3GPP TS 28.623 V16.20.0 (2025-06)

Technical Specification

3rd Generation Partnership Project;

Technical Specification Group Services and System Aspects;

Telecommunication management;

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Solution Set (SS) definitions

(Release 16)



The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP.   
The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented.   
This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification.  
Specifications and reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices.

Keywords

Generic, NRM, IRP, Converged Management

***3GPP***

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis

Valbonne - FRANCE

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

Internet

http://www.3gpp.org

***Copyright Notification***

No part may be reproduced except as authorized by written permission.  
The copyright and the foregoing restriction extend to reproduction in all media.

© 2025 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).

All rights reserved.

UMTS™ is a Trade Mark of ETSI registered for the benefit of its members

3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  
LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners

GSM® and the GSM logo are registered and owned by the GSM Association

Contents

Foreword 5

Introduction 5

1 Scope 6

2 References 6

3 Definitions and abbreviations 7

3.1 Definitions 7

3.2 Abbreviations 7

4 Solution Set (SS) definitions 7

Annex A (normative): CORBA Solution Set 8

A.0 General 8

A.1 Architectural features 8

A.1.1 Syntax for Distinguished Names 8

A.1.2 Rules for NRM extensions 8

A.1.2.0 Introduction 8

A.1.2.1 Allowed extensions 8

A.1.2.2 Extensions not allowed 8

A.2 Mapping 10

A.2.1 General mapping 10

A.2.2 Information Object Class (IOC) mapping 10

A.2.2.1 IOC SubNetwork 10

A.2.2.2 IOC ManagedElement 10

A.2.2.3 IOC MeContext 10

A.2.2.4 IOC ManagementNode 11

A.2.2.5 IOC VsDataContainer 11

A.2.2.6 IOC ManagedFunction 11

A.2.2.7 IOC IRPAgent 11

A.2.2.8 IOC Top 11

A.2.2.9 IOC Link 12

A.2.2.10 IOC EP\_RP 12

A.2.2.11 IOC ThresholdMonitoringCapability 12

A.2.2.12 IOC ThresholdMonitor 12

A.2.2.13 IOC TraceJob 13

A.3 Solution Set (SS) definitions 14

A.3.1 IDL definition structure 14

A.3.2 IDL specification "GenericNetworkResourcesIRPSystem.idl" 14

A.3.3 IDL specification "GenericNetworkResourcesNRMDefs.idl" 17

Annex B (normative): XML Definitions 20

B.0 General 20

B.1 Architectural features 20

B.1.0 Introduction 20

B.1.1 Syntax for Distinguished Names 20

B.2 Mapping 20

B.2.1 General mapping 20

B.2.2 Information Object Class (IOC) mapping 20

B.3 Solution Set (SS) definitions 21

B.3.1 XML definition structure 21

B.3.2 Graphical Representation 21

B.3.3 XML schema "genericNrm.xsd" 22

Annex C (normative): OpenAPI definitions 32

C.1 General 32

C.2 Void 32

C.3 Void 32

C.4 Solution Set (SS) definitions 32

C.4.1 Void 32

C.4.2 Void 32

C.4.2a OpenAPI document "TS28623\_ComDefs.yaml" 32

C.4.3 OpenAPI document "TS28623\_GenericNrm.yaml" 35

Annex D (normative): YANG definitions 58

D.1 General 58

D.2 Modules 59

D.2.1 module \_3gpp-common-ep-rp.yang 59

D.2.2 module \_3gpp-common-managed-element.yang 60

D.2.3 module \_3gpp-common-managed-function.yang 63

D.2.4 module \_3gpp-common-measurements.yang 67

D.2.5 module \_3gpp-common-subnetwork.yang 74

D.2.6 module \_3gpp-common-top.yang 77

D.2.6a module \_3gpp-common-subscription-control.yang 78

D.2.7 module \_3gpp-common-yang-extensions.yang 81

D.2.8 module \_3gpp-common-yang-types.yang 82

D.2.9 module \_3gpp-common-fm.yang 89

D.3 Void 112

D.4 Mount information 112

Annex E (informative): Change history 112

# 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.

# Introduction

The present document is part of a TS-family covering the 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; as identified below:

28.621 Generic Network Resource Model (NRM) Integration Reference Point (IRP); Requirements.

28.622 Generic Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS).

**28.623 Generic Network Resource Model (NRM)** **Integration Reference Point (IRP); Solution Set (SS) definitions.**

# 1 Scope

The TS 28.62x-series (Generic Network Resources IRP) define an Integration Reference Point (IRP) through which an "IRPAgent" (typically an Element Manager or Network Element) can communicate Network Management related information to one or several "IRPManagers" (typically Network Managers).

This TS-family specifies a generic Network Resource Model, NRM (also referred to as a Management Information Model - MIM) with definitions of Information Object Classes (IOCs) and Managed Object Classes (MOCs).

The present document specifies the Solution Set definition for the Generic NRM IRP.

The Solution Set definition is related to 3GPP TS 28.622. For deployment scenarios using the IRP framework the latest Rel-14 version of TS 28.623 is applicable.

# 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 32.600: "Telecommunication management; Configuration Management (CM); Concept and high-level requirements".

[4] 3GPP TS 28.622: “Generic Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS)”.

[5] 3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects".

[6] Void

[7] 3GPP TS 32.616: "Telecommunication management; Configuration Management (CM); Bulk CM Integration Reference Point (IRP); Solution Set (SS) definitions".

[8] W3C REC-xml11-20060816: "Extensible Markup Language (XML) 1.1 (Second Edition)".

[9] Void.

[10] W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures.

[11] W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes.

[12] W3C REC-xml-names-20060816: "Namespaces in XML 1.1 (Second Edition)".

[13] Void

[14] 3GPP TS 32.160: "Management and orchestration; Management Service Template".

[15] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[16] IETF RFC 8528: "YANG Schema Mount".

# 3 Definitions and abbreviations

## 3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [15], 3GPP TS 32.101 [1], 3GPP TS 32.102 [2], 3GPP TS 32.600 [3], 3GPP TS 28.622 [4] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [15] and 3GPP TS 32.101 [1], 3GPP TS 32.102 [2] and 3GPP TS 32.600 [3] and 3GPP TS 28.622 [4].

## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [15], 3GPP TS 32.600 [3] 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 [15] and 3GPP TS 32.600 [3].

JSON JavaScript Object Notation

SS Solution Set

# 4 Solution Set (SS) definitions

This specification defines the following 3GPP Generic NRM IRP Solution Set Definitions:

- 3GPP Generic NRM IRP CORBA SS (Annex A).

- 3GPP Generic NRM IRP XML Definitions (Annex B).

- 3GPP Generic NRM IRP JSON Definitions (Annex C).

- 3GPP Generic NRM IRP YANG Definitions (Annex D).

Annex A (normative):  
CORBA Solution Set

# A.0 General

This annex contains the CORBA Solution Set for the IRP whose semantics is specified in Generic NRM IRP: Information Service (3GPP TS 28.622 [4]).

# A.1 Architectural features

The overall architectural feature of Generic Network Resources IRP is specified in 3GPP TS 28.622 [4].   
This clause specifies features that are specific to the CORBA SS.

## A.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [5].

## A.1.2 Rules for NRM extensions

### A.1.2.0 Introduction

This clause discusses how the models and IDL definitions provided in the present document can be extended for a particular implementation and still remain compliant with 3GPP SA5's specifications.

### A.1.2.1 Allowed extensions

Vendor-specific MOCs may be supported. The vendor-specific MOCs may support new types of attributes.   
The 3GPP SA5-specified notifications may be issued referring to the vendor-specific MOCs and vendor-specific attributes. New MOCs shall be distinguishable from 3GPP SA5 MOCs by name. 3GPP SA5-specified and vendor-specific attributes may be used in vendor-specific MOCs. Vendor-specific attribute names shall be distinguishable from existing attribute names.

NRM MOCs may be subclassed. Subclassed MOCs shall maintain the specified behaviour of the 3GPP SA5's superior classes. They may add vendor-specific behaviour with vendor-specific attributes. When subclassing, naming attributes cannot be changed. The subclassed MOC shall support all attributes of its superior class. Vendor-specific attributes cannot be added to 3GPP SA5 NRM MOCs without subclassing.

When subclassing, the 3GPP SA5-specified containment rules and their specified cardinality shall still be followed.   
As an example, ManagementNode (or its subclasses) shall be contained under SubNetwork (or its subclasses).

Managed Object Instances may be instantiated as CORBA objects. This requires that the MOCs be represented in IDL. 3GPP SA5's NRM MOCs are not currently specified in IDL, but may be specified in IDL for instantiation or subclassing purposes. However, management information models should not require that IRPManagers access the instantiated managed objects other than through supported methods in the present document.

Extension rules related to notifications (Notification categories, Event Types, Extended Event Types etc.) are for further study.

### A.1.2.2 Extensions not allowed

The IDL specifications in the present document cannot be edited or altered. Any additional IDL specifications shall be specified in separate IDL files.

IDL interfaces (note: not MOCs) specified in the present document may not be subclassed or extended. New interfaces may be defined with vendor-specific methods.

# A.2 Mapping

## A.2.1 General mapping

Attributes modelling associations as defined in the NRM (here also called "reference attributes") are in this SS mapped to attributes. The names of the reference attributes in the NRM are mapped to the corresponding attribute names in the MOC. When the cardinality for an association is 0..1 or 1..1 the datatype for the reference attribute is defined as an MOReference. The value of an MO reference contains the distinguished name of the associated MO. When the cardinality for an association allows more than one referred MO, the reference attribute will be of type MOReferenceSet, which contains a sequence of MO references.

## A.2.2 Information Object Class (IOC) mapping

This Solution Set supports reference attributes for relations other than containment relations between objects. Reference attributes are therefore introduced in each MOC where needed.

### A.2.2.1 IOC SubNetwork

Mapping from NRM IOC SubNetwork attributes to SS equivalent MOC SubNetwork attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| dnPrefix | dnPrefix | string |
| userLabel | userLabel | string |
| userDefinedNetworkType | userDefinedNetworkType | string |
| setOfMcc | setOfMcc | GenericNetworkResourcesIRPSystem::AttributeTypes::StringSet |

### A.2.2.2 IOC ManagedElement

Mapping from NRM IOC ManagedElement attributes and association roles to SS equivalent MOC ManagedElement attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| dnPrefix | dnPrefix | string |
| userLabel | userLabel | string |
| locationName | locationName | string |
| vendorName | vendorName | string |
| userDefinedState | userDefinedState | string |
| managedElementType | managedElementType | GenericNetworkResourcesIRPSystem::AttributeTypes::StringSet |
| managedBy | managedBy | GenericNetworkResourcesIRPSystem::AttributeTypes::MOReferenceSet |
| swVersion | swVersion | string |

### A.2.2.3 IOC MeContext

Mapping from NRM IOC MeContext attributes to SS equivalent MOC MeContext attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| dnPrefix | dnPrefix | string |

### A.2.2.4 IOC ManagementNode

Mapping from NRM IOC ManagementNode attributes and association roles to SS equivalent MOC ManagementNode attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| userLabel | userLabel | string |
| locationName | locationName | string |
| vendorName | vendorName | string |
| userDefinedState | userDefinedState | string |
| managedElements | managedElements | GenericNetworkResourcesIRPSystem::AttributeTypes::MOReferenceSet |
| swVersion | swVersion | string |

### A.2.2.5 IOC VsDataContainer

Mapping from NRM IOC VsDataContainer attributes and association roles to SS equivalent MOC VsDataContainer attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| vsDataType | vsDataType | string |
| vsData | vsData | any |
| vsDataFormatVersion | vsDataFormatVersion | string |

### A.2.2.6 IOC ManagedFunction

Mapping from NRM IOC ManagedFunction attributes and association roles to SS equivalent MOC ManagedFunction attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| peeParametersList | peeParametersList | GenericNetworkResourcesIRPSystem::AttributeTypes:: PEEParametersListType |
| userLabel | userLabel | string |
| vnfParametersList | vnfParametersList | GenericNetworkResourcesIRPSystem::AttributeTypes:: VNFParametersListType |

### A.2.2.7 IOC IRPAgent

Mapping from NRM IOC IRPAgent attributes to SS equivalent MOC IRPAgent attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| systemDN | systemDN | string |

### A.2.2.8 IOC Top

Mapping from NRM IOC Top attributes to SS equivalent attributes in all MOCs

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| objectClass | CLASS | string |
| objectInstance | No direct mapping |  |

### A.2.2.9 IOC Link

Mapping from NRM IOC Link attributes to SS equivalent MOC IRPAgent attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| userLabel (see note 2) | userLabel | string |
| aEnd | aEnd | GenericNetworkResourcesIRPSystem::AttributeTypes::MOReference |
| zEnd | zEnd | GenericNetworkResourcesIRPSystem::AttributeTypes::MOReference |
| linkType | linkType | LinkTypeType |
| protocolName | protocolName | string |
| protocolVersion | protocolVersion | string |

NOTE 1: Void.

NOTE 2: Void.

### A.2.2.10 IOC EP\_RP

Mapping from NRM IOC EP\_RP attributes to SS equivalent MOC EP\_RP attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| id | id | string |
| userLabel | userLabel | string |
| farEndEntity | farEndEntity | GenericNetworkResourcesIRPSystem::AttributeTypes::MOReference |

### A.2.2.11 IOC ThresholdMonitoringCapability

Mapping from NRM IOC ThresholdMonitoringCapability attributes to SS equivalent MOC ThresholdMonitoringCapability attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| supportedMonitoringGPs | supportedMonitoringGPs | GenericNetworkResourcesIRPSystem::AttributeTypes::LongSet |

### A.2.2.12 IOC ThresholdMonitor

Mapping from NRM IOC ThresholdMonitor attributes to SS equivalent MOC ThresholdMonitor attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| thresholdInfoList | thresholdInfoList | GenericNetworkResourcesIRPSystem::AttributeTypes::ThresholdInfoListType |
| monitoringGP | monitoringGP | long |
| monitoringNotifTarget | monitoringNotifTarget | string |
| monitoredIOCName | monitoredIOCName | string |
| monitoredObjectDNs | monitoredObjectDNs | GenericNetworkResourcesIRPSystem::AttributeTypes::DNListType |

### A.2.2.13 IOC TraceJob

Mapping from NRM IOC TraceJob attributes to SS equivalent MOC TraceJob attributes

| IS Attributes | SS Attributes | SS Type |
| --- | --- | --- |
| tjJobType | tjJobType | tjJobType-Type |
| tjListOfInterfaces | tjListOfInterfaces | tjListOfInterfaces-Type |
| tjListOfNeTypes | tjListOfNeTypes | tjListOfNeTypes-Type |
| tjPLMNTarget | tjPLMNTarget | tjPLMNTarget-Type |
| tjStreamingTraceConsumerURI | tjTraceConsumer | StreamingTraceConsumerURI-Type |
| tjTraceCollectionEntityAddress | tjTraceConsumer | TraceCollectionEntityAddress-Type |
| tjTraceDepth | tjTraceDepth | tjTraceDepth-Type |
| tjTraceReference | tjTraceReference | tjTraceReference-Type |
| tjTraceReportingFormat | tjTraceReportingFormat | tjTraceReportingFormat-Type |
| tjTraceTarget | tjTraceTarget | tjTraceTarget-Type |
| tjTriggeringEvent | tjTriggeringEvent | tjTriggeringEvent-Type |
| tjMDTAnonymizationOfData | tjMDTAnonymizationOfData | tjMDTAnonymizationOfData-Type |
| tjMDTAreaConfigurationForNeighCell | tjMDTAreaConfigurationForNeighCell | tjMDTAreaConfigurationForNeighCell-Type |
| tjMDTAreaScope | tjMDTAreaScope | tjMDTAreaScope-Type |
| tjMDTCollectionPeriodRrmLte | tjMDTCollectionPeriodRrmLte | tjMDTCollectionPeriodRrmLte-Type |
| tjMDTCollectionPeriodRrmUmts | tjMDTCollectionPeriodRrmUmts | tjMDTCollectionPeriodRrmUmts-Type |
| tjMDTCollectionPeriodRrmNR | tjMDTCollectionPeriodRrmNR | tjMDTCollectionPeriodRrmNR-Type |
| tjMDTEventListForTriggeredMeasurement | tjMDTEventListForTriggeredMeasurement | tjMDTEventListForTriggeredMeasurement-Type |
| tjMDTEventThreshold | tjMDTEventThreshold | tjMDTEventThreshold-Type |
| tjMDTListOfMeasurements | tjMDTListOfMeasurements | tjMDTListOfMeasurements-Type |
| tjMDTLoggingDuration | tjMDTLoggingDuration | tjMDTLoggingDuration-Type |
| tjMDTLoggingInterval | tjMDTLoggingInterval | tjMDTLoggingInterval-Type |
| tjMDTMBSFNAreaList | tjMDTMBSFNAreaList | tjMDTMBSFNAreaList-Type |
| tjMDTMeasurementPeriodLTE | tjMDTMeasurementPeriodLTE | tjMDTMeasurementPeriodLTE-Type |
| tjMDTMeasurementPeriodUMTS | tjMDTMeasurementPeriodUMTS | tjMDTMeasurementPeriodUMTS-Type |
| tjMDTMeasurementQuantity | tjMDTMeasurementQuantity | tjMDTMeasurementQuantity-Type |
| tjMDTPLMList | tjMDTPLMList | tjMDTPLMList-Type |
| tjMDTPositioningMethod | tjMDTPositioningMethod | tjMDTPositioningMethod-Type |
| tjMDTReportAmount | tjMDTReportAmount | tjMDTReportAmount-Type |
| tjMDTReportingTrigger | tjMDTReportingTrigger | tjMDTReportingTrigger-Type |
| tjMDTReportInterval | tjMDTReportInterval | tjMDTReportInterval-Type |
| tjMDTReportType | tjMDTReportType | tjMDTReportType-Type |
| tjMDTSensorInformation | tjMDTSensorInformation | tjMDTSensorInformation-Type |
| tjMDTTraceCollectionEntityID | tjMDTTraceCollectionEntityID | tjMDTTraceCollectionEntityID-Type |

# A.3 Solution Set (SS) definitions

## A.3.1 IDL definition structure

Clause A.3.2 defines the types which are used by the Generic NRM IRP.

Clause A.3.3 defines the MO classes for the Generic NRM IRP.

## A.3.2 IDL specification "GenericNetworkResourcesIRPSystem.idl"

//File: GenericNetworkResourcesIRPSystem.idl

#ifndef \_GENERIC\_NETWORK\_RESOURCES\_IRP\_SYSTEM\_IDL\_

#define \_GENERIC\_NETWORK\_RESOURCES\_IRP\_SYSTEM\_IDL\_

// This statement must appear after all include statements

#pragma prefix "3gppsa5.org"

module GenericNetworkResourcesIRPSystem

{

/\*\*

\* The format of Distinguished Name (DN) is specified in "Name Convention

\* for Managed Objects (3GPP TS 32.300 [5])".

\*/

typedef string DN;

/\*\*

\* This module adds datatype definitions for types

\* used in the NRM which are not basic datatypes defined

\* already in CORBA.

\*/

module AttributeTypes

{

/\*\*

\* An MO reference refers to an MO instance.

\* "otherMO" contains the distinguished name of the referred MO.

\* A conceptual "null" reference (meaning no MO is referenced)

\* is represented as an empty string ("").

\*

\*/

struct MOReference

{

DN otherMO;

};

/\*\*

\* MOReferenceSet represents a set of MO references.

\* This type is used to hold 0..n MO references.

\* A referred MO is not allowed to be repeated (therefore

\* it is denoted as a "Set")

\*/

typedef sequence<MOReference> MOReferenceSet;

/\*\*

\* A set of strings.

\*/

typedef sequence<string> StringSet;

/\*\*

\* A set of long.

\*/

typedef sequence<long> LongSet;

/\*

\* The LinkListSet represents the Link\_X\_Y objects (or subclasses of

\* Link\_X\_Y objects) that have a relationship with this object instance.

\* Each Link\_X\_Y object models interface(s) between objects of class X and

\* Y. The object containing this attribute must either be a class of type X,

\* Y, XFunction, YFunction or a subclass of one of those classes. The

\* LinkListSet may be empty, or there may be no instances for a particular

\* Link\_X\_Y class name.

\*/

typedef MOReferenceSet LinkListSet;

/\*\*

\* VNFParameters includes several attributes of a VNF instance.

\* The detailed definition of the attributes, see clause 4.4.1 of [4].

\*/

struct VNFParameters

{

string vnfInstanceId;

string vnfdId;

string flavourId;

boolean autoScalable;

};

/\*\*

\* VNFParametersListType represents a list of VNFParameters.

\* The detailed definition of vnfParametersListType, see clause 4.4.1 of [4].

\*/

typedef sequence<VNFParameters> VNFParametersListType;

struct PEEParameters

{

string siteIdentification;

float siteLatitude;

float siteLongitude;

string siteDescription;

string equipmentType;

string environmentType;

string powerInterface;

};

/\*\*

\* PEEParametersListType represents a list of PEEParameters.

\* The detailed definition of PEEParametersListType, see clause 4.4.1 of [4].

