ETSI TS 132 157 V17.0.0 (2022-04)

Universal Mobile Telecommunications System (UMTS);
LTE;
Telecommunication management;
Integration Reference Point (IRP) Information Service (IS) template
(3GPP TS 32.157 version 17.0.0 Release 17)

Reference
RTS/TSGS-0532157vh00

Keywords
LTE,UMTS

ETSI

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

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

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

Intellectual Property Rights

Essential patents

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

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.

1 Scope

The present document contains the template to be used for the production of all Integration Reference Point (IRP) Information Service (IS) specifications for Converged 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 TS 32.101: "Telecommunication management; Principles and high level requirements".

[2] 3GPP TS 32.102: "Telecommunication management; Architecture".

[3] 3GPP TS 32.150: "Telecommunication management; Integration Reference Point (IRP) Concept and definitions".

[4] 3GPP TS 32.156: “Telecommunication management; Fixed Mobile Convergence (FMC) Model Repertoire

[5] 3GPP TS 32.302: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Information Service (IS)".

[6] ITU-T Recommendation M.3020 (07/2011): "Management interface specification methodology" – Annex E "Information type definitions – type repertoire".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 32.101 [1], 3GPP TS 32.102 [2], 3GPP TS 32.150 [3] and the following apply:

IRPAgent: See 3GPP TS 32.150 [3].

IRPManager: See 3GPP TS 32.150 [3].

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TS 32.101 [1], 3GPP TS 32.102 [2], 3GPP TS 32.150 [3] and the following apply:

IOC	Information Object Class
IRP	Integration Reference Point
IS	Information Service
OMG	Object Management Group
UML	Unified Modelling Language (OMG)

4 Information Service (IS) template

4.1 General

The present document contains the templates to be used for the production of all Integration Reference Point (IRP) Information Service (IS) specifications for Converged Management.

Clause 4.2 is applicable for NRM IRP IS specifications.

Clause 4.3 is applicable for Interface IRP IS specifications.

The IS template uses qualifiers M, O, CM, CO and C. The semantics of these qualifiers are defined in [4].

The IS template uses type definition as one characteristic to describe class attributes and operation/notification parameters. The valid type definitions that can be used and their semantics are defined in [4].

Usage of fonts shall be according to the following table.

Item	Font
Class names	Courier New
Attribute names	Courier New
Operation names	Courier New
Parameter names	Courier New
Assertion names	Courier New
Notification names	Courier New
Exception names	Courier New
State names	Arial
Matching Information	Courier New
Information Type	Courier New
Legal Values	Courier New
NOTE: These font requirements do not apply to UML diagrams.	

4.2 Template for NRM IRP IS

W1 Scope

The following quoted text is relevant for all NRM IRP ISs. It shall be copied as the first two paragraphs of this clause in the NRM IRP IS specification. IRP IS author may add additional paragraph(s) if necessary.

"

The present document specifies the <<n>> *(where <<n>> shall be substituted by the name of the NRM IRP IS concerned such as 'HNS', 'E_UTRAN', 'GERAN')* network resource information that can be communicated between an IRPAgent and an IRPManager for telecommunication network management purposes, including management of converged networks.

This document specifies the semantics and behaviour of information object class attributes and relations visible across the reference point in a protocol and technology neutral way. It does not define their syntax and encoding.

"

W2 References

The following text from the standard 3GPP TS template shall be copied into the NRM IRP IS specification, followed by the TS/TR references used in the specification:

"

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] …

[2] …

…

"

W3 Definitions and abbreviations

The following text from the standard 3GPP TS template shall be copied into the NRM IRP IS specification, completed by the definitions and abbreviations used in the specification:

"

W3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [1] and the following apply. For terms and definitions not found here, please refer to 3GPP TS *<list of relevant high-level 3GPP SA5 specifications>*. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1] and *< list of relevant high-level 3GPP SA5 specifications>*.

<defined term>: <definition>.

W3.2 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].

<ACRONYM> <Explanation>

"

W4 Model

W4.1 Imported and associated information entities

W4.1.1 Imported information entities and local labels

This clause identifies a list of information entities (e.g. information object class, interface, attribute) that have been defined in other specifications and that are imported in the present (target) specification. All imported entities shall be treated as if they are defined locally in the target specification. One usage of import is for inheritance purpose.

Each element of this list is a pair (label reference, local label). The label reference contains the name of the original specification where the information entity is defined, the information entity type and its name. The local label contains the name of the information entity that appears in the target specification, and the entity name in the local label shall be kept identical to the name defined in the original specification. The local label can then be used throughout the target specification instead of that which appears in the label reference.

This information is provided in a table. An example of such a table is given here below:

Label reference	Local label
3GPP TS 32.622 [xy], information object class, `Top`	`Top`

W4.1.2 Associated information entities and local labels

This clause identifies a list of information entities (e.g. information object class, interface, attribute) that have been defined in other specifications and that are associated with the information entities defined in the present (target) specification. For the associated information entity, only its properties (e.g., DN (see 5.2.2 of 32.156 [4]), attribute (see 5.2.1 of 32.156 [4]) of an instance of the associated information entity) used as associated information needs to be supported locally in the target specification.

Each element of this list is a pair (label reference, local label). The label reference contains the name of the original specification where the information entity is defined, the information entity type and its name. The local label contains the name of the information entity that appears in the target specification. The local label can then be used throughout the target specification instead of that which appears in the label reference.

