ETSI TS 128 312 V19.3.0 (2025-11)

TECHNICAL SPECIFICATION

LTE;
5G;
Management and orchestration;
Intent driven management services for mobile networks
(3GPP TS 28.312 version 19.3.0 Release 19)

Reference
RTS/TSGS-0528312vj30

Keywords
5G,LTE

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI IPR online database.

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found at 3GPP to ETSI numbering cross-referencing.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

shall indicates a mandatory requirement to do something

shall not indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

should indicates a recommendation to do something

should not indicates a recommendation not to do something

may indicates permission to do something

need not indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

can indicates that something is possible

cannot indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

will indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

will not indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

might indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

might not indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

is (or any other verb in the indicative mood) indicates a statement of fact

is not (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

Introduction

The current 5G networks brings more operational complexities, and the telecom system need to be able to adapt their operation to the business objectives of the operator as well as expectations of customer, which is driving customer to shift the focus from "how" to "what". An intent driven system will be able to learn the behaviour of networks and services and allows a customer to provide the desired state, without detailed knowledge of how to get to the desired state. Thus, the intent driven management is introduced to reduce the complexity of management without getting into the intricate detail of the underlying network resources.

1 Scope

The present document specifies concept, use cases, requirements and solutions for the intent driven management for service or network management.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 28.531: "Management and orchestration; Provisioning".

[3] 3GPP TS 28.532: "Management and orchestration; Generic management services".

[4] 3GPP TS 28.530: "Management and orchestration; Concept, use cases and requirements".

[5] 3GPP TS 28.541: "Management and orchestration; 5G Network Resource Model (NRM); Stage 2 and stage 3".

[6] 3GPP TS 28.622: "Telecommunication management; Generic Network Resource Model (NRM); Integration Reference Point (IRP); Information Service (IS)".

[7] TM Forum IG1253A: "Intent Common Model v1.1.0".

[8] 3GPP TS 38.104: "NR; Base Station (BS) radio transmission and reception".

[9] 3GPP TS 28.538: "Management and orchestration; Edge Computing Management".

[10] 3GPP TS 28.658: "Telecommunications management; Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP): Information Service (IS)".

[11] 3GPP TS 28.554: "Management and orchestration; 5G end to end Key Performance Indicators (KPI)".

[12] 3GPP TS 28.552: " Management and orchestration; 5G performance measurements".

[13] 3GPP TS 29.510: " 5G System; Network Function Repository Services".

[14] 3GPP TS 36.104: "Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception "

[15] 3GPP TS 23.003: "Numbering, Addressing and Identification".

[16] Void.

[17] 3GPP TS 28.623: "Telecommunication management; Generic Network Resource Model (NRM) Integration Reference Point (IRP); Solution Set (SS) definitions".

[18] TM Forum TR290B: "Intent Common Model - Intent Reporting v3.6.0".

[19] 3GPP TS 22.125: " Uncrewed Aerial System (UAS) support in 3GPP".

[20] TM Forum TR290A: "Intent Common Model - Intent Expression v3.6.0".

3 Definitions of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

intent: expectations including requirements, goals and constraints given to a 3GPP system, without specifying how to achieve them

intent handling function: a logical function with intent handling capabilities that provides the intent driven management service.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

IDMS Intent Driven Management Service

4 Concepts and Background

4.1 Intent concept

4.1.1 Introduction

An intent specifies the expectations including requirements, goals and constraints for a specific service or network management workflow. In short, an intent is a statement towards a desired/wanted state of a system. The intent may provide information on particular objective and possibly some related details. Following are some general concepts for intent:

- An intent is typically understandable by humans, and also needs to be interpreted by the machine without any ambiguity.
- An intent focuses more on describing the "What" needs to be achieved but less on "How" that outcomes should be achieved, The intent expresses the metrics that need to be achieved and not how to achieve them. This not only relieves the burden of the consumer knowing implementation details but also leaves room to allow the producer to explore alternative options and find optimal solutions. Intent describes the properties that allows a satisfactory outcome.
- The expectations expressed by an intent is agnostic to the underlying system implementation, technology and infrastructure. Area can be used as managed object in the expectations expressed by an intent to achieve system implementation, technology and infrastructure agnostic.

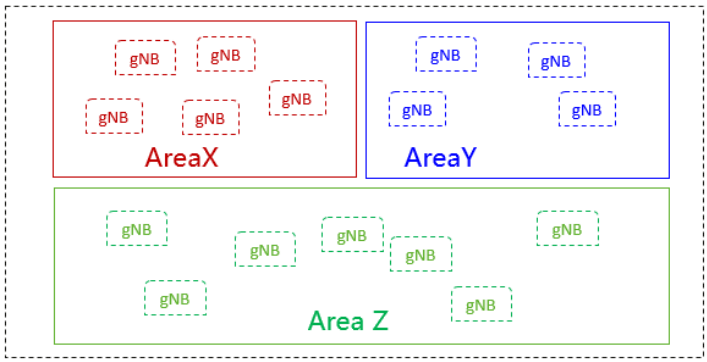


Figure 4.1.1-1

- An intent needs to be quantifiable from network data so that the fulfilment result can be measured and evaluated.

Intent can be categorized based on different user types or different management scenario types.

4.1.2 Intent categorizes based on user types

Based on roles related to 5G networks and network slicing management defined in clause 4.8 in 3GPP TS 28.530 [4], different kinds of intents are applicable for different kinds of standardized reference interfaces.

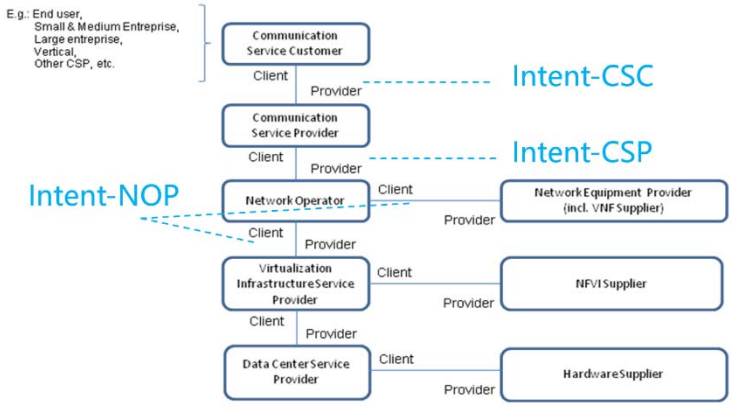


Figure 4.1.2-1: High-level model of different kind of intents expressed by different roles

- **Intent from Communication Service Customer (Intent-CSC):** Intent from Communication Service Customer enables Communication Service Customer (CSC) to express which properties of a communication service the CSC may request from CSP without knowing how to do the detailed management for communication service. For example, Intent-CSC can be 'Enable a V2X communication service for a group of vehicles in certain time'.
- **Intent from Communication Service Provider (Intent-CSP):** Intent from Communication Service Provider enables Communication Service Provider (CSP) to express an intent about what CSP would like to do for network without knowing how to do the detailed management for network. For example, Intent-CSP can be 'Provide a network service supporting V2X communications for highway-417 to support 500 vehicles simultaneously'.
- **Intent from Network Operator (Intent-NOP):** Intent from Network Operator enables Network Operator (NOP) to provide what NOP would like to do for group of network elements (i.e. subnetwork) management and control without knowing how to do the detailed management for the network elements. For example, Intent-NOP can be 'Provide a radio network service to satisfy the specified coverage requirements and UE throughput requirement in certain area'.

The Intent IOC defined in clause 6.2 can be instantiated to represent Intent-CSC, Intent-CSP, and Intent-NOP for intent driven management service.

- When the intent instance to be created is Intent-CSC, MnS Consumer is the Communication Service Customer and MnS Producer is Communication Service Provider.
- When the intent instance to be created is Intent-CSP, MnS Consumer is the Communication Service Provider and MnS Producer is the Network Operator.
- When the intent instance to be created is Intent-NOP, MnS Consumer is the Network Operator and MnS Producer is the Network Equipment Provider.

4.1.3 Intent expectations for different types of management needs

Intent expectations for different types of management needs:

- **Intent expectation for delivering network and service related object:** enables a consumer to express the intent expectation for the object (e.g. network, service, slice) to be delivered by the system. Examples of such intent expectations are:
  - "Delivering a radio network in the specified area with specified frequency information, transport information, and radio information (e.g. range of PCI, Cell Id), network capacity and performance information".
  - "Delivering a radio service in the specified area with certain service characteristics (e.g. SLS)".
- **Intent expectation for network and service related object performance:** enables a consumer to express the performance objectives of the object (e.g. network, service, slice) to be assured. Examples of such intent expectations are:
  - "Ensure the radio network in the specified area meets certain expected RAN UE throughput objectives (e.g. expected average RAN UE DL throughput, expected percentage of UE with the RAN UE DL throughout less than 5 Mbps)".
  - "Ensure the radio network in the specified area meets certain expected coverage objectives (e.g. expected coverage ratio, expected average RSRP)".

4.2 Intent driven management

4.2.1 Support for intent driven management

In intent driven management, the consumer provides its intent to the producer of a set of management services that would be consumed in a specific domain. For example, for the purpose of requesting a radio network with a new coverage, one possible solution (non-intent driven approach) is to use the set of classic MnSs (e.g. provisioning MnS) to decommission a cell and instantiate the cell to a new Node B for the new coverage. The alternative solution (intent driven approach) is to use management service produced by the domain, which may be referred to as the Intent-driven MnS by stating the intent for the radio network for the new coverage, based on the intent, system can trigger actions (e.g. decommission a cell and instantiate the cell to a new Node B) to satisfy received intent. The Intent driven MnS could in principle deployed as a replacement of the deployed classic MnSs for the same network and service management purpose, where the consumer focuses on the 'what' and the producer is concerned about the 'how'.

The producer of an Intent-driven MnS shall allow the consumer to manage the service and / or network resources through the use of intents. The producer shall support the capabilities for intent fulfilment, which include the following:

- The consumer states the intent to be fulfilled (which can be implemented by createMOI operation on the Intent IOC) and the producer receives and acknowledges the receipt of the intent.
- The producer translates the intent to identify which actions are needed for intent fulfilment.
- The producer executes the required actions to fulfil the intent.
- The producer may report about the fulfilment result of the intent.

4.2.2 Intent driven MnS

Introduction of service-based architecture for 5G, in combination with functional model of business roles, exceeds the level of complexity for managing network in different scenarios (including scenarios for design/planning, deployment, maintenance and optimization) both in a single and multivendor network. New/simpler ways of managing are needed.

Actions of an intent driven MnS related to the fulfilment of intents may be categorized as intent deployment and intent assurance. Intent fulfilment refers to the steps taken to satisfy a newly received intent or an update to an existing intent. The goal of intent fulfilment is to bring the network's or service's state to satisfy the new or updated intent. The fulfilment of some intents may end at the intent deployment, in these cases, the intent's goal only describes the availability or presence of a network or service. In other cases, the intent's goal describes additionally the assurance requirements for a network or service (e.g. quality of service, end user experience, SLS, etc.) in addition to the need of existence of a network or service. Those intents have their fulfilment tied to the operation of the referred network or service and may require frequent recurring actions to keep those assurance requirements achieved. This part of the intent fulfilment is referred to as intent assurance.

An Intent driven MnS allows its consumer to express intents for managing the network and services and obtain the feedback of intent evaluation result. The Intent-driven MnS producer have the following intent handling capabilities:

- Translate the received intent to executable actions as follows:
  - Performing service or network management tasks.
  - Identifying, formulating, and activating policies for service or network management.
- Evaluate the result/information about the intent fulfilment, including intent deployment (e.g. the intent is initially satisfied or not) and intent assurance (e.g. the intent is continuously satisfied).

Figure 4.2.2-1 shows the model of Intent-driven MnS.

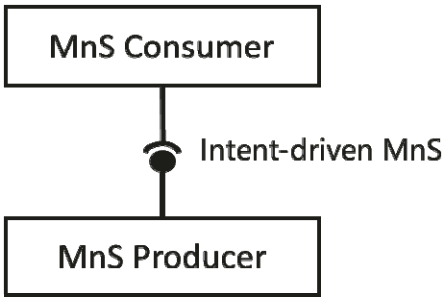


Figure 4.2.2-1: Intent-driven MnS

The intents may be fulfilled by utilizing multiple mechanisms including among others: Rule-based mechanisms, closed loop mechanisms and AI/ML based mechanisms. These mechanisms can be combined into solutions of various complexity, ranging from a simple approach used rule-based mechanisms, to more elaborate solutions combining AI/ML, closed loop automation to ensure the fulfilment of intents.

When the intent is created by MnS producer based on MnS consumer's request, the MnS producer may consume other management services (including non-intent driven MnS and intent driven MnS) to fulfil or satisfy the intent, e.g. creating new assurance closed control loop instance(s) or using assurance closed control loop instance (s) to satisfy the intent. The internal implementation of the intent fulfilment will however not be standardized.

An Intent driven MnS includes the following management capabilities to support intent lifecycle management:

- Create an intent, a MnS Consumer request MnS producer to create a new intent.
- Activate an intent, MnS Consumer request MnS producer to activate an intent when the intent is suspended.
- De-activate an intent, MnS consumer request MnS producer to de-activate an intent for a temporary suspension.
- Delete an intent, MnS Consumer request MnS producer to remove an intent.

- Modify an intent, MnS Consumer request MnS producer to modify the content of the intent (e.g. expectation targets).

- Query an intent, MnS Consumer request MnS producer to return the content and state (e.g. active, inactive) of the intent.

4.2.3 Intent translation

The Intent driven MnS producer is the provider of Intent driven MnS and is responsible for deriving activities for networks and services or other intent(s).

The MnS consumer may consume Intent Driven MnS(s) provided by the Intent driven MnS producer(s) or may have the consumer role for non-intent MnS producers.

The conflict(s) including conflict between the intent and other intent(s) and/or Non-intent requirements needs to be detected and resolved during the intent translation. Figure 4.2.3-1 illustrates a possible way to satisfy intent-CSC.

- Intent-CSC MnS producer provides intent driven MnS for communication services. Intent-CSC MnS producers receive the expressed intent and translate it to Intent-CSP or network requirements, then may consume Intent-CSP MnS(s) or Non-Intent MnS(s) for network to fulfil the intent-CSC.

- Intent-CSP MnS producer provides intent driven MnS for network services. Intent-CSP MnS producers receive the intent and translate it to new Intents for NOP or network requirements, then may consume Intent-NOP MnS(s) or Non-Intent MnS(s) for NE to fulfil the intent-CSP.

- Intent-NOP MnS producer provides intent driven MnS for network equipment. Intent-NOP MnS Producers receive the expressed intent, and translate it to detailed network requirements, then takes some internal actions to fulfil the intent-NOP.

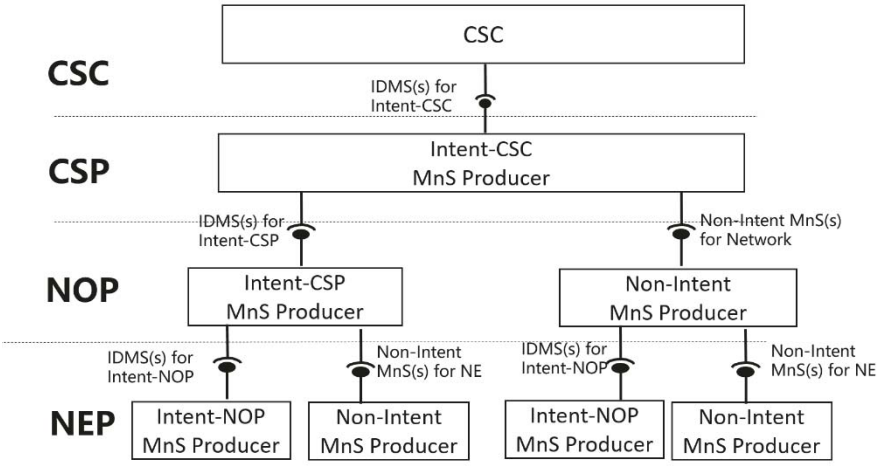


Figure 4.2.3-1: Potential way to satisfy intent-CSC

4.3 Intent driven closed-loop

Intent can be used for management and control of closed-loop automation (e.g. intent can be used to specify the goals for the closed-loop), which means the intent can be translated to policies and management tasks that the MnS producer needs to execute for the closed-loop automation. In the intent driven management approach, the mechanisms that the MnS producer using closed-loop automation mechanisms to satisfy the intent is the implementation of the MnS producer and shall not be standardized. The relation of the Intent driven MnS and the closed-loop automation with the Intent driven MnS producer is shown in the figure 4.3-1.

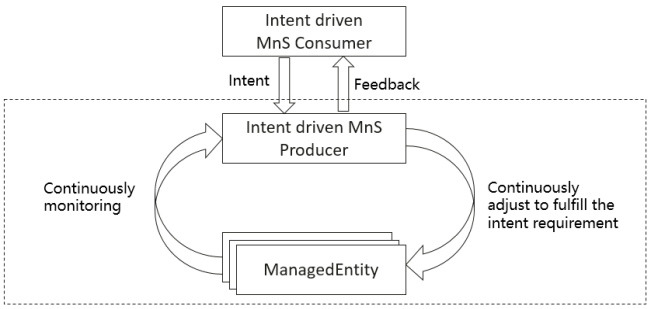


Figure 4.3-1: Intent driven closed-loop

4.4 Relation between rule, policy and intent

An intent specifies the expectations including requirements, goals, and constraints for a specific service or network management workflow, while a policy specifies the action(s) to be taken when given condition occurs and rules specifies the explicit or formula logics to be executed. For certain scenarios, policies can be used in conjunction with intents to achieve the autonomous purposes. Figure 4.4-1 describes the relation between rule, policy and intent in the "what-how" view. As it now stands, the telecom systems are mainly focused on "how" and "less what". The current 5G networks brings more operational complexities, and the telecom system need to be able to adapt their operation to the business objectives of the operator as well as expectations of customer, which is driving customer to shift the focus from "how" to "what". The first step towards that shift, has been shift from "Rule based management" to "Policy driven management", with more focus on "how" and less on "what" covering domain specific issues/aspects (an example for policy is when the average throughput is lower than certain threshold, take specified actions). As technologies are evolving and the level of complexity exceeds, the need for an abstraction level description (i.e. Intent) becomes more apparent (an example for intent is the target average throughput for certain area should be assured). An intent driven system will be able to learn the behaviour of networks and services and allows a customer to provide the desired state, without detailed knowledge of how to get to the desired state.

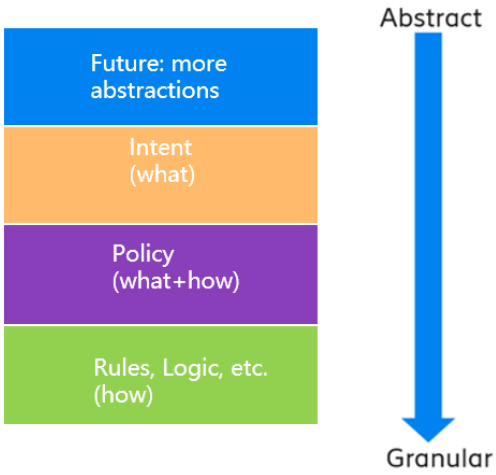


Figure 4.4-1: Relation between rule, policy and intent

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 and contexts for an object.

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

"objects $\{O_1, O_2, \ldots O_N\}$ with characteristics S"

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

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 expectation targets may include the metrics that characterize the performance of the object(s) or some abstract index that expresses the behaviour of the object(s). A given intent expectation may include multiple expectation targets on the same object or on different objects with the same properties. A consumer may for example require for the Network 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 expectation targets given a specific target context. As such with the characteristics as a combination of expectation targets and target contexts, the intent expectation may be stated as:

```
"ensure that for
    Expectation Object O,
    Expectation Target_1 is T_1, Target Context_1 is C_1
     …..,
    Expectation Target_m is T_m, Target Context_k is C_k;
```

Each expectation target expresses an aspect of the characteristics of the object under consideration, i.e. it expresses desired characteristics on a specific object. Each of the object characteristic may be desired to be equivalent to a specific value or constrained to a value or a range of values, e.g. as listed in Table 4.5.2-1. The combination of the name of characteristic (or simply the targetName), the condition constraining the characteristic and the value or value range for the characteristic is the target, i.e. the Expectation Target is the tuple:

Expectation Target = [targetName, condition, value range]

Table 4.5.2-1: Examples of Expectation Objects and Targets for different Intent expectations

Example of Intent Expectations	ExpectationObject	Example of Expectation Targets		
		targetName	Condition	Value range
example 1	Network Slice	Coverage area	Is greater than	40 km radius
example 2	Communication Service	User throughput	Is greater than	2 Mbps

4.5.3 Expectation Objects

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.

EXAMPLE 1: It may be easier to state "all network slices in city ABC" as opposed to listing the individual network slice. 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).

EXAMPLE 2: In the case of "all network slices in a city" there is an object context, which is the tuple (location, =, city_ABC) and (objectType,=, network slice).

4.5.4 Context

Each expectation target may be constrained to only be achieved for a very specific set of conditions as context constraints. The context describes a set of conditions to trigger corresponding management tasks to achieve the expectation targets. For example, the consumer may state that: "ensure that handoverFailureRate < 2 % if Load > 80 %", where the expectation target (HandoverFailureRate < 2 %) is to be achieved only in the context (Load > 80 %). In this example, the producer will perform handover tasks to achieve the expectation target "HandoverFailureRate < 2 %" when observe the context "Load > 80 %".

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

Although contexts and targets have the same structure, to distinguish between what needs to 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 wishes 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/modelled as context, the producer could interpret the request to mean RLF < 2 % and load > 50 %.

For a given expectation, the specific list of expectation targets may be desired to be achieved for given combined contexts, i.e. besides the expectation targets, an expectation may state a list of contexts which apply to all expectation 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 modelled to only contain the contexts that apply to all the contained sub elements.

Multiple contexts may be stated on an intent, on an intentExpecation, on an expectationObject or on an expectationtarget. However, the MnS consumer expressing multiple overlapping contexts may not have interest in simultaneously applying all the overlapping contexts but only in applying any one or more of the contexts. For example, the consumer may have requirements for any one of multiple physical locations at any given time. In such cases, the MnS consumer requires means to express the context to be selected among any one of these. Otherwise, the producer will either deliver for all contexts. Alternatively, the MnS consumer would only be able to express one context at time and wait e.g., for a feasibility check to fail to then resend the intent with revised context.

4.5.5 General Requirements

REQ-Intent_GEN-1 The intent driven MnS shall include a capability enabling MnS consumer to express intent containing one or more intent expectations, each expectation is a list of desired outcomes on objects of the same type with optional contexts for the desired outcomes, expectations or the intent.

REQ-Intent_ GEN-2 The intent driven MnS shall include a capability enabling MnS consumer to state the selection mechanism (including "ALL_OF", "ONE_OF", "ANY_OF") to be applied to the stated list of contexts.

4.6 Intent negotiation functionalities

4.6.1 Overview

An important aspect of intent-driven systems is the support of automation of interactions between the intent-driven MnS consumer and intent driven MnS producer. Introducing intent negotiation functionalities will be beneficial for the MnS producer and the MnS consumer to engage in a collaborative way. The intent negotiation can occur in the following two phases:

- Intent pre-evaluation, before the MnS consumer expresses the intent expectation to be fulfilled.

- Intent fulfilment, after the MnS consumer had expressed the intent expectation to be fulfilled.

4.6.2 Intent negotiation functionalities in Intent pre-evaluation phase

The intent negotiation functionalities for Intent pre-evaluation can be used to assist the MnS consumer to generate suitable intent information for the MnS producer. The network (including NEs) will not be changed during the intent pre-evaluation phase. The Intent negotiation functionalities in the intent pre-evaluation phase includes:

- Intent Feasibility check. The management capability to enable the MnS consumer to check if the proposed intent can be supported by the MnS producer.

- Intent Exploration. The management capability to enable the MnS consumer and the MnS producer to find the intent for fulfilment that is best aligned with MnS producer's capabilities.

4.6.3 Intent negotiation functionalities in Intent fulfilment phase

The intent negotiation functionalities for intent fulfilment can be used to enable the MnS producer and the MnS consumer to agree on the best way to fulfil an intent. The Intent negotiation functionalities in the intent fulfilment phase includes

- Checking for fulfillable outcomes. The management capability to enable the MnS consumer to obtain the fulfillable outcomes.

- Checking for best possible outcome. The management capability to enable the MnS consumer to obtain the best outcome.

- Recommending fulfillable outcomes. The management capability to enable the MnS producer to provide the recommended intent targets and contexts which are realizable.

- Advising on preferred outcome. The management capability to enable MnS consumer to advise on their (the MnS consumer's) preference among multiple outcomes.

Editor's Note: the definition for outcomes needs to be further clarified.

4.7 Intent handling state management

This clause provides Intent handling state diagram and corresponding state transition events to show the Intent handling state management.Figure 4.7-1 shows the intent handling state diagram, where the number in the figure identify the state changes. The explanations for the state changes are described in Table 4.7-1.

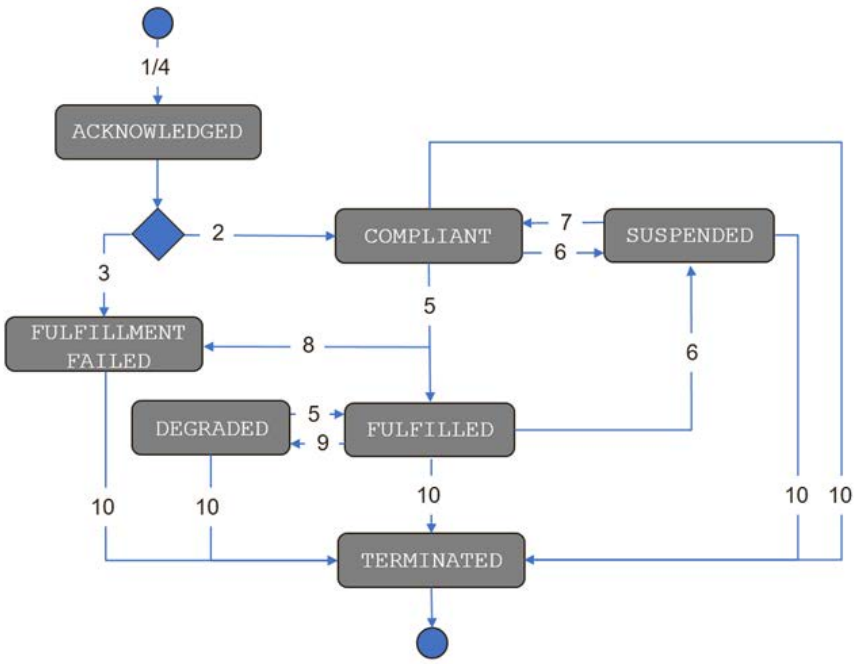


Figure 4.7-1: Intent handling state diagram

The transition numbers in the first column represent the state changes in Figure 4.7-1. The interactions specified under the column "The state transition events" of Table 4.7-1 should be present for the state transition.

Table 4.7-1: The intent handling state transition table

Transition number	The state transition events	State
1	The MnS producer creates the intent instances based on the received intent creation request.	ACKNOWLEDGED
2	The MnS Producer performs the feasibility check of the intent instance, and the feasibility check result is 'feasible'.	COMPLIANT
3	The MnS Producer performs the feasibility check of the intent instance, and the feasibility check result is 'infeasible'	FULFILLMENT_FAILED
4	The MnS producer modifies the intent instances based on the received intent modification request. Correspondingly, the state transitions from COMPLIANT, SUSPENDED, FULFILLED, FULFILLMENT_FAILED or DEGRADED to ACKNOWLEDGED.	ACKNOWLEDGED
5	The MnS producer considers that the intent has been fulfilled.	FULFILLED
6	MnS producer configures attribute intentAdminState with the value "DEACTIVATED"	SUSPENDED
7	MnS producer configures attribute intentAdminState with the value "ACTIVATED"	COMPLIANT
8	The MnS producer determines that he cannot do anything to fulfil the intent.	FULFILLMENT_FAILED
9	The MnS producer considers that the intent was fulfilled but after a period of observation it is found that the originally specified requirements are no longer met.	DEGRADED
10	The MnS producer deletes the intent instances based on the received intent deletion request.	TERMINATED

5 Specification Level Requirements

5.1 Use cases

5.1.1 Intent containing an expectation for delivering radio network

5.1.1.1 Introduction

This use case describes a scenario where a MnS consumer express intent containing an expectation for delivering a radio network in the specified area to a MnS producer. In this scenario, MnS consumer expresses its intent expectation for delivering a radio network to MnS producer, which may include coverage area information (e.g. geographical areas), radio setting parameter sets (e.g. frequency information, range of gNB Id, range of PCI, range of Cell Id, range of nRTAC), transport setting parameters (including OM transport information (e.g. OMlocalIPaddress, OMremoteIPaddress, OMNextHopInfo) and NG transport information (e.g. list of NGlocalIPaddress, list of NGremoteIPaddress)), and supported network capacity information (e.g. maximum UE number) and network performance information (e.g. UL/DL throughput, PRB resources, 5GS to EPS handover success rate). This can be applied not only to a single operator, but also to multi-operator scenarios while ensuring undifferentiated network performance (e.g. traffic, mobility) treatment.

Based on the intent containing an expectation for radio network provisioning received, MnS producer identifies corresponding RAN NEs discovered in the specified coverage area, analyses and generates the configuration parameters (including radio configuration parameters and transport configuration parameters) for each identified RAN NE and corresponding Cells, creates MOI(s) for each RAN NEs and Cells and configure the created MOI(s), and performs verification for configured RAN NEs to enable the radio network in the specified area is successfully delivered and satisfy the received intent.

MnS producer notifies MnS consumer about the fulfilment information of the intent containing an expectation for delivering radio network after the verification is finished.

5.1.1.2 Requirements

REQ-IDMS_RadioNetworkIntent-CON-1 The intent driven MnS producer for radio network shall have capability enabling MnS consumer to express intent containing an expectation for delivering a radio network for the specified area.

REQ-IDMS_RadioNetworkIntent-CON-2 The intent driven MnS producer for radio network shall have capability enabling MnS consumer to obtain intent report information (including fulfilment information) for the intent containing an expectation for delivering a radio network.

5.1.2 Intent containing an expectation for delivering a radio service

5.1.2.1 Introduction

This use case describes a scenario where a MnS consumer express intent containing an expectation for delivering radio service (radio network as service) in the specified area to a MnS producer.

In this scenario, MnS consumer expresses its intent containing an expectation for delivering a radio service to MnS producer, which may include coverage area information (e.g. geographical areas), and supported service capacity information (e.g. maxNumberofUEs, activityFactor) and service performance information (e.g. serviceType, dLThptPerUEPerSubnet, uLThptPerUEPerSubnet).

In addition, MnS consumer may expect the radio service to be delivered and assured at a scheduled time instead of all the time. The scheduled time can be one-time interval, daily periodicity, weekly periodicity or monthly periodicity.

NOTE: The slice agnostic parameters in RAN SliceProfile can be used for service capacity information and service performance information.

Based on the intent containing an expectation for delivering a radio service received, MnS producer decides to use radio network with slicing or radio network without slicing to support the intent:

- In case of using radio network with slicing, the use case for network slice subnet creation defined in 3GPP TS 28.531 [2] can be reused.
- In case of using radio network without slicing, MnS producer identifies corresponding RAN NEs and cells in the specified coverage area to support the intent, analyses and configure the service specific configuration parameters for corresponding RAN NE and Cells (e.g. RRM policies, supported services).

MnS producer notifies MnS consumer about the fulfilment information of the intent containing an expectation for delivering a radio service after the service configuration is finished.

5.1.2.2 Requirements

REQ-IDMS_RadioServiceIntent -CON-1 The intent driven MnS producer for radio service shall have capability enabling MnS consumer to express intent containing an expectation for delivering a radio service for the specified area.

REQ-IDMS_RadioServiceIntent-CON-2 The intent driven MnS producer for radio service shall have capability enabling MnS consumer to obtain intent report information (including fulfilment information) for the intent containing an expectation for a service.

REQ-IDMS_RadioServiceIntent-CON-3: The intent driven MnS producer for radio service shall have capability enabling MnS consumer to express intent containing an expectation for delivering a radio service in a scheduled time.

5.1.3 Intent containing an expectation for delivering a service at the edge.

5.1.3.1 Introduction

This use case describes a scenario where the MnS consumer, express the intent containing an expectation for delivering a service at the edge of the network. The intent expectation for a service includes service type (e.g.,URLLC, eMBB), service requirements (number of concurrent subscribers and number of concurrent sessions), service availability and the target location.

MnS producer notifies MnS consumer about the fulfilment information of the intent containing an expectation for delivering a service at the edge.

5.1.3.2 Requirements

REQ-IDMS_EdgeServiceIntent-CON-1 The intent driven MnS producer for edge service shall have capability enabling authorized MnS consumer to express intent containing an expectation for delivering a service at the edge of the network.

REQ-IDMS_EdgeServiceIntent-CON-2 The intent driven MnS producer for edge service shall have capability enabling authorized MnS consumer to obtain intent report information (including fulfilment information) for intent containing an expectation for delivering a service at the edge of the network.

5.1.4 Intent containing an expectation on coverage performance to be assured

5.1.4.1 Introduction

In this scenario, MnS consumer expresses its intent containing an expectation on coverage performances to be assured in the specified areas to NEP, which may include area information (e.g. geographical area), RATs (e.g. NR only, EUTRAN only, or all RATs), coverage targets (e.g. target average RSRP, target weak coverage ratio).

Based on the intent containing an expectation on coverage performance to be assured received, MnS producer collects and analyses corresponding coverage related data (e.g. RSRPs of the serving cell and neighbour cells reported by each UE with anonymous id (e.g. C-RNTI) and location information in the MDT reports) of corresponding RAN NEs in the specified areas, identifies the potential coverage issues which will impact the coverage targets satisfaction, analyses the identified coverage issue and corresponding solutions, evaluates, decides and adjusts the coverage configuration parameters. The Artificial intelligence or machine learning technologies may be used in above workflow to satisfy the intent, for example, online iteration optimization technologies may be used to selecting the best coverage configuration parameters rapidly.

MnS producer continuously monitors the coverage performance (e.g. weak coverage ratio, average RSRP) for the specified area, and decides whether coverage targets described in the intent is satisfied. If not satisfied, MnS producer iteratively executes above workflows (including collect, identification, analysis, evaluation, decision and adjustment) to fulfil the coverage targets.

MnS producer may notify MnS consumer about the intent fulfilment information, including coverage performance for the specified area (e.g. weak coverage ratio, coverage hole ratio, average RSRP) which enables MnS consumer to monitor the intent containing an expectation on coverage performance to be assured. MnS consumer may also request to MnS producer to report the intent fulfilment information.

5.1.4.2 Requirements

REQ-IDMS_RadioNetworkIntent-CON-3 The intent driven MnS producer for radio network shall have capability enabling MnS consumer to express intent containing an expectation on coverage performance to be assured for the specified area.

REQ-IDMS_RadioNetworkIntent-CON-4 The intent driven MnS producer for radio network shall have capability enabling MnS consumer to obtain intent report information (including fulfilment information) for the intent containing an expectation on coverage performance to be assured.

5.1.5 Intent containing an expectation on radio network performance to be assured

5.1.5.1 Introduction

In this scenario, MnS consumer expresses its intent containing an intent expectation on radio network performance (including RAN UE throughput performance, radio network capacity performance) to be assured to MnS producer, which may include area information (e.g. geographical area, a list of cells), RATs (e.g. NR only, EUTRAN only, or all

RATs), frequency information (e.g. nRFrequencyBand), radio network performance targets, optional performance scope (e.g. specific service type, specific UE groups).

The radio network performance targets include the targets in the following target categories based on what radio network performance MnS consumer expects to be assured.

- RAN UE throughput targets, for example, target average UL/DL RAN UE throughput, target percentage of UE with low UL/DL RAN UE throughput (e.g. < 5 Mbps), target percentage of UE with high UL/DL RAN UE throughput (e.g. > 50 Mbps).
- Radio network capacity targets, for example, target percentage of high UL/DL PRB Load (e.g. < 70%), target average UL/DL PRB load (e.g. <85%).
- User number related targets, for example, target maximum user numbers (e.g. 500 UEs).

MnS consumer can express radio network expectations on radio network traffic assurance for the scheduled times for specific events. For example, specified RAN UE throughput targets or user number requirements during a sporting event or concert. The schedule times can be one-time interval, daily periodicity, weekly periodicity or monthly periodicity.

MnS consumer may also provide requirements to be fulfilled in a specific combination of network scopes with specific time windows. For example, to support UAV pre-flight preparation in TS 22.125 [19], the MnS consumer may provide radio network expectations on radio network performance (e.g. network coverage) in the specific geographical area zone (e.g. aerial flight zone) to the MnS producer (The geographical area zone is composed of a list of GeoArea and TimeWindow pairs, i.e. a list of <GeoArea, TimeWindow>).In this example, the MnS producer identifies corresponding managed entities of the intent and configures the managed entities based on geographical area zones to fulfil the targets when the UEs need to be served by the managed entities (e.g. RAN NEs) in the specified geographical area zone.

Based on the intent containing expectation on radio network performance for the specified area to be assured received, MnS producer collects and analyses corresponding radio network performance related data (e.g. RAN UE throughput data, number of PRBs used for UL/DL traffic transmission) in the specified areas, identifies the potential radio network performance issues (e.g. low RAN UE throughput for certain areas, high load for certain areas, high UL/DL PRB Load issue), which will impact radio network performance intent satisfaction, analyses the cause, evaluates, decides and adjusts the radio feature configuration parameters for impacted RAN NEs/Cells to address the radio network performance issues in the specified areas. The artificial intelligence or machine learning technologies may be used to select the optimal radio feature configuration parameters to satisfy radio network performance targets.

MnS producer continuously monitors the radio network performance (e.g. average UL/DL RAN UE throughput, percentage of UE with low UL/DL RAN UE throughput (e.g. < 5 Mbps), percentage of UE with high UL/DL RAN UE throughput (e.g. > 50 Mbps), percentage of high UL/DL PRB Load (e.g. < 70%)) for the specified area, and decides whether radio network performance targets are satisfied.

MnS producer may notify MnS consumer about the intent fulfilment information and achieved value for radio network targets, including the radio network performance (e.g. average UL/DL RAN UE throughput, percentage of UE with low UL/DL RAN UE throughput, percentage of high UL/DL PRB Load) for the specified area which enables MnS consumer to monitor the intent containing an expectation on radio network performance to be assured.

5.1.5.2 Requirements

REQ-IDMS_RadioNetworkIntent-CON-5 The intent driven MnS producer for radio network shall have capability enabling MnS consumer to express intent containing an expectation on RAN UE throughput performance to be assured for specified area.

REQ-IDMS_RadioNetworkIntent-CON-6 The intent driven MnS producer for radio network shall have capability enabling MnS consumer to obtain intent report information (including fulfilment information) for the intent containing an expectation on RAN UE throughput performance to be assured.

REQ-IDMS_RadioNetworkIntent-CON-7: The intent driven MnS producer for radio network shall have capabilities enabling the MnS consumer to express intent containing an expectation on radio network capacity performance to be assured for the specified area.

REQ-IDMS_RadioNetworkIntent -CON-8: The intent driven MnS producer for radio network shall have capabilities enabling the MnS consumer to obtain intent report information (including fulfilment information and achieved value) for intent containing an expectation on radio network capacity performance to be assured.

NOTE: the example of radio network capacity performance is target percentage of high UL/DL PRB Load for a specified Geographical area.

5.1.6 Intent containing an expectation for end-to-end network optimization

5.1.6.1 Introduction

In this scenario, MnS consumer expresses its intent containing an intent expectation with targets on the whole Network including RAN and Core. The intent may for example be for optimization of the network resources, i.e. the intent expectation captures the objectives for an entity that undertakes optimization for the network. The expectation may be termed as network resources expectation. The network resources expectation targets may express the desired performance optimization outcomes. Depending on the stated targets, the MnS producer may as such configure one or more optimization functions to achieve the desired targets. The network optimization expectation targets may for example be end-to-end KPI targets that the optimization is required to achieve.

MnS producer notifies MnS consumer about the fulfilment information of the intent containing an expectation for end-to-end network optimization.

The network optimization expectation may include relative prioritizations of the different targets which indicate the relative interests of the intent MnS consumer on the different network attributes.

5.1.6.2 Requirements

REQ-IDMS_E2ENetworkIntent-CON-1 The intent driven MnS producer for end-to-end network shall have capability enabling MnS consumer to express intent containing an expectation with targets on the end-to-end network including the RAN and Core.

REQ-IDMS_E2ENetworkIntent-CON-2 The intent driven MnS producer for end-to-end network shall have capability enabling MnS consumer to express intent containing an expectation with the prioritization of the targets to be achieved.

REQ-IDMS_E2ENetworkIntent-CON-3 The intent driven MnS producer for end-to-end network shall have capability enabling MnS consumer to obtain intent report information (including fulfilment information) for the intent containing an expectation with targets for the end-to-end Network.

5.1.7 Intent containing an expectation for RAN energy saving

5.1.7.1 Introduction

Operators are aiming at decreasing power consumption in 5G networks to lower their operational expense with energy saving management solutions. Energy saving is achieved by executing the energy saving actions with suitable parameter configurations, e.g. energy saving state switch, start time and end time, the energy saving thresholds. However, the various combinations of energy saving actions can lead to conflicts. For example, different energy saving actions may be contradictory, or the energy saving actions may conflict with other activities (e.g. network optimization actions). Moreover, it is not straightforward to evaluate the influence on service experience (e.g. UL/DL RAN UE throughput, latency) of energy saving actions beforehand, which makes it difficult to balance the energy saving effect and service experience, for example the energy saving actions may deteriorate the service experience. To avoid affecting the service experience, MnS consumer may express energy saving target with the maximum value of RAN energy consumption in intent expectation, and MnS producer is able to choose an optimal value of RAN energy consumption to save energy as much as possible in the context to satisfy the service experience.

As clause 4.1.1 described, an intent focuses more on describing the "What" needs to be achieved but less on "How" that outcomes should be achieved, which not only relieves the burden of the consumer knowing implementation details but also leaves room to allow the producer to explore alternative options and find optimal solutions. So, introducing the intent approach for energy saving, which can enable the 3GPP management system to analyse and select the optimal balance between the energy saving effect and service experience by utilizing some intelligence mechanisms. In intent driven approach, a MnS consumer expresses intent expectation for RAN energy saving in the specified area (e.g. geographical area) to a MnS producer, which includes the RAN energy saving target (e.g. the maximum value of target RAN energy consumption, reduction radio of energy consumption) and service experience target (e.g. RAN UE throughput, latency), as well as the frequencies and RATs to be considered and will be used to determine the managed

entities that are to perform energy saving actions. Some contexts for RAN energy saving (e.g. RAN energy saving allowed time (e.g., 1:00 am-5:00 am), RAN energy saving trigger event (e.g. PRB load ratio < 50%)) also can be specified by MnS consumer to provide the conditions to allow corresponding energy saving actions to be triggered to satisfy the energy saving targets. MnS producer analyses and determines the optimal RAN energy saving solution (i.e. a set of energy saving actions) to satisfy MnS consumer's intent expectation for RAN energy saving. MnS producer continuously monitors the RAN energy saving performance (e.g. RAN energy consumption, RAN energy efficiency) and service experience performance (e.g. target average UL/DL RAN UE throughput, target) for the specified area, and decides whether RAN energy saving target is satisfied.

The MnS consumer may want to assure different RAN UE throughput performance for different contexts (including frequencies or rATs) in the same area when perform energy saving activities (e.g., same targets for RAN energy saving). The MnS consumer also needs to receive the target fulfilment result for different RAN UE throughput targets for different contexts (including frequencies or RATs).

MnS producer may report the intent fulfilment information and achieved value for RAN energy saving targets (e.g. RAN energy consumption, RAN energy efficiency) and the service experience targets for the specified area to MnS consumer which enables MnS consumer to monitor the intent containing an expectation for RAN energy saving.

5.1.7.2 Requirements

REQ-IDMS_ RadioNetworkIntent-CON-9: The intent driven MnS producer for radio network shall have capabilities enabling MnS consumer to express intent containing an expectation for RAN energy saving for the specified area.

REQ-IDMS_RadioNetworkIntent-CON -10: The intent driven MnS producer for radio network shall have capabilities enabling MnS consumer to obtain intent report information (i.e. fulfilment information, achieved value for corresponding targets) of the intent containing an expectation for RAN energy saving.

5.1.8 Intent containing an expectation for 5GC network

5.1.8.1 Introduction

This use case describes a scenario where a MnS consumer expresses intent containing an expectation related to 5GC network to the intent driven MnS producer. In this scenario, MnS consumer expresses its intent expectation which may include area information (e.g. geographic area, data center), type of the network (e.g. NPN), included 5GC NF list (e.g. NF type, range of NF instance ID), PLMN information, supported APN information, transport related parameters (e.g. list of end point addresses information), and target network capacity information (e.g. number of PDU session, number of registered subscribers, UL/DL throughput).

