

ETSI TS 128 312 V17.0.1 (2022-07)



**LTE;
5G;
Management and orchestration;
Intent driven management services for mobile networks
(3GPP TS 28.312 version 17.0.1 Release 17)**



Reference

DTS/TSGS-0528312vh01

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

Important notice

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

If you find a security vulnerability in the present document, please report it through our
Coordinated Vulnerability Disclosure Program:

<https://www.etsi.org/standards/coordinated-vulnerability-disclosure>

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2022.
All rights reserved.

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 Web server (<https://ipr.etsi.org/>).

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™** and **LTE™** 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 under <http://webapp.etsi.org/key/queryform.asp>.

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

| | |
|---|----|
| Intellectual Property Rights | 2 |
| Legal Notice | 2 |
| Modal verbs terminology..... | 2 |
| Foreword..... | 5 |
| Introduction | 6 |
| 1 Scope | 7 |
| 2 References | 7 |
| 3 Definitions of terms, symbols and abbreviations | 7 |
| 3.1 Terms..... | 7 |
| 3.2 Symbols..... | 7 |
| 3.3 Abbreviations | 8 |
| 4 Concepts and Background..... | 8 |
| 4.1 Intent concept | 8 |
| 4.1.1 Introduction..... | 8 |
| 4.1.2 Intent categorizes based on user types | 8 |
| 4.1.3 Intent expectations for different types of management needs..... | 9 |
| 4.2 Intent driven management | 10 |
| 4.2.1 Support for intent driven management..... | 10 |
| 4.2.2 Intent driven MnS | 10 |
| 4.2.3 Intent translation | 11 |
| 4.3 Intent driven closed-loop..... | 12 |
| 4.4 Relation between rule, policy and intent | 12 |
| 4.5 General concept of Intent Content..... | 13 |
| 4.5.1 Intent Expectation | 13 |
| 4.5.2 Expectation Targets | 13 |
| 4.5.3 Expectation Objects | 14 |
| 4.5.4 Context..... | 14 |
| 5 Specification Level Requirements..... | 15 |
| 5.1 Use cases | 15 |
| 5.1.1 Intent containing an expectation for delivering radio network | 15 |
| 5.1.1.1 Introduction..... | 15 |
| 5.1.1.2 Requirements | 15 |
| 5.1.2 Intent containing an expectation for delivering a radio service | 15 |
| 5.1.2.1 Introduction..... | 15 |
| 5.1.2.2 Requirements | 16 |
| 5.1.3 Intent containing an expectation for delivering a service | 16 |
| 5.1.3.1 Introduction..... | 16 |
| 5.1.3.2 Requirements | 16 |
| 5.1.4 Intent containing an expectation on coverage performance to be assured | 16 |
| 5.1.4.1 Introduction..... | 16 |
| 5.1.4.2 Requirements | 16 |
| 5.1.5 Intent containing an expectation on RAN UE throughput performance to be assured | 17 |
| 5.1.5.1 Introduction..... | 17 |
| 5.1.5.2 Requirements | 17 |
| 5.2 Generic requirements for intent driven MnS..... | 17 |
| 6 Stage 2 definition for Intent Driven Management..... | 17 |
| 6.1 Management operation for Intent (MnS component type A) | 17 |
| 6.2 Information model definition for Intent (MnS component typeB) | 18 |
| 6.2.1 Generic Information model definition | 18 |
| 6.2.1.1 Class diagram..... | 18 |
| 6.2.1.1.1 Relationship..... | 18 |
| 6.2.1.1.2 Inheritance | 18 |

| | | |
|--|--|-----------|
| 6.2.1.2 | Class definition | 19 |
| 6.2.1.2.1 | Intent <<IOC>> | 19 |
| 6.2.1.3 | DataType definition | 19 |
| 6.2.1.3.1 | IntentExpectation <<dataType>> | 19 |
| 6.2.1.3.3 | ExpectationObject <<dataType>> | 20 |
| 6.2.1.3.4 | ExpectationTarget <<dataType>> | 20 |
| 6.2.1.3.5 | Context <<dataType>> | 21 |
| 6.2.1.3.6 | FulfilmentInfo << dataType >> | 21 |
| 6.2.1.4 | Attribute definition | 22 |
| 6.2.2 | Scenario specific IntentExpectation definition | 25 |
| 6.2.2.1 | Scenario specific IntentExpectation definition | 25 |
| 6.2.2.1.1 | Radio Network Expectation | 25 |
| 6.2.2.1.2 | Service Support Expectation | 26 |
| 6.2.2.2 | Attribute definition | 28 |
| 6.3 | Procedures for intent management | 33 |
| 6.3.1 | Introduction | 33 |
| 6.3.2 | Create an intent | 33 |
| 6.3.3 | Modify an intent | 34 |
| 6.3.4 | Delete an intent | 35 |
| 6.3.5 | Query an intent | 36 |
| 7 | Stage 3 definition for Intent Driven Management | 36 |
| 7.1 | RESTful HTTP-based solution set | 36 |
| 7.2 | OpenAPI specification | 36 |
| 7.2.1 | OpenAPI document "TS28532_ProvMnS.yaml" | 36 |
| 7.2.2 | OpenAPI document "TS28312_IntentNrm.yaml" | 36 |
| Annex A (informative): PlantUML source code | | 46 |
| A.1 | Procedures for intent management | 46 |
| A.1.1 | Create an intent | 46 |
| A.1.2 | Modify an intent | 46 |
| A.1.3 | Delete an intent | 47 |
| A.1.4 | Query an intent | 47 |
| A.2 | Information model definition for Intent | 47 |
| A.2.1 | Relationship UML diagram for intent (Figure 6.2.1.1.2-1) | 47 |
| Annex B (informative): Intent Life Cycle Management | | 49 |
| B.1 | Intent Life Cycle Management | 49 |
| Annex C (informative): Mapping the 3GPP and the TM Forum intentExpectation Models | | 51 |
| Annex D (informative): Change history | | 52 |
| History | | 53 |

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".

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

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].

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. 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.

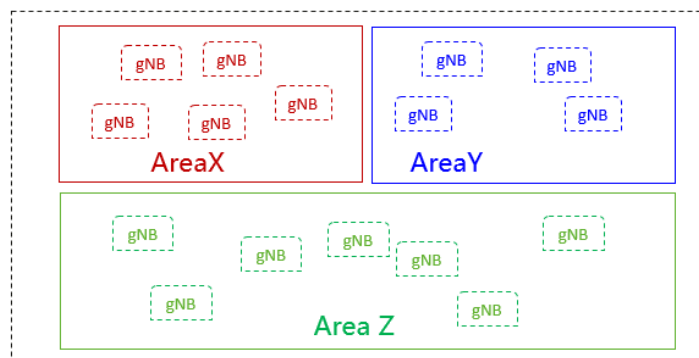


Figure 4.1.1-1

- 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.

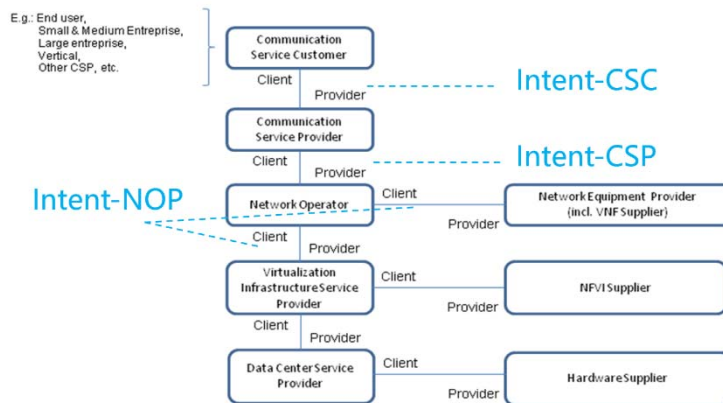


Figure 4.1.2-1: High-level model of different kind of intents expressed by different roles

- **Intent from Communication Service Customer (Intent-CSC):** Intent from Communication Service Customer (CSC) 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 (CSP) 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'.

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 throughput 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 provisioning MnSs 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 is what 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 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 validates the intent and then translates the intent to identify the required internal logic needed to fulfil the intent.
- The producer executes the compiled logic 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 or service's state to satisfy the new or updated intent. The fulfilment of some intents may end at the intent deployment, the case, if the intent's goal simply describes the availability or presence of a service. In other cases, the intent's goal describes 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 service. Those intents have their fulfilment tied to the operation of the referred service or network function 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 capabilities:

- Validate the intent.
- Translate the received intent to executable actions as follows:
 - Performing service or network management tasks.
 - Identifying, formulating and activating service or network management policies.
- Evaluate the result/information about the intent fulfilment (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.

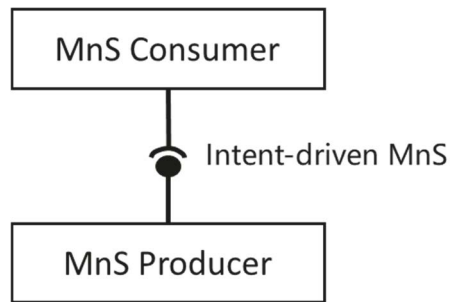


Figure 4.2.2-1: Intent-driven MnS

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 in solutions of various complexity, ranging from a simple approach rule-based mechanisms, to more elaborate solutions combining AI/ML, closed loop automation to ensure the fulfilment of intents.