This information is provided in a table. An example of such a table is given here below:

Label reference	Local label
3GPP TS 28.705 [12], IOC, `HSSFunction`	`HSSFunction`

W4.2 Class diagram

W4.2.1 Relationships

This first set of diagrams represents all classes defined in this IS with all their relationships and all their attributes, including relationships with imported information entities (if any). These diagrams shall contain class cardinalities (for associations as well as containment relationships) and may also contain role names. These shall be UML compliant class diagrams (see also [4]).

Characteristics (attributes, relationships) of imported information entities need not to be repeated in the diagrams. Allowable classes are specified in [4].

*Use this as the first paragraph: "*This clause depicts the set of classes (e.g. IOCs) that encapsulates the information relevant for this IRP. This clause provides an overview of the relationships between relevant classes in UML. Subsequent clauses provide more detailed specification of various aspects of these classes."

W4.2.2 Inheritance

This second set of diagrams represents the inheritance hierarchy of all classes defined in this specification. These diagrams do not need to contain the complete inheritance hierarchy but shall at least contain the parent classes of all classes defined in the present document. By default, a class inherits from the class "top".

Characteristics (attributes, relationships) of imported classes need not to be repeated in the diagrams.

NOTE: some inheritance relationships presented in clause W4.2.2 can be repeated in clause W4.2.1 to enhance readability.

*Use "*This subclause depicts the inheritance relationships." *as the first paragraph.*

W4.3 Class definitions

Each class is defined using the following structure.

Inherited items (attributes etc.) shall not be shown, as they are defined in the parent class(es) and thus valid for the subclass.

W4.3.a `InformationObjectClassName`

InformationObjectClassName is the name of the information object class.

The "a" represents a number, starting at 1 and increasing by 1 with each new definition of a class.

W4.3.a.1 Definition

This clause is written in natural language. The <definition> clause refers to the class itself.

Optionally, information on traceability back to one or more requirements supported by this class can be defined here, in the following form:

Referenced TS	Requirement label	Comment
3GPP TS 32.xyz [xy]	REQ-SM-CON-23	*Optional clarification*
3GPP TS 32.xyz [xy]	REQ-SM-FUN-11	*Optional clarification*

W4.3.a.2 Attributes

This clause presents the list of attributes, which are the manageable properties of the class. Each attribute is characterised by some of the attribute properties (see Table 1 of [4]), i.e. supportQualifier, isReadable, isWritable, isInvariant and isNotifyable.

The legal values and their semantics for attribute properties are defined in [4].

This information is provided in a table.

An example below indicates

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
eNodeBId	M	M	-	M	M

Another example below indicates that the attribute `password1` *is not readable, is writable, is not an invariant and no* `notifyAttributeValueChange` *will be emitted when the attribute value is changed.*

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
password1	O	-	M	-	-

Another example below indicates that the attribute `password2` *and* `password1` *(in example above) has same qualifiers for the shown properties except that of isReadable. In the case of password1, the standard specification determines the qualifier to be M, i.e. it is readable. In the case of password2, the standard specification does not make a determination. The vendor would make the determination if the attribute is readable or not readable.*

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
password2	O	O	M	-	-

In case there is one or more attributes related to role (see section 5.2.9 of [4]), the attributes related to role shall be specified at the bottom of the table with a divider "Attribute related to role", as shown in the following example:

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
aTMChannelTerminationPointid	M	M	-	M	M
…					
…					
Attribute related to role					
theATMPathTerminationPoint	M	M	-	-	M
theIubLink	M	M	-	-	M

This clause shall state "None." when there is no attribute to define.

W4.3.a.3 Attribute constraints

This clause presents constraints for the attributes, and one use is to present the predicates for conditional qualifiers (CM/CO).

This information is provided in a table. An example of such a table is given here below:

Name	Definition
pci CM write qualifier	Centralized PCI assignment (see TS 32.500, ref [xy] clause 6.1.6) is supported.
pciList CM support qualifier	Distributed PCI assignment (see TS 32.500, ref [xy] clause 6.1.6) is supported.
partOfSectorPower CM support qualifier	The IOC SectorEquipmentFunction is used.
attributeX max value	The value of attributeX shall be within the specified value range but may never be higher than the value of attributeY.

This clause shall state "None." when there is no attribute constraint to define.

W4.3.a.4 Notifications

This clause, for this class, presents one of the following options:

a) The class *defines (and independent from those inherited) the support of a set of notifications that is identical to that defined in clause W4.5. In such case, use "The common notifications defined in clause W4.5 are valid for this class, without exceptions or additions." as the lone sentence of this clause.*

b) The class defines (and independent from those inherited) the support of a set of notifications that is a superset of that defined in clause W4.5. In such case, use "The common notifications defined in clause W4.5 are valid for

this IOC. In addition, the following set of notification is also valid." as the lone paragraph of this clause. Then, define the 'additional' notifications in a table. See clause W4.5 for the notification table format.

c) The class defines (and independent from those inherited) the support of a set of notifications that is not identical to, nor a superset of, that defined in clause W4.5. In such case, use "The common notifications defined in clause W4.5 are not valid for this IOC. The set of notifications defined in the following table is valid." as the lone paragraph of this clause. Specify the set of notifications in a table. See clause W4.5 for the notification table format.

