlingCapabilityList' and allowing consumer to query the capabilities as defined in 3GPP TS 28.312 [2], clause E.2.1.

#### 5.13.3.5 Potential solution #5 - Intent satisfaction index

This solution proposes support for an MnS Consumer "satisfaction index" used by an Intent MnS Consumer to provide an indicator to the Intent MnS Producer of how satisfied it is with the intent fulfilment.

Potential updates to 3GPP TS 28.312 [2] are show below, using clause numbers and headers from it with modified text in **bold**:

*"6.2.1.2 Class definition*

*6.2.1.2.1 Intent <<InformationObjectClass>>*

*6.2.1.2.1.1 Definition*

*This IOC represents the properties of an Intent driven management information between MnS consumer and MnS producer.*

*The Intent IOC contains one or multiple IntentExpectation(s) which includes MnS consumer's requirements, goals and contexts given to a 3GPP system.*

*The Intent IOC also contains intentAdminState to support intent lifecycle management. In case MnS consumer wants to suspend an intent, MnS consumer can request MnS producer to configure attribute intentAdminState with the value "DEACTIVATED". A suspended intent means this intent is not considered for fulfillment. In case MnS consumer wants to resume an intent on the MnS producer side when the intent is suspended, MnS consumer can request MnS producer to configure attribute intentAdminState with the value "ACTIVATED".*

*The attribute "observationPeriod" indicates the time period for which the fulfilment process is observed and at the end of which the fulfilmentInfo for corresponding ExpectationTargets, IntentExpectations and Intent is updated. The observation period can be set by the MnS consumer or by the MnS producer if the MnS consumer does not provide a value.*

*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.*

*The Intent IOC includes contextSelectivity respectively used to define how to select among the stated intentContexts.*

*The Intent IOC includes* ***fulfillmentSatisfactionIndex*** *that enabes the MnS consumer to express their relative satisfaction for a given solution provided by the MnS producer.. The* ***fulfillmentSatisfactionIndex*** *is the computation of the utility achieved by a solution provided by the MnS producer according to the MnS consumer’s utility function.*

***The attribute fulfilmentSatisfactionIndex is used by MnS Consumer to indicate the level of satisfaction with the fulfilment.***

*6.2.1.2.1.2 Attributes*

*The Intent IOC includes attributes inherited from Top IOC (defined in 3GPP TS 28.622) and the following attributes.*

*Table 6.2.1.2.1.2-1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Attribute Name* | *Support Qualifier* | *isReadable* | *isWritable* | *isInvariant* | *isNotifyable* |
| *intentExpectations* | *M* | *T* | *T* | *F* | *F* |
| *userLabel* | *M* | *T* | *T* | *F* | *F* |
| *contextSelectivity* | *O* | *T* | *T* | *F* | *F* |
| *intentContexts* | *O* | *T* | *T* | *F* | *F* |
| *observationPeriod* | *O* | *T* | *T* | *F* | *F* |
| *intentPriority* | *O* | *T* | *T* | *F* | *T* |
| *intentAdminState* | *CM* | *T* | *T* | *F* | *F* |
| *intentPreemptionCapability* | *CM* | *T* | *T* | *F* | *F* |
| ***fulfillmentSatisfactionIndex*** | ***CM*** | ***T*** | ***T*** | ***F*** | ***F*** |
| ***Attribute related roles*** | | | | | |
| *intentReportReference* | *M* | *T* | *F* | *F* | *F* |

*6.2.1.2.1.3 Attribute constraints*

|  |  |
| --- | --- |
| *Name* | *Definition* |
| *intentAdminState*  *Support Qualifier* | *Condition: MnS consumer-suspension mechanism is supported.* |
| *intentPreemptionCapability*  *Support Qualifier* | *Condition: The pre-emption mechanism is supported.* |
| ***fulfillmentSatisfactionIndex*** | ***Condition: The intent handler supports receiving satisfaction information form the MnS consumer*** |

*6.2.1.2.1.4 Notifications*

*The common notifications defined in clause 6.2.1.5 are valid for this IOC. In addition, the following set of notifications is also valid.*

| *Name* | *S* | *Notes* |
| --- | --- | --- |
| *notifyMOIChanges* | *M* | *--* |