When the intent is created on the MnS producer, 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 to create a new intent on the MnS producer.
- Activate an intent, MnS Consumer request to activate an intent on the MnS producer when the intent is suspended.
- De-activate an intent, MnS consumer request to de-activate an intent on the MnS producer for a temporary suspension.
- Delete an intent, MnS Consumer request to remove an intent on the MnS producer.
- Modify an intent, MnS Consumer request to modify the content of the intent (e.g. expectation targets) on the MnS producer.
- Query an intent, MnS Consumer request to return the content and state (e.g. active, inactive) of the intent on the MnS producer.

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 illustrate the potential way to satisfy intents originating from 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 element 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 element requirements, then takes some internal actions to fulfil the intent-NEP.

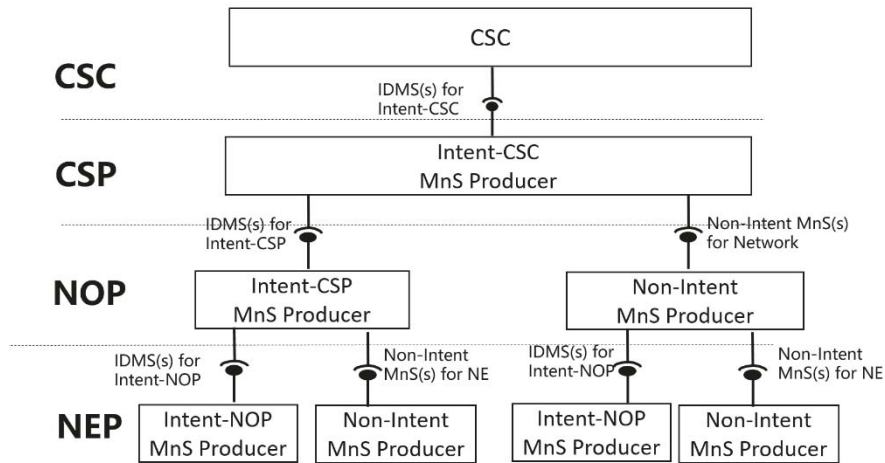


Figure 4.2.3-1: Potential way to satisfy intent-CSC originating from CSC

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.

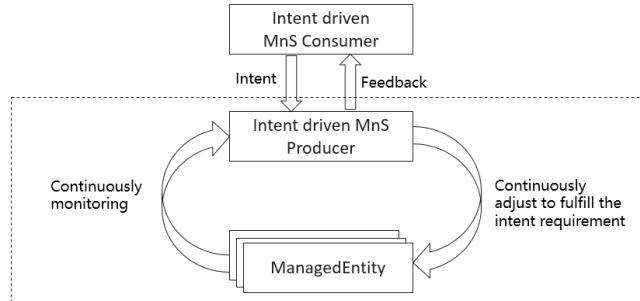


Figure 4.3-1: Intent driven closed-loop

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.

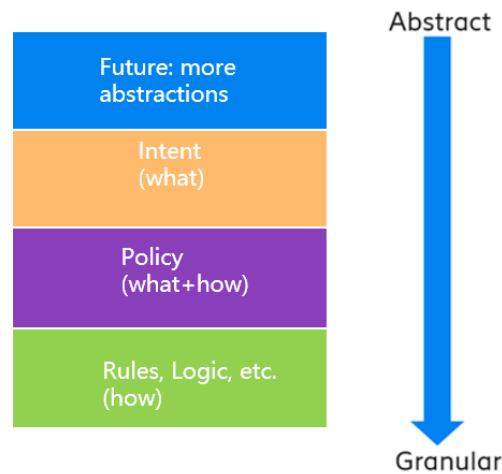


Figure 4.4-1: Relation between rule, policy and intent

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 {O₁, O₂, ... 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 behavior 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 a 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:

$$\text{Expectation Target} = [\text{targetName}, \text{condition}, \text{value range}]$$

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

| Example of Expectation Targets | ExpectationObject | targetName | Condition | Value range |
|--------------------------------|-----------------------|-----------------|-----------------|--------------|
| example 1 | Slice | Coverage area | Is at least | 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 slices in city ABC" as opposed to listing the individual slices. As such it may be easier to identify the objects by stating the object context information that filters and identifies the desired objects.

The objectContext is in form of a context list whose entries are each a tuple (attribute, condition, value range).

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

4.5.4 Context

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

Similar to the target, the context is also a tuple of < attribute, condition, value range > but 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 may wish that the Radio Link Failure rate (RLF) is less than 2 % when the load is more than 50 %. If the context (i.e. load > 50 %) is not explicitly stated/modelled as context, the producer could interpret the request to mean (RLF < 2 % and load > 50 %).

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

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).

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

REQ-Intent_Deploy_Net-CON-2 The intent driven MnS shall have capability enabling MnS consumer to obtain fulfilment information of 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).

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

REQ-Intent_Deploy_Sev-CON-2 The intent driven MnS shall have capability enabling MnS consumer to obtain fulfilment information of the intent containing an expectation for a service.

5.1.3 Intent containing an expectation for delivering a service

5.1.3.1 Introduction

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

5.1.3.2 Requirements

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

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, NEP 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.

5.1.4.2 Requirements

REQ-Intent_Opt_Cov-CON-1 The intent driven MnS shall have capability enabling MnS consumer to express intent containing an expectation on coverage performance to be assured for the specified area to MnS producer.

REQ-Intent_Opt_Cov-CON-2 The intent driven MnS shall have capability enabling MnS consumer to obtain fulfilment information of the intent containing an expectation on coverage performance to be assured.

5.1.5 Intent containing an expectation on RAN UE throughput performance to be assured

5.1.5.1 Introduction

In this scenario, MnS consumer expresses its intent containing an intent expectation on RAN UE throughput performance to be assured to MnS producer, which may include area information (e.g. geographical area), RATs (e.g. NR only, EUTRAN only, or all RATs), RAN UE throughput targets (e.g. 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)), optional performance scope (e.g. specific service type, specific UE groups).

Based on the intent containing expectation on RAN UE throughput performance to be assured received, MnS producer collects and analyses corresponding RAN UE throughput related data in the specified areas, identifies the potential RAN UE throughput issues (e.g. low RAN UE throughput for certain areas, high load for certain areas, frequent handover), which will impact RAN UE throughput intent satisfaction, analyses, evaluates, decides and adjusts the radio feature configuration parameters for impacted RAN NEs/Cells in the specified areas. The Artificial intelligence or machine learning technologies may be used to select the optimal radio feature configuration parameters set rapidly to satisfy RAN UE throughput target.

MnS producer continuously monitors the RAN UE throughput 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)) for the specified area, and decides whether RAN UE throughput target is satisfied.

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

5.1.5.2 Requirements

REQ-Intent_Opt_Thp-CON-1 The intent driven MnS shall have capability enabling MnS consumer to express intent containing an expectation on RAN UE throughput performance to be assured for specified area to MnS producer.

REQ-Intent_Opt_Thp-CON-2 The intent driven MnS shall have capability enabling MnS consumer to obtain fulfilment information of intent containing an expectation on RAN UE throughput performance to be assured.

5.2 Generic requirements for intent driven MnS

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

REQ-Intent_Driven_MnS-CON-2 The intent driven MnS shall have capability enabling MnS consumer to request MnS producer to remove an Intent object.

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

6 Stage 2 definition for Intent Driven Management

6.1 Management operation for Intent (MnS component type A)

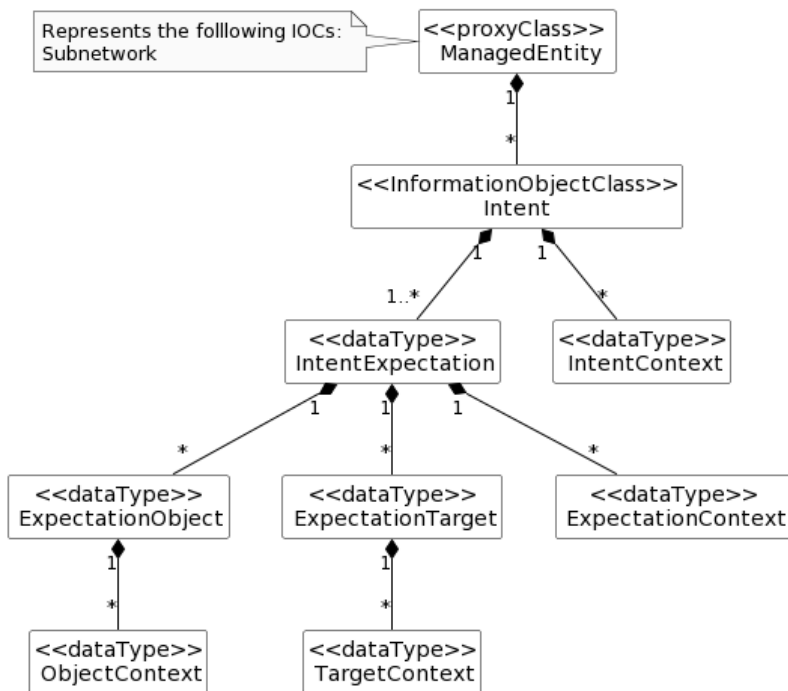
The operations (e.g. createMOI operations) and notifications (e.g. notifyMOIcreation) of generic provisioning MnS defined in 3GPP TS 28.532 [3] can be used for intent lifecycle management. The intent can be treated as object instance.