*d) The class does not define (and independent from those inherited) the support of any notification. In such case, use "There is n*o notification defined." as the lone sentence of this clause.

The notifications identified (i.e. option-a, option-b and option-c above) in this clause are notifications that can be emitted across the Itf-N, where the "object class" and "object instance" parameters of the notification header (see note 2) of these notifications identifies an instance of the class (or its direct or indirect derived class) defined by the encapsulating clause (i.e. clause W4.3.a).

The notifications identified (i.e. option-a and option-b above) in this clause, may originate from implementation object(s) whose identifier may or may not be the same as that carried in the notification parameters "object class" and "object instance". Hence the identification of notifications in this clause does not imply nor identify those notifications as being originated from an instance of the class (or its direct or indirect derived class) defined by the encapsulating clause (i.e. clause W4.3.a).

This clause shall state "This class does not support any notification." (see option-c) when there is no notification defined for this class. (Note that if its parent class has defined some notifications, the implementation of this class is capable of emitting those inherited defined notifications.)

The notification header is defined in the notification IRP Information service TS 32.302 [5].

The qualifier of a notification, specified in Notification Table, indicates if an implementation can generate a notification carrying the DN of the subject class. The qualifier of a notification, specified in an Interface IRP, indicates if an implementation of the Interface IRP can generate such notification in general.

An IRPManager can receive notification-XYZ that carries DN (the "object class" and "object instance") of class-ABC instance if and only if:

a) The class-ABC Notification Table defines the notification-XYZ and

b) The class-ABC instance implementation supports this notification-XYZ and

c) An Interface IRP defines the notification-XYZ and

d) The Interface IRP implementation supports this notification-XYZ.

W4.3.a.5 State diagram

This subclause contains state diagrams. A state diagram of an information object class defines permitted states of this information object class and the transitions between those states. A state is expressed in terms of individual attribute values or a combination of attribute values or involvement in relationships of the information object class being defined. This shall be a UML compliant state diagram.

This subclause shall state "None." when there is no State diagram defined.

W4.4 Attribute definitions

W4.4.1 Attribute properties

It has a lone paragraph "The following table defines the properties of attributes that are specified in the present document. ".

Each information attribute is defined using the following structure.

Inherited attributes shall not be shown, as they are defined in the parent class(es) and thus valid for this class.

An attribute has properties (see Table 1 of [4]). Some properties of an attribute are defined in W4.3.a.2 (e.g. Support Qualifier). The remaining properties of an attribute (e.g. documentation, default value) are defined here.

The information is provided in a table. In case a) attributes of the same name are specified in more than one class and b) the attributes have different properties, then the attribute names (first column) should be prefixed with the class name followed by a period.

An example is given below:

Attribute Name	Documentation and Allowed Values	Properties
xyzId	It identifies … allowedValues: …	type: Integer multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: False
Abc.state	It indicates … allowedValues: "ON": the state is on; "OFF": the state is off.	type: <<enumeration>> multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: False
Zyz.state	It indicates … allowedValues: "HIGH": the state is high; "MEDIUM": the state is medium; "LOW": the state is low.	type: <<enumeration>> multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: False
abc	It defines… allowedValues: …	type: … multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: …

In case there is one or more attributes related to role (see section 5.2.9 of [4]), the attributes related to role shall be specified at the bottom of the table with a divider "Attribute related to role". See example below.

Attribute Name	Documentation and Allowed Values	Properties
abc	It defines… allowedValues: …	type: <<dataType>> multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: …
Attribute related to role		
aEnd	It defines… allowedValues: Values to be conformant to TS 32.300 [xy] …	type: DN multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: False

This clause shall state "None." if there is no attribute to define.

W4.4.2 Constraints

This clause indicates whether there are any constraints affecting attributes. Each constraint is defined by a triplet (propertyName, affectedAttributes, propertyDefinition). PropertyDefinitions are expressed in natural language.

An example is given here below:

Name	Affected attribute(s)	Definition
inv_TimerConstraints	ntfTimeTickTimer	The ntfTimeTickTimer is lower than or equal to ntfTimeTick.

This clause shall state "None." if there is no constraint.

W4.5 Common notifications

This clause presents notifications that can be referred to by any class defined in the specification. This information is provided in tables.

W4.5.1 Alarm notifications

The following quoted text shall be copied as the only paragraph of this clause.

"This clause presents a list of notifications, defined in [x], that IRPManager can receive. The notification header attribute objectClass/objectInstance, defined in [y], shall capture the DN of an instance of a class defined in this specification."

The information is provided in a table. The following is an example.

Name	Qualifier	Notes
notifyNewAlarm	M	--

W4.5.2 Configuration notifications

The following quoted text shall be copied as the only paragraph of this clause.

"This clause presents a list of notifications, defined in [x], that IRPManager can receive. The notification header attribute objectClass/objectInstance, defined in [z], shall capture the DN of an instance of a class defined in this specification."

The information is provided in a table. The following is an example.

Name	Qualifier	Notes
notifyAttributeValueChange	O	--
notifyObjectCreation	O	--
notifyObjectDeletion	O	--

W4.6 System State Model

Some configurations of information are special or complex enough to justify the usage of a state diagram to clarify them. A state diagram in this clause defines permitted states of the system and the transitions between those states. A state is expressed in terms of a combination of attribute values constraints or involvement in relationships of one or more information object classes.