*6.2.1.4 Attribute definition*

*Table 6.2.1.4-1*

| *Attribute Name* | *Documentation and Allowed Values* | *Properties* |
| --- | --- | --- |
| ***fulfillmentSatisfactionIndex*** | ***It indicates the MnS consumer's evaluation of degree to which the fulfilment satisfies the MnS consumer's requirements. It is the computation of the utility achieved by a solution provided by the MnS producer according to the MnS consumer’s utility function. It is an integer in the range  [0,100] . The highest possible value indicates that the solution provided by the MnS producer achieves the best possible outcomes that the MnS consumer expected, e.g., that it achieves the highest range of a target whose desired values were defining as falling in a range. Correspondingly, the values lower than 100 indicate how far from the maximum satisfaction the outcome is.***  ***allowedValues: integers in the range [0,100]*** | ***type: Integer***  ***multiplicity: 1***  ***isOrdered: N/A***  ***isUnique: N/A***  ***defaultValue: None***  ***isNullable: False*** |

*"*

5.13.4 Evaluation of potential solutions

**Potential Solution 1:** Solution 1 is already covered by Solution 2. It is not recommended as a separate solution, but as part of Solution 2.

**Potential Solution 2:** This solution could be the basis on which to define a solution, with some further updates. Recommendation is to implement this as a new IOC (not datatype) to allow Intent MnS Consumer to provision utility function instances (REQ-Intent\_Util-3). These instances would be referenced (not contained) by the Intents to which this utility function is to be applied (REQ-Intent\_Util-5), allowing the same intent utility function instance to be referenced from multiple intents (REQ-Intent\_Util-6). The proposed attributes 'result' and 'error', along with calculated values, are added to the respective intent reports (REQ-Intent\_Util-4). The other proposed attributes require further discussion.

**Potential Solution 3:** It is unclear that the intent functions from another domain, and in other formats than those defined by SA5, would be of use to IDMS MnS. This solution is not recommended as a result.

**Potential Solution 4:** This proposal describes enhancements required for IDMS to advertise its support for utility functions (REQ-Intent\_Util-1) along with details about the supported methods (REQ-Intent\_Util-2). Details on how to express the specific capabilities (i.e. which functions are supported, and how) requires further discussion. This item has a dependency on the modelling to support provisioning instances of the utility functions.

**Potential Solution 5:** The solution provides the computation of the utility achieved by a solution provided by the MnS producer according to the MnS consumer’s utility function. It is an integer in the range  [0,100] . The highest possible value indicates that the solution provided by the MnS producer achieves the best possible outcomes that the MnS consumer expected, e.g., that it achieves the highest range of a target whose desired values were defining as falling in a range. Correspondingly, the values lower than 100 indicate how far from the maximum satisfaction the outcome is

It is recommended to support the combination of solution 2 and 5 to enable the MnS consumer to express their relative preference via a utility function according to solutions 2 and 4 and expressing the relative satisfaction of a given solution via the satisfaction index according to solution 5.

|  |
| --- |
| **Next modifications** |

## 5.16 Use case #16: Negotiation on fulfilment of intents

### 5.16.1 Description

In 3GPP TS 28.312 [2] Release 18, the intent fulfilment feasibility check is already supported. When an intent-driven MnS producer receives the intent instance creation or modification request, it automatically conducts a feasibility check to determine whether the intent instance is feasible. If the feasibility check result is feasible, the MnS producer performs the service or network management tasks to satisfy the intent instance.

However, the MnS Producer cannot decide the content of the intent it receives, and the MnS Consumer may express a feasible intent, but the producer has multiple ways to fulfil it. One of these solutions might be better in one aspect and another solution might be better in another aspect. Although this intent is feasible, the producer may not have the knowledge and capability to determine which solution is appropriate. For example, the MnS Consumer may express intent for energy saving but only specify the target for energy consumption reduction without more constraints, e.g. reduce 20 % energy consumption. There will be numerous solutions to fulfil the energy consumption target with various impacts, e.g. one solution may reduce the coverage and one may reduce the user number. The consumer may have concerns on the possible impact of the selected solution by the Producer, which means that negotiation with the MnS Consumer is necessary. In such cases, the MnS consumer and Mns producer may negotiate on the best way to fulfil the intent. For example, through an MnS capability enabling the MnS producer to provide feedback (e.g. possible solutions, possible outcomes, and possible impacts) to the Mns consumer allowing the MnS consumer to select the appropriate solution.