6.2 Information model definition for Intent (MnS component typeB)

6.2.1 Generic Information model definition

6.2.1.1 Class diagram

6.2.1.1.1 Relationship



NOTE: The model for IntentReport is not addressed in the present document.

Figure 6.2.1.1.1-1: Relationship UML diagram for intent

6.2.1.1.2 Inheritance

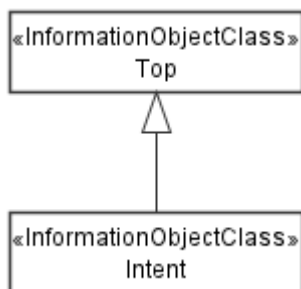


Figure 6.2.1.1.2-1: Inheritance UML diagram for intent

6.2.1.2 Class definition

6.2.1.2.1 Intent <<IOC>>

6.2.1.2.1.1 Definition

This IOC represents the properties of an Intent 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 includes the attribute objectClass and objectInstance from the TOP IOC. The value of attribute objectClass is "Intent" and the value of attribute objectInstance is the DN of the instance of Intent IOC.

6.2.1.2.1.2 Attributes

The Intent 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 | isNotifiable |
|----------------------|-------------------|------------|------------|-------------|--------------|
| intentExpectations | M | T | T | F | F |
| userLabel | M | T | T | F | F |
| intentContexts | O | T | T | F | F |
| intentFulfilmentInfo | M | T | F | F | T |

6.2.1.2.1.3 Attribute constraints

None.

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 information of IntentExpectation is generated by MnS consumer and delivered to MnS producer.

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 | isNotifiable |
|---------------------------|---|------------|------------|-------------|--------------|
| expectationId | M | T | T | T | F |
| expectationVerb | O | T | T | T | F |
| expectationObject | O | T | T | F | F |
| expectationTargets | M | T | T | F | F |
| expectationContexts | O | T | T | F | F |
| expectationfulfilmentInfo | O | T | F | F | T |
| NOTE: | The scenario/requirements-specific IntentExpectations are defined utilizing the constructs of this generic IntentExpectation model. | | | | |

6.2.1.3.1.3 Attribute constraints

None.

6.2.1.3.3 ExpectationObject <<dataType>>

6.2.1.3.3.1 Definition

The ExpectationObject <<dataType>> represents the Object of the IntentExpectation that are required to be applied on.

6.2.1.3.3.2 Attributes

The ExpectationObject includes the following attributes.

Table 6.2.1.3.3.2-1

| 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 |

6.2.1.3.3.3 Attribute constraints

Table 6.2.1.3.3.3-1

| Name | Definition |
|-------------------------------------|---|
| objectType Support Qualifier | 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 Support Qualifier | Condition: The intent expectation is for a specific object instance and MnS consumer have the knowledge of the DN of this specific object instance. |

6.2.1.3.4 ExpectationTarget <<dataType>>

6.2.1.3.4.1 Definition

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

6.2.1.3.4.2 Attributes

The ExpectationTarget includes the following attributes.

Table 6.2.1.3.4.2-1

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

6.2.1.3.4.3 Attribute constraints

None.

6.2.1.3.5 Context <<dataType>>

6.2.1.3.5.1 Definition

The Context <<dataType>> represents the properties of a context. A context describes the list of constraints and conditions that should evaluate to true when the targets are fulfilled but are themselves not to be enforced. The context may apply to the intent, the intent expectation, the intent targets or to the object.

6.2.1.3.5.2 Attributes

The Context includes the following attributes.

Table 6.2.1.3.5.2-1

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

6.2.1.3.5.3 Attribute constraints

None.

6.2.1.3.6 FulfilmentInfo << dataType >>

6.2.1.3.6.1 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 has been 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 fulfilled or not fulfilled. The possible values of the fulfilment include:

- NOTFULFILLED: This is the default status for any aspect of the intent and the fulfilmentStatus remains as "NOTFULFILLED" until the MnS producer is satisfied that 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 has been fulfilled as desired by the MnS consumer that created the intent. The consumer may provide a fulfilment satisfaction report that either confirms the fulfilment or describes its evaluation the fulfilment.

The degree of fulfilment of an intent with the NOTFULFILLED status may have multiple explanations and related states. These different progress states and conditions are recorded in the notFulfilledState field. The possible values of the notFulfilledState include:

- ACKNOWLEDGED: this is the default status and is the initial notFulfilledState right after the intent has been received.
- COMPLIANT: this is the state after the feasibility check has been run for the intent and the intent 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 decides to suspect 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 cancelled.
- **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 "SUSPENDED" and the "FULFILMENTFAILED" notFulfilledStates), the notFulfilledState should be supported by extra information describing or related to the state. This extra information is recorded into the notFulfilledReasons field.

6.2.1.3.6.2 Attributes

The FulfilmentInfo includes the following attributes.

Table 6.2.1.3.6.2-1

| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifiable |
|---------------------|-------------------|------------|------------|-------------|--------------|
| fulfilmentStatus | M | T | F | F | T |
| notFulfilledState | CM | T | F | F | T |
| notFulfilledReasons | CO | T | F | F | T |

6.2.1.3.6.3 Attribute constraints

Table 6.2.1.3.6.3-1

| Name | Definition |
|--|---|
| notFulfilledState Support Qualifier | Condition: when FulfilmentInfo is implemented for IntentFulfilmentInfo |
| notFulfilledReasons Support Qualifier | Condition: when FulfillmentInfo is implemented for IntentFulfilmentInfo |

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). allowedValues: Not Applicable | type: IntentExpectation multiplicity: 1..* isOrdered: False 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 |

| Attribute Name | Documentation and Allowed Values | Properties |
|---------------------------|--|---|
| 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 defaultValue: None isNullable: False |
| intentFulfillStatus | It describes the current status of the intent fulfilment result, which is configured by MnS producer and can be read by MnS consumer. allowedValues: "FULFILLED", "NOT_FULFILLED" | type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False |
| notFulfilledState | It describes the current progress of or the reason 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: None isNullable: False |
| notFulfilledReason | It describes the reasons/observations related to the specific noted notFulfilledState allowedValues: Not Applicable | type: String multiplicity: 1 isOrdered: N/A isUnique: N/A 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 user-friendly (and user assignable) name of the intentExpectation. 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 performance targets values allowedValues: Deliver, Ensure | type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False |

| Attribute Name | Documentation and Allowed Values | Properties |
|---------------------|--|--|
| expectationObject | It describes the expectation objects of the IntentExpectation that are required to be applied on. 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 are required to be applied on. It can be class name of the managed object. allowedValues: see scenario specific Intent Expectation | type: Enum multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None 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 | type: ExpectationTarget multiplicity: 1..* isOrdered: False 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: 1..* isOrdered: False isUnique: True defaultValue: None isNullable: False |
| targetName | It describes the name of the Expectation 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: Null isNullable: True |
| targetCondition | It expresses the limits within which the targetName is allowed/supposed to be allowedValues: is equal to; is less than; is greater than: - "is within the range"; - "is outside the range" | 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 targetName | type: Real multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: Null 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: 1..* 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: Null isNullable: True |
| contextCondition | It expresses the limits within which the ContextAttribute is allowed/supposed to be allowedValues: is equal to; is less than; is greater than; "is within the range"; "is outside the range" | 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 contextAttribute | type: Real multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: Null isNullable: True |

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 be 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.

Table 6.2.2.1.1.2-1

| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifiable |
|----------------------------|-------------------|------------|------------|-------------|--------------|
| coverageAreaPolygonContext | O | T | T | F | F |
| coverageTACContext | O | T | T | F | F |
| pLMNContext | O | T | T | F | F |
| nRFqBandContext | O | T | T | F | F |
| rATContext | 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 same with properties of ExpectationTargets defined in clause 6.2.1.3.

Table 6.2.2.1.1.3-1

| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifiable |
|---------------------------|-------------------|------------|------------|-------------|--------------|
| 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 |

6.2.2.1.2 Service Support Expectation

6.2.2.1.2.1 Definition

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

The 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 Service Support Expectation.