4.3 Template for Interface IRP IS

Y1 Scope

The following quoted text is relevant for all Interface IRP ISs. It shall be copied as the first two paragraphs of this clause in the Interface IRP IS specification. IRP IS author may add additional paragraph(s) if necessary.

“

The present document specifies the <<n>> (*where <<n>> shall be substituted by the name of the Interface IRP IS concerned such as “Alarm”, “Test”,” Entry Point”)* management operations and notifications that can be communicated between an IRPAgent and one or several IRPManagers for telecommunication network management purposes, including management of converged networks.

This document specifies the semantics and behaviour of operations, notifications and their parameters visible across the reference point in a protocol and technology neutral way. It does not define their syntax and encoding.

“

Y2 References

The following text from the standard 3GPP TS template shall be copied into the Interface IRP IS specification, followed by the TS/TR references used in the specification:

"

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] …

[2] …

…

"

Y3 Definitions and abbreviations

The following text from the standard 3GPP TS template shall be copied into the NRM IRP IS specification, completed by the definitions and abbreviations used in the specification:

"

W3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [x] and the following apply. For terms and definitions not found here, please refer to 3GPP TS *<list of relevant high-level 3GPP SA5 specifications>*. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [x] and *< list of relevant high-level 3GPP SA5 specifications>*.

<defined term>: <definition>.

W3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [x] 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 [x].

<ACRONYM> <Explanation>

"

Y4 System overview

The following quoted text is relevant for all Interface IRP ISs. It shall be copied as clause 4.1 in the Interface IRP IS specification.

"

Y4.1 System Context

The general definition of the System Context for the present IRP is found in 3GPP TS 32.150 [xy] subclause 4.7.

In addition, the set of related IRP(s) relevant to the present IRP is shown in figure…

<System context diagram(s)>

"

Y5 Model

Y5.1 Imported and associated information entities

Y5.1.1 Imported information entities and local labels

This clause identifies a list of information entities (e.g. information object class, interface, attribute) that have been defined in other specifications and that are imported in the present (target) specification. All imported entities shall be treated as if they are defined locally in the target specification. One usage of import is for inheritance purpose.

Each element of this list is a pair (label reference, local label). The label reference contains the name of the original specification where the information entity is defined, the information entity type and its name. The local label contains the name of the information entity that appears in the target specification, and the entity name in the local label shall be kept identical to the name defined in the original specification. The local label can then be used throughout the target specification instead of that which appears in the label reference.

This information is provided in a table. An example of such a table is given here below:

Label reference	Local label
3GPP TS 32.622 [xy], information object class, Top	Top

Y5.1.2 Associated information entities and local labels

This clause identifies a list of information entities (e.g. information object class, interface, attribute) that have been defined in other specifications and that are associated with the information entities defined in the present (target) specification. For the associated information entity, only its properties (e.g., DN (see 5.2.2 of 32.156 [4]), attribute (see 5.2.1 of 32.156 [4]) of an instance of the associated information entity) used as associated information needs to be supported locally in the target specification.

Each element of this list is a pair (label reference, local label). The label reference contains the name of the original specification where the information entity is defined, the information entity type and its name. The local label contains the name of the information entity that appears in the target specification. The local label can then be used throughout the target specification instead of that which appears in the label reference.

This information is provided in a table. An example of such a table is given here below:

Label reference	Local label
3GPP TS 28.705 [12], IOC, `HSSFunction`	`HSSFunction`

Y5.2 Class diagram

Y5.2.1 Relationships

This first set of diagrams represents all classes defined in this IS with all their relationships and all their attributes, including relationships with imported information entities (if any). These diagrams shall contain class cardinalities (for associations as well as containment relationships) and may also contain role names. These shall be UML compliant class diagrams (see also [4]).

Characteristics (attributes, relationships) of imported information entities need not to be repeated in the diagrams. Allowable classes are specified in [4].

*Use this as the first paragraph: "*This clause depicts the set of classes (e.g. IOCs) that encapsulates the information relevant for this IRP. This clause provides an overview of the relationships between relevant classes in UML. Subsequent clauses provide more detailed specification of various aspects of these classes."

Y5.2.2 Inheritance

This second set of diagrams represents the inheritance hierarchy of all classes defined in this specification. These diagrams do not need to contain the complete inheritance hierarchy but shall at least contain the parent classes of all classes defined in the present document. By default, a class inherits from the class "top".

Characteristics (attributes, relationships) of imported classes need not to be repeated in the diagrams.

NOTE: some inheritance relationships presented in clause Y5.2.2 can be repeated in clause Y5.2.1 to enhance readability.

*Use "*This subclause depicts the inheritance relationships." *as the first paragraph.*

Y5.3 Class definitions

Each class is defined using the following structure.

Inherited items (attributes etc.) shall not be shown, as they are defined in the parent class(es) and thus valid for the subclass.

Y5.3.a InformationObjectClassName

InformationObjectClassName is the name of the information object class.

The "a" represents a number, starting at 1 and increasing by 1 with each new definition of a class.

Y5.3.a.1 Definition

This clause is written in natural language. The <definition> clause refers to the class itself.