There are multiple negotiations that can happen for an intent that is feasible, many of them employing interaction that are similar.

NOTE: Although some of these may be applicable during the feasibility check process, they are considered part of intent negotiations and not part of feasibility checking.

#### 5.16.1.1 Checking for fulfillable outcomes

The MnS consumer wants to know the possible fulfillable outcomes for a given intent. The MnS consumer creates an intent that should be evaluated by the MnS producer be see what the MnS producer can deliver.



Figure 5.16.1.1-1: MnS consumer requests and receives a list of fulfillable outcomes on an intent.

Subsequently, the MnS producer provides a report indicating what is fulfillable for each intent aspect (intentExpectation and expectationTarget) within that intent. Since different fulfillable outcomes may have different impacts, the report should include the relative impacts of each outcome.

#### 5.16.1.2 Checking for best possible outcome on an intent, intent expectation, or expectation target

The MnS consumer wants to know the best possible outcome for a given intent or intent expectation or expectation target. This could be prior to or during fulfilment.



Figure 5.16.1.2-1: Checking for best possible outcome on intent or  
intent expectation or expectation target

The MnS consumer creates an intent that should be evaluated by the MnS producer with a request to provide the best possible outcome. Subsequently, the MnS producer provides a report indicating that best possible outcome. The best possible outcome is defined as follows:

- The request is to evaluate an intent with only 1 expectation target: The best possible outcome is the best value on that expectation target that does not adversely affect other aspects of the network.

- The request is to evaluate expectation target in an intent with multiple expectation targets (e.g. multiple expectations or one expectation with multiple expectation targets): The best possible outcome is the best value on that expectation target that maintains the other expectation targets to within the ranges specified in the intent and does not adversely affect other expectation targets or aspects of the network.

- The request is to evaluate all expectation targets in an intent with multiple expectation targets (e.g. multiple expectations or one expectation with multiple expectation targets): The best possible outcome is the best value on each expectation target that maintains the other expectation targets to within the ranges specified in the intent and does not adversely affect other aspects of the network.

The MnS producer should support an fulfillable outcomes report that lists the fulfillable outcomes for any of the three scenarios, the report including the related impact on other targets in the intent or on other metrics and contexts.

#### 5.16.1.3 MnS producer to provide information about possible fulfillment of the intent

The MnS consumer wants to know what the MnS producer recommends what to be applied for particular intent characteristics. The MnS consumer creates an intent and asks the MnS producer to recommend what changes could be made to the intent or other intents to make the intent fulfillable (Figure 5.16.1.3-1). Alternatively, the MnS producer has attempted to fulfil the intent and indicated that it cannot be fulfilled, so the MnS consumer asks the MnS producer to recommend what changes could be made to the intent or other intents to make the intent fulfillable.



Figure 5.16.1.3-1: Enabling the MnS consumer to request and receive a recommendation on  
the fulfillable intent properties prior to fulfilment



Figure 5.16.1.3-2: Enabling the MnS producer s in case of inability to fulfil an intent to provide to the MnS consumer information about possible fulfillment of the intent

Subsequently, the MnS producer provides a intent report indicating the information about possible fulfillment of the intent. The MnS producer should support a recommended-changes report that lists different information about possible fulfillment of the intent, e.g., candidate changes to unfulfillable targets within an unfulfillable expectation.

NOTE: The nature of the report and what can be included if FFS.

#### 5.16.1.4 MnS consumer advises on preferred alternatives

The MnS consumer wants an intent fulfilled. The intent is feasible, but the MnS producer has multiple alternatives related to fulfilling the intent. The MnS producer wants the MnS consumer to advise on their (the MnS consumer's) preference among these alternatives.

NOTE 1: An alternative is the combination a set of expectation target values that the MnS producer can achieve together with their (expected) impacts on the network (objects). E.g. for an expectation target on energy consumption, the impact may include which cells could be deactivated, or which other intents (e.g. coverage related intents) could be affected.



