ETSI GS NFV-IFA 050 V5.2.1 (2024-11)

Network Functions Virtualisation (NFV) Release 5; Management and Orchestration; Intent Management Service Interface and Information Model Specification

Disclaimer

The present document has been produced and approved by the Network Functions Virtualisation (NFV) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
RGS/NFV-IFA050ed521

Keywords
IM, IM service interface, modelling

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

Contents

Intellectual Property Rights

Essential patents

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

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

Trademarks

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

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

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Network Functions Virtualisation (NFV).

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.

1 Scope

The present document describes use cases related to intent management and specifies the following aspects:

- the intent management service interface, including interface requirements, service requirements, if necessary operations and their associated information model;
- the information model of intents.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at ETSI docbox.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Functional requirements specification".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GR NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.2] ETSI GR NFV-IFA 041: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Report on Enabling Autonomous Management in NFV-MANO".

[i.3] ISO/IEC 9646-7: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 7: Implementation Conformance Statements".

[i.4] ETSI GS NFV-IFA 027: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Performance Measurements Specification".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GR NFV 003 [i.1] and the following apply:

intent: formal specification of all expectations including requirements, goals, and constraints

intent object: management object whose information (models, properties and/or artifacts) is capable to capture the expectations of the intent

intent object instance: managed object instance that is instantiated at the Intent Handler based on the intent object received from the Intent Owner

Intent Owner: role performed by a management entity when formulating an intent object and using it in intent-driven management

Intent Handler: role performed by a management entity when processing an intent object and being responsible for its fulfilment

NFV intent: intent related to NFV capabilities

NOTE 1: Examples of NFV capabilities are lifecycle management of NS, VNF and virtualised resources, descriptors for NS and VNF, etc.

NOTE 2: Some of the definitions are adapted from ETSI GR NFV-IFA 041 [i.2].

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR NFV 003 [i.1] and the following apply:

IM Intent Management

4 Overview

4.1 Introduction

The Intent Management (IM) function is identified in ETSI GR NFV-IFA 041 [i.2]. Intents are regard as knowledge objects and managed by IM function. The functional requirements for the intent management specified in ETSI GS NFV-IFA 010 [1] shall apply. Hence, an IM function plays the producer role, and exposes the operations and/or services to a consumer through the interface named IM-1.

4.2 Relation to other NFV group specifications

The present document relates to other ETSI NFV deliverables as follows:

- ETSI GS NFV-IFA 010 [1]: specifies functional requirements for NFV management and orchestration, and general guidelines and requirements for NFV management and orchestration interface design, including intent management function. The present document specifies the corresponding service interface and its associated information model.

- ETSI GR NFV-IFA 041 [i.2]: evaluates possible enhancement to the framework of NFV-MANO to improve its automation capabilities and introduce autonomous management mechanisms, including high-level use cases, functional key issue analysis and architectural options of intent management function. The present document specifies the corresponding service interface and its associated information model.

4.3 Conventions

The following notations, defined in ISO/IEC 9646-7 [i.3], are used for the qualifier column of interface information elements:

- M mandatory - the capability is required to be supported.
- O optional - the capability may be supported or not.
- CM conditional mandatory - the capability is required to be supported and is conditional on the support of some condition. This condition shall be specified in the Description column.
- CO conditional optional - the capability may be supported or not and is conditional on the support of some condition. This condition shall be specified in the Description column.

The following notation is used for parameters that represent identifiers, and for attributes that represent identifiers in information elements and notifications:

- If parameters are referring to an identifier of an actual object, their type is "Identifier".
- If an object (information element or notification) contains an attribute that identifies the object, the type of that attribute is "Identifier" and the description states that the attribute is the identifier of that particular notification or information element.

EXAMPLE 1: Identifier "intentId" of the "Intent information element" has type "Identifier" and description "Identifier of this intent information element".

- If an object (information element or notification) contains an attribute that references another object or objects defined in an ETSI GS NFV, the type of the attribute is "Identifier", followed by the list of objects it references.

EXAMPLE 2: "Identifier (Reference to Vnfc)" or "Identifier (Reference to Vnfc, VirtualLink or VirtualStorage)".

- If the type of a parameter or attribute has been marked as "Not specified" in the "Content" column, this means that its specification is part of the protocol design/data model design.

4.4 Intent Management (IM)

Intent Management function is a set of capabilities that enable a simplified way for OSS/BSS to consume NFV-MANO services, by formulating only their NFV requirements and constraints without requiring any awareness of which operations are necessary based on NFV intents and how the operations translated from NFV intents are executed by NFV-MANO. In the present document, the scoping of the NFV intents is focused on Network Services.

There are two main roles that are involved in the NFV intent management scenarios: the Intent Owner (e.g. OSS/BSS) that formulates the NFV intent object, and the Intent Handler which processes and fulfills the NFV intent management request.

In the present document Intent Handler and Intent Owner are roles and not representations of corresponding management functions. The Intent Handler is a role of the IM which is a management function. IM could be part for example of NFVO, see ETSI GR NFV-IFA 041 [i.2], clause 7.2.1.

In principle as a management entity IM can perform the Intent Owner role, the Intent Handler role, or both. For example, an IM function can compile intent objects which are processed by another IM management entity. This special case will not be further analysed in the present document version.

The Intent Owner supports formulating the intent object.

The Intent Handler supports the NFV intent management requests handling as specified in clause 6.3.

The Intent Handler shall be able to process the received NFV intent management requests, by performing:

- Translation, analysis and decision making, where:
  - An analysis of the high-level requirements and constraints provided in the NFV intent management request, with the goal of mapping them into the appropriate NFV-MANO constructs and data models. For example, this leads to determining the NSD which has the right attributes to match the NFV intent requirements, identifying the right input parameters for NS LCM operations, or the derived runtime information from the NFV intent requirements which can be used to find a matching NS instance.
  - The Intent Handler uses various information to be able to determine:
    - If it can reuse an existing NS instance or a new NS instance needs to be created.
    - If it can reuse existing VNF instances to compose the NS instance, or it can create its own VNF instances to match a received NFV intent management request.
    - If there are sufficient virtualised resources available, with the appropriate characteristics, so that IM can trigger the corresponding creation of a new NS instance or update to an existing NS instance.
- Assessment and maintenance, where:
  - Intent Handler is using any relevant information obtained to assess the NFV intent object instance fulfillment status and to make decisions on any further actions to maintain the desired requirements and/or constraints specified in the NFV intent object:
    - Such decisions can be made continuously throughout the lifetime of an intent object instance, as the Intent Handler shall ensure the expectations of an NFV intent object instance are always fulfilled until the NFV intent object instance is deleted. Since the status of the Network Service changes over time, when the Intent Handler detects, based on the observation data, that the running NS instances it has used for the fulfillment of an NFV intent instance are no longer matching the expectations of that intent instance, then the Intent Handler shall report to the Intent Owner about the incident and the latter may decide to make requests of updates to intent object instances.
  - Intent reporting back to the Intent Owner:
    - Intent Handler provides intent reports back to the Intent Owner (e.g. OSS/BSS) to inform about the status of the NFV intent object instance. The intent reports are assembled by the Intent Handler according to the reporting requirements specified by the Intent Owner as part of the Intent object request. These reporting requirements can be based on regular timing, event-triggering, or other types of conditions. An Intent Owner can control the reporting by setting and/or changing the NFV intent object instance reporting criteria as needed.