Optionally, information on traceability back to one or more requirements supported by this class can be defined here, in the following form:

Referenced TS	Requirement label	Comment
3GPP TS 32.xyz [xy]	REQ-SM-CON-23	*Optional clarification*
3GPP TS 32.xyz [xy]	REQ-SM-FUN-11	*Optional clarification*

Y5.3.a.2 Attributes

This clause presents the list of attributes, which are the manageable properties of the class. Each attribute is characterised by some of the attribute properties (see Table 1 of [4]), i.e. supportQualifier, isReadable, isWritable, isInvariant and isNotifyable.

The legal values and their semantics for attribute properties are defined in [4].

This information is provided in a table.

An example below indicates

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
eNodeBId	M	M	-	M	M

Another example below indicates that the attribute `password1` *is not readable, is writable, is not an invariant and no* `notifyAttributeValueChange` *will be emitted when the attribute value is changed.*

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
password1	O	-	M	-	-

Another example below indicates that the attribute `password2` *and* `password1` *(in example above) has same qualifiers for the shown properties except that of isReadable. In the case of password1, the standard specification determines the qualifier to be M, i.e. it is readable. In the case of password2, the standard specification does not make a determination. The vendor would make the determination if the attribute is readable or not readable.*

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
password2	O	O	M	-	-

In case there is one or more attributes related to role (see section 5.2.9 of [4]), the attributes related to role shall be specified at the bottom of the table with a divider "Attribute related to role", as shown in the following example:

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifyable
aTMChannelTerminationPointid	M	M	-	M	M
…					
…					
Attribute related to role					
theATMPathTerminationPoint	M	M	-	-	M
theIubLink	M	M	-	-	M

This clause shall state "None." when there is no attribute to define.

Y5.3.a.3 Attribute constraints

This clause presents constraints for the attributes, and one use is to present the predicates for conditional qualifiers (CM/CO).

This information is provided in a table. An example of such a table is given here below:

Name	Definition
`pci` CM write qualifier	Centralized PCI assignment (see TS 32.500, ref [xy] clause 6.1.6) is supported.
`pciList` CM support qualifier	Distributed PCI assignment (see TS 32.500, ref [xy] clause 6.1.6) is supported.
`partOfSectorPower` CM support qualifier	The IOC `SectorEquipmentFunction` is used.
`attributeX max value`	The value of `attributeX` shall be within the specified value range but may never be higher than the value of `attributeY`.

This clause shall state "None." when there is no attribute constraint to define.

Y5.3.a.4 Notifications

This clause, for this class, presents one of the following options:

a) The class defines (and independent from those inherited) the support of a set of notifications that is identical to that defined in clause Y5.5. In such case, use "The common notifications defined in clause Y5.5 are valid for this class, without exceptions or additions." as the lone sentence of this clause.

b) The class defines (and independent from those inherited) the support of a set of notifications that is a superset of that defined in clause Y5.5. In such case, use "The common notifications defined in clause Y5.5 are valid for this IOC. In addition, the following set of notification is also valid." as the lone paragraph of this clause. Then, define the 'additional' notifications in a table. See clause Y5.5 for the notification table format.

c) The class defines (and independent from those inherited) the support of a set of notifications that is not identical to, nor a superset of, that defined in clause Y5.5. In such case, use "The common notifications defined in clause Y5.5 are not valid for this IOC. The set of notifications defined in the following table is valid." as the lone paragraph of this clause. Specify the set of notifications in a table. See clause Y5.5 for the notification table format.

d) The class does not define (and independent from those inherited) the support of any notification. In such case, use "There is no notification defined." as the lone sentence of this clause.

The notifications identified (i.e. option-a, option-b and option-c above) in this clause are notifications that can be emitted across the Itf-N, where the "object class" and "object instance" parameters of the notification header (see note 2) of these notifications identifies an instance of the class (or its direct or indirect derived class) defined by the encapsulating clause (i.e. clause Y5.3.a).

The notifications identified (i.e. option-a and option-b above) in this clause, may originate from implementation object(s) whose identifier may or may not be the same as that carried in the notification parameters "object class" and "object instance". Hence the identification of notifications in this clause does not imply nor identify those notifications as being originated from an instance of the class (or its direct or indirect derived class) defined by the encapsulating clause (i.e. clause Y5.3.a).

This clause shall state "This class does not support any notification." (see option-c) when there is no notification defined for this class. (Note that if its parent class has defined some notifications, the implementation of this class is capable of emitting those inherited defined notifications.)

The notification header is defined in the notification IRP Information service TS 32.302 [5].

The qualifier of a notification, specified in Notification Table, indicates if an implementation can generate a notification carrying the DN of the subject class. The qualifier of a notification, specified in an Interface IRP, indicates if an implementation of the Interface IRP can generate such notification in general.

An IRPManager can receive notification-XYZ that carries DN (the "object class" and "object instance") of class-ABC instance if and only if:

a) The class-ABC Notification Table defines the notification-XYZ and

b) The class-ABC instance implementation supports this notification-XYZ and

c) An Interface IRP defines the notification-XYZ and

d) The Interface IRP implementation supports this notification-XYZ.

Y5.3.a.5 State diagram

This subclause contains state diagrams. A state diagram of an information object class defines permitted states of this information object class and the transitions between those states. A state is expressed in terms of individual attribute values or a combination of attribute values or involvement in relationships of the information object class being defined. This shall be a UML compliant state diagram.