Table 6.2.2.1.2.1-1

| Attribute | Allowed Values |
|---------------------|--------------------------|
| ObjectType (CM) | ServiceSupport |
| objectInstance (CM) | DN of the ServiceSupport |

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 Service Support 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 6.2.2.1.2.2-1

| Attribute Name | Support Qualifier | isReadable | isWritable | isInvariant | isNotifyable |
|----------------------------|-------------------|------------|------------|-------------|--------------|
| edgeIdenficationIdContext | CM | T | T | F | F |
| edgeIdenficationLocContext | CM | T | T | F | F |
| coverageAreaTAContext | CM | T | T | F | F |

NOTE: Following are the qualifier description for attribute "edgeIdentificationId" and "edgeIdentificationLoc":

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

6.2.2.1.2.3 ExpectationTargets

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

Following provides the concrete ExpectationTargets for Service Deployment Expectation based on the common structure of ExpectationTarget. The attribute properties defined in the table below should be same with the properties defined for ExpectationTargets 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 |
| serviceEndTimeTargetContext | O | T | T | F | F |
| uEMobilityLevelContext | O | T | T | F | F |
| resourceSharingLevelContext | 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 | <p>It describes the coverage areas for the RAN SubNetwork that the intent expectation is applied in the form of polygon.</p> <p>CoverageAreaPolygonContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "CoverageAreaPolygon" - contextCondition: "With the range" - contextValueRange: a list of CoverageArea defined in 3GPP TS 28.541 [5] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| coverageTACContext | <p>It describes the coverage areas for the RAN SubNetwork that the intent expectation is applied in the form of TAC.</p> <p>CoverageTACContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "CoverageAreaTAC" - contextCondition: "With the range" - contextValueRange: a list of nRTAC defined in 3GPP TS 28.541 [5] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| plmnContext | <p>It describes the PLMN(s) supported by the RAN SubNetwork that the intent expectation is applied.</p> <p>PLMNContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "PLMN" - contextCondition: "With the range" - contextValueRange: a list of PLMNid defined in 3GPP TS 28.541 [5] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| nRFqBandContext | <p>It describes the nRFqBands supported by the RAN SubNetwork that the intent expectation is applied.</p> <p>nRFqBandContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "NRFqBand" - contextCondition: "With the range" - contextValueRange: a list of NRFqBand expressed as string. Valid frequency band values are specified in clause 5.4.2 in 3GPP TS 38.104 [8] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| rATContext | <p>It describes the RAT supported by the RAN SubNetwork that the intent expectation is applied.</p> <p>RATContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "RAT" - contextCondition: "With the range" - contextValueRange: a list of ENUM with allowed value: UTRAN, EUTRAN and NR | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |

| Attribute Name | Documentation and Allowed Values | Properties |
|---------------------------------------|--|---|
| WeakRSRPRatioTarget | <p>It describes the downlink weak coverage ratio target for the RAN SubNetwork that the intent expectation is applied.</p> <p>WeakRSRPRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetName: "WeakRSRPRatio" - targetCondition: "is less than" - targetValueRange: integer with allowed value [0,100] - targetContext: WeakRSRPContext | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| WeakRSRPRatioTarget . WeakRSRPContext | <p>It describes the threshold for downlink weak RSRP of the RAN SubNetwork that the intent expectation is applied.</p> <p>WeakRSRPContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "WeakRSRPThreshold" - contextCondition: "is less than" - contextValueRange: Float | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| LowSINRRatioTarget | <p>It describes the low SINR ratio target for the RAN SubNetwork that the intent expectation is applied.</p> <p>LowSINRRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetName: "WeakRSRPRatio" - targetCondition: "is less than" - targetValueRange: integer with allowed value [0,100] - targetContext: LowSINRContext | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| LowSINRRatioTarget . LowSINRContext | <p>It describes the threshold for low SINR for RAN SubNetwork that the intent expectation is applied.</p> <p>LowSINRContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "LowSINRThreshold" - contextCondition: "is less than" - contextValueRange: integer | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| aveULRANUETHptTarget | <p>It describes the average UL RAN UE throughput target for RAN SubNetwork that the intent expectation is applied.</p> <p>AveULRANUETHptTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetName: "AveULRANUETHpt" - targetCondition: "is greater than" - targetValueRange: integer | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| aveDLRANUETHptTarget | <p>It describes the average DL RAN UE throughput target for RAN SubNetwork that the intent expectation is applied.</p> <p>AveDLRANUETHptTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetName: "AveDLRANUETHpt" - targetCondition: "is greater than" - targetValueRange: integer | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |

| Attribute Name | Documentation and Allowed Values | Properties |
|---|--|---|
| lowULRANUEThptRatioTarget | <p>It describes the low UL RAN UE throughput ratio target for the RAN SubNetwork that the intent expectation is applied.</p> <p>LowULRANUEThptRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetName: "LowULRANUEThptRatio" - targetCondition: "is less than" - targetValueRange: integer with allowed value [0,100] - targetContext: LowULRANUEThptContext | <p>type: ExpectationTarget</p> <p>multiplicity: 1</p> <p>isOrdered: N/A</p> <p>isUnique: N/A</p> <p>defaultValue: False</p> <p>isNullable: True</p> |
| LowULRANUEThptRatioTarget.lowULRANUEThptContext | <p>It describes the threshold for the low UL RAN UE throughput of the RAN SubNetwork that the intent expectation is applied</p> <p>LowULRANUEThptContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "LowULRANUEThptThreshold" - contextCondition: "is less than" - contextValueRange: Float | <p>type: Context</p> <p>multiplicity: 1</p> <p>isOrdered: N/A</p> <p>isUnique: N/A</p> <p>defaultValue: False</p> <p>isNullable: True</p> |
| lowDLRANUEThptRatioTarget | <p>It describes the low DL RAN UE throughput ratio target for the RAN SubNetwork that the intent expectation is applied.</p> <p>LowDLRANUEThptRatioTarget is an ExpectationTarget including attributes: targetName, targetCondition, targetValueRange and targetContext.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetName: "LowDLRANUEThptRatio" - targetCondition: "is less than" - targetValueRange: integer with allowed value [0,100] - targetContext: LowDLRANUEThptContext | <p>type: ExpectationTarget</p> <p>multiplicity: 1</p> <p>isOrdered: N/A</p> <p>isUnique: N/A</p> <p>defaultValue: False</p> <p>isNullable: True</p> |
| LowDLRANUEThptRatioTarget.lowDLRANUEThptContext | <p>It describes the threshold for the low DL RAN UE throughput of the RAN SubNetwork that the intent expectation is applied.</p> <p>LowDLRANUEThptContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "LowDLRANUEThptThreshold" - contextCondition: "is less than" - contextValueRange: Float | <p>type: Context</p> <p>multiplicity: 1</p> <p>isOrdered: N/A</p> <p>isUnique: N/A</p> <p>defaultValue: False</p> <p>isNullable: True</p> |
| serviceStartTimeContext | <p>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.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "serviceStartTime" - contextCondition: "is equal than" - contextValueRange: start time stamp | <p>type: Context</p> <p>multiplicity: 1</p> <p>isOrdered: N/A</p> <p>isUnique: N/A</p> <p>defaultValue: False</p> <p>isNullable: True</p> |
| serviceEndTimeContext | <p>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.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "serviceEndTime" - contextCondition: "is equal than" - contextValueRange: end time stamp | <p>type: Context</p> <p>multiplicity: 1</p> <p>isOrdered: N/A</p> <p>isUnique: N/A</p> <p>defaultValue: False</p> <p>isNullable: True</p> |

| Attribute Name | Documentation and Allowed Values | Properties |
|-----------------------------------|--|--|
| edgeIdentificationIdContext | <p>This identifies the edge network where the service need to be deployed. This should be used when the edge identification is known to the consumer</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "edgeIdentificationId" - contextCondition: "is equal than" - contextValueRange: EDNIdentifier as defined in 3GPP TS 28.538 [9] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| edgeIdentificationLocationContext | <p>This identifies the location where the service need to be deployed. This should be used when the edge identification is not known to the consumer</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "edgeIdentificationTarget" - contextCondition: "is equal than" - contextValueRange: geographical target location. This will take a form of either single latitude & longitude or a TAI | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| coverageAreaTAContext | <p>It describes Tracking Coverage Areas for service supporting that the intent expectation is applied.</p> <p>coverageAreaTAContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "coverageAreaTA" - contextCondition: "With the range" - contextValueRange: a list of Tracking Coverage Areas, coverageAreaTAList in clause 6.3.1 in 3GPP TS 28.541 [5] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| dLThptPerUETarget | <p>It describes the DL throughput target by the per UE for the service Supporting that the intent expectation is applied.</p> <p>DLThptperUETarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange:</p> <ul style="list-style-type: none"> - targetName: "DLThptperUE" - targetCondition: "is greater than" - targetValueRange: dLThptPerUE defined in clause 6.3.1 of 3GPP TS 28.541 [5] | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| ULThptPerUETarget | <p>It describes the UL throughput target by the per UE for the service Supporting that the intent expectation is applied.</p> <p>ULThptperUETarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <ul style="list-style-type: none"> - targetName: "ULThptperUE" - targetCondition: "is greater than" - targetValueRange: uLThptPerUE defined in clause 6.3.1 of 3GPP TS 28.541 [5] | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| dLLatencyTarget | <p>It describes the DL latency target for the service Supporting that the intent expectation is applied.</p> <p>DLLatencyTarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <ul style="list-style-type: none"> - targetName: "ULThptperUE" - targetCondition: "is less than" - targetValueRange: dLLatency defined in clause 6.3.1 of 3GPP TS 28.541 [5] | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |

| Attribute Name | Documentation and Allowed Values | Properties |
|-----------------------------|--|--|
| uLLatencyTarget | <p>It describes the UL latency target for the service Supporting that the intent expectation is applied.</p> <p>UIThtperUETarget is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <ul style="list-style-type: none"> - targetName: "UIThtperUE" - targetCondition: "is less than" - targetValueRange: ULLatency defined in clause 6.3.1 of 3GPP TS 28.541 [5] | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| maxNumberOfUEsTarget | <p>It describes the maximum number of UEs for service supporting that the intent expectation is applied.</p> <p>maxNumberOfUEsContext is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetAttribute: "maxNumberOfUEs" - targetCondition: " is less than " - targetValueRange: maxNumberOfUEs in clause 6.3.1 in 3GPP TS 28.541 [5] | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| activityFactorTarget | <p>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.</p> <p>activityFactorContext is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetAttribute: " activityFactorContext " - targetCondition: " is equal than " - targetValueRange: activityFactor in clause 6.3.1 in 3GPP TS 28.541 [5] | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| uESpeedTarget | <p>It describes the maximum speed (in km/hour) supported for service supporting that the intent expectation is applied.</p> <p>uESpeedContext is an ExpectationTarget including attributes: targetName, targetCondition and targetValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - targetAttribute: "uESpeedContext" - targetCondition: " is less than " - targetValueRange: uESpeedContext in clause 6.3.1 in 3GPP TS 28.541 [5] | <p>type: ExpectationTarget multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| uEMobilityLevelContext | <p>It describes the mobility level of UE for service supporting that the intent expectation is applied.</p> <p>uEMobilityLevelContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: " uEMobilityLevel " - contextCondition: "With the range" - contextValueRange: uEMobilityLevel in clause 6.3.1 in 3GPP TS 28.541 [5] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |
| resourceSharingLevelContext | <p>It describes the resource sharing level for service supporting that the intent expectation is applied.</p> <p>resourceSharingLevelContext is a Context including attributes: contextAttribute, contextCondition and contextValueRange.</p> <p>Following are the allowed values:</p> <ul style="list-style-type: none"> - contextAttribute: "resourceSharingLevel" - contextCondition: "With the range" - contextValueRange: resourceSharingLevel in clause 6.3.1 in 3GPP TS 28.541 [5] | <p>type: Context multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: True</p> |

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.

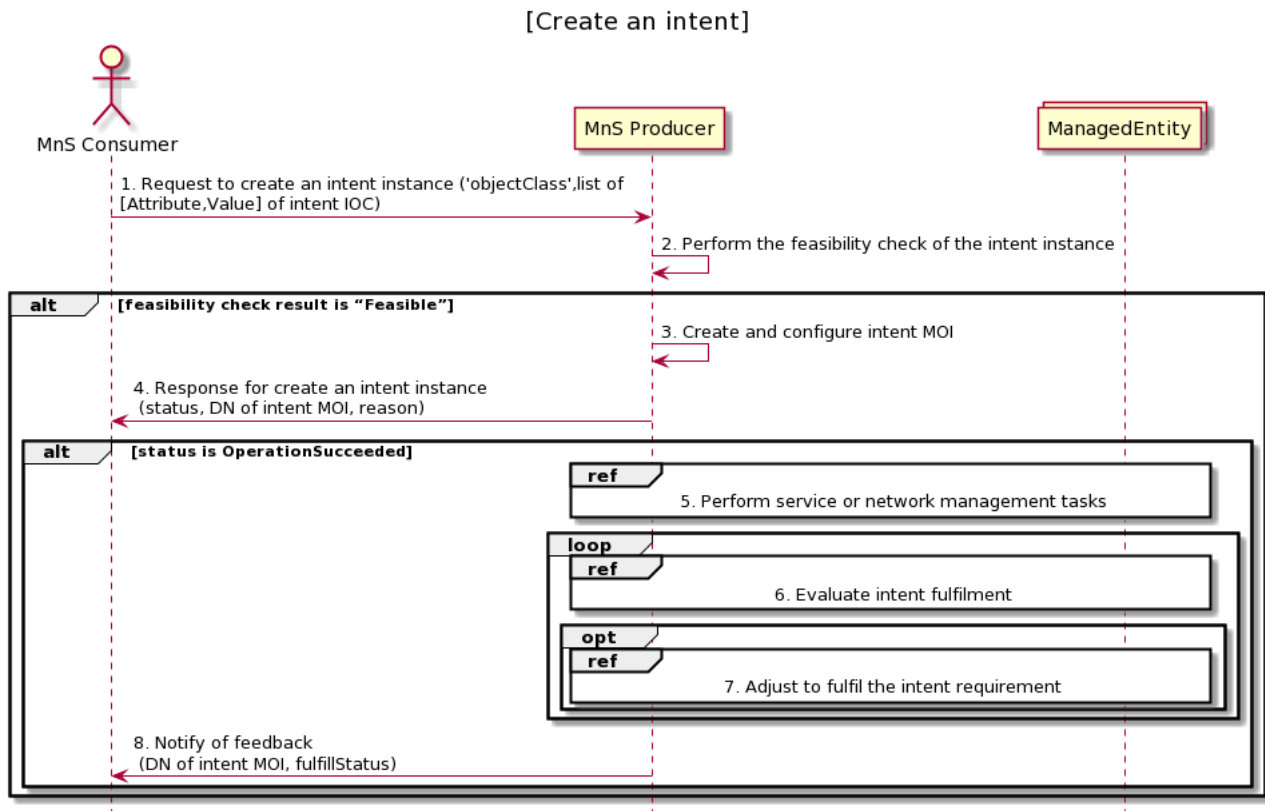


Figure 6.3.2-1: Procedure for create an intent

1. MnS Consumer sends a request to create an intent instance to MnS Producer with 'objectClass' and list of [Attribute, Value] for the new intent to be created. The detailed [Attribute, Value] see the concrete intent IOC defined in clause 6.2. 'objectClass' is the name for the concrete intent IOC.
2. MnS Producer perform 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: 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.

If the feasibility check result is 'feasible':

3. Based on the request, the MnS Producer creates the concrete intent MOI (i.e. instance of intent IOC) with value for attribute 'objectInstance' allocated, and configure the new created intent MOI with the received list of [Attribute, Value]. 'objectInstance' is the identifier (DN) for the concrete intent MOI.
4. MnS Producer sends a response to the MnS Consumer with status (OperationSucceeded or OperationFailed) and 'objectInstance' of the created intent MOI. The response information may also include the possible reasons for the unsuccessful executions (e.g. conflicting with existing intents).
5. Based on the 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 requirements) for these managed entities, then MnS producer deploys or configures corresponding managed entities to satisfy the intent.
6. During the execution of the intent, MnS Producer continuously monitors intent fulfilment status.
7. MnS Producer analyses and adjusts the managed entities to ensure the intent is continuously satisfied.
8. MnS Producer may notify MnS Consumer about the intent fulfilment information, including DN of intent MOI, and fulfillStatus.

If the feasibility check result is 'infeasible', MnS Producer does not create an intent MOI and feedback the feasibility check result information to MnS Consumer.