Figure 5.16.1.4-1: MnS consumer advises the MnS producer  
on the preferences among alternatives at the MnS producer

After the MnS consumer creates an intent to be fulfilled, the MnS producer determines that there are multiple alternatives, so the MnS producer and provides a report to the MnS consumer so that the MnS consumer may help chose the best alternative.

The report to the MnS consumer may include:

- The list of available/fulfillable expectation target values that the MnS producer is able to apply/achieve.

- The expected relative impacts of the different alternatives - on aspects of the submitted intent or other intents and intent expectations.

- A request to the MnS consumer to select one among the alternatives.

Given the alternatives, the MnS consumer takes any combination of these actions:

- Chooses and indicates the preferred alternative.

- Defines the relative importance of their expectation Targets (in form of a utility function) so that the MnS producer may consider these in deciding upon the solution/ solution approach/ closed loops/ action/ outcome to be applied/deployed/achieved.

- Provides the relative satisfaction level that each of the alternatives achieves. The satisfaction level is the computation of the utility achieved by each alternative according to the MnS consumer’s utility function. It is an integer in the range  [0,100] . The highest possible value indicates that the solution provided by the MnS producer achieves the best possible outcomes that the MnS consumer expected, e.g., that it achieves the highest range of a target whose desired values were defining as falling in a range. Correspondingly, the values lower than 100 indicate how far from the maximum satisfaction the outcome is.

#### 5.16.1.5 MnS producer requests for extra information to be used to select another alternative post initial fulfilment

The MnS consumer wants an intent fulfilled and the MnS producer has multiple alternatives related to fulfilling the intent. After the MnS consumer creates an intent to be fulfilled, the MnS producer independently choses the alternative to be applied. But on realizing that they cannot achieve better outcomes, MnS producer allows the MnS consumer to provide extra information that guarantee better satisfaction (see Figure 5.16.1.5-1). The MnS producer reports the fulfillment outcomes (imperfect fulfillment) and indicates to the MnS consumer that if the MnS consumer is unsatisfied with the outcomes, the MnS consumer should provide extra information to help select a better alternative.

The report to the MnS consumer may include one or more of the following:

- The list of available/ fulfillable expectation target values which can guide the MnS consumer when providing the preference policy.

- The relative impacts of the different alternatives - on the submitted intent or on other intents and intent expectations or on the network.

- A request to evaluate the fulfilment and provide information that could help improve fulfilment.

The extra information provided by the MnS consumer may be one or more of the following:

- A Binary indication that they accept the provided fulfillment or that they do not accept, and another alternative should be tried.

- A utility function indicating the MnS consumer's relative benefits of their expectation Targets. The utility function is the MnS consumer's policy for evaluating of the extent to which they are satisfied with the selected alternative.

- MnS consumer's level of satisfaction which is the evaluation of the extent to which the achieved outcomes match the MnS consumer's expectation as computed from the utility function.

- Changes to the expectation Targets or the relative importance of the expectation Targets to the MnS consumer's objectives.



Figure 5.16.1.5-1: MnS producer requests MnS consumer to provide information  
to help select (better) alternative

### 5.16.2 Potential Requirements

**REQ-Intent\_Negotiation \_01:** The intent-driven MnS should have the capability to enable the MnS producer to provide possible outcomes to the MnS Consumer.

**REQ-Intent\_Negotiation\_02:**The intent-driven MnS should have the capability to allow MnS Consumer decide appropriate outcome for fulfilment based on possible outcomes.

NOTE 1: The definition for outcome is For Further Study.

**REQ\_INT\_NEGOT-1:** The intent driver MnS producer should support a capability enabling an MnS consumer to provide an intent with a request for the MnS producer to provide information on the alternative fulfillable outcomes for an intent.

NOTE 2: An alternative is the combination of a set of expectation target values that the MnS producer can achieve together with their (expected) impacts.

NOTE 3: The impacts refer to information about the changes and outcomes on the expectation objects of the intent and on other related intents from the same intent MnS consumer. The exact characterization of what is reported is For Further Study.

E.g. for an expectation target on energy consumption, the impact may include which cells could be deactivated, or which other intents (e.g. coverage related intents) could be affected. be applied to the intent to make the intent fulfillable.

NOTE 4: Example changes could include: omitting certain intentExpectations and/or expectationTarget(s) or changing the properties of intentExpectations and/or expectationTarget(s).