Based on the received intent containing an expectation for 5GC network, the intent driven MnS producer decides whether to deploy a new 5GC network in the specified area or to re-use and modify an existing 5GC network. If a new 5GC network is to be deployed, the intent driven MnS producer translates the intent 5GC network expectations into appropriate 5GC network provisioning actions, this may include generation of network configuration parameters (including 5GC network/NFs configuration parameters and transport network configuration parameters) and triggering NS/VNF creation procedure by interworking with ETSI NFV MANO. If an existing 5GC network is to be re-used, the intent driven MnS producer identifies the potential 5GC network performance issues (e.g. low performance because of high load) for the existing 5GC network, modifies the 5GC NF configuration parameters if needed to satisfy the performance expectation targets (this may also trigger scaling procedure by interworking with ETSI NFV MANO). Multiple interactions between the intent driven MnS consumer and the intent driven MnS producer may be needed based on the intent management capabilities (e.g. intent feasibility check) provided by intent driven MnS producer.

The intent driven MnS producer continuously monitors the 5GC network performance (e.g. mean number of registered UE, mean number of created PDU session), and decides whether 5GC network expectation is satisfied. If the 5GC network expectation is not satisfied, the intent driven MnS producer identifies the potential 5GC network performance issues and modifies the 5GC NF configuration parameters if needed to satisfy the performance expectation targets.

On a regular basis, the intent driven MnS producer notifies MnS consumer about the fulfilment information of the intent.

5.1.8.2 Requirements

REQ-IDMS_5GCIntent-CON-1 The intent driven MnS producer for 5GC network shall have capability enabling MnS consumer to express intent containing an expectation for delivering a 5GC network for the specified area.

REQ-IDMS_5GCIntent -CON-2 The intent driven MnS producer for 5GC network shall have capability enabling MnS consumer to obtain intent report information (i.e. fulfilment information, achieved value for corresponding targets, conflict information and fulfilment feasibility check information) of the intent containing an expectation for delivering a 5GC network.

REQ-IDMS_5GCIntent -CON-3 The intent driven MnS producer for 5GC network shall have capability enabling MnS consumer to express intent containing an expectation on 5GC network performance to be assured.

REQ-IDMS_5GCIntent-CON-4 The intent driven MnS producer for 5GC network shall have capability enabling MnS consumer to obtain intent report information (i.e. fulfillment information, achieved value for corresponding targets, conflict information and fulfillment feasibility check information) of the intent containing an expectation on 5GC network performance to be assured.

5.1.9 Intent containing an expectation for network maintenance

5.1.9.1 Introduction

This use case describes a scenario in which an MnS consumer expresses its intent containing an expectation for a network maintenance work, for example, a software upgrade. The expectation termed as network maintenance expectation may include information to describe the characteristics of the network maintenance expectation, referring to any information specific to the network maintenance work, for example, maintenance target version, the start/end time of the intended network maintenance work and any additional guidelines for network maintenance.

The network maintenance expectation may also use expectation targets specified for other use-cases defined in the present document, for example, RAN Energy consumption target, if the maintenance of a RAN network element aims at reducing the RAN Energy consumption.

The network maintenance expectation may also use existing object contexts, for example, the location of NF instance for maintenance.

Based on network maintenance expectation received, MnS producer takes actions to fulfil the expectation and monitors the impact of the actions taken. The actions may include, for example, preparing the NE/NF for maintenance to avoid any disruption in the network, and carrying out the intended maintenance tasks such as upgrading the related files of NE/NF, and any other actions to ensure fulfilment of expectation targets of network maintenance expectation.

The MnS producer notifies MnS consumer about the fulfilment information of the intent containing an expectation for network maintenance, including information regarding conflicts that might occur during fulfilment of network maintenance expectation.

5.1.9.2 Requirements

REQ-Intent_NetworkMaintenance_CON_1: The intent driven MnS producer shall have the capability enabling MnS consumer to express intent containing a network maintenance expectation.

REQ-Intent_NetworkMaintenance_CON_2: The intent driven MnS producer shall have the capability enabling MnS consumer to obtain intent report information (including fulfilment information) for the intent containing a network maintenance expectation.

5.2 Generic requirements for intent driven MnS

REQ-IDMS_IntentLCM-CON-1 The intent driven MnS producer shall have capability enabling MnS consumer to request to create a new intent instance.

REQ-IDMS_IntentLCM -CON-2 The intent driven MnS producer shall have capability enabling MnS consumer to request to remove an intent instance.

REQ-IDMS_IntentLCM-CON-3 The intent driven MnS producer shall have capability to report intent fulfilment information to MnS consumer.

REQ-IDMS_IntentLCM-CON-4 The intent driven MnS producer shall have capability enabling MnS consumer to request to modify an existing intent instance.

REQ-IDMS_IntentLCM-CON-5 The intent driven MnS producer shall have capability enabling MnS consumer to query intent instance information.

REQ-IDMS_IntentLCM-CON-6 The intent driven MnS producer shall have a capability enabling an MnS consumer to request to activate a suspended intent instance.

REQ-IDMS_IntentLCM-CON-7 The intent driven MnS producer shall have a capability enabling an MnS consumer to request to deactivate an intent instance for a suspension.

5.3 Generic use case for intent driven management

5.3.1 Intent handling capability obtaining

5.3.1.1 Introduction

5.3.1.1.1 Information on supported expectation object and corresponding expectationTargets.

Clause 4.2.2 described that an Intent-driven MnS producer has the following capabilities: fulfil the received intent and report the result/information about the intent fulfilment, and clause 6.2.2 defined different scenario specific intent expectations with different expectation objects and expectation targets to support different use cases. In a network, multiple intent handling functions may be deployed to support different kinds of intents. Different intent handling functions may be deployed to support different intent expectation object domains, e.g. intent handling function A is deployed to handle the radio network related intents, intent handling function B is deployed to handle the 5GC network related intents, while intent handling function C is deployed to handle the service related intents. Or different intent handling functions are deployed to support different areas of the same intent expectation object domain, e.g. intent handling function D is deployed to support to handle the intent for radio network in Area#1, while intent handling function E is deployed to support to handle the intent for radio network in Area#2.

Before MnS consumer expresses the intent expectation targets and expectation objects to MnS producer, MnS consumer may want to know what expectation targets and expectation objects can be supported by MnS producer. Based on such supported expectation targets information and expectation objects information, the MnS consumer may use such information to select the proper intent handling function to express the intent.

In case the MnS producer updates the intent handling capabilities for one or more intent handling functions, the MnS producer may inform these updates to the MnS consumer.

5.3.1.1.2 Description of supported expectation targets

Besides the list of supported expectation objectType and expectationTargets, the intent handling capability needs to also describe the supported value ranges for the targets or the sets of targets that are supported. For instance, the intent driven MnS producer for network service management intent may want to expose the description of RAN services that they can offer. This means the intent driven MnS producer exposes the supported intents or intent expectations having a specific set of targets and a specific set of values for those targets as illustrated by Table 5.3.1.1.2-1. The IDMS producer should support expose of such capabilities.

Table 5.3.1.1.2-1: An example of a service offer description indicating the possible characteristics of network services that can be exposed via an intent handling capability as the service characteristic offered by an intent driven MnS producer for service management intents

Object Type		Network service							
Targets or Contexts	Units	Value Options							
End-to-end latency	ms	0.5	1	2	5	10	25	50	100
Jitter	ms	.001	0.01	0.1	1	2	5	10	20
Survival Time	ms	0	1	2	5	10	25	50	100
Availability	%	99	99.5.	99.9	99.995	99.999	99.9995	99.9999	.99999
Reliability	%	99	99.5.	99.9	99.995	99.999	99.9995	99.9999	.99999
End user Data rate	Mbps	0.1	0.2	0.5	1.0	2.0	5.0	10.0	20.0
Payload	Bytes	1	2	5	10	20	100	500	1000
Traffic density	Gbps/km^2	1	2	5	10	20	50	100	1000
Connection density	'000/km^2	0.1	0.5	1	5	10	20	50	100
Service area size	A km x B km	0.1x10	0.1x100	0.1x500	0.1 x 0.1	1 x1	10 x10	50 x50	100x100
:	:	...							

5.3.1.1.3 Describing alternative expectations

The IDMS may be used by the network service management MnS producer to enable the design of a specific network service management workflow between the MnS producer and the MnS consumer.

Based on the extended content of intent handling capability as described in 5.3.1.1.2, the network service MnS consumer can generate an artifact, say called a service-offer description that describes the different candidate characteristics of the desired network service from the MnS consumer's point of view. The MnS consumer provides a service-offer description to the network service management system for validation. The service-offer description may be viewed as a network service management intent that contains a list of candidate intent expectations, i.e., the expectations are candidates that may be separately submitted by the MnS consumer for fulfilment.

On receiving the service-offer description the network service Management MnS producer validates the service-offer description and confirms to what extent the network service intent as described by the service-offer description can be supported. For that, the network service management MnS producer returns the service-offer description but indicates only the supported combinations of network service features and values with contexts (conditions) under which the network service may be offered.

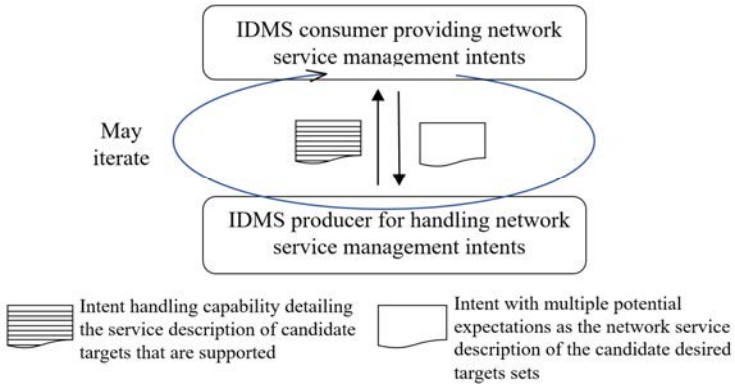


Figure 5.3.1.1.3-1: Example interaction between an MnS producer and MnS consumer on the exposure of capabilities for service management intents

The service offer description can use the capability of exposing alternative expectations to express the services that can be offered e.g. by a RAN service or Edge service intent handling function. It is up to the MnS producer how description information is provided. It is not indicative of ability to fulfil an intent as many factors can change over time depending on network conditions.

5.3.1.2 Requirements

REQ-IDMS_IHCO-CON-1 The intent driven MnS producer shall have capabilities enabling an MnS consumer to obtain intent handling capabilities of each intent handling function, including supported expectation object and targets.

REQ-IDMS_IHCO-CON-2: The intent driven MnS producer should support a capability to provide a description of the supported scenario specific intents to the authorized MnS consumer, including value ranges for the supported expectation targets.

REQ-IDMS_IHCO-CON-3: The intent driven MnS producer should support a capability enabling an MnS consumer to list the set of potential alternative expectations that the MnS consumer would like to be evaluated for whether it is feasible or fulfillable.

5.3.2 Intent report

5.3.2.1 Introduction

The intent fulfilment information is defined as one type of intent report information that includes the intentFulfilmentInfo, expectationFulfilmentInfo and targetFulfilmentInfo for the MnS consumer to monitor the intent fulfilment. It may also include the current value for performance indicated by corresponding expectation targets (e.g., WeakRSRPRatio for the weakRSRPRatioTarget, Average UL RAN UE Throughput for aveULRANUEThptTarget), which can be used by MnS consumer to validate whether the intent is fulfilled and to evaluate whether the intent (especially for expectation targets) needs to be updated (e.g., increasing or reducing the target value when corresponding target is not fulfilled).

Besides, intent conflict information, intent fulfilment feasibility check information, intent exploration information, intent fulfilment negotiation information and intent utility information sent by MnS producer to MnS consumer are other types of intent report information. So, following are the six types of information needs to be monitored by MnS consumer:

- Intent fulfilment information, which represents the properties of a specific fulfilment information for an aspect of the intent (i.e. either an expectation, a target or the whole intent), including fulfilment status and achieved values for targets.
- Intent conflict information, which represents intents with conflict. The information includes conflict type (i.e., intent conflict, expectation conflict and target conflict) and possible solutions (e.g. intent deletion, intent modification).

- Intent fulfilment feasibility check information, which indicates that the intent is feasible or infeasible. Intent fulfilment feasibility check information is provided after MnS producer automatically performs feasibility check when receiving the intent creation and modification request from MnS consumer.

- Intent exploration information, which represents the properties of the intent exploration result during negotiation in intent pre-evaluation phase. Intent exploration result is provided after MnS producer performs intent exploration pre-evaluation process as requested by MnS consumer.

- Intent fulfilment negotiation information, which represents the properties of intent negotiation information negotiation in intent fulfilment phase. Intent fulfilment negotiation information is provided to MnS Consumer by the MnS producer during intent fulfilment phase.

- Intent utility information, which provides the results of applicable Intent Utility Functions. The results are calculated and reported per utility function applicable to the intent specified in intentReference.

Different MnS consumer may have different requirements for intent report (e.g., some may want to have corresponding performance value information while others may not). Different MnS consumers may also want to calculate or monitor the performance values over different time periods.

MnS consumers may subscribe to get notifications of changes in the intent report. However, in some scenarios, the MnS consumer who expresses the intent may want to obtain the intent report by default, instead of triggering separate subscription action (i.e., request to create a NtfSubscriptionControl instance) to subscribe intent report information (especially intent fulfilment information).

In addition, MnS consumer who expresses the intent may want to obtain the customized intent report based on specified conditions. When the condition is satisfied, the MnS producer will send the intent report to the MnS consumer automatically. Then, the MnS consumer may delete or modify the intent (e.g., reduce the target value) based on the intent report. Therefore, the capability to support implicit intent report subscription is important for the MnS consumer who express the intent.

By expressing the requirements of the intent report in an intent, the MnS consumers can customize the content of an intent report. For example, the MnS consumer may specify a condition to indicate that a target is not fulfilled (i.e. the targetAchievedValue is out of the targetValueRange) and the achieved value for the target is stable (i.e. the targetAchievedValue remains constant for a specified observation periods).

An intent under fulfilment may go through multiple states, i.e., the life cycle includes multiple alternative transitions. The transitions are triggered either by actions of the MnS consumer or observations at the MnS producer or its intent handling function. Generally, intents received at the MnS producer will have one of the states represented in Figure 4.7-1.

Figure 5.3.2.1-1: Void

Intent reports should be delivered for each of these different states and related state transitions, specifically for:

1. `ACKNOWLEDGED:` When an intent instance is created, its default state is "`ACKNOWLEDGED`". An intent report should be delivered to indicate that the intent instance has been created. The transitions from "`ACKNOWLEDGED`" state can only be to "`COMPLIANT`", "`FULFILLMENT_FAILED`" or "`TERMINATED` " states as described hereafter.

2. `COMPLIANT:` When the feasibility check for the intent is successful and the intent is accepted as being compliant, the intent state is changed to "`COMPLIANT`". An intent report should be delivered to indicate the results of the feasibility check. In the case of an intent delivered for fulfillment, a corresponding intent report should be delivered to indicate the status change. The payload for the intent feasibility report (outcome of the feasibility check) may be one of the following:

    1. an indication for feasible or infeasible;

    2. a detailed report indicating which intent expectations or ExpectationTargets are infeasible and the corresponding reasons;

The transitions from "COMPLIANT" state are either to "ACKNOWLEDGED" state when the intent attributes are modified by the MnS consumer, or to "SUSPENDED", " FULFILLED " or "FULFILLMENT_FAILED" states as described hereafter.

3. SUSPENDED: If the MnS consumer decides to suspend the intent, the intent state changes to "SUSPENDED". Alternatively, an event may occur while the MnS producer attempts to fulfil the intent, in which case the MnS producer may suspend the intent pending further actions. Such events at the MnS producer may among others include conflicts from another intent and resource constraints. An intent report should be delivered to indicate that the intent has been suspended. The intent report may include a reason for the suspension when the intent was suspended by the MnS producer.

   The transitions from "SUSPENDED" state are either to "ACKNOWLEDGED" state when the intent attributes are modified by the MnS consumer, or to "COMPLIANT" state if the suspension is lifted by the entity (MnS consumer or MnS producer) that suspended the intent. Otherwise, the state is "TERMINATED" if the intent is deleted.

4. FULFILLED: If the MnS producer considers that the intent, expectation or target has been fulfilled as stated by the MnS consumer, the state changes to "FULFILLED". An intent report should be delivered to indicate that the intent has been fulfilled.

   The transitions from "FULFILLED" state are either to "ACKNOWLEDGED" state when the intent attributes are modified by the MnS consumer or to "DEGRADED" as described for the "DEGRADED" state below.

5. DEGRADED: If an intent that was previously fulfilled but after a period of observation it is found not be meeting the initially stated requirements, the state changes to "DEGRADED". An intent report should be delivered to indicate that the intent has degraded.

   The transitions from "DEGRADED" state are either to "ACKNOWLEDGED" state when the intent attributes are modified by the MnS consumer, or to "FULFILLED" state if the MnS producer once again fulfills the intent.

6. FULFILLMENT_FAILED: If the MnS producer determines that they cannot do anything to fulfil the intent, the state changes to "FULFILLMENT_FAILED". An intent report should be delivered to indicate that the fulfillment of the intent has ceased. The intent is not deleted unless the MnS consumer explicitly requests for it to be deleted.

   The only transition from "FULFILLMENT_FAILED" state is to "ACKNOWLEDGED" state when the intent attributes are modified by the MnS consumer.

7. TERMINATED: If the MnS consumer requests to delete the intent, the state changes to "TERMINATED". An intent report should be delivered to indicate that the intent has been terminated.

   There is no possible transition from TERMINATED state.

8. The intent may be modified when in any of the states "ACKNOWLEDGED", "COMPLIANT ", "SUSPENDED " "FULFILLED ", "DEGRADED" or "FULFILLMENT_FAILED". An intent report should be delivered to indicate that the intent has been modified.

9. The intent may be deleted when in any of the states "ACKNOWLEDGED", "COMPLIANT ", "SUSPENDED " "FULFILLED ", "DEGRADED" or "FULFILLMENT_FAILED". An intent report should be delivered to indicate that the intent has been deleted.

5.3.2.2 Requirements

REQ-IDMS_IntentReport-CON-1: The intent driven MnS producer shall have the capability to enable the MnS consumer to request intent report information.

REQ-IDMS_IntentReport-CON-2: The intent driven MnS producer shall have the capability to enable the MnS consumer to obtain intent report information with intent fulfilment information (including fulfilment status and achieved values for targets).

REQ-IDMS_IntentReport-CON-3: The intent driven MnS producer shall have the capability to enable the MnS consumer to obtain intent report information with intent conflict information.

REQ-IDMS_IntentReport-CON-4: The intent driven MnS producer shall have the capability to enable the MnS consumer to obtain intent report information with intent fulfilment feasibility check information.

REQ-IDMS_IntentReport-CON-5: The intent driven MnS producer shall have capability enabling MnS consumer to specify the content of the intent report.

REQ-IDMS_IntentReport-CON-6: The intent driven MnS producer shall have capability enabling MnS consumer to configure the frequency of the intent reporting.

REQ-IDMS_IntentReport-CON-7: The intent driven MnS producer shall have the capability enabling MnS consumer to receive reports, with different content and intervals according to its specified requirements.

REQ-IDMS_IntentReport-CON-8 The intent driven MnS producer shall have capability enabling the MnS consumer to receive an intent report when any of the following happens:

- the intent state has been changed to either "ACKNOWLEDGED" (intent has been created or modified), "COMPLIANT ", "SUSPENDED " "FULFILLED ", "DEGRADED" or "TERMINATED" (intent has been deleted).

REQ-IDMS_IntentReport-CON-9 The intent driven MnS producer shall have capability enabling MnS consumer to receive an intent report indicating the reasons associated with any of the following events: "SUSPENDED", "DEGRADED" or "TERMINATED".

REQ-IDMS_IntentReport-CON-10 The intent driven MnS producer shall have capability enabling MnS consumer to receive an intent report on the outcomes of a feasibility check, the report indicating either of 1) an indication for feasible or infeasible; 2) a detailed report indicating which intent expectations or ExpectationTargets are infeasible and corresponding reasons.

REQ-IDMS_IntentReport-CON-11 The intent driven MnS producer should have the capability to enable an MnS consumer, who expresses an intent, to specify intent report control information in the intent when creating or modifying the intent.

REQ-IDMS_IntentReport-CON-12 The intent driven MnS producer should have the capability to enable the MnS consumer to specify the conditions in the intent which triggers an intent report.

5.3.3 Intent fulfilment feasibility check

5.3.3.1 Introduction

The Intent fulfilment feasibility check can be performed in the following scenarios:

- Before MnS consumer expresses the intent expectations to MnS producer, MnS consumer may want to verify or check whether the proposed intent expectation is feasible for an MnS producer. Such feasibility check capability during intent pre-evaluation phase can be used to assist MnS consumer to generate and submit the suitable intent information for fulfilment to the intent driven MnS producer. For example, when the operator receives the service booking request for a video live broadcast service from vertical customer for a time window, the operator (as MnS consumer) may need to check with the MnS producer (e.g. RAN management system) the feasibility of the requested radio service.
- When intent driven MnS producer receives the intent instance creation or modification request from MnS consumer, the intent driven MnS producer may automatically perform feasibility check.

MnS producer performs the feasibility check activities to determine whether the submitted intent instance is feasible (including check the satisfaction of intent fulfilment and potential conflicts between one or more intent instances), and notify MnS consumer the result of feasibility check. If the result of intent fulfillment feasibility check is feasible, the MnS producer performs the service or network management tasks to satisfy the intent instance based on MnS consumer's intent fulfilment request. In case the result of intent fulfillment feasibility check is infeasible, MnS producer notifies the MnS consumer the reason of infeasibility and corresponding recommendations, then the MnS consumer decides how to handle the issue that intent is infeasible, e.g. update the intent, suspend the intent, delete the intent, etc. The information indicating which intent expectations and targets are infeasible is also included in the intent feasibility check result. This information is important for the MnS consumer to generate a feasible intent for an intent reported infeasible.

5.3.3.2 Requirements

REQ-IDMS_IntentFeasibilityCheck-CON-1: The intent-driven MnS producer should have capability to report the authorized MnS consumer the output of automatic feasibility check when receive the intent creation and modification request.

REQ-IDMS_IntentFeasibilityCheck-CON-2: The intent-driven MnS producer shall have capability to inform the authorized MnS consumer about the result of intent fulfilment feasibility check, including feasible or infeasible.

REQ-IDMS_IntentFeasibilityCheck-CON-3: The intent-driven MnS producer shall have capability to inform the authorized MnS consumer about the infeasible reason and corresponding recommendations if the result of intent fulfilment feasibility check is infeasible.

REQ-IDMS_IntentFeasibilityCheck-CON-4: The intent-driven MnS producer shall have the capability to allow an authorized MnS consumer to request a feasibility check of a given intent.

REQ-IDMS_IntentFeasibilityCheck-CON-5: The intent driven MnS producer should have the capability to provide the intent feasibility check result including the list of infeasible expectations and targets to the authorized MnS consumer.

5.3.4 Intent-related conflicts

5.3.4.1 Introduction

An intent may contain multiple intent expectations, and each intent expectation may contain multiple expectation targets. The MnS producer (or its intent handling functions) may realize that conflict between different intents or conflict between different intent expectations within an intent or conflict between different targets within an intent expectation. The intent may have conflict with other intent if any of its targets conflict with another target in another intent. The conflicts should be detected and resolved.

5.3.4.2 Detecting Intent-related conflicts

Given two expectation targets, e.g., target_1=: throughput > V1 and target_2=: interference < V2, the MnS producer will see that the targets are conflicting if one target gets degraded while attempting to achieve the other, i.e., actions taken by the producer for the two targets have mutually exclusive effects. There is intent conflict if any of its targets conflict with another target in another intents, expectation conflict if any of its targets conflict with another target in the same intent but different intent expectations, otherwise it is a target conflict if any of its targets conflict with another target in the same intent expectation. Accordingly, there are three intent related conflict scenarios:

- Target conflict, which represents the conflict between two or more expectation targets within the same intent expectation.
- Expectation conflict, which represents the conflict between two or more intent expectations within the same intent.
- Intent conflict, which represents the conflict between two or more different intents.

When such conflicts are detected, the MnS producer needs to notify the MnS consumer about the conflict, indicating the intent, intent expectations or expectation targets which give rise to the conflict. The MnS producer may also notify the MnS consumer about additional information related to the conflict (e.g., the likely impact on other expectation targets).

5.3.4.3 Resolving Intent-related conflicts

To resolve the conflict, the MnS producer may derive required solutions, such as suspension of the whole intent that contains conflicts or has conflicts with another intent. Alternatively, the MnS producer may recommend a new intent in place of the conflict intent (e.g., adding recommended expectations or targets, or termination of part of the intent). Also, the MnS producer may suggest alternative intent fulfilment e.g., by updating the execution time of the intent.

The decisions taken by the MnS producer may also be guided by the MnS consumer through explicit conflict handling guidelines. The MnS consumer may provide intent expectations and expectation targets as an ordered list for which the first intent expectations or expectation targets are the most important and should thus be prioritized in case of conflicts. Alternatively, the MnS consumer may assign explicit priorities for such intents or intent expectations or expectation

targets so that the MnS producer uses the priorities to choose the intents or intent expectations or expectation targets to be discarded in favour of others. The priority mechanism not only applies to targets, but also to the expected managed entities (listed or scoped by the object context). The MnS consumer may also send the reduced scope of the expected managed entities in intent creation or modification request. Then, when the intent cannot be fulfilled in the expected managed entities (i.e. actions taken for the expected managed entities can not fulfil a target because of the intent conflict), different actions would be taken by the MnS producer to only ensure the fulfilment of the target on the part of the expected managed entities until the intent conflict is solved.

5.3.4.4 Requirements

REQ-IDMS_IntentConflict-CON-1: The intent driven MnS producer should have the capability to inform an authorized MnS consumer about any detected intent-related conflicts, including intent conflict, expectation conflict and target conflict.

REQ-IDMS_IntentConflict-CON-2: The intent driven MnS producer should have the capability to inform an authorized MnS consumer about the decision taken by the MnS producer to resolve or handle a detected intent-related conflict.

REQ-IDMS_IntentConflict-CON-3: The intent driven MnS producer should have the capability to inform an authorized MnS consumer about the MnS producer's recommended actions that may be taken by the MnS consumer towards resolving a detected intent-related conflict, including recommendations on how to revise the intent expectations or expectation targets and to delete one of the conflicting intents.

REQ-IDMS_IntentConflict-CON-4: The intent driven MnS producer should have the capability enabling an authorized MnS consumer to provide intents with priorities to be applied when handling conflicts among intents, expectations, or targets.

5.3.5 Intent Exploration

5.3.5.1 Introduction

Before the MnS consumer expresses the intent expectation to the MnS producer (i.e. intent handling function), it is challenging for the MnS consumer to assign the values for corresponding targets and contexts to be best aligned with both MnS consumer's expectation and MnS producer's capabilities. This depends on the current resource situation and capabilities of the system (availability of MnS Producer resources in certain area, time, etc.). For example, the high expectation on target value improves MnS consumer's satisfaction but may be out of MnS producer's capabilities (e.g. insufficient radio resources). In other aspect, the low expectation on target value can be achieved by MnS producer, but reduces MnS consumer's satisfaction and may not be best aligned with the MnS producer' capabilities (e.g. radio resources are not effectively used). So MnS consumer needs to find out if the expected intent expectations are realistic and best aligned with MnS producer's capability before expressing the intent expectation to MnS producer.

Especially for an intent with multiple targets, it is more challenging for MnS consumer to find out best combination of the values for multiple targets in an intent. For example, it is difficult for MnS consumer to assign the best values for rANEnergyConsumptionTarget and aveDLRANUEThptTarget in RAN Energy saving intent defined in clause 6.2.2 to optimally balance the energy consumption expectation and RAN UE throughput performance.

Based on above analysis, it is important to introduce the MnS capability to enable MnS consumer to explore the best values for intent targets and contexts within a specific intent during intent pre-evaluation phase (i.e. before MnS consumer express the intent expectation to be fulfilled) to learn more about the MnS producer's capabilities. This would allow the MnS consumer to determine the values for intent targets and contexts which are best aligned with both MnS consumer's expectation and MnS producer's capabilities. Following are the two potential scenarios for intent exploration during intent pre-evaluation phase:

- **Scenario#1:** MnS consumer requests the MnS producer to explore the best value for a given target or context in an intent with a target name or attribute name specified. The MnS consumer may specify the values for other context attributes and targets in the intent to limit the exploration result.
- **Scenario#2:** MnS consumer requests the MnS Producer to explore the best combination of the values for multiple targets or contexts in an intent with a list of target names and context attribute names specified. MnS consumer may specify the values for other context attributes and targets in the intent to limit the exploration result. It is MnS producer's decision to provide one or multiple best combination of the values which are best aligned with MnS producer's capabilities. For example, for RAN energy saving intent which includes the energy

consumption target and RAN UE throughput target, MnS producer may provide two best combination values for these two targets. One provides the better energy consumption but lower RAN UE throughput, while another provides the better RAN UE throughput but the energy consumption is not good compared to the first one.

MnS producer (i.e. intent handling function) can perform simulation/evaluation activities to provide the possible values for one or multiple targets and contexts in an intent accounting for all intents applied to the expected network. The expected network will not act on the intent exploration request.

5.3.5.2 Requirements

REQ-IDMS-IntentExploration-CON-1: The intent driven MnS producer should have the capability enabling the MnS consumer to request to explore the best value for a given target or context in an intent.

REQ- IDMS-IntentExploration-CON-2: The intent driven MnS producer should have the capability to explore the best combination of the values for multiple targets or contexts in an intent.

5.3.6 Intent degradation based on expectation preference

5.3.6.1 Introduction

Sometimes an MnS consumer may express multiple intent expectations with different preferences for the same requirement within the same intent. The MnS producer should fuifil the IntentExpectation(s) based on the preference information. For example, the MnS consumer proposes requirements that can be translated to the following expectation:

- IntentExpectation1, the consumer has preference on E2E latency.
- IntentExpectation2, the consumer has higher preference on air latency.
- IntentExpectation3, minimum UL/DL UE throughput larger than 50 Mbps.

The MnS consumer requires the producer to fulfil all expectations above but has extra preference on latency feature. The consumer prefers lower air latency as indicated in the IntentExpectation2 and will be happy if the producer can guarantee IntentExpectation2. Such information can be utilized by the MnS producer to help resolve intent degradation automatically.

EXAMPLE: The MnS producer can degrade the intent automatically by just satisfying IntentExpectation1 and IntentExpectation3 when IntentExpectation2 cannot be satisfied.

Based on the analysis above, it is significant to allow the MnS consumer to express its preference in an intent. Then, the MnS producer can leverage such preference information to address possible intent degradation.

5.3.6.2 Requirements

REQ-IDMS_IHCO-CON-1 The intent driven MnS producer should have the capability to enable the MnS Consumer to express its preference on expectations and targets in an intent.

5.3.7 Enablers for Intent Fulfilment

5.3.7.1 Introduction

An intent focuses more on describing the "What" needs to be achieved but less on "How" that outcomes should be achieved. In this way, MnS producer (intent handling function) takes the important network management control, however, MnS consumer may need to know some information of how MnS producer fulfil the intent to be sure that MnS producer works in an expected manner. MnS producer may modify the intent or delete the intent if MnS producer doesn't works in an expected manner.

For example, for a radio network optimization intent, MnS consumer may need to know:

- Which base stations are updated to fulfil the intent.

Based on above analysis, it is important to introduce the MnS capability to enable MnS consumer to obtain the information of enablers for intent fulfilment to be sure that MnS producer works in an expected manner.

5.3.7.2 Requirements

REQ-IDMS-EnablerInfo-1: The intent driven MnS producer should have the capability to enable the MnS consumer to request to obtain the additional information of enablers for intent fulfilment.

5.3.8 Intent Utility Function

5.3.8.1 Introduction

In some cases, it may not be obvious to the MnS producer how to select from multiple candidate solutions for fulfilling an intent. For example, an MnS consumer expressing multiple intents may want to provide further preference information to the MnS producer to improve overall fulfilment, rather than rely on simple satisfied/unsatisfied indication.

Extra information provided in an intent can indicate preferences between expectation targets and assist the MnS producer in balancing between acceptable solutions.

This is especially relevant when no solution fully satisfies all intents or meets the expectations of a particular intent. In such situations, balancing trade-offs among different intents becomes necessary.

For example, a network performance function could be used to quantify network performance relative to latency, providing a means for the MnS Consumer "to quantify an intent's relative value as higher when the latency is lower":

EXAMPLE 1: U1 = f(throughput, latency, packet loss); a single utility function defined for an intent

EXAMPLE 2: U2 = f(latency) and U3=f(throughput); multiple utility functions defined for an intent, e.g. for different expectations

An MnS consumer could express this in the form of an Intent Utility Function along with their expected target value. This assists the MnS producer(s) in fulfilling an MnS consumer's intents, as it provides an indicator to the MnS Producer of the satisfaction level of the intent fulfilment. The intent utility function can be for multiple intents or as a part of an intent.

Utility functions can be used for intent negotiation.

5.3.8.2 Definition

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 levels of fulfilment. Their basic components include:

- variables: of the function to quantify specific aspects of the fulfilment, e.g., expectation target for network performance.
- weights: to define the relative importance of each variable, e.g., for network performance, a variable representing low latency might be assigned a higher weight than throughput.
- function: the mathematical function to be applied to the variables, e.g., linear, logarithmic, polynomials, scalars.
- result: the output of the function which represents the utility level achieved, i.e., the satisfaction of the current fulfilment based on the acceptability of potential solutions defined by the consumer.

MnS producer(s) can use Intent Utility Function information to assess the acceptability of potential solutions, 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 MnS consumer may also specify that an existing (i.e., predefined) utility function is used for one or

multiple intent(s). Predefined utility functions may be vendor specified (e.g., system pre-installed) and/or specified by the MnS consumer.

A single intent may contain multiple intent expectations, each having its own utility function. In such cases, aggregation of these utility functions can be used to represent the overall utility of the intent. Normalization may also be needed when dealing with utility functions expressed in different scales (e.g., latency in milliseconds, throughput in Mbps, energy consumption in Joules).

Figure 5.3.8.2-1 represents a case where an MnS Consumer expresses an intent (A) with expectation which should be less than **a**, while expectation of another intent (B) should be greater than **b**. There may be no solution to fulfil both expectations. However, different solutions may be available for each intent which leads to more or less preferred outcomes, as indicated by the black and red circles. While both solutions would fulfil intent A, neither would fulfil intent B, and there is limited ability for the MnS producer to decide which would be better from MnS consumer perspective.

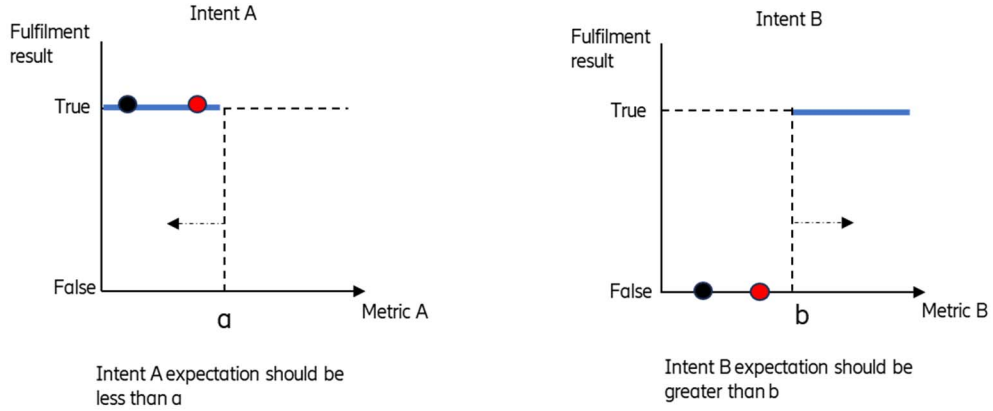


Figure 5.3.8.2-1: Intents with binary fulfilment result

Figure 5.3.8.2-2 represents a case where an MnS Consumer defines Utility Functions A and B. Each provides the MnS producer with a means to calculate a utility value based on the metrics A and B. These utility results can be used by the MnS producer to select from different solutions to improve the overall utility across multiple intents.

For example, the solution with the red circle yields a better outcome since it can maximize Utility Function A and has higher value for Utility Function B than the black solution. From the MnS consumer's perspective, such a case is considered "quantitative fulfilment", as the MnS producer has a means by which to apply a quantitative value to each solution and use such information to better select on behalf of the consumer.

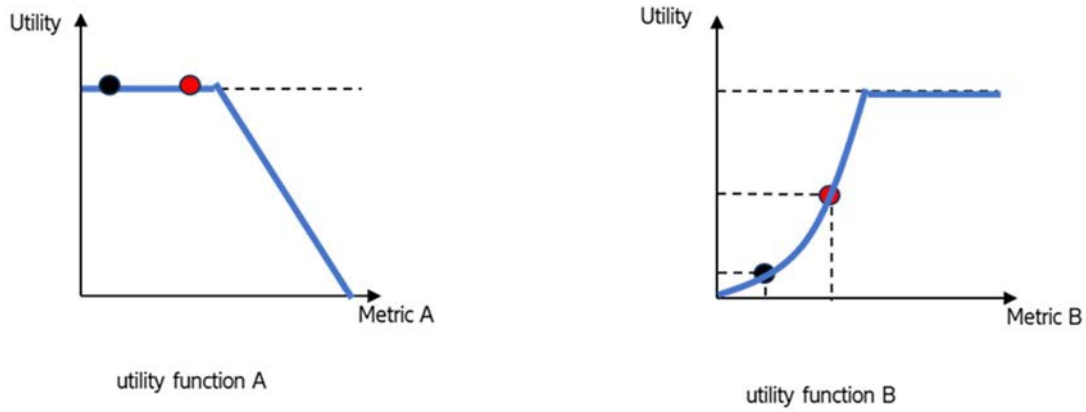


Figure 5.3.8.2-2: Example 1 of intent with differentiated solution preference through utility function

Figure 5.3.8.2-3 represents a case where two different intents consider the same metric A but use different utility functions (Utility Function A and Utility Function B). In this scenario, no solutions can fully maximize either of the utility results, and therefore, none of the intents can be completely satisfied. The MnS producer may choose to prioritize

the red circle solution as its aggregate value (utility function A and utility function B) is higher than the one for the black circle solution. This would contribute to a better overall utility level across multiple intents.

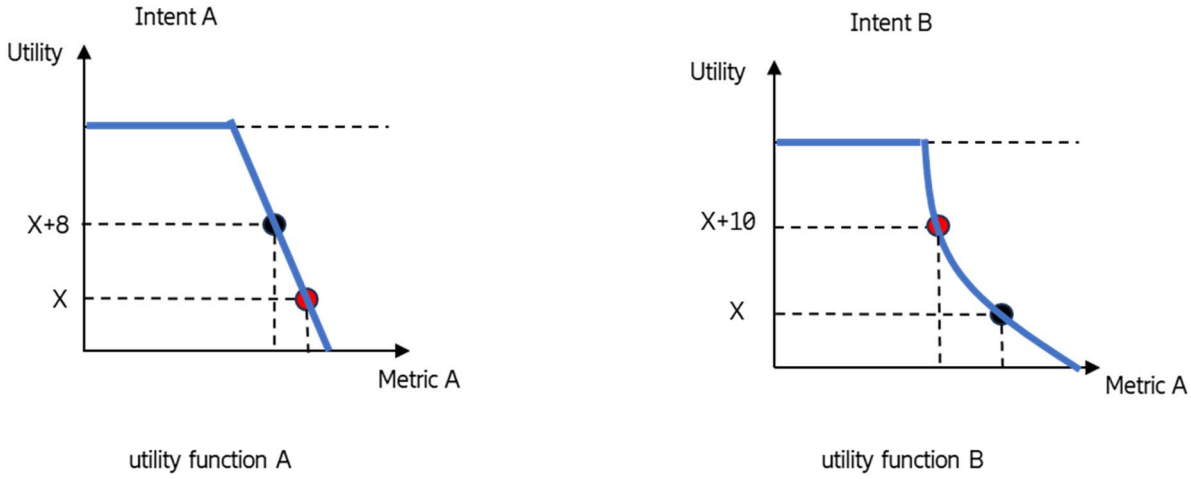


Figure 5.3.8.2-3: Example 2 of intents with differentiated solution preference through utility function

5.3.8.3 Requirements

REQ-IDMS_IntentUtility-CON-01: The intent driven MnS producer should have the capability to advertise its support for allowing MnS Consumers to express utility functions.

REQ-IDMS_IntentUtility-CON-02: The intent driven MnS producer should have the capability to advertise the mechanisms (e.g. data and format) by which an MnS Consumer can express utility functions.

REQ-IDMS_IntentUtility-CON-03: The intent driven MnS producer should have the capability to report potential outcomes including results for utility functions when applicable.

REQ-IDMS_IntentUtility-CON-04: The intent driven MnS producer should allow an authorized MnS consumer to specify utility functions as part of an intent.

REQ-IDMS_IntentUtility-CON-05: The intent driven MnS producer should allow an authorized MnS consumer to reference utility functions defined external to the intent.

5.3.9 Negotiation on fulfilment of intents

5.3.9.1 Introduction

For a given intent, an MnS Consumer may express a feasible intent, but the producer may have 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 preferable from the MnS Consumer´s point of view.

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 number of users.

The MnS consumer may have concerns on the possible impact of the selected solutions chosen by the MnS 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.

Intent negotiation is a set of procedures which are applicable throughout the life cycle of an intent. Several negotiations are possible for a feasible intent, many employing interactions that are similar.

NOTE 1: Some of these negotiations can also be applicable during the feasibility check process.

Intent negotiation procedures can be initiated by MnS Consumers. Consequently, if an MnS Consumer does not support intent negotiation capabilities, it does not trigger such procedures.

5.3.9.2 Checking for fulfillable outcomes

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

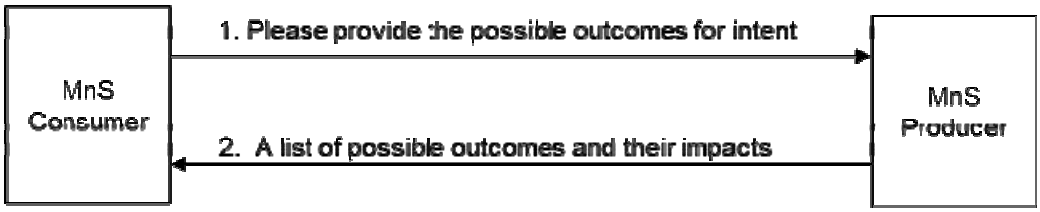


Figure 5.3.9.2-1: MnS consumer requests and receives a list of possible fulfillable outcomes on an intent

Many dynamic factors impact possible outcomes. As a result, it is unrealistic to expect all potential outcomes to be reported. The list of possible outcomes an MnS Producer may provide is left to implementation.

Subsequently, the MnS producer provides a report indicating what is fulfillable for each intent aspect (intentExpectation and expectationTarget) within that intent. Since different possible fulfillable outcomes may have different impacts, the report should include the relative impacts of each outcome. The relative impact is defined as the MnS Producer´s computed impact to existing expectations of fulfilling one or more expectations within the requested intent on the expectations within another intent from the same MnS Consumer. For example, an energy-saving intent expectation can be fulfilled, causing a degradation on the coverage intent expectation. Relative impact from different MnS Consumers shall not be reported since different MnS Consumers are not to be aware of each other's intents.

NOTE 1: The mechanism by which MnS Consumer can indicate a preference for a specific possible outcome is out of scope of this TS.

NOTE 2: How the Producer would report "the relative impacts of each outcome" is FFS.

5.3.9.3 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.

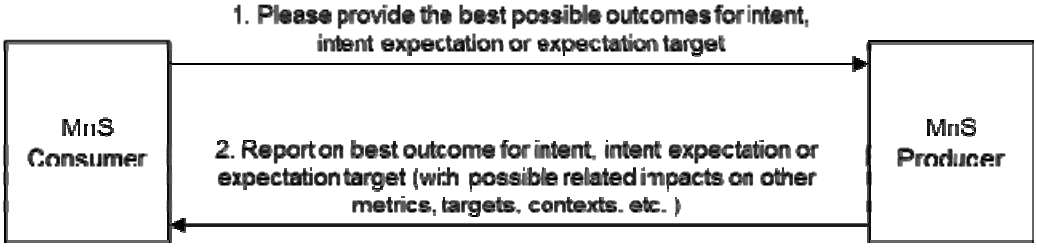


Figure 5.3.9.3-1: Checking for best possible outcome on intent or intent expectation or expectation target

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 in 2 possible scenarios as follows:

- The request is to evaluate specific expectation target(s) 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 specific expectation target(s) that maintains the other expectation targets to within the ranges specified in the intent.