This subclause shall state "None." when there is no State diagram defined.

Y5.4 Attribute definitions

Y5.4.1 Attribute properties

It has a lone paragraph "The following table defines the properties of attributes that are specified in the present document. ".

Each information attribute is defined using the following structure.

Inherited attributes shall not be shown, as they are defined in the parent class(es) and thus valid for this class.

An attribute has properties (see Table 1 of [4]). Some properties of an attribute are defined in Y5.3.a.2 (e.g. Support Qualifier). The remaining properties of an attribute (e.g. documentation, default value) are defined here.

The information is provided in a table. In case a) attributes of the same name are specified in more than one class and b) the attributes have different properties, then the attribute names (first column) should be prefixed with the class name followed by a period.

An example is given below:

Attribute Name	Documentation and Allowed Values	Properties
xyzId	It identifies … allowedValues: …	type: Integer multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: False
Abc.state	It indicates … allowedValues: "ON": the state is on; "OFF": the state is off.	type: <<enumeration>> multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: False
Zyz.state	It indicates … allowedValues: "HIGH": the state is high; "MEDIUM": the state is medium; "LOW": the state is low.	type: <<enumeration>> multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: False isNullable: False
abc	It defines… allowedValues: …	type: … multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: …

In case there is one or more attributes related to role (see section 5.2.9 of [4]), the attributes related to role shall be specified at the bottom of the table with a divider "Attribute related to role". See example below.

Attribute Name	Documentation and Allowed Values	Properties
abc	It defines… allowedValues: …	type: <<dataType>> multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: …
Attribute related to role		
aEnd	It defines… allowedValues: Values to be conformant to TS 32.300 [xy] …	type: DN multiplicity: … isOrdered: … isUnique: … defaultValue: … isNullable: False

This clause shall state "None." if there is no attribute to define.

Y5.4.2 Constraints

This clause indicates whether there are any constraints affecting attributes. Each constraint is defined by a triplet (propertyName, affectedAttributes, propertyDefinition). PropertyDefinitions are expressed in natural language.

An example is given here below:

Name	Affected attribute(s)	Definition
`inv_TimerConstraints`	`ntfTimeTickTimer`	The `ntfTimeTickTimer` is lower than or equal to `ntfTimeTick`.

This clause shall state "None." if there is no constraint.

Y5.5 Common Notifications

This clause presents notifications that can be referred to by any class defined in the specification. This information is provided in tables.

This clause shall state "None." if there are no common notifications.

Y5.5.1 Alarm notifications

The following quoted text shall be copied as the only paragraph of this clause.

"This clause presents a list of notifications, defined in [x], that IRPManager can receive. The notification header attribute `objectClass/objectInstance`, defined in [y], shall capture the DN of an instance of a class defined in this specification."

The information is provided in a table. The following is an example.

Name	Qualifier	Notes
`notifyNewAlarm`	M	--

Y5.5.2 Configuration notifications

The following quoted text shall be copied as the only paragraph of this clause.

"This clause presents a list of notifications, defined in [x], that IRPManager can receive. The notification header attribute `objectClass/objectInstance`, defined in [z], shall capture the DN of an instance of a class defined in this specification."

The information is provided in a table. The following is an example.

Name	Qualifier	Notes
notifyAttributeValueChange	O	--
notifyObjectCreation	O	--
notifyObjectDeletion	O	--

Y6 Interface Definition

Y6.1 Class diagram representing interfaces

Each interface is defined in one or more UML compliant class diagrams (see also 3GPP TS 32.156 [4]).

Y6.2 Generic rules

The following rules are relevant for all ISs. They shall simply be copied as part of the specification.

Rule 1: each operation with at least one input parameter supports a pre-condition valid_input_parameter which indicates that all input parameters shall be valid with regards to their information type. Additionally, each such operation supports an exception operation_failed_invalid_input_parameter which is raised when pre-condition valid_input_parameter is false. The exception has the same entry and exit state.

Rule 2: Each operation with at least one optional input parameter supports a set of pre-conditions supported_optional_input_parameter_yyy where "yyy" is the name of the optional input parameter and the pre-condition indicates that the operation supports the named optional input parameter. Additionally, each such operation supports an exception operation_failed_unsupported_optional_input_parameter_yyy which is raised when (a) the pre-condition supported_optional_input_parameter_yyy is false and (b) the named optional input parameter is carrying information. The exception has the same entry and exit state.

Rule 3: each operation shall support a generic exception operation_failed_internal_problem which is raised when an internal problem occurs and that the operation cannot be completed. The exception has the same entry and exit state.

Y6.b InterfaceName Interface (supportQualifier)

InterfaceName is the name of the interface followed by a qualifier indicating whether the interface is Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).

"b" represents a number, starting at 3 and increasing by 1 with each new definition of an interface.

Each interface is defined by its name and by a sequence of operations or notifications.

Interfaces related to operations shall be listed before the interfaces related to notifications.

If the interface is related to operation(s), the following Y.b.a "Operation OperationName (supportQualifier)" *shall be applied.*

*If the interface is related to notification(s), the next Y.b.a "*Notification NotificationName (supportQualifier)" *below shall be applied.*

Y6.b.a Operation OperationName (supportQualifier)

OperationName is the name of the operation followed by a qualifier indicating whether the operation is Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).

"a" represents a number, starting at 1 and increasing by 1 with each new definition of an operation.