5 Use cases related to NFV-MANO intent management (informative)

5.1 Overview

Intent based management is applicable to NFV-MANO, where the OSS/BSS acts as the Intent Owner to express its intent by delivering intent expectation associated to network service management. The IM function acts as the Intent Handler to translate the received expectation of the intent object to corresponding NS operation(s). The Intent Management function interacts with NFV-MANO to execute the derived intent operations, and monitor intent fulfilment status to evaluate whether the intent expectation is fulfilled. It also notifies the fulfilment information via intent report to the OSS/BSS. The following use cases related to intent-based network service management are included in this clause:

- Intent object instance Creation.
- Intent object instance Update.
- Intent object instance Query.

- Intent object instance Deletion.
- Intent object Negotiation.

NOTE: Other relevant use cases (e.g. intent feasibility assessment) are not pursued in the present document, but may be subject to future releases of the present document.

In all the use cases described in the following subclauses the role of the Intent Owner is performed by OSS/BSS and the role of Intent Handler is performed by the IM function.

5.2 Use Case #1: Intent Object Instance Creation

5.2.1 Overview

In this use case, the Intent Owner expresses a new intent by delivering NS expectation and constraints information. For example, NS expectations can be related to functional requirements (e.g. which VNFs are needed), performance requirements (e.g. the minimum incoming/outgoing data rate of a certain SAP), geographical location, isolation requirements (e.g. whether or not it is allowed to share any resources with other NSs), special security requirements (e.g. use of secure enclaves). The IM (i.e. Intent Handler) translates intent to corresponding NS operation(s) (e.g. instantiate a new NS, update an existing NS or terminate an existing NS) to fulfil the intent requirements.

5.2.2 Actors and roles

Table 5.2.2-1 describes the actors and roles.

Table 5.2.2-1: Actors and roles for intent object instance creation

#	Role	Description
1	Intent Owner	Determines an intent identifying the requirements, constraints and characteristics it needs for the NS functionality and captures them in the NFV intent object expectations.
2	IM	Interprets the intent object and maps it to corresponding NS operation(s) (e.g. instantiate a new NS or update an existing NS, etc.), transfers the corresponding NS operation(s) to be executed by NFVO and continuously verifies that the intent remains fulfilled).
3	NFVO	Executes requests received from IM.

5.2.3 Pre-conditions

Table 5.2.3-1 describes the use case pre-conditions.

Table 5.2.3-1: Pre-conditions for intent object instance creation

#	Pre-condition	Additional description
1	The IM is up and running.	
2	The IM has synchronized the NS instances' states as well as the NSD information (e.g. the constituent VNF or nested NS description of an NS implemented by using the NSD) with the NFVO.	

5.2.4 Post-conditions

Table 5.2.4-1 describes the use case post-conditions.

Table 5.2.4-1: Post-conditions for intent object instance creation

#	Post-condition	Additional description
1	The NS instance(s) operations that can fulfil the NFV intent are executed and can satisfy the requirements, constraints and characteristics expectations expressed in the NFV Intent object.	

5.2.5 Flow description

Table 5.2.5-1 describes the use case flow for intent object instance creation.

Table 5.2.5-1: Flow for Intent object instance creation

#	Actor/Role	Action/Description
Begins when	Intent Owner	The Intent Owner determines an intent object which contains the expectations for desired NS(s). For example, functional requirements (e.g. which type of VNFs are needed), performance requirements (e.g. the minimum incoming/outgoing data rate of a certain SAP), geographical location, isolation requirements (e.g. whether it is allowed to share any resources with other NSs), special security requirements (e.g. use of secure enclaves).
Step 1	Intent Owner -> IM	The Intent Owner sends the desired NFV intent objects to the IM.
Step 2	IM-> NFVO	The IM processes the received intent expectations. Based on the state of NS instance(s) and/or the NSD information, the IM translates intent expectations into appropriate NS operations towards the NFVO. Such NS operations can be any of the following: In case that there is no available NS instance that fulfil NS expectations described by the intent: • NS identifier creation and NS instantiation operation according to a selected NSD file, including the chosen flavourId and NS instantiation levels, geographic location, etc., where IM selects the on-boarded NSD, matching the desired NS functionality. In case that there is available NS instance(s) that fulfil NS expectations described by the intent, and the NS state is "INSTANTIATED": • NS scale operation, if IM determined that an existing NS can be reused or shared, based on NSD file, including the selected scaling related attributes, e.g. scaleNsToLevelData; or • NS update operation, with the updated attributes; or • NS terminate operation, to terminate an existing NS; or • NS feasibility check operation, with or without resource reservation, depending on the decisions taken by the IM during the intent processing step. The IM can also make PM, FM, and configuration management requests for a given NS instance, if needed.
Step 3	NFVO-> IM	The NFVO returns the results of the NS operations to the IM. If the NS operations were successful, continue to the next step. Optionally, if it can't be achieved, the Intent Owner should be informed of the expected performance metric values in the intent negotiation process (as described in clause 5.6). If the Intent Owner accepts the modified performance metric values, proceed to the next step.
Step 4	IM<->NFVO	The IM starts to monitor the status, performance and other events related to the NS instance(s) associated with the NFV intent. As example, the IM subscribes to the NFVO for: • the NS LCM occurrence events for the NS instance; • NS instance metric data that the IM has derived from the intent expectation.
Step 5	IM-> Intent Owner	According to the intent reporting requirements it received from the Intent Owner, the IM assembles the intent report whenever the criteria and requirements for intent reporting are met. The IM sends intent reports to the Intent Owner about the intent fulfillment status, all relevant events in the NS LCM or in NS PM, FM, that the IM has derived from the intent reporting expectations it received from the Intent Owner.
Step 6	Intent Owner<-> IM <-> NFVO	The IM continues: • to perform any required closed loop actions to monitor/observe, analyse/orient, decide and actuate, with the goal to maintain the NS instance(s) associated to that NFV intent at the expected status and performance levels; and • to send intent reports to the Intent Owner, according to the reporting requirements expressed by the Intent Owner.

5.3 Use Case # 2: Intent Object Instance update

5.3.1 Overview

This use case describes a scenario where an Intent Owner requests to update an existing intent object instance. For example, the request contains an expectation for updating the performance of an NS instance. In this use case, the Intent Owner updates its intent object expectation for containing the new performance goals of the NS (e.g. increase/decrease the incoming data rate of a specific SAP). Based on the intent expectation, the IM (i.e. Intent Handler) translates the updated intent expectation to a scaling NS operation to fulfil the updated intent requirements.

5.3.2 Actors and roles

Table 5.3.2-1 describes the actors and roles.

Table 5.3.2-1: Actors and roles for Intent object instance update

#	Role	Description
1	Intent Owner	Formulates the requirements and constraints in the intent object (the "what") without having any knowledge about the "how" the intent is executed.
2	IM	Processes the intent object and determines which specific NFV-MANO operations are to be requested to the NFVO. The IM responses to the Intent Owner about the NFV intent fulfillment information, based on the given intent requirements.
3	NFVO	Executes the requests received from IM.

5.3.3 Pre-conditions

Table 5.3.3-1 describes the use case pre-conditions.

Table 5.3.3-1: Pre-conditions for Intent object instance update

#	Pre-condition	Additional description
1	The IM is up and running.	
2	An existing intent object instance is created. The target NS is running and its performance data is collected by NFV-MANO.	
3	The IM updates the subscription to the NS metric data (e.g. create some new PM Job) of the corresponding intent object instance to NFV-MANO according to the new intent expectation.	