- 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.

The MnS producer should support a possible outcomes report that lists the possible outcomes for any of the two scenarios, the report including the relative impact on other targets in the intent or on other metrics and contexts.

5.3.9.4 MnS producer to provide information about possible fulfilment of the intent

The MnS consumer wants and requests to know for a specific property what the MnS producer recommends that could be applied for particular intent characteristics, i.e. the MnS consumer wants to query MnS producer for recommendations for particular intent characteristics. In one case, 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 their intent or another of their intents to make the intent fulfillable. In the other case, the MnS consumer may already have the preference for the solution, i.e., a change to a specific attribute (a target, expectation object, or context). In this case, the MnS consumer sends an intent that contains the MnS consumer's preference for solution to the MnS producer and requests to fulfil intent. When the targets cannot be fulfilled, the solutions recommended by the MnS producer will only involve changes to that attribute. Then, the MnS consumer can receive recommendations from MnS producer and use the response to modify their existing intents in an attempt to improve the ability to fulfil the new intents.

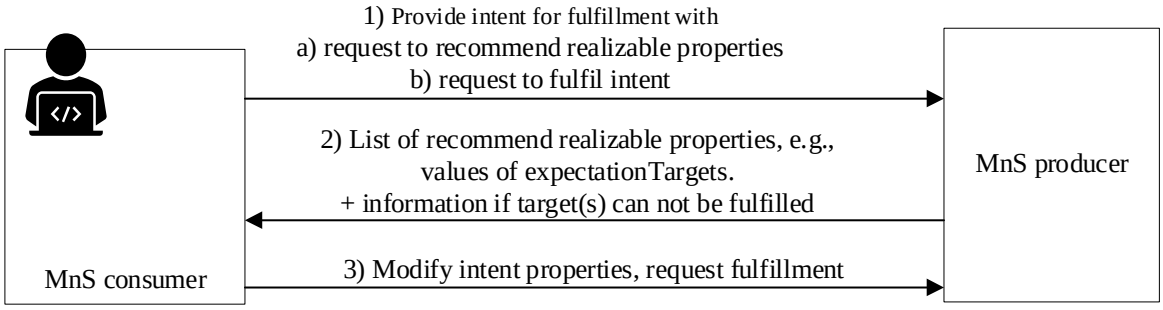


Figure 5.3.9.4-1: Enabling the MnS consumer to request and receive a recommendation on the possible fulfillable intent properties prior to or in case of failure of fulfilment

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

5.3.9.5 MnS consumer advises on preferred alternatives

The MnS consumer wants an intent to be 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 coverage area as a result of the energy consumption.

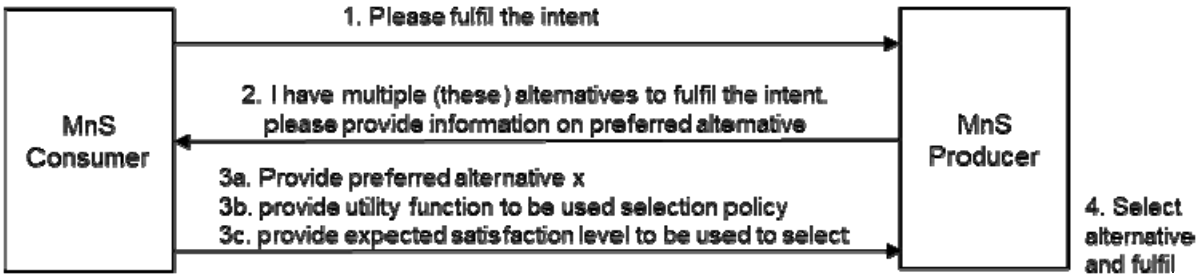


Figure 5.3.9.5-1: MnS consumer advises the MnS producer on the preferences among alternatives at the MnS producer

After the MnS consumer creates an intent to be fulfilled, the MnS producer determines that there are multiple alternatives, so the MnS producer 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 their intents and intent expectations.
- An information requesting 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 the 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 defined as falling in a range. Correspondingly, values lower than 100 indicate how far from the outcome is from the maximum satisfaction.

5.3.9.6 Requirements

REQ-IDMS_IntentFulfilmentNegotiation-CON-01: The intent-driven MnS producer should have the capability to provide possible outcomes to the MnS Consumer.

REQ-IDMS_IntentFulfilmentNegotiation-CON-02: The intent-driven MnS producer should have the capability enabling MnS Consumer to decide appropriate outcome for fulfilment based on possible outcomes.

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

REQ-IDMS_IntentFulfilmentNegotiation-CON-03: The intent driven 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 achieves 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 impact refers to outcomes of the contexts and targets that MnS consumers have not explicitly pointed out. The exact characterization of what is reported is For Further Study. E.g. for an expectation target on energy consumption, the impact may include coverage area as a result of the energy consumption.

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

REQ-IDMS_IntentFulfilmentNegotiation-CON-04: 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 supports for the provided intent, intent expectations, or expectation Targets and the expected relative impacts of the different alternatives.

REQ-IDMS_IntentFulfilmentNegotiation-CON-05: 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 supports for the provided intent, intent expectations, or expectation Targets.

REQ-IDMS_IntentFulfilmentNegotiation-CON-06: The intent driven MnS producer should support a capability enabling an authorized MnS consumer to provide information on a description of its preferences (e.g. in form of a utility function or satisfaction index) that should be used by the MnS producer to select among the alternatives available at the MnS producer.

REQ_IDMS_IntentFulfilmentNegotiation-CON-07: The intent driven MnS producer should support a capability to request the MS consumer to provide an evaluation of the selected alternative among the MnS producer's alternatives based on the expected relative impacts of the different alternatives.

REQ_IDMS_IntentFulfilmentNegotiation-CON-08: The intent driven MnS producer should support a capability enabling an MnS consumer to provide an evaluation of the MnS producer's alternatives.

REQ_IDMS_IntentFulfilmentNegotiation-CON-09: The intent driven MnS producer should support a capability to inform an authorized MnS consumer that their selected alternative has been applied.

REQ_IDMS_IntentFulfilmentNegotiation-CON-10: The intent driven MnS producer should support a capability to inform an authorized MnS consumer that no more improvement to intent fulfillment is possible with the applied alternative and request the MnS consumer to provide extra information which helps improve satisfaction.

REQ_IDMS_IntentFulfilmentNegotiation-CON-11: 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_IDMS_IntentFulfilmentNegotiation-CON-12: The intent driven MnS producer should support a capability enabling an authorized MnS consumer to provide information on the degree of satisfaction which the MnS producer uses to differently attempt the fulfillment.

REQ-IDMS_IntentFulfilmentNegotiation-CON-13: The MnS producer should support a capability to provide an intent report including information on what is the best possible outcome for each intentExpectation or expectationTarget within that intent and the relative impact of achieving that outcome.

REQ-IDMS_IntentFulfilmentNegotiation-CON-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 for selected intent expectations and or expectation targets, while keeping other intent expectations unmodified.

REQ-IDMS_IntentFulfilmentNegotiation-CON-15: 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 of their intents to make the intent fulfillable.

NOTE 5: Further discussion is needed for this requirement.

REQ-IDMS_IntentFulfilmentNegotiation-CON-16: 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

NOTE 6: The definition of outcomes is to be clarified

REQ-IDMS_IntentFulfilmentNegotiation-CON-17: The MnS producer may support a capability to provide implicit intent, i.e., the additional intent expectation target and contexts, which includes the information that MnS consumers have not explicitly pointed out in its intent.

6 Stage 2 definition for Intent Driven Management

6.1 Management operation for intent driven management (MnS component type A)

The operations (e.g. createMOI operations) and notifications (e.g. notifyMOIcreation) of generic provisioning MnS defined in 3GPP TS 28.532 [3] can be used for intent driven management, including intent lifecycle management, intent report management and intent handling capability obtaining. The Intent, IntentReport and IntentHandlingFunction can be treated as Managed Object instances.

Following is the IS to support intent lifecycle management:

Table 6.1-1: IS to support intent lifecycle management

intent lifecycle management	IS operation
Create an intent	createMOI operation
Delete an intent	deleteMOI operation
Modify an intent	modifyMOIAttributes operation
Query an intent	getMOIAttributes operation
Activate an intent	modifyMOIAttributes operation
Deactivate an intent	modifyMOIAttributes operation

Following is the IS to support intent report management:

Table 6.1-2: IS to support intent report management

intent lifecycle management	IS operation
Query an intent report	getMOIAttributes operation
Subscribe an intent report	createMOI operation
Notify an intent report	notifyMOIAttributeValueChanges notification
Unsubscribe an intent report	deleteMOI operation
Query an intent report subscription	getMOIAttributes operation

Following is the IS to support intent handling capability obtaining:

Table 6.1-3: IS to support intent handling capability obtaining

intent lifecycle management	IS operation
Query intent handling capability	getMOIAttributes operation

Following is the IS to support intent negotiation management in pre-evaluation phase and fulfilment phase:

Table 6.1-4: IS to support intent negotiation management

intent lifecycle management	IS operation
Intent feasibility check	createMOI operation
Intent exploration	createMOI operation
Intent fulfilment negotiation	createMOI operation
	modifyMOIAttributes operation

6.2 Information model definition for Intent (MnS component typeB)

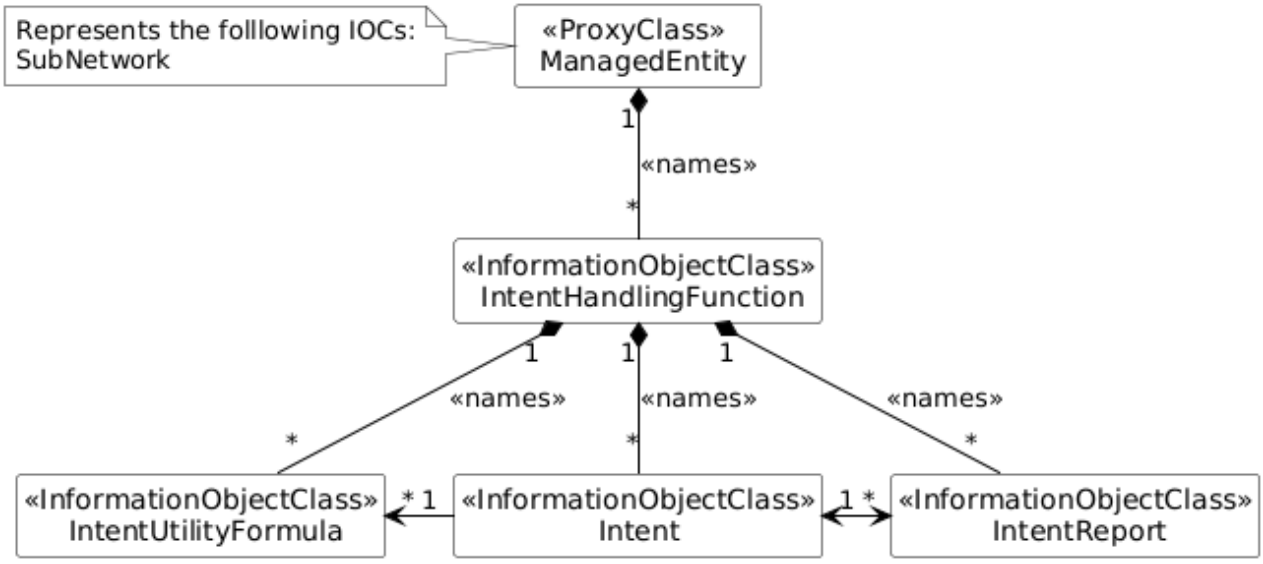
6.2.1 Generic Information model definition

6.2.1.0 Imported information entities and local labels

3GPP TS 28.622 [6], dataType, DateTime	DateTime
3GPP TS 28.622 [6], dataType, GeoArea	GeoArea
3GPP TS 28.658 [10], dataType, PLMNId	PLMNId
3GPP TS 28.622 [6], dataType, TimeWindow	TimeWindow
3GPP TS 28.622 [6], dataType, GeoCoordinate	GeoCoordinate
3GPP TS 28.622 [6], choice, SchedulingTime	SchedulingTime
3GPP TS 28.622 [6], IOC, Top	Top
3GPP TS 28.622 [6], IOC, SubNetwork	SubNetwork

6.2.1.1 Class diagram

6.2.1.1.1 Relationship



NOTE: Void

Figure 6.2.1.1.1-1: Relationship UML diagram for intent driven management

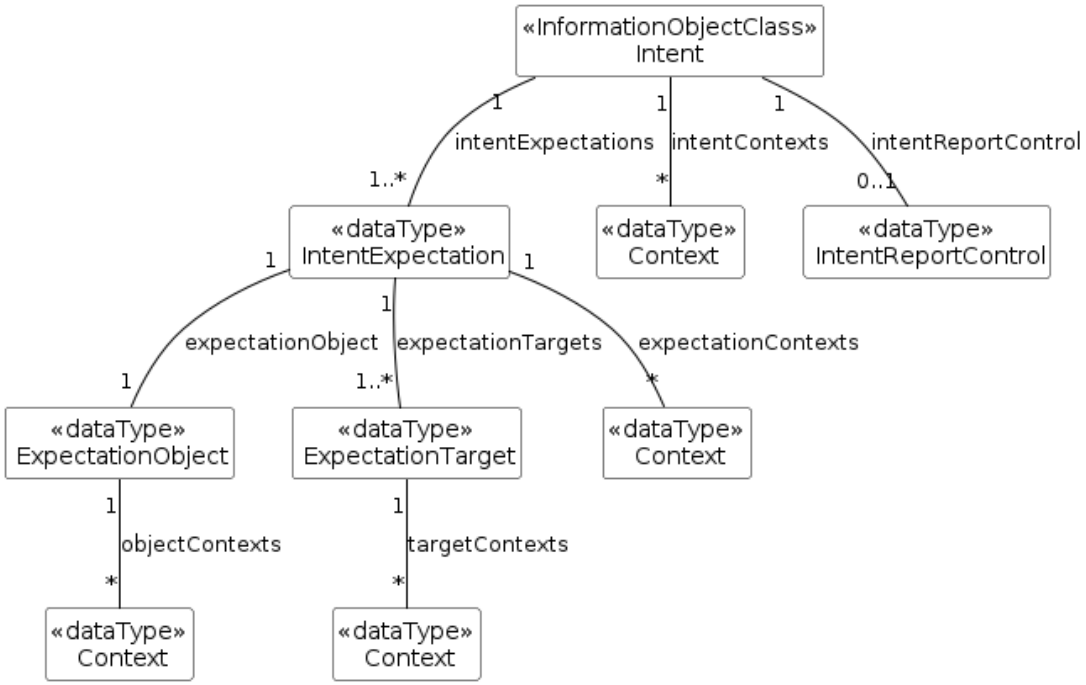


Figure 6.2.1.1.1-2: Relationship UML diagram for intent

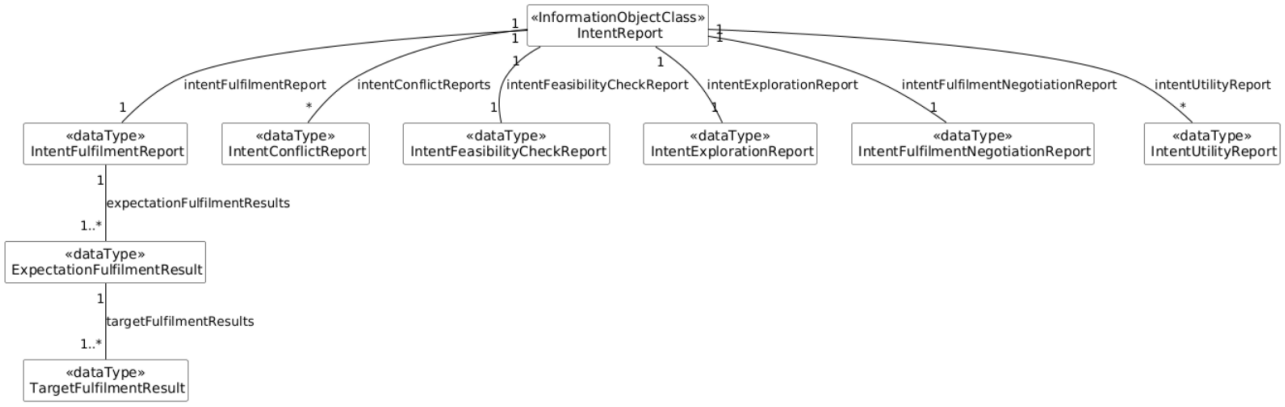


Figure 6.2.1.1.1-3: Relationship UML diagram for intent report

6.2.1.1.2 Inheritance

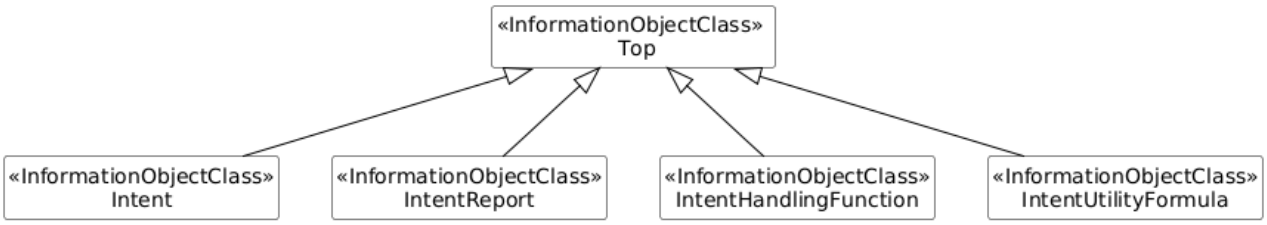


Figure 6.2.1.1.2-1: Inheritance UML diagram for intent driven management

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 suspension mechanism. 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 "intentReportControl" indicates the intent report control and subscription information. MnS consumer needs to specify the values for the attribute "intentReportControl" when the MnS consumer wants to obtain the intent report with customized requirements by default, instead of triggering a separate subscription action.

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 `implicitIntentIndex` used to indicate whether the discovery of implicit information is enabled. In case the MnS consumer wants the MnS producer to discover the implicit information like additional intent expectation which is not explicitly pointed out, the `implicitIntentIndex` will be `"True".`

The Intent IOC includes `expectationSelectivity` used to define how to select among the stated `intentExpectations.` It enables the MnS consumer to trigger evaluation of different alternative intentExpectations. The MnS consumer can provide alternatives descriptions of service-offers as intent expectations that describes the different candidate characteristics of the desired service from the MnS consumer's point of view that the MnS consumer wants to be validated. By providing the intent containing an `expectationSelectivity` for feasibility checking, the MnS consumer indicates how the set of candidate services expressed in the intentExpectations are to be validated, i.e., "ALL_OF", "ONE_OF", "ANY_OF" the intentExpectations.

6.2.1.2.1.2 Attributes

The `Intent` IOC includes attributes inherited from `Top` IOC (defined in 3GPP TS 28.622 [6]) 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
intentMgmtPurpose	M	T	T	F	F
contextSelectivity	O	T	T	F	F
expectationSelectivity	O	T	T	F	F
intentContexts	O	T	T	F	F
intentReportControl	M	T	T	F	F
intentPriority	O	T	T	F	T
intentAdminState	CM	T	T	F	F
intentPreemptionCapability	CM	T	T	F	F
implicitIntentIndex	M	T	T	F	F
consumerSatisfactionIndexThreshold	CO	T	T	F	F
Attribute related roles					
intentReportReference	M	T	F	F	F
intentUtilityFormulaRef	CM	T	T	F	T

6.2.1.2.1.3 Attribute constraints

Name	Definition
intentAdminState	Condition: MnS consumer-suspension mechanism is supported.
intentPreemptionCapability	Condition: The preemption mechanism is supported.
intentUtilityFormulaRef	Condition: Intent Utility Function is supported.
consumerSatisfactionIndexThreshold	Condition: Customer Satisfaction Index is supported.

6.2.1.2.1.4 Notifications

The common notifications defined in clause 6.2.1.5 are valid for this IOC, without exceptions or additions.

6.2.1.2.2 IntentReport <<InformationObjectClass>

6.2.1.2.2.1 Definition

This IOC represents intent report information from MnS producer to MnS consumer. The IntentReport instance is created by MnS producer as a result of the IntentReportControl (specified in the Intent IOC) that specifies the types and conditions for reporting. When the MnS producer deletes an intent instance based on a request from MnS consumer, the corresponding intent report instance is also deleted by MnS producer automatically.

The `IntentReport` IOC includes

- `intentFulfilmentReport`, which represents the properties of fulfillment information for expectation target, intent expectation, and the whole intent. The fulfilmentReport will be observed from the start of each observation period (specified in the IntentReportControl data type), then at the end of each observation period, the corresponding values will be derived and configured.
- `intentConflictReports`, which represents detected conflict information, including conflict type (i.e., intent conflict, expectation conflict and target conflict) and possible solution recommendations to address the conflicts.
- `intentFeasibilityCheckReport`, which indicates that the intent is feasible or infeasible. Intent feasibility check information is provided after MnS producer automatically performs feasibility check when receiving the intent creation and modification request from MnS consumer.

- intentExplorationReport, which represents the properties of the intent exploration result during negotiation in intent pre-evaluation phase. Intent exploration result is provided after MnS producer performs intent exploration pre-evaluation process as requested by MnS consumer.

- intentFulfilmentNegotiationReport, which represents the properties of intent negotiation information negotiation in intent fulfilment phase.. Intent fulfilment negotiation information is provided to MnS Consumer by the MnS producer during intent fulfilment phase.

- intentUtilityReports, which provides the results of applicable Intent Utility Functions. The results are calculated and reported per utility function applicable to the intent specified in intentReference.

Each instance of IntentReport IOC can contain one or any combination of intentFulfilmentReport, intentConflictReport, intentFeasibilityCheckReport, intentExplorationReport, intentFulfilmentNegotiationReport and intentUtilityReport.

Different MnS consumers can use the "getMOIAttributes" operation to query different attributes of the IntentReport <<IOC>> to obtain corresponding intent report information (including intentFulfilmentReport, intentConflictReport, intentFeasibilityCheckReport, intentExplorationReport, intentFulfilmentNegotiationReport and intentUtilityReport).

Different MnS consumers can subscribe attribute value change notifications for IntentReport <<IOC>> to obtain the notification for different intent report information.

6.2.1.2.2.2 Attributes

The IntentReport <<IOC>> includes attributes inherited from Top IOC (defined in TS 28.622 [6]) and the following attributes

Table 6.2.1.2.2.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
intentFulfilmentReport	CM	T	F	F	T
intentConflictReports	CM	T	F	F	T
intentFeasibilityCheckReport	CM	T	F	F	T
intentExplorationReport	CM	T	F	F	T
intentFulfilmentNegotiationReport	CM	T	F	F	T
intentUtilityReports	CM	T	F	F	T
lastUpdatedTime	M	T	F	F	T
Attribute related to roles					
intentReference	M	T	F	F	F

6.2.1.2.2.3 Attribute constraints

Table 6.2.1.2.2.3-1

Name	Definition
intentFulfilmentReport	Condition: intent fulfilment information reporting capability is supported by intent handling function
intentConflictReports	Condition: intent conflict information reporting capability is supported by intent handling function
intentFeasibilityCheckReport	Condition: intent feasibility check information reporting capability is supported by intent handling function
intentExplorationReport	Condition: intent exploration information reporting capability is supported by intent handling function
intentFulfilmentNegotiationReport	Condition: intent fulfilment negotiation information reporting capability during fulfilment phase is supported by the intent handling function
intentUtilityReports	Condition: Intent Utility Function is supported

6.2.1.2.2.4 Notifications

The common notifications defined in clause 6.2.1.5 are valid for this IOC, without exceptions or additions.

6.2.1.2.3 IntentHandlingFunction <<InformationObjectClass>>

6.2.1.2.3.1 Definition

This IOC represents the intent handling capabilities can be supported by a specific intent handling function of MnS producer. `IntentHandlingFunction` instances are created by the MnS producer or are pre-installed, and also are modified, deleted by the MnS producer if needed. MnS consumers cannot request to create, modify or delete `IntentHandlingFunction` instances.

An MnS consumer can query the `IntentHandlingFunction` IOC to obtain the intent handling capability information for a specific intent handling function of MnS producer. Based on the obtained intent handling capability information and management requirements, MnS consumer generates the corresponding intent information and sends it to MnS producer. The intent information includes the expectation object and expectation targets which are not only supported by the obtained intent handing capabilities, but also satisfy the MnS consumer's management requirements.

The MnS consumer also can use the DN of IntentHandlingFunction instance to query all Intent instances handled by a specific intent handling function.

6.2.1.2.3.2 Attributes

The IntentHandlingFunction <<IOC>> includes attributes inherited from `Top` IOC (defined in TS 28.622 [6]) and the following attributes

Table 6.2.1.2.3.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`intentHandlingCapabilityList`	M	T	F	F	T
`supportedUtilityList`	CM	T	F	F	T

6.2.1.2.3.3 Attribute constraints

Table 6.2.1.2.3.3-1

Name	Definition
`supportedUtilityList`	Condition: Intent Utility Function is supported.

6.2.1.2.3.4 Notifications

The common notifications defined in clause 6.2.1.5 are valid for this IOC, without exceptions or additions.

6.2.1.2.4 IntentUtilityFormula <<InformationObjectClass>>

6.2.1.2.4.1Definition

The `IntentUtilityFormula` <<IOC>> represents a utility function instance.

The attribute `utilityFunctionId` specifies the function to be performed. The value identifies a function from the list of utility functions supported by the `IntentHandlingFunction`.

The attribute `utilityParameterList` provides the values for the parameters required for the calculation of the function specified in `utilityFunctionId`.

The attributes `utilityScale` and `utilityOffset` are applied to the utility function result prior to perform applicable scaling and offset to the reported result. For example, the same utility function when applied to multiple intents will have different results when defined with different scaling and/or offset values.

6.2.1.2.4.2Attributes

The `IntentUtilityFormula` <<IOC>> includes attributes inherited from Top IOC (defined in TS 28.622 [6]) and the following attributes.

Table 6.2.1.2.3.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`utilityFunctionId`	M	T	T	F	F
`utilityParameterList`	M	T	T	F	F
`utilityScale`	O	T	T	F	F
`utilityOffset`	O	T	T	F	F

6.2.1.2.4.3Attribute constraints

None.

6.2.1.2.4.4Notifications

The common notifications defined in clause 6.2.1.5 are valid for this IOC, without exceptions or additions.

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 e`xpectationContexts`.

The `IntentExpectation` <<dataType>> includes a `preferenceWeight` used to represent the preference weight on Intent Expectations to allow the consumer to express preference information on intent expectation.

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
`expectationContexts`	O	T	T	F	F
`preferenceWeight`	O	T	T	F	F
NOTE: The scenario specific 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.

6.2.1.3.1.4 Notifications

The notifications specified for the IOC using this <<dataType>> for its attribute(s), shall be applicable.

6.2.1.3.2 ExpectationObject <<dataType>>

6.2.1.3.2.1 Definition

The `ExpectationObject` <<dataType>> represents the Object to which the `IntentExpectation` should apply.

6.2.1.3.2.2 Attributes

The `ExpectationObject` includes the following attributes.

Table 6.2.1.3.2.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`objectType`	CM	T	T	F	F
`objectInstance`	CM	T	T	F	F
`objectContexts`	O	T	T	F	F

6.2.1.3.2.3 Attribute constraints

Table 6.2.1.3.2.3-1

Name	Definition
`objectType`	Condition: The intent expectation is not for a specific object instance or MnS consumer have no knowledge of the DN of this specific object instance.
`objectInstance`	Condition: The intent expectation is for a specific object instance and MnS consumer have the knowledge of the DN of this specific object instance.

6.2.1.3.3 ExpectationTarget <<dataType>>

6.2.1.3.3.1 Definition

The `ExpectationTarget` <<dataType>> represents the target of the `IntentExpectation` that are required to be achieved.

The `ExpectationTarget` <<dataType>> includes a `contextSelectivity` used to define how to select among the stated `targetContexts.`

The `ExpectationTarget` <<dataType>> includes a `preferenceWeight` used to represent the preference weight on Expectation Targets to allow the consumer to express preference information on expectation target.

6.2.1.3.3.2 Attributes

The `ExpectationTarget` includes the following attributes.

Table 6.2.1.3.3.2-1

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

6.2.1.3.3.3 Attribute constraints

None.

6.2.1.3.3.4 Notifications

The notifications specified for the IOC using this <<dataType>> for its attribute(s), shall be applicable.

6.2.1.3.4 Context <<dataType>>

6.2.1.3.4.1 Definition

The `Context` <<dataType>> represents the properties of a context. A context describes the condition. The context may apply to the intent, the intent expectation, the expectation targets or to the expectation object.

6.2.1.3.4.2 Attributes

The `Context` includes the following attributes.

Table 6.2.1.3.4.2-1

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

6.2.1.3.4.3 Attribute constraints

None.

6.2.1.3.5 FulfilmentInfo << dataType >>

6.2.1.3.5.1 Definition

This dataType represents the properties of a specific fulfilment information for an aspect of the intent (i.e. either an expectation, a target or the whole intent). The fulfilment information describes the MnS producer's assessment of the degree to which a specific aspect of the intent is being fulfilled. The MnS consumer may however assess the fulfilment differently, e.g. the MnS consumer may evaluate the delivered outcome or network state to compute its fulfilment satisfaction.

The `fulfilmentStatus` field indicates whether the intent is being fulfilled or not being fulfilled. The possible values of the fulfilment include:

- `NOT_FULFILLED`: This is the default status for any aspect of the intent and the `fulfilmentStatus` remains as "`NOT_FULFILLED`" until the actions undertaken meet the requirements as stated by the MnS consumer.
- `FULFILLED`: This is the status if the MnS producer considers that the intent, expectation or target is being fulfilled as desired by the MnS consumer that created the intent.

The degree of fulfilment of an intent with the NOT_FULFILLED status may have multiple explanations and related states. These different progress states and conditions are recorded in the notFulfilledState field. notFulfilledState is present only when FulfilmentInfo is implemented for IntentFulfilmentInfo. The possible values of the notFulfilledState include:

- ACKNOWLEDGED: this is the default state and is the initial notFulfilledState right after the intent has been received and its instance has been created.
- COMPLIANT: this is the state after the feasibility check has been run for the intent and the intent is accepted as being compliant for fulfilment.
- DEGRADED: this is the state if an intent that was previously fulfilled but after a period of observation it is found not be meeting the initially stated requirements.
- SUSPENDED: this is the state if the MnS producer or MnS consumer decides to suspend the fulfilment of the intent, expectation or target for whatever reason. This notFulfilledState shall be supported by a reason such as the event(s) that were observed when fulfilment was attempted.
- TERMINATED: This state is registered if the respective aspect of the intent (i.e. either an expectation, a target or the whole intent) shall not be considered for fulfilment e.g. when an authorized MnS consumer sends an indication terminating the specific aspect of the intent. For instance, if the MnS consumer sends an update of the intent in which a particular target is eliminated, then that target shall be marked as "TERMINATED".
- FULFILMENTFAILED: This is the state when the MnS producer decides that the intent, expectation or target cannot be fulfilled. This state shall be supported by a reason such as the event(s) that were observed when fulfilment was attempted.

For some scenarios (in particular for the notFulfilledState with value "DEGRADED", "TERMINATED", "SUSPENDED" and "FULFILMENTFAILED"), the notFulfilledState should be supported by extra information describing or related to the state. This extra information is recorded into the notFulfilledReasons field.

6.2.1.3.5.2 Attributes

The FulfilmentInfo includes the following attributes.

Table 6.2.1.3.5.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
fulfilmentStatus	M	T	F	F	T
notFulfilledState	CM	T	F	F	T
notFulfilledReasons	CO	T	F	F	T

6.2.1.3.5.3 Attribute constraints

Table 6.2.1.3.5.3-1

Name	Definition
notFulfilledState	Condition: when FulfilmentInfo is implemented for IntentFulfilmentInfo
notFulfilledReasons	Condition: when FulfilmentInfo is implemented for IntentFulfilmentInfo

6.2.1.3.5.4 Notifications

The notifications specified for the IOC using this <<dataType>> for its attribute(s), shall be applicable.

6.2.1.3.6 IntentFulfilmentReport <<dataType>>

6.2.1.3.6.1 Definition

This <<dataType>> includes the `intentFulfilmentInfo` and `expectationFulfilmentResults`. The `intentFulfilmentInfo` describes status of fulfilment of an intent and the related reasons for the infeasible status.

IntntFulfilmentReport also optionally includes additionalFulfilmentInfo, which can be used to represent the provide additional information. Examples of additional fulfilment information could be:

- Types for the list of managed objects (e.g. NE, NF, Cell) which are updated during intent fulfilment, or

- Instance information for the list of managed objects (e.g. NE, NF, Cell) and corresponding attributes which are updated during intent fulfilment.

6.2.1.3.6.2 Attributes

The `IntentFulfilmentReport` includes the following attributes.

Table 6.2.1.3.6.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`intentFulfilmentInfo`	M	T	F	F	T
`expectationFulfilmentResults`	O	T	F	F	T
`additionalFulfilmentInfo`	O	T	F	F	T

6.2.1.3.6.3 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.

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

6.2.1.3.7.3 Attribute constraints

None.

6.2.1.3.8 TargetFulfilmentResult<<dataType>>

6.2.1.3.8.1 Definition

`TargetFulfilmentResult` <<dataType>> includes `targetFulfilmentInfo` and `targetAchievedValue` for each ExpectationTarget. The `targetFulfilmentInfo` describes status of fulfilment of an expectationTarget and the related reasons for the infeasible status. The `targetAchievedValue` describes current performance value for the

ExpectationTarget. Different instances of TargetFulfilmentResult can be instantiated for the same targetName but with different targetContexts. For examples, different TargetFulfilmentResult instance for aveDLRANUEThptTarget with different dlFrequencyContexts.

6.2.1.3.8.2 Attributes

The `TargetFulfilmentResult` includes the following attributes.

Table 6.2.1.3.8.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
targetName	M	T	F	F	T
targetFulfilmentInfo	M	T	F	F	T
targetAchievedValue	O	T	F	F	T
targetContexts	O	T	F	F	T

6.2.1.3.8.3 Attribute constraints

None.

6.2.1.3.9 IntentConflictReport << dataType >>

6.2.1.3.9.1 Definition

`IntentConflictReport` <<dataType>> represents the conflict information for the detected conflict.

When a conflict is detected, the MnS producer will configure the value of attributes of IntentConflictReport and notify the MnS consumer about the conflict, indicating the intent, intent expectation or expectation target which give rise to the conflict. The value of recommendedSolutions may be configured by MnS producer and notified to MnS consumer.

The `IntentConflictReport` includes the `conflictingIntent` for any any type of conflict that is observed. The `IntentConflictReport` includes the `conflictingExpectation` if the conflict is of type EXPECTATION_CONFLICT. The `IntentConflictReport` includes the `conflictingExpectation` and `conflictingTarget` if the conflict is of type TARGET_CONFLICT.

6.2.1.3.9.2 Attributes

The `IntentConflictReport` includes the following attributes.

Table 6.2.1.3.9.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
conflictId	**M**	**T**	**F**	**T**	**T**
conflictType	M	T	F	F	T
conflictingIntent	CM	T	F	F	T
conflictingExpectation	CM	T	F	F	T
conflictingTarget	CM	T	F	F	T
recommendedSolutions	O	T	F	F	T

6.2.1.3.9.3 Attribute constraints

Table 6.2.1.3.9.3-1

Name	Definition
conflictingIntent	The MnS producer supports reporting on INTENT_CONFLICT or EXPECTATION_CONFLICT or TARGET_CONFLICT

conflictingExpectation	The MnS producer supports reporting on conflicts of type EXPECTATION_CONFLICT or TARGET_CONFLICT
conflictingTarget	The MnS producer supports reporting on conflicts of type TARGET_CONFLICT

6.2.1.3.10 IntentFeasibilityCheckReport <<dataType>>

6.2.1.3.10.1 Definition

The IntentFeasibilityCheckReport <<dataType>> represents the intent feasibility check information. Intent feasibility check information is provided after MnS producer automatically performs feasibility check when the MnS producer received the intent creation or modification request from the MnS consumer. In case the feasibility check result is 'INFEASIBLE' the MnS producer will notify the MnS consumer.

6.2.1.3.10.2 Attributes

The IntentFeasibilityCheckReport includes the following attributes.

Table 6.2.1.3.10.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
feasibilityCheckResult	M	T	F	F	T
inFeasibleExpectationInfos	O	T	F	F	T
infeasibilityReasons	M	T	F	F	T

6.2.1.3.10.3 Attribute constraints

Void.

6.2.1.3.11 IntentHandlingCapability <<dataType>>

6.2.1.3.11.1 Definition

The IntentHandlingCapability <<dataType>> represents expectation object information and expectation target information which can be supported by a specific intent handling function of MnS producer.

The IntentHandlingCapability <<dataType>> includes a supportedExpectationObjectType and corresponding supportedExpectationTargetInfoList.

6.2.1.3.11.2 Attributes

The IntentHandlingCapability includes the following attributes.

Table 6.2.1.3.11.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
intentHandlingCapabilityId	M	T	F	F	T
supportedExpectationObjectType	M	T	F	F	T
supportedExpectationTargetInfoList	M	T	F	F	T

6.2.1.3.11.3 Attribute constraints

None.

6.2.1.3.12 ValueRangeType<<choice>>

6.2.1.3.12.1 Definition

This <<choice>> defines the data type for value of the "`targetValueRange`" and "`contextValueRange`".

6.2.1.3.12.2 Attributes

Table 6.2.1.3.12.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`CHOICE_1.1 real`	CM	T	T	F	F
`CHOICE_2.1 enum`	CM	T	T	F	F
`CHOICE_3.1 string`	CM	T	T	F	F
`CHOICE_4.1 boolean`	CM	T	T	F	F
`CHOICE_5.1 integer`	CM	T	T	F	F
`CHOICE_6.1 timeWindow`	CM	T	T	F	F
`CHOICE_7.1 dateTime`	CM	T	T	F	F
`CHOICE_8.1 geoArea`	CM	T	T	F	F
`CHOICE_9.1 pLMNId`	CM	T	T	F	F
`CHOICE_10.1 geoCoordinate`	CM	T	T	F	F
`CHOICE_11.1 uEGroup`	CM	T	T	F	F
`CHOICE_12.1 frequency`	CM	T	T	F	F
`CHOICE_13.1 schedulingTime`	CM	T	T	F	F

6.2.1.3.12.3 Attribute constrains

Table 6.2.1.3.12.3-1

Name	Definition
`CHOICE_1.1 real`	Condition: This attribute shall be supported, when the type is Real.
`CHOICE_2.1 enum`	Condition: This attribute shall be supported, when the type is Enum.
`CHOICE_3.1 string`	Condition: This attribute shall be supported, when the type is String.
`CHOICE_4.1 boolean`	Condition: This attribute shall be supported, when the type is Boolean.
`CHOICE_5.1 integer`	Condition: This attribute shall be supported, when the type is Integer.
`CHOICE_6.1 timeWindow`	Condition: This attribute shall be supported, when the type is TimeWindow.
`CHOICE_7.1 dateTime`	Condition: This attribute shall be supported, when the type is DateTime.
`CHOICE_8.1 geoArea`	Condition: This attribute shall be supported, when the type is GeoArea.
`CHOICE_9.1 pLMNId`	Condition: This attribute shall be supported, when the type is PLMNId.
`CHOICE_10.1 geoCoordinate`	Condition: This attribute shall be supported, when the type is `GeoCoordinate`.
`CHOICE_11.1 uEGroup`	Condition: This attribute shall be supported, when the type is `UEGroup`.
`CHOICE_12.1 frequency`	Condition: This attribute shall be supported, when the type is `frequency`.
`CHOICE_13.1 schedulingTime`	Condition: This attribute shall be supported, when the type is SchedulingTime.

6.2.1.3.13 Frequency<<dataType>>

6.2.1.3.13.1 Definition

It describes the RF reference frequency (i.e. Absolute Radio Frequency Channel Number) and/or the frequency operating band used for a given direction (UL or DL) in FDD or for both UL and DL directions in TDD.

6.2.1.3.13.2 Attributes

Table 6.2.1.3.13.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
arfcn	CM	T	T	F	F
freqband	CM	T	T	F	F

6.2.1.3.13.3 Attribute constrains

Table 6.2.1.3.13.3-1

Name	Definition
arfcn	Condition: This attribute shall be supported, when the frequency information represent RF reference frequency.
freqband	Condition: This attribute shall be supported, when the frequency information represent frequency operating band.

6.2.1.3.14 UEGroup <<dataType>>

6.2.1.3.14.1 Definition

This <<dataType>> describes the UE Group, which is represented by specific 5QI, specific S-NSSAI, specific PLMNId or a specific combination of S-NSSAI, 5QI, PLMNId.

6.2.1.3.14.2 Attributes

Table 6.2.1.3.14.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
pLMNId	CM	T	T	F	F
fiveQI	CM	T	T	F	F
sNSSAI	CM	T	T	F	F

6.2.1.3.14.3 Attribute constrains

Table 6.2.1.3.14.3-1

Name	Definition
plMNId	Condition: This attribute shall be supported, when UE group is represented by PLMNId.
fiveQI	Condition: This attribute shall be supported, when UE group is represented by 5QI.
sNSSAI	Condition: This attribute shall be supported, when UE group is represented by S-NSSAI.

6.2.1.3.15 InFeasibleExpectationInfo <<dataType>>

6.2.1.3.15.1 Definition

This <<dataType>> describe a list of InFeasibleExpectationInfo for all infeasible IntentExpectations in the intent. Each InFeasibleExpectationInfo includes the list of TargetNames for the InFeasibleTargets.

6.2.1.3.15.2 Attributes

Table 6.2.1.3.15.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
expectationId	M	T	F	F	F
inFeasibleTargets	M	T	F	F	F

6.2.1.3.15.3 Attribute constrains

None

6.2.1.3.16 IntentReportControl <<dataType>>

6.2.1.3.16.1 Definition

This <<dataType>> describes intent report subscription information, including customized requirements on intent report. Each IntentReportControl is associated with a distinct IntentReport MOI.

The attribute "reportRecipientAddress" indicates the address of report recipient for MnS consumer.

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 attribute "expectedReportTypes" indicates the types of IntentReports, which can be one/any/all of "IntentFulfilmentReport", "IntentConflictReport", "IntentFeasibilityCheckReport", "IntentExplorationReport", "IntentFulfilmentNegotiationReport" and "IntentUtilityReport".

The attribute "reportingConditions" indicates the specified conditions for intent reporting. TimeCondition is one choice for reportingCondition. For example, TimeCondition can be an interval, a specific time, or a time window. TagrtFulfilmentCondition in another choice for reportCondition. For example, the intent report needs to be sent by MnS producer when the achievedValue for specific target value crosses the pre-defined threshold.

The intent report will be sent to the MnS Consumer when the specified reportingConditions are satisfied.

The attribute "reportingTargets" indicates the specified targets needed to be reported. All the targets described in the corresponding Intent instance need to be reported if the MnS consumer does not provide values for the attribute "reportingTargets".

6.2.1.3.16.2 Attributes

Table 6.2.1.3.16.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
reportRecipientAddress	CM	T	T	F	F
observationPeriod	M	T	T	F	F
expectedReportTypes	O	T	T	F	F
reportingConditions	O	T	T	F	F
reportingTargets	O	T	T	F	F

6.2.1.3.16.3 Attribute constrains

Name	Definition
reportRecipientAddress	Condition: The implicit intent report subscription mechanism is supported.

6.2.1.3.17 ReportingConditions <<Choice>>

6.2.1.3.17.1 Definition

This <<dataType>> describes the specified conditions for intent reporting. TimeCondition and targetFulfilmentConfition are choice for reportingConditions.

6.2.1.3.17.2 Attributes

Table 6.2.1.3.17.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
CHOICE_1.1 timeCondition	CM	T	T	F	F
CHOICE_1.2 targetFulfilmentCondition	CM	T	T	F	F

6.2.1.3.17.3 Attribute constrains

Name	Definition
CHOICE_1.1 timeCondition	This attribute shall be supported, when MnS producer support the capability to allow a MnS consumer to specify time condition for intent reporting
CHOICE_1.2 targetFulfilmentCondition	This attribute shall be supported, when MnS producer support the capability to allow a MnS consumer to specify condition of targets for intent reporting

6.2.1.3.18 TargetFulfilmentCondition <<dataType>>

6.2.1.3.18.1 Definition

`TargetFulfilmentCondition` <<dataType>> indicates the specified conditions of target fulfilment for intent reporting. The `TargetFulfilmentCondition` <<dataType>> includes a `targetName`, `targetCondition`, and `targetValueRange`.

6.2.1.3.18.2 Attributes

Table 6.2.1.3.18.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
targetName	M	T	T	F	T
targetCondition	M	T	T	F	T
targetValueRange	M	T	T	F	T

6.2.1.3.18.3 Attribute constraints

None.