6.3.3 Modify an intent

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

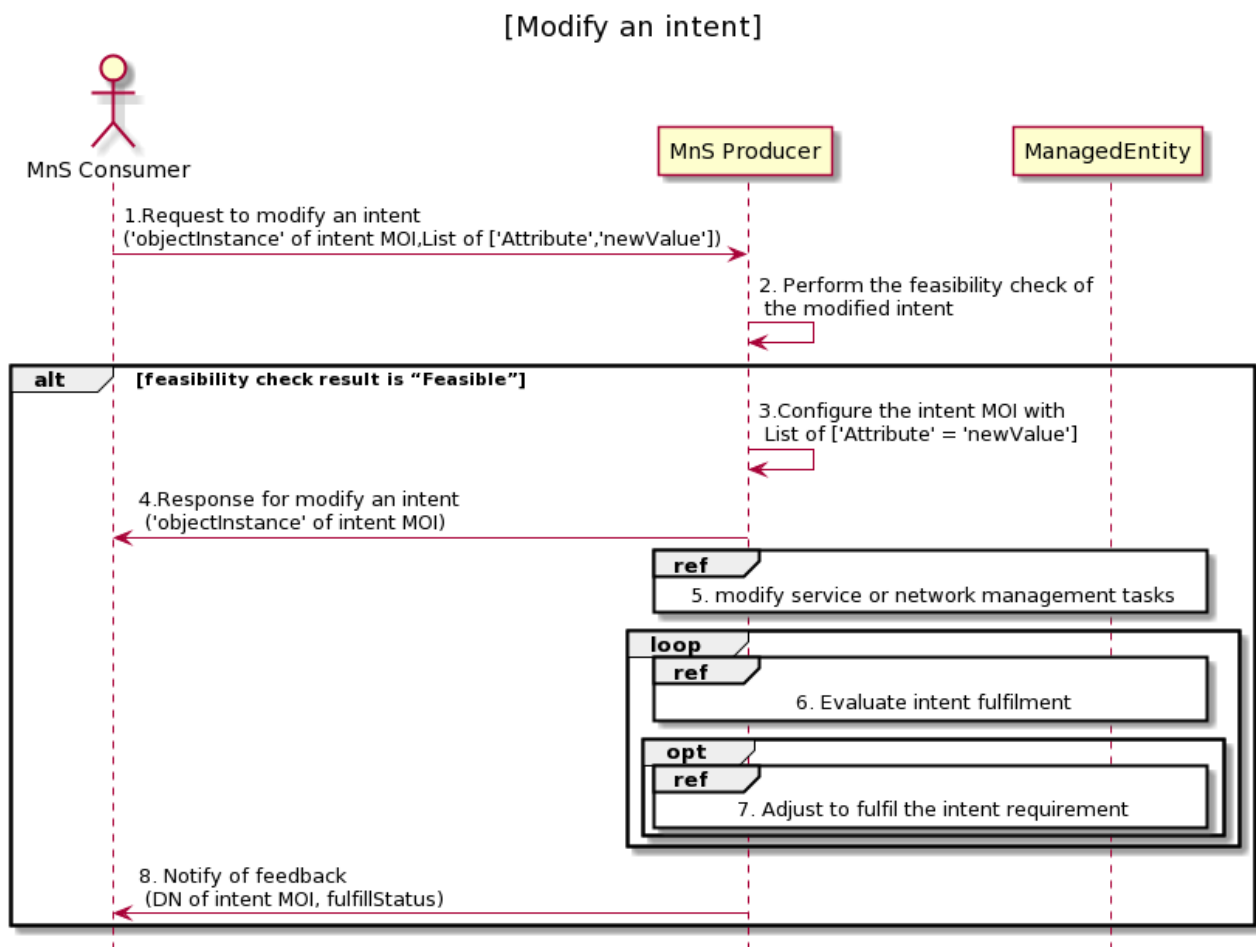


Figure 6.3.3-1: Procedure for modify an intent

1. MnS Consumer sends a request to modify an intent instance to MnS Producer with 'objectInstance' of the intent MOI and List of ['Attribute', 'newValue'] to be modified. The detailed [Attribute,Value] see the concrete intent IOC defined in clause 6.2.
2. MnS Producer perform the feasibility check of the modified intent instance. Whether to perform the feasibility check can be determined according to the feasibility check enabling policy.

If the feasibility check result is 'feasible':

3. Based on the request, MnS Producer configure the intent MOI with list of 'Attribute' = 'newValue' which is required to be modified.
4. MnS Producer sends a response to the MnS consumer with status (OperationSucceeded or OperationFailed), and 'objectInstance' of the modified intent MOI, and list of ['Attribute', 'newValue'] which is modified. The response information may also include possible reasons for the unsuccessful executions (e.g. conflicting with other intents).
5. 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 intent.
6. During the execution of the intent, MnS producer continuously tracks intent fulfilment status.
7. MnS producer analyses and adjusts the managed entities to ensure the intent is continuously satisfied.
8. MnS Producer may notify MnS Consumer about the intent fulfilment information, including DN of intent MOI, and fulfillStatus via notification or intent reporting.

If the feasibility check result is 'infeasible', MnS Producer does not modify the intent MOI and feedback the feasibility check result information to MnS Consumer.

6.3.4 Delete an intent

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

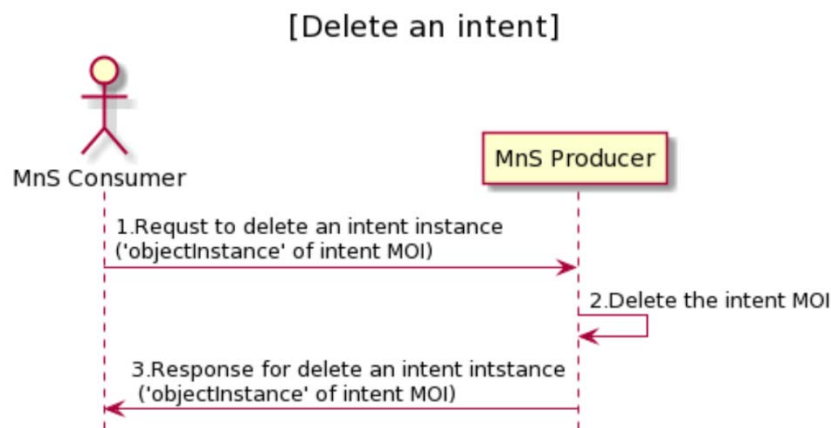


Figure 6.3.4-1: Procedure for delete an intent

1. MnS Consumer sends a request to delete an intent to MnS Producer with 'objectInstance' of the intent MOI.
2. Based on the request, MnS Producer delete the intent MOI.
3. MnS Producer sends response to the MnS consumer with status (OperationSucceeded or OperationFailed), 'objectInstance' of the deleted intent MOI.

6.3.5 Query an intent

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

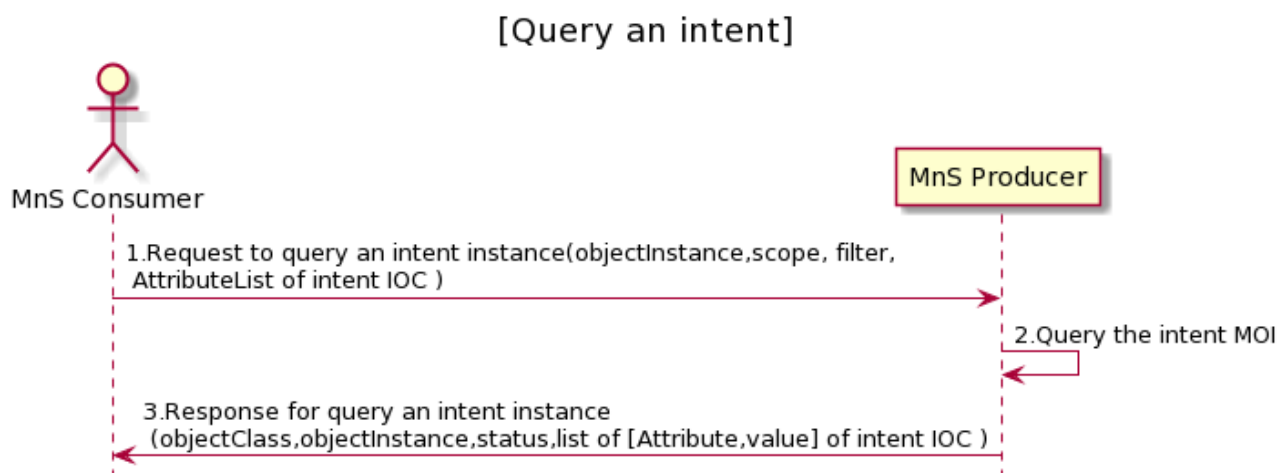


Figure 6.3.5-1: Procedure for query an intent

1. MnS Consumer sends a request to query an intent instance to MnS Producer, with objectInstance of the existing intent MOI, scope, and list of attributes of intent IOC. The list of attributes 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 to the MnS consumer with objectClass, objectInstance, status (e.g. fulfillStatus and other status), and list of [Attribute,Value] which is defined in clause 6.2.

7 Stage 3 definition for Intent Driven Management

7.1 RESTful HTTP-based solution set

The 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.

7.2 OpenAPI specification

7.2.1 OpenAPI document "TS28532_ProvMnS.yaml"

OpenAPI definition of the provisioning MnS which includes the provisioning MnS operations and the provisioning MnS notifications see clause A.1.1 in 3GPP TS 28.532 [3].

7.2.2 OpenAPI document "TS28312_IntentNrm.yaml"

```

openapi: 3.0.1
info:
  title: Intent NRM
  version: 17.0.0
  description: >-
    OAS 3.0.1 definition of the Intent NRM
    © 2022, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).
    All rights reserved.
externalDocs:
  description: 3GPP TS 28.312; Intent driven management services for mobile networks
  url: http://www.3gpp.org/ftp/Specs/archive/28_series/28.312/
  