\*/

typedef sequence<PEEParameters> PEEParametersListType;

typedef any ThresholdValueType;

enum Direction {INCREASING, DECREASING};

union HysteresisType switch(boolean)

{

case TRUE: long long\_value;

case FALSE: float float\_value;

};

struct ThresholdPackElement

{

ThresholdValueType thresholdValue;

short thresholdLevel;

HysteresisType hysteresis;

};

typedef sequence<ThresholdPackElement> ThresholdPackType;

struct ThresholdInfo

{

string measurementType;

Direction direction\_;

ThresholdPackType thresholdPack;

};

typedef sequence<ThresholdInfo> ThresholdInfoListType;

};

/\*\*

\* This module adds datatype definitions for PM Control

\*/

module PMControlTypes

{

Struct Measurements

{

measurementTypes StringSet,

gPs LongSet

};

typedef sequence <Measurements> Measurements;

enum PMAdministrativeStateType

{

LOCKED,

SHUTTINGDOWN,

UNLOCKED

};

enum PMOperationalStateType

{

ENABLED,

DISABLED

};

typedef MOReferenceSet ManagedObjectDNsType;

typedef MOReferenceSet ManagedObjectDNsBasicType;

typedef integer DefaultFileBasedGPType;

typedef integer DefaultFileReportPeriodType;

typedef string DefaultFileLocationType;

typedef integer DefaultStreamBasedGPType;

typedef string DefaultStreamTargetType;

typedef integer FileBasedGPType;

typedef integer FileReportingPeriodType;

typedef string FileLocationType;

typedef integer StreamBasedGPType;

typedef string StreamTargetType;

};

};

#endif // \_GENERIC\_NETWORK\_RESOURCES\_IRP\_SYSTEM\_IDL\_

## A.3.3 IDL specification "GenericNetworkResourcesNRMDefs.idl"

//File: GenericNetworkResourcesNRMDefs.idl

#ifndef \_GENERIC\_NETWORK\_RESOURCES\_NRM\_DEFS\_IDL\_

#define \_GENERIC\_NETWORK\_RESOURCES\_NRM\_DEFS\_IDL\_

// This statement must appear after all include statements

#pragma prefix "3gppsa5.org"

/\*\*

\* This module defines constants for each MO class name and

\* the attribute names for each defined MO class.

\*/

module GenericNetworkResourcesNRMDefs

{

/\*\*

\* Definitions for MO class Top

\*/

interface Top

{

// Attribute Names

//

const string CLASS = "Top";

};

/\*\*

\* Definitions for MO class SubNetwork

\*/

interface SubNetwork : Top

{

const string CLASS = "SubNetwork";

// Attribute Names

//

const string id = "id";

const string dnPrefix = "dnPrefix";

const string userLabel = "userLabel";

const string userDefinedNetworkType = "userDefinedNetworkType";

const string setOfMcc = "setOfMcc";

};

/\*\*

\* Definitions for MO class ManagedElement

\*/

interface ManagedElement : Top

{

const string CLASS = "ManagedElement";

// Attribute Names

//

const string id = "id";

const string dnPrefix = "dnPrefix";

const string managedElementType = "managedElementType";

const string userLabel = "userLabel";

const string vendorName = "vendorName";

const string userDefinedState ="userDefinedState";

const string locationName ="locationName";

const string managedBy = "managedBy";

const string swVersion = "swVersion";

};

/\*\*

\* Definitions for MO class MeContext

\*/

interface MeContext : Top

{

const string CLASS = "MeContext";

// Attribute Names

//

const string id = "id";

const string dnPrefix = "dnPrefix";

};

/\*\*

\* Definitions for MO class ManagementNode

\*/

interface ManagementNode : Top

{

const string CLASS = "ManagementNode";

// Attribute Names

//

const string id = "id";

const string userLabel = "userLabel";

const string vendorName = "vendorName";

const string userDefinedState = "userDefinedState";

const string locationName = "locationName";

const string managedElements = "managedElements";

const string swVersion = "swVersion";

};

/\*\*

\* Definitions for abstract MO class ManagedFunction

\*

\*/

interface ManagedFunction : Top

{

const string CLASS = "ManagedFunction";

// Attribute Names

//

const string id = "id";

const string peeParametersList = "peeParametersList";

const string userLabel = "userLabel";

const string vnfParametersList = "vnfParametersList";

};

/\*\*

\* Definitions for MO class IRPAgent

\*/

interface IRPAgent : Top

{

const string CLASS = "IRPAgent";

// Attribute Names

//

const string id = "id";

const string systemDN = "systemDN";

};

/\*\*

\* Definitions for abstract MO class Link

\* This inherits from ManagedFunction

\* The attributes aEnd and zEnd are populated with the DNs  
 \* of the entities associated via the link class.  
 \* The aEnd takes the DN of the 1st entity in alphabetical order,

\* the zEnd takes the 2nd entity in alphabetical order of the class

\* names.

\*/

interface Link : ManagedFunction

{

const string CLASS = "Link";

// Attribute Names

//

const string aEnd = "aEnd";

const string zEnd = "zEnd";

const string linkType = "linkType";

const string protocolName = "protocolName";

const string protocolVersion = "protocolVersion";

};

/\*\*

\* Definitions for MO class VsDataContainer

\*/

interface VsDataContainer : Top

{

const string CLASS = "VsDataContainer";

// Attribute Names

//

const string id = "id";

const string vsDataType = "vsDataType";

const string vsData = "vsData";

const string vsDataFormatVersion = "vsDataFormatVersion";

};

/\*\*

\* Definitions for abstract MO class EP\_RP

\*/

interface EP\_RP : Top

{

const string CLASS = "EP\_RP";

// Attribute Names

//

const string farEndEntity = "farEndEntity";

const string id = "id";

const string userLabel = "userLabel";

};

/\*\*

\* Definitions for MO class ThresholdMonitoringCapability

\*/

interface ThresholdMonitoringCapability : Top

{

const string CLASS = "ThresholdMonitoringCapability";

// Attribute Names

//

const string supportedMonitoringGPs = "supportedMonitoringGPs";

};

/\*\*

\* Definitions for MO class ThresholdMonitor

\*/

interface ThresholdMonitor : Top

{

const string CLASS = "ThresholdMonitor";

// Attribute Names

//

const string thresholdInfoList = "thresholdInfoList";

const string monitoringGP = "monitoringGP";

const string monitoringNotifTarget = "monitoringNotifTarget";

const string monitoredIOCName = "monitoredIOCName";

const string monitoredObjectDNs = "monitoredObjectDNs";

};

/\*\*

\* This module adds datatypes definitions for the Link Class

\* These attributes are not the basic datatypes already defined

\*/

module LinkAttributeTypes

{

enum LinkType

{

SIGNALLING,

BEARER,

OAM\_AND\_P,

OTHER

};

typedef sequence <LinkType> LinkTypeType;

};

};

#endif // \_GENERIC\_NETWORK\_RESOURCES\_NRM\_DEFS\_IDL\_

Annex B (normative):  
XML Definitions

# B.0 General

This annex contains the XML Definitions for the Generic NRM IRP as it applies to Itf-N, in accordance with Generic NRM IRP IS definitions TS 28.622 [4].

The XML file formats are based on XML W3C REC-xml11-20060816 [8], W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures [10] W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes [11] and W3C REC-xml-names-20060816 [12] standards.

# B.1 Architectural features

## B.1.0 Introduction

The overall architectural feature of Generic Network Resources IRP is specified in 3GPP TS 28.622 [4].

This clause specifies features that are specific to the Schema definitions.

## B.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [5].

# B.2 Mapping

## B.2.1 General mapping

An IOC maps to an XML element of the same name as the IOC's name in the IS. An IOC attribute maps to a sub-element of the corresponding IOC's XML element, and the name of this sub-element is the same as the attribute's name in the IS.

## B.2.2 Information Object Class (IOC) mapping

The mapping is not present in the current version of this specification.

# B.3 Solution Set (SS) definitions

## B.3.1 XML definition structure

The overall description of the file format of configuration data XML files is provided by 3GPP TS 28.616 [7].

Annex B.3.3 of the present document defines the NRM-specific XML schema genericNrm.xsd for the Generic Network Resources IRP NRM defined in 3GPP TS 28.622 [4].

XML schema genericNrm.xsd explicitly declares NRM-specific XML element types for the related NRM.

The definition of those NRM-specific XML element types complies with the generic mapping rules defined in 3GPP TS 28.616 [7], with the following exception: as defined in 3GPP TS 28.616 [7], the vsData XML element type has an empty XML content.

Additionally, XML schema genericNrm.xsd also provides the following global XML declarations and definitions:

- XML complex type NrmClass: derivation base type (see [8], [10] and [11]) for all NRM class associated XML element types (see 3GPP TS 28.616 [7]);

- XML element type vsData: derivation base type (see [8], [10] and [11]) for all vendor-specific XML element types (see 3GPP TS 28.616 [7]);

- XML element type SubNetworkOptionallyContainedNrmClass: substitution group head (see [8], [10] and [11]) for all XML element types associated to further NRM classes optionally contained under SubNetwork NRM class;

- XML element type ManagedElementOptionallyContainedNrmClass: substitution group head (see [8], [10] and [11]) for all XML element types associated to further NRM classes optionally contained under ManagedElement NRM class.

## B.3.2 Graphical Representation

The graphical representation is not present in the current version of this specification.

## B.3.3 XML schema "genericNrm.xsd"

<?xml version="1.1" encoding="UTF-8"?>

<!--

3GPP TS 28.623 Generic Network Resources IRP

Bulk CM Configuration data file NRM-specific XML schema

genericNrm.xsd

-->

<schema

targetNamespace="http://www.3gpp.org/ftp/specs/archive/28\_series/28.623#genericNrm"

elementFormDefault="qualified"

attributeFormDefault="unqualified"

xmlns="http://www.w3.org/2001/XMLSchema"

xmlns:xn="http://www.3gpp.org/ftp/specs/archive/28\_series/28.623#genericNrm"

xmlns:sp="http://www.3gpp.org/ftp/specs/archive/28\_series/28.629#sonPolicyNrm"

>

<import namespace="http://www.3gpp.org/ftp/specs/archive/28\_series/28.629#sonPolicyNrm"/>

<!-- Base XML type for all NRM class associated XML elements -->

<complexType name="NrmClass">

<attribute name="id" type="string" use="required"/>

<attribute name="modifier" use="optional">

<simpleType>

<restriction base="string">

<enumeration value="create"/>

<enumeration value="delete"/>

<enumeration value="update"/>

</restriction>

</simpleType>

</attribute>

</complexType>

<!-- Generic Network Resources IRP NRM attribute related XML types -->

<simpleType name="dn">

<restriction base="string">

<maxLength value="400"/>

</restriction>

</simpleType>

<complexType name="dnList">

<sequence minOccurs="0" maxOccurs="unbounded">

<element name="dn" type="xn:dn"/>

</sequence>

</complexType>

<simpleType name="linkType">

<list>

<simpleType>

<restriction base="string">

<enumeration value="Signalling"/>

<enumeration value="Bearer"/>

<enumeration value="OAM\_AND\_P"/>

<enumeration value="Other"/>

</restriction>

</simpleType>

</list>

</simpleType>

<complexType name="linkListType">

<sequence minOccurs="0" maxOccurs="unbounded">

<element name="dn" type="xn:dn"/>

</sequence>

</complexType>

<complexType name="managedElementTypeListType">

<sequence minOccurs="0" maxOccurs="unbounded">

<element name="managedElementType" type="string"/>

</sequence>

</complexType>

<complexType name="vnfParametersListType">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="vnfInstanceId" type="string"/>

<element name="vnfdId" type="string" minOccurs="0"/>

<element name="flavourId" type="string" minOccurs="0"/>

<element name="autoScalable" type="boolean"/>

</sequence>

</complexType>

<simpleType name="latitude">

<restriction base="decimal">

<fractionDigits value="4"/>

<minInclusive value="-90.0000"/>

<maxInclusive value="90.0000"/>

</restriction>

</simpleType>

<simpleType name="longitude">

<restriction base="decimal">

<fractionDigits value="4"/>

<minInclusive value="-180.0000"/>

<maxInclusive value="180.0000"/>

</restriction>

</simpleType>

<complexType name="peeParametersListType">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="siteIdentification" type="string"/>

<element name="siteLatitude" type="xn:latitude" minOccurs="0"/>

<element name="siteLongitude" type="xn:longitude" minOccurs="0"/>

<element name="siteDescription" type="string"/>

<element name="equipmentType" type="string"/>

<element name="environmentType" type="string"/>

<element name="powerInterface" type="string"/>

</sequence>

</complexType>

<simpleType name="pMAdministrativeStateType">

<restriction base="string">

<enumeration value="LOCKED"/>

<enumeration value="SHUTTINGDOWN"/>

<enumeration value="UNLOCKED"/>

</restriction>

</simpleType>

<simpleType name="pMOperationalStateType">

<restriction base="string">

<enumeration value="ENABLED"/>

<enumeration value="DISABLED"/>

</restriction>

</simpleType>

<simpleType name="nFServiceType">

<restriction base="string">

<enumeration value="Namf\_Communication"/>

<enumeration value="Namf\_EventExposure"/>

<enumeration value="Namf\_MT"/>

<enumeration value="Namf\_Location"/>

<enumeration value="Nsmf\_PDUSession"/>

<enumeration value="Nsmf\_EventExposure"/>

<enumeration value="others"/>

</restriction>

</simpleType>

<simpleType name="usageStateType">

<restriction base="string">

<enumeration value="IDEL"/>

<enumeration value="ACTIVE"/>

<enumeration value="BUSY"/>

</restriction>

</simpleType>

<simpleType name="registrationStateType">

<restriction base="string">

<enumeration value="LOCKED"/>

<enumeration value="SHUTTING\_DOWN"/>

<enumeration value="UNLOCKED"/>

</restriction>

</simpleType>

<simpleType name="NFType">

<restriction base="string">

<enumeration value="NRF"/>

<enumeration value="UDM"/>

<enumeration value="AMF"/>

<enumeration value="SMF"/>

<enumeration value="AUSF"/>

<enumeration value="NEF"/>

<enumeration value="PCF"/>

<enumeration value="SMSF"/>

<enumeration value="NSSF"/>

<enumeration value="UDR"/>

<enumeration value="GMLC"/>

<enumeration value="5G EIR"/>

<enumeration value="SEPP"/>

<enumeration value="UPF"/>

<enumeration value="N3IWF"/>

<enumeration value="AF"/>

<enumeration value="UDSF"/>

<enumeration value="DN"/>

</restriction>

</simpleType>

<simpleType name="operationSemanticsType">

<restriction base="string">

<enumeration value="REQUEST\_RESPONSE"/>

<enumeration value="SUBSCRIBE\_NOTIFY"/>

</restriction>

</simpleType>

<complexType name="SAP">

<sequence>

<element name="host" type="xn:hostType"/>

<element name="port" type="integer"/>

</sequence>

</complexType>

<complexType name="hostType">

<sequence>

<element name="ipv4Address" type="string"/>

<element name="ipv6Address" type="string"/>

<element name="fqdn" type="string"/>

</sequence>

</complexType>

<complexType name="operationsList">

<sequence>

<element name="operation" type="xn:operationType" minOccurs="1" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="operationType">

<sequence>

<element name="name" type="string"/>

<element name="allowedNFTypes" type="xn:NFType"/>

<element name="operationSemantics" type="xn:operationSemanticsType"/>

</sequence>

</complexType>

<complexType name="MeasurementTypeList">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="measurementType" type="string"/>

</sequence>

</complexType>

<complexType name="GPList">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="gP" type="integer"/>

</sequence>

</complexType>

<complexType name="Measurements">

<sequence>

<element name="measurementTypes" type="xn:MeasurementTypeList"/>

<element name="GPs" type="xn:GPList"/>

</sequence>

</complexType>

<complexType name="MeasurementsList">

<sequence>

<element name="measurements" type="xn:Measurements" minOccurs="1" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="GPListType">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="GP" type="integer"/>

</sequence>

</complexType>

<complexType name="KPINameList">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="kPIName" type="string"/>

</sequence>

</complexType>

<complexType name="KPIs">

<sequence>

<element name="kPITypes" type="xn:KPINameList"/>

<element name="GPs" type="xn:GPList"/>

</sequence>

</complexType>

<complexType name="KPIsList">

<sequence>

<element name="kPIs" type="xn:KPIs" minOccurs="1" maxOccurs="unbounded"/>

</sequence>

</complexType>

<simpleType name="directionType">

<list>

<simpleType>

<restriction base="string">

<enumeration value="increasing"/>

<enumeration value="decreasing"/>

</restriction>

</simpleType>

</list>

</simpleType>

<complexType name="thrsholdPackType">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="thresholdPackElement" type="xn:thresholdPackElementType"/>

</sequence>

</complexType>

<complexType name="thresholdPackElementType">

<all>

<element name="thresholdValue" type="string"/>

<element name="thresholdLevel" type="integer"/>

<element name="hysteresis" type="decimal" minOccurs="0"/>

</all>

</complexType>

<complexType name="thresholdInfoType">

<all>

<element name="measurementType" type="string"/>

<element name="direction" type=" xn:directionType"/>

<element name="thresholdPack" type=" xn:thrsholdPackType"/>

</all>

</complexType>

<complexType name="thresholdInfoListType">

<sequence minOccurs="1" maxOccurs="unbounded">

<element name="ThresholdInfoElement" type="xn:thresholdInfoType"/>

</sequence>

</complexType>

<simpleType name="ScopeType">

<restriction base="string">

<enumeration value="BASE\_ONLY"/>

<enumeration value="BASE\_ALL"/>

<enumeration value="BASE\_NTH\_LEVEL"/>

<enumeration value="BASE\_SUBTREE"/>

</restriction>

</simpleType>

<complexType name="Scope">

<sequence>

<element name="scopeType" type="xn:ScopeType"/>

<element name="scopeLevel" type="integer" minOccurs="0"/>

</sequence>

</complexType>

<!-- Generic Network Resources IRP NRM class associated XML elements -->

<element name="SubNetwork">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="dnPrefix" minOccurs="0"/>

<element name="userLabel"/>

<element name="userDefinedNetworkType"/>

<element name="setOfMcc" minOccurs="0"/>

<element name="priority" type="integer" minOccurs="0"/>

<element name="measurementsList" type="xn:MeasurementsList" minOccurs="0"/>

<element name="kPIsList" type="xn:KPIsList" minOccurs="0"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:SubNetwork"/>

<element ref="xn:ManagedElement"/>

<element ref="xn:MeContext"/>

<element ref="xn:ManagementNode"/>

<element ref="xn:IRPAgent"/>

<element ref="xn:SubNetworkOptionallyContainedNrmClass"/>

<element ref="xn:VsDataContainer"/>

<element ref="xn:ThresholdMonitoringCapability"/>

<element ref="xn:ThresholdMonitor"/>

<element ref="xn:MeasurementControl"/>

<element ref="xn:NtfSubscriptionControl"/>

</choice>

<choice minOccurs="0" maxOccurs="1">

<element ref="sp:ESPolicies"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="ManagedElement">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="dnPrefix"/>

<element name="managedElementTypeList" type="xn:managedElementTypeListType" minOccurs="0"/>

<element name="userLabel"/>

<element name="vendorName"/>

<element name="userDefinedState"/>

<element name="locationName"/>

<element name="swVersion"/>

<element name="managedBy" type="xn:dnList" minOccurs="0"/>

<element name="priority" type="integer" minOccurs="0"/>

<element name="measurementsList" type="xn:MeasurementsList" minOccurs="0"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:IRPAgent"/>

<element ref="xn:ManagedElementOptionallyContainedNrmClass"/>

<element ref="xn:VsDataContainer"/>

<element ref="xn:ThresholdMonitoringCapability"/>

<element ref="xn:ThresholdMonitor"/>

<element ref="xn:MeasurementControl"/>

<element ref="xn:NtfSubscriptionControl"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="ManagedFunction">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="userLabel" type="string"/>

<element name="vnfParametersList" type="xn:vnfParametersListType"/>

<element name="peeParametersList" type="xn:peeParametersListType"/>

<element name="priority" type="integer" minOccurs="0"/>

<element name="measurementsList" type="xn:MeasurementsList" minOccurs="0"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:VsDataContainer"/>

<element ref="xn:EP\_RP"/>

<element ref="xn:ThresholdMonitoringCapability"/>

<element ref="xn:ThresholdMonitor"/>

<element ref="xn:MeasurementControl"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="ManagedNFService">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="userLabel" type="string"/>

<element name="nFServiceType" type="xn:nFServiceType"/>

<element name="AdministrativeState" type="xn:pMAdministrativeStateType"/>

<element name="OperationalState" type="xn:pMOperationalStateType"/>

<element name="usageState" type="xn:usageStateType"/>

<element name="registrationState" type="xn:registrationStateType"/>

<element name="sAP" type="xn:SAP" minOccurs="0"/>

<element name="operations" type="xn:operationsList" minOccurs="0"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:VsDataContainer"/>

<element ref="xn:ThresholdMonitoringCapability"/>

<element ref="xn:ThresholdMonitor"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="MeContext">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="dnPrefix" minOccurs="0"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:ManagedElement"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="ManagementNode">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="userLabel"/>

<element name="vendorName"/>

<element name="locationName"/>

<element name="managedElements" type="xn:dnList" minOccurs="0"/>

<element name="swVersion"/>

<element name="userDefinedState"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:IRPAgent"/>

<element ref="xn:VsDataContainer"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="MeasurementControl">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="pMAdministrativeState" type="xn:pMAdministrativeStateType"/>

<element name="pMOperationalState" type="xn:pMOperationalStateType"/>

<element name="defaultFileBasedGP" type="integer"/>

<element name="defaultFileReportingPeriod" type="integer"/>

<element name="defaultFileLocation" type="string"/>

<element name="defaultStreamBasedGP" type="integer"/>

<element name="defaultStreamTarget" type="string"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:MeasurementReader"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="MeasurementReader">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="measurementTypes"/>