6.2.1.3.19 IntentExplorationReport <<dataType>>

6.2.1.3.19.1 Definition

The `IntentExplorationReport` <<dataType>> represents the intent exploration result (including outcomes) for intent negotiation, which includes the list of expectation exploration results (including outcomes. Intent exploration result is provided after MnS producer performs intent exploration pre-evaluation process as requested by MnS consumer.

6.2.1.3.19.2 Attributes

The `IntentExplorationReport` includes the following attributes.

Table 6.2.1.3.19.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`expectationExplorationResults`	M	T	F	F	T

6.2.1.3.19.3 Attribute constraints

Void.

6.2.1.3.20 ExpectationExplorationResult <<dataType>>

6.2.1.3.20.1 Definition

The `ExpectationExplorationResult` <<dataType>> represents the expectation exploration result for a specific intent expectation, which includes the list of exploration results (i.e. recommended best values) for the expectation targets and/or expectation contexts. There maybe multiple ExpectationExplorationResults for a specified intent expectations.

6.2.1.3.20.2 Attributes

The `IntentExplorationResult` includes the following attributes.

Table 6.2.1.3.20.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`expectationId`	M	T	F	F	T
`targetExplorationResults`	M	T	F	F	T
`contextExplorationResults`	O	T	F	F	T

6.2.1.3.20.3 Attribute constraints

Void.

6.2.1.3.21 ImplicitIntent <<dataType>>

6.2.1.3.21.1 Definition

`ImplicitIntent` <<dataType>> refers to implicit intent which includes the implict information that MnS consumers have not explicitly pointed out in its intent.

It means implicit intent contains the additional intent expectation target and contexts which represents the constraints and conditions to apply for the entire implicit intent.

The MnS producer receives the intent sent by the MnS consumer. If the intent is feasible, the MnS producers will figure out the implicit information associated with the intent according to the intent related information, e.g. historical intent data and resource status etc. The MnS producer sends an intent report with implicit information to MnS consumer, so that the MnS consumer can determine corresponding intent expectations according to the implict intent and may modify the intent. The MnS producer receives the modified intent sent by the MnS consumer.

For example, an MnS consumer expresses its intent including an expectation for reducing energy consumption at base stations, but this intent does not explicitly specify the associated expectations such as cell coverage area or the number

of accessing users. However, the MnS consumer might be concerned about these expectations. In such cases, the MnS producer needs to provide feedback to the MnS consumer regarding this implicit information, i.e., the implicit intent.

6.2.1.3.21.2 Attributes

The `implicitIntent` includes the following attributes.

Table 6.2.1.3.21.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`implicitIntentExpectations`	M	T	F	F	T
`implicitIntentContexts`	O	T	F	F	T

6.2.1.3.21.3 Attribute constraints

None.

6.2.1.3.22 IntentFulfilmentNegotiationReport <<dataType>>

6.2.1.3.22.1 Definition

This <<dataType>> represents the MnS producer's information to the MnS consumer regarding intent negotiations during fulfilment phase. It may contain information on:

- `possibleIntentOutcomeList`, which is the list of possible intent outcomes available at the MnS producer. It is an ordered list where the top entry indicates the best outcomes according to information available at the MnS producer. If used to indicate the best possible outcome, it will contain only 1 entry. Each `possibleIntentOutcome` may include information on the impact on the related ExpectationObjects. Providing a list possible intent outcomes inherently asks the MnS consumer to provide information on the MnS consumer's preference among the possible outcomes.
- intentNegotiationConsumerFeedback, which is written by MnS consumer when providing feedback on a specific intent negotiation report.
- `implicitIntent`, which includes at least one implicit intent expectation that MnS consumer has not explicitly pointed out. In case the `implicitIntentIndex` is `"True"`, the MnS producer discovers the implicit information like additional intent expectation which is not explicitly pointed out.

6.2.1.3.22.2 Attributes

Table 6.2.1.3.22.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`possibleIntentOutcomeList`	O	T	F	F	T
`intentFulfilmentNegotiationConsumerFeedback`	O	T	T	T	T
`implicitIntent`	CM	T	F	T	T

6.2.1.3.22.3 Attribute constrains

Name	Definition
`implicitIntent` Support Qualifier	Condition: `ImplicitIntentIndex` is enabled.

6.2.1.3.23 PossibleIntentOutcome <<dataType>>

6.2.1.3.23.1 Definition

This <<dataType>> indicates a single outcome evaluated by the MnS producer. It may contain information on the impact on the related ExpectationObjects..

6.2.1.3.23.2 Attributes

The PossibleIntentOutcome includes the following attributes.

Table 6.2.1.3.23.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
possibleIntentOutcomeId	M	T	F	F	T
intentFulfilmentInfo	M	T	F	F	T
expectationFulfilmentResults	O	T	F	F	T
possibleImpact	O	T	F	T	T

6.2.1.3.23.3 Attribute constraints

None.

6.2.1.3.24 IntentFulfilmentNegotiationFeedback <<dataType>>

6.2.1.3.24.1 Definition

This <<dataType>> contains the feedback information that the MnS consumer's provides to the MnS producer as response during intent fulfilment Negotiation. The data is written into the intent for which the MnS consumer provides feedback. It contains:

- `preferedIntentOutcomeId`, which indicates for a specific possible intent outcome that the MnS consumer is confirmed (e.g. operator). MnS consumer will review possible outcomes (including possible possibleImpact for each outcome) and decide a final outcome for MnS producer to fulfil the intent.
- `consumerSatisfactionIndex`, which indicates the MnS consumer's satisfaction with one or more of the MnS producer's possible outcome. It may contain a list indicating the MnS consumer's expected satisfaction from the different possible outcome e.g. as evaluated form the MnS consumer's utility function. It may also indicate a single value for a single solution that has been deployed by the MnS producer.

6.2.1.3.24.2 Attributes

Table 6.2.1.3.24.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
preferredIntentOutcomeId	O	T	T	T	F
consumerSatisfactionIndex	O	T	T	T	F

6.2.1.3.24.3 Attribute constrains

None.

6.2.1.3.25 PossibleImpact <<dataType>>

6.2.1.3.25.1 Definition

`PossibleImpact` <<dataType>> indicates the possible impact of the possible outcome.

`ImpactedObject` refers to managed objects that may be impacted by the recommended candidate alternatives. The impactedAttributes defines the name-value pair, where the name indicates the name of the attribute that is impacted and the value indicates the updated value.

EXAMPLE: For an expectation target on energy consumption, the `impactedObject` may include impacted cells could be deactivated to make the intent fulfillable, name is energySavingState, and the value is "IS_ENERGYSAVING".

6.2.1.3.25.2 Attributes

The `PossibleImpact` includes the following attributes.

Table 6.2.1.3.25.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`impactedObjects`	O	T	F	F	T
`impactedAttributes`	O	T	F	F	T

6.2.1.3.25.3 Attribute constraints

None.

6.2.1.3.26 SupportedExpectationTargetInfo <<dataType>>

6.2.1.3.26.1 Definition

The `SupportedExpectationTargetInfo` indicates the detailed information about what the intent driven MnS producer supports for a given `supportedExpectationObjectType`. It allows the intent driven MnS producer to indicate the support in any one of the three ways below as illustrated by Figure 6.2.1.3.26.1-1:

1) As a list of names of supported expectation targets.

2) As a list of names of supported expectation targets and the value ranges within which they are supported.

3) As a set expressing the combination of expectation targets and value ranges that can be jointly supported. The valuerange indicates the general limits supported by the intent handler. A feasibility check may still be needed to confirm feasibility of specific values.

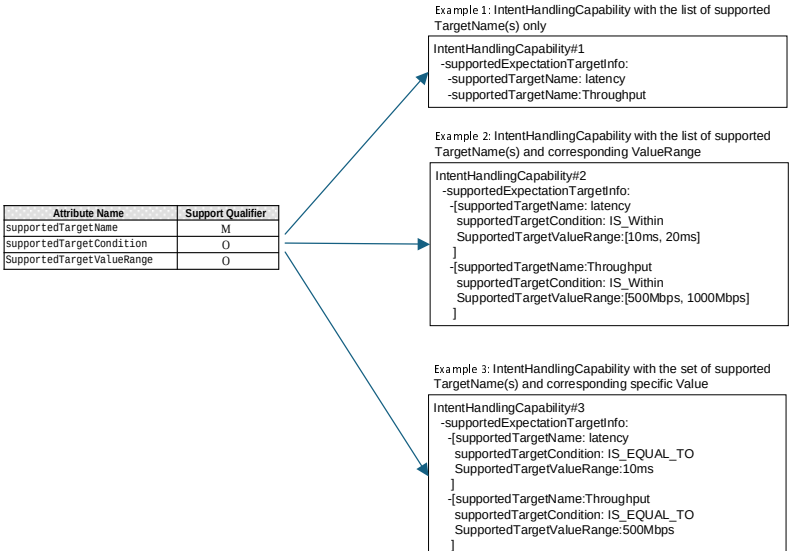


Figure 6.2.1.3.26.1-1: MnS producer can express supported capabilities in any of the 3 possible ways.

6.2.1.3.26.2 Attributes

The `supportedExpectationTargetInfo` includes the following attributes.

Table 6.2.1.3.26.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`supportedTargetName`	M	T	F	F	T
`supportedTargetCondition`	O	T	F	F	T
`supportedTargetValueRange`	O	T	F	F	T

6.2.1.3.26.3 Attribute constraints

None.

6.2.1.3.27 UtilityParameter <<dataType>>

6.2.1.3.27.1 Definition

This <<dataType>> provides the inputs for the specified Intent Utility Function.

6.2.1.3.27.2 Attributes

Table 6.2.1.3.27.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`parameterName`	M	T	T/F (NOTE)	F	F
`parameterWeight`	M	T	T/F (NOTE)	F	F
NOTE: The isWritable qualifier is "F" if the attribute is used as output (e.g. `utilityParameterList` in `UtilityDefinition`). The isWritable qualifier is "T" if the attribute is used as input (e.g. `utilityParameterList` in `IntentUtilityFormula`).					

NOTE: Void.

6.2.1.3.27.3 Attribute constraints

None.

6.2.1.3.28 UtilityResult <<dataType>>

6.2.1.3.28.1 Definition

This <<dataType>> provides the result for the specified Intent Utility Function.

6.2.1.3.28.2 Attributes

Table 6.2.1.3.28.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
utilityFunctionId	M	T	F	F	F
utilityResult	M	T	F	F	F

6.2.1.3.28.3 Attribute constraints

None.

6.2.1.3.29 UtilityDefinition <<dataType>>

6.2.1.3.29.1 Definition

This <<dataType>> is used to describe each supported Intent Utility Function.

The attribute `utilityDefinitionId` provides a unique identifier for each utility function definition.

The attribute `utilityDescription` provides a description of the utility function.

The attribute `utilityParameterList` specifies the input parameters required for the utility function.

6.2.1.3.29.2 Attributes

Table 6.2.1.3.29.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
utilityDefinitionId	M	T	F	F	F
utilityDescription	M	T	F	F	F
utilityParameterList	M	T	F	F	F

6.2.1.3.29.3 Attribute constraints

None.

6.2.1.3.30 IntentUtilityReport <<dataType>>

6.2.1.3.30.1 Definition

This <<dataType>> provides the report of the intent utility functions.

6.2.1.3.30.2 Attributes

Table 6.2.1.3.30.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
utilityResultList	M	T	F	F	F

6.2.1.3.30.3 Attribute constraints

Table 6.2.1.3.30.3-1

Name	Definition

6.2.1.4 Attribute definition

Table 6.2.1.4-1

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: N/A isUnique: N/A defaultValue: None isNullable: False
intentExpectations	It describes the expectations including requirements, goals and contexts (including constraints and filter information) given to a 3GPP system. It states the list of specific outcomes desired to be realized for expectation object(s). The intentExpectations are arranged in an ordered list such that the most important intentExpectations are on the top of the list. allowedValues: Not Applicable	type: IntentExpectation multiplicity: 1..* isOrdered: True isUnique: True defaultValue: None isNullable: False
intentFulfilmentInfo	It describes status of fulfilment of an intent and the related reasons for that status. allowedValues: Not Applicable	type: FulfilmentInfo multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
additionalFulfilmentInfo	It describes the additional information of intent intent fulfilment. The content and format is vendor specific. allowedValues: Not Applicable	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
expectationFulfilmentInfo	It describes status of fulfilment of an intentExpectation and the related reasons for that status. allowedValues: Not Applicable	type: FulfilmentInfo multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
targetFulfilmentInfo	It describes status of fulfilment of an expectationTarget and the related reasons for that status. allowedValues: Not Applicable	type: FulfilmentInfo multiplicity: 1 isOrdered: N/A isUnique: N/A

Attribute Name	Documentation and Allowed Values	Properties
		defaultValue: None isNullable: False
fulfilmentStatus	It describes the current status of the fulfilment result for intent, intentExpectation or expectationTarget, which is configured by MnS producer and can be read by MnS consumer. allowedValues: "FULFILLED", "NOT_FULFILLED"	type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "NOT_FULFILLED " isNullable: False
notFulfilledState	It describes the current state for not achieving fulfilment for the intent, intentExpectation or expectationTarget. It is configured/written by MnS producer and can be read by MnS consumer. allowedValues: "ACKNOWLEDGED", "COMPLIANT", "DEGRADED", "SUSPENDED", "TERMINATED" "FULFILMENTFAILED"	type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "ACKNOWLEDGE D" isNullable: False
notFulfilledReasons	It describes the reasons/observations related to the specific notFulfilledState allowedValues: Not Applicable	type: String multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
intentContexts	It describes the list of IntentContext(s) which represents the constraints and conditions that should apply for the entire intent even if there may be specific contexts defined for specific parts of the intent. allowedValues: triple of (attribute, condition, value range)	type: Context multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
expectationId	A unique identifier of the intentExpectation within the intent. allowedValues: Not Applicable	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
expectationVerb	It describes the characteristic of the intentExpectation and is the property that describes the types of intentExpectations. Examples of verbs and their related types of expectation are Deliver: DeliveryIntentExpectation, e.g. Deliver a RAN network, Service, Slice, function Ensure: AssuranceintentExpectation, e.g. Ensure the target performance value. Maintain: MaintenanceIntentExpectation, e.g. Maintain the network element according to a target version. allowedValues: DELIVER, ENSURE, MAINTAIN Vendor extensions are allowed	type: String multiplicity: 1 isOrdered:N/A isUnique: N/A defaultValue: None isNullable: False
expectationObject	It describes the expectation objects to which the IntentExpectation should apply. allowedValues: Not Applicable	type: ExpectationObject multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
objectType	It describes the type of expectation object of the IntentExpectation that is required to be applied to. It can be class name of the managed object. allowedValues: see scenario specific IntentExpectation	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None

Attribute Name	Documentation and Allowed Values	Properties
		isNullable: False
objectInstance	It describes a specific object instance (e.g. instance of managed object) to which the intentExpectation should apply. allowedValues: Not Applicable	type: DN multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
objectContexts	It describes the list of ObjectContext(s) which represents the constraints and conditions to be used as filter information to identify the object(s) 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. The concrete ObjectContext depends on the ExpectationObject, which is defined in clause 6.2.2. All the concrete ObjectContexts follow the common structure of ObjectContext.	type: Context multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
expectationTargets	It describes the list of ExpectationTarget(s) which represent specific outcomes on 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. The concrete ExpectationTarget depends on the ExpectationObject, which is defined in clause 6.2.2. All the concrete ExpectationTargets follow the common structure of ExpectationTarget. The expectionTargets are arranged in an ordered list such that the most important expectionTargets are on the top of the list.	type: ExpectationTarget multiplicity: 1..* isOrdered: True isUnique: True defaultValue: None isNullable: False
expectationContexts	It describes the list of context(s) which represents the 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 Expectation Object in the IntentExpectation	type: Context multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
targetName	It describes the name of the expectation target which represents specific outcomes on 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 ExpectationObject in the IntentExpectation	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None 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", "IS_WITHIN_RANGE", "IS_OUTSIDE_RANGE", "IS_ONE_OF", " IS_EQUAL_TO_OR_LESS_THAN", "IS_EQUAL_TO_OR_GREATER_THAN", "IS_NOT_ONE_OF", "IS_ALL_OF"	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "IS_EQUAL_TO" isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
targetValueRange	It describes the range of values that applicable to the targetName and the targetCondition. allowedValues: depends on the targetCondition. The value will be a single value when the targetCondition is either "IS_EQUAL_TO", "IS_LESS_THAN", "IS_GREATER_THAN", "IS EQUAL TO OR LESS THAN", "IS EQUAL TO OR GREATER THAN" The value will be a pair of values when the targetCondition is either "IS_WITHIN_RANGE", "IS_OUTSIDE_RANGE" The value will be a list when the targetCondition is "IS_ONE_OF", "IS_NOT_ONE_OF","IS_ALL_OF". See NOTE.	type: ValueRangeType multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: True
targetContexts	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: Not Applicable	type: Context multiplicity: * isOrdered: False isUnique: True 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: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None 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", "IS_WITHIN_RANGE", "IS_OUTSIDE_RANGE, "IS_ONE_OF", "IS_EQUAL_TO_OR LESS_THAN", "IS_EQUAL_TO_OR_GREATER_THAN", "IS_NOT_ONE_OF", "IS_ALL_OF"	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "IS_EQUAL_TO" isNullable: False
contextValueRange	It describes the range of values that applicable to the ContextAttribute and the ContextCondition. AllowedValue: depends on the contextCondition The value will be a single value when the contextCondition is either "IS_EQUAL_TO", "IS_LESS_THAN", "IS_GREATER_THAN", "IS_EQUAL_TO_OR_LESS_THAN", "IS_EQUAL_TO_OR_GREATER_THAN". The value will be a pair of values when the contextCondition is either "IS_WITHIN_RANGE", "IS_OUTSIDE_RANGE" The value will be a list when the contextCondition is "IS_ONE_OF", "IS_NOT_ONE_OF","IS_ALL_OF". See NOTE.	type: ValueRangeType multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: True
intentPriority	It expresses the priority of the stated intent within an MnS consumer. AllowedValue: values in the range [1-100] where 1 indicates the highest priority and 100 indicates the lowest priority. NOTE: The handing of the priorities across MnS consumers is left to implementation	type: integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: 1 isNullable: False
geoArea	It describes a geographical area defined in 3GPP TS 28.622[6]. AllowedValue: As defined by the data type	type: GeoArea multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
pLMNId	It describes the information of a PLMN identification defined in 3GPP 28.658[10]	type: PLMNId multiplicity: 1 isOrdered: N/A

Attribute Name	Documentation and Allowed Values	Properties
	AllowedValue: As defined by the data type	isUnique: N/A defaultValue: None isNullable: True
dateTime	It describes the information of a date time defined in 3GPP TS 28.622[6]. AllowedValue: As defined by the data type	type: DateTime multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
timeWindow	It describes the information of a time window (including startTime, endTime) defined in 3GPP TS 28.622[6]. AllowedValue: As defined by the data type	type: TimeWindow multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
geoCoordinate	It describes the information of a geoCoordinate defined in 3GPP TS 28.622[6]. AllowedValue: As defined by the data type	type: GeoCoordinate multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
pLMNInfo	It describes the information of PLMNInfo (including PLMN and S-NSSAI in the PLMN in case of network slicing feature) defined in 3GPP TS 28.541[5]. AllowedValue: As defined by the data type	type: PLMNInfo multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
schedulingTime	It describes the information of SchedulingTime (including one-time interval, daily periodicity, weekly periodicity or monthly periodicity) defined in 3GPP TS 28.622 [6].	type: SchedulingTime multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
frequency	It desribes the RF reference frequency (i.e. Absolute Radio Frequency Channel Number) and/or the frequency operating band used for a given direction (UL or DL) in FDD or for both UL and DL directions in TDD. AllowedValue: As defined by the data type	type: Frequency multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
arfcn	It desribes the RF reference frequency (i.e. Absolute Radio Frequency Channel Number). Allowed Value: For NR, see TS 38.104 [8] clause 5.4.2.1. For EUTRAN, see TS 36.104 [14] clause 5.7.3.	type: Integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
freqband	It desribes the the frequency operating band. Allowed Value: For NR, see TS 38.104 [8] clause 5.4.2.3. For EUTRAN, see TS 36.104 [14] clause 5.7.3.	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
uEGroup	It describes the information of a UE Group (represented by specific 5QI, specific S-NSSAI, specific PLMNId or a specific combination of S-NSSAI, 5QI, PLMNId).	type: UEGroup multiplicity: 1 isOrdered: N/A

Attribute Name	Documentation and Allowed Values	Properties
	AllowedValue: As defined by the data type	isUnique: N/A defaultValue: None isNullable: True
fiveQI	It describes the information of a 5QI defined in 3GPP TS 28.541[5]. AllowedValue: 0 - 255	type: integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
sNSSAI	It describes the information of a S-NSSAI defined in 3GPP TS 28.541[5]. AllowedValue: As defined by the data type	type: S-NSSAI multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
intentAdminState	It describes the intent administrative state, which enables the MnS consumer to suspend an intent or cancel the suspension for a suspended intent. A suspended intent means this intent is not considered for fulfilment allowedValues: "ACTIVATED", "DEACTIVATED"	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "ACTIVATED" isNullable: False
intentReference	It indicates the associated intent instance allowedValues: Not Applicable	type: DN multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
intentReportReference	It indicates the associated intent report instance(s) allowedValues: Not Applicable	type: DN multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
intentReportControl	It indicates the intent report control and subscription information. allowedValues: Not Applicable	type: IntentReportControl multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
reportRecipientAddress	It indicates the address of report recipient for MnS consumer. allowedValues: Not Applicable	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
observationPeriod	It represents the observation period of the fulfilmentInfo for corresponding ExpectationTargets, IntentExpectations and Intent. At the end of the observation period, the corresponding fulfilment info is updated in the intent report. The observation period can be assigned by MnS consumer through requesting the MnS producer to set attribute "observationPeriod". MnS producer also can assign the observation period if MnS consumer didn't assign it. The observation time is expressed in seconds. allowedValues: Not Applicable	type: Integer multiplicity: 0..1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
expectedReportTypes	It indicates the type of IntentReports, which can be one/any/all of "IntentFulfilmentReport", "IntentConflictReport" "IntentFeasibilityCheckReport", "intentExplorationReport", "IntentFulfilmentNegotiationReport" and " intentUtilityReport ". allowedValues: INTENT_FULFILMENT_REPORT, INTENT_CONFLICT_REPORT, INTENT_FEASIBILITY_CHECK_REPORT, INTENT_EXPLORATION_REPORT, INTENT_FULFILMENT_NEGOTIATION_REPORT, INTENT_UTILITY_REPORT	type: ENUM multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
reportingConditions	It indicates the specified conditions for intent reporting. The intent report will be sent when the specified reportingConditions is satisfied. allowedValues: Not Applicable	type: ReportingCondition multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
reportingTargets	It indicates the specified targets needed to be reported. allowedValues: the scenario specific targetName defined in clause 6.2.2 Scenario specific IntentExpectation definition	type: String multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
timeCondition	It indicates the specified times for intent reporting. The specified times can be one-time interval, daily periodicity, weekly periodicity or monthly periodicity allowedValues: Not Applicable	type: SchedulingTime multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
targetFulfilmentCondition	It indicates the specified conditions of target Fulfilment for intent reporting. The targetFulfimentConfition can be described based on the achieved value for a specific targetName. allowedValues: Not Applicable	type: TargetFulfilmentCondition multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
intentFulfilmentReport	It describes the fulfillment information which is reported for the associated intent instance. allowedValues: Not Applicable	type: IntentFulfilmentReport multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
intentConflictReports	It describes the conflict information which is reported for associated intent instance if needed. allowedValues: Not Applicable	type: IntentConflictReport multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
conflictId	It is used to identify the detected conflict within an IntentReport instance. allowedValues: Not Applicable	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
conflictType	It describes the type of intent conflict. allowedValues: INTENT_CONFLICT, EXPECTATION_CONFLICT, TARGET_CONFLICT	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
conflictingIntent	It describes the DN of the conflicting intent allowedValues: Not Applicable	type: DN multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
conflictingExpectation	It describes the expectationId of the conflicting IntentExpectation within an Intent. allowedValues: Not Applicable	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
conflictingTarget	It describes the targetName of the conflicting ExpectationTarget within an IntentExpectation. allowedValues: Not Applicable	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
recommendedSolutions	It describes the action recommended by the MnS producer to be undertaken by the MnS consumer to resolve intent conflict. The recommended solution applies only for the specific intent whose intent report contains this attribute. allowedValues: "MODIFY", "DELETE"	type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
expectationFulfilmentResults	It includes the expectationFulfilmentInfo and targetFulfilmentResults for each IntentExpectation. The expectationFulfilmentInfo describes status of fulfilment of an intentExpectation and the related reasons for infeasible status. allowedValues: Not Applicable	type: ExpectationFulfilmentResult multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
targetFulfilmentResults	It includes targetFulfilmentInfo and targetAchievedValue for each ExpectationTarget. The targetFulfilmentInfo describes status of fulfilment of an expectationTarget and the related reasons for infeasible status. The targetAchieveValue describes current performance value for the ExpectationTarget. allowedValues: Not Applicable	type: TargetFulfilmentResult multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
targetAchievedValue	It describes the value that has been achieved for the expectation target at the time at which the report is generated. allowedValues: Not Applicable	type: ValueRangeType multiplicity: 0..1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
intentFeasibilityCheckReport	It describes the intent feasibility check information which is reported if needed. allowedValues: Not Applicable	type: IntentFeasibilityCheckReport multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
feasibilityCheckResult	It describes the result of intent fulfilment feasibility check allowedValues: FEASIBLE, INFEASIBLE	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
infeasibilityReasons	It describes the reason (e.g. invalid intent expression, the intent conflict) of the result of intent fulfilment feasibility check is INFEASIBLE allowedValues: INVALID_INTENT_EXPRESSION, INTENT_CONFLICT	type: ENUM multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False
expectationExplorationResults	It describes the intent exploration result, which includes the list of expectation exploration results. Intent exploration result is provided after MnS producer performs intent exploration pre-evaluation process as requested by MnS consumer.	type: ExpectationExplorationResult multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False
targetExplorationResults	It describes the list of exploration results (i.e. recommended best values) for the expectation targets. Each TargetExplorationResult includes the TargetName, Target Condition and a recommended TargetValueRange	type: ExpectationTarget multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False
contextExplorationResults	It describes the list of exploration results (i.e. recommended best values) for the expectation context (e.g. coverageAreaPolygonContext). Each ContextExplorationResult includes the ContextAttribute, Context Condition and a recommended ContextValueRange.	type: Context multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
intentHandlingCapabilityList	It describes the list of expectation object information and expectation target information which can be supported by intent handling function. allowedValues: Not Applicable	type: IntentHandlingCapability multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False
intentHandlingCapabilityId	A unique identifier of property of intent handling capability should be supported by the intent handling function of MnS producer. allowedValues: Not Applicable	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
supportedExpectationObjectType	It describes the expectation object type which can be supported by a specific intent handling function of MnS producer. allowedValues: objectType defined in clause 6.2.1.3.2.	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
lastUpdatedTime	It describes the time for the latest update of the IntentReport Instance.	type: DateTime multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
ContextSelectivity	It expresses the way in which all or a subset of the `contexts` may be applied. The `contextSelectivity` indicates which contexts are to be applied, i.e., "ALL_OF", "ONE_OF", or "ANY_OF" the contexts. AllowedValue: "ALL_OF", "ONE_OF", "ANY_OF"	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "ALL_OF" isNullable: False
expectationSelectivity	It expresses the way in which the set of `intentExpectations` in an `intent` may be applied, e.g. for validation. The MnS consumer can provide intent expectations that describes the different alternatives candidate characteristics of the desired service from the MnS consumer's point of view that the MnS consumer wants to be validated. The `expectationSelectivity` indicates which intentExpectations are to be validated, i.e., "ALL_OF", "ONE_OF", "ANY_OF" the intentExpectations AllowedValue: "ALL_OF", "ONE_OF", "ANY_OF"	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "ALL_OF" isNullable: False
intentPreemptionCapability	It describes the pre-emption capability. The attribute is used by MnS producer to decide the target of intent deletion or intent modification allowedValue: TRUE, FALSE	type: Boolean multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "FALSE" isNullable: False
intentMgmtPurpose	It describes the MnS consumer requirements for the management purpose (required procedures) of the created or modified intent instance. allowedValue: - FEASIBILITYCHECK, represent required procedures for checking the feasibility of corresponding intent expectations during negotiation in intent pre-evaluation phase. Intent feasibility reports are produced for the intents.	type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "FULFILMENT_WITHOUT_NEGOTIATION" isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
	- EXPLORATION, represent the required procedures for exploring the best possible values that can be achieved for a specific intent during negotiation in intent pre-evaluation phase. Intent exploration reports are produced for the intents. - FULFILMENT_WITHOUT_NEGOTIATION, represent required procedures to fulfil corresponding intent expectation without negotiation. - FULFILMENT_WITH_NEGOTIATION, represent required procedures to fulfil corresponding intent expectation with negotiation in intent fulfilment phase and interaction with intentFulfilmentNegotiationReport.	
inFeasibleExpectationInfos	It describes the list of InFeasibleExpectationInfo for all infeasible IntentExpectations in the intent.	type: InFeasibleExpectationInfo multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False
inFeasibleTargets	It describes the list of TargetNames for the InFeasibleTargets	type: String multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False
IntentExpectation.preferenceWeight	It represents the preference information of the Consumer on the intent expectation, which indicates the preference weight on the preferred intentExpectations. It is an integer in the range [0,10] used to indicate the extent of preference among intent expectations: - When the allowed value is larger than 0, it means that preference information is specified by the consumer for the intentExpectation. The larger value indicates the intentExpectation is preferred by the Consumer. - When the allowed value is equal to 0, it means that no preference information is specified by the consumer for the intentExpectation, i.e., intentExpectation is considered optional. AllowedValue: values in the range [0-10]	type: Integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: 0 isNullable: False
ExpectationTarget.preferenceWeight	It represents the preference information of the Consumer on the expectation target, which indicates the preference weight on the preferred expectationTargets. It is an integer in the range [0,10] used to indicate the extent of preference among expectation targets: - When the allowed value is larger than 0, it means that preference information is specified by the consumer for the expectationTarget. The larger value indicates the expectationTarget is preferred by the Consumer. - When the allowed value is equal to 0, it means that no preference information is specified by the consumer for the expectationTarget, i.e., expectationTarget is considered optional. AllowedValue: values in the range [0-10]	type: Integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: 0 isNullable: False
implicitIntentIndex	It indicates whether the MnS consumer enables the MnS producer to perform implicit intent discovery the implicit information like additional intent expectation which is not explicitly pointed out.	type: Boolean multiplicity: 1 isOrdered: N/A isUnique: N/A

Attribute Name	Documentation and Allowed Values	Properties
	allowedValue: TRUE, FALSE	defaultValue: "FALSE" isNullable: False
implicitIntentExpectations	It describes the implicit intent expectation of the MnS consumer pointed out by the MnS producer.	type: IntentExpectation multiplicity: 1..* isOrdered: True isUnique: True defaultValue: None isNullable: False
implicitIntentContexts	It describes the list of implicit IntentContext(s) which represents the constraints and conditions that should apply for the entire implicit intent. allowedValues: triple of (attribute, condition, value range)	type: Context multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: False
intentFulfilmentNegotiationReport	It contains the information that exchanged between the MnS producer and the MnS consumer during intent fulfilment negotiations procedures.	type: IntentFulfilmentNegotiationReport multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
possibleIntentOutcomeList	it indicates the possible outcomes for the intent expectations and expectationTargets within that intent and the relative cost/impact (on the related ExpecationObjects) of achieving that outcome For each possible outcome, a PossibleIntentOutcome is provided. Impacts on ExpecationObjects that were not in the original intent may be added to the report as new intentExpectations	type: PossibleIntentOutcome multiplicity: 1 .. * isOrdered: True isUnique: True defaultValue: None isNullable: False
intentFulfilmentNegotiationConsumerFeedback	It contains the feedback information that the MnS consumer provides to the MnS producer as response during intent Negotiation.	type: IntentFulfilmentNegotiationFeedback multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
preferredIntentOutcomeId	It indicates for a specific possible intent outcome confirmed by MnS consumer among those indicated by the MnS producer. It indicates the identifier of one of the possible intent outcomes among those provided to the MnS consumer by the MnS producer.	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
consumerSatisfactionIndex	It indicates the MnS consumer's satisfaction with one or more of the MnS producer's alternatives. It represents the computed outcomes of MnS consumer's utility function 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. If it is provided in response to a report from an MnS producer indicating several candidate alternatives and their impacts, the satisfaction index is ordered according to the order of the reports. If it is provided as feedback for a single solution that was selected by the MnS producer and deployed, it indicates the MnS consumer's satisfaction with the deployed solution.	type: integer multiplicity: 1 isOrdered: True isUnique: True defaultValue: None isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
possibleImpacts	It refers to the possible impacts of the possible candidate alternative. allowedValues: Not Applicable	type: PossibleImpact multiplicityl: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False
possibleIntentOutcomeId	It identifies the number the outcome in the possible possibleOutcomeList	type: integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
impactedObjects	It indicates the DN of managed object that may be impacted by the corresponding possible intent outcomes. allowedValues: Not Applicable	type: DN multiplicity: 1..* isOrdered: True isUnique: False defaultValue: None isNullable: False
impactedAttributes	It defines the name-value pair, where the name indicates the name of the attribute that is impacted and the value indicates the updated value.	type: ttributeNameValuePairSet multiplicity: 1..* isOrdered: True isUnique: False defaultValue: None isNullable: False
supportedExpectationTargetInfoList	It describes the supported expectation targets for the supported expectation object type.	type: SupportedExpectationTargetInfo multiplicity: 1 … * isOrdered: False isUnique: True defaultValue: None isNullable: False
supportedTargetName	It indicates the name of the supported expectation targets for the supported expectation object type. allowedValues: depends on ExpectationObject in the IntentExpectation	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
supportedTargetCondition	It expresses the limits within which the supportedExpectationTargetName shall be supported. allowedValues: targetCondition defined in clause 6.2.1.3.3	type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: "IS_EQUAL_TO" isNullable: False
supportedTargetValueRange	It describes the range of values that applicable to the supportedExpectationTargetName and the supportedTargetCondition. allowedValues: targetValueRange defined in clause 6.2.1.3.3	type: ValueRangeType multiplicity: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
utilityScale	Provides the scaling factor to be applied to the utility function result. allowedValues: >=0	type: Real multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: 1 isNullable: False
utilityOffset	Provides the offset to be applied to the utility function result. allowedValues: >=0	type: Real multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: 0 isNullable: False
utilityDescription	Provides the description of the Intent Utility Function. allowedValues: vendor specified.	type: String multiplicity: 0..* isOrdered: False isUnique: True defaultValue: N/A isNullable: False
supportedUtilityList	Provides a list of utility function definitions supported by the Intent Handling Function. The definitions define the input used to provision instances of the utility function. allowedValues: Each entry is identified by a unique Id.	type: UtilityDefinition multiplicity: 0..* isOrdered: False isUnique: True defaultValue: N/A isNullable: False
utilityParameterList	An ordered list of parameters used to calculate the utility function result.	type: UtilityParameter multiplicity: 0..* isOrdered: True isUnique: True defaultValue: N/A isNullable: False
intentUtilityReports	Provides the report of intent utility results.	type: IntentUtilityReport multiplicity: 0..* isOrdered: True isUnique: N/A defaultValue: N/A isNullable: False
utilityResult	Result of the utility function evaluation. allowedValues: 0..100	type: Real multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: N/A isNullable: False
parameterName	A parameter name. allowedValues: an ‘expectationName.targetName’ valid for the scenario specific expectationtargets and expectationcontexts in clause 6.2.2, and/or vendor specified.	type: String multiplicity: 1 isOrdered: False isUnique: N/A defaultValue: None isNullable: False
parameterWeight	A parameter weight. Default value is 1. allowedValues: value between 0 and 1.	type: Real multiplicity: 1 isOrdered: False isUnique: N/A defaultValue: 1 isNullable: False
utilityFunctionId	The identifier of the function to be used to calculate the utility result. allowedValues: vendor specified.	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
intentUtilityFormulaRef	List of intent utility instances. Each object instance is identified by its DN. allowedValues: N/A	type: DN multiplicity: 0..* isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
consumerSatisfactionIndex Threshold	This defines the consumer satisfaction index threshold. If the value provided in consumerSatisfactionIndex goes below this threshold, the producer may refer to another mechanism to try to find a better outcome based on its functionality (e.g. utility function, priority).	type: Integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: N/A isNullable: False
utilityResultList	List of utility results.	type: UtilityResult multiplicity: 1..* isOrdered: True isUnique: N/A defaultValue: N/A isNullable: False
NOTE: For "IS_ALL_OF", the value shall be a match of the entire list.		

6.2.1.5 Common notifications

6.2.1.5.1 Configuration notifications

This clause presents a list of notifications, defined in TS 28.532 [3], that an MnS consumer may receive. The notification header attribute objectClass/objectInstance shall capture the DN of an instance of a class defined in the present document.

Table 6.2.1.5.1-1

Name	Qualifier	Notes
notifyMOICreation	O	--
notifyMOIDeletion	O	--
notifyMOIAttributeValueChanges	O	--

6.2.2 Scenario specific IntentExpectation definition

6.2.2.1 Scenario specific IntentExpectation definition

6.2.2.1.1 Radio Network Expectation

6.2.2.1.1.1 Definition

Radio Network Expectation is an IntentExpectation which can be used to represent MnS consumer's expectations for radio network (RAN SubNetwork) delivering and performance assurance.

The Radio Network Expectation is defined by utilizing the construct of the generic IntentExpectation <<dataType>> with set of allowed values and concrete dataTypes specified.

Following are the specific allowed values when implemented the IntentExpectation for Radio Network Expectation.

Table 6.2.2.1.1.1-1

Attribute Name	Allowed Values
objectType (CM)	RAN_SUBNETWORK
objectInstance (CM)	DN of the RAN_SUBNETWORK

NOTE: Following are the qualifier description for attribute "objectType" and "objectInstance":

- In case of the intent expectation is not for a specific RAN SubNetwork instance or/and MnS consumer have no knowledge of the DN of this RAN SubNetwork instance, the attribute "objectType" needs to be specified.
- In case of the intent expectation is for a specific RAN SubNetwork instance and MnS consumer have the knowledge of the DN of this RAN SubNetwork instance, the attribute "objectInstance" needs to specified.

6.2.2.1.1.2 ObjectContexts

Following provides the concrete ObjectContexts for Radio Network Expectation based on the common structure of ObjectContext. The properties of the attributes in the following table should be same with properties of ObjectContexts defined in clause 6.2.1.3. The usage of following contexts for corresponding use cases see Table 8.1 Guidelines for using scenario specific intent expectation for intent driven use cases.

Table 6.2.2.1.1.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
coverageAreaPolygonContext	O	T	T	F	F
coverageTACContext	O	T	T	F	F
timeBasedAreaContext	O	T	T	F	F
cellContext	O	T	T	F	F
pLMNContext	O	T	T	F	F
dlFrequencyContext	O	T	T	F	F
ulFrequencyContext	O	T	T	F	F
rATContext	O	T	T	F	F
uEGroupContext	O	T	T	F	F

6.2.2.1.1.3 ExpectationTargets

Following provides the concrete ExpectationTargets for Radio Network Expectation based on the common structure of ExpectationTarget. The properties of the attributes in the following table should be the same with properties of ExpectationTargets defined in clause 6.2.1.3. The usage of following targets for corresponding use cases see Table 8.1 Guidelines for using scenario specific intent expectation for intent driven use cases.

Table 6.2.2.1.1.3-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
weakRSRPRatioTarget	O	T	T	F	F
lowSINRRatioTarget	O	T	T	F	F
aveULRANUEThptTarget	O	T	T	F	F
aveDLRANUEthptTarget	O	T	T	F	F
lowULRANUEThptRatioTarget	O	T	T	F	F
lowDLRANUEThptRatioTarget	O	T	T	F	F
highUlPrbLoadRatioTarget	O	T	T	F	F
highDlPrbLoadRatioTarget	O	T	T	F	F
aveUlPrbLoadTarget	O	T	T	F	F
aveDlPrbLoadTarget	O	T	T	F	F
rANEnergyConsumptionTarget	O	T	T	F	F
rANEnergyEfficiencyTarget	O	T	T	F	F
activeUEsNumTarget	O	T	T	F	F
pRBsTarget	O	T	T	F	F
interRAThandoverTarget	O	T	T	F	F

6.2.2.1.1.4 ExpectationContexts

Following provides the concrete ExpectationContexts for Radio Network Expectation based on the common structure of ExpectationContext. The attribute properties defined in the table below should be the same as the properties defined for ExpectationContexts in clause 6.2.1.3.

Table 6.2.2.1.1.4-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`targetAssuranceTimeContext`	O	T	T	F	F
`schedulingTimeContext`	O	T	T	F	F

6.2.2.1.2 Edge Service Support Expectation

6.2.2.1.2.1 Definition

Edge Service Support Expectation is an IntentExpectation which can be used to represent MnS consumer's expectations for edge service deployment.

The Edge Service Support Expectation is defined utilizing the constructs of the generic IntentExpectation <<dataType>> with set of allowed values and concrete dataTypes specified.

Following are the specific allowed values when implemented the IntentExpectation for Edge Service Support Expectation.

Table 6.2.2.1.2.1-1

Attribute	Allowed Values
objectType (CM)	EDGE_SERVICE_SUPPORT
objectInstance (CM)	DN of the EDGE_SERVICE_SUPPORT

NOTE: Following are the qualifier description for attribute "objectType" and "objectInstance":

- In case of the intent expectation is not for a specific service instance or/and MnS consumer have no knowledge of the DN of this service instance, the attribute "objectType" needs to be specified.
- In case of the intent expectation is for a specific service instance and MnS consumer have the knowledge of the DN of this service instance, the attribute "objectInstance" needs to be specified.

6.2.2.1.2.2 ObjectContexts

Following provides the concrete ObjectContexts for Edge Service Support Expectation based on the common structure of ObjectContext. The properties of the attributes in the following table should be the same as the properties of ObjectContexts defined in clause 6.2.1.3.

Table 6.2.2.1.2.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`edgeIdentificationIdContext`	CM	T	T	F	F
`edgeIdentificationLocContext`	CM	T	T	F	F
`coverageAreaTAContext`	CM	T	T	F	F

NOTE: Following are the qualifier description for attribute "`edgeIdentificationIdContext`" and "`edgeIdentificationLocContext`":

- In case of the Service deployment is needed at a particular edge data network, the attribute "`edgeIdentificationIdContext`" needs to be specified.

- In case of the Service deployment is needed at a particular location, the attribute "`edgeIdentificationLocContext`" needs to be specified.

- In case of the Service deployment is needed at a particular tracking area, the attribute "`coverageAreaTAContext`" needs to be specified.

6.2.2.1.2.3 ExpectationTargets

Following provides the concrete ExpectationTargets for Edge Service Support Expectation based on the common structure of ExpectationTarget. The attribute properties defined in the table below should be the same as the properties defined for ExpectationTargets in clause 6.2.1.3.

Table 6.2.2.1.2.3-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`dlThptPerUETarget`	O	T	T	F	F
`UlThptPerUETarget`	O	T	T	F	F
`dLLatencyTarget`	O	T	T	F	F
`uLLatencyTarget`	O	T	T	F	F
`maxNumberofUEsTarget`	O	T	T	F	F
`activityFactorTarget`	O	T	T	F	F
`uESpeedTarget`	O	T	T	F	F

6.2.2.1.2.4 ExpectationContexts

Following provides the concrete ExpectationContexts for Service Deployment Expectation based on the common structure of ExpectationContext. The attribute properties defined in the table below should be the same as the properties defined for ExpectationContexts in clause 6.2.1.3.

Table 6.2.2.1.2.4-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
`serviceStartTimeContext`	O	T	T	F	F
`serviceEndTimeContext`	O	T	T	F	F
`uEMobilityLevelContext`	O	T	T	F	F
`resourceSharingLevelContext`	O	T	T	F	F

6.2.2.1.3 End-to-end Network Resource Optimization Expectation

6.2.2.1.3.1 Definition

End-to-end Network Resource Optimization Expectation is an IntentExpectation which can be used to represent MnS consumer's expectations for the resource optimization of network resources across multiple network domains, i.e., MnS consumer expresses its intent containing an intent expectation with targets on the whole network comprising RAN and Core Network.

The End-to-end Network Resource Optimization Expectation is defined utilizing the constructs of the generic IntentExpectation <<dataType>> with set of allowed values and concrete dataTypes specified. It enables optimization among desired targets for end-to-end network optimization based on a list of expectation targets that are provided in an ordered list where those at the top of the list are the most important.

Following are the specific allowed values when implemented the IntentExpectation for End-to-end Network Resource Optimization Expectation.