5.3.4 Post-conditions

Table 5.3.4-1 describes the use case post-conditions.

Table 5.3.4-1: Post-conditions for Intent object instance update

#	Post-condition	Additional description
1	The target intent object instance is updated and the related NS is scaled to fulfil the requirements as described in the updated intent instance.	

5.3.5 Flow description

Table 5.3.5-1 describes the use case flow for Intent object containing an expectation for updating an NS performance.

Table 5.3.5-1: Flow for Intent object instance update

#	Actor/Role	Action/Description
Begins when	Intent Owner	The Intent Owner determines to update an existing intent object instance to provide the new performance requirement of a NS. For example, as described in ETSI GS NFV-IFA 027 [i.4], the updated intent object expectation might indicate that the upper limit of incoming data rate of a specific SAP of a network service instance needs to be increased to fulfil its service requirements.
Step 1	Intent Owner -> IM	The Intent Owner sends the update intent request to provide the new performance requirement of an NS to the IM. Identification about the NS instance whose performance is expected to be updated is also included in the request.
Step 2	IM<->NFVO	The IM interacts with the NFVO to determine whether the new expectation of updating the performance requirement of the NS can be achieved. If it can be achieved, continue to the next step. If it can't be achieved, the Intent Owner is informed of the expected performance metric values. If the Intent Owner accepts the modified performance metric values, proceed to the next step. Otherwise, the update of the intent object instance fails.
Step 3	IM-> NFVO	The IM processes the update intent object request and translates it to scale NS operation to be executed by NFVO for one or more requested Intent object instances for which the request applies. Based on the updated expectation of the intent and the current status of the NS instances and the NSD information of the NS, the IM selects the suitable type for scaling the NS, e.g. scaling in/out a particular VNF or scaling in/out the NS with a particular aspect.
Step 4	NFVO->IM	The NFVO performs scale NS operation determined in step 3. The NFVO collects the performance data of the SAP after scale NS operation is finished and notifies the IM about the intent object instance update result.
Step 5	IM-> Intent Owner	The IM sends response to the Intent Owner about the intent object instance update result.
Ends when	IM-> Intent Owner	The IM continues to receive the NS metric data of the corresponding updated intent object instance from NFVO according to the updated intent expectation, and analyses whether the relevant metric data meets the updated intent expectation. The IM feeds back the intent fulfillment information to the Intent Owner. See note.
NOTE: The intent fulfillment information can be either actively reported to the Intent Owner by the IM, or queried from the Intent Owner to the IM.		

5.4 Use Case # 3: Intent Object Instance Deletion

5.4.1 Overview

In this use case, the Intent Owner requests deleting an existing intent object instance. The request contains the identifier of the intent object instance to be deleted.

5.4.2 Actors and roles

Table 5.4.2-1 describes the actors and roles.

Table 5.4.2-1: Actors and roles for intent object instance deletion

#	Role	Description
1	Intent Owner	Determines the existing intent object instance to be deleted.
2	IM	Interprets the intent deletion request and maps it to corresponding operation(s), transfers the corresponding operation(s) to be executed by NFV-MANO.
3	NFV-MANO	Executes corresponding operation(s), e.g. release the resources or corresponding configurations of NS instances dedicated to the fulfillment of the intent to be deleted.

5.4.3 Pre-conditions

Table 5.4.3-1 describes the use case pre-conditions.

Table 5.4.3-1: Pre-conditions for intent object instance deletion

#	Pre-condition	Additional description
1	The IM is up and running.	
2	An existing intent object instance has been created.	

5.4.4 Post-conditions

Table 5.4.4-1 describes the use case post-conditions.

Table 5.4.4-1: Post-conditions for intent object instance deletion

#	Post-condition	Additional description
1	The intent object instance is deleted and the corresponding dedicated resources and/or configurations are freed, and IM returns the response to Intent Owner.	

5.4.5 Flow description

Table 5.4.5-1 describes the use case flow for intent instance deletion.

Table 5.4.5-1: Flow for intent object instance deletion

#	Actor/Role	Action/Description
Begins when	Intent Owner -> IM	The Intent Owner sends to the IM a request to delete an intent object instance. The request contains the identifier of the intent object instance to be deleted.
Step 1	IM	The IM translates the received request into the corresponding NFV-MANO operation(s) with the corresponding resources (e.g. NS instances and their configurations).
Step 2	IM<-> NFV-MANO	The IM requests NFV-MANO the execution of the corresponding operations to the intent related targeted entity (such as NS instance or VNF instance), and then the IM deletes the intent object instance, and asks NFV-MANO to release the resources or corresponding configurations dedicated to the fulfillment of the intent to be deleted.
Ends when	IM-> Intent Owner	The IM returns a successful deletion response to the Intent Owner.

5.5 Use Case # 4: Intent Object Instance Query

5.5.1 Overview

This use case describes a scenario where an Intent Owner queries the IM about information (e.g. execution result, status, etc.) related to the existing intent object instance(s) which fits the query criteria. By querying existing intent object instance(s), the Intent Owner expects, as a response to the query, up-to-date information. Additionally, after the intent creation, intent report can be provided asynchronously according to the reporting expectations.

5.5.2 Actors and roles

Table 5.5.2-1 describes the actors and roles.

Table 5.5.2-1: Actors and roles for Intent object instance query

#	Role	Description
1	Intent Owner	Sends the query request to IM asking for the information related to existing intent object instance(s).
2	IM	Manages the intent object instances, and sends the information to Intent Owner based on the query request.

5.5.3 Pre-conditions

Table 5.5.3-1 describes the use case pre-conditions.

Table 5.5.3-1: Pre-conditions for Intent object instance query

#	Pre-condition	Additional description
1	The IM is up and running.	
2	Existing intent instances have been created.	

5.5.4 Post-conditions

Table 5.5.4-1 describes the use case post-conditions.

Table 5.5.4-1: Post-conditions for Intent object instance query

#	Post-condition	Additional description
1	The requested information of the existing intent object instance(s) or the latest content of an intent object instance is returned to the Intent Owner.	

5.5.5 Flow description

Table 5.5.5-1 describes the use case flow for Intent object instance query.

Table 5.5.5-1: Flow for Intent object instance query

#	Actor/Role	Action/Description
Begins when	Intent Owner -> IM	The Intent Owner sends a query request to IM asking for the information (e.g. execution result, status, etc.) related to existing intent object instance(s) which fit the query criteria.
Step 1	IM	IM receives the query request from the Intent Owner and checks the existing intent object instances against the query criteria.
Step 2	IM-> Intent Owner	IM sends response to Intent Owner with the information of the existing intent object instance(s) based on the query request.
Ends when	Intent Owner	The Intent Owner receives the requested information.

5.6 Use Case # 5: Intent Negotiation

5.6.1 Overview

In this use case, the Intent Owner negotiates with the IM on achievable intent object parameters. The intent object parameters here can be parameters related to performance requirements (e.g. the minimal incoming/outgoing data rate of a certain SAP), parameters related to geographical location, parameters related to isolation requirements (e.g. whether or not it is allowed to share any resources with other NSs), or parameters related to special security requirements (e.g. use of secure enclaves). IM (i.e. Intent Handler) will analyse, translate, and evaluate the intent object parameters proposed by the Intent Owner, determine the values of the intent parameters that can be achieved, and return the updated intent object to the Intent Owner based on the achievable intent object parameter confirmation results.