**REQ\_INT\_NEGOT-2:** The intent driven MnS producer should support a capability to provide to an MnS consumer an intent report indicating the alternatives that the MnS producer can support for the provided intent, intent expectations, or expectation Targets and the expected relative impacts of the different alternatives.

**REQ\_INT\_NEGOT-3:** The intent driven MnS producer should support a capability to request an MnS consumer to indicate its preference among a set of alternatives that the MnS producer can support for the provided intent, intent expectations, or expectation Targets.

**REQ\_INT\_NEGOT-4:** The intent driven MnS producer should support a capability enabling an MnS consumer to provide to the MnS producer information indicating the MnS consumer's preference among alternatives that the MnS producer can support for the provided intent, intent expectations, or expectation Targets.

**REQ\_INT\_NEGOT-5:** The intent driven MnS producer should support a capability enabling an authorized MnS consumer to provide information on a policy that should be used by the be used by the MnS producer to select among the alternatives available at the MnS producer.

NOTE 5: The policy may be provided in the form of a utility function.

**REQ\_INT\_NEGOT-6:** The intent driven MnS producer should support a capability to request the MS consumer to provide an evaluation of the MnS producer's alternatives based on the expected relative impacts of the different alternatives.

**REQ\_INT\_NEGOT-7:** The intent driven MnS producer should support a capability enabling an MnS consumer to provide an evaluation of the MnS producer's alternatives to then be used to select among the alternatives.

**REQ\_INT\_NEGOT-8:** The intent driven MnS producer should support a capability to inform an authorized MnS consumer that an alternative among multiple alternatives has been selected and (will be/has been) applied for the submitted intent.

**REQ\_INT\_NEGOT-9:** The intent driven MnS producer should support a capability to inform an authorized MnS consumer that since no more improvement to intent fulfillment shall be possible the MnS consumer should evaluate the extent to which the applied alternative satisfies the MnS consumer's objectives or provide extra information which can help improve satisfaction.

**REQ\_INT\_NEGOT-10:** The intent driven MnS producer should support a capability enabling an authorized consumer to inform the producer that the alternative selected by the MnS producer was not satisfactory and another alternative should be applied.

**REQ\_INT\_NEGOT-11:** The intent driven MnS producer should support a capability enabling an authorized MnS consumer to provide information on the level of fulfillment which the MnS producer can use to differently attempt the fulfillment.

**REQ\_INT\_NEGOT-12 (already supported):** The intent driven MnS producer should support a capability enabling an authorized MnS consumer to revise the properties of an intent as the means to improve the chances of the intent being fulfillable.

**INT\_NEGOT\_REQ 13:** The MnS producer should support a capability to provide an intent report including information on what is achievable for each intent aspect (intentExpectation and expectationTarget) within that intent and the relative cost/impact of achieving that outcome.

**INT\_NEGOT\_REQ 14:** The MnS producer should support a capability enabling an MnS consumer to provide an intent with a request for the MnS producer to provide the best possible outcome on an intent or intent expectation or expectation target.

**INT\_NEGOT\_REQ 15:** The MnS producer should support a capability to provide an intent report including information on the best possible outcome on intent or intent expectation or expectation target.

Editor's Note: This requirement may be combined with REQ-13.

**INT\_NEGOT\_REQ-16:** The MnS producer should support a capability enabling an MnS consumer to provide an intent with a request for the MnS producer to provide information on what changes could be made to the intent properties or to properties of other intents to make the intent fulfillable.

Editor's Note: Further discussion is needed for this requirement.

**INT\_NEGOT\_REQ-17:** The MnS producer may support a capability to provide a report indicating the changes, which if applied to the intent, would make the intent fulfillable

### 5.16.3 Potential Solutions

This solution proposes to enhance the existing Intent IOC and IntentReport IOC defined in 3GPP TS 28.312 [2] to describe intent negotiation information between MnS producer and MnS consumer during fulfilment phase. e Following are the proposed enhancements:

**Enhancement on Intent IOC:** Introduce attribute " expectedReportType" and corresponding ENUM Value "IntentNegotiationReport" in the Intent IOC to determine the intent negotiation information including the possible outcomes the MnS Consumer can obtain from the MnS Producer). The detailed definition for attribute "expectedReportType" see clause 5.3.3.