Table 6.2.2.1.3.1-1

Attribute	Allowed Values
objectType (CM)	Subnetwork
objectInstance (CM)	DN of the Subnetwork

NOTE: Following are the qualifier description for attribute "objectType" and "objectInstance":

- In case of the intent expectation is not for a specific subnetwork instance or/and MnS consumer have no knowledge of the DN of this subnetwork instance, the attribute "objectType" needs to be specified with the value as "subnetwork".
- In case of the intent expectation is for a specific Subnetwork instance and MnS consumer have the knowledge of the DN of this Subnetwork instance, the attribute "objectInstance" needs to be specified.

6.2.2.1.3.2 ObjectContexts

Based on the common structure of ObjectContext, the concrete ObjectContexts for end-to-end Network Resource Optimization Expectation is the combination of ObjectContext defined for the RAN in clause 6.2.2.1.1.2 and for the 5GC in clause 6.2.2.1.4.2.

6.2.2.1.3.3 ExpectationTargets

Following provides the concrete ExpectationTargets for end-to-end Network Resource Optimization Expectation based on the common structure of ExpectationTarget. The attribute properties defined for these ExpectationTargets should be the same with the properties defined for ExpectationTargets in clause 6.2.1.3.

Editor's Note: Which KPIs defined in TS 28.554 can be used is FFS.

6.2.2.1.4 5GC Network Expectation

6.2.2.1.4.1 Definition

5GC Network Expectation is an IntentExpectation which can be used to represent MnS consumer's expectations for 5GC network (5GC SubNetwork) delivering

The 5GC Network Expectation is defined by utilizing the construct of the generic IntentExpectation <<dataType>> with set of allowed values and concrete dataTypes specified.

Following are the specific allowed values when implemented the IntentExpectation for 5GC Network Expectation.

Table 6.2.2.1.4.1-1

Attribute Name	Allowed Values
`objectType (CM)`	5GC_SUBNETWORK
`objectInstance (CM)`	DN of the 5GC_SUBNETWORK

NOTE: Following are the qualifier description for attribute "objectType" and "objectInstance":

- In case of the intent expectation is not for a specific 5GC SubNetwork instance or/and MnS consumer have no knowledge of the DN of this 5GC SubNetwork instance, the attribute "objectType" needs to be specified.
- In case of the intent expectation is for a specific 5GC SubNetwork instance and MnS consumer have the knowledge of the DN of this 5GC SubNetwork instance, the attribute "objectInstance" needs to specified.

6.2.2.1.4.2 ObjectContexts

Following provides the concrete ObjectContexts for 5GC Network Expectation based on the common structure of ObjectContext. The properties of the attributes in the following table should be same with properties of ObjectContexts defined in clause 6.2.1.3.

Table 6.2.2.1.4.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
nfTypeContext	O	T	T	F	F
nfInstanceLocationContext	O	T	T	F	F
pLMNContext	O	T	T	F	F
taiContext	O	T	T	F	F
servingScopeContext	O	T	T	F	F
dnnContext	O	T	T	F	F

6.2.2.1.4.3 ExpectationTargets

Following provides the concrete ExpectationTargets for 5GC Network Expectation based on the common structure of ExpectationTarget. The properties of the attributes in the following table should be the same with properties of ExpectationTargets defined in clause 6.2.1.3.

Table 6.2.2.1.4.3-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
maxNumberofPDUsessionsTarget	O	T	T	F	F
maxNumberofRegisteredsubscribersTarget	O	T	T	F	F
incomingDataTarget	O	T	T	F	F
outgoingDataTarget	O	T	T	F	F

6.2.2.1.4.4 ExpectationContexts

Following provides the concrete ExpectationContexts for 5GC network Expectation based on the common structure of ExpectationContext. The attribute properties defined in the table below should be the same as the properties defined for ExpectationContexts in clause 6.2.1.3.

Table 6.2.2.1.4.4-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
startTimeContext	O	T	T	F	F
resourceSharingLevelContext	O	T	T	F	F

6.2.2.1.5 Radio Service Expectation

6.2.2.1.5.1 Definition

Radio Service Expectation is an IntentExpectation which can be used to represent MnS consumer's expectations for radio service (radio network as a service) delivering and assurance in the specified area.

The Radio Service Expectation is defined by utilizing the construct of the generic IntentExpectation <<dataType>> with set of allowed values and concrete dataTypes specified.

Following are the specific allowed values when implemented the IntentExpectation for Radio Service Expectation.

Table 6.2.2.1.5.1-1

Attribute Name	Allowed Values
objectType	RADIO_SERVICE

6.2.2.1.5.2 ObjectContexts

Following provides the concrete ObjectContexts for Radio Service Expectation based on the common structure of ObjectContext. The properties of the attributes in the following table should be the same with the properties of ObjectContexts defined in clause 6.2.1.3.

Table 6.2.2.1.5.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
coverageAreaPolygonContext	CM	T	T	F	F
dlFrequencyContext	CO	T	T	F	F
ulFrequencyContext	CO	T	T	F	F
cellContext	CM	T	T	F	F
geoCoordinateContext	CM	T	T	F	F
serviceTypeContext	M	T	T	F	F
uEGroupContext	M	T	T	F	F

NOTE: Following are the qualifier description for above attributes:

- In case of a coverage area described in the form of polygon, the attribute "coverageAreaPolygonContext" needs to be supported. In additional, the attribute "dlFrequencyContext" and "ulFrequencyContext" need to be supported when the radio service intent is applied for a specific frequency of the specified coverage area.
- In case of a coverage area described in the form of a list of cells, the attribute "cellContext" needs to be supported.
- In case of coverage area described in the form of geocoordinate, the attribute "geoCoordinateContext" needs to be supported.

6.2.2.1.5.3 ExpectationTargets

Following provides the concrete ExpectationTargets for Radio Service Expectation based on the common structure of ExpectationTarget. The properties of the attributes in the following table should be the same with the properties of ExpectationTargets defined in clause 6.2.1.3.

Table 6.2.2.1.5.3-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
dLLatencyTarget	O	T	T	F	F
uLLatencyTarget	O	T	T	F	F
dLThptPerUETarget	O	T	T	F	F
uLThptPerUETarget	O	T	T	F	F
numberofUEsTarget	O	T	T	F	F

NOTE: At least one of above targets needs to be supported.

6.2.2.1.5.4 ExpectationContexts

Following provides the concrete ExpectationContexts for Radio Service Expectation based on the common structure of ExpectationContext. The attribute properties defined in the table below should be the same as the properties defined for ExpectationContexts in clause 6.2.1.3.

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
schedulingTimeContext	O	T	T	F	F

6.2.2.1.6 Network Maintenance Expectation

6.2.2.1.6.1 Definition

Network Maintenance Expectation is an IntentExpectation which can be used to represent MnS consumer's expectation for maintaining a network.

The Network Maintenance Expectation is defined by utilizing the construct of the generic IntentExpectation <<dataType>> with a set of allowed valued and concrete dataTypes specified.

Following are the specific allowed values when implemented the IntentExpectation for Network Maintenance.

Table 6.2.2.1.6.1-1

Attribute Name	Allowed Values
objectInstance (M)	DN of the SubNetwork

NOTE: Following are the qualifier description for attribute "objectInstance":

- Network maintenance expectation is for a specific SubNetwork instance and MnS consumer has the knowledge of the DN of this SubNetwork instance, the attribute "objectInstance" needs to be specified.

6.2.2.1.6.2 ObjectContexts

Based on common structure of ObjectContexts, the concrete ObjectContext for Network Maintenance Expectation is the:

- ObjectContext defined for RAN in clause 6.2.2.1.1.2, including coverageAreaPolygonContext, coverageTACContext, pLMNContext, dlFrequencyContext, ulFrequencyContex, or
- ObjectContext defined for the 5GC in clause in 6.2.2.1.4.2, including nfTypeContext, nfInstanceLocationContext, pLMNContext, taiContext, servingScopeContext and dnnContext.

Following attributes provided in Table 6.2.2.1.6.2-1 based on the common structure of ObjectContext is applied for both RAN and 5GC. The properties of the attributes in the following table should be same with properties of ObjectContexts defined in clause 6.2.1.3.4. The usage of following contexts for corresponding use cases, see Table 8.1 Guidelines for using scenario specific intent expectation for intent driven use cases.

Table 6.2.2.1.6.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
maintenanceVersionContext	O	T	T	F	F
maintenanceOrderContext	O	T	T	F	F
maintenanceTypeContext	O	T	T	F	F

6.2.2.1.6.3 ExpectationTargets

Based on the common structure of ExpectationTargets, the concrete ExpectationTarget for Network Maintenance Expectation is the:

- ExpectationTarget defined for RAN in clause 6.2.2.1.1.3, or,
- ExpectationTarget defined for 5GC in clause 6.2.2.1.4.3.

The following attribute properties provided in Table 6.2.2.1.6.3-1 based on common structure of ExpectationTarget are applied for both RAN and 5GC. The properties of the attributes in the following table should be the same with the properties defined for ExpectationTargets in clause 6.2.1.3.3. The usage of following targets for corresponding use cases, see Table 8.1 Guidelines for using scenario specific intent expectation for intent driven use cases.

Table 6.2.2.1.6.3-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
maintenanceVersionTarget	O	T	T	F	F

6.2.2.1.6.4 ExpectationContexts

Following provides the concrete ExpectationContexts for Network Maintenance Expectation based on common structure of ExpectationContext. The attribute properties defined in the table below should be the same as the properties defined for ExpectationContext in Clause 6.2.1.3.

Table 6.2.2.1.6.4-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
maintenanceTimeContext	O	T	T	F	F

6.2.2.2 Attribute definition

Table 6.2.2.2-1

Attribute Name	Documentation and Allowed Values	Properties
coverageAreaPolygonContext	It describes the coverage areas for the RAN SubNetwork that the intent expectation is applied in the form of polygon. CoverageAreaPolygonContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "coverageAreaPolygon" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of GeoArea defined in 3GPP TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
coverageTACContext	It describes the coverage areas for the RAN SubNetwork that the intent expectation is applied in the form of TAC. CoverageTACContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "coverageTAC" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of TAC defined in 3GPP TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
timeBasedAreaContext	It describes the areas for specific times for the RAN SubNetwork that the intent expectation is applied in the form of a <GeoArea, TimeWindow> list. TimeBasedAreaContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: " timeBasedAreaContext " - contextCondition: "IS_ALL_OF" - contextValueRange: a list of GeoArea and TimeWindow pairs, where GeoArea and TimeWindow is defined in 3GPP TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
plMNContext	It describes the PLMN(s) supported by the RAN SubNetwork that the intent expectation is applied. PLMNContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "pLMN" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of PLMNId defined in TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
cellContext	It describes the coverage areas for the RAN SubNetwork that the intent expectation is applied in the form of a list of cells (including E-UTRAN cells identified by E-UTRAN-CGI and NR cells identified by NG-RAN CGI). CellContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "cell" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of EutraCellId or NrCellId defined in 3GPP TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
dlFrequencyContext	It describes the downlink frequency information (RF reference frequencies and/or the frequency operating band) supported by the RAN SubNetwork that the intent expectation is applied. dLFrequencyContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "dLFrequency" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of Frequency defined in clause 6.2.1.3.13	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
ulFrequencyContext	It describes the uplink frequency information (RF reference frequencies and/ or the frequency operating band) supported by the RAN SubNetwork that the intent expectation is applied. uLFrequencyContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: uLFrequency" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of Frequency defined in clause 6.2.1.3.13	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
rATContext	It describes the RAT supported by the RAN SubNetwork that the intent expectation is applied. RATContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "rAT" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of ENUM with allowed value: UTRAN, EUTRAN and NR	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
uEGroupContext	It describes the UE Groups (represented by specific 5QI, specific S-NSSAI, or specific combination of S-NSSAI and 5QI) that the intent expectation is applied. UEGroupContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values:	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
	- contextAttribute: "UEGroup" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of UEGroup <<dataType>>	
targetAssuranceTimeContext	It describes the timeWindows (including startTime, endTime) when the targets in the Intent Expectation need to be assured. - contextAttribute: "targetAssuranceTime" - contextCondition: "IS_EQUAL_TO" - contextValueRange: a list of TimeWindow(s) defined in TS 28.622 [6].	type: Context multiplicity: * isOrdered: False isUnique: True defaultValue: None isNullable: True
weakRSRPRatioTarget	It describes the downlink weak coverage ratio target for the RAN SubNetwork that the intent expectation is applied. The numerator is the number of the cells with downlink weak RSRP, and the denominator is the total number of cells of the RAN Subnetwork in the specified area. WeakRSRPRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext. Multiple weakRSRPRatioTargets can be included in the same RadioNetworkExpectation with different targetValueRanges and corresponding WeakRSRPContexts. Following are the allowed values: - targetName: "weakRSRPRatio" - targetCondition: "IS_LESS_THAN" - targetValueRange: integer with allowed value [0,100] % - targetContext: WeakRSRPContext	type: ExpectationTarget multiplicity: 1..* isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
weakRSRPRatioTarget.weakRSRPContext	It describes the threshold for downlink weak RSRP of the cells (see RSRP measurements in TS 28.552 [6]) of the RAN SubNetwork that the intent expectation is applied. WeakRSRPContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "weakRSRPThreshold" - contextCondition: "IS_LESS_THAN" - contextValueRange: Float	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
lowSINRRatioTarget	It describes the low SINR ratio target for the RAN SubNetwork that the intent expectation is applied. The numerator is the number of the cells with low SINR, and the denominator is the total number of cells of the RAN Subnetwork in the specified area. LowSINRRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContxt. Multiple lowSINRRatioTargets can be included in the same RadioNetworkExpectation with different targetValueRanges and corresponding LowSINRContexts. Following are the allowed values: - targetName: "lowSINRRatio" - targetCondition: "IS_LESS_THAN" - targetValueRange: integer with allowed value [0,100] - targetContext: LowSINRContext	type:ExpectationTarget multiplicity: 1..* isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
lowSINRRatioTarget.lowSINRContext	It describes the threshold for low SINR of the cells (see SINR measurements in TS 28.552 [6]) of the RAN SubNetwork that the intent expectation is applied. LowSINRContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "lowSINRThreshold" - contextCondition: "IS_LESS_THAN" - contextValueRange: integer	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
aveULRANUEThptTarget	It describes the average UL RAN UE throughput target for RAN SubNetwork (see UL RAN UE throughput for a sub-network in TS 28.554[11]) that the intent expectation is applied. AveULRANUEThptTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContxt. Multiple aveULRANUEThptTaregts can be included in the same RadioNetworkExpectations with different targetValueRanges and corresponding ulFrequencyContexts or rATContexts. Following are the allowed values: - targetName: "aveULRANUEThpt" - targetCondition: "IS_GREATER_THAN" - targetValueRange: integer - targetContext: ulFrequencyContext or rATContext	type: ExpectationTarget multiplicity: 1..* isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
aveDLRANUEThptTarget	It describes the average DL RAN UE throughput target for RAN SubNetwork (see DL RAN UE throughput for a sub-network in TS 28.554[11]) that the intent expectation is applied. AveDLRANUEThptTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext. Multiple aveDLRANUEThptTaregts can be included in the same RadioNetworkExpectations with different targetValueRanges and corresponding ulFrequencyContexts or rATContexts. Following are the allowed values: - targetName: "aveDLRANUEThpt" - targetCondition: "IS_GREATER_THAN" - targetValueRange: integer - targetContext: dlFrequencyContext or rATContext	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
lowULRANUEThptRatioTarget	It describes the low UL RAN UE throughput ratio target for the RAN SubNetwork that the intent expectation is applied. The numerator is the number of the cells with low UL RAN UE throughput, and the denominator is the total number of cells of the RAN Subnetwork in the specified area. LowULRANUEThptRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext. Following are the allowed values: - targetName: "lowULRANUEThptRatio" - targetCondition: "IS_LESS_THAN" - targetValueRange: integer with allowed value [0,100] % - targetContext: LowULRANUEThptContext	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
lowULRANUEThptRatioTarget.lowULRANUEThptContext	It describes the threshold for the low UL RAN UE throughput cells (see average UL RAN UE throughput in gNB and distribution of UL UE throughput in gNB in TS 28.552[6]) of the RAN SubNetwork that the intent expectation is applied LowULRANUEThptContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "lowULRANUEThptThreshold" - contextCondition: "IS_LESS_THAN" - contextValueRange: Float	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
lowDLRANUEThptRatioTarget	It describes the low DL RAN UE throughput ratio target for the RAN SubNetwork that the intent expectation is applied. The numerator is the number of the cells with low DL RAN UE throughput, and the denominator is the total number of cells of the RAN Subnetwork in the specified area.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
	LowDLRANUEThptRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext. Following are the allowed values: - targetName: "lowDLRANUEThptRatio" - targetCondition: "IS_LESS_THAN " - targetValueRange: integer with allowed value [0,100] - targetContext: LowDLRANUEThptContext	
lowDLRANUEThptRatioTarget.lowDLRANUEThptContext	It describes the threshold for the low DL RAN UE throughput cells (see average DL RAN UE throughput in gNB and distribution of DL UE throughput in gNB in TS 28.552[6]) of the RAN SubNetwork that the intent expectation is applied. LowDLRANUEThptContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "lowDLRANUEThptThreshold" - contextCondition: "IS_LESS_THAN" - contextValueRange: Float	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioNetworkExpectation.activeUEsNumTarget	It describes the number of active UEs for the specified areas. This target is related to mean number of active UEs in the DL per cell and mean number of active UEs in the UL per cell defined in 3GPP TS 28.552 [6]. ActiveUEsNumTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: " activeUEsNum" - targetCondition: " IS_WITHIN_RANGE " - targetValueRange: a pair of integer values to represent minimum number of active UEs and maximum number of active UEs.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioNetworkExpectation.schedulingTimeContext	It describes the scheduled times (including one-time interval, daily periodicity, weekly periodicity or monthly periodicity) for the Radio Network that the intent expectation is applied. For details see SchedulingTime <<choice>> defined in clause 4.3.63 of TS 28.622 [6]. schedulingTimeContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "schedulingTime" - contextCondition: " IS_ALL_OF " - contextValueRange: a list of SchedulingTime <<choice>> defined in TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
nfTypeContext	It identifies the types of NF supported by the 5GC SubNetwork that the intent expectation is applied. nfTypeContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: " nfType " - contextCondition:" IS_ALL_OF " - contextValueRange: a list of ENUM with allowed value: Enumeration NFType in 3GPP TS 29.510[13]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
nfInstanceLocationContext	It describes the location of NF instance supported by the 5GC SubNetwork that the intent expectation is applied. nfInstancelocationContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: " nfInstanceLocation " - contextCondition:" IS_ALL_OF "	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
	- contextValueRange: a list of string. See Locality in TS 29.510 [13]	
taiContext	It describes the tracking area Identifiers supported by the 5GC SubNetwork that the intent expectation is applied. taiContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "tai" - contextCondition:" IS_ALL_OF " - contextValueRange: a list of tai defined in TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
maxNumberofPDUsessionsTarget	It describes the maximum number of PDU sessions for 5GC SubNetwork supporting that the intent expectation is applied. For details, see maxNumberofPDUsessions in clause 5.3.1.2 in TS 28.552 [12] maxNumberofPDUsessionsTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "maxNumberofPDUsessions" - targetCondition: " IS_LESS_THAN" - targetValueRange: integer - targetContext: 5GSessionContext.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
maxNumberofPDUsessionsTarget.5GSessionContext	It describes the maximum supported 5G PDU session of the 5GC SubNetwork related to the intent expectation. 5GSessionContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "5GSession" - contextCondition: "IS_ LESS_THAN" - contextValueRange: integer	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
maxNumberofRegisteredsubscribersTarget	It describes the maximum number of Registered subscribers for 5GC SubNetwork supporting that the intent expectation is applied. For details, see maxNumberofRegisteredsubscribers in clause 5.6.2 in TS 28.552 [12] maxNumberofRegisteredsubscribersTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "maxNumberofRegisteredsubscribers" - targetCondition: " IS_LESS_THAN" - targetValueRange: Integer	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
highUlPrbLoadRatioTarget	It describes the high UL PRB load ratio target (as percentage) for the RAN SubNetwork that the intent expectation is applied. The numerator is the number of the cells with high UL PRB load, and the denominator is the total number of cells of the RAN Subnetwork in the specified area. HighUlPrbLoadRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition,targetValueRange and targetContext. Multiple HighUlPrbLoadRatioTargets can be included in the same RadioNetworkExpectations with different targetValueRanges and corresponding HighUlPrbLoadContexts. Following are the allowed values: - targetName: "highUlPrbLoadRatio" - targetCondition: "IS_LESS_THAN " - targetValueRange: integer with allowed value [0,100] % - targetContext: HighUlPrbLoadContext	type: ExpectationTarget multiplicity: 1..* isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
highUlPrbLoadRatioTarget.HighUlPrbLoadContext	It describes the threshold for high uplink PRB load (i.e. UL Total PRB Usage in TS 28.552 [12] to represent the percentage of UL PRBs used)	type: Context multiplicity: 1 isOrdered: N/A

Attribute Name	Documentation and Allowed Values	Properties
	of the cells of the RAN SubNetwork in the specified area that the intent expectation is applied. HighUlPrbLoadContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "HighUlPrbLoad" - contextCondition: "IS_GREATER_THAN" - contextValueRange: integer with allowed value [0,100] %	isUnique: N/A defaultValue: None isNullable: True
highDlPrbLoadRatioTarget	It describes the high DL PRB load ratio target (as percentage) for the RAN SubNetwork that the intent expectation is applied. The numerator is the number of the cells with high DL PRB load, and the denominator is the total number of cells of the RAN Subnetwork in the specified area. HighDlPrbLoadRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext. Multiple HighDlPrbLoadRatioTargets can be included in the same RadioNetworkExpectations with different targetValueRanges and corresponding HighDlPrbLoadContexts. Following are the allowed values: - targetName: "highDlPrbLoadRatio" - targetCondition: "IS_LESS_THAN " - targetValueRange: integer with allowed value [0,100] % - targetContext: HighDlPrbLoadContext	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
highDlPrbLoadRatioTarget.HighDlPrbLoadContext	It describes the threshold for high downlink PRB load (i.e. DL Total PRB Usage in TS 28.552 [12] to represent the percentage of DL PRBs used) of the cells of the RAN SubNetwork in the specified area that the intent expectation is applied. HighDlPrbLoadContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "HighDlPrbLoad" - contextCondition: "IS_GREATER_THAN" - contextValueRange: integer with allowed value [0,100] %	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
aveUlPrbLoadTarget	It describes the average uplink PRB load target (i.e. UL Total PRB Usage in TS 28.552 [12] to represent the percentage of UL PRBs used) of the cells of the RAN SubNetwork that the intent expectation is applied. AveULPrbLoadTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "aveULPrbLoad" - targetCondition: "IS_LESS_THAN" - targetValueRange: integer with allowed value [0,100] %	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
aveDlPrbLoadTarget	It describes the average dowlink PRB load (i.e. DL Total PRB Usage in TS 28.552 [12] to represent the percentage of DL PRBs used) target for RAN SubNetwork that the intent expectation is applied. AveDLPrbLoadTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "aveDLPrbLoad" - targetCondition: "IS_LESS_THAN" - targetValueRange: integer with allowed value [0,100] %	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
rANEnergyConsumptionTarget	It describes the RAN energy consumption target for RAN SubNetwork that the intent expectation is applied. The definition for RAN energy consumption see $EC_{NG\text{-}RAN}$ in clause 6.7.3.4.1 in TS 28.554 [11].	type: ExpectationTarget multiplicity: 1

Attribute Name	Documentation and Allowed Values	Properties
	RANEnergyConsumptionTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "rANEnergyConsumption" - targetCondition: "IS_LESS_THAN" - targetValueRange: Integer	isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
rANEnergyEfficiencyTarget	It describes the RAN energy efficiency target for RAN SubNetwork that the intent expectation is applied. The unit of this target is bit/J. The definition for RAN energy efficiency target for RAN SubNetwork see $EE_{MN,DV}$ in clause 6.7.1.1 in TS 28.554 [11] RANEnergyEfficiencyTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: " rANEnergyEfficiency " - targetCondition: " IS_GREATER_THAN" - targetValueRange: Integer	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
serviceStartTimeContext	This describes the start time at which the service shall be available. This contributes to the selection of the appropriate edge data network to be used for service deployment. Following are the allowed values: - contextAttribute: "serviceStartTime" - contextCondition: "IS_EQUAL_TO" - contextValueRange: DateTime	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
serviceEndTimeContext	This describes the end time after which the service shall not be available. This contributes to the selection of the appropriate edge data network to be used for service deployment. Following are the allowed values: - contextAttribute: "serviceEndTime" - contextCondition: "IS_EQUAL_TO" - contextValueRange: DateTime	type:Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
edgeIdentificationIdContext	This identifies the edge network where the service needs to be deployed. For details see EDNidentifier defined in TS 28.538 [9]. This should be used when the edge identification is known to the consumer Following are the allowed values: - contextAttribute: "edgeIdentificationId" - contextCondition: "IS_EQUAL_TO" - contextValueRange: String	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
edgeIdentificationLocContext	This identifies the location where the service needs to be deployed. This should be used when the edge identification is not known to the consumer Following are the allowed values: - contextAttribute: "edgeIdentificationLoc" - contextCondition: "IS_EQUAL_TO" - contextValueRange: GeoCoordinate dfined in TS 28.622 [6].	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
coverageAreaTAContext	It describes Tracking Coverage Areas for service supporting that the intent expectation is applied. coverageAreaTAContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "coverageAreaTA" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of TAC defined in 3GPP TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
EdgeServiceSupport Expectation.dlThptPerUETarget	It describes the DL throughput target by the per UE for the edge service Supporting that the intent expectation is applied. For details see dlThptPerUE defined in clause 6.3.1 of TS 28.541 [5].	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A

Attribute Name	Documentation and Allowed Values	Properties
	DLThptperUETarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange: - targetName: "DLThptperUE" - targetCondition: "IS_GREATER_THAN" - targetValueRange: Integer.	defaultValue: None isNullable: True
EdgeServiceSupport Expectation. ulThptPerUETarget	It describes the UL throughput target by the per UE for the edge service Supporting that the intent expectation is applied. For details see ulThptPerUE defined in clause 6.3.1 of TS 28.541 [5]. ULThptperUETarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. - targetName: "ulThptperUE" - targetCondition: "IS_GREATER_THAN" - targetValueRange: Integer.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
EdgeServiceSupport Expectation. dLLatencyTarget	It describes the DL latency target for the edge service Supporting that the intent expectation is applied. DLLatencyTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. For details see attribute dlLatency defined in clause 6.3.1 of TS 28.541 [5]. - targetName: "dLLatency" - targetCondition: "IS_LESS_THAN" - targetValueRange: Integer.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
EdgeServiceSupport Expectation. uLLatencyTarget	It describes the UL latency target for the edge service Supporting that the intent expectation is applied. For details see attribute ulLatency defined in clause 6.3.1 of TS 28.541 [5]. uLLatencyTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. - targetName: "uLLatency" - targetCondition: "IS_LESS_THAN" - targetValueRange: Integer.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
EdgeServiceSupport Expectation. maxNumberofUEsTarget	It describes the the number of UEs for edge service supporting that the intent expectation is applied. For details see attribute maxNumberofUE defined in clause 6.3.1 of of TS 28.541 [5]. maxNumberofUEsContext is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "maxNumberofUEs" - targetCondition: " IS_LESS_THAN" - targetValueRange: Integer.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
EdgeServiceSupport Expectation. activityFactorTarget	It describes the percentage value of the amount of simultaneous active UEs to the total number of UEs where active means the UEs are exchanging data with the network for service supporting that the intent expectation is applied. For details see activityFactor in clause 6.3.1 in TS 28.541 [5.] activityFactorTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: " activityFactor " - targetCondition: " IS_EQUAL_TO" - targetValueRange: Integer	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
EdgeServiceSupport Expectation. uESpeedTarget	It describes the speed (in km/hour) supported for edge service supporting that the intent expectation is applied. For details see uESpeed in clause 6.3.1 in TS 28.541[5]. uESpeedTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values:	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
	- targetName: "uESpeed" - targetCondition: " IS_LESS_THAN" - targetValueRange: Integer	
EdgeServiceSupport Expectation. uEMobilityLevelContext	It describes the mobility level of UE for edge service supporting that the intent expectation is applied. For details see uEMobilityLevel in clause 6.3.1 in TS 28.541 [5.] uEMobilityLevelContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: " uEMobilityLevel " - contextCondition: "IS_EQUAL_TO" - contextValueRange: ENUM.	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
EdgeServiceSupport Expectation. resourceSharingLevelContext	It describes the resource sharing level for which the intent expectation is applied. For details see resourceSharinglevel in clause 6.3.1 in TS 28.541 [5]. resourceSharingLevelContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "resourceSharingLevel" - contextCondition: "IS_EQUAL_TO" - contextValueRange: ENUM	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.coverageAreaPolygonContext	It describes the coverage areas for the Radio Service that the intent expectation is applied in the form of polygon. CoverageAreaPolygonContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "coverageAreaPolygon" - contextCondition: "IS_ALL_OF" - contextValueRange: a list of CoverageArea defined in 3GPP TS 28.622 [6].	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.serviceTypeContext	It describes the service type for the Radio Service that the intent expectation is applied. For details see sST in clause 6.4.1 in TS 28.541 [5]. ServiceTypeContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "serviceType" - contextCondition: "IS_EQUAL_TO" - contextValueRange: string	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.uEGroupContext	It describes the UE Groups (represented by specific 5QI, specific S-NSSAI, specific PLMNId or a specific combination of S-NSSAI, 5QI, and and PLMNId) supported by the Radio Service that the intent expectation is applied. ServiceTypeContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: UEGroup - contextCondition: " IS_ALL_OF " - contextValueRange: a list of UEGroup <<dataType>>	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.dlThptPerUETarget	It describes the DL throughput target per UE for the Radio Service that the intent expectation is applied. DLThptperUETarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange: - targetName: "DLThptperUE" - targetCondition: "IS_GREATER_THAN" - targetValueRange: Float.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

Attribute Name	Documentation and Allowed Values	Properties
RadioServiceExpectation.ulThptPerUETarget	It describes the UL throughput target per UE for the Radio Service that the intent expectation is applied. ULThptperUETarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. - targetName: "ulThptperUE" - targetCondition: "IS_GREATER_THAN" - targetValueRange: Float.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.dLLatencyTarget	It describes the DL latency target for the Radio Service that the intent expectation is applied. DLLatencyTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. - targetName: "dLLatency" - targetCondition: "IS_LESS_THAN" - targetValueRange: Integer.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.uLLatencyTarget	It describes the UL latency target for the Radio Service that the intent expectation is applied. uLLatencyTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. - targetName: "uLLatency" - targetCondition: "IS_LESS_THAN" - targetValueRange: Integer.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.numberofUEsTarget	It describes the number of UEs target for the Radio Service that the intent expectation is applied. numberofUEsContext is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "numberofUEs" - targetCondition: " IS_LESS_THAN" - targetValueRange: Integer.	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.schedulingTimeContext	It describes the scheduled times (including one-time interval, daily periodicity, weekly periodicity or monthly periodicity) for the Radio Service that the intent expectation is applied. For details see SchedulingTime <<choice>> defined in clause 4.3.63 of TS 28.622 [6]. schedulingTimeContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "schedulingTime" - contextCondition: " IS_ALL_OF " - contextValueRange: a list of SchedulingTime <<choice>> defined in TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
servingScopeContext	It describes the served area(s) of the 5GC NF instance supported by the 5GC SubNetwork that the intent expectation is applied. For detail, see servingScope in TS 29.510[13]. servingScopeContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: " servingScope " - contextCondition:" IS_ALL_OF " - contextValueRange: a list of string.	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
dnnContext	It describes the DNN of the 5GC NF instance supported by the 5GC SubNetwork that the intent expectation is applied. dnnContext is a Context including attributes: contextAtrribute, contextCondition and contextValueRange.	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A

Attribute Name	Documentation and Allowed Values	Properties
	Following are the allowed values: - contextAttribute: " dnn " - contextCondition:" IS_ALL_OF " - contextValueRange: a list of string as specified in 3GPP TS 23.003 [15]	defaultValue: None isNullable: True
incomingDataTarget	It describes the maximum incoming data packets for 5GC SubNetwork related to the intent expectation. For details, see N6 incoming link usage measurement in clause 5.4.2.1 in TS 28.552 [12] incomingDataTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "incomingData" - targetCondition: " IS_LESS_THAN" - targetValueRange: integer	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
outgoingDataTarget	It describes the maximum outgoing data packets for 5GC SubNetwork related to the intent expectation. For details, see N6 outgoing link usage measurement in clause 5.4.2.2 in TS 28.552 [12] outgoingDataTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. Following are the allowed values: - targetName: "outgoingData" - targetCondition: " IS_LESS_THAN" - targetValueRange: integer	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
startTimeContext	This describes the start time at which the expected result of the expectation shall be available. Following are the allowed values: - contextAttribute: "startTime" - contextCondition: "IS_EQUAL_TO" - contextValueRange: DateTime	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
maintenanceVersionContext	It describes the scope of the maintenance objects based on software version. Following are the allowed values: - contextAttribute: "swVersion" - contextCondition:" IS_ALL_OF " - contextValueRange: A list of String	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
maintenanceOrderContext	It describes the list of DNs for NFs/NEs based on their order for maintenance. Following are the allowed values: - contextAttribute: "maintenanceOrder" - contextCondition:" IS_ALL_OF " - contextValueRange: A list of String	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
networkMaintenanceExpectation.maintenanceVersionTarget	It describes the maintenance version target for the network maintenance, such as version of the maintenance object. Following are the allowed values: - targetName: "maintenanceVersion" - targetCondition: "IS_EQUAL_TO" - targetValueRange: String	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
networkMaintenanceExpectation.maintenanceTimeContext	It describes the information of a time window (including start and end time) for network maintenance Following are the allowed values: - contextAttribute: "maintenanceTimes" - contextCondition:" IS_EQUAL_TO "	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None

Attribute Name	Documentation and Allowed Values	Properties
	- contextValueRange: timeWindow as defined in 3GPP TS 28.622 [6]	isNullable: True
RadioNetworkExpectation.pRBsTarget	Iit describes total available number of physical resource blocks (PRBs) for each single cell in a specified area. pPRBsTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. The PRB target is equal to the difference between the DL total available PRB plus UL total available PRB of operator A and the DL total available PRB plus UL total available PRB of operator B. For details see attribute DL total available PRB and UL total available PRB defined in clause 5.1.1.2.6 and 5.1.1.2.8 of TS 28.552 [12]. - targetName: "pRBs" - targetCondition: "IS_EQUAL_TO" - targetValueRange: integer with allowed value 00	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioNetworkExpectation.interRAThandoverTarget	Iit shows how often a handover from 5GS to EPS is successful for each single cell in a specified area. iInterRAThandoverTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange. The InterRAThandover target is equal to the difference between 5GS to EPS handover success rate of operator A and 5GS to EPS handover success rate of operator B. For details see attribute 5GS to EPS handover success rate defined in clause 6.6.4 of TS 28.554 [11]. - targetName: "interRAThandover" - targetCondition: "IS_LESS THAN" - targetValueRange: integer with allowed value [0,100] %	type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
RadioServiceExpectation.geoCoordinateContext	It describes the location (i.e. geocoordinate) that radio service intent is applied. GeoCoordinateContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange. Following are the allowed values: - contextAttribute: "GeoCoordinate" - contextCondition: "IS_EQUAL_TO" - contextValueRange: GeoCoordinate as defined in 3GPP TS 28.622 [6]	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True
maintenanceTypeContext	It describes the type of maintenance expected for the maintenance object. Following are the allowed values: - contextAttribute: "maintenanceType" - contextCondition:" IS_EQUAL_TO " - contextValueRange: a list of ENUM with allowed value: UPGRADE, DOWNGRADE.	type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: True

6.3 Procedures for intent management

6.3.1 Introduction

This clause describes the procedures for intent management.

6.3.2 Create an intent

Figure 6.3.2-1 illustrates the procedure for create a new intent for fulfilment.

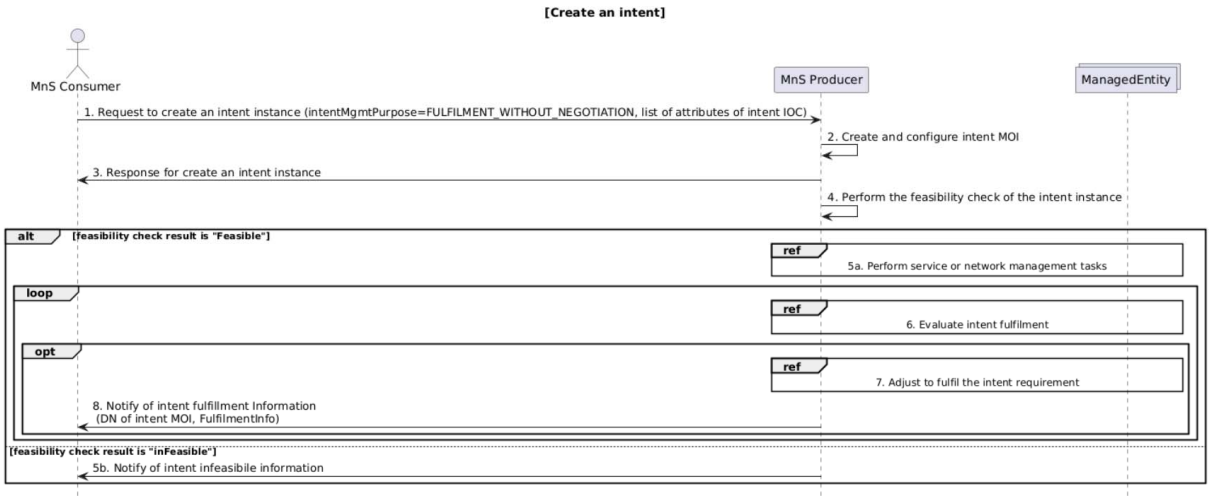


Figure 6.3.2-1: Procedure for create an intent

1. MnS Consumer generates the intent information and sends a request to create an intent instance for fulfilment (see createMOI operation defined in TS 28.532 [3]) to MnS Producer, providing intent information for the new intent to be created. The detailed intent information sees attributes (attribute which "isWritable" property is "True") of the concrete intent IOC defined in clause 6.2. In the provided intent information, the intentMgmtPurpose should be specified as "FULFILMENT_WITHOUT_NEGOTIATION" or NULL, indicating that the MnS Consumer request the MnS Producer to fulfil the intent without negotiation.

2. Based on the received request, the MnS Producer creates the concrete intent instance (i.e., instance of intent IOC) and configure the new created intent MOI with the received intent information.

3. MnS Producer sends a response (see createMOI operation defined in TS 28.532[3]) to the MnS Consumer with attribute "objectInstance" of the created intent instance.

4. Based on the created intent instance, MnS Producer performs the feasibility check of the intent instance. MnS Producer can perform the feasibility check and get the results based on latest statistics of network or service performance metrics, historical experience (e.g. experience based feasible value range or threshold of performance gain), current operating status including network resource utilization and availability, prediction results based on network simulation system, and predefined checking rules or policies.

NOTE 1: Whether to perform the feasibility check can be determined according to the feasibility check enabling policy (e.g., enforce to perform feasibility check in any case, enforce to perform feasibility check in specific cases, not to perform feasibility check in specific cases, not to perform feasibility check in any case). And the feasibility check enabling policy can be predefined/configured in the MnS Producer or sent with the intent creation request from the MnS Consumer.

NOTE 2: No sequence restriction for above step 3 and step 4.

In case the feasibility check result is 'feasible', steps 5a-step8 are continuously executed:

5a. Based on the created intent, MnS Producer identifies the MOI for managed entities (e.g. ManagedElement, ManagedFunction) and derives one or more executable management tasks (including deployment and configuration parameters) for these managed entities, then MnS producer deploys or configures corresponding managed entities with deployment and configuration parameters to satisfy the intent instance.

6. During the execution of the intent, MnS Producer continuously monitors intent fulfilment information.

7. MnS Producer analyses and adjusts the managed entities to ensure the intent is continuously satisfied.

8. MnS Producer should notify (see notifyMOIAttributeValueChanges notification) MnS Consumer about attribute "objectInstance" of intent report instance and corresponding intent report information that contains one or any combination of `intentFulfilmentReport`, `intentConflictReport` and `intentFeasibilityCheckReport` which are required by the MnS Consumer.

In case the feasibility check result is 'infeasible', following step 5b is executed.

5b. MnS Producer notifies MnS consumer about `intentFeasibilityCheckReport` with infeasible information for the created intent instance. The reasons why the feasibility check result is infeasible (e.g., invalid intent expression, the intent conflict) and corresponding recommendations also can be included in the notification.

6.3.2a Intent feasibility check before intent fulfilment

Figure 6.3.2a-1 illustrates the procedure for checking the feasibility of an intent.

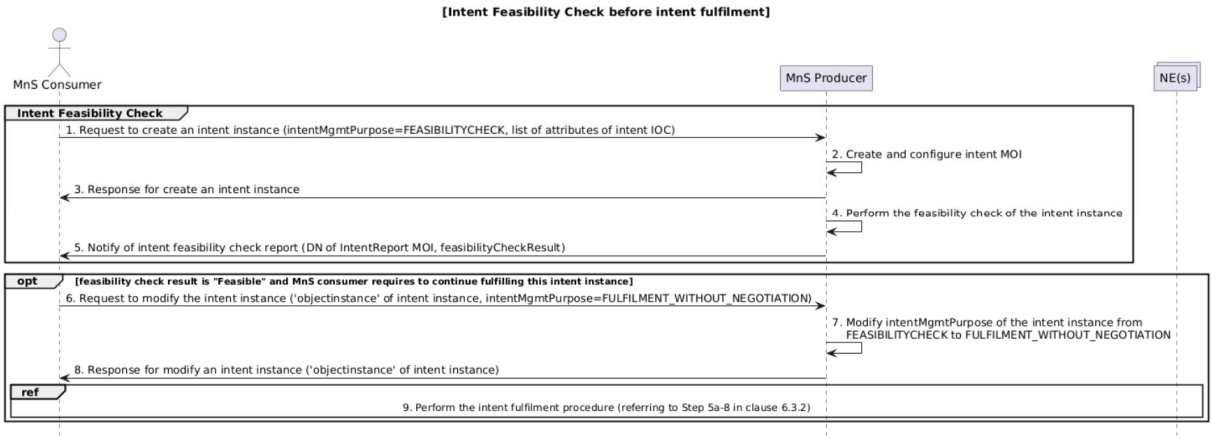


Figure 6.3.2a-1: Procedure for Intent Feasibility Check before intent fulfilment

1) MnS Consumer generates the intent information sends a request to create an Intent MOI (see createMOI operation defined in TS 28.532 [3]) to MnS Producer, providing the intent information for the intent to be created. In the provided intent information, the intentMgmtPurpose should be specified as "FEASIBILITYCHECK"

2) Based on the received request, the MnS Producer creates the concrete intent instance (i.e., instance of intent IOC) and configure it with the received intent information.

3) MnS Producer sends a response (see createMOI operation defined in TS 28.532[3]) to MnS Consumer with attribute "objectInstance" of the created intent instance.

4) MnS Producer performs an internal feasibility check as described in clause 5.3.3 to determine whether the requested intent can be achieved within the available network or service resource.

5) MnS Producer should notify (see notifyMOIAttributeValueChanges notification defined in TS 28.532[3]) Mns Consumer about attribute "objectInstance" of the intent report instance and corresponding intents report information of `intentFeasibilityCheckReport`, including feasibilityCheckResult and optional inFeasibleExpectationInfos (if the result is "inFeasible"), to inform whether the intent is feasible or not.

NOTE: The process may conclude after obtaining the feasibility check result in Step 5.

6) In case the feasibility check result is "Feasible" and MnS consumer requires to continue fulfilling this intent instance, MnS Consumer sends a request to modify the intent instance (see modifyMOIAttributes operation defined in TS 28.532 [3]) to MnS Producer, specifying the "objectInstance" of the intent MOI and setting the `intentMgmtPurpose` attribute to "FULFILMENT_WITHOUT_NEGOTIATION".

7) Based on the received request, MnS Producer modifies the intent instance, setting the value of `intentMgmtPurpose` attribute from "FEASIBILITYCHECK" to "FULFILMENT_WITHOUT_NEGOTIATION".

NOTE: This modifyMOIAttributes request will not trigger an automatic feasibility check by the MnS Producer.

8) MnS Producer sends a response (see modifyMOIAttributes operation defined in TS 28.532 [3]) to MnS consumer with the attribute "objectInstance" of the modified intent instance, confirming that the `intentMgmtPurpose` attribute has been modified to "FULFILMENT_WITHOUT_NEGOTIATION".