NOTE: This use case only captures the scenarios where Intent Owner initiates the negotiation, and not those where Intent Handler initiates the negotiation.

5.6.2 Actors and roles

Table 5.6.2-1 describes the actors and roles.

Table 5.6.2-1: Actors and roles for intent object negotiation

#	Role	Description
1	Intent Owner	Determines an intent object identifying the requirements, constraints and characteristics it needs for NS functionality and captures them in the parameters related to NFV intent, and initiate a negotiation request for the above parameters.
2	IM	Interprets the intent object and maps it to corresponding NS operation(s) (e.g. instantiate a new NS or update an existing NS, etc.), and confirm the feasibility of the corresponding NS operation through NFV-MANO. If it is unfeasible, provide suggestions for modifying the intent parameters based on the feasible NS operation confirmed by NFV-MANO.
3	NFV-MANO	Confirms the feasibility of the corresponding NS-related operation(s) (e.g. NS LCM, NS PM, NS FM, etc.).

5.6.3 Pre-conditions

Table 5.6.3-1 describes the use case pre-conditions.

Table 5.6.3-1: Pre-conditions for intent object negotiation

#	Pre-condition	Additional description
1	The IM is up and running.	

5.6.4 Post-conditions

Table 5.6.4-1 describes the use case post-conditions.

Table 5.6.4-1: Post-conditions for intent object negotiation

#	Post-condition	Additional description
1	The result of negotiation is completed and shared between the Intent Handler and the Intent Owner.	

5.6.5 Flow description

Table 5.6.5-1 describes the use case flow for intent object negotiation.

Table 5.6.5-1: Flow for intent object negotiation

#	Actor/Role	Action/Description
Begins when	Intent Owner	The Intent Owner determines an intent object which contains the expectations for desired NS(s). For example, performance requirements (e.g. the minimal incoming/outgoing data rate of a certain SAP), geographical location, isolation requirements (e.g. whether or not it is allowed to share any resources with other NS(s), special security requirements (e.g. use of secure enclaves).
Step 1	Intent Owner -> IM	The Intent Owner sends the desired NFV intent object to be negotiated towards the IM.
Step 2	IM <-> NFV-MANO	After receiving the intent object from the Intent Owner, that is to be evaluated, IM interprets the requirements and map them to corresponding operation(s), whose feasibility is checked through NFV-MANO.
Step 3	NFV-MANO → IM	NFV-MANO evaluates the best values that can be achieved for the negotiated requirement or expectation and communicates the result to IM. If for any reason, it fails to provide the result intent object, relevant reasons of negotiation failure are communicated to IM.
Step 4 (M)	IM-> Intent Owner	After receiving the result from NFV-MANO for the feasibility check of the corresponding operations for the desired intent object which has been evaluated, IM would share either the intent object with the confirmed feasible parameters or the reason for the negotiation failure to the Intent Owner.

#	Actor/Role	Action/Description
Step 5 (M)	Intent Owner <-> IM	The Intent Owner after receiving the proposals on the feasible values that can be achieved for the negotiation request, if the proposal does not meet its requirements then it would start with a new variant of intent and ask for renewed proposal, by returning to step 1.
Ends when	IM-> Intent Owner	If the Intent Owner and IM determine that the intent negotiation result is completed, or the Intent Owner abandons the negotiation, the intent negotiation process ends.

6 Requirements of intent management service interface

6.1 Introduction

This clause specifies the set of requirements applicable to intent management service interface.

6.2 Service requirements of intent management

Table 6.2-1 specifies requirements applicable to the services provided by intent management.

Table 6.2-1: Intent management service requirements

Identifier	Requirement
ImSvc.001	The Intent Management shall provide an Intent Management service.

6.3 Interface requirements of intent management service

Table 6.3-1 specifies the interface requirements of operations to be supported by the Intent management service interface.

Table 6.3-1: Intent management service interface requirements

Identifier	Requirement
ImInf.001	Intent management service interface produced by the Intent Management shall support creating an intent object instance.
ImInf.002	Intent management service interface produced by the Intent Management shall support updating an existing intent object instance.
ImInf.003	Intent management service interface produced by the Intent Management shall support querying the information of an existing intent object instance. See note.
ImInf.004	Intent management service interface produced by the Intent Management shall support deleting an existing intent object instance.
ImInf.005	Intent management service interface produced by the Intent Management shall support reporting the information of an existing intent object instance.
ImInf.006	Intent management service interface produced by the Intent Management shall support providing notifications about events related to intent processing.
ImInf.007	Intent management service interface produced by the Intent Management shall support managing subscriptions to notifications about events related to intent processing.
ImMgtInf.008	Intent management service interface produced by the Intent Management shall enable negotiation of intent object between Intent Owner and Intent Handler regarding the expectations that can be delivered for the intent.
NOTE: Reporting in ImMgtInf.005 is related to notifications sent by the IM to authorized consumers (asynchronous mode), while ImMgtInf.003 is about synchronous replies to requests about information of an existing intent object instance.	

7 Intent management service interface

7.1 Description

This clause provides an overview of the intent management service interface, which is used for intent information exchange between the Intent Owner and the IM. The Intent management service interface supports the following operations:

- Create Intent Object Instance Operation.
- Update Intent Object Instance Operation.
- Delete Intent Object Instance Operation.
- Query Intent Object Instance Operation.
- Subscribe Intent Object Instance Operation.
- Unsubscribe Intent Object Instance Operation.
- Negotiate Intent Object Operation.
- Notify Intent Object Instance Operation.

7.2 Create Intent Object Instance Operation

7.2.1 Description

This operation enables the Intent Owner to request creation of a new intent object instance from the Intent Handler.

Table 7.2.1-1 lists the information flow exchanged between Intent Owner and Intent Handler.

Table 7.2.1-1: Create intent object instance operation

Message	Requirement	Direction
CreateIntentRequest	Mandatory	Intent Owner -> Intent Handler
CreateIntentResponse	Mandatory	Intent Handler -> Intent Owner

7.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.2-1.

Table 7.2.2-1: Create intent object instance operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
intentObject	M	1	IntentObject	Describes the expectations, requirements and constraints to be fulfilled by the intent object instance to be created.

7.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.3-1.

Table 7.2.3-1: Create intent object instance operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
intentObjectInstanceId	M	0..1	Identifier	Identifier of the newly created intent object instance.

7.2.4 Operation results

If the operation is successful, the Intent Handler creates the corresponding intent object instance and returns the intentObjectInstanceId.

If the operation fails, the corresponding error information is returned with possible reasons.

7.3 Delete Intent Object Instance Operation

7.3.1 Description

This operation enables the Intent Owner to delete an existing intent object instance from the Intent Handler (i.e. IM).

Table 7.3.1-1 lists the information flow exchanged between Intent Owner and Intent Handler.

Table 7.3.1-1: Delete intent object instance operation

Message	Requirement	Direction
DeleteIntentRequest	Mandatory	Intent Owner -> Intent Handler
DeleteIntentResponse	Mandatory	Intent Handler -> Intent Owner

7.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.2-1.

Table 7.3.2-1: Delete intent object instance operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
intentObjectInstanceId	M	1	Identifier	Identifier of the intent object instance to be deleted.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to delete multiple intent object instances in one request, or as a series of requests that delete one intent object instance at a time.				