Y6.b.a.1 Definition

This subclause is written in natural language.

Information on traceability back to one or more requirements supported by this operation should also be defined here, in the following form:

Referenced TS	Requirement label	Comment
3GPP TS 32.xyz [xy]	REQ-SM-CON-23	*Optional clarification*
3GPP TS 32.xyz [xy]	REQ-SM-FUN-11	*Optional clarification*

Y6.b.a.2 Input parameters

List of input parameters of the operation. Each element is a tuple (Parameter Name, Support Qualifier, Information Type (see [6] and Note 1) and an optional list of Legal Values supported by the parameter, Comment). Legal Values for the Support Qualifier are: Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).

This information is provided in a table. An example of such a table is given here below:

Parameter Name	Support Qualifier	Information Type / Legal Values	Comment
`eventIdList`	M	`SET OF INTEGER / --`	One or more event identifiers

Note 1: Information Type qualifies the parameter of Parameter Name. In the case where the Legal Values can be enumerated, each element is a pair (Legal Value Name, Legal Value Semantics), unless a Legal Value Semantics applies to several values in which case the definition is provided only once. When the Legal Values cannot be enumerated, the list of Legal Values is defined by a single definition.

Y6.b.a.3 Output parameters

List of output parameters of the operation. Each element is a tuple (Parameter Name, Support Qualifier, Matching Information / Information Type (see [6]) (Note 1) and an optional list of Legal Values supported by the parameter, Comment). Legal Values for the Support Qualifier are: Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).

This information is provided in a table. An example of such a table is given here below:

Parameter Name	Support Qualifier	Matching Information / Information Type / Legal Values	Comment
`eventTime`	M	`AlarmInformation.alarmRaisedTime / GeneralizedTime / --`	The parameter carries the • alarmRaisedTime in case notificationType carries notifyNewAlarm, • alarmChangedTime in case notificationType carries notifyChangedAlarm, • alarmClearedTime in case notificationType carries notifyClearedAlarm.

Note 1: Information Type qualifies the parameter of Parameter Name. In the case where the Legal Values can be enumerated, each element is a pair (Legal Value Name, Legal Value Semantics), unless a Legal Value Semantics applies to several values in which case the definition is provided only once. When the Legal Values cannot be enumerated, the list of Legal Values is defined by a single definition.

This table shall also include a special parameter 'status' to indicate the completion status of the operation (success, partial success, failure reason etc.).

Y6.b.a.4 Pre-condition

A pre-condition is a collection of assertions joined by AND, OR, and NOT logical operators. The pre-condition must be held to be true before the operation is invoked. An example is given here below:

notificationCategoriesNotAllSubscribed OR
notificationCategoriesParameterAbsentAndNotAllSubscribed

Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the pre-condition are provided in a table. An example of such a table is given here below:

Assertion Name	Definition
notificationCategoriesNotAllSubscribed	At least one notificationCategory identified in the notificationCategories input parameter is supported by IRPAgent and is not a member of the ntfNotificationCategorySet attribute of an NtfSubscription which is involved in a subscription relationship with the NtfSubscriber identified by the managerReference input parameter.
notificationCategoriesParameterAbsentAndNotAllSubscribed	The notificationCategories input parameter is absent and at least one notificationCategory supported by IRPAgent is not a member of the ntfNotificationCategorySet attribute of an ntfSsubscription which is involved in a subscription relationship with the NtfSubscriber identified by the managerReference input parameter.

Y6.b.a.5 Post-condition

A post-condition is a collection of assertions joined by AND, OR, and NOT logical operators. The post-condition must be held to be true after the completion of the operation. When nothing is said in a post-condition regarding an information entity, the assumption is that this information entity has not changed compared to what is stated in the pre-condition. An example is given here below:

subscriptionDeleted OR allSubscriptionDeleted

Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the post-condition are provided in a table. An example of such a table is given here below:

Assertion Name	Definition
subscriptionDeleted	The ntfSubscription identified by subscriptionId input parameter is no more involved in a subscription relationship with the ntfSubscriber identified by the managerReference input parameter and has been deleted. If this ntfSubscriber has no more ntfSubscription, it is deleted as well.
allSubscriptionDeleted	In the case subscriptionId input parameter was absent, the ntfSubscriber identified by the managerReference input parameter is no more involved in any subscription relationship and is deleted, the corresponding ntfSubscription have been deleted as well.

Y6.b.a.6 Exceptions

List of exceptions that can be raised by the operation. Each element is a tuple (exceptionName, condition, ReturnedInformation, exitState).

Y6.b.a.6.c exceptionName

ExceptionName is the name of an exception.

"c" represents a number, starting at 1 and increasing by 1 with each new definition of an exception.

This information is provided in a table. An example of such a table is given here below:

Exception Name	Definition
ope_failed_existing_subscription	**Condition:** (notificationCategoriesNotAllSubscribed OR notificationCategoriesParameterAbsentAndNotAllSubscribed) not verified. **Returned information:** output parameter status is set to OperationFailedExistingSubscription. **Exit state:** Entry State.

Y6.b.a.7 Constraints

This subclause presents constraints for the operation or its parameters.

This subclause does not need to be present when there are no constraints to define.

Y6.b.a Notification NotificationName (supportQualifier)

NotificationName is the name of the notification followed by a qualifier indicating whether the notification is Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO) or SS-Conditional (C).