9) Since the intent instance has been set for fulfilment successfully, it is equivalent to creating a feasible intent for fulfilment up to this step. For the following steps about intent fulfilment, please refer to Step 5a through Step 8 of the procedure of "Create an intent" in clause 6.3.2.

6.3.3 Modify an intent

Figure 6.3.3-1 illustrates the procedure for modify an existing intent.

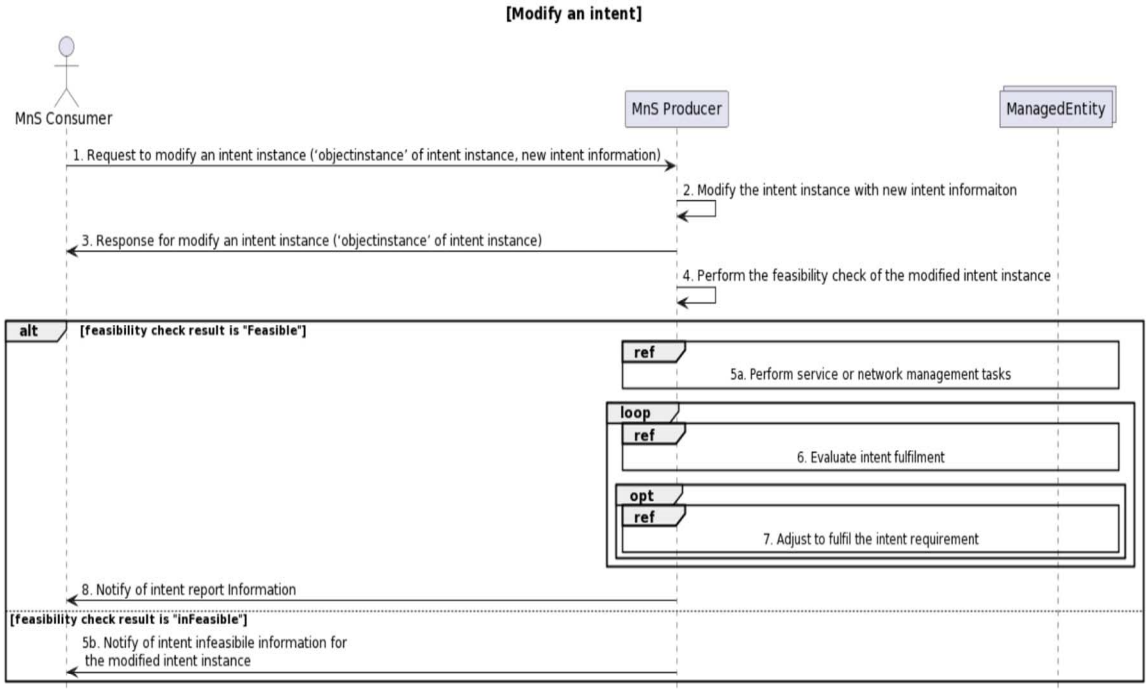


Figure 6.3.3-1: Procedure for modify an intent

1. MnS Consumer sends a request to modify an intent instance (see modifyMOIAttributes operation defined in TS 28.532 [3]) to MnS Producer with 'objectInstance' of the intent MOI and intent information to be modified. The detailed intent information see attributes (attributes which "isWritable" property is "True") of the concrete intent IOC defined in clause 6.2.

2. Based on the received request, MnS Producer modify the intent instance with received intent information which is required to be modified.

3. MnS Producer sends a response (see modifyMOIAttributes operation defined in TS 28.532 [3]) to the MnS consumer with the attribute 'objectInstance' of the modified intent instance and the intent information which is modified.

4. Based on the received request, MnS Producer performs the feasibility check of the modified intent instance. Whether to perform the feasibility check can be determined according to the feasibility check enabling policy.

In case the feasibility check result is 'feasible', following step 5a-step8 is performed:

5a. MnS Producer derives one or more executable management tasks for these managed entities, then MnS producer deploys or configures corresponding managed entities to satisfy the modified intent instance.

6. During the execution of the intent, MnS producer continuously tracks intent fulfilment information.

7. MnS producer analyses and adjusts the managed entities to ensure the intent is continuously satisfied.

8. MnS Producer should notify (see notifyMOIAttributeValueChanges notification defined in TS 28.532 [3]) MnS Consumer about attribute "objectInstance" of the intent report instance and corresponding intents report information that contains one or any combination of `intentFulfilmentReport`, `intentConflictReport` and `intentFeasibilityCheckReport`.

In case the feasibility check result is 'infeasible', following step 5b is executed:

5b. MnS Producer notifies MnS consumer about `intentFeasibilityCheckReport` with infeasible information for the modified intent instance. The reasons why the feasibility check result is infeasible (e.g. invalid intent expression, the intent conflict) and corresponding recommendations also can be included in the notification

6.3.4 Delete an intent

Figure 6.3.4-1 illustrates the procedure for deleting an existing intent.

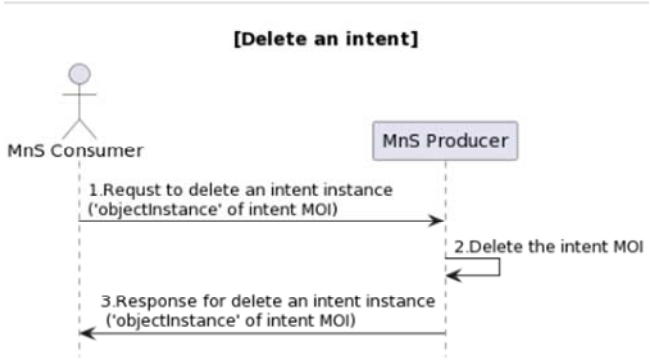


Figure 6.3.4-1: Procedure for delete an intent

1. MnS Consumer sends a request to delete an intent instance (see deleteMOI operation defined in TS 28.532[3]) to MnS Producer with 'objectInstance' of the intent MOI.
2. Based on the request, MnS Producer deletes the intent MOI.
3. MnS Producer sends response (see deleteMOI operation defined in TS 28.532 [3]) to the MnS consumer with status (OperationSucceeded or OperationFailed) and 'objectInstance' of the deleted intent MOI.

6.3.5 Query an intent

Figure 6.3.5-1 illustrates the procedure for query an intent.

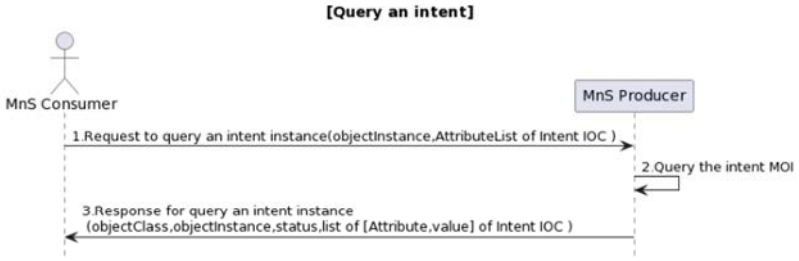


Figure 6.3.5-1: Procedure for query an intent

1. MnS Consumer sends a request to query an intent instance (see getMOIAttributes operation defined in TS 28.532[3]) to MnS Producer with 'objectInstance' of the existing intent MOI and list of attribute names of intent IOC. The list of attribute names identifies the attributes to be returned by this operation.
2. Based on the request, the MnS Producer queries the concrete intent MOI.
3. MnS Producer sends a response (see getMOIAttributes operation defined in TS 28.532[3]) to the MnS consumer with 'objectClass', 'objectInstance', and list of [Attribute,Value] which is defined in clause 6.2.

6.3.6 Intent conflict resolution

6.3.6.0 Procedure

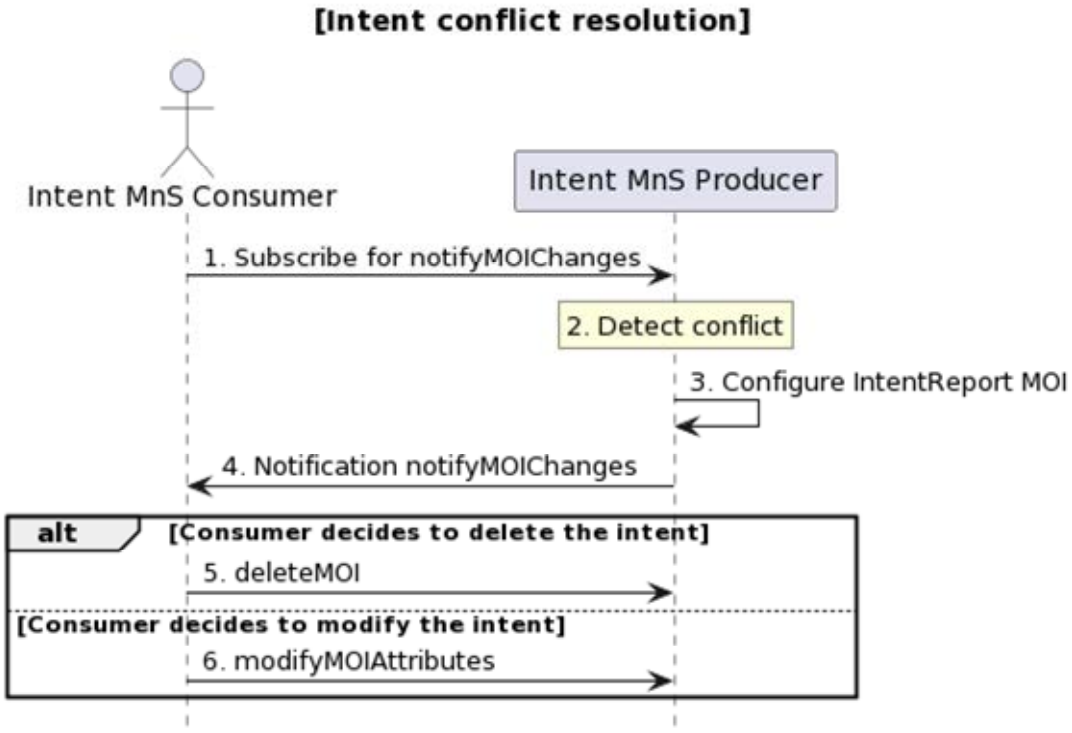


Figure 6.3.6-1: Procedure for Intent Conflict Resolution

1) MnS Consumer subscribes to receive notifyMOIChanges related to IntentReport MOI using NtfSubscriptionControl as defined in clause 4.3.22 of TS 28.622 [6].

2) MnS Producer detects the intent-related conflict(s).

3) MnS Producer configures IntentReport MOI with attributes describing the intent conflict.

4) MnS Producer notifies the modification of IntentReport MOI using notifyMOIChanges as defined in TS 28.532 [3]. The notification contains the IntentReport MOI attributes which may include information to assist MnS Consumer to resolve the conflict.

5) MnS Consumer may send a deleteMOI request to delete the intent identified by the MnS Producer in the report.

6) MnS Consumer may send a modifyMOIAttributes request to modify the Intent, Expectation and/or Target based on the information provided in the report.

6.3.6.1 Resolution of an intent conflict based on pre-emption

In the procedure for intent conflict resolution, it is necessary to recommend which intent instance is the target of intent deletion or intent modification. Typically, MnS producer chooses the intent instance based on intent priority level. Intent instance that has lower intent priority will be recommended as the target of intent deletion or intent modification. However, if two conflicting intents have the same priority, it is impossible for MnS producer to choose the intent instance that is target of intent deletion or intent modification. In this case, the intent instance that is target of intent deletion or intent modification is identified using pre-emption. Based on pre-emption, the choise is made using the attributes of " `intentPreemptionCapability`". The " `intentPreemptionCapability`" specifies whether pre-emption shall be applied. The role of MnS Consumer and MnS Producer is as follows: MnS Consumer can set intent pre-emption capability as "TRUE" for prioritized intent during pre-emption. MnS Producer decides the target of intent deletion or intent modification during pre-emption. Here, there are existing intent and new intent. When MnS Producer creates new intent, there is a conflict between the existing intent and new intent with same priority. If

`intentPreemptionCapability` ="TRUE" for existing intent, target of intent deletion or intent modification is new intent. If `intentPreemptionCapability` ="FALSE" for existing intent and `intentPreemptionCapability` ="TRUE" for new intent, target of intent deletion or intent modification will be existing intent. If both existing intent and new intent have the same value for `intentPreemptionCapability`, new intent will be target of intent deletion or intent modification.

6.3.7 Intent Report Management

6.3.7.1 Overview of Intent Report Management

Figure 6.3.7.1-1 illustrates the procedure for intent report management.

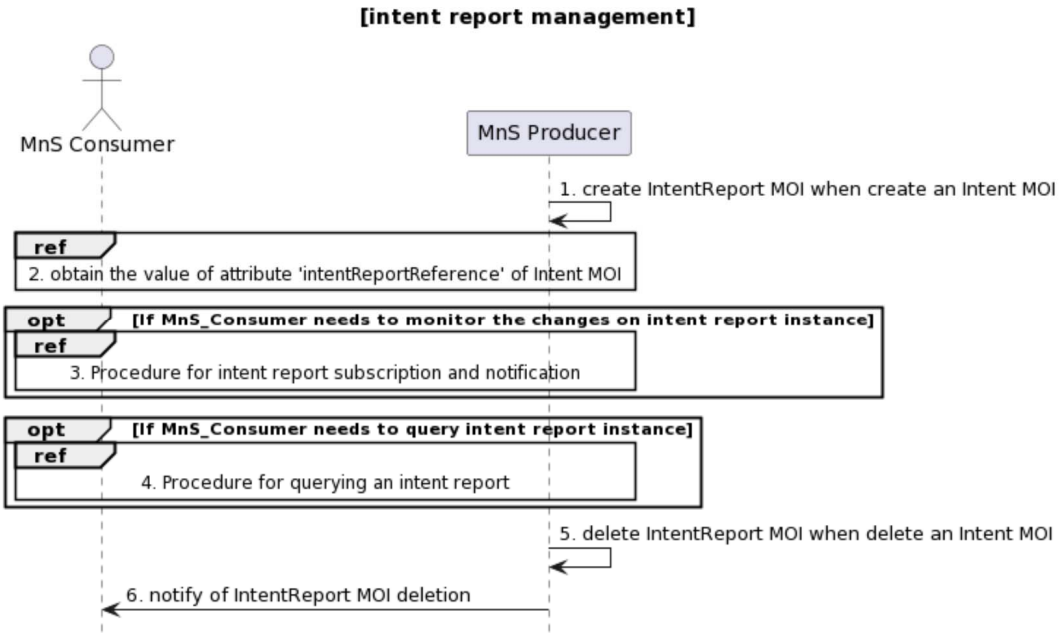


Figure 6.3.7.1-1: intent report management

1. MnS Producer creates the IntentReport MOI (i.e. instance of IntentReport IOC) when creating an Intent MOI. MnS producer also configures 'intentReportReference' of the Intent MOI. The detailed intent MOI creation see step 1 and step 2 in procedure to create an intent in clause 6.3.2.

2. MnS Consumer obtains the value of attribute 'intentReportReference' of Intent MOI. The value of attribute 'intentReportReference' of Intent MOI represents the DN of the created IntentReport MOI.

3. If MnS Consumer needs to monitor the changes on intent report instance, MnS consumer triggers the procedure for intent report subscription and notification as defined in clause E.1.2.

4. If MnS Consumer needs to query intent report instance , MnS consumer triggers the procedure for querying intent report instance as defined in clause E.1.1. This step can happen anytime until the IntentReport MOI is deleted.

5. MnS Producer deletes the IntentReportMOI when deleting an Intent MOI. The detailed intent MOI deletion see procedure for deleting an intent in clause 6.3.4.

6. MnS producer sends a NotifyMOIDeletion notification notifies the MnS consumer about deletion of IntentReport MOI.

6.3.8 Intent exploration

Figure 6.3.8-1 illustrates the procedure for intent exploration for intent negotiation during intent pre-evaluation phase.

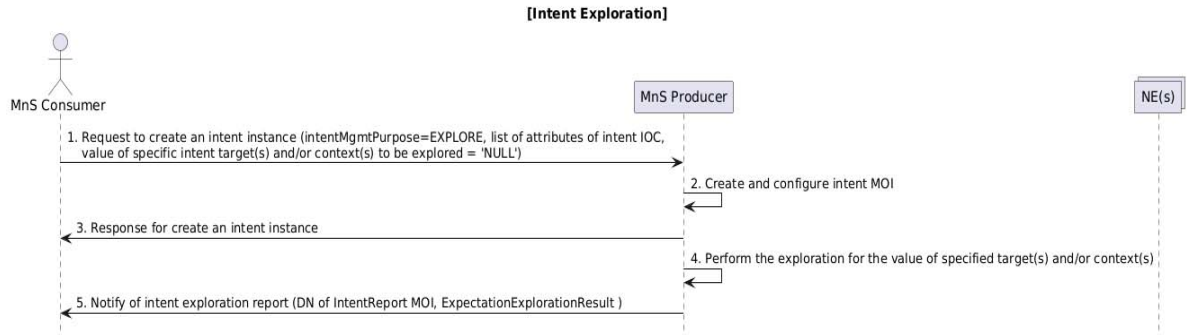


Figure 6.3.8-1: Procedure for intent exploration for intent negotiation during intent pre-evaluation phase

1. MnS Consumer generates the intent information and sends a request to create an intent instance for intent exploration (see createMOI operation defined in TS 28.532 [3]) to MnS Producer, providing intent information for the new intent to be created. The detailed intent information sees attributes (attribute which "isWritable" property is "True") of the concrete intent IOC defined in clause 6.2. In the provided intent information, the intentMgmtPurpose is specified as "EXPLORATION" and the value of specific intent targets or contexts that need to be explored is initially set to 'NULL', indicating that the MnS consumer is requesting the producer to seek the possible values for these target(s) and/or context(s) and send intent exploration report.

2. Based on the received request, the MnS Producer creates the concrete intent instance (i.e., instance of intent IOC) and configure the new created intent MOI with the received intent information.

3. MnS Producer sends a response (see createMOI operation defined in TS 28.532 [3]) to the MnS Consumer with attribute "objectInstance" of the created intent instance.

4. Based on the created intent instance with intentMgmtPurpose as " EXPLORATION ", MnS Producer performs the exploration process to evaluate and determine the best possible values for the specified targets and/or contexts, considering current resource situation and capabilities of the system. The intent instance will not impact the network as the MnS producer will not take any action on the network based on the intent exploration request.

5. MnS Producer should notify (see notifyMOIAttributeValueChanges notification) MnS Consumer about attribute "objectInstance" of intent report instance and corresponding intent exploration report information that contains the best possible values (targetExplorationResults and/or contextExplorationResults) which are required by the MnS Consumer. Then, MnS consumer can determine the best value to be fulfilled based on best possible values in the intent exploration report.

NOTE: The MnS consumer may further negotiate with the MnS producer according to the intent exploration report for best value, if MnS consumer is not satisfied with the possible values.

6.3.9 Intent negotiation during fulfilment phase

Figure 6.3.9-1 illustrates the procedure for intent negotiation during fulfilment phase.

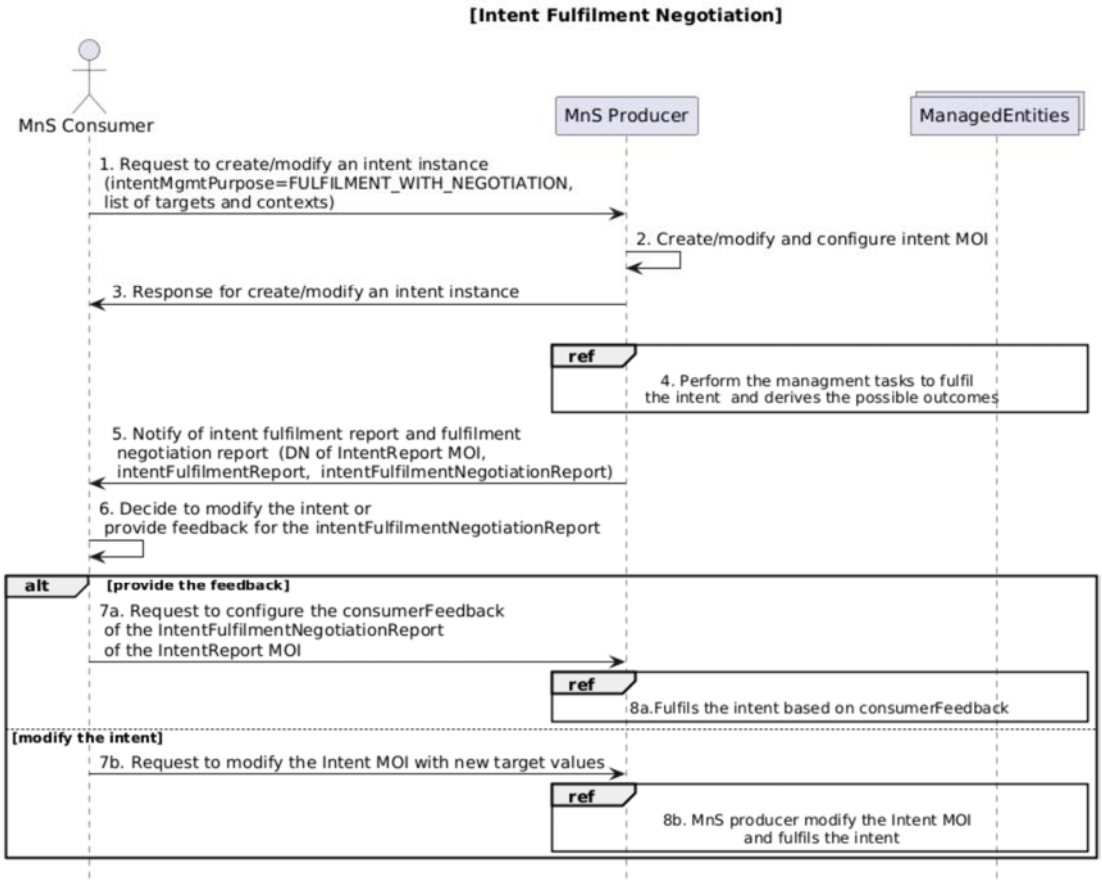


Figure 6.3.9-1: Procedure for intent negotiation during fulfilment phase

1. MnS Consumer generates the intent information and sends a request to create or modify an intent instance (see createMOI operation or modifyMOIAttributes operation defined in TS 28.532 [3]) to MnS Producer, providing intent information for the new intent to be created. The detailed intent information sees attributes (attribute which "isWritable" property is "True") of the concrete intent IOC defined in clause 6.2. The value for attribute "intentMgmtPurpose" is "FULFILMENT_WITH_NEGOTIATION".

2. Based on the received request, the MnS Producer creates the concrete intent instance (i.e., instance of intent IOC) and configures the new created intent MOI with the received intent information.

3. MnS Producer sends a response (see createMOI operation or modifyMOIAttributes operation defined in TS 28.532 [3]) to the MnS Consumer with attribute "objectInstance" of the created or modified intent instance.

4. Based on the created or modified intent, MnS Producer performs the management tasks to fulfil the intent and derives the possible outcomes.

5. MnS producer should notify (see notifyMOIAttributeValueChanges notification) MnS Consumer about attribute "objectInstance" of intent report instance and corresponding intentFulfilmentReport and intentFulfilmentNegotiationReport. Regarding the IntentReport, following are examples of different intent report information.

EXAMPLE 1: Intent is fulfilled, MnS Producer recommend better outcomes to enable MnS consumer to decide whether to change the intent content.

```
IntentReport MOI:
  - intentFulfilmentReport
     - fulfilmentStatus = Fulfilled, list of targetAchievedValue=XXX
  - intentFulfilmentNegotiationReport
     - possibleOutcomesList= [fulfilmentStatus = Fulfilled, expectationFulfilmentResults]
```

EXAMPLE 2: Intent is fulfilled, MnS producer could fulfil the with multiple ways and MnS producer provides the possible outcomes with possible impact to enable MnS consumer to select the preferred outcome.

```
IntentReport MOI:
  - intentFulfilmentReport
     - fulfilmentStatus = Fulfilled, list of targetAchievedValue=XXX
  - intentFulfilmentNegotiationReport
     - possibleOutcomesList= [fulfilmentStatus = Fulfilled, expectationFulfilmentResults,
possibleImpact (O)]
```

EXAMPLE 3: Intent is not fulfilled, MnS producer recommend fulfillable outcomes with possible impact to enable MnS consumer to decide whether to change the intent content or select the preferred outcome.

```
IntentReport MOI:
  - intentFulfilmentReport
     - fulfilmentStatus = notFulfilled, list of targetAchievedValue=XXX
  - intentFulfilmentNegotiationReport
     - possibleOutcomesList= [fulfilmentStatus = Fulfilled, expectationFulfilmentResults ,
possibleImpact(O)]
```

6. MnS consumer decides to modify the intent or provide the feedback for fulfilment based on the received intentFulfilmentNegotiationReport.

Following step 7a-8a describe the sub-procedures for providing the feedback for fulfilment based on the received intentFulfilmentNegotiationReport

7a. MnS consumer sends a request to modify an IntentReport instance (see modifyMOIAttributes operation defined in TS 28.532 [3]) to MnS Producer with 'objectInstance' of the intentReport MOI and detailed values for ConsumerFeedback. The ConsumerFeedback includes preferedIntentOutcomeId.

8a. MnS producer modify the specified IntentReport instance with detailed values for ConsumerFeedback, and perform management tasks to fulfil the intent based on the specified ConsumerFeedback.

Following step 7b-8b describe the sub-procedures for modifying the intent based on the received intentFulfilmentNegotiationReport

7b. MnS consumer sends a request to modify an Intent instance (see modifyMOIAttributes operation defined in TS 28.532 [3]) to MnS Producer with 'objectInstance' of the intent MOI and the intent information to be modified.

8b. MnS producer modifies the specified intent instance and performs management tasks to fulfil the modified intent.

7 Stage 3 definition for Intent Driven Management

7.1 RESTful HTTP-based solution set

he RESTful HTTP-based solution set for generic provisioning management service is defined in clause 12.1.1 in 3GPP TS 28.532 [3]. Corresponding className is Intent, IntentReport and IntentHandlingFunction.

Following is the SS to support intent lifecycle management based on Table 12.1.1.1.1-1 in TS 28.532 [3].

Table 7.1-1: SS to support intent lifecycle management

intent lifecycle management	IS operation	HTTP Method	Resource URI
Create an intent	createMOI operation	PUT	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id}
		POST	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}
Delete an intent	deleteMOI operation	DELETE	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id}
Modify an intent	modifyMOIAttributes operation	PUT PATCH	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id}
Query an intent	getMOIAttributes operation	GET	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id}
Activate an intent	modifyMOIAttributes operation	PUT PATCH	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id}
Deactivate an intent	modifyMOIAttributes operation	PUT PATCH	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id}

Following is the SS to support intent report management based on Table 12.1.1.1.1-1 and Table 12.1.1.2.1-1 in TS 28.532 [3].

Table 7.1-2: SS to support intent report management

intent report management	IS operation	HTTP Method	Resource URI
Query an intent report	getMOIAttributes operation	GET	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intentReport}={id}
Subscribe an intent report	createMOI operation	PUT	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{NtfSubscriptionControl }={id}
		POST	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}
Notify an intent report	notifyMOIAttributeValueChanges notification	POST	{notificationTarget}
Unsubscribe an intent report	deleteMOI operation	DELETE	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{NtfSubscriptionControl }={id}
Query an intent report subscription	getMOIAttributes operation	GET	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{NtfSubscriptionControl }={id}

NOTE: The NtfSubscriptionControl is defined in TS 28.622 [6].

Following is the SS to support intent handling capability obtaining based on Table 12.1.1.1.1-1 in TS 28.532 [3]. The NtfSubscriptionControl IOC is only used for explicit intent report subscription. The "intentReportControl" attributes in Intent IOC is used for implicit intent report subscription mechanism, see intent lifecycle management in Table 7.1-1.

Table 7.1-2A: SS to support intent handling capability obtaining

intent report management	IS operation	HTTP Method	Resource URI
Query intent handling capability	getMOIAttributes operation	GET	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intentHandlingFunction}={id}

Following is the SS to support intent negotiation management in pre-evaluation phase and fulfilment phase based on Table 12.1.1.1.1-1 in TS 28.532 [3].

Table 7.1-3: SS to support intent negotiation management

Intent negotiation	IS operation	HTTP Method	Resource URI
Intent feasibility check	createMOI operation	PUT	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id}. NOTE 1: The intentMgmtPurpose for intent resource is FEASIBILITYCHECK.
		POST	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}
Intent exploration	createMOI operation	PUT	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id} NOTE 2: The intentMgmtPurpose for intent resource is EXPLORATION.
		POST	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}
Intent fulfilment negotiation	createMOI operation	PUT	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id} NOTE 3: The intentMgmtPurpose for intent resource is FULFILMENT_WITH_NEGOTIATION.
		POST	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}
	modifyMOIAttributes operation	PUT PATCH	{MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{intent}={id} NOTE 4: The intentMgmtPurpose for intent resource is FULFILMENT_WITH_NEGOTIATION.

7.2 OpenAPI specification

7.2.1 OpenAPI document for provisioning MnS

The OpenAPI/YAML definitions for provisioning MnS are specified in 3GPP Forge, refer to clause 4.3 (OpenAPI Definitions) of TS 28.623 [17] for the Forge location. An example of Forge location is: "https://forge.3gpp.org/rep/sa5/MnS/-/tree/Tag_Rel18_SA104/".

Directory: OpenAPI

File: TS28532_ProvMnS.yaml

7.2.2 OpenAPI document for intent NRM

The OpenAPI/YAML definitions for intent NRM are specified in 3GPP Forge , refer to clause 4.3 (OpenAPI Definitions) of TS 28.623 [17] for the Forge location. An example of Forge location is: "https://forge.3gpp.org/rep/sa5/MnS/-/tree/Tag_Rel18_SA104/".

Directory: OpenAPI

File: TS28312_IntentNrm.yaml

7.2.3 OpenAPI document for scenario specific IntentExpectation

The OpenAPI/YAML definitions for scenario specific IntentExpectation are specified in 3GPP Forge, refer to clause 4.3 (OpenAPI Definitions) of TS 28.623 [17] for the Forge location. An example of Forge location is: "https://forge.3gpp.org/rep/sa5/MnS/-/tree/Tag_Rel18_SA104/".

Directory: OpenAPI

File: TS28312_IntentExpectations.yaml

8 Guidelines for using scenario specific intent expectation for intent driven use cases

This clause describes guidelines for using scenario specific intent expectation defined in clause 6.2.2 to satisfy the intent driven use cases defined in clause 5.1. Following table provides the information on which ObjectContexts and ExpectationTargets defined in clause 6.2.2 are used for the corresponding use case.

Table 8-1: Guidelines for using scenario specific intent expectation for intent driven use cases

Use case	Scenario specific IntentExpectation	ExpectationObject. ObjectContext	ExpectationTarget	Expectation Context
Intent containing an expectation for delivering radio network (clause 5.1.1)	Radio Network Expectation	- coverageAreaPolygonContext - coverageTACContext - pLMNContext - dlFrequencyContext - ulFrequencyContext - rATContext	- weakRSRPRatioTarget - lowSINRRatioTarget - aveULRANUEThptTarget - aveDLRANUEthptTarget	
Intent containing an expectation for delivering a service at the edge (clause 5.1.3)	Edge Service Support Expectation	- edgeIdentificationIdContext - edgeIdentificationLocContext - coverageAreaTAContext	- dlThptPerUETarget - ulThptPerUETarget - dLLatencyTarget - uLLatencyTarget - maxNumberofUEsTarget - activityFactorTarget - uESpeedTarget	
Intent containing an expectation on coverage performance to be assured (clause 5.1.4)	Radio Network Expectation	- coverageAreaPolygonContext - dlFrequencyContext - ulFrequencyContext - rATContext	- weakRSRPRatioTarget - lowSINRRatioTarget	
Intent containing an expectation on RAN UE throughput performance to be assured (clause 5.1.5)	Radio Network Expectation	- coverageAreaPolygonContext - dlFrequencyContext - ulFrequencyContext - rATContext - uEGroupContext	- aveULRANUEThptTarget - aveDLRANUEthptTarget - lowULRANUEThptRatioTarget - lowDLRANUEThptRatioTarget	
Intent containing an expectation for delivering 5GC network (clause 5.1.8)	5GC Network Expectation	- nfTypeContext - nfInstanceLocationContext - pLMNContext- taiContext - servingScopeContext - dnnContext	- maxNumberofPDUsessionsTarget - maxNumberofRegisteredsubscribersTarget - incomingDataTarget - outgogingDataTarget	
Intent containing an expectation on RAN capacity performance to be assured (clause 5.1.5)	Radio Network Expectation	- coverageAreaPolygonContext - dlFrequencyContext - ulFrequencyContext - rATContext	- highUlPrbLoadRatioTarget - highDlPrbLoadRatioTarget - aveUlPrbLoadTarget - aveDlPrbLoadTarget	
Intent containing an expectation on RAN energy saving (clause 5.1.7)	Radio Network Expectation	- coverageAreaPolygonContext - cellContext - pLMNContext - dlFrequencyContext - ulFrequencyContext - rATContext	- rANEnergyConsumptionTarget - rANEnergyEfficiencyTarget - aveULRANUEThptTarget - aveDLRANUEThptTarget	schedulingTimeContext
Intent containing an expectation for delivering radio service (clause 5.1.2)	Radio Service Expectation	- coverageAreaPolygonContext - dlFrequencyContext - ulFrequencyContext - serviceTypeContext - cellContext - geoCoordinateContext - uEGroupContext	- dLLatencyTarget - uLLatencyTarget - dLThptPerUETarget - uLThptPerUETarget - numberofUEsTarget	schedulingTimeContext

Intent containing an expectation on radio network traffic assurance (clause 5.1.5)	Radio Network Expectation	- cellContext - uEGroupContext	- weakRSRPRatioTarget - aveULRANUEThptTarget - aveDLRANUEThptTarget - activeUEsNumTarget	schedulingTimeContext
Intent containing an expectation for network maintenance (clause 5.1.9)	Network Maintenance Expectation	- maintenanceVersionContext - maintenanceOrderContext - maintenanceTypeContext	- maintenanceVersionTarget	maintenanceTimeContext

Annex A (informative): PlantUML source code

A.1 Procedures for intent management

A.1.1 Create an intent

```
@startuml
title "[Create an intent]"
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
Collections "ManagedEntity" as ManagedEntity
MnS_Consumer -> MnS_Producer: 1. Request to create an intent instance
(intentMgmtPurpose=FULFILMENT_WITHOUT_NEGOTIATION, list of attributes of intent IOC)
MnS_Producer -> MnS_Producer: 2. Create and configure intent MOI
MnS_Producer -> MnS_Consumer: 3. Response for create an intent instance
MnS_Producer -> MnS_Producer: 4. Perform the feasibility check of the intent instance

alt feasibility check result is "Feasible"
  Ref over MnS_Producer, ManagedEntity: 5a. Perform service or network management tasks
  loop
   Ref over MnS_Producer, ManagedEntity: 6. Evaluate intent fulfilment
     opt
  Ref over MnS_Producer, ManagedEntity: 7. Adjust to fulfil the intent requirement
  MnS_Producer -> MnS_Consumer:8. Notify of intent fulfillment Information\n (DN of intent MOI,
FulfilmentInfo)
     end
  end


else feasibility check result is "inFeasible"
  MnS_Producer -> MnS_Consumer: 5b. Notify of intent infeasibile information
end

hide footbox
@enduml
```

A.1.2 Modify an intent

```
@startuml
title "[Modify an intent]"
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
Collections "ManagedEntity" as ManagedEntity
MnS_Consumer -> MnS_Producer: 1. Request to modify an intent instance ('objectinstance' of intent
instance,new intent information)
MnS_Producer -> MnS_Producer: 2. Modify the intent instance with new intent information
MnS_Producer -> MnS_Consumer: 3. Response for modify an intent instance ('objectinstance' of intent
instance)
MnS_Producer -> MnS_Producer: 4. Perform the feasibility check of the modified intent instance

alt feasibility check result is "Feasible"
  Ref over MnS_Producer, ManagedEntity: 5a. Perform service or network management tasks
  loop
   Ref over MnS_Producer, ManagedEntity: 6. Evaluate intent fulfilment
     opt
  Ref over MnS_Producer, ManagedEntity: 7. Adjust to fulfil the intent requirement
     end
  end
  MnS_Producer -> MnS_Consumer:8. Notify of intent report Information

else feasibility check result is "inFeasible"
  MnS_Producer -> MnS_Consumer: 5b. Notify of intent infeasibile information for \n the modified
intent instance
end
hide footbox
@enduml
```

A.1.3 Delete an intent

```
@startuml
title "[Delete an intent]"
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
MnS_Consumer -> MnS_Producer: 1.Requst to delete an intent instance\n('objectInstance' of intent
MOI)
MnS_Producer -> MnS_Producer: 2.Delete the intent MOI
MnS_Producer -> MnS_Consumer: 3.Response for delete an intent instance\n ('objectInstance' of intent
MOI)
hide footbox
@enduml
```

A.1.4 Query an intent

```
@startuml
title "[Query an intent]"
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
MnS_Consumer -> MnS_Producer: 1.Request to query an intent instance(objectInstance,AttributeList of
Intent IOC )
MnS_Producer -> MnS_Producer: 2.Query the intent MOI
MnS_Producer -> MnS_Consumer: 3.Response for query an intent instance \n
(objectClass,objectInstance,status,list of [Attribute,value] of Intent IOC )
hide footbox
@enduml
```

A.1.5 Intent Report Management

A.1.5.1 Intent report management

```
@startuml
title [intent report management]
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
MnS_Producer -> MnS_Producer: 1. create IntentReport MOI when create an Intent MOI
Ref over MnS_Consumer, MnS_Producer: 2. obtain the value of attribute 'intentReportReference' of
Intent MOI

Opt If MnS_Consumer needs to monitor the changes on intent report instance
Ref over MnS_Consumer, MnS_Producer: 3. Procedure for intent report subscription and notification
End

Opt If MnS_Consumer needs to query intent report instance
Ref over MnS_Consumer, MnS_Producer: 4. Procedure for querying an intent report
End

MnS_Producer -> MnS_Producer: 5. delete IntentReport MOI when delete an Intent MOI
MnS_Producer -> MnS_Consumer: 6. notify of IntentReport MOI deletion

hide footbox
@enduml
```

A.1.5.2 Query an intent report

```
@startuml
title [Query an intent report]
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
MnS_Consumer -> MnS_Producer: 1. Request to query an intent report
instance(objectInstance,AttributeList of IntentReport IOC )
MnS_Producer -> MnS_Producer: 2. Read the intent report MOI
MnS_Producer -> MnS_Consumer: 3. Response for querying an intent report instance \n
(objectClass,objectInstance,status,list of [Attribute,value] of IntentReport IOC )
hide footbox
@enduml
```

A.1.5.3 Intent report subscription and notification

```
@startuml
title [Intent report subscription and notification]
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
Alt 'Explicit intent report subscription'
MnS_Consumer -> MnS_Producer: 1a. Requst to create NtfSubscriptionControl instance for an intent
report instance\n(including objectInstance and list of NtfSubscriptionControl IOC)
MnS_Producer -> MnS_Producer: 1b. Create a NtfSubscriptionControl MOI for the intent report instance
\n and configure the created NtfSubscriptionControl MOI
MnS_Producer -> MnS_Consumer: 1c. Reponse for creating NtfSubscriptionControl instance for an intent
report instance
else 'Implicit intent report subscription'
MnS_Consumer -> MnS_Producer: 2a.Request to configure intentReportControl of the Intent instance \n
(including objectInstance of the Intent MOI and list attributes of IntentReportControl)
MnS_Producer -> MnS_Producer: 2b.Configure the intentReportControl of Intent instance with the \n
specified vlaues for the list attributes of IntentReportControl
MnS_Producer -> MnS_Consumer: 2c. Reponse for configuring intentReportControl of the Intent instance
end
loop until the NtfSubscriptionControl MOI or Intent MOI is deleted

MnS_Producer -> MnS_Producer: 3. Configure the value of the atttributes for IntentReport MOI
MnS_Producer -> MnS_Consumer: 4. Notify intent report information \n (objectinstance,
attributeListValueChanges of IntentReport MOI)

end

hide footbox
@enduml
```

A.1.6 Intent Handling Capability Obtaining

A.1.6.1 Query intent handling capability suported by an intentHandlingFunction

```
@startuml
title [Query intent handling capability supported by an intentHandlingFunction]
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
MnS_Consumer -> MnS_Producer: 1. Request to query intent handling capabilities supported by an
IntentHandlingFunction \n (objectInstance of IntentHandlingFunction MOI,AttributeList of
IntentHandlingFunction MOI)
MnS_Producer -> MnS_Producer: 2. Read the value of the attribute \n "intentHandlingCapabilityList"
of intentHandlingFunction MOI
MnS_Producer -> MnS_Consumer: 3. Response for querying intent handling capabilities supported by an
IntentHandlingFunction\n objectInstance,list of [Attribute,value] of IntentHandlingFunction MOI )
hide footbox
@enduml
```

A.1.7 Intent conflict resolution

```
@startuml
title "[Intent conflict resolution]"
actor "Intent MnS Consumer" as Consumer
participant "Intent MnS Producer" as Producer
Consumer -> Producer: 1. Subscribe for notifyMOIChanges
rnote over Producer
 2. Detect conflict
endrnote
Producer -> Producer: 3. Configure IntentReport MOI
Producer -> Consumer: 4. Notification notifyMOIChanges
alt Consumer decides to delete the intent
   Consumer -> Producer: 5. deleteMOI
else Consumer decides to modify the intent
   Consumer -> Producer: 6. modifyMOIAttributes
end alt
hide footbox
@enduml
```

A.1.8 Intent feasibility check before intent fulfilment

```
@startuml
title "[Intent Feasibility Check before intent fulfilment]"
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
Collections "NE(s)" as ManagedEntity
Group Intent Feasibility Check
MnS_Consumer -> MnS_Producer: 1. Request to create an intent instance
(intentMgmtPurpose=FEASIBILITYCHECK, list of attributes of intent IOC)
MnS_Producer -> MnS_Producer: 2. Create and configure intent MOI
MnS_Producer -> MnS_Consumer: 3. Response for create an intent instance
MnS_Producer -> MnS_Producer: 4. Perform the feasibility check of the intent instance
MnS_Producer -> MnS_Consumer: 5. Notify of intent feasibility check report (DN of IntentReport MOI,
feasibilityCheckResult)
End

opt feasibility check result is "Feasible" and MnS consumer requires to continue fulfilling this
intent instance
MnS_Consumer -> MnS_Producer: 6. Request to modify the intent instance ('objectinstance' of intent
instance, intentMgmtPurpose=FULFILMENT_WITHOUT_NEGOTIATION)
MnS_Producer -> MnS_Producer: 7. Modify intentMgmtPurpose of the intent instance from\n
FEASIBILITYCHECK to FULFILMENT_WITHOUT_NEGOTIATION
MnS_Producer -> MnS_Consumer: 8. Response for modify an intent instance ('objectinstance' of intent
instance)
Ref over MnS_Consumer, ManagedEntity: 9. Perform the intent fulfilment procedure (referring to Step
5a-8 in clause 6.3.2)
End
hide footbox
@enduml
```

A.1.9 Intent exploration

```
@startuml
title "[Intent Exploration]"
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
Collections "NE(s)" as ManagedEntity
MnS_Consumer -> MnS_Producer: 1. Request to create an intent instance (intentMgmtPurpose=EXPLORE,
list of attributes of intent IOC, \n    value of specific intent target(s) and/or context(s) to be
explored = 'NULL')
MnS_Producer -> MnS_Producer: 2. Create and configure intent MOI
MnS_Producer -> MnS_Consumer: 3. Response for create an intent instance
MnS_Producer -> MnS_Producer: 4. Perform the exploration for the best value of specified target(s)
and/or context(s)
MnS_Producer -> MnS_Consumer: 5. Notify of intent exploration report (DN of IntentReport MOI,
ExpectationExplorationResult )
hide footbox
@enduml
```

A.1.10 Intent inegotiation during fulfilment phase

```
title "[Intent Fulfilment Negotiation]"
actor "MnS Consumer" as MnS_Consumer
participant "MnS Producer" as MnS_Producer
Collections "ManagedEntities" as ManagedEntity
MnS_Consumer -> MnS_Producer: 1. Request to create/modify an intent instance \n
(intentMgmtPurpose=FULFILMENT_WITH_NEGOTIATION, \n list of targets and contexts)
MnS_Producer -> MnS_Producer: 2. Create/modify and configure intent MOI
MnS_Producer -> MnS_Consumer: 3. Response for create/modify an intent instance
Loop For each observation period until the intentMgmtPurpose=FULFILMENT_WITH_NEGOTIATION udpated to
FULFILMENT_WITHOUT_NEGOTIATION
Ref over MnS_Producer, ManagedEntity: 4. Perform the managment tasks to fulfil \n the intent  and
derives the possible outcomes
MnS_Producer -> MnS_Consumer: 5. Notify of intent fulfilment report and fulfilment  \n negotiation
report  (DN of IntentReport MOI, \n intentFulfilmentReport,  intentFulfilmentNegotiationReport)
MnS_Consumer  ->MnS_Consumer: 6. Decide to modify the intent or \n provide feedback for the
intentFulfilmentNegotiationReport
Alt provide the feedback
MnS_Consumer  -> MnS_Producer: 7a. Request to configure the consumerFeedback \n of the
IntentFulfilmentNegotiationReport \n of the IntentReport MOI
Ref over MnS_Producer, ManagedEntity: 8a.Fulfils the intent based on consumerFeedback
```

```
Else modify the intent
MnS_Consumer  -> MnS_Producer: 7b. Request to modify the Intent MOI with new target values
Ref over MnS_Producer, ManagedEntity: 8b. MnS producer modify the Intent MOI \n and fulfils the
intent
End
hide footbox
@enduml
```