7.3.3 Output parameters

None.

7.3.4 Operation results

If the operation succeeds, the Intent Handler (IM) deletes the corresponding intent object instance.

If the operation fails, the corresponding error information is returned with possible reasons.

7.4 Update Intent Object Instance Operation

7.4.1 Description

This operation enables the Intent Owner to request the Intent Handler (i.e. IM) for updating an existing intent object instance.

Table 7.4.1-1 lists the information flow exchanged between Intent Owner (e.g. OSS/BSS) and Intent Handler.

Table 7.4.1-1: Update intent object instance operation

Message	Requirement	Direction
UpdateIntentRequest	Mandatory	Intent Owner -> Intent Handler
UpdateIntentResponse	Mandatory	Intent Handler -> Intent Owner

7.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.2-1.

Table 7.4.2-1: Update intent object instance operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
intentObject	M	1	IntentObject	Intent expression with intended expectations, requirements and constraints.
intentObjectInstanceId	M	1	Identifier	Identifier of the intent object instance to be updated.

7.4.3 Output parameters

None.

7.4.4 Operation results

If the operation is successful, the Intent Handler updates the corresponding intent object instance.

If the operation fails, the corresponding error information is returned with possible reasons.

7.5 Query Intent Object Instance Operation

7.5.1 Description

This operation enables the Intent Owner to query about the information related to the existing intent object instance(s) from the Intent Handler (i.e. IM).

Table 7.5.1-1 lists the information flow exchanged between Intent Owner and Intent Handler.

Table 7.5.1-1: Query intent object instance operation

Message	Requirement	Direction
QueryIntentRequest	Mandatory	Intent Owner -> Intent Handler
QueryIntentResponse	Mandatory	Intent Handler -> Intent Owner

7.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.2-1.

Table 7.5.2-1: Query intent object instance operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
intentFilter	M	0..1	Not Specified	Filtering criteria to select one or a set of intent object instances. If absent, the information related to all the intent object instances in operation which were created by the Intent Owner are returned.
contentFilter	M	0..1	Not Specified	Filtering criteria to select one or a set of information elements from matched intent object instances. If absent, all the contents recorded for matching intent object instances are returned.

7.5.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.5.3-1.

Table 7.5.3-1: Query intent object instance operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not Specified	Information related to the intent object instance(s) matching the query.

7.5.4 Operation results

If the operation succeeds, the information related to the intent object instance(s) that matches the filter (if present) should be returned.

If the operation fails, the corresponding error information is returned with possible reasons.

7.6 Subscribe Intent Object Instance Operation

7.6.1 Description

This operation enables the authorized entities to subscribe to notifications sent by the Intent Handler (i.e. IM) for events related to one or several intent object instances in operation matching the filter criteria.

Table 7.6.1-1 lists the information flow exchanged between subscribing entity and Intent Handler.

NOTE: No restriction is imposed on which is the entity which is subscribing to NFV intent object instances related notifications (e.g. OSS/BSS). The entity sending notifications is the Intent Handler.

Table 7.6.1-1: Subscribe intent object instance operation

Message	Requirement	Direction
SubscribeIntentRequest	Mandatory	Subscribing entity -> Intent Handler
SubscribeIntentResponse	Mandatory	Intent Handler -> Subscribing entity

7.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.6.2-1.

Table 7.6.2-1: Subscribe intent object instance operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Not Specified	Input filter for selecting specific intent object instances and specific content (e.g. the intent expectation, fulfilment information, etc.) of interest to the subscriber to be included in the notification to be returned.
intentObjectInstanceId	M	1	Identifier	Identifier of the intent object instance for which information are to be reported.

7.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.6.3-1.

Table 7.6.3-1: Subscribe intent object instance operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	0..1	Identifier	Identifier of the subscription realized

7.6.4 Operation results

If the operation succeeds, the subscription ID is returned. For a specific subscription with content filter, only the information that matches the filter is delivered to intent subscriber.

If the operation fails, the corresponding error information is returned with possible reasons.

7.7 Terminate Intent Object Instance Subscription Operation

7.7.1 Description

This operation enables authorized entities to terminate a particular intent subscription.

Table 7.7.1-1 lists the information flow exchanged between subscribing entity and Intent Handler.

Table 7.7.1-1: Terminate intent object instance subscription operation

Message	Requirement	Direction
TerminateSubscribeIntentRequest	Mandatory	Subscribing entity -> Intent Handler
TerminateSubscribeIntentResponse	Mandatory	Intent Handler -> Subscribing entity

7.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.2-1.

Table 7.7.2-1: Terminate intent object instance subscription operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription to be terminated.

7.7.3 Output parameters

None.

7.7.4 Operation results

If the operation succeeds, the identified subscription does not exist anymore, and no intent notification related to that subscription can any longer be sent.

If the operation fails, the corresponding error information is returned with possible reasons.

7.8 Negotiate Intent Object Operation

7.8.1 Description

This operation enables negotiation of intent object between Intent Owner and Intent Handler (i.e. IM) for the intent elements that can be delivered for the intent. Negotiation is about intent object feasibility checking and confirmation of feasible parameters.

NOTE: This operation only captures the scenarios where Intent Owner initiates the negotiation before intent creation, and not those where Intent Handler initiates the negotiation.

Table 7.8.1-1 lists the information flow exchanged between Intent Owner and Intent Handler.

Table 7.8.1-1: Negotiate intent object operation

Message	Requirement	Direction
NegotiateIntentRequest	Mandatory	Intent Owner -> Intent Handler
NegotiateIntentResponse	Mandatory	Intent Handler -> Intent Owner

7.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.2-1.

Table 7.8.2-1: Negotiate intent object operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
intentObjectIn	M	1	IntentObject	Information about the intent object to be negotiated.

7.8.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.3-1.

Table 7.8.3-1: Negotiate intent object operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
intentObjectOut	M	1	IntentObject	Information about the intent object after negotiation.

7.8.4 Operation results

If the operation is successful, the result of negotiation is shared by Intent Handler to the Intent Owner and agreed by the Intent Owner. Negotiation operation includes for example feasibility checking, evaluation of desired and confirmed feasible parameters.

If the operation fails, the corresponding error information is returned with possible reasons.

7.9 Notify Intent Object Instance operation

7.9.1 Description

This operation distributes notifications to a subscribing entity or the Intent Owner. It is a one-way operation issued by the Intent Handler that cannot be invoked as an operation by the consumer (i.e. the subscribing entity or Intent Owner).

In order to receive notifications, the subscribing entity shall have a subscription. By means of report expectations declared in the intent object, the Intent Owner is enabled to express specific requirements for intent reporting.

Table 7.9.1-1 lists the information flow exchanged between Intent Handler and the subscribing entity or Intent Owner.

Table 7.9.1-1: Notify intent object instance operation

Message	Requirement	Direction
Notify	Mandatory	Intent Handler -> Subscribing entity/Intent Owner

The following notifications can be notified/sent by this operation:

- IntentReportNotification (see clause 8.2.10).

8 Information elements exchanged

8.1 Introduction

The clause below defines information elements related to intent management for NFV-MANO.

Figure 8.1-1 gives the UML class diagram for information elements related to intent management for NFV-MANO.