"a" represents a number, starting at 1 and increasing by 1 with each new definition of a notification.

Y6.b.a.1 Definition

This subclause is written in natural language.

Information on traceability back to one or more requirements supported by this notification should also be defined here, in the following form:

Referenced TS	Requirement label	Comment
3GPP TS 32.xyz [xy]	REQ-SM-CON-23	*Optional clarification*
3GPP TS 32.xyz [xy]	REQ-SM-FUN-11	*Optional clarification*

Y6.b.a.2 Input parameters

List of input parameters of the notification. Each element is a tuple (Parameter Name, Qualifiers, Matching Information / Information Type (see [6]) (Note 1) and an optional list of Legal Values supported by the parameter, Comment).

The column "Qualifiers" contains the two qualifiers, Support Qualifier and Filtering Qualifier, separated by a comma. The Support Qualifier indicates whether the attribute is Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C). The Filtering Qualifier indicates whether the parameter of the notification can be filtered or not. Values are Yes (Y) or No (N).

This information is provided in a table. An example of such a table is given here below:

Parameter Name	Qualifiers	Matching Information / Information Type / Legal Values	Comment
managerReference	M,Y	ntfSubscriber.ntfManagerReference / STRING / --	It specifies the reference of IRPManager to which notifications shall be sent.
alarmType	M,Y	AlarmInformation.eventType / ENUMERATED / "Communications Alarm": a communication error alarm. "Processing Error Alarm": a processing error alarm. "Environmental Alarm": an environmental violation alarm. "Quality Of Service Alarm": a quality of service violation alarm. "Equipment Alarm": an alarm related to equipment malfunction.	

Note 1: Information Type qualifies the parameter of Parameter Name. In the case where the Legal Values can be enumerated, each element is a pair (Legal Value Name, Legal Value Semantics), unless a Legal Value Semantics applies to several values in which case the definition is provided only once. When the Legal Values cannot be enumerated, the list of Legal Values is defined by a single definition.

Y6.b.a.3 Triggering event

The triggering event for the notification to be sent is the transition from the information state defined by the "from state" subclause to the information state defined by the "to state" subclause.

Y6.b.a.3.1 From state

This subclause is a collection of assertions joined by AND, OR, and NOT logical operators. An example is given here below:

`alarmMatched` AND `alarmInformationNotCleared`

Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the state "from state" are provided in a table. An example of such a table is given here below:

Assertion Name	Definition
`alarmMatched`	The matching-criteria-attributes of the newly generated network alarm has values that are identical (matches) with ones in one `AlarmInformation` in `AlarmList`..
`alarmInformationNotCleared`	The `perceivedSeverity` of the newly generated network alarm is not `Cleared`.

Y6.b.a.3.2 To state

This subclause is a collection of assertions joined by AND, OR and NOT logical operators. When nothing is said in a to-state regarding an information entity, the assumption is that this information entity has not changed compared to what is stated in the from-state. An example is given here below:

resetAcknowledgementInformation AND perceivedSeverityUpdated

Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the state "to state" are provided in a table. An example of such a table is given here below:

Assertion Name	Definition
`resetAcknowledgementInformation`	The matched `AlarmInformation` identified in inv_alarmMatched in pre-condition has been updated according to the following rule: `ackTime`, `ackUserId` and `ackSystemId` are updated to contain no information; `ackState` is updated to "unacknowledged".
`perceivedSeverityUpdated`	The `perceivedSeverity` attribute of matched `AlarmInformation` identified in inv_alarmMatched in pre-condition has been updated.

Y6.b.a.4 Constraints

This subclause presents constraints for the notification or its parameters.

This subclause does not need to be present when there are no constraints to define.

Y6.c Scenario

This subclause contains one or more sequence diagrams, each describing a possible scenario. These shall be UML compliant sequence diagrams. This is an optional subclause.

Annex A (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2013-06	SA#60	SP-130304	001	1	Correct the description of the attribute definition table	11.0.0	11.1.0
2013-09	SA#61	SP-130433	002	1	Correction of ambiguous statements	11.1.0	11.2.0
2013-12	SA#62	SP-130614	003	1	Move instruction to the correct subclause	11.2.0	11.3.0
2014-06	SA#64	SP-140359	004	-	remove the feature support statements	11.3.0	11.4.0
2014-10	-	-	-	-	Update to Rel-12 version (MCC)	11.4.0	**12.0.0**
2016-01	-	-	-	-	Update to Rel-13 version (MCC)	12.0.0	**13.0.0**

Change history							
Date	**Meeting**	**TDoc**	**CR**	**Rev**	**Cat**	**Subject/Comment**	**New version**
2016-12	SA#74	SP-160855	0005	-	B	Include Interface IRP template	14.0.0
2017-04						Various editorial fixes (MCC)	14.0.1
2018-06	SA#80	SP-180423	0009	1	A	Align terminology	14.1.0
2018-06	-	-	-	-	-	Update to Rel-15 version (MCC)	**15.0.0**
2018-07						Correccion in history table (MCC)	**15.0.1**
2020-07	-	-	-	-	-	Update to Rel-16 version (MCC)	**16.0.0**
2022-03	-	-	-	-	-	Update to Rel-17 version (MCC)	**17.0.0**

History

Document history		
V17.0.0	April 2022	Publication