1. Introduce a dataType representing the feedback information for the MnS consumer's response to the MnS producer. It may be called IntentNegotiationFeedback. It may contain:

- An attribute for a specific alternative among those indicated by the MnS producer.

- A policy that should be used by the be used by the MnS producer to select among the alternatives. The policy may for example be MnS consumer's utility function.

- An attribute indicating the MnS consumer's satisfaction from a deployed fulfillment as computed form the MnS consumer's utility function:

1. It may contain a list indicating the MnS consumer's expected satisfaction from the different alternatives e.g. as evaluated form the MnS consumer's utility function. This may be a key-value pair where the key is an identifier of the alternative and the value's the MnS consumer's expected satisfaction:

- It may also be used to indicate the true MnS consumer's satisfaction as evaluated for an alternative that has been applied.

- Optionally includes an ENUM attribute indicating the type of extra action information being provided. The values representing the type of extra information could include: IS\_SELECTED\_OPTION\_NOT\_OK, SELECTED\_OPTION, SATISAFACTION\_POLICY

NOTE 1: The fulfillableOutcomesReport and SupportedAlternativesReport will include the impact on the related ExpectationObjects.

**Enhancement on IntentReport IOC:** Introduce an IntentNegotiationReport<<dataType>> as an attribute of Intent Report IOC to represent the MnS producer's information on the negotiations. It may indicate what is fulfillable for an intent, what the MnS producer recommends for an intent or the MnS producer's alternatives from which the MnS consumer may chose. IntentNegotiationReoprt <<dataType>> includes following attributes:

NOTE 2: Whether IntentNegotiationReport<<dataType>> as an attribute of IntentReport IOC or intentFulfilmentReport dataType needs further clarification.

1. PossibleOutcomeList, it indicates the possible or fulfillable outcomes for the intent expectations and expectationTargets within that intent and the relative cost/impact (on the related ExpecationObjects) of achieving that outcome.

NOTE 3: The "PossibleOutcomeList" will include the potential impact on the related ExpectationObjects

1. PossibleOutcomeType, an attribute for what the information indicates, i.e. either:

- The fulfillableOutcomes.

1. RecommendedChanges, it indicates the potential changes to the intent that could make the intent fulfillable.
2. SupportedAlternativesReport. Inclusion of a SupportedAlternativesReport inherently asks the MnS consumer to choose one alternative among those in the SupportedAlternativesReport.
3. ExtraActionInformaton indication, an ENUM attribute indicating need for extra information. The values representing the type of extra information needed could include: SELECT\_FROM\_OPTIONS, IS\_SELECTED\_OPTION\_OK, PROVIDE\_EXTRA\_INFO\_.

### 5.16.4 Evaluation of solutions

The potential solution described in clause 5.16.3 is a fully NRM-based approach that extends the existing NRM fragments to realise the exchanges between the Mns consumer and the MnS producer. The enhancement is small yet comprehensive enough to allow all the different negotiations to be supported. The implementation of this NRM-based solution is straightforward.

Therefore, the solution described in clause 5.16.3 is a feasible solution for negotiation during fulfilment of intents.

However, some aspects of the solution still require to be improved, so only some parts of the use case can be recommended for normative work.

|  |
| --- |
| **Next modifications** |

# 6 Conclusions and recommendations

## 6.1 Use case #1: Enhance the radio service delivering use case

The scenario of delivering a radio service in a scheduled time is introduced in clause 5.1. In this scenario, the intent driven MnS producer for radio service needs to support the capability to allow the MnS consumer expresses the expectations for delivering a radio service in a scheduled time.

The solution proposes an enhancement of the RadioServiceExpectation defined in 3GPP TS 28.312 [2] to add "PLMNInfoContext", "maxNumberofUEsTarget" and "SchedulingTimeContext".

It is recommended to enhance the radio service delivering use case and RadioServiceExpectation model defined in 3GPP TS 28.312 [2] to support radio service delivering scenarios. The detailed solution in clause 5.1.3 is used as baseline for normative work.

## 6.2 Use case #2: Enhancement of radio network expectation

The following scenarios are described in clause 5.2 which requires enhancement of radio network expectation:

- RAN energy saving scenario.