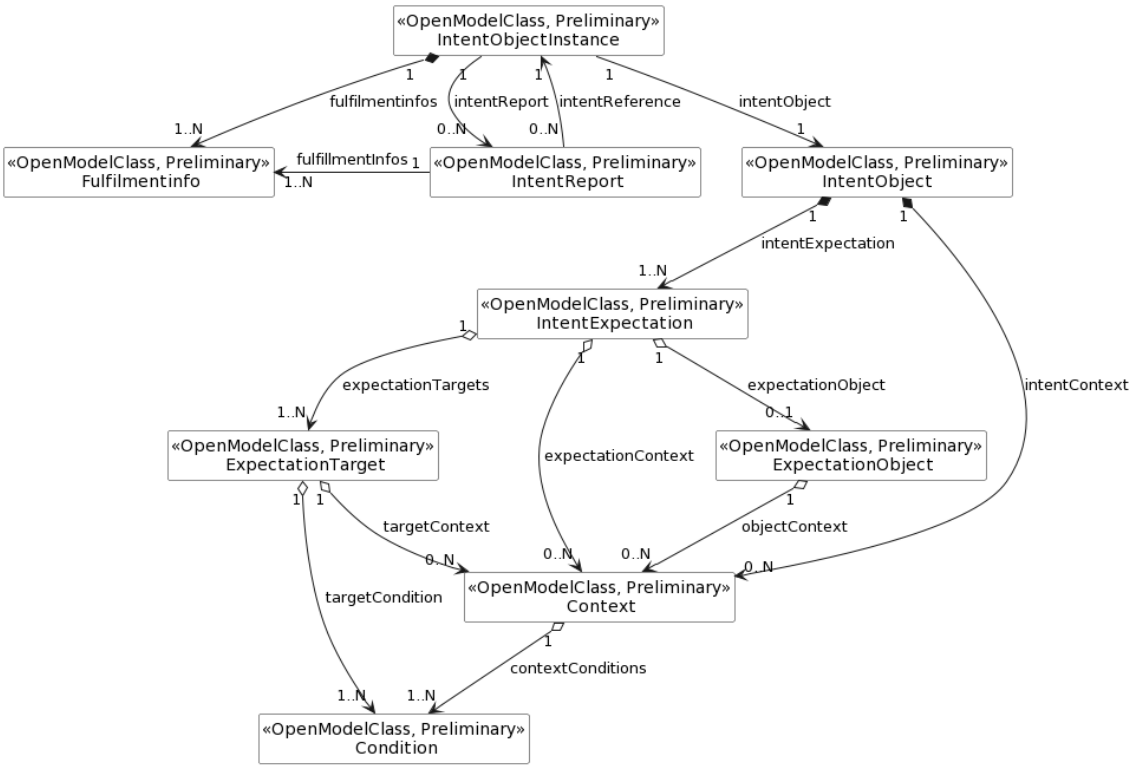

Figure 8.1-1: Intent Information Modeling Diagram

8.2 Information elements of intent management for NFV-MANO

8.2.1 IntentExpectation information element

8.2.1.1 Description

An IntentExpectation describes the expectation of an intent including requirements, goals and contexts (including constraints and filter information for the expectation) given to the Intent Handler.

8.2.1.2 Attributes

The attributes of the IntentExpectation information element shall follow the indications provided in table 8.2.1.2-1.

Table 8.2.1.2-1: Attributes of the IntentExpectation information element

Attribute	Qualifier	Cardinality	Content	Description
expectationId	M	1	Identifier	Identifier of this IntentExpectation information element.
expectationName	M	1	string	Human readable name of this IntentExpectation.
expectationType	M	1	Not specified	It describes the type of the intentExpectation. For example, the possible types are: • Delivery Expectation type. • Ensure Expectation type. • Property Expectation type. • Report Expectation type.
expectationObject	M	0..1	ExpectationObject	It describes the objects managed by NFV-MANO that subject to the IntentExpectation.
expectationTargets	M	1..N	ExpectationTarget	It specifies the targets for each of the expectation type defined in the Intent so for the system to achieve the desired or wanted state from its present state. E.g. a system expected to reach a goal based on a metrics with a target value or for a system to reach a target configuration, etc.
expectationContexts	M	0..N	Context	It describes the constraints and conditions that apply for a specific intentExpectation.
NOTE: When an intentObject contains multiple intentExpectations, the intentContexts at the intent level takes effect on the entire intentObject, and the expectationContexts at the IntentExpectation level takes effect on only one expectation. It is the Intent Owner's responsibility to avoid conflict between intentContexts and expectationContexts before issuing the corresponding intent management request.				

8.2.2 IntentObject information element

8.2.2.1 Description

An IntentObject presents the properties of the management object whose information is capable to capture the requirements and context of the intent object.

8.2.2.2 Attributes

The attributes of the IntentObject information element shall follow the indications provided in table 8.2.2.2-1.

Table 8.2.2.2-1: Attributes of the IntentObject information element

Attribute	Qualifier	Cardinality	Content	Description
intentId	M	1	Identifier	Unique identifier of this intent object. Assigned by the Intent Owner.
intentName	M	1	string	Name of the intent.
intentExpectations	M	1..N	IntentExpectation	It describes the expectations including requirements, goals and contexts (including constraints and filter information for each expectation) given to the Intent Handler. It states the list of specific outcomes desired to be realized for expectation object(s).
intentContexts	M	0..N	Context	It describes the constraints and conditions that should apply for the entire intent (i.e. all the intentExpectations).

8.2.3 Condition information element

8.2.3.1 Description

A Condition describes an assertion which needs to be met for expectation target and context.

8.2.3.2 Attributes

The attributes of the Condition information element shall follow the indications provided in table 8.2.3.2-1.

Table 8.2.3.2-1: Attributes of the Condition information element

Attribute	Qualifier	Cardinality	Content	Description
conditionId	M	1	Identifier	The identifier of this condition.
conditionName	M	1	String	It describes the name of the condition.
operator	M	1	Enum	If conditionValue exists, represents the specific operator for condition. VALUES: • greater than • equal to • less than • not equal to • one of • some of • all of • maximum value • minimum value • median • credibility If conditionList exists, represents the relationship between all conditions. VALUES: • and • or
conditionValue	M	0..1	String	For a simple condition, represents the specific value that composes the condition. See note.
conditionList	M	0..N	Identifier (Reference to Condition)	For a composite condition, identifies the list of conditions that compose the specific condition. See note.
NOTE: Either conditionValue or conditionList, but not both, shall be present.				

8.2.4 Context information element

8.2.4.1 Description

A Context describes the applicability constraints and conditions that may apply to the intent and its information elements.

8.2.4.2 Attributes

The attributes of the Context information element shall follow the indications provided in table 8.2.4.2-1.

Table 8.2.4.2-1: Attributes of the Context information element

Attribute	Qualifier	Cardinality	Content	Description
contextId	M	1	Identifier	The identifier of this context.
contextName	M	1	String	It describes the name of the context.
contextConditions	M	1..N	Identifier (Reference to Condition)	It identifies the list of conditions that compose the specific context.

8.2.5 IntentObjectInstance information element

8.2.5.1 Description

An IntentObjectInstance describes the managed object instance that is instantiated at the Intent Handler based on the intentObject received from the Intent Owner.

8.2.5.2 Attributes

The attributes of the IntentObjectInstance information element shall follow the indications provided in table 8.2.5.2-1.