<element name="fileBasedGP" type="integer" minOccurs="0"/>

<element name="fileReportingPeriod" type="integer" minOccurs="0"/>

<element name="fileLocation" type="string" minOccurs="0"/>

<element name="streamBasedGP" type="integer" minOccurs="0"/>

<element name="streamTarget" type="string" minOccurs="0"/>

<element name="managedObjectDNsBasic" type="xn:dnList" minOccurs="0"/>

<element name="managedObjectDNs" type="xn:dnList" minOccurs="0"/>

</all>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="IRPAgent">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element ref="xn:systemDN" minOccurs="0"/>

</all>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="EP\_RP">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="farEndEntity" type="xn:dn" minOccurs="0"/>

<element name="userLabel" type="string" minOccurs="0"/>

<element name="measurementsList" type="xn:MeasurementsList" minOccurs="0"/>

</all>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="VsDataContainer">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="vsDataType"/>

<element name="vsDataFormatVersion"/>

<element ref="xn:vsData"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:VsDataContainer"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="ThresholdMonitoringCapability">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="supportedMonitoringGPs" type="xn:GPListType"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:ThresholdMonitoringCapabilityOptionallyContainedNrmClass"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="HeartbeatControl">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="heartbeatNtfPeriod" type="integer"/>

<element name="triggerHeartbeatNtf" type="boolean"/>

</all>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="ThresholdMonitor">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="thresholdInfoList" type="xn:thresholdInfoListType"/>

<element name="monitoringGP" type="integer"/>

<element name="monitoringNotifTarget" type="string"/>

<element name="monitoredIOCName" type="string"/>

<element name="monitoredObjectDNs" type="xn:dnList"/>

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="unbounded">

<element ref="xn:ThresholdMonitorOptionallyContainedNrmClass"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<element name="NtfSubscriptionControl">

<complexType>

<complexContent>

<extension base="xn:NrmClass">

<sequence>

<element name="attributes" minOccurs="0">

<complexType>

<all>

<element name="notificationRecipientAddress" type="string"/>

<element name="notificationTypes" type="string" minOccurs="0" />

<element name="scope" type="xn:Scope"/>

<element name="notificationFilter" type="string" minOccurs="0" />

</all>

</complexType>

</element>

<choice minOccurs="0" maxOccurs="1">

<element ref="xn:HeartbeatControl"/>

</choice>

</sequence>

</extension>

</complexContent>

</complexType>

</element>

<!--

IRPAgent IOC attributes

-->

<element name="systemDN" type="xn:dn"/>

<!--

VsDataContainer NRM class vsData attribute associated empty XML element

-->

<complexType name="vsData"/>

<element name="vsData" type="xn:vsData"/>

<!--

Abstract head XML element for all XML elements associated to further

NRM classes optionally contained under SubNetwork NRM class

-->

<element

name="SubNetworkOptionallyContainedNrmClass"

type="xn:NrmClass"

abstract="true"

/>

<!--

Abstract head XML element for all XML elements associated to further

NRM classes optionally contained under ManagedElement NRM class

-->

<element

name="ManagedElementOptionallyContainedNrmClass"

type="xn:NrmClass"

abstract="true"

/>

<!--

Abstract head XML element for all XML elements associated to further

NRM classes optionally contained under ThresholdMonitoringCapability NRM class

-->

<element

name="ThresholdMonitoringCapabilityOptionallyContainedNrmClass"

type="xn:NrmClass"

abstract="true"

/>

<!--

Abstract head XML element for all XML elements associated to further

NRM classes optionally contained under ThresholdMonitor NRM class

-->

<element

name="ThresholdMonitorOptionallyContainedNrmClass"

type="xn:NrmClass"

abstract="true"

/>

</schema>

Annex C (normative):  
OpenAPI definitions

# C.1 General

This annex contains the OpenAPI definition of the Generic NRM in YAML format.

The Information Service (IS) of the Generic NRM is defined in 3GPP TS 28.622 [4].

Mapping rules to produce the OpenAPI definition based on the IS are defined in 3GPP TS 32.160 [14].

# C.2 Void

# C.3 Void

# C.4 Solution Set (SS) definitions

## C.4.1 Void

## C.4.2 Void

## C.4.2a OpenAPI document "TS28623\_ComDefs.yaml"

openapi: 3.0.1

info:

title: Common Type Definitions

version: 16.9.0

description: >-

OAS 3.0.1 specification of common type definitions in the Generic NRM

© 2021, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).

All rights reserved.

externalDocs:

description: 3GPP TS 28.623; Generic NRM; Common type definitions

url: http://www.3gpp.org/ftp/Specs/archive/28\_series/28.623/

paths: {}

components:

schemas:

Float:

type: number

format: float

DateTime:

type: string

format: date-time

Latitude:

type: number

format: float

minimum: -90

maximum: 90

Longitude:

type: number

format: float

minimum: -180

maximum: 180

Dn:

type: string

DnList:

type: array

items:

$ref: '#/components/schemas/Dn'

Mcc:

type: string

pattern: '^[0-9]{3}$'

Mnc:

type: string

pattern: '^[0-9]{2,3}$'

Nid:

type: string

pattern: '^[A-Fa-f0-9]{11}$'

PlmnId:

type: object

properties:

mcc:

$ref: '#/components/schemas/Mcc'

mnc:

$ref: '#/components/schemas/Mnc'

Tac:

type: string

pattern: '(^[A-Fa-f0-9]{4}$)|(^[A-Fa-f0-9]{6}$)'

EutraCellId:

type: string

pattern: '^[A-Fa-f0-9]{7}$'

NrCellId:

type: string

pattern: '^[A-Fa-f0-9]{9}$'

Fqdn:

type: string

Ipv4Addr:

type: string

pattern: '^(([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-5])\.){3}([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-5])$'

example: '198.51.100.1'

Ipv6Addr:

type: string

allOf:

- pattern: '^((:|(0?|([1-9a-f][0-9a-f]{0,3}))):)((0?|([1-9a-f][0-9a-f]{0,3})):){0,6}(:|(0?|([1-9a-f][0-9a-f]{0,3})))$'

- pattern: '^((([^:]+:){7}([^:]+))|((([^:]+:)\*[^:]+)?::(([^:]+:)\*[^:]+)?))$'

example: '2001:db8:85a3::8a2e:370:7334'

Ipv6Prefix:

type: string

allOf:

- pattern: '^((:|(0?|([1-9a-f][0-9a-f]{0,3}))):)((0?|([1-9a-f][0-9a-f]{0,3})):){0,6}(:|(0?|([1-9a-f][0-9a-f]{0,3})))(\/(([0-9])|([0-9]{2})|(1[0-1][0-9])|(12[0-8])))$'

- pattern: '^((([^:]+:){7}([^:]+))|((([^:]+:)\*[^:]+)?::(([^:]+:)\*[^:]+)?))(\/.+)$'

example: '2001:db8:abcd:12::0/64'

IpAddr:

oneOf:

- $ref: '#/components/schemas/Ipv4Addr'

- $ref: '#/components/schemas/Ipv6Addr'

- $ref: '#/components/schemas/Ipv6Prefix'

HostAddr:

# This definition will be deprecated, when all occurances of HostAddr

# are replaced by Host.

oneOf:

- $ref: '#/components/schemas/Ipv4Addr'

- $ref: '#/components/schemas/Ipv6Addr'

- $ref: '#/components/schemas/Fqdn'

Host:

oneOf:

- $ref: '#/components/schemas/IpAddr'

- $ref: '#/components/schemas/Fqdn'

Uri:

type: string

AdministrativeState:

type: string

enum:

- LOCKED

- UNLOCKED

OperationalState:

type: string

enum:

- ENABLED

- DISABLED

UsageState:

type: string

enum:

- IDEL

- ACTIVE

- BUSY

AttributeNameValuePairSet:

description: >-

The key of this map is the attribute name, and the value the attribute value.

type: object

minProperties: 1

additionalProperties:

nullable: true

AttributeValueChangeSet:

description: >-

The first array item contains the attribute name value pairs with the new values,

and the second array item the attribute name value pairs with the optional old values.

type: array

items:

$ref: '#/components/schemas/AttributeNameValuePairSet'

minItems: 1

maxItems: 2

Filter:

description: >-

The filter format shall be compliant to XPath 1.0.

type: string

SystemDN:

type: string

NotificationId:

type: integer

NotificationType:

oneOf:

- $ref: 'TS28532\_FaultMnS.yaml#/components/schemas/AlarmNotificationTypes'

- $ref: 'TS28532\_ProvMnS.yaml#/components/schemas/CmNotificationTypes'

- $ref: 'TS28532\_PerfMnS.yaml#/components/schemas/PerfNotificationTypes'

- $ref: 'TS28532\_HeartbeatNtf.yaml#/components/schemas/HeartbeatNotificationTypes'

- $ref: 'TS28532\_FileDataReportingMnS.yaml#/components/schemas/FileNotificationTypes'

NotificationHeader:

type: object

properties:

href:

$ref: '#/components/schemas/Uri'

notificationId:

$ref: '#/components/schemas/NotificationId'

notificationType:

$ref: '#/components/schemas/NotificationType'

eventTime:

$ref: '#/components/schemas/DateTime'

systemDN:

$ref: '#/components/schemas/SystemDN'

required:

- href

- notificationId

- notificationType

- eventTime

- systemDN

ErrorResponse:

description: >-

Default schema for the response message body in case the request

is not successful.

type: object

properties:

error:

type: object

properties:

errorInfo:

type: string

## C.4.3 OpenAPI document "TS28623\_GenericNrm.yaml"

<CODE BEGINS>

openapi: 3.0.1

info:

title: Generic NRM

version: 16.17.0

description: >-

OAS 3.0.1 definition of the Generic NRM

© 2023, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).

All rights reserved.

externalDocs:

description: 3GPP TS 28.623; Generic NRM

url: http://www.3gpp.org/ftp/Specs/archive/28\_series/28.623/

paths: {}

components:

schemas:

#-------- Definition of types-----------------------------------------------------

RegistrationState:

type: string

enum:

- REGISTERED

- DEREGISTERED

VnfParameter:

type: object

properties:

vnfInstanceId:

type: string

vnfdId:

type: string

flavourId:

type: string

autoScalable:

type: boolean

PeeParameter:

type: object

properties:

siteIdentification:

type: string

siteDescription:

type: string

siteLatitude:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Latitude'

siteLongitude:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Longitude'

equipmentType:

type: string

environmentType:

type: string

powerInterface:

type: string

ThresholdInfo:

type: object

properties:

thresholdDirection:

type: string

enum:

- UP

- DOWN

- UP\_AND\_DOWN

thresholdValue:

oneOf:

- type: integer

- $ref: 'TS28623\_ComDefs.yaml#/components/schemas/Float'

hysteresis:

oneOf:

- type: integer

minimum: 0

- type: number

format: float

minimum: 0

Operation:

type: object

properties:

name:

type: string

allowedNFTypes:

$ref: '#/components/schemas/NFType'

operationSemantics:

$ref: '#/components/schemas/OperationSemantics'

NFType:

type: string

description: ' NF name defined in TS 23.501'

enum:

- NRF

- UDM

- AMF

- SMF

- AUSF

- NEF

- PCF

- SMSF

- NSSF

- UDR

- LMF

- GMLC

- 5G\_EIR

- SEPP

- UPF

- N3IWF

- AF

- UDSF

- DN

OperationSemantics:

type: string

enum:

- REQUEST\_RESPONSE

- SUBSCRIBE\_NOTIFY

SAP:

type: object

properties:

host:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/HostAddr'

port:

type: integer

NFServiceType:

type: string

enum:

- Namf\_Communication

- Namf\_EventExposure

- Namf\_MT

- Namf\_Location

- Nsmf\_PDUSession

- Nsmf\_EventExposure

- Others

TransportProtocol:

anyOf:

- type: string

enum:

- TCP

- type: string

SupportedPerfMetricGroup:

type: object

properties:

performanceMetrics:

type: array

items:

type: string

granularityPeriods:

type: array

items:

type: integer

minimum: 1

reportingMethods:

type: array

items:

type: string

enum:

- FILE\_BASED\_LOC\_SET\_BY\_PRODUCER

- FILE\_BASED\_LOC\_SET\_BY\_CONSUMER

- STREAM\_BASED

reportingPeriods:

type: array

items:

type: integer

minimum: 1

ReportingCtrl:

oneOf:

- type: object

properties:

fileReportingPeriod:

type: integer

- type: object

properties:

fileReportingPeriod:

type: integer

fileLocation:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Uri'

- type: object

properties:

streamTarget:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Uri'

Scope:

type: object

properties:

scopeType:

type: string

enum:

- BASE\_ONLY

- BASE\_ALL

- BASE\_NTH\_LEVEL

- BASE\_SUBTREE

scopeLevel:

type: integer

AreaScope:

oneOf:

- type: array

items:

$ref: '#/components/schemas/EutraCellId'

- type: array

items:

$ref: '#/components/schemas/NrCellId'

- type: array

items:

$ref: '#/components/schemas/Tac'

- type: array

items:

$ref: '#/components/schemas/Tai'

Tai:

type: object

properties:

mcc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mcc'

mnc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mnc'

tac:

$ref: '#/components/schemas/Tac'

AreaConfig:

type: object

properties:

freqInfo:

$ref: '#/components/schemas/FreqInfo'

pciList:

type: array

items:

type: integer

FreqInfo:

description: specifies the carrier frequency and bands used in a cell.

type: object

properties:

arfcn:

type: integer

freqBands:

type: array

items:

type: integer

MbsfnArea:

type: object

properties:

mbsfnAreaId:

type: integer

minimum: 1

earfcn:

type: integer

minimum: 1

Tac:

type: string

pattern: '(^[A-Fa-f0-9]{4}$)|(^[A-Fa-f0-9]{6}$)'

EutraCellId:

type: string

pattern: '^[A-Fa-f0-9]{7}$'

NrCellId:

type: string

pattern: '^[A-Fa-f0-9]{9}$'

IpAddr:

oneOf:

- $ref: 'TS28623\_ComDefs.yaml#/components/schemas/Ipv4Addr'

- $ref: 'TS28623\_ComDefs.yaml#/components/schemas/Ipv6Addr'

#-------- Definition of types used in Trace control NRM fragment------------------

jobType-Type:

type: string

description: Specifies whether the TraceJob represents only MDT, Logged MBSFN MDT, Trace or a combined Trace and MDT job. Applicable for Trace, MDT, RCEF and RLF reporting. See 3GPP TS 32.422 clause 5.9a for additional details.

enum:

- IMMEDIATE\_MDT\_ONLY

- LOGGED\_MDT\_ONLY

- TRACE\_ONLY

- IMMEDIATE\_MDT AND TRACE

- RLF\_REPORT\_ONLY

- RCEF\_REPORT\_ONLY

- LOGGED\_MBSFN\_MDT

listOfInterfaces-Type:

description: The interfaces to be recorded in the Network Element. See 3GPP TS 32.422 clause 5.5 for additional details.

type: object

properties:

MSCServerInterfaces:

type: array

items:

type: string

enum:

- A

- Iu-CS

- Mc

- MAP-G

- MAP-B

- MAP-E

- MAP-F

- MAP-D

- MAP-C

- CAP

MGWInterfaces:

type: array

items:

type: string

enum:

- Mc

- Nb-UP

- Iu-UP

RNCInterfaces:

type: array

items:

type: string

enum:

- Iu-CS

- Iu-PS

- Iur

- Iub

- Uu

SGSNInterfaces:

type: array

items:

type: string

enum:

- Gb

- Iu-PS

- Gn

- MAP-Gr

- MAP-Gd

- MAP-Gf

- Ge

- Gs

- S6d

- S4

- S3

- S13

GGSNInterfaces:

type: array

items:

type: string

enum:

- Gn

- Gi

- Gmb

S-CSCFInterfaces:

type: array

items:

type: string

enum:

- Mw

- Mg

- Mr

- Mi

P-CSCFInterfaces:

type: array

items:

type: string

enum:

- Gm

- Mw

I-CSCFInterfaces:

type: array

items:

type: string

enum:

- Cx

- Dx

- Mg

- Mw

MRFCInterfaces:

type: array

items:

type: string

enum:

- Mp

- Mr

MGCFInterfaces:

type: array

items:

type: string

enum:

- Mg

- Mj

- Mn

IBCFInterfaces:

type: array

items:

type: string

enum:

- Ix

- Mx

E-CSCFInterfaces:

type: array

items:

type: string

enum:

- Mw

- Ml

- Mm

- Mi/Mg

BGCFInterfaces:

type: array

items:

type: string

enum:

- Mi

- Mj

- Mk

ASInterfaces:

type: array

items:

type: string

enum:

- Dh

- Sh

- ISC

- Ut

HSSInterfaces:

type: array

items:

type: string

enum:

- MAP-C

- MAP-D

- Gc

- Gr

- Cx

- S6d

- S6a

- Sh

- N70

- N71

- NU1

EIRInterfaces:

type: array

items:

type: string

enum:

- MAP-F

- S13

- MAP-Gf

BM-SCInterfaces:

type: array

items:

type: string

enum:

- Gmb

MMEInterfaces:

type: array

items:

type: string

enum:

- S1-MME

- S3

- S6a

- S10

- S11

- S13

SGWInterfaces:

type: array

items:

type: string

enum:

- S4

- S5

- S8

- S11

- Gxc

PDN\_GWInterfaces:

type: array

items:

type: string

enum:

- S2a

- S2b

- S2c

- S5

- S6b

- Gx

- S8

- SGi

eNBInterfaces:

type: array

items:

type: string

enum:

- S1-MME

- X2

en-gNBInterfaces:

type: array

items:

type: string

enum:

- S1-MME

- X2

- Uu

- F1-C

- E1

AMFInterfaces:

type: array

items:

type: string

enum:

- N1

- N2

- N8

- N11

- N12

- N14

- N15

- N20

- N22

- N26

AUSFInterfaces:

type: array

items:

type: string

enum:

- N12

- N13

NEFInterfaces:

type: array

items:

type: string

enum:

- N29

- N30

- N33

NRFInterfaces:

type: array

items:

type: string

enum:

- N27

NSSFInterfaces:

type: array

items:

type: string

enum:

- N22

- N31

PCFInterfaces:

type: array

items:

type: string

enum:

- N5

- N7

- N15

SMFInterfaces:

type: array

items:

type: string

enum:

- N4

- N7

- N10

- N11

- S5-C

- N16

- N16a

- N38

SMSFInterfaces:

type: array

items:

type: string

enum:

- N20

- N21

UDMInterfaces:

type: array

items:

type: string

enum:

- N8

- N10

- N13

- N21

- NU1

UPFInterfaces:

type: array

items:

type: string

enum:

- N4

ng-eNBInterfaces:

type: array

items:

type: string

enum:

- NG-C

- Xn-C

- Uu

gNB-CU-CPInterfaces:

type: array

items:

type: string

enum:

- NG-C

- Xn-C

- Uu

- F1-C

- E1

- X2-C

gNB-CU-UPInterfaces:

type: array

items:

type: string

enum:

- E1

gNB-DUInterfaces:

type: array

items:

type: string

enum:

- F1-C

listOfNETypes-Type:

description: The Network Element types where Trace Session activation is needed. See 3GPP TS 32.422 clause 5.4 for additional details.

type: array

items:

type: string

enum:

- MSC\_SERVER

- SGSN

- MGW

- GGSN

- RNC

- BM\_SC

- MME

- SGW

- PGW

- ENB

- EN\_GNB

- GNB\_CU\_CP

- GNB\_CU\_UP

- GNB\_DU

- AMF

- PCF

- SMF

- UPF

- AUSF

- SMSF

- HSS

- UDM

pLMNTarget-Type:

type: object

description: The PLMN for which sessions shall be selected in the Trace Session in case of management based activation when several PLMNs are supported in the RAN (this means that shared cells and not shared cells are allowed for the specified PLMN. Note that the PLMN Target might differ from the PLMN specified in the Trace Reference, as that specifies the PLMN that is containing the management system requesting the Trace Session from the NE. See 3GPP TS 32.422 clause 5.9b for additional details.

properties:

mcc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mcc'

mnc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mnc'

traceDepth-Type:

description: Specifies how detailed information should be recorded in the Network Element. The Trace Depth is a paremeter for Trace Session level, i.e., the Trace Depth is the same for all of the NEs to be traced in the same Trace Session. See 3GPP TS 32.422 clause 5.3 for additional details.

type: string

enum:

- MINIMUM

- MEDIUM

- MAXIMUM

- VENDORMINIMUM

- VENDORMEDIUM

- VENDORMAXIMUM

traceReference-Type:

type: object

description: The Trace Reference parameter shall be globally unique, therefore the Trace Reference shall compose as follows - MCC+MNC+Trace ID, where the MCC and MNC are coming with the Trace activation request from the management system to identify one PLMN containing the management system, and Trace ID is a 3 byte Octet String. See 3GPP TS 32.422 clause 5.6 for additional details.

properties:

mcc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mcc'

mnc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mnc'

traceId:

type: string

traceReportingFormat-Type:

type: string

description: Specifies whether file-based or streaming reporting shall be used for this Trace Session. See 3GPP TS 32.422 clause 5.11 for additional details.

enum:

- FILE-BASED

- STREAMING

traceTarget-Type:

type: object

description: Trace target conveying both the type and value of the target ID. For additional details see 3GPP TS 32.422

properties:

TargetIdType:

type: string

enum:

- IMSI

- IMEI

- IMEISV

- PUBLIC\_ID

- UTRAN\_CELL

- E-UTRAN\_CELL

- NG-RAN\_CELL

- eNB

- RNC

- gNB

- SUPI

TargetIdValue:

type: string

triggeringEvents-Type:

type: object

description: Specifies when to start a Trace Recording Session and which message shall be recorded first, when to stop a Trace Recording Session and which message shall be recorded last respectively. See 3GPP TS 32.422 clause 5.1 for additional details.

properties:

MSC\_SERVER:

type: array

items:

type: string

enum:

- MO\_MT\_CALLS

- MO\_MT\_SMS

- LU\_IMSIattach\_IMSIdetach

- HANDOVER

- SS

SGSN:

type: array

items:

type: string

enum:

- PDPcontext

- MO\_MT\_SMS

- RAU\_GPRSattach\_GPRSdetach

- MBMScontext

MGW:

type: array

items:

type: string

enum:

- CONTEXT

GGSN:

type: array

items:

type: string

enum:

- PDPcontext

- MBMScontext

IMS:

type: array

items:

type: string

enum:

- SIPsession\_StandaloneTransaction

BM\_SC:

type: array

items:

type: string

enum:

- MBMSactivation

MME:

type: array

items:

type: string

enum:

- UEinitiatedPDNconnectivityRequest

- ServiceRequest

- InitialAttach\_TAU\_Detach

- UEinitiatedPDNdisconnection

- BearerActivationModificationDeletion

- Handover

SGW:

type: array

items:

type: string

enum:

- PDNconnectionCreation

- PDNconnectionTermination

- BearerActivationModificationDeletion

PGW:

type: array

items:

type: string

enum:

- PDNconnectionCreation

- PDNconnectionTermination

- BearerActivationModificationDeletion

AMF:

type: array

items:

type: string

enum:

- Registration

- ServiceRequest

- Handover

- UEderegistration

- NetworkDeregistration

- UEMobilityFromEPC

- UEMobilityToEPC

SMF:

type: array

items:

type: string

enum:

- PDUsessionEstablishment

- PDUsessionModification

- PDUsessionRelease

- PDUsessionUPactivationDeactivation

- MobilityBtw3gppAndN3gppTo5GC

- MobilityFromEpc

PCF:

type: array

items:

type: string

enum:

- AMpolicy

- SMpolicy

- Authorization

- BDTpolicy

UPF:

type: array

items:

type: string

enum:

- N4Session

AUSF:

type: array

items:

type: string

enum:

- UEauthentication

NEF:

type: array

items:

type: string

enum:

- EventExposure

- PFDmanagement

- ParameterProvision

- Trigger

NRF:

type: array

items:

type: string

enum:

- NFmanagement

- NFdiscovery

NSSF:

type: array

items:

type: string

enum:

- NSSelection

- NSSAI

SMSF:

type: array

items:

type: string

enum:

- SMservice

UDM:

type: array

items:

type: string

enum:

- UEcontext

- SubscriberData

- UEauthentication

- EventExposure

anonymizationOfMDTData-Type:

description: Specifies level of MDT anonymization. For additional details see 3GPP TS 32.422 clause 5.10.12.

type: string

enum:

- NO\_IDENTITY

- TAC\_OF\_IMEI

collectionPeriodRRMLTE-Type:

description: See details in 3GPP TS 32.422 clause 5.10.20.

type: string

enum:

- 100ms

- 1000ms

- 1024ms

- 1280ms

- 2048ms

- 2560ms

- 5120ms

- 10000ms

- 10240ms

- 60000ms

collectionPeriodM6LTE-Type:

description: See details in 3GPP TS 32.422 clause 5.10.32.

type: string

enum:

- 1024ms

- 2048ms

- 5120ms

- 10240ms

collectionPeriodM7LTE-Type:

description: See details in 3GPP TS 32.422 clause 5.10.33.

type: integer

minimum: 1

maximum: 60

collectionPeriodRRMUMTS-Type:

description: See details in 3GPP TS 32.422 clause 5.10.21.

type: string

enum:

- 100ms

- 250ms

- 500ms

- 1000ms

- 2000ms

- 3000ms

- 4000ms

- 6000ms

collectionPeriodRRMNR-Type:

description: See details in 3GPP TS 32.422 clause 5.10.30.

type: string

enum:

- 1024ms

- 2048ms

- 5120ms

- 10240ms

- 60000ms

collectionPeriodM6NR-Type:

description: See details in 3GPP TS 32.422 clause 5.10.34.

type: string

enum:

- 120ms

- 240ms

- 480ms

- 640ms

- 1024ms

- 2048ms

- 5120ms

- 10240ms

- 20480ms

- 40960ms

- 1min

- 6min

- 12min

- 30min

collectionPeriodM7NR-Type:

description: See details in 3GPP TS 32.422 clause 5.10.35.

type: integer

minimum: 1

maximum: 60

eventListForEventTriggeredMeasurement-Type:

description: See details in 3GPP TS 32.422 clause 5.10.28.

type: string

enum:

- OUT\_OF\_COVERAGE

- A2\_EVENT

eventThreshold-Type:

description: See details in 3GPP TS 32.422 clause 5.10.7 (LTE/NR Event A2 RSRP, RSRQ, SINR), 5.10.13 (UMTS Event 1f) and 5.10.14 (UMTS Event 1i).

type: object

properties:

EventThresholdRSRP:

oneOf:

- type: integer

minimum: 0

maximum: 97

- type: integer

minimum: 0

maximum: 127

EventThresholdRSRQ:

oneOf:

- type: integer

minimum: 0

maximum: 34

- type: integer

minimum: 0

maximum: 127

EventThresholdSINR:

type: integer

minimum: 0

maximum: 127

EventThreshold1F:

type: object

properties:

CPICH\_RSCP:

type: integer

minimum: -120

maximum: 25

CPICH\_EcNo:

type: integer

minimum: -24

maximum: 0

PathLoss:

type: integer

minimum: 30

maximum: 165

EventThreshold1I:

type: integer

minimum: -120

maximum: 25

listOfMeasurements-Type:

description: See details in 3GPP TS 32.422 clause 5.10.3 for details.

type: object

properties:

UMTS:

type: array

items:

type: string

enum:

- M1

- M2

- M3

- M4

- M5

- M6\_DL

- M6\_UL

- M7\_DL

- M7\_UL

LTE:

type: array

items:

type: string

enum:

- M1

- M2

- M3

- M4

- M5

- M1\_EVENT\_TRIGGERED

- M6

- M7

- M8

- M9

NR:

type: array

items:

type: string

enum:

- M1

- M2

- M3

- M4

- M5

- M6

- M7

- M1\_EVENT\_TRIGGERED

- M8

- M9

loggingDuration-Type:

description: See details in 3GPP TS 32.422 clause 5.10.9.

type: string

enum:

- 600s

- 1200s

- 2400s

- 3600s

- 5400s

- 7200s

loggingInterval-Type:

description: See details in 3GPP TS 32.422 clause 5.10.8.

type: object

properties:

UMTS:

type: array

items:

type: string

enum:

- 1.28s

- 2.56s

- 5.12s

- 10.24s

- 20.48s

- 30.72s

- 40.96s

- 61.44s

LTE:

type: array

items:

type: string

enum:

- 1.28s

- 2.56s

- 5.12s

- 10.24s

- 20.48s

- 30.72s

- 40.96s

- 61.44s

NR:

type: array

items:

type: string

enum:

- 0.32s

- 0.64s

- 1.28s

- 2.56s

- 5.12s

- 10.24s

- 20.48s

- 30.72s

- 40.96s

- 61.44s

- INFINITY

eventThresholdL1-Type:

description: See details in 3GPP TS 32.422 clause 5.10.36.

type: object

properties:

RSRP:

type: integer

minimum: 0

maximum: 127

RSRQ:

type: integer

minimum: 0

maximum: 127

hysteresisL1-Type:

description: See details in 3GPP TS 32.422 clause 5.10.37.

type: integer

minimum: 0

maximum: 30

timeToTriggerL1-Type:

description: See details in 3GPP TS 32.422 clause 5.10.38.

type: string

enum:

- 0ms

- 40ms

- 64ms

- 80ms

- 100ms

- 128ms

- 160ms

- 256ms

- 320ms

- 480ms

- 512ms

- 640ms

- 1024ms

- 1280ms

- 2560ms

- 5120ms

measurementPeriodLTE-Type:

description: See details in 3GPP TS 32.422 clause 5.10.23.

type: string

enum:

- 1024ms

- 2048ms

- 5120ms

- 10240ms

- 1min

measurementPeriodUMTS-Type:

description: See details in 3GPP TS 32.422 clause 5.10.22.

type: string

enum:

- 1000ms

- 2000ms

- 3000ms

- 4000ms

- 6000ms

- 8000ms

- 12000ms

- 16000ms

- 20000ms

- 24000ms

- 28000ms

- 32000ms

- 64000ms

measurementQuantity-Type:

description: See details in 3GPP TS 32.422 clause 5.10.15.

type: string

enum:

- CPICH\_EcNo

- CPICH\_RSCP

- PathLoss

eventThresholdUphUMTS-Type:

description: See details in 3GPP TS 32.422 clause 5.10.39.

type: integer

minimum: 0

maximum: 31

pLMNList-Type:

description: See details in 3GPP TS 32.422 clause 5.10.24.

type: array

items:

type: object

properties:

mcc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mcc'

mnc:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mnc'

maxItems: 16

positioningMethod-Type:

description: See details in 3GPP TS 32.422 clause 5.10.19.

type: string

enum:

- GNSS

- E-CELL\_ID

reportAmount-Type:

description: See details in 3GPP TS 32.422 clause 5.10.6.

type: string

enum:

- 1

- 2

- 4

- 8

- 16

- 32

- 64

- INFINITY

reportingTrigger-Type:

description: See details in 3GPP TS 32.422 clause 5.10.4.

type: array

items:

type: string

enum:

- PERIODICAL

- A2\_FOR\_LTE\_NR

- 1F\_FOR\_UMTS

- 1I\_FOR\_UMTS\_MCPS\_TDD

- A2\_TRIGGERED\_PERIODIC\_FOR\_LTE\_NR

- ALL\_CONFIGURED\_RRM\_FOR\_LTE\_NR

- ALL\_CONFIGURED\_RRM\_FOR\_UMTS

reportInterval-Type:

description: See details in 3GPP TS 32.422 clause 5.10.5.

type: object

properties:

UMTS:

type: array

items:

type: string

enum:

- 250ms

- 500ms

- 1000ms

- 2000ms

- 3000ms

- 4000ms

- 6000ms

- 8000ms

- 12000ms

- 16000ms

- 20000ms

- 24000ms

- 28000ms

- 32000ms

- 64000ms

LTE:

type: array

items:

type: string

enum:

- 120ms

- 240ms

- 480ms

- 640ms

- 1024ms

- 2048ms

- 5120ms

- 10240ms

- 60000ms

- 360000ms

- 720000ms

- 1800000ms

- 3600000ms

NR:

type: array

items:

type: string

enum:

- 120ms

- 240ms

- 480ms

- 640ms

- 1024ms

- 2048ms

- 5120ms

- 10240ms

- 20480ms

- 40960ms

- 60000ms

- 360000ms

- 720000ms

- 1800000ms

reportType-Type:

description: Report type for logged NR MDT. See details in 3GPP TS 32.422 clause 5.10.27.

type: string

enum:

- PERIODICAL

- EVENT\_TRIGGERED

sensorInformation-Type:

description: See details in 3GPP TS 32.422 clause 5.10.29.

type: array

items:

type: string

enum:

- BAROMETRIC\_PRESSURE

- UE\_SPEED

- UE\_ORIENTATION

traceCollectionEntityId-Type:

description: See details in 3GPP TS 32.422 clause 5.10.11. Only TCE Id value may be sent over the air to the UE being configured for Logged MDT.

type: integer

#-------- end of Definition of types used in Trace control NRM fragment ----------

#-------- Definition of abstract IOC Top -----------------------------------------

Top-Attr:

# This definition will be deprecated, when all occurances of Top-Attr

# are replaced by Top.

type: object

properties:

id:

type: string

nullable: true

objectClass:

type: string

objectInstance:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

VsDataContainer:

$ref: '#/components/schemas/VsDataContainer-Multiple'

required:

- id

Top:

type: object

properties:

id:

type: string

nullable: true

objectClass:

type: string

objectInstance:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

VsDataContainer:

$ref: '#/components/schemas/VsDataContainer-Multiple'

required:

- id

#-------- Definition of IOCs with new name-containments defined in other TS ------

SubNetwork-Attr:

type: object

properties:

dnPrefix:

type: string

userLabel:

type: string

userDefinedNetworkType:

type: string

setOfMcc:

type: array

items:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Mcc'

priorityLabel:

type: integer

supportedPerfMetricGroups:

type: array

items:

$ref: '#/components/schemas/SupportedPerfMetricGroup'

ManagedElement-Attr:

type: object

properties:

dnPrefix:

type: string

managedElementTypeList:

type: array

items:

type: string

userLabel:

type: string

locationName:

type: string

managedBy:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

vendorName:

type: string

userDefinedState:

type: string

swVersion:

type: string

priorityLabel:

type: integer

supportedPerfMetricGroups:

type: array

items:

$ref: '#/components/schemas/SupportedPerfMetricGroup'

SubNetwork-ncO:

type: object

properties:

ManagementNode:

$ref: '#/components/schemas/ManagementNode-Multiple'

MnsAgent:

$ref: '#/components/schemas/MnsAgent-Multiple'

MeContext:

$ref: '#/components/schemas/MeContext-Multiple'

PerfMetricJob:

$ref: '#/components/schemas/PerfMetricJob-Multiple'

ThresholdMonitor:

$ref: '#/components/schemas/ThresholdMonitor-Multiple'

NtfSubscriptionControl:

$ref: '#/components/schemas/NtfSubscriptionControl-Multiple'

TraceJob:

$ref: '#/components/schemas/TraceJob-Multiple'

AlarmList:

$ref: '#/components/schemas/AlarmList-Single'

ManagedElement-ncO:

type: object

properties:

MnsAgent:

$ref: '#/components/schemas/MnsAgent-Multiple'

PerfMetricJob:

$ref: '#/components/schemas/PerfMetricJob-Multiple'

ThresholdMonitor:

$ref: '#/components/schemas/ThresholdMonitor-Multiple'

NtfSubscriptionControl:

$ref: '#/components/schemas/NtfSubscriptionControl-Multiple'

TraceJob:

$ref: '#/components/schemas/TraceJob-Multiple'

AlarmList:

$ref: '#/components/schemas/AlarmList-Single'

#-------- Definition of abstract IOCs --------------------------------------------

ManagedFunction-Attr:

type: object

properties:

userLabel:

type: string

vnfParametersList:

type: array

items:

$ref: '#/components/schemas/VnfParameter'

peeParametersList:

type: array

items:

$ref: '#/components/schemas/PeeParameter'

priorityLabel:

type: integer

supportedPerfMetricGroups:

type: array

items:

$ref: '#/components/schemas/SupportedPerfMetricGroup'

EP\_RP-Attr:

type: object

properties:

userLabel:

type: string

farEndEntity:

type: string

supportedPerfMetricGroups:

type: array

items:

$ref: '#/components/schemas/SupportedPerfMetricGroup'

TraceJob-Attr:

type: object

description: abstract class used as a container of all TraceJob attributes

properties:

jobType:

$ref: '#/components/schemas/jobType-Type'

listOfInterfaces:

$ref: '#/components/schemas/listOfInterfaces-Type'

listOfNeTypes:

$ref: '#/components/schemas/listOfNETypes-Type'

plmnTarget:

$ref: '#/components/schemas/pLMNTarget-Type'

traceReportingConsumerUri:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Uri'

traceCollectionEntityIPAddress:

$ref: '#/components/schemas/IpAddr'

traceDepth:

$ref: '#/components/schemas/traceDepth-Type'

traceReference:

$ref: '#/components/schemas/traceReference-Type'

traceReportingFormat:

$ref: '#/components/schemas/traceReportingFormat-Type'

traceTarget:

$ref: '#/components/schemas/traceTarget-Type'

triggeringEvents:

$ref: '#/components/schemas/triggeringEvents-Type'

anonymizationOfMDTData:

$ref: '#/components/schemas/anonymizationOfMDTData-Type'

areaConfigurationForNeighCell:

type: array

items:

$ref: '#/components/schemas/AreaConfig'

maxItems: 32

areaScope:

type: array

items:

$ref: '#/components/schemas/AreaScope'

collectionPeriodRRMLTE:

$ref: '#/components/schemas/collectionPeriodRRMLTE-Type'

collectionPeriodM6LTE:

$ref: '#/components/schemas/collectionPeriodM6LTE-Type'

collectionPeriodM7LTE:

$ref: '#/components/schemas/collectionPeriodM7LTE-Type'

collectionPeriodRRMUMTS:

$ref: '#/components/schemas/collectionPeriodRRMUMTS-Type'

collectionPeriodRRMNR:

$ref: '#/components/schemas/collectionPeriodRRMNR-Type'

collectionPeriodM6NR:

$ref: '#/components/schemas/collectionPeriodM6NR-Type'

collectionPeriodM7NR:

$ref: '#/components/schemas/collectionPeriodM7NR-Type'

eventListForEventTriggeredMeasurement:

$ref: '#/components/schemas/eventListForEventTriggeredMeasurement-Type'

eventThreshold:

$ref: '#/components/schemas/eventThreshold-Type'

listOfMeasurements:

$ref: '#/components/schemas/listOfMeasurements-Type'

loggingDuration:

$ref: '#/components/schemas/loggingDuration-Type'

loggingInterval:

$ref: '#/components/schemas/loggingInterval-Type'

eventThresholdL1:

$ref: '#/components/schemas/eventThresholdL1-Type'

hysteresisL1:

$ref: '#/components/schemas/hysteresisL1-Type'

timeToTriggerL1:

$ref: '#/components/schemas/timeToTriggerL1-Type'

mbsfnAreaList:

type: array

items:

$ref: '#/components/schemas/MbsfnArea'

maxItems: 8

measurementPeriodLTE:

$ref: '#/components/schemas/measurementPeriodLTE-Type'

measurementPeriodUMTS:

$ref: '#/components/schemas/measurementPeriodUMTS-Type'

measurementQuantity:

$ref: '#/components/schemas/measurementQuantity-Type'

eventThresholdUphUMTS:

$ref: '#/components/schemas/eventThresholdUphUMTS-Type'

pLMNList:

$ref: '#/components/schemas/pLMNList-Type'

positioningMethod:

$ref: '#/components/schemas/positioningMethod-Type'

reportAmount:

$ref: '#/components/schemas/reportAmount-Type'

reportingTrigger:

$ref: '#/components/schemas/reportingTrigger-Type'

reportInterval:

$ref: '#/components/schemas/reportInterval-Type'

reportType:

$ref: '#/components/schemas/reportType-Type'

sensorInformation:

$ref: '#/components/schemas/sensorInformation-Type'

traceCollectionEntityId:

$ref: '#/components/schemas/traceCollectionEntityId-Type'

ManagedFunction-ncO:

type: object

properties:

PerfMetricJob:

$ref: '#/components/schemas/PerfMetricJob-Multiple'

ThresholdMonitor:

$ref: '#/components/schemas/ThresholdMonitor-Multiple'

ManagedNFService:

$ref: '#/components/schemas/ManagedNFService-Multiple'

TraceJob:

$ref: '#/components/schemas/TraceJob-Multiple'

#-------- Definition of concrete IOCs --------------------------------------------

VsDataContainer-Single:

type: object

properties:

id:

type: string

attributes:

type: object

properties:

vsDataType:

type: string

vsDataFormatVersion:

type: string

vsData:

nullable: true

VsDataContainer:

$ref: '#/components/schemas/VsDataContainer-Multiple'

ManagedNFService-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

userLabel:

type: string

nFServiceType:

$ref: '#/components/schemas/NFServiceType'

sAP:

$ref: '#/components/schemas/SAP'

operations:

type: array

items:

$ref: '#/components/schemas/Operation'

administrativeState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/AdministrativeState'

operationalState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/OperationalState'

usageState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/UsageState'

registrationState:

$ref: '#/components/schemas/RegistrationState'

ManagementNode-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

userLabel:

type: string

managedElements:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

vendorName:

type: string

userDefinedState:

type: string

locationName:

type: string

swVersion:

type: string

MnsAgent:

$ref: '#/components/schemas/MnsAgent-Multiple'

MnsAgent-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

systemDN:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Dn'

MeContext-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

dnPrefix:

type: string

PerfMetricJob-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

administrativeState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/AdministrativeState'

operationalState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/OperationalState'

jobId:

type: string

performanceMetrics:

type: array

items:

type: string

granularityPeriod:

type: integer

minimum: 1

objectInstances:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

rootObjectInstances:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

reportingCtrl:

$ref: '#/components/schemas/ReportingCtrl'

ThresholdMonitor-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

administrativeState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/AdministrativeState'

operationalState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/OperationalState'

performanceMetrics:

type: array

items:

type: string

thresholdInfoList:

type: array

items:

$ref: '#/components/schemas/ThresholdInfo'

monitorGranularityPeriod:

type: integer

minimum: 1

objectInstances:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

rootObjectInstances:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DnList'

NtfSubscriptionControl-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

notificationRecipientAddress:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Uri'

notificationTypes:

type: array

items:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/NotificationType'

scope:

$ref: '#/components/schemas/Scope'

notificationFilter:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/Filter'

HeartbeatControl:

$ref: '#/components/schemas/HeartbeatControl-Single'

HeartbeatControl-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

heartbeatNtfPeriod:

type: integer

minimum: 0

triggerHeartbeatNtf:

type: boolean

TraceJob-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

$ref: '#/components/schemas/TraceJob-Attr'

AlarmList-Single:

allOf:

- $ref: '#/components/schemas/Top'

- type: object

properties:

attributes:

type: object

properties:

administrativeState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/AdministrativeState'

operationalState:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/OperationalState'

numOfAlarmRecords:

type: integer

lastModification:

$ref: 'TS28623\_ComDefs.yaml#/components/schemas/DateTime'

alarmRecords:

description: >-

This resource represents a map of alarm records.