- Radio network traffic assurance for scheduled events scenario.

- Network support for UAV pre-flight preparation.

- Radio Network support MOCN undifferentiated radio service.

It is recommended to enhance the RadioNetworkExpectation and IntentReport defined in 3GPP TS 28.312 [2] to support above identified scenarios by:

- Adding "ulFrequencyContext" and "rATContext" as targetContexts for aveULRANUEThptTarget and aveDLRANUEThptTarget defined in RadioNetworkExpectation.

- Adding "ActiveUEsTarget", "PRBsTarget" and "InterRATHandoverTargets" as ExpectationTargets for RadioNetworkExpectation.

- Adding "SchedulingTimeContext" as ExpectationContext for RadioNetworkExpectation.

- Adding "cellContext" and "timeBasedAreaContext" as ObjectContexts for RadioNetworkExpectation.

- Adding attribute "targetContext" for TargetFulfilmentResult<<dataType>> defined in IntentReport.

- Adding "timeBasedExclusionAreas" as an attribute of IntentFeasibilityCheckReport <<dataType>> and ExpectationFulfilmentResult <<dataType>> defined in IntentReport.

The detailed solution in clause 5.2.2 is used as baseline for normative work.

## 6.3 Use case #3: Implicit intent report subscription with customized requirements

The use case of implicit intent report subscription with customized requirements is described in clause 5.3. In this use case, the intent driven MnS producer needs to support following capabilities:

- MnS consumer who expresses the intent may want to obtain the intent report by default, instead of triggering separate subscription action to subscribe intent report information.

- MnS consumer who express the intent may subscribe the intent report with customized requirements (e.g. condition for intent report).

The solution proposes to introduce the intentReport <<dataType>> as attribute of Intent IOC defined in 3GPP TS 28.312 [2], which includes "reportRecipientAddress", "expectedReportTypes", "reportingConditions", "reportingTargets" and "observationPeriod" to support above new intent report functionalities.

It is recommended to enhance the intent report use case as well as the model to support implicit intent report subscription with customized requirements in 3GPP TS 28.312 [2]. The detailed solution in clause 5.3.3.3 is used as baseline for normative work.

## 6.4 Use case #4: Intent containing an expectation to guarantee specific service characteristics

There is no proposed solution currently defined and it is unclear from the REQs alone which changes would be required to support this functionality.

Further discussion is recommended, including consideration for how this is defined in other SDOs.

## 6.5 Use case #5: Improve intent life cycle documentation

This item is not yet defined sufficiently described to make a recommendation.

## 6.6 Use case #6: Investigate potential impacts to support natural language intents translation

This item is For Further Study.

## 6.7 Use case #7: Enablers for Intent Fulfilment

The use case enablers for Intent Fulfilment is described in clause 5.7. In this use case, the intent driven MnS producer needs to support the capability to allow MnS consumer to subscribe and obtain the information of enablers for intent fulfilment. The solution supports the capability to enable the MnS consumer to request to obtain the information of enablers for intent fulfilment including:

- Enhance the Intent IOC to add an optional attribute "enablerRequest" to describe the MnS consumer requirements for the information of enablers for intent fulfilment.

- Enhance the IntentReport IOC to add an attribute "enablerInfo" in IntentFulfilmentReport <<dataType>> to represent the information of enablers for intent fulfilment.

It is recommended to introduce the use case, requirements and corresponding solution to support intent fulfilment enabler information subscription and obtaining in 3GPP TS 28.312 [2]. The detailed solution in clause 5.7.3 is used as baseline for normative work.

## 6.8 Use case #8: Network support for UAV pre-flight preparation

See clause 6.2.

## 6.9 Use case #9: Enhancement of Radio Network Expectation to support MOCN

See clause 6.2.

6.10 Use case #10: Improve intent handling state management

The Intent handling state diagram and corresponding state transition events are described in clause 5.10.

It is recommended to add intent handling state diagram and intent handling state transition table in 3GPP TS 28.312 [2]. The detailed solution in clause 5.10.2 is used as baseline for normative work.

## 6.11 Use case #11: Enhancement for Intent handling capability

It is recommended to move on to the normative specification development phase for the use case on Extension of Intent handling capability. The normative specification development should a solution that is a combination of the solutions in clauses 5.11.3.1 and 5.13.3.2.