Table 8.2.5.2-1: Attributes of the IntentObjectInstance information element

Attribute	Qualifier	Cardinality	Content	Description
intentObjectInstanceId	M	1	Identifier	The identifier of the intentObjectInstance.
intentId	M	1	Identifier (Reference to IntentObject)	It references the corresponding intentObject, by its unique identifier as assigned by the Intent Owner.
fulfilmentInfos	M	1..N	FulfillmentInfo	Fulfilmentinfos of intent, intentExpectations, expectationTargets or conditions.
intentReports	M	0..N	Identifier (Reference to IntentReport)	It identifies a series of intent reports generated based on this intentObjectInstance.

8.2.6 FulfilmentInfo information element

8.2.6.1 Description

A FulfilmentInfo describes the fulfilment related information for an aspect of the intent (i.e. either an expectation, an expectation target, a condition or the whole intent).

8.2.6.2 Attributes

The attributes of the FulfilmentInfo information element shall follow the indications provided in table 8.2.6.2-1.

Table 8.2.6.2-1: Attributes of the FulfilmentInfo information element

Attribute	Qualifier	Cardinality	Content	Description
fulfilmentObjectId	M	1	Identifier(Reference to IntentObjectInstance, IntentExpectation, ExpectationTarget, Condition)	Reference to an intentObjectInstance, intentExpectation, expectationTarget or condition to which this FulfilmentInfo applies. See note 1.
fulfilmentStatus	M	1	Enum	It describes the current status of the fulfilment result. VALUES: • FULFILLED • NOT_FULFILLED
notFulfilledState	M	0..1	Enum	This attribute is present when fulfilmentStatus equals to NOT_FULFILLED. It describes the current progress state. VALUES: • ACKNOWLEDGED • COMPLIANT • DEGRADED • TERMINATED • FAILED See note 2.
notFulfilledReasons	M	0..1	String	This attribute is present if fulfilmentStatus equals to NOT_FULFILLED. It describes the reasons/observations related to the specific noted notFulfilledState. See note 2.
achievedValue	M	0..1	Not Specified	It describes the actual value of fulfillment for an expectation target. The attribute shall be able to associate which values associate to which conditions in an expectationTarget.
objectInstances	M	0..N	Identifier	This attribute is present if FulfilmentInfo object refers to an intentExpectation.
NOTE 1: It is assumed that the identifiers are unique. NOTE 2: The notFulfilledState attribute and notFulfilledReasons attribute are present if FulfilmentInfo object represents an intent.				

8.2.6.3 Intent State Machine Diagram

An intent object instance can be in one of the following states:

- FULFILLED: This is the state if the Intent Handler considers that the intent has been fulfilled as desired by the Intent Owner.
- ACKNOWLEDGED: This is the default state and is the initial notFulfilledState right after the intent management request is acknowledged (i.e. the intent object has been accepted by the Intent Handler but has not been processed).
- COMPLIANT: This is the state after internal evaluation has been performed by the Intent Handler and the intent object has been accepted to be executed by the Intent Handler. In this state it means that the intent object instance has been created.
- DEGRADED: This is the state if an intent object that was previously accepted, the intent object instance has been created but after a period of observation it is found not be meeting the initially stated requirements.
- TERMINATED: This is the state reached when the Intent Owner sends a request to delete the intent object (and the corresponding intent object instance) or the Intent Handler after acknowledging the reception determines that it cannot perform the execution of the intent.
- FAILED: This is the state when the Intent Handler determines that the intent cannot be fulfilled.

Figure 8.2.6.3-1 illustrates the intent state machine diagram.

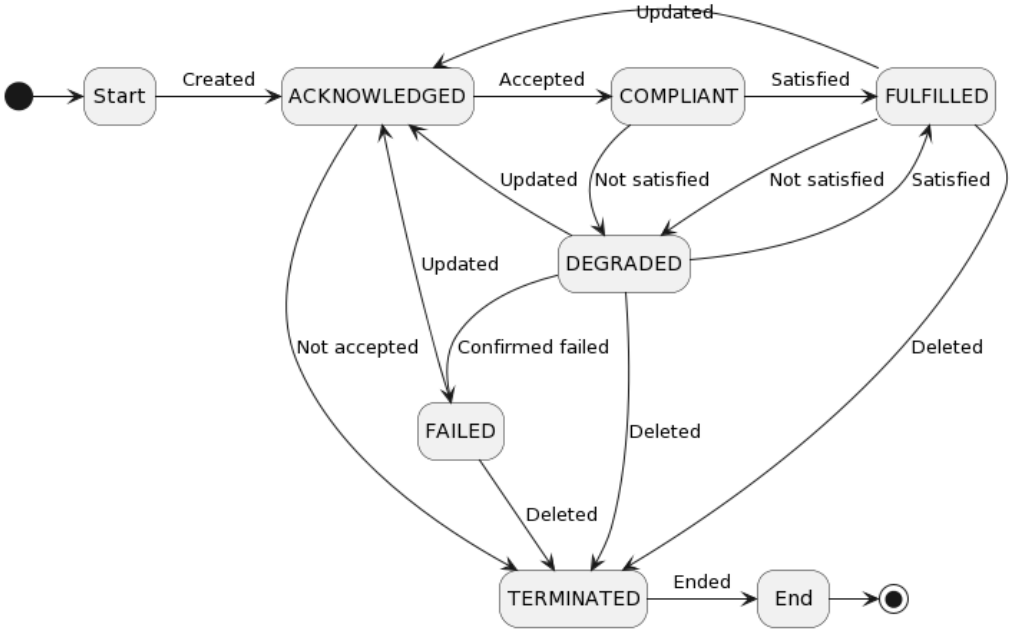

Figure 8.2.6.3-1: Intent state machine diagram

8.2.7 IntentReport information element

8.2.7.1 Description

An IntentReport describes the information provided by the Intent Handler to the subscribing entity or Intent Owner about status and progress regarding the received intent.

8.2.7.2 Attributes

The attributes of the IntentReport information element shall follow the indications provided in table 8.2.7.2-1.

Table 8.2.7.2-1: Attributes of the IntentReport information element

Attribute	Qualifier	Cardinality	Content	Description
intentReportId	M	1	Identifier	The identifier of the report.
intentReference	M	1	Identifier	It identifies the corresponding intent object instance.
fulfillmentInfos	M	1..N	FulfillmentInfo	FulfillmentInfos of intentExpectations and expectationTargets.
reportTime	M	1	DateTime	The creation time of the report.

8.2.8 ExpectationObject information element

8.2.8.1 Description

An ExpectationObject describes the objects managed by NFV-MANO that are subject to the IntentExpectation.

8.2.8.2 Attributes

The attributes of the ExpectationObject information element shall follow the indications provided in table 8.2.8.2-1.

Table 8.2.8.2-1: Attributes of the ExpectationObject information element

Attribute	Qualifier	Cardinality	Content	Description
objectType	M	0..1	String	It describes the type of object of the IntentExpectation that are expected to be applied on. It can be the type of the managed object, e.g. NS. See note.
objectInstanceId	M	0..N	Identifier	Identifier of a specific object instance managed by NFV-MANO to which the intentExpectation applies. For example, an Identifier of a NS instance. See note.
objectFilter	M	0..1	Not Specified	It describes the constraints and conditions to be used as filter information to identify the managed object(s) to which a given intentExpectation applies. See note.
NOTE: Either objectType or objectInstanceId, but not both, shall be present. The objectType is present only when the intent expectation is for a type of managed objects and the Intent Owner does not know which are the specific instances. While the objectInstanceId is present only when intent expectation is for specific object instance(s). objectFilter is only present when objectType is present.				