The alarmIds are used as keys in the map.

type: object

additionalProperties:

$ref: 'TS28532\_FaultMnS.yaml#/components/schemas/AlarmRecord'

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

VsDataContainer-Multiple:

type: array

items:

$ref: '#/components/schemas/VsDataContainer-Single'

ManagedNFService-Multiple:

type: array

items:

$ref: '#/components/schemas/ManagedNFService-Single'

ManagementNode-Multiple:

type: array

items:

$ref: '#/components/schemas/ManagementNode-Single'

MnsAgent-Multiple:

type: array

items:

$ref: '#/components/schemas/MnsAgent-Single'

MeContext-Multiple:

type: array

items:

$ref: '#/components/schemas/MeContext-Single'

PerfMetricJob-Multiple:

type: array

items:

$ref: '#/components/schemas/PerfMetricJob-Single'

ThresholdMonitor-Multiple:

type: array

items:

$ref: '#/components/schemas/ThresholdMonitor-Single'

TraceJob-Multiple:

type: array

items:

$ref: '#/components/schemas/TraceJob-Single'

NtfSubscriptionControl-Multiple:

type: array

items:

$ref: '#/components/schemas/NtfSubscriptionControl-Single'

#-------- Definitions in TS 28.623 for TS 28.532 ---------------------------------

resources-genericNrm:

oneOf:

- $ref: '#/components/schemas/VsDataContainer-Single'

- $ref: '#/components/schemas/ManagementNode-Single'

- $ref: '#/components/schemas/MnsAgent-Single'

- $ref: '#/components/schemas/MeContext-Single'

- $ref: '#/components/schemas/ManagedNFService-Single'

- $ref: '#/components/schemas/PerfMetricJob-Single'

- $ref: '#/components/schemas/ThresholdMonitor-Single'

- $ref: '#/components/schemas/TraceJob-Single'

- $ref: '#/components/schemas/NtfSubscriptionControl-Single'

- $ref: '#/components/schemas/HeartbeatControl-Single'

- $ref: '#/components/schemas/AlarmList-Single'

<CODE ENDS>

Annex D (normative):  
YANG definitions

# D.1 General

This annex contains the YANG definitions for the Generic NRM.

# D.2 Modules

## D.2.1 module \_3gpp-common-ep-rp.yang

<CODE BEGINS>

module \_3gpp-common-ep-rp {

yang-version 1.1;

namespace "urn:3gpp:sa5:\_3gpp-common-ep-rp";

prefix "eprp3gpp";

import \_3gpp-common-yang-types { prefix types3gpp ; }

import ietf-inet-types { prefix inet; }

import \_3gpp-common-measurements { prefix meas3gpp; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "Common/basic class/grouping to be inherited/reused.

This IOC represents an end point of a link used across a reference

point between two network entities.";

reference

"3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)

3GPP TS 28.620

Umbrella Information Model (UIM)";

revision 2020-06-08 { reference "CR-0092"; }

revision 2019-06-17 {

description "Initial revision";

}

grouping EP\_RPGrp {

description "Abstract class, represents an end point of a link used

across a reference point between two network entities.

For naming the subclasses of EP\_RP, the following rules shall apply:

- The name of the subclassed IOC shall have the form ’EP\_<rp>’,

where <rp> is a string that represents the name of the reference point.

Thus, two valid examples of EP\_RP subclassed IOC names would be:

EP\_S1 and EP\_X2.";

leaf userLabel {

type string;

description "A user-friendly (and user assignable) name of this object.";

}

leaf farEndEntity {

config false;

type types3gpp:DistinguishedName;

}

}

grouping EP\_Common {

uses EP\_RPGrp;

uses meas3gpp:SupportedPerfMetricGroupGrp;

list localAddress {

description "Local IP address and VLAN ID.";

key "ipAddress vlanId";

min-elements 1;

max-elements 1;

uses types3gpp:AddressWithVlan;

}

leaf remoteAddress {

description "Remote IP address.";

mandatory true;

type inet:ip-address;

}

}

}

<CODE ENDS>

## D.2.2 module \_3gpp-common-managed-element.yang

<CODE BEGINS>

module \_3gpp-common-managed-element {

yang-version 1.1;

namespace urn:3gpp:sa5:\_3gpp-common-managed-element;

prefix "me3gpp";

import \_3gpp-common-yang-types { prefix types3gpp ; }

import \_3gpp-common-top { prefix top3gpp; }

import \_3gpp-common-measurements { prefix meas3gpp; }

import \_3gpp-common-subscription-control { prefix subscr3gpp; }

import \_3gpp-common-fm { prefix fm3gpp; }

import \_3gpp-common-trace { prefix trace3gpp; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "Defines ManagedElement which will be augmented

by other IOCs";

reference "3GPP TS 28.623

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Solution Set (SS) definitions

3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)

3GPP TS 28.620

Umbrella Information Model (UIM)";

revision 2021-01-16 { reference "CR-0120"; }

revision 2020-08-06 { reference "CR-0102"; }

revision 2020-08-03 { reference "CR-0095"; }

revision 2020-06-08 { reference "CR-0092"; }

revision 2020-05-12 { reference "CR0084"; }

revision 2020-02-24 { reference "S5-201365"; }

revision 2019-06-17 { reference " S5-203316"; }

revision 2019-05-08 { reference "Initial revision"; }

feature MeasurementsUnderManagedElement {

description "The MeasurementSubtree shall be contained under

ManagedElement";

}

feature SubscriptionControlUnderManagedElement {

description "The SubscriptionControlSubtree shall be contained under

ManagedElement";

}

feature FmUnderManagedElement {

description "The FmSubtree shall be contained under ManagedElement";

}

feature TraceUnderManagedElement {

description "The TraceSubtree shall be contained under ManageElement";

}

feature DESManagementFunction {

description "Class representing Distributed SON or Domain-Centralized SON

Energy Saving feature. The DESManagementFunction shall be contained under

ManagedElement.";

}

feature DMROFunction {

description "Class representing D-SON function of MRO feature. The

DMROFunction shall be contained under ManagedElement.";

}

feature DRACHOptimizationFunction {

description "Class representing D-SON function of RACH optimization

feature. The DRACHOptimizationFunction shall be contained under

ManagedElement.";

}

feature DPCIConfigurationFunction {

description "Class representing Distributed SON or Domain-Centralized SON

function of PCI configuration feature. The DPCIConfigurationFunction shall

be contained under ManagedElement.";

}

feature CPCIConfigurationFunction {

description "Class representing Cross Domain-Centralized SON function of PCI

configuration feature. The CPCIConfigurationFunction shall be contained

under ManagedElement.";

}

feature CESManagementFunction {

description "Class representing Cross Domain-Centralized SON Energy Saving

feature. The CESManagementFunction shall be contained under

ManagedElement.";

}

grouping ManagedElement\_Grp {

description "Abstract class representing telecommunications resources.

An ME communicates with a manager (directly or indirectly) for the

purpose of being monitored and/or controlled. MEs may perform element

management functionality.

An ME (and its contained Function\_(s)) may or may not be geographically

distributed. An ME (and its contained Function\_(s)) is often referred

to as a Network Element";

leaf dnPrefix {

description "Provides naming context that allows the Managed

Elements to be partitioned into logical domains.

A Distingushed Name(DN) is defined by 3GPP TS 32.300,

which splits the DN into a DN Prefix and Local DN";

type types3gpp:DistinguishedName;

}

leaf userLabel {

description "A user-friendly (and user assignable) name of this object.";

type string;

}

leaf locationName {

description "The physical location (e.g. an address) of an entity

represented by a (derivative of) ManagedElement\_. It may contain no

information to support the case where the derivative of

ManagedElement\_ needs to represent a distributed multi-location NE.";

config false;

type string;

}

leaf-list managedBy {

description "Relates to the role played by ManagementSystem\_ in the

between ManagedSystem\_ and ManagedElement\_. This attribute contains

a list of the DN(s) of the related subclasses of

ManagementSystem\_ instance(s).";

config false;

type types3gpp:DistinguishedName;

}

leaf-list managedElementTypeList {

description "The type of functionality provided by the ManagedElement.

It may represent one ME functionality or a combination of

more than one functionality.

1) The allowed values of this attribute are the names of the IOC(s)

that are (a) derived/subclassed from ManagedFunction and (b) directly

name-contained by ManagedElement IOC (on the first level below

ManagedElement), but with the string ’Function’ excluded.

2) If a ManagedElement contains multiple instances of a ManagedFunction

this attribute will not contain repeated values.

3) The capitalisation (usage of upper/lower case) of characters in this

attribute is insignificant. Thus, the NodeB should be case insensitive

when reading these values.

4) Two examples of allowed values are:

- NodeB;

- HLR, VLR.";

config false;

min-elements 1;

type string;

}

}

grouping ManagedElementGrp {

description "Represents telecommunications equipment or

TMN entities within the telecommunications network providing support

and/or service to the subscriber.";

uses ManagedElement\_Grp;

uses meas3gpp:SupportedPerfMetricGroupGrp;

leaf vendorName {

config false;

type string;

}

leaf userDefinedState {

type string;

description "An operator defined state for operator specific usage";

}

leaf swVersion {

config false;

type string;

}

leaf priorityLabel {

type uint32;

mandatory true;

}

}

list ManagedElement {

description "Represents telecommunications equipment or TMN entities within

the telecommunications network providing support and/or service to the

subscriber. An ManagedElement IOC is used to represent a Network Element

defined in TS 32.101 including virtualizeation or non-virtualization

scenario. An ManagedElement instance is used for communicating with a

manager (directly or indirectly) over one or more management interfaces

for the purpose of being monitored and/or controlled. ManagedElement may

or may not additionally perform element management functionality.

An ManagedElement contains equipment that may or may not be geographically

distributed.

A telecommunication equipment has software and hardware components. The

ManagedElement IOC described above represents following two case:

- In the case when the software component is designed to run on dedicated

hardware component, the ManagedElement IOC description includes both

software and hardware components.

- In the case when the software is designed to run on ETSI NFV defined

NFVI [15], the ManagedElement IOC description would exclude the NFVI

component supporting the above mentioned subject software.

A ManagedElement may be contained in either a SubNetwork or in a MeContext

instance. A single ManagedElement may also exist stand-alone with no

parent at all.

The relation of ManagedElement IOC and ManagedFunction IOC can be

described as following：

- A ManaagedElement instance may have 1..1 containment relationship to a

ManagedFunction instance. In this case, the ManagedElement IOC may be

used to represent a NE with single functionality. For example, a

ManagedElement is used to represent the 3GPP defined RNC node;

- A ManagedElement instance may have 1..N containment relationship to

multiple ManagedFunction IOC instances. In this case, the ManagedElement

IOC may be used to represent a NE with combined ManagedFunction

funcationality (as indicated by the managedElementType attribute and the

contained instances of different ManagedFunction IOCs).For example, a

ManagedElement is used to represent the combined functionality of 3GPP

defined gNBCUCPFuntion, gNBCUUPFunction and gNBDUFunction";

key id;

uses top3gpp:Top\_Grp;

container attributes {

uses ManagedElementGrp;

}

uses meas3gpp:MeasurementSubtree {

if-feature MeasurementsUnderManagedElement;

}

uses subscr3gpp:SubscriptionControlSubtree {

if-feature SubscriptionControlUnderManagedElement;

}

uses fm3gpp:FmSubtree {

if-feature FmUnderManagedElement;

}

uses trace3gpp:TraceSubtree {

if-feature TraceUnderManagedElement;

}

}

}

<CODE ENDS>

## D.2.3 module \_3gpp-common-managed-function.yang

<CODE BEGINS>

module \_3gpp-common-managed-function {

yang-version 1.1;

namespace urn:3gpp:sa5:\_3gpp-common-managed-function;

prefix mf3gpp;

import \_3gpp-common-yang-types { prefix types3gpp; }

import \_3gpp-common-yang-extensions { prefix yext3gpp; }

import \_3gpp-common-top { prefix top3gpp; }

import \_3gpp-common-measurements { prefix meas3gpp; }

import \_3gpp-common-trace { prefix trace3gpp; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "The module defines a base class/grouping for major 3GPP

functions.";

reference

"3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)

3GPP TS 28.620

Umbrella Information Model (UIM)";

revision 2022-11-03 { reference CR-0192; }

revision 2021-01-25 { reference "CR-0122"; }

revision 2020-09-30 { reference "CR-bbbb"; }

revision 2020-08-06 { reference "CR-0102"; }

revision 2020-08-03 { reference "CR-0095"; }

revision 2020-06-23 { reference "CR-085"; }

revision 2020-06-08 { reference "CR-0092"; }

revision 2019-11-21 { reference "S5-197275, S5-197735"; }

revision 2019-10-28 { reference S5-193518 ; }

revision 2019-06-18 { reference "Initial revision"; }

feature MeasurementsUnderManagedFunction {

description "The MeasurementSubtree shall be contained under ManageElement";

}

feature TraceUnderManagedFunction {

description "The TraceSubtree shall be contained under ManagedFunction";

}

grouping Operation {

description "This data type represents an Operation.";

reference "3gpp TS 28.622";

leaf name {

type string;

mandatory true;

yext3gpp:notNotifyable;

}

leaf-list allowedNFTypes {

type string;

min-elements 1;

description "The type of the managed NF service instance

The specifc values allowed are described in TS 23.501";

}

leaf operationSemantics {

type enumeration {

enum REQUEST\_RESPONSE;

enum SUBSCRIBE\_NOTIFY;

}

config false;

mandatory true;

description "Semantics type of the operation.";

reference "3GPP TS 23.502";

}

}

grouping ManagedNFServiceGrp {

description "A ManagedNFService represents a Network Function (NF) service.";

reference "Clause 7 of 3GPP TS 23.501.";

leaf userLabel {

type string;

description "A user-friendly (and user assignable) name of this object.";

}

leaf nFServiceType {

config false;

mandatory true;

type string;

description "The type of the managed NF service instance

The specifc values allowed are described in clause 7.2 of TS 23.501";

yext3gpp:notNotifyable;

}

list sAP {

key "host port";

min-elements 1;

max-elements 1;

description "The service access point of the managed NF service instance";

uses types3gpp:SAP;

}

list operations {

key name;

min-elements 1;

uses Operation ;

description "Set of operations supported by the managed NF

service instance";

}

leaf administrativeState {

type types3gpp:AdministrativeState;

mandatory true;

description "Permission to use or prohibition against using the instance";

}

leaf operationalState {

type types3gpp:OperationalState;

config false;

mandatory true;

description "Describes whether the resource is installed and working";

}

leaf usageState {

type types3gpp:usageState ;

config false;

mandatory true;

description "Describes whether the resource is actively in use at a

specific instant, and if so, whether or not it has spare

capacity for additional users.";

}

leaf registrationState {

type enumeration {

enum REGISTERED;

enum DEREGISTERED;

}

config false;

}

}

grouping Function\_Grp {

description "A base grouping for 3GPP functions.";

leaf userLabel {

type string;

description "A user-friendly (and user assignable) name of this object.";

}

}

grouping ManagedFunctionGrp {

description "Abstract root class to be inherited/reused by classes

representing 3GPP functions.

Anywhere this grouping is used by classes inheriting from ManagedFunction

the list representing the inheriting class needs to include all

contained classes of ManagedFunction too. Contained classes are

either

- augmented into the Function class or

- shall be included in the list representing the inheriting class

using the grouping ManagedFunctionContainedClasses:

1) EP\_RP solved using augment

2) uses mf3gpp:ManagedFunctionContainedClasses;

";

uses Function\_Grp;

list vnfParametersList {

key vnfInstanceId;

description "Contains the parameter set of the VNF

instance(s) corresponding to an NE.

The presence of this list indicates that the ManagedFunction

represented is realized by one or more VNF instance(s). Otherwise it

shall be absent.

The presence of a vnfParametersList entry, whose vnfInstanceId with a

string length of zero, in createMO operation can trigger the

instantiation of the related VNF/VNFC instances.";

leaf vnfInstanceId {

type string ;

description "VNF instance identifier";

reference "ETSI GS NFV-IFA 008 v2.1.1:

Network Functions Virtualisation (NFV); Management and Orchestration;

Ve-Vnfm reference point - Interface and Information Model Specification

section 9.4.2

ETSI GS NFV-IFA 015 v2.1.2: Network Functions Virtualisation (NFV);

Management and Orchestration; Report on NFV Information Model

section B2.4.2.1.2.3";

}

leaf vnfdId {

type string ;

description "Identifier of the VNFD on which the VNF instance is based.

The absence of the leaf or a string length of zero for vnfInstanceId

means the VNF instance(s) does not exist (e.g. has not been

instantiated yet, has already been terminated).";

reference "ETSI GS NFV-IFA 008 v2.1.1:

Network Functions Virtualisation (NFV); Management and Orchestration;

Ve-Vnfm reference point - Interface and Information Model Specification

section 9.4.2";

}

leaf flavourId {

type string ;

description "Identifier of the VNF Deployment Flavour applied to this

VNF instance.";

reference "ETSI GS NFV-IFA 008 v2.1.1:

Network Functions Virtualisation (NFV) Management and Orchestration";

}

leaf autoScalable {

type boolean ;

mandatory true;

description "Indicator of whether the auto-scaling of this

VNF instance is enabled or disabled.";

}

}

list peeParametersList {

key idx;

description "Contains the parameter set for the control

and monitoring of power, energy and environmental parameters of

ManagedFunction instance(s).";

leaf idx { type uint32; }

leaf siteIdentification {

type string;

mandatory true;

description "The identification of the site where the

ManagedFunction resides.";

}

leaf siteLatitude {

type decimal64 {

fraction-digits 4;

range "-90.0000..+90.0000";

}

description "The latitude of the site where the ManagedFunction

instance resides, based on World Geodetic System (1984 version)

global reference frame (WGS 84). Positive values correspond to

the northern hemisphere. This attribute is optional in case of

BTSFunction and RNCFunction instance(s).";

}

leaf siteLongitude {

type decimal64 {

fraction-digits 4;

range "-180.0000..+180.0000";

}

description "The longitude of the site where the ManagedFunction

instance resides, based on World Geodetic System (1984 version)

global reference frame (WGS 84). Positive values correspond to

degrees east of 0 degrees longitude. This attribute is optional in

case of BTSFunction and RNCFunction instance(s).";

}

leaf siteDescription {

type string;

mandatory true;

description "An operator defined description of the site where

the ManagedFunction instance resides.";

}

leaf equipmentType {

type string;

mandatory true;

description "The type of equipment where the managedFunction

instance resides.";

reference "clause 4.4.1 of ETSI ES 202 336-12";

}

leaf environmentType {

type string;

mandatory true;

description "The type of environment where the managedFunction

instance resides.";

reference "clause 4.4.1 of ETSI ES 202 336-12";

}

leaf powerInterface {

type string;

mandatory true;

description "The type of power.";

reference "clause 4.4.1 of ETSI ES 202 336-12";

}

}

leaf priorityLabel {

mandatory true;

type uint32;

}

uses meas3gpp:SupportedPerfMetricGroupGrp;

}

grouping ManagedFunctionContainedClasses {

description "A grouping used to containe classes (lists) contained by

the abstract IOC ManagedFunction";

list ManagedNFService {

description "Represents a Network Function (NF)";

reference "3GPP TS 23.501";

key id;

uses top3gpp:Top\_Grp;

container attributes {

uses ManagedNFServiceGrp;

}

}

uses meas3gpp:MeasurementSubtree {

if-feature MeasurementsUnderManagedFunction ;

}

uses trace3gpp:TraceSubtree {

if-feature TraceUnderManagedFunction ;

}

}

}

<CODE ENDS>

## D.2.4 module \_3gpp-common-measurements.yang

<CODE BEGINS>

module \_3gpp-common-measurements {

yang-version 1.1;

namespace "urn:3gpp:sa5:\_3gpp-common-measurements";

prefix "meas3gpp";

import \_3gpp-common-top { prefix top3gpp; }

import \_3gpp-common-yang-types { prefix types3gpp; }

import \_3gpp-common-yang-extensions { prefix yext3gpp; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "Defines Measurement and KPI related groupings

Any list/class intending to use this should include 2 or 3 uses statements

controlled by a feature:

A)

+++ feature MeasurementsUnderMyClass {

+++ description 'Indicates whether measurements and/or KPIs are supported

+++ for this class.';

+++ }

B) include the attribute measurementsList and/or kPIsList indicating the

supported measurment and KPI types and GPs. Note that for classes

inheriting from ManagedFunction, EP\_RP or SubNetwork these attributes are

already inherited, so there is no need to include them once more. E.g.