```

```

paths: {}
components:
  schemas:

    Intent-Single:
      allOf:
        - $ref: 'TS28623_GenericNrm.yaml#/components/schemas/Top'
        - type: object
      properties:
        userLabel:
          type: string
        intentExpectations:
          type: array
          items:
            $ref: "#/components/schemas/IntentExpectation"
        intentContexts:
          type: array
          items:
            $ref: "#/components/schemas/IntentContext"
        intentFulfilmentInfo:
          $ref: "#/components/schemas/FulfilmentInfo"

    IntentExpectation:
      type: object
      properties:
        expectationId:
          type: string
        expectationVerb:
          $ref: "#/components/schemas/ExpectationVerb"
        expectationObjects:
          type: array
          items:
            $ref: "#/components/schemas/ExpectationObject"
        expectationTargets:
          type: array
          items:
            $ref: "#/components/schemas/ExpectationTarget"
        expectationContexts:
          type: array
          items:
            $ref: "#/components/schemas/ExpectationContext"
        expectationfulfilmentInfo:
          $ref: "#/components/schemas/FulfilmentInfo"

    ExpectationObject:
      type: object
      properties:
        objectType:
          type: string
          enum:
            - RAN_SubNetwork #value for Radio Network Expectation--#
            - Service_Support #value for Service Support Expectation--#
            - TBD #This will be added based on defined scenario specific intent expectation-#
        objectInstance:
          $ref: "TS28623_ComDefs.yaml#/components/schemas/Dn"
        objectContexts:
          type: array
          items:
            $ref: "#/components/schemas/ObjectContext"

    ExpectationTarget:
      oneOf:
        #--ExpectationTargets defind for Radio Network Expectation (ObjectType is RAN_SubNetwork)- #
        - $ref: "#/components/schemas/WeakRSRPRatioTarget"
        - $ref: "#/components/schemas/LowSINRRatioTarget"
        - $ref: "#/components/schemas/AveULRANUEThptTarget"
        - $ref: "#/components/schemas/AveDLRANUEThptTarget"
        - $ref: "#/components/schemas/LowULRANUEThptRatioTarget"
        - $ref: "#/components/schemas/LowDLRANUEThptRatioTarget"
        #--ExpectationTargets defind for Radio Network Expectation (ObjectType is RAN_SubNetwork)- #
        #--ExpectationTargets defind for Service Support Expectation (ObjectType is ServiceSupport)- #
        - $ref: "#/components/schemas/DLThptPerUETarget"
        - $ref: "#/components/schemas/ULThptPerUETarget"
        - $ref: "#/components/schemas/DLLatencyTarget"
        - $ref: "#/components/schemas/ULLatencyTarget"
        - $ref: "#/components/schemas/MaxNumberOfUEsTarget"
        - $ref: "#/components/schemas/ActivityFactorTarget"
        - $ref: "#/components/schemas/UESpeedTarget"
        #--ExpectationTargets defind for Service Support Expectation (ObjectType is ServiceSupport)- #

```

```

#--ExpectationTargets definid for other Scenario specific Expectation-----TBD-----#
#--ExpectationTargets definid for other Scenario specific Expectation-----TBD-----#

ObjectContext:
  oneOf:
    #--ObjectContext definid for Radio Network Expectation (ObjectType is RAN_SubNetwork)-----#
    - $ref: "#/components/schemas/CoverageAreaPolygonContext"
    - $ref: "#/components/schemas/CoverageTACContext"
    - $ref: "#/components/schemas/PLMNContext"
    - $ref: "#/components/schemas/NRFqBandContext"
    - $ref: "#/components/schemas/RATContext"
    #--ObjectContext definid for Radio Network Expectation (ObjectType is RAN_SubNetwork)-----#
    #--ObjectContext definid for Service Support Expectation (ObjectType is ServiceSupport)-----#
    - $ref: "#/components/schemas/EdgeIdenfiticationIdContext"
    - $ref: "#/components/schemas/EdgeIdenfiticationLocContext"
    - $ref: "#/components/schemas/CoverageAreaTACContext"
    #--ObjectContext definid for Service Support Expectation (ObjectType is ServiceSupport)-----#

    #--ObjectContext definid for other Scenario specific Expectation -----TBD-----#
    #--ObjectContext definid for other Scenario specific Expectation -----TBD-----#

ExpectationContext:
  oneOf:
    #--ExpectationContext definid for Service Support Expectation (ObjectType is Service_Support)- #
    - $ref: "#/components/schemas/ServiceStartTimeContext"
    - $ref: "#/components/schemas/ServiceEndTimeContext"
    - $ref: "#/components/schemas/UEMobilityLevelContext"
    - $ref: "#/components/schemas/ResourceSharingLevelContext"
    #--ExpectationContext definid for Service Support Expectation (ObjectType is Service_Support)- #

IntentContext:
  type: object
  properties:
    contextAttribute:
      type: string
    contextCondition:
      $ref: "#/components/schemas/Condition"
    contextValueRange:
      type: array
      items:
        type: number

Condition:
  type: string
  enum:
    - Is_equal_to
    - Is_less_than
    - Is_greater_than
    - Is_within_the_range

FulfilStatus:
  type: string
  enum:
    - FULFILLED
    - NOT_FULFILLED

NotFulfilledState:
  type: string
  enum:
    - ACKNOWLEDGED
    - COMPLIANT
    - DEGRADED
    - SUSPENDED
    - TERMINATED
    - FULFILMENTFAILED

FulfilmentInfo:
  type: object
  properties:
    fulfilStatus:
      $ref: "#/components/schemas/FulfilStatus"
    notFullfilledState:
      description: ->
      An attribute which is used when FulfilmentInfo is implemented for IntentFulfilmentInfo
      $ref: "#/components/schemas/NotFulfilledState"
    notFulfilledReasons:
      description: ->

```

```

    An attribute which is used when FulfilmentInfo is implemented for IntentFulfilmentInfo
    type: string

ExpectationVerb:
  type: string
  enum:
    - DELIVER
    - ENSURE

#-----Definition of the concrete ExpectationTarget dataType-----#
WeakRSRPRatioTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - WeakRSRPRatio
    targetCondition:
      type: string
      enum:
        - Is_less_than
    targetValueRange:
      type: integer
      minimum: 0
      maximum: 100
    targetContexts:
      $ref: "#/components/schemas/WeakRSRPContext"
    targetFulfilmentInfo:
      $ref: "#/components/schemas/FulfilmentInfo"

WeakRSRPContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - WeakRSRPThreshold
    contextCondition:
      type: string
      enum:
        - Is_less_than
    contextValueRange:
      type: number

LowSINRRatioTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - LowSINRRatio
    targetCondition:
      type: string
      enum:
        - Is_less_than
    targetValueRange:
      type: integer
      minimum: 0
      maximum: 100
    targetContexts:
      $ref: "#/components/schemas/LowSINRContext"
    targetFulfilmentInfo:
      $ref: "#/components/schemas/FulfilmentInfo"

LowSINRContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - LowSINRThreshold
    contextCondition:
      type: string
      enum:
        - Is_less_than
    contextValueRange:
      type: integer

```



```
AveULRANUEThptTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - AveULRANUEThpt
    targetCondition:
      type: string
      enum:
        - Is_greater_than
    targetValueRange:
      type: integer
    targetFulfilmentInfo:
      $ref: "#/components/schemas/FulfilmentInfo"

AveDLRANUEThptTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - AveDLRANUEThpt
    targetCondition:
      type: string
      enum:
        - Is_greater_than
    targetValueRange:
      type: integer
    targetFulfilmentInfo:
      $ref: "#/components/schemas/FulfilmentInfo"

LowULRANUEThptRatioTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - LowULRANUEThptRatio
    targetCondition:
      type: string
      enum:
        - Is_less_than
    targetValueRange:
      type: integer
      minimum: 0
      maximum: 100
    targetContexts:
      $ref: "#/components/schemas/LowULRANUEThptContext"
    targetFulfilmentInfo:
      $ref: "#/components/schemas/FulfilmentInfo"

LowULRANUEThptContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - LowULRANUEThptThreshold
    contextCondition:
      type: string
      enum:
        - Is_less_than
    contextValueRange:
      type: number

LowDLRANUEThptRatioTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - LowDLRANUEThptRatio
    targetCondition:
      type: string
      enum:
        - Is_less_than
    targetValueRange:
```

```
    type: integer
    minimum: 0
    maximum: 100
  targetContexts:
    $ref: "#/components/schemas/LowDLRANUEThptContext"
  targetFulfilmentInfo:
    $ref: "#/components/schemas/FulfilmentInfo"

LowDLRANUEThptContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - LowDLRANUEThptThreshold
    contextCondition:
      type: string
      enum:
        - Is_less_than
    contextValueRange:
      type: number
DLThptPerUETarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - DLThptPerUE
    targetCondition:
      type: string
      enum:
        - Is_greater_than
    targetValueRange:
      $ref: "TS28541_SliceNrm.yaml#/components/schemas/XLThpt"
ULThptPerUETarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - ULThptPerUE
    targetCondition:
      type: string
      enum:
        - Is_greater_than
    targetValueRange:
      $ref: "TS28541_SliceNrm.yaml#/components/schemas/XLThpt"
DLLatencyTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - DLLatency
    targetCondition:
      type: string
      enum:
        - Is_less_than
    targetValueRange:
      type: integer
ULLatencyTarget:
  type: object
  properties:
    targetName:
      type: string
      enum:
        - ULLatency
    targetCondition:
      type: string
      enum:
        - Is_less_than
    targetValueRange:
      type: integer
MaxNumberOfUEsTarget:
  type: object
  properties:
    targetAttribute:
      type: string
```

```

    enum:
      - maxNumberOfUEs
    targetCondition:
      type: string
      enum:
        - Is_less_than
    targetValueRange:
      type: integer
  ActivityFactorTarget:
    type: object
    properties:
      targetAttribute:
        type: string
        enum:
          - activityFactor
      targetCondition:
        type: string
        enum:
          - Is_equal_than
      targetValueRange:
        type: integer
  UESpeedTarget:
    type: object
    properties:
      targetAttribute:
        type: string
        enum:
          - uESpeed
      targetCondition:
        type: string
        enum:
          - Is_less_than
      targetValueRange:
        type: integer

#-----Definition of the concrete ExpectationTarget dataType-----#

#-----Definition of the concrete ObjectTarget dataType-----#
CoverageAreaPolygonContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - CoverageAreaPolygon
    contextCondition:
      type: string
      enum:
        - Is_within_the_range
    contextValueRange:
      type: array
      items:
        $ref: "#/components/schemas/CoverageArea"
CoverageArea:
  type: string
CoverageTACContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - CoverageAreaTac
    contextCondition:
      type: string
      enum:
        - Is_within_the_range
    contextValueRange:
      type: array
      items:
        $ref: "TS28541_NrNrm.yaml#/components/schemas/NrTac"

PLMNContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - PLMN