A.2 Information model definition for intent

A.2.1 Relationship UML diagram for intent (figure 6.2.1.1.1-1)

```
@startuml
hide circle
hide methods
hide members

skinparam class {
    AttributeIconSize 0
    BackgroundColor white
    BorderColor black
    ArrowColor black
}
skinparam   Shadowing false
skinparam  Monochrome true
skinparam  ClassBackgroundColor White
skinparam  NoteBackgroundColor White

class "<<ProxyClass>> \n  ManagedEntity " as ManagedEntity{}
class  "<<InformationObjectClass>>\n IntentHandlingFunction" as IntentHandlingFunction{}
class "<<InformationObjectClass>>\n Intent " as Intent {}
class "<<InformationObjectClass>>\n IntentReport" as IntentReport{}
class "<<InformationObjectClass>>\n IntentUtilityFormula" as IntentUtilityFunction{}

ManagedEntity "1" *-- "*" IntentHandlingFunction : <<names>>
IntentHandlingFunction "1" *-- "*" Intent : <<names>>
IntentHandlingFunction "1" *-- "*" IntentReport : <<names>>
IntentHandlingFunction "1" *-- "*" IntentUtilityFunction : <<names>>
IntentReport "*" <-left-> "1" Intent
Intent "1" -left-> "*" IntentUtilityFunction

note left of ManagedEntity
Represents the folllowing IOCs:
SubNetwork
end note

@enduml
```

A.2.2 Relationship UML diagram for intent (figure 6.2.1.1.1-2)

```
@startuml
hide circle
hide methods
hide members

skinparam class {
    AttributeIconSize 0
    BackgroundColor white
    BorderColor black
    ArrowColor black
}
skinparam   Shadowing false
skinparam  Monochrome true
skinparam  ClassBackgroundColor White
skinparam  NoteBackgroundColor White


class "<<InformationObjectClass>>\n Intent " as Intent{}
class "<<dataType>>\n IntentExpectation" as IntentExpectation{}
class "<<dataType>>\n Context" as IntentContext{}
class "<<dataType>>\n IntentReportControl" as IntentReportControl{}
```

```
class "<<dataType>>\n ExpectationObject" as ExpectationObject{}
class "<<dataType>>\n Context" as ObjectContext{}
class "<<dataType>>\n ExpectationTarget" as ExpectationTarget{}
class "<<dataType>>\n Context" as TargetContext{}
class "<<dataType>>\n Context" as ExpectationContext{}
class "<<dataType>>\n IntentReportControl" as IntentReportControl{}

Intent "1" -- "1..*" IntentExpectation: intentExpectations
Intent "1" -- "*" IntentContext: intentContexts
Intent "1" -- "0..1" IntentReportControl: intentReportControl

IntentExpectation "1" -- "1..*" ExpectationTarget: expectationTargets
IntentExpectation "1" -- "1" ExpectationObject: expectationObject
IntentExpectation "1" -- "*" ExpectationContext: expectationContexts
ExpectationObject "1" -- "*" ObjectContext: objectContexts
ExpectationTarget "1" -- "*" TargetContext: targetContexts

@enduml
```

A.2.3 Relationship UML diagram for intentReport IOC (figure 6.2.1.1.1-3)

```
@startuml
hide circle
hide methods
hide members

skinparam class {
    AttributeIconSize 0
    BackgroundColor white
    BorderColor black
    ArrowColor black
}
skinparam   Shadowing false
skinparam  Monochrome true
skinparam  ClassBackgroundColor White
skinparam  NoteBackgroundColor White


class "<<InformationObjectClass>>\n IntentReport" as IntentReport{}
class "<<dataType>>\n IntentFulfilmentReport" as IntentFulfilmentReport{}
class "<<dataType>>\n IntentConflictReport" as IntentConflictReport{}
class "<<dataType>>\n IntentFeasibilityCheckReport" as IntentFeasibilityCheckReport{}
class "<<dataType>>\n ExpectationFulfilmentResult" as ExpectationFulfilmentResult{}
class "<<dataType>>\n TargetFulfilmentResult" as TargetFulfilmentResult{}
class "<<dataType>>\n IntentExplorationReport" as IntentExplorationReport{}
class "<<dataType>>\n IntentFulfilmentNegotiationReport" as IntentFulfilmentNegotiationReport{}
class "<<dataType>>\n IntentUtilityReport" as IntentUtilityReport{}

IntentReport "1" -- "1" IntentFulfilmentReport: intentFulfilmentReport
IntentReport "1" -- "*" IntentConflictReport: intentConflictReports
IntentReport "1" -- "1" IntentFeasibilityCheckReport: intentFeasibilityCheckReport
IntentReport "1" -- "1" IntentExplorationReport: intentExplorationReport
IntentReport "1" -- "1" IntentFulfilmentNegotiationReport: intentFulfilmentNegotiationReport
IntentReport "1" -- "*" IntentUtilityReport: intentUtilityReport
IntentFulfilmentReport "1" -- "1..*" ExpectationFulfilmentResult: expectationFulfilmentResults
ExpectationFulfilmentResult "1" -- "1..*" TargetFulfilmentResult: targetFulfilmentResults

@enduml
```

A.2.4 Relationship UML diagram for Inheritance UML diagram for intent driven management (figure 6.2.1.1.2-1)

```
@startuml
hide circle
hide methods
hide members

skinparam class {
    AttributeIconSize 0
    BackgroundColor white
    BorderColor black
    ArrowColor black
```

```
}
skinparam   Shadowing false
skinparam  Monochrome true
skinparam  ClassBackgroundColor White
skinparam  NoteBackgroundColor White

class "<<InformationObjectClass>> \n  Top " as TOP{}
class "<<InformationObjectClass>>\n Intent " as Intent{}
class "<<InformationObjectClass>>\n IntentReport " as IntentReport{}
class "<<InformationObjectClass>>\n IntentHandlingFunction " as IntentHandlingFunction{}
class "<<InformationObjectClass>>\n IntentUtilityFormula " as IntentUtilityFunction{}
TOP  <|--  Intent
TOP  <|--  IntentReport
TOP  <|--  IntentHandlingFunction
TOP  <|--  IntentUtilityFunction

@enduml
```

A.3 UML

The following is the PlantUML used to produce the diagrams in this annex.

```
@startuml
hide circle
hide methods
hide members

skinparam class {
    AttributeIconSize 0
    BackgroundColor white
    BorderColor black
    ArrowColor black
}
skinparam  Shadowing false
skinparam  Monochrome true
skinparam  ClassBackgroundColor White
skinparam  NoteBackgroundColor White

class "SubNetwork.1" as ManagedEntity{}

class "IntentHandlingFunction.1\n\nsupportedUtilityList:{\n{utilityFunctionId: "utilFunc1",\n
utilityDescription: "RAN DL capacity prioritization function.",\nutilityParameterList:
{{parameterName: aveDLRANUEthptTarget, parameterWeight: },\n{parameterName:
lowDLRANUEThptRatioTarget, parameterWeight: },\n,{parameterName: aveDlPrbLoadTarget,
parameterWeight: }}\n},\n{utilityFunctionId: "utilFunc2",\n utilityDescription: "RAN UL performance
prioritization function.",\nutilityParameterList:{{parameterName: aveULRANUEthptTarget,
parameterWeight: },\n{parameterName: lowULRANUEThptRatioTarget, parameterWeight: }}}}\n}\n}" as
IntentHandlingFunction.1{}

class "IntentHandlingFunction.2\n\nsupportedUtilityList:{\n{utilityFunctionId: "utilFunc3",\n
utilityDescription: "RAN throughput optimization function."\nutilityParameterList: {{parameterName:
vendorDefinedParm1, parameterWeight: },\n{parameterName: vendorDefinedParm2, parameterWeight:
},\n,{parameterName: vendorDefinedParm1, parameterWeight: }}}}" as IntentHandlingFunction.2{}

class "Intent.1\n" as Intent.1 {}
class "Intent.2\n" as Intent.2 {}
class "IntentReport.1\n\nintentUtilityReport: {utilityFunction: utilFunc1, utilityResult: 80}" as
IntentReport.1{}
class "IntentReport.2\n" as IntentReport.2{}

class "IntentUtilityFormula.1\n\nutilityFunctionId: "utilFunc1"\nutilityParameterList:
{{parameterName: aveDLRANUEthptTarget, parameterWeight: 2},\n{parameterName:
lowDLRANUEThptRatioTarget, parameterWeight: 2},\n{parameterName: aveDlPrbLoadTarget,parameterWeight:
6}}, \nutilityScale: 0.5,\nutilityOffset: 0" as IntentUtilityFormula.1{}
class "IntentUtilityFormula.2\n" as IntentUtilityFormula.2{}


ManagedEntity "1" *-- "*" IntentHandlingFunction.1 : <<names>>
ManagedEntity "IntentHandlingFunction.1" *-- "*" IntentHandlingFunction.2 : <<names>>

IntentHandlingFunction.1 "1" *-- "*" Intent.1 : <<names>>
IntentHandlingFunction.1 "1" *-- "1..*" IntentReport.1 : <<names>>
IntentHandlingFunction.1  "1" *-- "1..*" IntentUtilityFormula.1 : An instance of the supported
function 'utilFunc1' and its parameters.
Intent.1 --> "\n" IntentUtilityFormula.1 : +intentUtilityFunctionRef
```

```
IntentReport.1 <.. "\n" IntentUtilityFormula.1 : Report result:\nintentUtilityReport.
IntentReport.1 <-l- "\n" Intent.1

@enduml
```

Annex B (informative): Intent Life Cycle Management

B.1 Intent Life Cycle Management

As the MnS producer's (i.e. 3gpp system) capabilities (e.g. number and/or availability of the system resources) can change even after the Intent is accepted by the MnS producer, the Intent content (i.e. a list of Intent Expectations) might not be best aligned with the MnS producer' capabilities**.** For example, the resources in MnS producer are overbooked, and the intent content is failing to meet expectations of the MnS consumer or the resources of the MnS producer become underbooked which makes such a solution very expensive and therefore useless. Hence the creation/adjustment of an Intent content (i.e. a list of Intent Expectations) and keeping it aligned with the MnS producer's capabilities, can be automated.

This means that the life cycle of the Intent can begin before Intent content is retrieved by the MnS producer, e.g. the Intent content is being defined by an MnS consumer based on requirements towards a MnS producer (e.g. to deliver a service with certain characteristics), then be optimized based on the MnS producer's capabilities (e.g. availability of MnS Producer resources in certain area, time, etc.), then be refined if the initially captured requirement needs further elaboration, etc.

The intent lifecycle consists of the following phases.

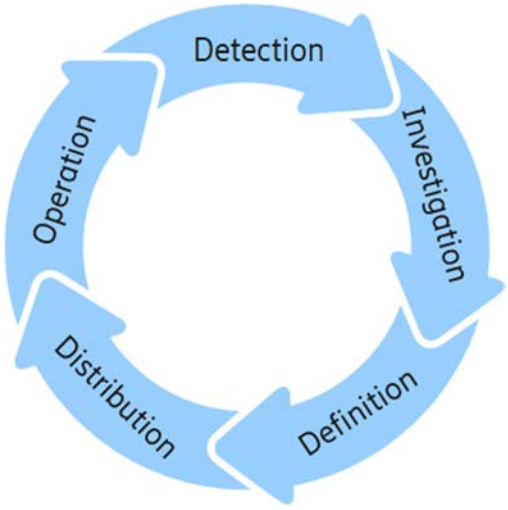


Figure B.1-1: Intent Lifecycle Phases

Detection:

In the detection phase, the MnS Consumer as the system generating the intent content (a list of expectations), identifies if there is a need to define new or change/remove existing intent expectations to set requirements, goals, and constraints. The MnS Consumer has its own terminal expectations to fulfill. It would break its terminal expectations down into a suitable set of detailed instrumental expectations. Typically, these instrumental expectations need to be fulfilled by other management functions and domains and therefore they need to be not only defined but distributed to suitable MnS producer. In the detection phase, the MnS consumer can react to changes in its own terminal expectations or to changes in the fulfillment in its instrumental expectations. In this respect the MnS consumer deriving the expectations will need to collect information about the expectation' fulfillment. Intent reports coming from MnS producer, as a system to receive intent expectations are one source for this information. Through intent reports the MnS Consumer is able to react on intent handling outcomes provided by the MnS producer. In any case it is task of the MnS consumer to assure the fulfillment of its terminal expectations and the first step is to detect if any changes are needed in its instrumental expectations.

Investigation:

In the investigation phase, the MnS Consumer finds out what intent content (a list of expectations) are feasible before expressing the intent expectations to be fulfilled by the MnS producer. This has two aspects:

- Intent handling capability obtaining. The MnS consumer needs to find the right MnS producer who has the necessary domain responsibilities and supports the intent expectations that the MnS consumer wants to define.

- Intent pre-evaluation. The MnS consumer needs to find out whether the wanted intent expectations are realistic (i.e. suitable value of the wanted intent expectations and whether these wanted intent expectations are feasible). This means, if the MnS producer would be able to successfully fulfil the wanted expectations. Typically, the feasibility of intent expectations is determined through a guided negotiation process between the MnS producer and MnS consumer. The MnS consumer can explore what the results of the wanted intent expectations would be and what is the best result the MnS producer can achieve.

Definition:

At the end of the investigation phase the MnS consumer knows what is possible and what the MnS producer to be involved. By combining this information with the needs that were identified in detection, the MnS consumer can now decide and plan all needed intent expectations. In the definition phase the MnS consumer formulates the intent expectations it needs to use.

Distribution:

In the distribution phase the MnS consumer contacts a MnS producer in order to create a new intent object or modify or change an existing one to include the intent expectations derived in the Definition phase. This way the MnS consumer acts on the plan it has made in definition phase. In this phase a MnS producer starts handling the intent expectations by receiving them and included in the intent object. The MnS producer decides if it can accept the intent expectations. If not, it would send a report with the rejection reason back to the MnS consumer. While this finishes the lifecycle of this particular intent, the MnS consumer can start over with detection to create a new plan. If the MnS producer accepts the intent, it starts operating based on it.

Operation:

Each intent expectations yet another set of requirements, goals and constraints to be considered for decisions and actions by the MnS producers. The MnS producers operate their domains of responsibility according to the given intent expectations. They also report back to the MnS consumer about status and success while continuously reacting to intent fulfillment threats. Intent reports would be evaluated by the MnS consumer as part of its detection process, which leads to the next iteration of the intent life cycle.

Annex C (informative): Mapping the 3GPP and the TM Forum intent expectation and intent report 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. Expectations are defined in the Intent Common Model (ICM) Intent Expression TR290A [20] and the reporting is defined in the Intent Common Model Intent Reporting TR290B [18].

Table C.1 illustrates the mapping between information elements defined in 3GPP IntentExpectation and TM Forum ICM Intent.

Table C.1. Mapping between 3GPP IntentExpectation and TM Forum ICM Intent

3GPP Generic Intent Information Model - IntentExpectation	TM Forum ICM IntentExpression (TR290A [20])
Information element	**Information element**
`expectationObject`	`icm:target`
`expectationTargets`	properties of `icm:Expectation` class instance
`expectationContexts`	`icm:context`

The TM Forum defines the structure of an intent report as a list of ExpectationReport corresponding to each expectation.

Table C.2 illustrates the mapping between information elements defined in 3GPP IntentReport and TM Forum ICM IntentReport.

Table C.2. Mapping between 3GPP IntentReport and TM Forum ICM IntentReport

3GPP Generic Intent Information Model - Intent Report	TM Forum ICM - Intent Report (TR290B [18])
Information element	**Information element**
`intentFulfilmentReport`, `intentConflictReport`, `intentFeasibilityCheckReport`, `intentExplorationReport`, `intentFulfilmentNegotiationReport`, `intentUtilityReport`	properties of `icm:ExpectationReport` class instance (see NOTE)
NOTE: Content of intent reports is determined by the subclass of `icm:ReportingExpectation` used to express an intent report request.	

Annex D (informative): YAML document examples for scenario specific intent instance

D.0 Introduce

This clause provides the YAML document examples for scenario specific intent instance according to the OpenAPI specification defined in clause 7.2. Following examples do not include the complete attributes defined in the OpenAPI specification.

D.1 YAML document example for Intent containing an expectation for delivering radio network

```
Intent:
  id: 'Intent_1'
  userLabel: 'Radio_Network_Deliver'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Deliver'
      expectationObjects:
        - objectType: 'RAN_SubNetwork'
          objectContexts:
            - contextAttribute: 'CoverageAreaPolygon'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - convexGeoPolygon:
                  - latitude: '31.2696'
                    longitude: '121.6322'
                  - latitude: '31.2668'
                    longitude: '121.6323'
                  - latitude: '31.2669'
                    longitude: '121.6412'
                  - latitude: '31.2696'
                    longitude: '121.6410'
            - contextAttribute: 'PLMN'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - '46000'
            - contextAttribute: 'CoverageAreaTA'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - '4457507'
            - contextAttribute: 'DlFrequency'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - arfcn: '384000'
                 - freqband: 'n39'
            - contextAttribute: 'UlFrequency'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - arfcn: '380000'
                 - freqband: 'n2'
            - contextAttribute: 'RAT'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                  - 'NR'
      expectationTargets:
        - targetName: 'WeakRSRPRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '10'
          targetContexts:
            - contextAttribute: 'WeakRSRPThreshold'
              contextCondition: 'IS_LESS_THAN'
              contextValueRange: '-130.00'
        - targetName: 'LowSINRRatio'
          targetCondition: 'IS_LESS_THAN'
```

```
          targetValueRange: '5'
          targetContexts:
            - contextAttribute: 'LowSINRThreshold'
              contextCondition: 'IS_LESS_THAN'
              contextValueRange: '-20'
        - targetName: 'AveULRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '100'
        - targetName: 'AveDLRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '300'
  intentPriority: '1'
  observationPeriod: '60'
  intentReportReferece: 'IntentReport_1'
```

D.2 YAML document example for Intent containing an expectation for delivering a service

```
Intent:
  id: 'Intent_2'
  userLabel: 'Edge_Service_Deliver'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Deliver'
      expectationObjects:
        - objectType:  'Edge_Service_Support'
          objectContexts:
            - contextAttribute: 'EdgeIdentificationId'
              contextCondition: 'IS_EQUAL_TO'
              contextValueRange:
                - '46000'
            - contextAttribute: 'EdgeIdentificationLoc'
              contextCondition: 'IS_EQUAL_TO'
              contextValueRange:
                - latitude: '31.2696'
                  longitude: '121.6322'
            - contextAttribute: 'CoverageAreaTA'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - '7'
      expectationTargets:
        - targetName: 'DlThptPerUE'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '30'
        - targetName: 'UlThptPerUE'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '10'
        - targetName: 'DlLatency'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '25'
        - targetName: 'ULLatency'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '15'
        - targetName: 'MaxNumberofUEs'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '40'
        - targetName: 'ActivityFactor'
          targetCondition: 'IS_EQUAL_TO'
          targetValueRange: '20'
        - targetName: 'UESpeed'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '120'
      expectationContexts:
        - contextAttribute: 'ServiceStartTime'
          contextCondition: 'IS_EQUAL_TO'
          contextValueRange: '2023-05-06 14:11:30'
        - contextAttribute: 'ServiceEndTime'
          contextCondition: 'IS_EQUAL_TO'
          contextValueRange: '2023-05-07 14:11:30'
        - contextAttribute: 'UEMobilityLevel'
          contextCondition: 'IS_EQUAL_TO'
          contextValueRange:
            - 'NOMADIC'
```

```
        - contextAttribute: 'ResourceSharingLevel'
          contextCondition: 'IS_EQUAL_TO'
          contextValueRange:
            - 'SHARED'
  intentPriority: '2'
  observationPeriod: '60'
  intentReportReferece: 'IntentReport_2'
```

D.3 YAML document example for Intent containing an expectation on coverage performance to be assured

```
Intent:
  id: 'Intent_3'
  userLabel: 'Radio_Network_Coverage_Performance_Assurance'

intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Ensure'
      expectationObjects:
        - objectInstance: 'SubNetwork_1'
        - objectContexts:
            - contextAttribute: 'CoverageAreaPolygon'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - convexGeoPolygon:
                  - latitude: '31.2696'
                    longitude: '121.6322'
                  - latitude: '31.2668'
                    longitude: '121.6323'
                  - latitude: '31.2669'
                    longitude: '121.6412'
                  - latitude: '31.2696'
                    longitude: '121.6410'
            - contextAttribute: 'DlFrequency'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - arfcn: '384000'
            - contextAttribute: 'RAT'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - 'NR'
      expectationTargets:
        - targetName: 'WeakRSRPRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '10'
          targetContexts:
            - contextAttribute: 'WeakRSRPThreshold'
              contextCondition: 'IS_LESS_THAN'
              contextValueRange: '-130.00'
        - targetName: 'LowSINRRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '5'
          targetContexts:
            - contextAttribute: 'LowSINRThreshold'
              contextCondition: 'IS_LESS_THAN'
              contextValueRange: '-20'
  intentPriority: '4'
  observationPeriod: '60'
  intentReportInference: 'IntentReport_3'
```

D.4 YAML document example for Intent containing an expectation on RAN UE throughput performance to be assured

```
Intent:
  Id: 'Intent_4'
```

```
  userLabel: 'Radio_Network_RANUEThpt_Performance_Assurance'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Ensure'
      expectationObjects:
        - objectInstance: 'SubNetwork_1'
        - objectContexts:
              contextAttribute: 'CoverageAreaPolygon'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - convexGeoPolygon:
                  - latitude: '31.2696'
                    longitude: '121.6322'
                  - latitude: '31.2668'
                    longitude: '121.6323'
                  - latitude: '31.2669'
                    longitude: '121.6412'
                  - latitude: '31.2696'
                    longitude: '121.6410'
            - contextAttribute: 'DlFrequency'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - arfcn: '384000'
            - contextAttribute: 'RAT'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - 'NR'
            - contextAttribute: 'UEGroup'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - sNSSAI: '1'
                 - fiveQIValue: '5'
                 - sNSSAI: '2'
                   fiveQIValue: '6'
      expectationTargets:
        - targetName: 'lowULRANUEThptRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '10'
          targetContexts:
            - contextAttribute: 'LowULRANUEThpt'
              contextCondition: 'IS_LESS_THAN'
              contextValueRange: '50'
        - targetName: 'lowDLRANUEThptRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '5'
          targetContexts:
            - contextAttribute: 'LowDLRANUEThpt'
              contextCondition: 'IS_LESS_THAN'
              contextValueRange: '200'
        - targetName: 'AveULRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '100'
        - targetName: 'AveDLRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '300'
  intentPriority: '5'
  observationPeriod: '60'
  intentReportRefernece: 'IntentReport_4'
```

D.5 YAML document example for Intent containing an expectation on RAN energy saving

```
Intent:
  Id: 'Intent_5'
  userLabel: 'RAN_Energy_Saving'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Ensure'
      expectationObjects:
        - objectInstance: 'SubNetwork_1'
          objectContexts:
            - contextAttribute: 'CoverageAreaPolygon'
              contextCondition: 'IS_ALL_OF'
```

```
              contextValueRange:
                - convexGeoPolygon:
                  - latitude: '31.2696'
                    longitude: '121.6322'
                  - latitude: '31.2668'
                    longitude: '121.6323'
                  - latitude: '31.2669'
                    longitude: '121.6412'
                  - latitude: '31.2696'
                    longitude: '121.6410'
            - contextAttribute: 'PLMN'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - '46000'
            - contextAttribute: 'DlFrequency'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - arfcn: '384000'
            - contextAttribute: 'RAT'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - 'NR'
                 - 'EUTRAN'
      expectationTargets:
        - targetName: 'RANEnergyConsumption'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '1000'
        - targetName: 'RANEnergyEfficiency'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '400000'
        - targetName: 'AveULRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '100'              - targetName: 'AveDLRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '300'
          targetContexts:
            - contextAttribute: 'rAT'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - NR
        - targetName: 'AveDLRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '100'
          targetContexts:
            - contextAttribute: 'rAT'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - EUTRAN
      expectationContexts:
        - contextAttribute: 'TargetAssuranceTime'
          contextCondition: 'IS_EQUAL_TO'
          contextValueRange:
             - startTime: '2023-10-27-22-00-00'
             - endTime: '2023-10-28-06-00-00'
  intentPriority: '6'
  observationPeriod: '60'
  intentReportReference: 'IntentReport_5'
```

D.6 YAML document example for Intent containing an expectation on radio network capacity performance to be assured

```
Intent:
  id: 'Intent_6'
  userLabel: 'Radio_Network_Capacity_Performance_Assurance'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Ensure'
      expectationObjects:
        - objectInstance: 'SubNetwork_1'
          objectContexts:
```

```
            - contextAttribute: 'CoverageAreaPolygon'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - convexGeoPolygon:
                  - latitude: '31.2696'
                    longitude: '121.6322'
                  - latitude: '31.2668'
                    longitude: '121.6323'
                  - latitude: '31.2669'
                    longitude: '121.6412'
                  - latitude: '31.2696'
                    longitude: '121.6410'
            - contextAttribute: 'DlFrequency'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - arfcn: '384000'
                - freqband: 'n39'
            - contextAttribute: 'RAT'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - 'NR'
      expectationTargets:
        - targetName: 'highUlPrbLoadRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '20'
          targetContexts:
            - contextAttribute: 'HighUlPrbLoad'
              contextCondition: 'IS_Greater_THAN'
              contextValueRange: '85'
        - targetName: 'highDlPrbLoadRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '10'
          targetContexts:
            - contextAttribute: 'HighUlPrbLoad'
              contextCondition: 'IS_Greater_THAN'
              contextValueRange: '85'
        - targetName: 'AveUlPrbLoad'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '85'
        - targetName: 'AveDlPrbLoad'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '90'
  intentPriority: '7'
  observationPeriod: '60'
  intentReportReference: 'IntentReport_6'
```

D.7 YAML document example for Intent containing an expectation for delivering 5GC network

```
Intent:
  Id: 'Intent_7'
  userLabel: '5GC_Network_Deliver'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Deliver'
      expectationObjects:
        - objectType: '5GC_SubNetwork'
          objectContexts:
            - contextAttribute: 'NfType'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - 'UPF'
            - contextAttribute: 'NfInstanceLocation'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - 'Beijing, China'
            - contextAttribute: 'PLMN'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - '46000'
            - contextAttribute: 'Tai'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
```

```
                - '460000x65'
    expectationTargets:
      - targetName: 'MaxNumberofPDUsessions'
        targetCondition: 'IS_LESS_THAN'
        targetValueRange: '250000'
      - targetName: 'maxNumberofRegisteredsubscribers'
        targetCondition: 'IS_LESS_THAN'
        targetValueRange: '2500'
  intentPriority: '3'
  observationPeriod: '60'
  intentReportReference: 'IntentReport_7'
```

D.8 YAML document example for Intent report instance

```
IntentReport:
  id: 'RAN_Energy_Saving_Report'
  intentFulfilmentReport:
    intentFulfilmentInfo:
      fulfilmentStatus: 'NOTFULFILLED'
      notFulfilledState: 'SUSPENDED'
      notFulfilledReasons:
        - 'Target_conflict_detected'
    expectationFulfilmentResults:
      - expectationId: '1'
        expectationFulfilmentInfo:
          fulfilmentStatus: 'NOTFULFILLED'
        targetFulfilmentResults:
          - targetName: 'RANEnergyConsumption'
            targetFulfilmentInfo:
              fulfilmentStatus: 'FULFILLED'
            targetAchievedValue: '900'
          - targetName: 'RANEnergyEfficiencyTarget'
            targetFulfilmentInfo:
              fulfilmentStatus: 'FULFILLED'
            targetAchievedValue: '410000'
          - targetName: 'AveULRANUEThpt'
            targetFulfilmentInfo:
              fulfilmentStatus: 'FULFILLED'
            targetAchievedValue: '100'
          - targetName: 'AveDLRANUEThpt'
            targetFulfilmentInfo:
              fulfilmentStatus: 'NOTFULFILLED'
            targetAchievedValue: '200'
            targetContexts:
              - contextAttribute: 'rAT'
                contextCondition: 'IS_ALL_OF'
                contextValueRange:
                  - NR
          - targetName: 'AveDLRANUEThpt'
            targetFulfilmentInfo:
              fulfilmentStatus: 'FULFILLED'
            targetAchievedValue: '200'
            targetContexts:
              - contextAttribute: 'rAT'
                contextCondition: 'IS_ALL_OF'
                contextValueRange:
                  - EUTRAN
  intentConflictReports:
    - conflictType: 'TARGET_CONFLICT'
      conflictingTarget: 'RANEnergyConsumption'
      recommendedSolutions: 'MODIFY'
    - conflictType: 'TARGET_CONFLICT'
      conflictingTarget: 'AveDLRANUEThpt'
      recommendedSolutions: 'MODIFY'
  intentFeasibilityCheckReport:
    feasibilityCheckResult: 'FEASIBLE'
  lastUpdatedTime: '2023-09-15-14-37-50'
  intentReference: 'RAN_Energy_Saving'
```

D.9 YAML document example for Intent containing an expectation for delivering radio service

```
Intent:
  Id: 'intent_8'
  userLabel: 'Radio_Service_Deliver'
  IntentExpectation:
    - expectationId: '1'
      expectationVerb: 'Deliver'
      expectationObjects:
        - objectType: 'Radio_Service'
        - objectContexts:
            - contextAttribute: 'CoverageAreaPolygon'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - convexGeoPolygon:
                  - latitude: '31.2696'
                    longitude: '121.6322'
                  - latitude: '31.2668'
                    longitude: '121.6323'
                  - latitude: '31.2669'
                    longitude: '121.6412'
                  - latitude: '31.2696'
                    longitude: '121.6410'
            - contextAttribute: 'ServiceType'
              contextCondition: 'IS_EQUAL_TO'
              contextValueRange:
                  - 'eMBB'
- contextAttribute: 'UEGroup'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - pLMNId: '46000'
                 - sNSSAI: '1'
      expectationTargets:
        - targetName: 'ULLatency'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '15'
        - targetName: 'DLLatency'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '20'
        - targetName: 'ULThptPerUE'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '100'
        - targetName: 'DLThptPerUE'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '300'
        - targetName: 'numberOfUEs'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '500'
      expectationContexts:
        - contextAttribute: 'schedulingTime'
          targetCondition: 'IS_ALL_OFF'
          targetValueRange:
            - timeWindow :
                - startTime: '2024-11-01-08-00-00'
                - endTime: '2024-11-01-20-00-00'
  intentPriority: '8'
  observationPeriod: '60'
  intentReportReference: 'IntentReport_8'
```

D.10 YAML document example for Intent containing an expectation on radio network traffic assurance

```
Intent:
  Id: 'Intent_9'
  userLabel: 'Radio_Network_Traffic_Assurance'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Ensure'
      expectationObjects:
```

```
        - objectInstance: 'SubNetwork_1'
        - objectContexts:
              contextAttribute: 'Cell'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - 'NRCell_1'
                - 'NRCell_2'
                - 'NRCell_3'
                - 'NRCell_4'
                - 'NRCell_5'
            - contextAttribute: 'UEGroup'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - sNSSAI: '1'
                 - fiveQIValue: '5'
                 - sNSSAI: '2'
                   fiveQIValue: '6'
      expectationTargets:
        - targetName: 'WeakRSRPRatio'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '10'
          targetContexts:
            - contextAttribute: 'WeakRSRPThreshold'
              contextCondition: 'IS_LESS_THAN'
              contextValueRange: '-130.00'
        - targetName: 'AveULRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '100'
        - targetName: 'AveDLRANUEThpt'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '300'
        - targetName: 'ActiveUEsNumTarget'
          targetCondition: 'IS_WITHIN_RANGE'
          targetValueRange:
            - '1000'
            - '5000'
      expectationContexts:
        - contextAttribute: 'schedulingTime'
          targetCondition: 'IS_ALL_OFF'
          targetValueRange:
            - timeWindow :
              - startTime: '2024-11-01-16-00-00'
              - endTime: '2024-11-01-20-00-00'
  intentPriority: '1'
  observationPeriod: '60'
  intentReportRefernece: 'IntentReport_4'
```

D.11 YAML document example for Intent containing an expectation for network maintenance

```
Intent:
  Id: 'Intent_Y'
  userLabel: 'Network_Maintenance'
  intentExpectation:
    - expectationId: '1'
      expectationVerb: 'Maintain'
      expectationObjects:
        - objectType: 'SubNetwork'
          objectContexts:
            - contextAttribute: 'swVersion'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - '17.4.2'
            - contextAttribute: 'NfType'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - 'UPF'
            - contextAttribute: 'NfInstanceLocation'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                 - 'Munich, Germany'
            - contextAttribute: 'PLMN'
              contextCondition: 'IS_ALL_OF'
```

```
              contextValueRange:
                 - '46000'
     expectationTargets:
       - targetName: 'maintenanceVersionTarget'
         targetCondition: 'IS_EQUAL_TO'
         targetValueRange: '19.1.1'
       - targetName: 'maxNumberofRegisteredsubscribers'
         targetCondition: 'IS_LESS_THAN'
         targetValueRange: '25000'
      expectationContexts:
       - contextAttribute: 'maintenanceTime'
         contextCondition: 'IS_EQUAL_TO'
         contextValueRange:
            - startTime: '2025-01-27-22-00-00'
            - endTime: '2025-01-28-06-00-00'
  intentPriority: '3'
  observationPeriod: '60'
  intentReportReference: 'IntentReport_Y'
```

D.12 YAML document examples for intent and intent report instance for intent feasibility check

YAML document example for intent which is used for feasibility check:

```
Intent:
  id: 'Intent_1'
  userLabel: 'Radio_Service_Feasibility_Check'
  intentMgmtPurpose: 'FEASIBILITYCHECK'
  intentExpectation:
    - expectationId: '1'
      expectationObjects:
        - objectType:  'RADIO_SERVICE'
          objectContexts:
            - contextAttribute: 'CoverageAreaPolygon'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                  - convexGeoPolygon:
                    - latitude: '31.2696'
                      longitude: '121.6322'
                    - latitude: '31.2668'
                      longitude: '121.6323'
                    - latitude: '31.2669'
                      longitude: '121.6412'
                    - latitude: '31.2696'
                      longitude: '121.6410'
      expectationTargets:
        - targetName: 'DlThptPerUE'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '300'
        - targetName: 'DlLatency'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '15'
```

YAML document example for intent report for feasibility check:

```
IntentReport:
  id: 'Radio_Service_ Feasibility_Check _Report'
  intentFeasibilityCheckReport:
    feasibilityCheckResult: 'INFEASIBLE'
    inFeasibleExpectationInfos:
      - expectationId: '1'
        inFeasibleTargets: 'DlLatency'
    infeasibilityReasons: 'INTENT_CONFLICT'
  lastUpdatedTime: '2025-03-15-14-37-50'
  intentReference: 'Intent_1'
```

D.13 YAML document examples for intent and intent report instance for intent exploration

YAML document example for intent which is used for exploration:

```
Intent:
  id: 'Intent_2'
  userLabel: 'Radio_Service_Exploration'
  intentMgmtPurpose: 'EXPLORATION'
  intentExpectation:
    - expectationId: '1'
      expectationObjects:
        - objectType:  'RADIO_SERVICE'
          objectContexts:
            - contextAttribute: 'cell'
              contextCondition: 'IS_ALL_OF'
              contextValueRange:
                - 'NRCell_1'
      expectationTargets:
        - targetName: 'DlThptPerUE'
          targetCondition: 'IS_GREATER_THAN'
          targetValueRange: '300'
        - targetName: 'DlLatency'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: '15'
        - targetName: 'NumberofUEs'
          targetCondition: 'IS_LESS_THAN'
          targetValueRange: ''
```

YAML document example for intent report for exploration:

```
IntentReport:
  id: 'Radio_Service_Exploration_Report'
  intentExplorationReport:
    expectationExplorationResults:
      - expectationId: '1'
        targetExplorationResults:
        - targetName: 'NumberofUEs'
          targetCondition: 'IS_EQUAL_TO'
          targetValueRange: '5'
  lastUpdatedTime: '2025-03-15-14-37-50'
  intentReference: 'Intent_2'
```

D.14 YAML document examples for IntentHandlingCapability

```
IntentHandlingFunction:
  IntentHandlingCapabilityList:
    - intentHandlingCapabilityId: 'Intent_Handling_Capability_1'
      supportedExpectationObjectType: 'RADIO_SERVICE'
      supportedExpectationTargetInfoList:
        - supportedTargetName: 'DLLatency'
          supportedTargetCondition: 'IS_LESS_THAN'
          supportedTargetValueRange: '20'
        - supportedTargetName: 'AveDLRANUEThpt'
          supportedTargetCondition: 'IS_WITHIN_RANGE'
          supportedTargetValueRange:
            - '200'
            - '300'
    - intentHandlingCapabilityId: 'Intent_Handling_Capability_2'
      supportedExpectationObjectType: 'RADIO_SERVICE'
      supportedExpectationTargetInfoList:
        - supportedTargetName: 'ULLatency'
          supportedTargetCondition: 'IS_LESS_THAN'
          supportedTargetValueRange: '30'
        - supportedTargetName: 'AveULRANUEThpt'
          supportedTargetCondition: 'IS_WITHIN_RANGE'
```

```
          supportedTargetValueRange:
            - '300'
            - '400'
```

Annex E (informative): Intent management procedures

E.1 Basic intent report management

E.1.1 Query an intent report

Figure E.1.1-1 illustrates the procedure for querying an intent report.

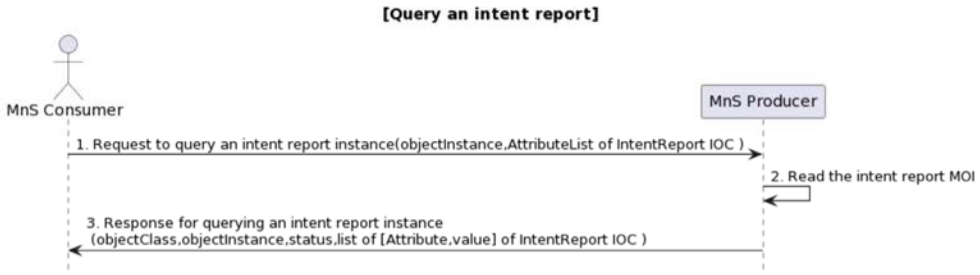


Figure E.1.1-1: Procedure for querying an intent report

1. The MnS Consumer sends a request to query an intent report instance (see getMOIAttributes operation defined in TS 28.532 [3]) to the MnS Producer with 'objectInstance' of the existing IntentReport MOI and a list of attribute names of IntentReport IOC. The 'objectInstance' of the existing IntentReport MOI is obtained from attribute 'intentReportRef' of corresponding Intent MOI.

   - If the MnS Consumer wants to obtain the Intent fulfilment information, the attribute name 'intentFulfilmentReport' needs to be specified.

   - If the MnS Consumer wants to obtain Intent conflict information, the attribute name 'intentConflictReports' needs to be specified.

   - If the MnS Consumer wants to obtain Intent fulfilment feasibility check information, the attribute name ' intentFeasibilityCheckReport' needs to be specified.

   - If the MnS Consumer wants to obtain Intent exploration information, the attribute name 'intentExplorationReport' needs to be specified.

   - If the MnS Consumer wants to obtain Intent fulfilment negotiation information, the attribute name 'intentFulfilmentNegotiationReport' needs to be specified.

   - If the MnS Consumer wants to obtain Intent utility information, the attribute name 'intentUtilityReport' needs to be specified.

2. Based on the request, the MnS Producer reads the values of required attributes in IntentReport MOI.

3. The MnS Producer sends a response (see getMOIAttributes operation defined in TS 28.532 [3]) to the MnS Consumer with 'objectInstance' of IntentReport MOI and other values of attributes in IntentReport MOI (defined in clause 6.2.1.2.2) based on a specified list of attribute names of IntentReport IOC, including at least one of the attributes 'intentFulfilmentReport', 'intentConflictReports', 'intentFeasibilityCheckReport', 'intentExplorationReport', 'intentFulfilmentNegotiationReport' and 'intentUtilityReport'.

E.1.2 Intent report subscription and notification

Figure E.1.2-1 illustrates the procedure for intent report subscription and notification.

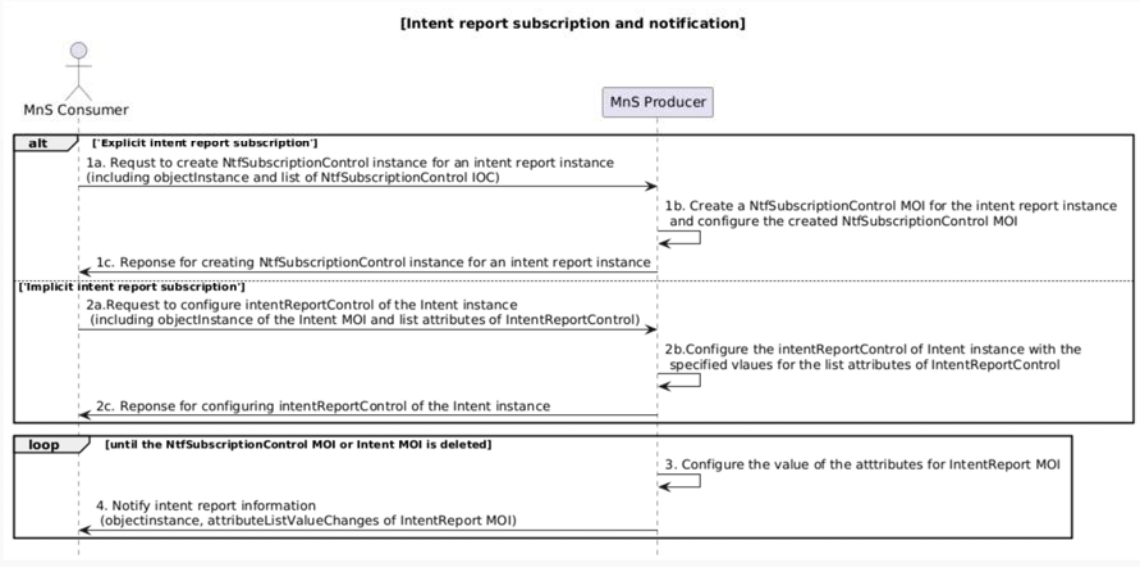


Figure E.1.2-1: Procedure for intent report subscription and notification

The following step 1a-1c describes the procedures for explicit intent report subscription.

1a. The MnS Consumer sends a request to create a NtfSubscriptionControl instance (see createMOI operation defined in TS 28.532 [3]) to the MnS Producer with intent report subscription information (see attributes of NtfSubscriptionControl IOC in TS 28.622 [6]) to establish a subscription for the attribute value change notification for IntentReport MOI. The "objectInstance" of IntentReport MOI is specified in the attribute "scope" of NtfSubscriptionControl IOC.

- If the MnS Consumer wants to obtain Intent fulfilment information, the attribute name 'intentFulfilmentReport' needs to be specified in the attribute 'notificationFilter' of NtfSubscriptionControl IOC.

- If the MnS Consumer wants to obtain Intent conflict information, the attribute name 'intentConflictReports' needs to be specified in the attribute 'notificationFilter' of NtfSubscriptionControl IOC.

- If the MnS Consumer wants to obtain Intent fulfilment feasibility check information, the attribute name 'intentFeasibilityCheckReport' needs to be specified in the attribute 'notificationFilter' of NtfSubscriptionControl IOC.

- If the MnS Consumer wants to obtain Intent exploration information, the attribute name 'intentExplorationReport' needs to be specified in the attribute 'notificationFilter' of NtfSubscriptionControl IOC.

- If the MnS Consumer wants to obtain Intent fulfilment negotiation information the attribute name 'intentFulfilmentNegotiationReport' needs to be specified in the attribute 'notificationFilter' of NtfSubscriptionControl IOC.

- If the MnS Consumer wants to obtain Intent utility information, the attribute name 'intentUtilityReport' needs to be specified in the attribute 'notificationFilter' of NtfSubscriptionControl IOC.

1b. Based on the request, the MnS Producer creates and configures NtfSubscriptionControl MOI to establish a subscription for the attribute value change notification for IntentReport MOI.