+++ grouping MyClassGrp {

+++ uses meas3gpp:SupportedPerfMetricGroup;

+++ }

C) include the class PerfmetricJob to control the measurements/KPIs. E.g.

list MyClass {

container attributes {

uses MyClassGrp;

}

+++ uses meas3gpp:MeasurementSubtree {

+++ if-feature MeasurementsUnderMyClass ;

+++ }

}

Measurements can be contained under ManagedElement, SubNetwork, or

any list representing a class inheriting from Subnetwork or

ManagedFunction. Note: KPIs will only be supported under SubNetwork";

reference "3GPP TS 28.623

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Solution Set (SS) definitions

3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)";

revision 2023-02-16 { reference "CR-0238";}

revision 2022-11-03 { reference CR-0192; }

revision 2021-07-22 { reference "CR-0137"; }

revision 2020-11-06 { reference "CR-0118"; }

revision 2020-09-04 { reference "CR-000107"; }

revision 2020-06-08 { reference "CR-0092"; }

revision 2020-05-31 { reference "CR-0084"; }

revision 2020-03-11 { reference "S5-201581, SP-200229"; }

revision 2019-11-21 { reference "S5-197275, S5-197735"; }

revision 2019-10-28 { reference "S5-193516"; }

revision 2019-06-17 { reference " "; }

grouping ThresholdInfoGrp {

description "Defines a single threshold level.";

leaf-list measurementTypes {

type string;

description "The Measurement type can be those specified in TS 28.552,

TS 32.404 and can be those specified by other SDOs or can be

vendor-specific.";

}

leaf thresholdLevel {

type uint64;

mandatory true;

description "Number (key) for a single threshold in the threshold list

applicable to the monitored performance metric.";

}

leaf thresholdDirection {

type enumeration {

enum UP;

enum DOWN;

enum UP\_AND\_DOWN;

}

must '. = "UP\_AND\_DOWN" or not(../hysteresis)' {

error-message "In case a threshold with hysteresis is configured, the "

+"threshold direction attribute shall be set to 'UP\_AND\_DOWN'.";

}

mandatory true;

description "Direction of a threshold indicating the direction for which

a threshold crossing triggers a threshold.

When the threshold direction is configured to 'UP', the associated

treshold is triggered only when the performance metric value is going

up upon reaching or crossing the threshold value. The treshold is not

triggered, when the performance metric is going down upon reaching or

crossing the threshold value.

Vice versa, when the threshold direction is configured to 'DOWN', the

associated treshold is triggered only when the performance metric is

going down upon reaching or crossing the threshold value. The treshold

is not triggered, when the performance metric is going up upon reaching

or crossing the threshold value.

When the threshold direction is set to 'UP\_AND\_DOWN' the treshold is

active in both direcions.

In case a threshold with hysteresis is configured, the threshold

direction attribute shall be set to 'UP\_AND\_DOWN'.";

}

leaf thresholdValue {

type union {

type int64;

type decimal64 {

fraction-digits 2;

}

}

mandatory true;

description "Value against which the monitored performance metric is

compared at a threshold level in case the hysteresis is zero";

}

leaf hysteresis {

type union {

type uint64;

type decimal64 {

fraction-digits 2;

range "0..max";

}

}

description "Hysteresis of a threshold. If this attribute is present

the monitored performance metric is not compared against the

threshold value as specified by the thresholdValue attribute but

against a high and low threshold value given by

threshold-high = thresholdValue + hysteresis

threshold-low = thresholdValue - hysteresis

When going up, the threshold is triggered when the performance metric

reaches or crosses the high threshold value. When going down, the

hreshold is triggered when the performance metric reaches or crosses

the low threshold value.

A hysteresis may be present only when the monitored performance

metric is not of type counter that can go up only. If present

for a performance metric of type counter, it shall be ignored.";

}

}

grouping SupportedPerfMetricGroupGrp {

list SupportedPerfMetricGroup {

config false;

description "Captures a group of supported performance metrics and

associated parameters related to their production and reporting.

A SupportedPerfMetricGroup attribute which is part of an MOI may

define performanceMetrics for any MOI under the subtree contained

under that MOI, e.g. SupportedPerfMetricGroup on a ManagedElement

can specify supported metrics for contained ManagedFunctions

like a GNBDUFunction.";

leaf-list performanceMetrics {

type string;

min-elements 1;

description "Performance metrics include measurements defined in

TS 28.552 and KPIs defined in TS 28.554. Performance metrics can

also be specified by other SDOs or be vendor specific.

Performance metrics are identfied with their names.

For measurements defined in TS 28.552 the name is constructed as

follows:

- 'family.measurementName.subcounter' for measurement types with

subcounters

- 'family.measurementName' for measurement types without subcounters

- 'family' for measurement families

For KPIs defined in TS 28.554 the name is defined in the KPI

definitions template as the component designated with e).

A name can also identify a vendor specific performance metric or a

group of vendor specific performance metrics.";

}

leaf-list granularityPeriods {

type uint32 {

range 1..max ;

}

units seconds;

description "Granularity periods supported for the associated

measurement types. The period is defined in seconds.";

}

leaf-list reportingMethods {

type enumeration {

enum FILE\_BASED\_LOC\_SET\_BY\_PRODUCER;

enum FILE\_BASED\_LOC\_SET\_BY\_CONSUMER;

enum STREAM\_BASED;

}

min-elements 1;

}

leaf-list reportingPeriods {

type uint32 {

range 1..max ;

}

units seconds;

description "Reporting periods supported for the associated

measurement types. The period is defined in seconds.";

}

}

}

grouping PerfMetricJobGrp {

description "Represents the attributtes of the IOC PerfMetricJob";

leaf administrativeState {

default UNLOCKED;

type types3gpp:AdministrativeState ;

description "Enable or disables production of the metrics";

}

leaf operationalState {

config false;

mandatory true;

type types3gpp:OperationalState ;

description "Indicates whether the PerfMetricJob is working.";

}

leaf jobId {

type string;

description "Id for a PerfMetricJob job.";

}

leaf-list performanceMetrics {

type string;

min-elements 1;

description "Performance metrics include measurements defined in

TS 28.552 and KPIs defined in TS 28.554. Performance metrics can

also be those specified by other SDOs or vendor specific metrics.

Performance metrics are identfied with their names. A name can also

identify a vendor specific group of performance metrics.

For measurements defined in TS 28.552 the name is constructed as

follows:

- 'family.measurementName.subcounter' for measurement types with

subcounters

- 'family.measurementName' for measurement types without subcounters

- 'family' for measurement families

For KPIs defined in TS 28.554 the name is defined in the KPI

definitions template as the component designated with e).";

}

leaf granularityPeriod {

type uint32 {

range 1..max ;

}

units seconds;

mandatory true;

description "Granularity period used to produce measurements. The value

must be one of the supported granularity periods for the metric.";

}

leaf-list objectInstances {

type types3gpp:DistinguishedName;

}

leaf-list rootObjectInstances {

type types3gpp:DistinguishedName;

description "Each object instance designates the root of a subtree that

contains the root object and all descendant objects.";

}

choice reportingCtrl {

mandatory true;

description "This choice defines the method for reporting collected

performance metrics to MnS consumers as well as the parameters for

configuring the reporting function. It is a choice between the control

parameter required for the reporting methods, whose presence selects

the reporting method as follows:

- When only the fileReportingPeriod attribute is present, the MnS

producer shall store files on the MnS producer at a location selected

by the MnS producer and inform the MnS consumer about the availability

of new files and the file location using the notifyFileReady

notification.

- When only the fileReportingPeriod and fileLocation attributes are

present, the MnS producer shall store the files on the MnS consumer at

the location specified by fileLocation. No notification is emitted by

the MnS producer.

- When only the streamTarget attribute is present, the MnS producer

shall stream the data to the location specified by streamTarget.

For the file-based reporting methods the fileReportingPeriod attribute

specifies the time window during which collected measurements are stored

into the same file before the file is closed and a new file is opened.";

case file-based-reporting {

leaf fileReportingPeriod {

type uint32 {

range 1..max;

}

units minutes;

must '(number(.)\*"60") mod number(../granularityPeriod) = "0"' {

error-message

"The time-period must be a multiple of the granularityPeriod.";

}

mandatory true;

description "For the file-based reporting method this is the time

window during which collected measurements are stored into the same

file before the file is closed and a new file is opened.

The time-period must be a multiple of the granularityPeriod.

Applicable when the file-based reporting method is supported";

}

leaf fileLocation {

type string ;

description "Applicable and must be present when the file-based

reporting method is supported, and the files are stored on the MnS

consumer.";

}

}

case stream-based-reporting {

leaf streamTarget {

type string;

mandatory true;

description "Applicable when stream-based reporting method is

supported.";

}

}

}

}

grouping ThresholdMonitorGrp {

description "A threshold monitor that is created by the consumer for

the monitored entities whose measurements are required by consumer

to monitor.";

leaf administrativeState {

default UNLOCKED;

type types3gpp:AdministrativeState ;

description "Enables or disables the ThresholdMonitor.";

}

leaf operationalState {

config false;

mandatory true;

type types3gpp:OperationalState ;

description "Indicates whether the ThresholdMonitor is working.";

}

list thresholdInfoList {

key idx;

min-elements 1;

leaf idx { type uint32 ; }

uses ThresholdInfoGrp;

description "List of threshold infos.";

}

leaf monitorGranularityPeriod {

type uint32 {

range "1..max";

}

units second;

mandatory true;

description " Granularity period used to monitor measurements for

threshold crossings. ";

}

leaf-list objectInstances {

type types3gpp:DistinguishedName;

yext3gpp:notNotifyable;

}

leaf-list rootObjectInstances {

type types3gpp:DistinguishedName;

description "Each object instance designates the root of a subtree that

contains the root object and all descendant objects.";

yext3gpp:notNotifyable;

}

}

grouping MeasurementSubtree {

description "Contains classes that define measurements.

Should be used in all classes (or classes inheriting from)

- SubNnetwork

- ManagedElement

- ManagedFunction

If a YANG module wants to augment these classes/list/groupings they must

augment all user classes!

If a class uses this grouping in its list it shall also use the

grouping SupportedPerfMetricGroupGrp to add SupportedPerfMetricGroup as

an attribute to its grouping";

list PerfMetricJob {

description "This IOC represents a performance metric production job. It

can be name-contained by SubNetwork, ManagedElement, or ManagedFunction.

To activate the production of the specified performance metrics, a MnS

consumer needs to create a PerfMetricJob instance on the MnS producer.

For ultimate deactivation of metric production, the MnS consumer should

delete the job to free up resources on the MnS producer.

For temporary suspension of metric production, the MnS consumer can

manipulate the value of the administrative state attribute. The MnS

producer may disable metric production as well, for example in overload

situations. This situation is indicated by the MnS producer with setting

the operational state attribute to disabled. When production is resumed

the operational state is set back to enabled.

The jobId attribute can be used to associate metrics from multiple

PerfMetricJob instances. The jobId can be included when reporting

performance metrics to allow a MnS consumer to associate received

metrics for the same purpose. For example, it is possible to configure

the same jobId value for multiple PerfMetricJob instances required to

produce the measurements for a specific KPI.

The attribute performanceMetrics defines the performance metrics to be

produced and the attribute granularityPeriod defines the granularity

period to be applied.

All object instances below and including the instance name-containing

the PerfMetricJob (base object instance) are scoped for performance

metric production. Performance metrics are produced only on those object

instances whose object class matches the object class associated to the

performance metrics to be produced.

The attributes objectInstances and rootObjectInstances allow to restrict

the scope. When the attribute objectInstances is present, only the object

instances identified by this attribute are scoped. When the attribute

rootObjectInstances is present, then the subtrees whose root objects are

identified by this attribute are scoped. Both attributes may be present

at the same time meaning the total scope is equal to the sum of both

scopes. Object instances may be scoped by both the objectInstances and

rootObjectInstances attributes. This shall not be considered as an error

by the MnS producer.

When the performance metric requires performance metric production on

multiple managed objects, which is for example the case for KPIs, the

MnS consumer needs to ensure all required objects are scoped. Otherwise

a PerfMetricJob creation request shall fail.

The attribute reportingCtrl specifies the method and associated control

parameters for reporting the produced measurements to MnS consumers.

Three methods are available: file-based reporting with selection of the

file location by the MnS producer, file-based reporting with selection

of the file location by the MnS consumer and stream-based reporting.

For file-based reporting, all performance metrics that are produced

related to a 'PerfMetricJob' instance for a reporting period shall be

stored in a single reporting file.

When the administrative state is set to 'UNLOCKED' after the creation

of a 'PerfMetricJob' the first granularity period shall start. When

the administrative state is set to 'LOCKED' or the operational state

to 'DISABLED', the ongoing reporting period shall be aborted, for

streaming the ongoing granularity period. When the administrative

state is set back to 'UNLOCKED' or the operational state to 'ENABLED'

a new reporting period period shall start, in case of streaming a new

granularity period.

Changes of all other configurable attributes shall take effect only at

the beginning of the next reporting period, for streaming at the

beginning of the next granularity period.

When the 'PerfMetricJob' is deleted, the ongoing reporting period shall

be aborted, for streaming the ongoing granularity period.

A PerfMetricJob creation request shall fail, when the requested

performance metrics, the requested granularity period, the requested

repoting method, or the requested combination thereof is not supported

by the MnS producer.

Creation and deletion of PerfMetricJob instances by MnS consumers is

optional; when not supported, PerfMetricJob instances may be created and

deleted by the system or be pre-installed.";

key id;

uses top3gpp:Top\_Grp ;

container attributes {

uses PerfMetricJobGrp ;

}

}

list ThresholdMonitor {

key id;

description "Represents a threshold monitor for performance metrics.

It can be contained by SubNetwork, ManagedElement, or ManagedFunction.

A threshold monitor checks for threshold crossings of performance metric

values and generates a notification when that happens.

The ThresholdMonitor is used only when NRM based threshold monitoring

is supported.

To activate threshold monitoring, a MnS consumer needs to create a

ThresholdMonitor instance on the MnS producer. For ultimate deactivation

of threshold monitoring, the MnS consumer should delete the monitor to

free up resources on the MnS producer.

For temporary suspension of threshold monitoring, the MnS consumer can

manipulate the value of the administrative state attribute. The MnS

producer may disable threshold monitoring as well, for example in

overload situations. This situation is indicated by the MnS producer with

setting the operational state attribute to disabled. When monitoring is

resumed the operational state is set again to enabled.

All object instances below and including the instance containing the

ThresholdMonitor (base object instance) are scoped for performance

metric production. Performance metrics are monitored only on those

object instances whose object class matches the object class associated

to the performance metrics to be monitored.

The optional attributes objectInstances and rootObjectInstances allow to

restrict the scope. When the attribute objectInstances is present, only

the object instances identified by this attribute are scoped. When the

attribute rootObjectInstances is present, then the subtrees whose root

objects are identified by this attribute are scoped. Both attributes may

be present at the same time meaning the total scope is equal to the sum

of both scopes. Object instances may be scoped by both the objectInstances

and rootObjectInstances attributes. This shall not be considered as an

error by the MnS producer.

Multiple thresholds can be defined for multiple performance metric sets

in a single monitor using thresholdInfoList. The attribute

monitorGranularityPeriod defines the granularity period to be applied.

The value is a multiple of a supported granularity period for the

measurements being monitored.

Each threshold is identified with a number (key) called thresholdLevel.

A threshold is defined using the attributes thresholdValue ,

thresholdDirection and hysteresis.

When hysteresis is absent or carries no information, a threshold is

triggered when the thresholdValue is reached or crossed. When hysteresis

is present, two threshold values are specified for the threshold as

follows: A high treshold value equal to the threshold value plus the

hysteresis value, and a low threshold value equal to the threshold value

minus the hysteresis value. When the monitored performance metric

increases, the threshold is triggered when the high threshold value is

reached or crossed. When the monitored performance metric decreases, the

threshold is triggered when the low threshold value is reached or crossed.

The hsyteresis ensures that the performance metric value can oscillate

around a comparison value without triggering each time the threshold when

the threshold value is crossed.

Using the thresholdDirection attribute a threshold can be configured in

such a manner that it is triggered only when the monitored performance

metric is going up or down upon reaching or crossing the threshold.

A ThresholdMonitor creation request shall be rejected, if the performance

metrics requested to be monitored, the requested granularity period, or

the requested combination thereof is not supported by the MnS producer.

A creation request may fail, when the performance metrics requested to be

monitored are not produced by a PerfMetricJob.

Creation and deletion of ThresholdMonitor instances by MnS consumers is

optional; when not supported, ThresholdMonitor instances may be created

and deleted by the system or be pre-installed.";

uses top3gpp:Top\_Grp ;

container attributes {

uses ThresholdMonitorGrp ;

}

}

}

}

<CODE ENDS>

## D.2.5 module \_3gpp-common-subnetwork.yang

<CODE BEGINS>

module \_3gpp-common-subnetwork {

yang-version 1.1;

namespace "urn:3gpp:sa5:\_3gpp-common-subnetwork";

prefix "subnet3gpp";

import \_3gpp-common-yang-types { prefix types3gpp; }

import \_3gpp-common-top { prefix top3gpp; }

import \_3gpp-common-measurements { prefix meas3gpp; }

import \_3gpp-common-subscription-control { prefix subscr3gpp; }

import \_3gpp-common-fm { prefix fm3gpp; }

import \_3gpp-common-trace { prefix trace3gpp; }

import ietf-yang-schema-mount { prefix yangmnt; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "Defines basic SubNetwork which will be augmented by other IOCs";

reference "3GPP TS 28.623

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Solution Set (SS) definitions

3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)

3GPP TS 28.620

Umbrella Information Model (UIM)";

revision 2021-01-16 { reference "CR-0120"; }

revision 2020-08-06 { reference "CR-0102"; }

revision 2020-06-08 { reference "CR-0092"; }

revision 2020-05-08 {

reference "S5-203316";

}

revision 2020-03-11 {

description "Added KPIs and corrections";

reference "S5-201365, S5-201581, SP-200229";

}

revision 2020-02-24 {

reference "S5-201365";

}

revision 2019-06-17 {

reference "Initial revision";

}

feature ExternalsUnderSubNetwork {

description "Classes representing external entities like EUtranFrequency,

ExternalGNBCUCPFunction, ExternalENBFunction

are contained under a Subnetwork list/class.";

}

feature MeasurementsUnderSubNetwork {

description "The MeasurementSubtree shall be contained under SubNetwork";

}

feature SubscriptionControlUnderSubNetwork {

description "The SubscriptionControlSubtree shall be contained under

SubNetwork";

}

feature FmUnderSubNetwork {

description "The FmSubtree shall be contained under SubNetwork";

}

feature TraceUnderSubNetwork {

description "The TraceSubtree shall be contained under SubNetwork";

}

feature DESManagementFunction {

description "Class representing Distributed SON or Domain-Centralized SON

Energy Saving feature. The DESManagementFunction shall be contained under

SubNetwork.";

}

feature DMROFunction {

description "Class representing D-SON function of MRO feature. The

DMROFunction shall be contained under SubNetwork.";

}

feature DRACHOptimizationFunction {

description "Class representing D-SON function of RACH optimization feature.

The DRACHOptimizationFunction shall be contained under SubNetwork.";

}

feature DPCIConfigurationFunction {

description "Class representing Distributed SON or Domain-Centralized SON

function of PCI configuration feature. The DPCIConfigurationFunction shall

be contained under SubNetwork.";

}

feature CPCIConfigurationFunction {

description "Class representing Cross Domain-Centralized SON function of PCI

configuration feature. The CPCIConfigurationFunction shall be contained

under SubNetwork.";

}

feature CESManagementFunction {

description "Class representing Cross Domain-Centralized SON Energy Saving

feature. The CESManagementFunction shall be contained under SubNetwork.";

}

grouping Domain\_Grp {

description "A domain is a partition of instances of managed entities

such that :

- the group represents a topological structure which describes the

potential for connectivity

- Subject to common administration

- With common characteristics";

leaf dnPrefix {

type types3gpp:DistinguishedName;

reference "Annex C of 32.300 ";

}

leaf userLabel {

type string;

description "A user-friendly (and user assignable) name of this object.";

}

leaf userDefinedNetworkType {

type string;

description "Textual information indicating network type, e.g. 'UTRAN'.";

}

}

grouping SubNetworkGrp {

uses Domain\_Grp;

uses meas3gpp:SupportedPerfMetricGroupGrp;

leaf-list setOfMcc {

description "Set of Mobile Country Code (MCC).

The MCC uniquely identifies the country of domicile

of the mobile subscriber. MCC is part of the IMSI (3GPP TS 23.003)

This list contains all the MCC values in subordinate object

instances to this SubNetwork instance.

See clause 2.3 of 3GPP TS 23.003 for MCC allocation principles.

It shall be supported if there is more than one value in setOfMcc

of the SubNetwork. Otherwise the support is optional.";

type types3gpp:Mcc;

}

leaf priorityLabel {

mandatory true;

type uint32;

}

}

list SubNetwork {

key id;

description "Represents a set of managed entities";

uses top3gpp:Top\_Grp;

container attributes {

uses SubNetworkGrp;

leaf-list parents {

description "Reference to all containg SubNetwork instances

in strict order from the root subnetwork down to the immediate

parent subnetwork.

If subnetworks form a containment hierarchy this is

modeled using references between the child SubNetwork and the parent

SubNetworks.

This reference MUST NOT be present for the top level SubNetwork and

MUST be present for other SubNetworks.";

type leafref {

path "../../../SubNetwork/id";

}

}

leaf-list containedChildren{

description "Reference to all directly contained SubNetwork instances.