```

```
contextCondition:
  type: string
  enum:
    - Is_within_the_range
contextValueRange:
  type: array
  items:
    $ref: "TS28541_NrNrm.yaml#/components/schemas/PlmnId"

NRFqBandContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - NRFqBand
    contextCondition:
      type: string
      enum:
        - Is_within_the_range
    contextValueRange:
      type: array
      items:
        type: string

RATContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - RAT
    contextCondition:
      type: string
      enum:
        - Is_within_the_range
    contextValueRange:
      type: array
      items:
        type: string
        enum:
          - UTRAN
          - EUTRAN
          - NR

EdgeIdenficationIdContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - edgeIdentificationId
    contextCondition:
      type: string
      enum:
        - Is_equal_than
    contextValueRange:
      type: array
      items:
        type: string

EdgeIdenficationLocContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - edgeIdentificationTarget
    contextCondition:
      type: string
      enum:
        - Is_equal_than
    contextValueRange:
      type: array
      items:
        type: string

CoverageAreaTAContext:
  type: object
  properties:
    contextAttribute:
```

```

    type: string
    enum:
      - coverageAreaTA
  contextCondition:
    type: string
    enum:
      - Is_within_the_range
  contextValueRange:
    type: array
    items:
      $ref: "#/components/schemas/CoverageAreaTAList"
CoverageAreaTAList:
  type: integer

#-----Definition of the concrete ObjectTarget dataType-----#
#-----Definition of the concrete ExpectationContext dataType-----#
ServiceStartTimeContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - ServiceStartTime
    contextCondition:
      type: string
      enum:
        - Is_equal_than
    contextValueRange:
      type: string
ServiceEndTimeContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - ServiceEndTime
    contextCondition:
      type: string
      enum:
        - Is_equal_than
    contextValueRange:
      type: string
UEMobilityLevelContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - UEMobilityLevel
    contextCondition:
      type: string
      enum:
        - Is_within_the_range
    contextValueRange:
      type: array
      items:
        $ref: "TS28541_SliceNrm.yaml#/components/schemas/MobilityLevel"
ResourceSharingLevelContext:
  type: object
  properties:
    contextAttribute:
      type: string
      enum:
        - ResourceSharingLevel
    contextCondition:
      type: string
      enum:
        - Is_within_the_range
    contextValueRange:
      type: array
      items:
        $ref: "TS28541_SliceNrm.yaml#/components/schemas/SharingLevel"
#-----Definition of the concrete ExpectationContext dataType-----#

#-----Definition of JSON arrays for name-contained IOCs -----#

Intent-Multiple:

```

```
    type: array
    items:
      $ref: '#/components/schemas/Intent-Single'
#-----Definition of JSON arrays for name-contained IOCs -----#

#----- Definitions in TS 28.312 for TS 28.532 -----#
resources-intentNrm:
  oneOf:
    - $ref: '#/components/schemas/Intent-Single'
#----- Definitions in TS 28.312 for TS 28.532 -----#
```

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 ('objectClass',list of
\n[Attribute,Value] of intent IOC)
MnS_Producer -> MnS_Producer: 2. Perform the feasibility check of the intent instance
alt feasibility check result is "Feasible"
MnS_Producer -> MnS_Producer: 3. Create and configure intent MOI
MnS_Producer -> MnS_Consumer: 4. Response for create an intent instance \n (status, DN of intent
MOI, reason)
alt status is OperationSucceeded
  Ref over MnS_Producer, ManagedEntity: 5. 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 feedback\n (DN of intent MOI, fulfillStatus)
end
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
MnS_Consumer -> MnS_Producer: 1.Request to modify an intent\n('objectInstance' of intent MOI,List of
['Attribute','newValue'])
MnS_Producer -> MnS_Producer: 2. Perform the feasibility check of \n the modified intent
alt feasibility check result is "Feasible"
  MnS_Producer -> MnS_Producer: 3.Configure the intent MOI with \n List of ['Attribute' =
'newValue']
  MnS_Producer -> MnS_Consumer: 4.Response for modify an intent\n ('objectInstance' of intent MOI)
  Ref over MnS_Producer, ManagedEntity: 5. modify 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 feedback\n (DN of intent MOI, fulfillStatus)
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.Request 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,scope, filter,\n 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.2 Information model definition for Intent

A.2.1 Relationship UML diagram for intent (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

class "<<proxyClass>> \n ManagedEntity " as ManagedEntity{}
class "<<InformationObjectClass>>\n Intent " as Intent {}
class "<<dataType>>\n IntentExpectation" as IntentExpectation{}
class "<<dataType>>\n IntentContext" as IntentContext{}
class "<<dataType>>\n ExpectationObject" as ExpectationObject{}
class "<<dataType>>\n ObjectContext" as ObjectContext{}
class "<<dataType>>\n ExpectationTarget" as ExpectationTarget{}
class "<<dataType>>\n TargetContext" as TargetContext{}
class "<<dataType>>\n ExpectationContext" as ExpectationContext{}

ManagedEntity "1" *-- "*" Intent
Intent --> IntentReport
Intent "1" *-- "1..*" IntentExpectation
Intent "1" *-- "*" IntentContext

IntentExpectation "1" *-- "*" ExpectationTarget
IntentExpectation "1" *-- "*" ExpectationObject
IntentExpectation "1" *-- "*" ExpectationContext
ExpectationObject "1" *-- "*" ObjectContext
ExpectationTarget "1" *-- "*" TargetContext

note left of ManagedEntity
```


Represents the following IOCs:

Subnetwork

end note

@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 in a 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 detalization, etc.

The intent lifecycle consists of the following phases.

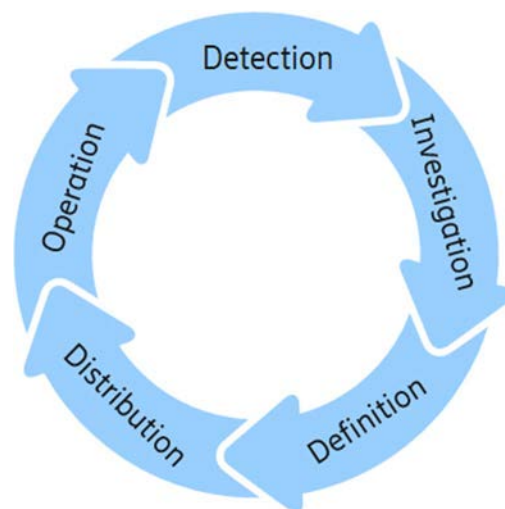


Figure B.1-1: Intent Lifecycle Phases

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 in 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. This has two aspects: first, it needs to find right MnS producer that have the right domain responsibilities and support the intent

expectations the MnS consumer wants to define. MnS producer capability management and detection would be used for this process.

The other aspect of investigation would be finding out if the wanted intent expectations are realistic. This means, if the MnS producer would be able to successfully reach the wanted expectations. This depends on the current resource situation and capabilities of the system and can vary over time. Typically, the feasibility of intent expectations is done through a guided negotiation process between the MnS Producer and MnS Consumer. The MnS Consumer can explore what the handling result of wanted intent expectations would be, what would be the best result the MnS producer can achieve, or what would be the most challenging requirements, the aspiring MnS producer can offer to fulfill.

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 intentExpectation Models

The TM forum defines the structure of an intent as a list of expectations with each expectation containing the requirements goals and constraints to be achieved. The expectation is defined to contain 2 attributes - the imm:target and the imm:params.

The intentExpectation defined in 3GPP (see clause 6.2.1.2.2) contains some attributes which can be mapped to the TM Forum model.

Table C.1 illustrates the mapping between 3GPP Intent Expectation and TM Forum ICM IntentExpectation.

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

| IntentExpectation3GPP Intent Expectation | TM Forum Intent Expectation (IG1253A v1.1.0 [7]) |
|---|---|
| Class Property | Attribute |
| expectationObject.ObjectInstance | icm:target |
| expectationTargets | icm:propertyParams |
| expectationContexts | |
| expectationObject.objectType | icm:deliveryParams |
| expectationObject.ObjectContexts | |

Annex D (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 |

History

| Document history | | |
|-------------------------|-----------|-------------|
| V17.0.1 | July 2022 | Publication |
| | | |
| | | |
| | | |
| | | |