8.2.9 ExpectationTarget information element

8.2.9.1 Description

An ExpectationTarget describes the expectation targets for each of the expectation type defined in the intent so for the system to achieve the desired or wanted state from its present state.

8.2.9.2 Attributes

The attributes of the ExpectationTarget information element shall follow the indications provided in table 8.2.9.2-1.

Table 8.2.9.2-1: Attributes of the ExpectationTarget information element

Attribute	Qualifier	Cardinality	Content	Description
targetId	M	1	Identifier	The identifier of this expectation target.
targetName	M	1	String	It describes the name of the expectation target.
targetConditions	M	1..N	Condition	It represents the conditions that the expectation target shall meet. See note.
targetContexts	M	0..N	Context	It describes the list of constraints that applies for a specific expectation target. See note.
NOTE: TargetConditions expresses the requirements/expectations that shall be met while evaluating the specific expectation target, while targetContexts expresses the constraints on when such evaluation shall be performed. E.g. the targetName could be incomingPackets, and the targetConditions could include the conditions for each SAP of the NS, and the targetContexts could include the time range that the target are expected to be fulfilled.				

8.2.10 IntentReportNotification

8.2.10.1 Description

The IntentReportNotification notification provides information to the subscribing entity about intent reports. Intent reports are created in response to: meeting the intent report trigger conditions set by the subscribing entity through subscribe intent object instance operation; or meeting the intent report trigger conditions set by the Intent Owner through report expectations.

8.2.10.2 Trigger conditions

Intent report has been created.

8.2.10.3 Attributes

The IntentReportNotification shall follow the indications provided in table 8.2.10.3-1.

Table 8.2.10.3-1: Attributes of the IntentReportNotification

Attribute	Qualifier	Cardinality	Content	Description
intentReport	M	1	IntentReport	Information about an intent report including intentReportId, corresponding intent object instance identifier, fulfillmentInfoList, and reportTime.

Annex A (informative): PlantUML source code for intent modeling

```
@startuml
hide circle
hide methods
hide members

class "<<OpenModelClass, Preliminary>>\n IntentObjectInstance " as IntentObjectInstance {}
class "<<OpenModelClass, Preliminary>>\n IntentReport" as IntentReport{}
class "<<OpenModelClass, Preliminary>>\n IntentExpectation" as IntentExpectation{}
class "<<OpenModelClass, Preliminary>>\n IntentObject " as IntentObject {}
class "<<OpenModelClass, Preliminary>>\n Context" as Context{}
class "<<OpenModelClass, Preliminary>>\n Fulfilmentinfo" as Fulfilmentinfo{}
class "<<OpenModelClass, Preliminary>>\n ExpectationObject" as ExpectationObject{}
class "<<OpenModelClass, Preliminary>>\n ExpectationTarget" as ExpectationTarget{}
class "<<OpenModelClass, Preliminary>>\n Condition" as Condition{}


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

'skinparam linetype polyline
'skinparam linetype ortho

IntentObjectInstance "1" --> "1" IntentObject:intentObject
IntentObjectInstance "1" --> "0..N" IntentReport:intentReport
IntentObjectInstance "1" <-- "0..N" IntentReport:intentReference
IntentObjectInstance "1" *--> "1..N" Fulfilmentinfo:fulfilmentinfos
IntentObject "1" *--> "1..N" IntentExpectation:intentExpectation
IntentObject "1" *--> "0..N  " Context:intentContext
IntentReport "1" -l-> "1..N" Fulfilmentinfo:fulfillmentInfos
IntentExpectation "1" o--> "0..1" ExpectationObject:expectationObject
IntentExpectation "1" o--> "1..N" ExpectationTarget:expectationTargets
IntentExpectation "1" o--> "0..N" Context:expectationContext
ExpectationObject "1" o--> "0..N" Context:objectContext
ExpectationTarget "1" o--> "0..N" Context:targetContext
ExpectationTarget "1" o--> "1..N" Condition:targetCondition
Context"1" o--> "1..N" Condition:contextConditions
@enduml
```

Annex B (informative): Change History

Date	Version	Information about changes
July 2022	0.0.1	First draft, introducing the skeleton of the GS NFVIFA(22)000350r3
December 2022	0.0.2	Early draft including the following contributions approved until NFVIFA#313 meeting: NFVIFA(22)000565r2, NFVIFA(22)000566r1, NFVIFA(22)000570, NFVIFA(22)000791r7, NFVIFA(22)000792r7
March 2023	0.0.3	Early draft including the following contributions approved until NFVIFA#326 meeting: NFVIFA(23)000047r5, NFVIFA(23)000019r5, NFVIFA(23)000080r5, NFVIFA(23)000116r1, NFVIFA(23)000039r2, NFVIFA(22)000904r4
April 2023	0.1.0	Early draft including the following contribution approved until NFVIFA#331 meeting: NFVIFA(23)000194r4, NFVIFA(23)000121r8, NFVIFA(23)000122r7, NFVIFA(23)000118r5, NFVIFA(23)000173r2, NFVIFA(23)000907r4, NFVIFA(23)000048r2, NFVIFA(23)000120r4, NFVIFA(23)000038r2, NFVIFA(23)000119r5, NFVIFA(23)000906r3
April 2023	0.2.0	Early draft including the following contribution approved until NFVIFA#333 meeting: NFVIFA(23)000257r2, NFVIFA(23)000240r1, NFVIFA(23)000165r2, NFVIFA(23)000164r2, NFVIFA(23)000193r7, NFVIFA(23)000227r1, NFVIFA(23)000234r1, NFVIFA(23)000258r2
May 2023	0.3.0	Early draft including the following contribution approved until NFVIFA#337 meeting: NFVIFA(23)000314r3, NFVIFA(23)000325r2, NFVIFA(23)000324r1, NFVIFA(23)000326r1, NFVIFA(23)000166r4, NFVIFA(23)000163r4, NFVIFA(23)000167r2, NFVIFA(23)000456r2, NFVIFA(23)000368r3
August 2023	0.4.0	Stable draft including the following contributions approved until NFVIFA#342 meeting: NFVIFA(23)000493r2, NFVIFA(23)000494r1, NFVIFA(23)000496, NFVIFA(23)000501, NFVIFA(23)000511, NFVIFA(23)000516, NFVIFA(23)000531r1, NFVIFA(23)000551r1, NFVIFA(23)000562, NFVIFA(23)000554r2, NFVIFA(23)000555r3, NFVIFA(23)000551r1
August 2023	0.4.1	Stable draft including the following contributions approved until NFVIFA#342 meeting: NFVIFA(23)000511, NFVIFA(23)000554r3, NFVIFA(23)000555r4, NFVIFA(23)000561r2
August 2023	0.4.2	Stable draft including minor changes suggested during the IFA email approval.
August 2023	0.4.3	Stable draft including minor changes suggested during the IFA email approval.
August 2023	0.4.4	Stable draft including minor changes suggested during the IFA email approval.
August 2024	5.1.2	First draft for ed521 created from published version 4.5.1.
October 2024	5.1.3	Early draft including the following contribution approved until NFVIFA#399 meeting: NFVIFA(24)000559r2

History

Document history		
V5.2.1	November 2024	Publication