If subnetworks form a containment hierarchy this is

modeled using references between the child SubNetwork and the parent

SubNetwork.";

type leafref {

path "../../../SubNetwork/id";

}

}

}

uses meas3gpp:MeasurementSubtree {

if-feature MeasurementsUnderSubNetwork;

}

uses subscr3gpp:SubscriptionControlSubtree {

if-feature SubscriptionControlUnderSubNetwork;

}

uses fm3gpp:FmSubtree {

if-feature FmUnderSubNetwork;

}

uses trace3gpp:TraceSubtree {

if-feature TraceUnderSubNetwork;

}

yangmnt:mount-point children-of-SubNetwork {

description "Mountpoint for ManagedElement";

reference "RFC8528 YANG Schema Mount";

}

// augment external parts here

}

}

<CODE ENDS>

## D.2.6 module \_3gpp-common-top.yang

<CODE BEGINS>

module \_3gpp-common-top {

yang-version 1.1;

namespace urn:3gpp:sa5:\_3gpp-common-top;

prefix top3gpp;

organization "3gpp SA5";

description "The model defines a YANG mapping of the top level

information classes used for management of 5G networks and

network slicing.";

reference

"3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)

3GPP TS 28.620

Umbrella Information Model (UIM)";

revision 2019-06-17 {

description "Initial revision";

}

grouping Top\_Grp {

description "Abstract class supplying a naming attribute";

reference "3GPP TS 28.620";

leaf id {

type string;

description "Key leaf (namingAttribute) for a class/list.

Should be used as a key leaf for lists representing

stage 2 classes.";

reference "3GPP TS 32.300 Name convention for managed objects";

}

}

}

<CODE ENDS>

## D.2.6a module \_3gpp-common-subscription-control.yang

<CODE BEGINS>

module \_3gpp-common-subscription-control {

yang-version 1.1;

namespace "urn:3gpp:sa5:\_3gpp-common-subscription-control";

prefix "subscr3gpp";

import \_3gpp-common-yang-extensions { prefix yext3gpp; }

import \_3gpp-common-top { prefix top3gpp; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "Defines IOCs for subscription and heartbeat control.";

reference "3GPP TS 28.623

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Solution Set (SS) definitions

3GPP TS 28.623";

revision 2023-08-12 { reference "CR-0258"; }

revision 2022-11-03 { reference CR-0192; }

revision 2021-01-16 { reference "CR-0120"; }

revision 2020-08-26 { reference "CR-0106"; }

revision 2019-11-29 { reference "S5-197648 S5-197647 S5-197829 S5-197828"; }

grouping NtfSubscriptionControlGrp {

description "Attributes of a specific notification subscription";

leaf notificationRecipientAddress {

type string;

mandatory true;

}

leaf-list notificationTypes {

type string;

description "Defines the types of notifications that are candidates

for being forwarded to the notification recipient.

If the notificationFilter attribute is not supported or not present

all candidate notifications types are forwarded to the notification;

discriminated by notificationFilter attribute.";

}

list scope {

key "scopeType";

min-elements 1;

max-elements 1;

description "Describes which object instances are selected with

respect to a base object instance.";

leaf scopeType {

type enumeration {

enum BASE\_ONLY;

enum BASE\_ALL;

enum BASE\_NTH\_LEVEL;

enum BASE\_SUBTREE;

}

description "If the optional scopeLevel parameter is not supported

or absent, allowed values of scopeType are BASE\_ONLY and BASE\_ALL.

The value BASE\_ONLY indicates only the base object is selected.

The value BASE\_ALL indicates the base object and all of its

subordinate objects (incl. the leaf objects) are selected.

If the scopeLevel parameter is supported and present, allowed

values of scopeType are BASE\_ALL, BASE\_ONLY, BASE\_NTH\_LEVEL

and BASE\_SUBTREE.

The value BASE\_NTH\_LEVEL indicates all objects on the level,

which is specified by the scopeLevel parameter, below the base

object are selected. The base object is at scopeLevel zero.

The value BASE\_SUBTREE indicates the base object and all of its

subordinate objects down to and including the objects on the level,

which is specified by the scopeLevel parameter, are selected.

The base object is at scopeLevel zero.";

}

leaf scopeLevel {

when '../scopeType = "BASE\_NTH\_LEVEL" or ../scopeType = "BASE\_SUBTREE"';

type uint16;

mandatory true;

description "See description of scopeType.";

}

}

leaf notificationFilter {

type string;

description "Defines a filter to be applied to candidate notifications

identified by the notificationTypes attribute.

If notificationFilter is present, only notifications that pass the

filter criteria are forwarded to the notification recipient; all other

notifications are discarded.

The filter can be applied to any field of a notification.";

}

}

grouping HeartbeatControlGrp {

description "Attributes of HeartbeatControl.";

leaf heartbeatNtfPeriod {

type uint32;

mandatory true;

units seconds;

description "Specifies the periodicity of heartbeat notification emission.

The value of zero has the special meaning of stopping the heartbeat

notification emission.";

}

leaf triggerHeartbeatNtf {

type boolean;

default false;

description "Setting this attribute to 'true' triggers an immediate

additional heartbeat notification emission. Setting the value to

'false' has no observable result.

The periodicity of notifyHeartbeat emission is not changed.

After triggering the heartbeat the system SHALL set the value

back to false.";

yext3gpp:notNotifyable;

}

}

grouping SubscriptionControlSubtree {

description "Contains notification subscription related classes.

Should be used in all classes (or classes inheriting from)

- SubNetwork

- ManagedElement

If some YAM wants to augment these classes/list/groupings they must

augment all user classes!";

list NtfSubscriptionControl {

description "A NtfSubscriptionControl instance represents the

notification subscription of a particular notification recipient.

The scope attribute is used to select managed object instances.

The base object instance of the scope is the object instance

name-containing the NtfSubscriptionControl instance.

The notifications related to the selected managed object instances

are candidates to be sent to the address specified by the

notificationRecipientAddress attribute.

The notificationType attribute and notificationFilter attribute

allow MnS consumers to exercise control over which candidate

notifications are sent to the notificationRecipientAddress.

If the notificationType attribute is supported and present, its

value identifies the

types of notifications that are candidate to be sent to the

notificationRecipientAddress. If the notificationType attribute is

not supported or not present, all types of notifications are

candidate to be sent to notificationRecipientAddress.

If supported, the notificationFilter attribute defines a filter that

is applied to the set of candidate notifications. Only candidate

notifications that pass the filter criteria are sent to the

notificationRecipientAddress. If the notificationFilter attribute is

not supported all candidate notificatios are sent to the

notificationRecipientAddress.

To receive notifications, a MnS consumer has to create

NtfSubscriptionControl object instancess on the MnS producer.

A MnS consumer can create a subscription for another MnS consumer

since it is not required the notificationRecipientAddress be his own

address.

When a MnS consumer does not wish to receive notifications any more

the MnS consumer shall delete the corresponding NtfSubscriptionControl

instance.

Creation and deletion of NtfSubscriptionControl instances by MnS

consumers is optional; when not supported, the NtfSubscriptionControl

instances may be created and deleted by the system or be

pre-installed.";

key id;

uses top3gpp:Top\_Grp;

container attributes {

uses NtfSubscriptionControlGrp;

}

list HeartbeatControl {

description "MnS consumers (i.e. notification recipients) use heartbeat

notifications to monitor the communication channels between

themselves and MnS producers configured to emit notifications.

A HeartbeatControl instance allows controlling the emission of

heartbeat notifications by MnS producers. The recipients of heartbeat

notifications are specified by the notificationRecipientAddress

attribute of the NtfSubscriptionControl instance containing the

HeartbeatControl instance.

Note that the MnS consumer managing the HeartbeatControl instance

and the MnS consumer receiving the heartbeat notifications may not be

the same.

As a pre-condition for the emission of heartbeat notifications, a

HeartbeatControl instance needs to be created. Creation of an instance

with an initial non-zero value of the heartbeatNtfPeriod attribute

triggers an immediate heartbeat notification emission. Creation of an

instance with an initial zero value of the heartbeatPeriod attribute

does not trigger an emission of a heartbeat notification. Deletion of

an instance does not trigger an emission of a heartbeat notification.

Once the instance is created, heartbeat notifications are emitted with

a periodicity defined by the value of the heartbeatNtfPeriod

attribute. No heartbeat notifications are emitted if the value is

equal to zero. Setting a zero value to a non zero value, or a non zero

value to a different non zero value, triggers an immediate heartbeat

notification, that is the base for the new heartbeat period. Setting a

non zero value to a zero value stops emitting heartbeats immediately;

no final heartbeat notification is sent.

Creation and deletion of HeartbeatControl instances by MnS Consumers

is optional; when not supported, the HeartbeatControl instances may be

created and deleted by the system or be pre-installed.

Whether and when to emit heartbeat notifications is controlled by

HeartbeatControl. Subscription for heartbeat is not supported via the

NtfSubscriptionControl.";

max-elements 1;

key id;

uses top3gpp:Top\_Grp;

container attributes {

uses HeartbeatControlGrp;

}

}

}

}

}<CODE ENDS>

## D.2.7 module \_3gpp-common-yang-extensions.yang

<CODE BEGINS>

module \_3gpp-common-yang-extensions {

yang-version 1.1;

namespace urn:3gpp:sa5:\_3gpp-common-yang-extensions ;

prefix yext3gpp ;

organization "3GPP SA5";

description "The module defines YANG extensions needed

3GPP YANG modeling.

Copyright (c) 2019 3GPP. All rights reserved.

Extensions MUST be defined with the following structure in the

description statement:

- What is this statement.

- Newline,

- This statement can be a substatement of the xxx statements with

cardinality x..y.

- This statement can have the following substatements with

cardinality x..y.

- Newline

- Is changing this statement an editorial, BC(backwards compatible)

or NBC(non-BC) change?

- Newline.

- The argument its meaning and type. Preferably use YANG types and

constraints to define the argument's type.

Any extension statement can be added with a

deviation/deviate add statement. In this case the restriction about

the parent statement of the extension SHALL be evaluated based on the

target of the deviation statement.

Support for this module does not mean that a YANG server implements

support for each of these extensions.

Implementers of each specific module using an extensions MUST check

if the server implements support for the used extension.

Note: modules use many extensions which individual

implementations MAY or MAY NOT support.

If support for an extension is missing the extension statement needs

individual handling or it SHOULD be removed from the module using

the extension e.g. with a deviation.

";

revision 2022-11-03 { reference CR-0192; }

revision 2019-06-23 { description "Initial version"; }

extension notNotifyable {

description

"Indicates that data change notifications shall not be sent

for this attribute. If the extension is not present and other

conditions are fulfilled data change notification should be sent.

If a list or container already has the notNotifyable

extension, that is also valid for all contained data nodes.

The statement MUST only be a substatement of a leaf, leaf-list, list,

container statement that is contained within the 'attributes'

container of an IOC and that represents an attribute or sub-parts of

an attribute .

Zero or one notNotifyable statement is allowed per parent statement.

NO substatements are allowed.

Adding this statement is an NBC change, removing it is BC.";

}

extension inVariant {

description

"Indicates that the value for the data node can only be set when its

parent data node is being created. To change the value after that, the

parent data node must be deleted and recreated with the data node

having the new value.

It is unnecessary to use and MUST NOT be used for key leafs.

The statement MUST only be a substatement of a leaf, leaf-list, list

statements that is config=true.

Zero or one inVariant statement is allowed per parent statement.

NO substatements are allowed.

Adding this statement is an NBC change, removing it is BC.";

}

extension initial-value {

description "Specifies a value that the system will set for a leaf

leaf-list if a value is not specified for it when its parent list

or container is created. The value has no effect in any other

modification e.g. changing or removing the value.

The description statement of the parent statement SHOULD contain

the label 'Initial-value: ' followed by the text from the argument.

The statement MUST only be a substatement of a leaf or leaf-list.

The statement MUST NOT be present if the leaf or the leaf-list

has a default statement or the type used for the data node

has a default value.

The statement MUST NOT be used for config=false data or in an

action, rpc or notification.

Zero or one initial-value statements are allowed for a leaf parent

statement. Zero or more initial-value statements are allowed for a

leaf-list parent statement. If the leaf-list is ordered-by user, the

initial values are stored in the order they appear in the YANG definition.

NO substatements are allowed.

Always consider using a YANG-default statement instead.

Modification of the initial-value is a non-backwards-compatible change.

The argument specifies a single initial value for a leaf or leaf-list.

The value MUST be part of the valuespace of the leaf/leaf-list.

It follows the same rules as the argument of the default statement.";

argument "initial-value";

}

}<CODE ENDS>

## D.2.8 module \_3gpp-common-yang-types.yang

<CODE BEGINS>

module \_3gpp-common-yang-types {

yang-version 1.1;

namespace "urn:3gpp:sa5:\_3gpp-common-yang-types";

prefix "types3gpp";

import ietf-inet-types { prefix inet; }

import ietf-yang-types { prefix yang; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "The model defines a YANG mapping of the top level

information classes used for management of 5G networks and

network slicing.";

reference "3GPP TS 28.623";

revision 2021-11-01 { reference "CR-0141"; }

revision 2020-11-06 {

description "Removed incorrect S-NSSAI definitions.";

reference "CR-0118";

}

revision 2020-03-10 {

description "Removed faulty when statements.";

reference “SP-200229”;

}

revision 2019-10-25 {

description "Added ManagedNFProfile.";

reference "S5-194457";

}

revision 2019-10-16 {

description "Added SAP and usageState.";

reference "S5-193518";

}

revision 2019-06-23 {

reference "Initial version.";

}

typedef EnabledDisabled {

type enumeration {

enum DISABLED ;

enum ENABLED ;

}

}

// grouping ManagedNFProfile will be removed as it is

// being moved to \_3gpp-5gc-nrm-nfprofile

grouping ManagedNFProfile {

description "Defines profile for managed NF";

reference "3GPP TS 23.501";

leaf idx { type uint32 ; }

leaf nfInstanceID {

config false;

mandatory true;

type yang:uuid ;

description "This parameter defines profile for managed NF.

The format of the NF Instance ID shall be a

Universally Unique Identifier (UUID) version 4,

as described in IETF RFC 4122 " ;

}

leaf-list nfType {

config false;

min-elements 1;

type NfType;

description "Type of the Network Function" ;

}

leaf hostAddr {

mandatory true;

type inet:host ;

description "Host address of a NF";

}

leaf authzInfo {

type string ;

description "This parameter defines NF Specific Service authorization

information. It shall include the NF type (s) and NF realms/origins

allowed to consume NF Service(s) of NF Service Producer.";

reference "See TS 23.501" ;

}

leaf location {

type string ;

description "Information about the location of the NF instance

(e.g. geographic location, data center) defined by operator";

reference "TS 29.510" ;

}

leaf capacity {

mandatory true;

type uint16 ;

description "This parameter defines static capacity information

in the range of 0-65535, expressed as a weight relative to other

NF instances of the same type; if capacity is also present in the

nfServiceList parameters, those will have precedence over this value.";

reference "TS 29.510" ;

}

leaf nFSrvGroupId {

type string ;

description "This parameter defines identity of the group that is

served by the NF instance.

May be config false or true depending on the ManagedFunction.

Config=true for Udrinfo. Config=false for UdmInfo and AusfInfo.

Shall be present if ../nfType = UDM or AUSF or UDR. ";

reference "TS 29.510" ;

}

leaf-list supportedDataSetIds {

type enumeration {

enum SUBSCRIPTION;

enum POLICY;

enum EXPOSURE;

enum APPLICATION;

}

description "List of supported data sets in the UDR instance.

May be present if ../nfType = UDR";

reference "TS 29.510" ;

}

leaf-list smfServingAreas {

type string ;

description "Defines the SMF service area(s) the UPF can serve.

Shall be present if ../nfType = UPF";

reference "TS 29.510" ;

}

leaf priority {

type uint16;

description "This parameter defines Priority (relative to other NFs

of the same type) in the range of 0-65535, to be used for NF selection;

lower values indicate a higher priority. If priority is also present

in the nfServiceList parameters, those will have precedence over

this value. Shall be present if ../nfType = AMF ";

reference "TS 29.510" ;

}

}

typedef usageState {

type enumeration {

enum IDLE;

enum ACTIVE;

enum BUSY;

}

description "It describes whether or not the resource is actively in

use at a specific instant, and if so, whether or not it has spare

capacity for additional users at that instant. The value is READ-ONLY.";

reference "ITU T Recommendation X.731";

}

grouping SAP {

leaf host {

type inet:host;

mandatory true;

}

leaf port {

type inet:port-number;

mandatory true;

}

description "Service access point.";

reference "TS 28.622";

}

typedef Mcc {

description "The mobile country code consists of three decimal digits,

The first digit of the mobile country code identifies the geographic

region (the digits 1 and 8 are not used):";

type string {

pattern '[02-79][0-9][0-9]';

}

reference "3GPP TS 23.003 subclause 2.2 and 12.1";

}

typedef Mnc {

description "The mobile network code consists of two or three

decimal digits (for example: MNC of 001 is not the same as MNC of 01)";

type string {

pattern '[0-9][0-9][0-9]|[0-9][0-9]';

}

reference "3GPP TS 23.003 subclause 2.2 and 12.1";

}

grouping PLMNId {

leaf mcc {

mandatory true;

type Mcc;

}

leaf mnc {

mandatory true;

type Mnc;

}

reference "TS 23.658";

}

typedef Nci {

description "NR Cell Identity. The NCI shall be of fixed length of 36 bits

and shall be coded using full hexadecimal representation.

The exact coding of the NCI is the responsibility of each PLMN operator";

reference "TS 23.003";

type union {

type string {

length 36;

pattern '[01]+';

}

type string {

length 9;

pattern '[a-fA-F0-9]\*';

}

}

}

typedef OperationalState {

reference "3GPP TS 28.625 and ITU-T X.731";

type enumeration {

enum DISABLED {

value 0;

description "The resource is totally inoperable.";

}

enum ENABLED {

value 1;

description "The resource is partially or fully operable.";

}

}

}

typedef AdministrativeState {

reference "3GPP TS 28.625 and ITU-T X.731";

type enumeration {

enum LOCKED {

value 0;

description "The resource is administratively prohibited from performing

services for its users.";

}

enum UNLOCKED {

value 1;

description "The resource is administratively permitted to perform

services for its users. This is independent of its inherent

operability.";

}

enum SHUTTINGDOWN {

value 2;

description "Use of the resource is administratively permitted to

existing instances of use only. While the system remains in

the shutting down state the manager or the managed element

may at any time cause the resource to transition to the

locked state.";

}

}

}

typedef AvailabilityStatus {

type enumeration {

enum IN\_TEST;

enum FAILED;

enum POWER\_OFF;

enum OFF\_LINE;

enum OFF\_DUTY;

enum DEPENDENCY;

enum DEGRADED;

enum NOT\_INSTALLED;

enum LOG\_FULL;

}

}

typedef CellState {

type enumeration {

enum IDLE;

enum INACTIVE;

enum ACTIVE;

}

}

typedef Nrpci {

type uint32;

description "Physical Cell Identity (PCI) of the NR cell.";

reference "TS 36.211 subclause 6.11";

}

typedef Tac {

type int32 {

range 0..16777215 ;

}

description "Tracking Area Code";

reference "TS 23.003 clause 19.4.2.3";

}

typedef AmfRegionId {

type union {

type uint8 ;

type string {

length 8;

pattern '[01]\*';

}

}

reference "clause 2.10.1 of 3GPP TS 23.003";

}

typedef AmfSetId {

type union {

type uint16 {

range '0..1023';

}

type string {

length 8;

pattern '[01]\*';

}

}

reference "clause 2.10.1 of 3GPP TS 23.003";

}

typedef AmfPointer {

type union {

type uint8 {

range '0..63';

}

type string {

length 6;

pattern '[01]\*';

}

}

reference "clause 2.10.1 of 3GPP TS 23.003";

}

grouping AmfIdentifier {

leaf amfRegionId {

type AmfRegionId;

}

leaf amfSetId {

type AmfSetId;

}

leaf amfPointer {

type AmfPointer;

}

description "The AMFI is constructed from an AMF Region ID,

an AMF Set ID and an AMF Pointer.

The AMF Region ID identifies the region,

the AMF Set ID uniquely identifies the AMF Set within the AMF Region, and

the AMF Pointer uniquely identifies the AMF within the AMF Set. ";

}

// type definitions especially for core NFs

typedef NfType {

type enumeration {

enum NRF;

enum UDM;

enum AMF;

enum SMF;

enum AUSF;

enum NEF;

enum PCF;

enum SMSF;

enum NSSF;

enum UDR;

enum LMF;

enum GMLC;

enum 5G\_EIR;

enum SEPP;

enum UPF;

enum N3IWF;

enum AF;

enum UDSF;

enum BSF;

enum CHF;

}

}

typedef NotificationType {

type enumeration {

enum N1\_MESSAGES;

enum N2\_INFORMATION;

enum LOCATION\_NOTIFICATION;

}

}

typedef Load {

description "Latest known load information of the NF, percentage ";

type uint8 {

range 0..100;

}

}

typedef N1MessageClass {

type enumeration {

enum 5GMM;

enum SM;

enum LPP;

enum SMS;

}

}

typedef N2InformationClass {

type enumeration {

enum SM;

enum NRPPA;

enum PWS;

enum PWS\_BCAL;

enum PWS\_RF;

}

}

grouping DefaultNotificationSubscription {

leaf notificationType {

type NotificationType;

}

leaf callbackUri {

type inet:uri;

}

leaf n1MessageClass {

type N1MessageClass;

}

leaf n2InformationClass {

type N2InformationClass;

}

}

grouping Ipv4AddressRange {

leaf start {

type inet:ipv4-address;

}

leaf end {

type inet:ipv4-address;

}

}

grouping Ipv6PrefixRange {

leaf start {

type inet:ipv6-prefix;

}

leaf end {

type inet:ipv6-prefix;

}

}

typedef NsiId {

type string;

}

typedef UeMobilityLevel {

type enumeration {

enum STATIONARY;

enum NOMADIC;

enum RESTRICTED\_MOBILITY;

enum FULLY\_MOBILITY;

}

}

typedef ResourceSharingLevel {

type enumeration {

enum SHARED;

enum NOT\_SHARED;

}

}

typedef TxDirection {

type enumeration {

enum DL;

enum UL;

enum DL\_AND\_UL;

}

}

grouping AddressWithVlan {

leaf ipAddress {

type inet:ip-address;

}

leaf vlanId {

type uint16;

}

}

typedef DistinguishedName { // TODO is this equivalent to TS 32.300 ?

type string {

pattern '([a-zA-Z][a-zA-Z0-9-]\*=(\\( |#|\\|>|<|;|"|\+|,|[a-fA-F0-9]{2})|[^\\><;"+,# ])'

+ '((\\( |#|\\|>|<|;|"|\+|,|[a-fA-F0-9]{2})|[^\\><;"+,])\*'

+ '(\\( |#|\\|>|<|;|"|\+|,|[a-fA-F0-9]{2})|[^\\><;"+, ]))?'