1c. The MnS Producer sends a response (see createMOI operation defined in TS 28.532 [3]) to the MnS Consumer with attribute "objectInstance" of the created NtfSubscriptionControl MOI intent report instance.

The following steps 2a-2c describes the procedures for implicit intent report subscription.

2a. The MnS Consumer sends a request to configure the IntentReportControl of the intent instance to the MnS Producer with intent report control information (see IntentReportControl <<dataType>> in clause 6.2.1.3.16) to establish a subscription for the attribute value change notification for IntentReport MOI. The request can be intent instance creation (see createMOI operation defined in TS 28.532 [3]) or modification request (see modifyMOIAttributes operation defined in TS 28.532 [3]).

2b. Based on the request, the MnS Producer configures NtfSubscriptionControl of the IntentReport MOI to establish a subscription for the attribute value change notification for IntentReport MOI. If the request is intent instance creation request, MnS producer needs to create the intent MOI before configuring the NtfSubscriptionControl of the IntentReport MOI.

2c. The MnS Producer sends a response (see createMOI operation or modifyMOIAttributes defined in TS 28.532 [3]) to the MnS Consumer with attribute "objectInstance" of the created or modified Intent MOI.

The following steps are executed until NtfSubscriptionControl MOI is deleted.

3. The MnS Producer configures the value of attributes for IntentReport MOI at the end of each observation period.

4. The MnS Producer sends a notification (see notifyMOIAttributeValueChanges notification defined in TS 28.532 [3]) to the subscribed MnS Consumer about intent report information, including at least one of the attributes 'intentFulfilmentReport', 'intentConflictReports', 'intentFeasibilityCheckReport', 'intentExplorationReport', 'intentFulfilmentNegotiationReport', and 'intentUtilityReport'.

E.2 Intent Handling Capability obtaining

E.2.1 Query intent handling capability provided by an intentHandlingFunction

Figure 6.3.7.1-1 illustrates the procedure for querying intent handling capability provided by an intentHandlingFunction.

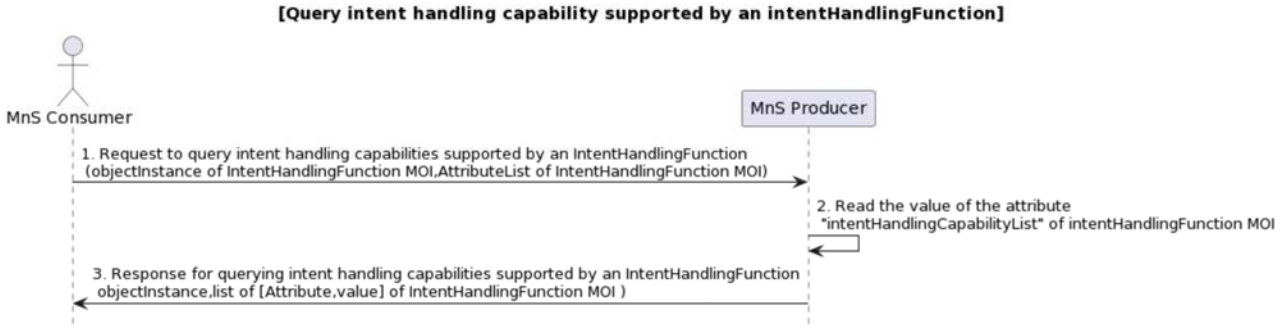


Figure E.2.1-1: Procedure for querying intent handling capabilities supported by an intentHandlingFunction

1. MnS Consumer sends a request to query intent handling capabilities supported by an intentHandlingFunction (see getMOIAttributes operation defined in TS 28.532 [3]) to MnS Producer with 'objectInstance' of the existing IntentHandlingFunction MOI and a list of attribute names of IntentHandlingFunction (including attribute 'intentHandlingCapabilityList', see clause 6.2.1.2.3).

2. Based on the request, the MnS Producer reads the values of required attribute 'intentHandlingCapabilityList' in IntentHandlingFunction MOI.

3. MnS Producer sends a response (see getMOIAttributes operation defined in TS 28.532 [3]) to the MnS Consumer with 'objectInstance' of IntentHandlingFunction MOI and the value of attribute 'intentHandlingCapabilityList' (defined in clause 6.2.1.2.3).

Annex F (informative): Potential deployment scenarios for intent interface

F.1 Description

Clause 4.1.2 describes different kinds of intents which can be applicable for different kinds of standardized reference interfaces based on roles related to 5G networks and network slicing management, see Figure 4.1.2-1.

The present clause provides potential deployment scenarios for intent interface for the management of 3GPP network and services.

F.2 Potential deployment scenario#1

In this deployment scenario, 3GPP intent driven MnS (including Management operation for Intent defined in clause 6.1 and generic Information model definition in clause 6.2.1) can be applicable for following kinds of standardized reference interfaces for the management of 3GPP network and services:

- Management interactions for Intent-NOP between NOP and NEP;
- Management interactions for Intent-CSP between CSP and NOP;
- Management interactions for Intent-CSC between CSC and CSP.

The scenario specific IntentExpectation definition can be different for above kinds of standardized reference interfaces. For example, the RadioNetworkExpectation can be used for Management interactions for Intent-NOP between NOP and NEP, while the EdgeServiceSupportExpectation can be based for Management interactions for Intent-CSP between CSP and NOP.

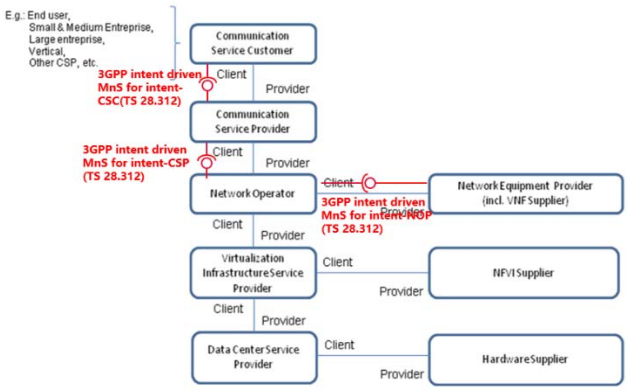


Figure F.2-1: Potential intent interface deployment scenario#1

F.3 Potential deployment scenario#2

In this deployment scenario, 3GPP intent driven MnS can be applicable for following kinds of standardized reference interfaces for the management of 3GPP network and services:

- Management interactions for Intent-NOP between NOP and NEP;
- Management interactions for Intent-CSP between CSP and NOP.

The TM Forum intent management API [7] can be applicable for following kinds of standardized reference interfaces for the management of 3GPP network and services:

- Management interactions for Intent-CSC between CSC and CSP.

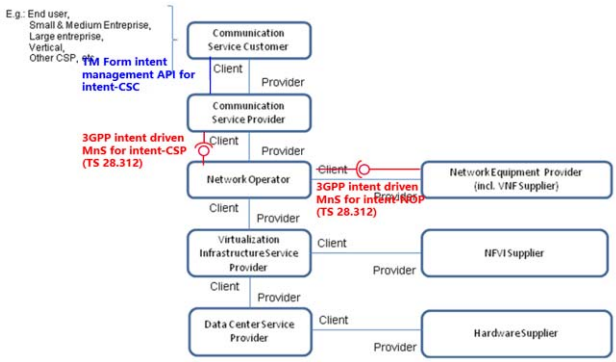


Figure F.3-1: Potential intent interface deployment scenario#2

In this intent interface deployment scenario, following contents can be used as guidelines for transformation functionality between the TM Forum intent management API for intent-CSC and the 3GPP intent driven MnS for intent-CSP.

- Mapping the 3GPP and the TM Forum intentExpectation Models described in Annex C.

Annex G (informative): Guidelines for using Intent generic information model to support new scenario which is not standardized

The intent driven MnS solution is a simplified and flexible interface, which not only allows for scenario specific intent defined in the present document, but also allows for intent extensions to support any new specific scenarios which are not standardized. So, it is important to describe guidelines to illustrate how to use intent generic information model defined in clause 6.2.1 to support new specific scenarios which are not standardized.

The new scenario which is not standardized (new specific scenario) can be satisfied by following intent model:

- Standardized Intent Expectation with vendor specific combination of standardized ExpectationObject(s) and ExpectationTarget(s). The standardized combination of standardized ExpectationObject and ExpectationTarget is documented in clause 8 Guidelines for using scenario specific intent expectation for intent driven use cases.

- Standardized Intent Expectation with standardized ExpectationObject(s) and ExpectationTarget(s) and additional vendor defined ExpectationObject(s) and/or ExpectationTarget(s).

- Vendor defined Intent Expectation by utilizing generic IntentExpectation <<dataType>> defined in clause 6.2.1.3.1. The Vendor defined Intent Expectation includes vendor extension ExpectationObject(s) and/or ExpectationTarget(s).

The vendor defined extension ExpectationObject is defined by utilizing the generic ExpectationObject <<dataType>> in clause 6.2.1.3.2. The vendor defined ExpectationTarget is defined by utilizing ExpectationTarget <<dataType>> in clause 6.2.1.3.3.

Annex H:
Example Utility Function Usage

H.1 Overview

The following shows how an MnS Producer can expose its supported Intent Utility Functions via intent capabilities, and how the MnS Consumer can then use such information to providions that function to be used.

A vendor defined Intent Utility Function identified as 'utilityFunctionId = utilFunction1' is used in the example. For brevity, only applicable IOCS, datatypes and parameters are shown, with simplified values.

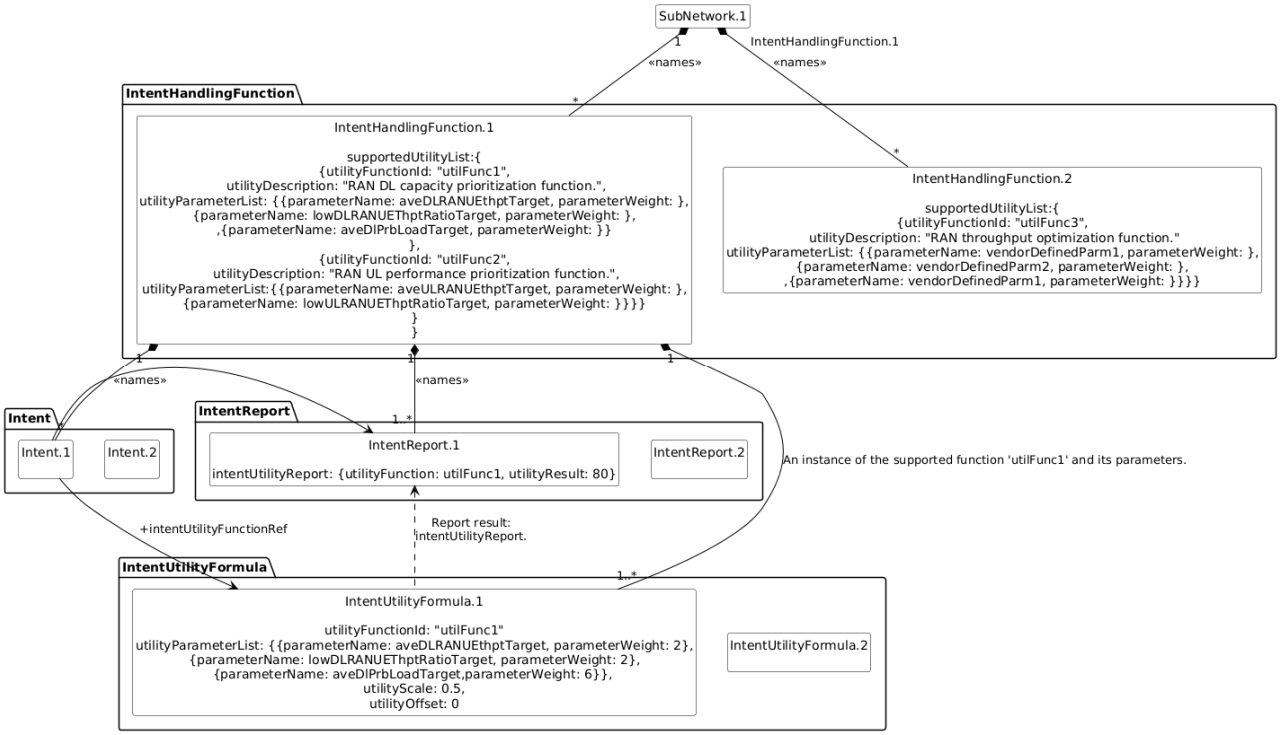


Figure H.1-1: Intent Utility Function usage example

H.2 Producer capabilities exposure

The MnS Producer exposes supported Intent Utility Functions via the supported capabilities of the Intent Handling Function.

In the above example, the `IntentHandlingFunction.1` exposes `intentHandlingCapabilityList.supportedUtilityList` = {utilFunc1, utilFunc2}.

As a result, the MnS Consumer knows the 'utilFunction1' utility function is supported by `IntentHandlingFunction.1` with further details provided in the `utilityDescription`.

H.3 Consumer utility function usage

The MnS Consumer can define specific Intent Utility Functions by provisioning instances of supported Intent Utility Functions.

In the above example, MnS consumer provisions `IntentUtilityFormula.1` with "`utilityFunctionId= utilFunc1`" and its required parameters in `utilityFunctionParameterList`.

The MnS Consumer can then provision specific Intent instances to use the provisioned Intent Utility Functions.

In the above example, MnS consumer provisions `Intent.1` with "`intentUtilityFunctionRef =IntentUtilityFormula.1`).

H.4 Producer utility function usage

MnS Producer calculates and uses the results of the utility function as required. For example, to prioritize fulfilment options or during negotiation procedures defined elsewhere in this specification.

Subject to the definitions of the utility function the determination of the relative value across multiple intent instances may require further processing (e.g. unit normalization, type conversions).

H.5 Consumer utility function result report

The MnS Producer reports the results of the utility result calculations via the Intent Report.

In the above example, MnS Producer provides the results in `IntentReport.1` via attribute `utilityFunctionReport`.

Annex I (informative): Change history

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2022-06	SA#96	SP-220491				Presented for approval	2.0.0
2022-06	SA#96					Upgrade to change control version	17.0.0
2022-06	SA#96					Editorial fixes according to EditHelp	17.0.1
2022-09	SA#97e	SP-220852	0001	-	F	Add missing guidelines for using scenario specific intent expectation for intent driven use cases	17.1.0
2022-09	SA#97e	SP-220852	0002	-	F	Correct the misalignment information in Annex C	17.1.0
2022-09	SA#97e	SP-220852	0003	-	F	Update intentNRM yaml file to distinguish the generic intent expectation part and scenario specific intent part	17.1.0
2022-09	SA#97e	SP-220852	0004	-	F	Correct procedures for intent management	17.1.0
2022-09	SA#97e					Alignment with content in FORGE	17.1.1
2022-12	SA#98e	SP-221175	0005	2	F	Correction to Context and Expectation Object definitions	17.2.0
2022-12	SA#98e	SP-221175	0006	2	F	Correction to Stage 3 and Stage 2 definitions for Intent Driven Management	17.2.0
2022-12	SA#98e	SP-221175	0007	2	F	Addition of notification clauses, correction of mis-numbered clauses and addition of common notifications	17.2.0
2022-12	SA#98e	SP-221175	0008	-	F	Add clarification for ambiguous relation description between classic MnS and intent MnS	17.2.0
2022-12	SA#98e	SP-221175	0009	-	F	Update Enum value to use upper case characters to align with TS 32.156 (Stage2 and Stage3)	17.2.0
2022-12	SA#98e	SP-221175	0010	-	F	Correct the procedure for create an intent and modify an intent	17.2.0
2022-12	SA#98e	SP-221175	0011	-	F	Add missing generic requirements for intent driven MnS	17.2.0
2022-12	SA#98e	SP-221175	0012	1	F	Correct intent class diagram	17.2.0
2022-12	SA#98e	SP-221175	0013	-	F	Correct notFulfilledReasons attribute	17.2.0
2023-03	SA#99	SP-230195	0015	1	F	Update procedures for delete an intent and query an intent	17.3.0
2023-03	SA#99	SP-230195	0016	1	F	Udpate Annex C Mapping the 3GPP and the TM Forum intentExpectation Models	17.3.0
2023-03	SA#99	SP-230195	0017	1	F	Update clause 4.2.2 Intent driven MnS	17.3.0
2023-03	SA#99	SP-230196	0018	1	F	Update stage 3 PlmnId reference	17.3.0
2023-03	SA#99	SP-230195	0019	1	F	Add clarification on clause 4.5 General concept of Intent Content	17.3.0
2023-03	SA#99	SP-230195	0020	-	F	Update clause 6.2.2.1.2.4 ExpectationContexts	17.3.0
2023-03	SA#99	SP-230195	0021	-	F	Correct the value of the defaultValue in Table 6.2.2.2-1	17.3.0
2023-03	SA#99	SP-230195	0022	1	F	Correct Context date type definition	17.3.0
2023-03	SA#99	SP-230195	0023	-	F	Change targetAttribute to targetName	17.3.0
2023-03	SA#99	SP-230195	0024	-	F	Update the figure 6.2.1.1.1-1:Relationship UML diagram for intent	17.3.0
2023-03	SA#99	SP-230196	0026	3	F	Correction to Context and Expectation object definitions Title	17.3.0
2023-03	SA#99					Correction of an implementation error	17.3.1
2023-06	SA#100	SP-230670	0044	1	F	Correct the supported qualifier for ExpectationObject and allowed value for contextCondition	17.4.0
2023-06	SA#100	SP-230670	0045	-	F	Update IntentNRM YAML file to align with stage2	17.4.0
2023-06	SA#100	SP-230670	0046	1	F	Separate YAML file for generic Information model definition and scenario specific IntentExpectation definition	17.4.0
2023-06	SA#100	SP-230670	0047	1	F	Correct the errors in Table 6.2.2.2-1	17.4.0
2023-06	SA#100	SP-230670	0058	1	F	Clarify the definition of intent expectation fulfilment	17.4.0
2023-06	SA#100	SP-230670	0066	1	F	Fixing documentation and allowed value bug in contextValueRange attribute	17.4.0
2023-06	SA#100	SP-230654	0031	1	B	Use case and Requirements on expectation for network optimization	18.0.0
2023-06	SA#100	SP-230654	0034	1	B	Clarify how generic provisioning MnS can be used for intent lifecycle management	18.0.0
2023-06	SA#100	SP-230654	0035	-	B	Add missing yaml document examples for scenario specific intent instance	18.0.0
2023-06	SA#100	SP-230654	0036	-	B	Add use case and requirements for intent driven approach for RAN energy saving	18.0.0
2023-06	SA#100	SP-230654	0037	1	B	Add use case and requirements for intent driven approach for radio network capacity optimization	18.0.0
2023-06	SA#100	SP-230654	0038	2	B	Add general concept and requirements for intent handling capability obtaining	18.0.0
2023-06	SA#100	SP-230654	0039	1	B	Add general concept and requirements for intent report	18.0.0
2023-06	SA#100	SP-230654	0043	1	B	Add new capabilities for intent driven management for Intent fulfilment feasibility checks	18.0.0
2023-06	SA#100	SP-230654	0052	1	B	Remove concept of intent validation	18.0.0
2023-06	SA#100	SP-230654	0055	2	B	Add the use cases for Intent containing an expectation for delivering 5GC subnetwork	18.0.0
2023-06	SA#100	SP-230654	0059	-	C	Clarify service support intent expectation	18.0.0
2023-06	SA#100	SP-230654	0067	2	B	Add description on monitoring the intent fulfilment information	18.0.0

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2023-09	SA#101	SP-230962	0069	1	A	Correct use case for delivering a service at the edge	18.1.0
2023-09	SA#101	SP-230962	0070	1	A	Add missing stage 3	18.1.0
2023-09	SA#101	SP-230963	0072	1	B	Add solution for intent driven approach for RAN energy saving	18.1.0
2023-09	SA#101	SP-230963	0073	1	B	Add solution for intent driven approach for intent report and intent handling capability obtaining	18.1.0
2023-09	SA#101	SP-230963	0074	-	B	Add solution for intent driven approach for radio network capacity optimization	18.1.0
2023-09	SA#101	SP-230963	0075	1	B	Add requirements and solution for intent activate and intent deactivate	18.1.0
2023-09	SA#101	SP-230963	0076	1	B	Update the RadioNetworkExpectation to support targeted scope	18.1.0
2023-09	SA#101	SP-230963	0077	-	F	Extend the allowed value for contextValueRange and targetValueRange	18.1.0
2023-09	SA#101	SP-230963	0078	-	F	Correct the description in clause 5.3.2 Intent report	18.1.0
2023-09	SA#101	SP-230963	0080	1	B	Enhance clause 6.3 Procedures for intent management	18.1.0
2023-09	SA#101	SP-230963	0083	1	B	Resolve intent conflict	18.1.0
2023-09	SA#101	SP-230963	0085	1	B	Requirements on Intent conflicts	18.1.0
2023-09	SA#101	SP-230963	0086	1	B	Support for intent priorities	18.1.0
2023-09	SA#101	SP-230963	0087	1	B	Solution for End-to-end Network Resource Expectation	18.1.0
2023-09	SA#101	SP-230962	0093	1	A	Correct issues for generic intent information model	18.1.0
2023-09	SA#101	SP-230962	0095	2	A	Correct issues for Service Support Expectation	18.1.0
2023-09	SA#101	SP-230962	0097	-	A	Update stage3 to align with stage2	18.1.0
2023-09	SA#101	SP-230962	0099	1	A	Clarification on intent translate	18.1.0
2023-09	SA#101	SP-230963	0100	1	F	5GC use case updates	18.1.0
2023-09	SA#101	SP-230963	0101	1	B	Adding new requirement for feasibility check	18.1.0
2023-09	SA#101	SP-230963	0102	1	B	Add the solution for 5GC Network Expectation	18.1.0
2023-09	SA#101	SP-230963	0111	-	F	Correct the use of ExpectationObjects	18.1.0
2023-09	SA#101	SP-230963	0112	1	B	Intent Conflict Resolution Procedure	18.1.0
2023-09	SA#101	SP-230962	0114	-	A	Editorial corrections in 3GPP TS 28.312 stage 3 OpenAPI section	18.1.0
2023-09	SA#101					Placing new clause 5.4.3 in its correct place	18.1.1
2023-12	SA#102	SP-231475	0117	2	F	Rel-18 CR TS 28.312 Update the stage3 for intent report to align with the stage2 definition	18.2.0
2023-12	SA#102	SP-231475	0120	-	F	Rel-18 CR TS 28.312 Correct issues for Relationship UML diagram for intent	18.2.0
2023-12	SA#102	SP-231475	0121	-	F	Rel-18 CR TS 28.312 Correct issues for the use of context in the OpenAPI document.	18.2.0
2023-12	SA#102	SP-231474	0123	1	A	Rel-18 CR TS28.312 Correct issues for area related attributes definition in stage2 and stage3	18.2.0
2023-12	SA#102	SP-231475	0126	-	B	Rel-18 CR TS 28.312 Update 6.1 and 7.1 to support intent report and intent handling capability obtaining	18.2.0
2023-12	SA#102	SP-231475	0127	1	B	Add YAML document examples for intent report instance	18.2.0
2023-12	SA#102	SP-231475	0128	1	B	Rel-18 CR TS 28.312 Add procedure for intent report and intent handling capability obtaining	18.2.0
2023-12	SA#102	SP-231475	0129	1	F	Rel-18 CR TS 28.312 Correct issues for IntentReport model(6.4.4.3)	18.2.0
2023-12	SA#102	SP-231475	0130	-	B	Rel-18 CR TS28.312 Enhance the RadioNetworkExpectation	18.2.0
2023-12	SA#102	SP-231475	0131	1	B	Rel-18 CR TS 28.312 Add Intent interface deployment scenarios in Annex	18.2.0
2023-12	SA#102	SP-231475	0132	1	B	Add RadioServiceExpectation	18.2.0
2023-12	SA#102	SP-231475	0133	-	F	Rel-18 CR TS 28.312 Rapporteur clean up	18.2.0
2023-12	SA#102	SP-231474	0135	-	A	Clarification on intent management procedure	18.2.0
2023-12	SA#102	SP-231475	0136	-	B	Adding 5GC expectation in Intent IOC stage 3	18.2.0
2023-12	SA#102	SP-231474	0139	2	A	Rel-18 CR TS 28.312 Update the definition of expectationId in Table 6.2.1.4-1	18.2.0
2023-12	SA#102	SP-231475	0140	1	B	Rel-18 CR TS 28.312 Selection among Expectation, targets and contexts	18.2.0
2023-12	SA#102	SP-231475	0141	1	B	Rel18 CR 28312 Requirements on intent Reporting	18.2.0
2023-12	SA#102	SP-231475	0142	3	F	Rel-18 CR 28.312 Intent Conflict Resolution Procedure	18.2.0
2023-12	SA#102	SP-231475	0143	1	F	Rel-18 CR TS 28.312 Correct issues for the reference to maxNumberofPDUsessions	18.2.0
2023-12	SA#102	SP-231475	0150	1	B	Rel-18 CR TS 28.312 add intent conflict resolution based on intent preemption	18.2.0
2023-12	SA#102	SP-231475	0152	1	C	Add one case and requirement for supporting change of intent handling function	18.2.0
2023-12	SA#102	SP-231475	0153	2	F	Clarify MnS consumer how handling infeasible results of intent fulfilment feasibility check	18.2.0
2023-12	SA#102	SP-231475	0154	1	F	Enhance the description of intent report	18.2.0
2023-12	SA#102	SP-231475	0155	1	B	Add description and potential requirement for intent report	18.2.0
2023-12	SA#102	SP-231476	0156	-	F	Correct issues for Class definition including Intent and IntentReport	18.2.0
2023-12	SA#102	SP-231476	0158	1	F	Rel-18 CR TS 28.312 Correct issues of attibutes in IntentFulfillmentReport.	18.2.0

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2023-12	SA#102	SP-231476	0159	1	F	Rel-18 CR TS 28.312 Enhancement for Intent handling capability obtaining	18.2.0
2023-12	SA#102	SP-231476	0161	-	F	Rel-18 CR TS 28.312 Correct use case for Intent containing an expectation for 5GC network	18.2.0
2023-12	SA#102	SP-231474	0166	1	A	Correct the description for Context <<dataType>>	18.2.0
2023-12	SA#102	SP-231476	0171	1	F	Rel-18 CR TS 28.312 Update RadioNetworkExpectation definition	18.2.0
2023-12	SA#102	SP-231476	0172		F	Rel-18 CR TS 28.312 Update Annex D YAML document example to align with the latest intentNRM and intentExpectationNRM YAML definition	18.2.0
2023-12	SA#102	SP-231476	0173	1	F	Rel-18 CR TS 28.312 Update ValueRangeType to support scenario specific intent expectation	18.2.0
2023-12	SA#102	SP-231476	0174	1	F	Rel-18 CR TS 28.312 Correct the stage2 and stage3 definition for intentPriority	18.2.0
2023-12	SA#102	SP-231476	0175	1	F	Rel-18 CR TS28.312 Update the stage3 to aign with the latest stage2	18.2.0
2023-12	SA#102	SP-231476	0176	1	B	Rel-18 CR TS 28.312 add recommendedSolutions value in IntentReport	18.2.0
2023-12	SA#102	SP-231476	0177	1	B	Rel-18 CR TS 28.312 add attributes of intent preemption	18.2.0
2023-12	SA#102	SP-231476	0179	1	F	Rel-18 CR 28.312 Missing attribute definitions	18.2.0
2023-12	SA#102	SP-231476	0182	1	C	Rel18_CR_28312 Clarify observation period and fulfilment deadline	18.2.0
2023-12	SA#102	SP-231476	0183	-	F	Rel-18 CR 28.312 Errors in attribute definitions	18.2.0
2023-12	SA#102	SP-231476	0184		B	Update 5GC expectation	18.2.0
2023-12	SA#102	SP-231476	0185	1	F	Rel-18 CR TS 28.312 Update the description of requirements for obtaining intent report information in use cases	18.2.0
2023-12	SA#102	SP-231476	0186	-	F	Rel-18 CR TS 28.312 Update the description about intent handling function	18.2.0
2023-12	SA#102	SP-231474	0188	-	A	Rel-18 CR TS28.312 Clarify the description of object instance	18.2.0
2023-12	SA#102	SP-231476	0190	1	F	Rel-18 CR TS 28.312 5.3.4.3 Adjust the content in clause 5.3.4.3 Resolving Intent-related conflicts	18.2.0
2023-12	SA#102	SP-231476	0191	1	F	Correct issues for End-to-end Network Resource Expectation	18.2.0
2023-12	SA#102	SP-231474	0193	1	A	Clarify verbs in attribute definitions	18.2.0
2023-12	SA#102	SP-231474	0195	1	A	Correct intent driven MnS description in clause 4.2.2	18.2.0
2023-12	SA#102	SP-231476	0196	-	F	Correct Yaml documents in Annex D	18.2.0
2023-12	SA#102	SP-231476	0198	1	F	Rel-18 CR TS 28.312 Revisions to ValueRange	18.2.0
2023-12	SA#102	SP-231476	0199	1	B	Rel-18 CR TS 28.312 Update 6.1 and 7.1 to support the operations of unsubscribing intent reporting and querying intent reporting subscriptions	18.2.0
2023-12	SA#102	SP-231474	0201	1	A	Rel-18 CR 28.312 Correct create an intent procedure	18.2.0
2023-12	SA#102					Adding code files to Zip file	18.2.1
2024-03	SA#103	SP-240186	0202	-	F	TS28.312 Rel18 correction to Schema definition Issues for SubNetwork of OpenAPI SS	18.3.0
2024-03	SA#103	SP-240170	0205	1	F	Rel-18 CR TS 28.554 Correction of Intent Conflict Resolution	18.3.0
2024-03	SA#103	SP-240170	0206	1	F	Rel-18 CR TS 28.312 Rapporteur clean up	18.3.0
2024-03	SA#103	SP-240170	0207	1	F	Rel-18 CR TS 28.312 Update the YAML document examples to align with yaml definition	18.3.0
2024-03	SA#103	SP-240170	0208	1	F	Rel-18 CR TS 28.312 Address CR implementation issue for CR0171 and CR0173	18.3.0
2024-03	SA#103	SP-240170	0210	1	F	Rel-18 CR TS 28.312 clarification about notFulfilledState condition	18.3.0
2024-03	SA#103	SP-240170	0211	1	F	Rel-18 CR TS 28.312 Update the description of clause 5.3.2 intent report	18.3.0
2024-03	SA#103	SP-240170	0212	-	F	Rel-18 CR TS 28.312 Remove optional predicted value in intent report	18.3.0
2024-03	SA#103	SP-240170	0213	1	F	Rel-18 CR TS 28.312 normative yaml code in 3gpp forge	18.3.0
2024-06	SA#104	SP-240846	0214	-	F	Rel-18 CR 28312 Fix incorrect attribute values and improve wording	18.4.0
2024-06	SA#104	SP-240846	0218	-	F	Rel-18 CR TS 28.312 Correct the misalignment issues for requirements	18.4.0
2024-06	SA#104	SP-240846	0219	-	F	Rel-18 CR TS 28.312 Rapporteur clean up	18.4.0
2024-06	SA#104	SP-240846	0220	1	F	Rel-18 CR TS 28.312 Update the stage3 to align with stage2	18.4.0
2024-06	SA#104	SP-240846	0221	1	F	Rel-18 CR TS 28.312 correction of attribute definition	18.4.0
2024-06	SA#104	SP-240846	0224	1	F	Rel-18 CR TS 28.312 Correction of Expectation Targets	18.4.0
2024-06	SA#104	SP-240846	0226	-	F	Rel-18 CR TS 28.312 Correct incorrect description for clause 5.1.5	18.4.0
2024-06	SA#104	SP-240846	0227	-	F	Rel-18 CR TS 28.312 Correction of attribute definition and incorrect description	18.4.0
2024-06	SA#104	SP-240846	0228	-	F	Rel-18 CR TS 28.312 Update the Annex D to align with stage2	18.4.0
2024-06	SA#104	SP-240803	0230	-	A	Rel-18 CR TS 28.312 correct the definition for FulfilmentInfo dataType	18.4.0

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2024-09	SA#105	SP-241186	0233	-	F	Rel-18 CR TS 28.312 Update forge link to align with endorsed S5-242202	18.5.0
2024-09	SA#105	SP-241186	0234	1	F	Rel-18 CR TS 28.312 Correct the description of FulfilmentInfo – MCC: the last 2 changes could not be implemented due to wrong baseline.	18.5.0
2024-09	SA#105	SP-241186	0235	-	F	Rel-18 CR TS 28.312 Rapp clean up	18.5.0
2024-09	SA#105	SP-241173	0238	1	F	Rel-18 CR TS 28.312 Fix editors note in clause 6.2.1.4	18.5.0
2024-09	SA#105	SP-241186	0242	1	F	Rel-18 CR TS 28.312 Fix wrong attribute	18.5.0
2024-12	SA#106	SP-241655	0247	-	F	Rel-18 CR TS 28.312 Correct issues for targetAssuranceTimeContext and pLMNContext	18.6.0
2024-12	SA#106	SP-241633	0248	1	A	Rel-18 CR TS 28.312 intent definition clarification	18.6.0
2024-12	SA#106	SP-241655	0252	1	F	Rel-18 CR TS 28.312 Fix stage 2 type definition of infeasibilityReason	18.6.0
2024-12	SA#106	SP-241633	0262	1	A	Rel-18 CR TS 28.312 Corrections on the Mapping of Intent Model between 3GPP and TMF	18.6.0
2024-12	SA#106	SP-241655	0263	1	F	Rel-18 CR TS 28.312 Update the Mapping of Intent Model between 3GPP and TMF	18.6.0
2024-12	SA#106	SP-241655	0274	1	F	Rel-18 CR 28.312 Fix mismatch between stage 2 and stage 3	18.6.0
2024-12	SA#106	SP-241639	0246	1	C	Rel-19 CR TS 28.312 Implement readonly attributes for openAPI SS	19.0.0
2024-12	SA#106	SP-241639	0251	-	F	Rel-19 CR 28.312 Enhance the isUnique property for stage 3 OpenAPI	19.0.0
2024-12	SA#106	SP-241655	0253	-	C	Rel 19 CR TS 28.312 Remove Support Qualifier from attribute constraints	19.0.0
2024-12	SA#106	SP-241639	0272	-	B	Rel-19 CR 28.312 Enhance stage 3 OpenAPI for multiplicity property in TS28312_IntentNrm	19.0.0
2025-03	SA#107	SP-250152	0277		B	Rel-19 CR TS 28.312 Update the Relationship UML Diagram for Intent. MCC: change on figure 6.2.1.1.1-2 could not be implemented due to clash with CR 0304.	19.1.0
2025-03	SA#107	SP-250152	0278		B	Rel-19 CR TS 28.312 Enhance the use case and solution to support the scenario of delivering a radio service in a scheduled time	19.1.0
2025-03	SA#107	SP-250152	0279		B	Rel-19 CR TS 28.312 Enhance the use case and solution to support radio network traffic assurance for scheduled events scenario	19.1.0
2025-03	SA#107	SP-250153	0281	1	B	Rel-19 CR TS 28.312 Update the Mapping of IntentReport model	19.1.0
2025-03	SA#107	SP-250152	0282		B	Rel-19 CR TS 28.312 Enhance the use case and solution to support RAN energy saving scenario	19.1.0
2025-03	SA#107	SP-250152	0283		B	Rel-19 CR TS 28.312 Add concept for intent negotiations	19.1.0
2025-03	SA#107	SP-250152	0284		B	Rel-19 CR TS 28.312 Update use case, requirements and solution for intent feasibility check	19.1.0
2025-03	SA#107	SP-250152	0285		B	Rel-19 CR TS 28.312 Add use case and requirements for intent exploration	19.1.0
2025-03	SA#107	SP-250152	0286		B	Rel-19 CR TS 28.312 Enhance the intent report use case, reqs and solution to support implicit intent report subscription with customized requirements	19.1.0
2025-03	SA#107	SP-250152	0287		B	Rel-19 CR TS 28.312 Add Use Case, Requirements, and Solution of Intent Degradation based on Expectation Preference	19.1.0
2025-03	SA#107	SP-250152	0289	1	B	Rel-19 CR TS 28.312 Add use case and requirements for Enablers for Intent Fulfilment	19.1.0
2025-03	SA#107	SP-250152	0290	1	B	Rel-19 CR TS 28.312 Add solution for intent exploration	19.1.0
2025-03	SA#107	SP-250152	0291	1	B	Rel-19 CR TS 28.312 Enhancement of Intent handling state management	19.1.0
2025-03	SA#107	SP-250153	0292	1	B	Rel-19 CR TS 28.312 Add guidelines for using intent generic information model to support new scenario which is not standardized	19.1.0
2025-03	SA#107	SP-250152	0293		B	Rel-19 CR TS 28.312 Intent lifecycle documentation improvement	19.1.0
2025-03	SA#107	SP-250152	0294	1	B	Rel-19 CR TS 28.312 Add procedure for intent feasibility check	19.1.0
2025-03	SA#107	SP-250152	0295		B	Rel-19 CR TS 28.312 Add YAML definition for Intent capability exposure	19.1.0
2025-03	SA#107	SP-250177	0298	1	A	Rel-19 CR TS 28.312 Address the CR implementation issue for infeasibilityReasons in CR252	19.1.0
2025-03	SA#107	SP-250152	0299	1	B	Rel-19 CR 28.312 Add generic utility function support (stage 1)	19.1.0
2025-03	SA#107	SP-250152	0300		B	Rel-19 CR TS 28.312 Add use-case and requirements for intent driven management of network maintenance	19.1.0
2025-03	SA#107	SP-250153	0301	1	B	Rel-19 CR TS 28.312 Add solution for intent driven management of network maintenance	19.1.0
2025-03	SA#107	SP-250152	0302	1	B	Rel-19 CR TS 28.312 Add description of intent negotiation	19.1.0
2025-03	SA#107	SP-250150	0304	2	A	Rel-19 CR TS 28.312 Add attribute names on association lines. MCC: change on figure 6.2.1.1.1-2 could not be implemented due to clash with CR 0277.	19.1.0

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2025-03	SA#107	SP-250152	0305		B	Rel-19 CR TS28.312 Negotiation on intent fulfillment	19.1.0
2025-03	SA#107	SP-250152	0306		B	Rel-19 CR TS28.312 extend Intent capability exposure	19.1.0
2025-06	SA#108	SP-250536	0288	3	B	Rel-19 CR TS 28.312 Enhance the use case and solution to support radio network support for UAV pre-flight preparation	19.2.0
2025-06	SA#108	SP-250536	0307	1	F	Rel-19 CR TS 28.312 Update the solution for Intent capability exposure	19.2.0
2025-06	SA#108	SP-250536	0308	1	F	Rel-19 CR TS 28.312 Update the solution for intent negotiation report	19.2.0
2025-06	SA#108	SP-250536	0309	3	B	Rel-19 CR TS 28.312 Add procedure for intent exploration	19.2.0
2025-06	SA#108	SP-250536	0310	1	B	Rel-19 CR TS 28.312 Update procedures for intent report management to support implicit intent report subscription	19.2.0
2025-06	SA#108	SP-250536	0311	1	F	Rel-19 CR TS 28.312 Update 5.3.1 Intent handling capability obtaining	19.2.0
2025-06	SA#108	SP-250536	0312	1	B	Rel-19 CR TS 28.312 Add missing abbreviation and definition	19.2.0
2025-06	SA#108	SP-250536	0313	3	F	Rel-19 CR TS 28.312 Enhance the definition for intentMgmtPurpose	19.2.0
2025-06	SA#108	SP-250536	0314	-	F	Rel-19 CR TS 28.312 Enhance the solution for RadioServiceExpectation	19.2.0
2025-06	SA#108	SP-250536	0316	1	B	Rel-19 CR TS 28.312 Enhance the use case and solution to support MOCN undifferentiated radio service in a specified area	19.2.0
2025-06	SA#108	SP-250536	0317	1	B	Rel-19 CR TS 28.312 Update clause 6.1 and 7.1 to support intent negotiation	19.2.0
2025-06	SA#108	SP-250536	0318	-	B	Rel-19 CR TS 28.312 Add Stage-3 Solution for intent driven management of network maintenance (FORGE ONLY)	19.2.0
2025-06	SA#108	SP-250536	0326	-	F	Rel-19 CR TS 28.312 Modify the Generic Information model definition	19.2.0
2025-06	SA#108	SP-250536	0329	1	F	Rel-19 CR TS 28.312 Add an example to support intent report based on condition	19.2.0
2025-06	SA#108	SP-250536	0330	-	F	Rel-19 CR TS 28.312 Correction on introduction in clause 5.1.9.1	19.2.0
2025-06	SA#108	SP-250536	0331	1	F	Rel-19 CR TS 28.312 Correction on introduction for intent fulfilment feasibility check	19.2.0
2025-06	SA#108	SP-250536	0332	1	C	Rel-19 CR 28.312 Clarifications on negotiation of intent fulfillment	19.2.0
2025-06	SA#108	SP-250536	0333	2	B	CR TS 28.312 introduce mapping table	19.2.0
2025-06	SA#108	SP-250535	0336	1	A	Rel-18 CR TS 28.312 Correct import table	19.2.0
2025-06	SA#108	SP-250536	0337	1	B	Rel-19 CR 28.312 Add generic utility function support (stage 2)	19.2.0
2025-06	SA#108	SP-250536	0338	1	B	Rel-19 CR TS 28.312 Add stage3 openAPI definition for intent negotiation report (FORGE ONLY)	19.2.0
2025-06	SA#108	SP-250537	0339	1	F	Rel-19 CR TS 28.312 Update procedure for intent feasibility check	19.2.0
2025-06	SA#108	SP-250537	0340	1	B	Rel-19 CR TS 28.312 Add procedure for intent negotiation during fulfilment phase	19.2.0
2025-06	SA#108	SP-250537	0341	1	F	Rel-19 CR TS 28.312 Address stage2 and stage3 misalignment issue for several attribute definition	19.2.0
2025-06	SA#108	SP-250537	0342	1	B	Rel-19 Update UML diagram for IntentReport	19.2.0
2025-06	SA#108	SP-250537	0343	1	F	Rel-19 CR TS 28.312 Update constraint description for IntentReport	19.2.0
2025-06	SA#108	SP-250537	0344	1	C	Rel-19 CR TS 28.312 correct TargetAchievedValue	19.2.0
2025-06	SA#108	SP-250537	0345	1	C	Rel-19 CR TS 28.312 Update intent conflict reporting	19.2.0
2025-06	SA#108	SP-250537	0346	1	F	Rel-19 CR TS 28.312 Clarification on Usage of Intent Utility Function	19.2.0
2025-06	SA#108	SP-250537	0347	1	F	Rel-19 CR TS 28.312 Clarify the usage of Intent NRM	19.2.0
2025-06	SA#108	SP-250537	0349	-	F	Rel-19 CR TS 28.312 Correct typo of Intent NRM	19.2.0
2025-06	SA#108	SP-250537	0350	1	B	Rel-19 CR TS 28.312 Add YAML document examples for Intent feasibility check and intent exploration	19.2.0
2025-06	SA#108	SP-250537	0351	1	B	Rel-19 CR TS 28.312 Add YAML document examples for IntentHandlingCapability	19.2.0
2025-06	SA#108	SP-250537	0352	1	B	Rel-19 CR TS 28.312 Enhance the solution for network maintenance expectation	19.2.0
2025-06	SA#108	SP-250558	0353	-	F	Rel-19 TS 28.312 Add qualifier description for coverageAreaTAContext	19.2.0
2025-06	SA#108	SP-250537	0354	1	C	Rel-19 CR TS 28.312 Add description of intent negotiation initiated by MnS producer	19.2.0
2025-06	SA#108	SP-250537	0355	1	C	Rel-19 CR TS 28.312 Add description of implicitIntent	19.2.0
2025-06	SA#108	SP-250537	0356	1	B	Rel-19 CR TS 28.312 Update intent utility function report	19.2.0
2025-06	SA#108					MCC: Move the new clause A.3 to clause H.	19.2.1
2025-09	SA#109	SP-251089	0357	2	F	Rel-19 CR TS 28.312 Clean up	19.3.0
2025-09	SA#109	SP-251077	0358		C	Rel-19 CR TS 28.312 Correct RESTful HTTP-based solution set for intent driven management	19.3.0
2025-09	SA#109	SP-251089	0359	1	F	Rel-19 CR TS 28.312 Correct issues for stage3 YAML files	19.3.0
2025-09	SA#109	SP-251089	0362	1	F	Rel-19 CR TS 28.312 Clarification on the description of implicitIntent	19.3.0

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2025-09	SA#109	SP-251089	0364		F	Update Clause 5.3.8 Utility Function	19.3.0
2025-09	SA#109	SP-251089	0365	1	F	Update intent report control	19.3.0
2025-09	SA#109	SP-251089	0366	1	F	Update intent report types	19.3.0
2025-09	SA#109	SP-251089	0367	1	F	Update Annex C	19.3.0

History

Document history		
V19.3.0	November 2025	Publication