## 6.12 Use case #12: Intent degradation based on expectation preference

The use case, requirements and solutions for Use case #12: Intent Degradation Based on Expectation Preference is described in clause 5.12. In this use case, the intent driven MnS producer is enhanced to allow MnS Consumer to express its preference on expectation for fulfilment in an intent.

The solution proposes to introduce "PreferenceWeight" to the IntentExpectation <<datatype>> and the ExpectationTarget to support the intent degradation based on expectation preference.

It is recommended to start normative work using the detailed solution #1 as the baseline.

## 6.13 Use case #13: Utility function support

It is recommended to enable the MnS consumer to express their relative preference among solutions using a utility function according to a combination of solution 2 and solution 4

It is recommended to enable the MnS consumer to express their relative satisfaction from a given solution provided by the MnS producer using the satisfaction index according to solution 5.

## 6.14 Use case #14: Intent feasibility check

Intent feasibility check capability is introduced for MnS producer to assist the MnS consumer to generate the feasible intent information during pre-evaluation phase. The intent driven MnS producer needs to provide the intent feasibility check result including the list of infeasible expectations and targets. Following enhancements are identified to support intent feasibility check capability:

- Add attribute "intentMgmtPurpose" in Intent IOC for the management purpose (required procedures) of the created or modified intent instance. The allowed values for the new attribute"intentMgmtPurpose" can be FEASIBILITYCHECK, EXPLORE and FULFILMENT.

- Add attribute "inFeasibleExpectationInfos" in the existing IntentFeasibilityCheckReport <<dataType>>. It is the list of InFeasibleExpectationInfo for all infeasible IntentExpectations in the intent.

It is recommended to introduce the use case, requirements and corresponding solution for the intent feasibility check capability during pre-evaluation phase in 3GPP TS 28.312 [2]. The detailed solution in clause 5.14.3 is used as baseline for normative work.

## 6.15 Use case #15: Intent Exploration

The intent exploration capability is introduced for MnS producer to enable MnS consumer to explore the best values for intent targets and contexts within a specific intent during intent pre-evaluation phase to learn more about the MnS producer's capabilities. Following enhancements are identified to support intent exploration capability:

- Add attribute "intentMgmtPurpose" in Intent IOC. The detailed definition for attribute "intentMgmtPurpose" see clause 6.14.

- Add IntentExploreResult <<dataType>> as an attribute of IntentReport IOC to represent the recommended best values for one or multiple targets and contexts in an intent provided by MnS producer.

It is recommended to introduce the use case, requirements and corresponding solution for the intent exploration capability during pre-evaluation phase in 3GPP TS 28.312 [2]. The detailed solution in clause 5.15.3 is used as baseline for normative work.

## 6.16 Use case #16: Negotiation on fulfilment of intents

The present technical report described potential enhancements to the intent driven management specification.

It is recommended to move on to the normative specification development phase for the use case on intent negotiations during the fulfilment of intents. For the normative work:

- 5.16.1.1 is recommended.

- 5.16.1.2 is not recommended - the decision on what is best is subjective. The MnS consumer can use the solution in 5.16.1.1 to get possible alternatives and chose the one to apply.

- 5.16.1.3 is recommended - the producer may provide information about possible fulfillment of the intent.

- 5.16.1.4 is recommended - the consumer is enabled to provide the utility function, satisfaction index or both The satisfaction index represents the consumer’s computed outcome of the utility function.

- 5.16.1.5 is not recommended. It requires further study.

## 6.17 Use case #17: Negotiation on the possible outcomes during the fulfilment phase

See clause 6.16.

## 6.18 Use case #18: Guidelines for using Intent generic information model to support new scenario which is not standardized

The description requirements and solution for Use case #18: Guidelines for using Intent generic information model is described in clause 5.18.

It is recommended to start the normative work to add the guidelines for using intent generic information model in 3GPP TS 28.312 [2] to support new scenario which is not standardized. The detailed solution in clause 5.18.3 is used as baseline for normative work.

## 6.19 Use case #19: Intent driven management for network maintenance

To Be Defined.

## 6.20 Use case #20: Intent containing an expectation for delivering a communication service

It is recommended to start the normative work to define the intent expectation for communication service.

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