+ '[,\+])\*[a-zA-Z][a-zA-Z0-9-]\*=(\\( |#|\\|>|<|;|"|\+|,|[a-fA-F0-9]{2})|[^\\><;"+,# ])'

+ '((\\( |#|\\|>|<|;|"|\+|,|[a-fA-F0-9]{2})'

+ '|[^\\><;"+,])\*(\\( |#|\\|>|<|;|"|\+|,|[a-fA-F0-9]{2})|[^\\><;"+, ]))?';

}

description "Represents the international standard for the representation

of Distinguished Name (RFC 4512).

The format of the DistinguishedName REGEX is:

{AttributeType = AttributeValue}

AttributeType consists of alphanumeric and hyphen (OIDs not allowed).

All other characters are restricted.

The Attribute value cannot contain control characters or the

following characters : \\ > < ; \" + , (Comma) and White space

The Attribute value can contain the following characters if they

are excaped : \\ > < ; \" + , (Comma) and White space

The Attribute value can contain control characters if its an escaped

double digit hex number.

Examples could be

UID=nobody@example.com,DC=example,DC=com

CN=John Smith,OU=Sales,O=ACME Limited,L=Moab,ST=Utah,C=US";

reference "RFC 4512 Lightweight Directory Access Protocol (LDAP):

Directory Information Models";

} // recheck regexp it doesn't handle posix [:cntrl:]

typedef QOffsetRange {

type int8 {

range "-24 | -22 | -20 | -18 | -16 | -14 | -12 | -10 | -8 | -6 | " +

" -5 | -4 | -3 | -2 | -1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 8 | 10 | " +

" 12 | 14 | 16 | 18 | 20 | 22 | 24";

}

units dB;

}

}

<CODE ENDS>

## D.2.9 module \_3gpp-common-fm.yang

<CODE BEGINS>

module \_3gpp-common-fm {

yang-version 1.1;

namespace "urn:3gpp:sa5:\_3gpp-common-fm";

prefix "fm3gpp";

import ietf-yang-types { prefix yang; }

import \_3gpp-common-top { prefix top3gpp; }

import \_3gpp-common-yang-types { prefix types3gpp; }

import \_3gpp-common-yang-extensions { prefix yext3gpp; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "Defines a Fault Management model";

reference "3GPP TS 28.623

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Solution Set (SS) definitions

3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)";

revision 2022-11-03 { reference CR-0192; }

revision 2021-08-08 { reference "CR-0132"; }

revision 2021-06-02 { reference "CR-0130"; }

revision 2020-06-03 { reference "CR-0091"; }

revision 2020-02-24 {

reference "S5-201365";

}

typedef eventType {

type enumeration {

enum COMMUNICATIONS\_ALARM {

value 2;

}

enum QUALITY\_OF\_SERVICE\_ALARM {

value 3;

}

enum PROCESSING\_ERROR\_ALARM {

value 4;

}

enum EQUIPMENT\_ALARM {

value 5;

}

enum ENVIRONMENTAL\_ALARM {

value 6;

}

enum INTEGRITY\_VIOLATION {

value 7;

}

enum OPERATIONAL\_VIOLATION {

value 8;

}

enum PHYSICAL\_VIOLATIONu {

value 9;

}

enum SECURITY\_SERVICE\_OR\_MECHANISM\_VIOLATION {

value 10;

}

enum TIME\_DOMAIN\_VIOLATION {

value 11;

}

}

description "General category for the alarm.";

}

typedef severity-level {

type enumeration {

enum CRITICAL { value 3; }

enum MAJOR { value 4; }

enum MINOR { value 5; }

enum WARNING { value 6; }

enum INDETERMINATE { value 7; }

enum CLEARED { value 8; }

}

description "The possible alarm serverities.

Aligned with ERICSSON-ALARM-MIB.";

}

grouping AlarmRecordGrp {

description "Contains alarm information of an alarmed object instance.

A new record is created in the alarm list when an alarmed object

instance generates an alarm and no alarm record exists with the same

values for objectInstance, alarmType, probableCause and specificProblem.

When a new record is created the MnS producer creates an alarmId, that

unambiguously identifies an alarm record in the AlarmList.

Alarm records are maintained only for active alarms. Inactive alarms are

automatically deleted by the MnS producer from the AlarmList.

Active alarms are alarms whose

a) perceivedSeverity is not CLEARED, or whose

b) perceivedSeverity is CLEARED and its ackState is not ACKNOWLEDED.";

leaf alarmId {

type string;

mandatory true;

description "Identifies the alarmRecord";

yext3gpp:notNotifyable;

}

leaf objectInstance {

type string;

config false ;

mandatory true;

yext3gpp:notNotifyable;

}

leaf notificationId {

type int32;

config false ;

mandatory true;

yext3gpp:notNotifyable;

}

leaf alarmRaisedTime {

type yang:date-and-time ;

config false ;

yext3gpp:notNotifyable;

}

leaf alarmChangedTime {

type yang:date-and-time ;

config false ;

description "not applicable if related alarm has not changed";

yext3gpp:notNotifyable;

}

leaf alarmClearedTime {

type yang:date-and-time ;

config false ;

description "not applicable if related alarm was not cleared";

yext3gpp:notNotifyable;

}

leaf alarmType {

type eventType;

config false ;

description "General category for the alarm.";

yext3gpp:notNotifyable;

}

leaf probableCause {

type string;

config false ;

yext3gpp:notNotifyable;

}

leaf specificProblem {

type string;

config false ;

reference "ITU-T Recommendation X.733 clause 8.1.2.2.";

yext3gpp:notNotifyable;

}

leaf perceivedSeverity {

type severity-level;

description "This is Writable only if producer supports consumer

to set perceivedSeverity to CLEARED";

yext3gpp:notNotifyable;

}

leaf backedUpStatus {

type string;

config false ;

description "Indicates if an object (the MonitoredEntity) has a back

up. See definition in ITU-T Recommendation X.733 clause 8.1.2.4.";

yext3gpp:notNotifyable;

}

leaf backUpObject {

type string;

config false ;

yext3gpp:notNotifyable;

}

leaf trendIndication {

type string;

config false ;

description "Indicates if some observed condition is getting better,

worse, or not changing. ";

reference "ITU-T Recommendation X.733 clause 8.1.2.6.";

yext3gpp:notNotifyable;

}

grouping ThresholdPackGrp {

leaf thresholdLevel {

type string;

}

leaf thresholdValue {

type string;

}

leaf hysteresis {

type string;

description "The hysteresis has a threshold high and a threshold

low value that are different from the threshold value.

A hysteresis, therefore, defines the threshold-high and

threshold-low levels within which the measurementType value is

allowed to oscillate without triggering the threshold crossing

notification.";

}

}

grouping ThresholdInfoGrp {

leaf measurementType {

type string;

mandatory true;

}

leaf direction {

type enumeration {

enum INCREASING;

enum DECREASING;

}

mandatory true;

description "

If it is 'Increasing', the threshold crossing notification is

triggered when the measurement value equals or exceeds a

thresholdValue.

If it is 'Decreasing', the threshold crossing notification is

triggered when the measurement value equals or below a

thresholdValue.";

}

uses ThresholdPackGrp;

}

list thresholdInfo {

config false ;

uses ThresholdInfoGrp;

yext3gpp:notNotifyable;

}

leaf stateChangeDefinition {

type string;

config false ;

description "Indicates MO attribute value changes. See definition

in ITU-T Recommendation X.733 clause 8.1.2.11.";

yext3gpp:notNotifyable;

}

leaf monitoredAttributes {

type string;

config false ;

description "Indicates MO attributes whose value changes are being

monitored.";

reference "ITU-T Recommendation X.733 clause 8.1.2.11.";

yext3gpp:notNotifyable;

}

leaf proposedRepairActions {

type string;

config false ;

description "Indicates proposed repair actions. See definition in

ITU-T Recommendation X.733 clause 8.1.2.12.";

yext3gpp:notNotifyable;

}

leaf additionalText {

type string;

config false ;

yext3gpp:notNotifyable;

}

anydata additionalInformation {

config false ;

yext3gpp:notNotifyable;

}

leaf rootCauseIndicator {

type enumeration {

enum YES;

enum NO;

}

config false ;

description "It indicates that this AlarmInformation is the root cause

of the events captured by the notifications whose identifiers are in

the related CorrelatedNotification instances.";

yext3gpp:notNotifyable;

}

leaf ackTime {

type yang:date-and-time ;

config false ;

description "It identifies the time when the alarm has been

acknowledged or unacknowledged the last time, i.e. it registers the

time when ackState changes.";

yext3gpp:notNotifyable;

}

leaf ackUserId {

type string;

description "It identifies the last user who has changed the

Acknowledgement State.";

yext3gpp:notNotifyable;

}

leaf ackSystemId {

type string;

description "It identifies the system (Management System) that last

changed the ackState of an alarm, i.e. acknowledged or unacknowledged

the alarm.";

yext3gpp:notNotifyable;

}

leaf ackState {

type enumeration {

enum ACKNOWLEDGED {

description "The alarm has been acknowledged.";

}

enum UNACKNOWLEDGED {

description "The alarm has unacknowledged or the alarm has never

been acknowledged.";

}

}

yext3gpp:notNotifyable;

}

leaf clearUserId {

type string;

description "Carries the identity of the user who invokes the

clearAlarms operation.";

yext3gpp:notNotifyable;

}

leaf clearSystemId {

type string;

yext3gpp:notNotifyable;

}

leaf serviceUser {

type string;

config false ;

description "It identifies the service-user whose request for service

provided by the serviceProvider led to the generation of the

security alarm.";

yext3gpp:notNotifyable;

}

leaf serviceProvider {

type string;

config false ;

description "It identifies the service-provider whose service is

requested by the serviceUser and the service request provokes the

generation of the security alarm.";

yext3gpp:notNotifyable;

}

leaf securityAlarmDetector {

type string;

config false ;

yext3gpp:notNotifyable;

}

}

grouping AlarmListGrp {

description "Represents the AlarmList IOC.";

leaf administrativeState {

type types3gpp:AdministrativeState ;

default LOCKED;

description "When set to UNLOCKED, the alarm list is updated.

When the set to LOCKED, the existing alarm records are not

updated, and new alarm records are not added to the alarm list.";

}

leaf operationalState {

type types3gpp:OperationalState ;

default DISABLED;

config false;

description "The producer sets this attribute to ENABLED, indicating

that it has the resource and ability to record alarm in AlarmList

else, it sets the attribute to DISABLED.";

}

leaf numOfAlarmRecords {

type uint32 ;

config false;

mandatory true;

description "The number of alarm records in the AlarmList";

yext3gpp:notNotifyable;

}

leaf lastModification {

type yang:date-and-time ;

config false;

description "The last time when an alarm record was modified";

yext3gpp:notNotifyable;

}

list alarmRecords {

key alarmId;

description "List of alarmRecords";

uses AlarmRecordGrp;

yext3gpp:notNotifyable;

}

}

grouping FmSubtree {

description "Contains FM related classes.

Should be used in all classes (or classes inheriting from)

- SubNetwork

- ManagedElement

If some YAM wants to augment these classes/list/groupings they must

augment all user classes!";

list AlarmList {

key id;

max-elements 1;

description "The AlarmList represents the capability to store and manage

alarm records. The management scope of an AlarmList is defined by all

descendant objects of the base managed object, which is the object

name-containing the AlarmList, and the base object itself.

AlarmList instances are created by the system or are pre-installed.

They cannot be created nor deleted by MnS consumers.

When the alarm list is locked or disabled, the existing alarm records

are not updated, and new alarm records are not added to the alarm list";

uses top3gpp:Top\_Grp ;

container attributes {

uses AlarmListGrp ;

}

}

}

}<CODE ENDS>

D.2.10 module \_3gpp-common-trace.yang

<CODE BEGINS>

module \_3gpp-common-trace {

yang-version 1.1;

namespace "urn:3gpp:sa5:\_3gpp-common-trace";

prefix "trace3gpp";

import \_3gpp-common-top { prefix top3gpp; }

import \_3gpp-common-yang-types {prefix types3gpp; }

import ietf-inet-types { prefix inet; }

organization "3GPP SA5";

contact "https://www.3gpp.org/DynaReport/TSG-WG--S5--officials.htm?Itemid=464";

description "Trace handling";

reference "3GPP TS 28.623

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Solution Set (SS) definitions

3GPP TS 28.622

Generic Network Resource Model (NRM)

Integration Reference Point (IRP);

Information Service (IS)";

revision 2025-02-04 { reference "CR-0501"; }

revision 2024-11-05 { reference "CR-0468"; }

revision 2023-11-04 { reference "CR-0292 CR-0300"; }

revision 2023-02-16 { reference "CR-0231"; }

revision 2023-02-14 { reference "CR-0235"; }

revision 2022-04-27 { reference "CR-0158"; }

revision 2021-10-18 { reference "CR-0139"; }

revision 2021-07-22 { reference "CR-0137"; }

revision 2021-01-25 { reference "CR-0122"; }

revision 2020-11-16 { reference "CR-0117"; }

revision 2020-08-06 { reference "CR-0102"; }

grouping TraceReference {

leaf mcc {

mandatory true;

type types3gpp:Mcc;

}

leaf mnc {

mandatory true;

type types3gpp:Mnc;

}

leaf traceId {

mandatory true;

type int64;

}

}

grouping TraceJobGrp {

leaf jobType {

type enumeration {

enum IMMEDIATE\_MDT\_ONLY;

enum LOGGED\_MDT\_ONLY;

enum TRACE\_ONLY;

enum IMMEDIATE\_MDT\_AND\_TRACE;

enum RLF\_REPORT\_ONLY;

enum RCEF\_REPORT\_ONLY;

enum LOGGED\_MBSFN\_MDT;

}

default TRACE\_ONLY;

description "Specifies the MDT mode and it specifies also whether the

TraceJob represents only MDT, Logged MBSFN MDT, Trace or a combined

Trace and MDT job. The attribute is applicable for Trace, MDT, RCEF and

RLF reporting.";

reference "Clause 5.9a of 3GPP TS 32.422 for additional details on the

allowed values.";

}

list listOfInterfaces {

key idx;

must 'count(MSCServerInterfaces)+count(MGWInterfaces)+count(RNCInterfaces)'

+'+count(SGSNInterfaces)+count(GGSNInterfaces)+count(S-CSCFInterfaces)'

+'+count(P-CSCFInterfaces)+count(I-CSCFInterfaces)+count(MRFCInterfaces)'

+'+count(MGCFInterfaces)+count(IBCFInterfaces)+count(E-CSCFInterfaces)'

+'+count(BGCFInterfaces)+count(ASInterfaces)+count(HSSInterfaces)'

+'+count(EIRInterfaces)+count(BM-SCInterfaces)+count(MMEInterfaces)'

+'+count(SGWInterfaces)+count(PDN\_GWInterfaces)+count(eNBInterfaces)'

+'+count(en-gNBInterfaces)+count(AMFInterfaces)+count(AUSFInterfaces)'

+'+count(NEFInterfaces)+count(NRFInterfaces)+count(NSSFInterfaces)'

+'+count(PCFInterfaces)+count(SMFInterfaces)+count(SMSFInterfaces)'

+'+count(UDMInterfaces)+count(UPFInterfaces)+count(ng-eNBInterfaces)'

+'+count(gNB-CU-CPInterfaces)+count(gNB-CU-UPInterfaces)'

+'+count(gNB-DUInterfaces)';

description "Specifies the interfaces that need to be traced.

The attribute is applicable only for Trace. In

case this attribute is not used, it carries a null semantic.";

reference "Clause 5.5 of 3GPP TS 32.422 for additional details on the

allowed values.";

leaf idx { type uint32 ; }

leaf-list MSCServerInterfaces {

type enumeration {

enum A ;

enum Iu-CS ;

enum Mc ;

enum MAP-G ;

enum MAP-B ;

enum MAP-E ;

enum MAP-F ;

enum MAP-D ;

enum MAP-C ;

enum CAP ;

}

}

leaf-list MGWInterfaces {

type enumeration {

enum Mc ;

enum Nb-UP ;

enum Iu-UP ;

}

}

leaf-list RNCInterfaces {

type enumeration {

enum Iu-CS ;

enum Iu-PS ;

enum Iur ;

enum Iub ;

enum Uu ;

}

}

leaf-list SGSNInterfaces {

type enumeration {

enum Gb ;

enum Iu-PS ;

enum Gn ;

enum MAP-Gr ;

enum MAP-Gd ;

enum MAP-Gf ;

enum Ge ;

enum Gs ;

enum S6d ;

enum S4 ;

enum S3 ;

enum S13 ;

}

}

leaf-list GGSNInterfaces {

type enumeration {

enum Gn ;

enum Gi ;

enum Gmb ;

}

}

leaf-list S-CSCFInterfaces {

type enumeration {

enum Mw ;

enum Mg ;

enum Mr ;

enum Mi ;

}

}

leaf-list P-CSCFInterfaces {

type enumeration {

enum Gm ;

enum Mw ;

}

}

leaf-list I-CSCFInterfaces {

type enumeration {

enum Cx ;

enum Dx ;

enum Mg ;

enum Mw ;

}

}

leaf-list MRFCInterfaces {

type enumeration {

enum Mp ;

enum Mr ;

}

}

leaf-list MGCFInterfaces {

type enumeration {

enum Mg ;

enum Mj ;

enum Mn ;

}

}

leaf-list IBCFInterfaces {

type enumeration {

enum Ix ;

enum Mx ;

}

}

leaf-list E-CSCFInterfaces {

type enumeration {

enum Mw ;

enum Ml ;

enum Mm ;

enum Mi-Mg ;

}

}

leaf-list BGCFInterfaces {

type enumeration {

enum Mi ;

enum Mj ;

enum Mk ;

}

}

leaf-list ASInterfaces {

type enumeration {

enum Dh ;

enum Sh ;

enum ISC ;

enum Ut ;

}

}

leaf-list HSSInterfaces {

type enumeration {

enum MAP-C ;

enum MAP-D ;

enum Gc ;

enum Gr ;

enum Cx ;

enum S6d ;

enum S6a ;

enum Sh ;

}

}

leaf-list EIRInterfaces {

type enumeration {

enum MAP-F ;

enum S13 ;

enum MAP-Gf ;

}

}

leaf-list BM-SCInterfaces {

type enumeration {

enum Gmb ;

}

}

leaf-list MMEInterfaces {

type enumeration {

enum S1-MME ;

enum S3 ;

enum S6a ;

enum S10 ;

enum S11 ;

enum S13 ;

}

}

leaf-list SGWInterfaces {

type enumeration {

enum S4 ;

enum S5 ;

enum S8 ;

enum S11 ;

enum Gxc ;

}

}

leaf-list PDN\_GWInterfaces {

type enumeration {

enum S2a ;

enum S2b ;

enum S2c ;

enum S5 ;

enum S6b ;

enum Gx ;

enum S8 ;

enum SGi ;

}

}

leaf-list eNBInterfaces {

type enumeration {

enum S1-MME ;

enum X2 ;

}

}

leaf-list en-gNBInterfaces {

type enumeration {

enum S1-MME ;

enum X2 ;

enum Uu ;

enum F1-C ;

enum E1 ;

}

}

leaf-list AMFInterfaces {

type enumeration {

enum N1 ;

enum N2 ;

enum N8 ;

enum N11 ;

enum N12 ;

enum N14 ;

enum N15 ;

enum N20 ;

enum N22 ;

enum N26 ;

}

}

leaf-list AUSFInterfaces {

type enumeration {

enum N12 ;

enum N13 ;

}

}

leaf-list NEFInterfaces {

type enumeration {

enum N29 ;

enum N30 ;

enum N33 ;

}

}

leaf-list NRFInterfaces {

type enumeration {

enum N27 ;

}

}

leaf-list NSSFInterfaces {

type enumeration {

enum N22 ;

enum N31 ;

}

}

leaf-list PCFInterfaces {

type enumeration {

enum N5 ;

enum N7 ;

enum N15 ;

}

}

leaf-list SMFInterfaces {

type enumeration {

enum N4 ;

enum N7 ;

enum N10 ;

enum N11 ;

enum S5-C ;

enum N38 ;

enum N16 ;

enum N16a ;

}

}

leaf-list SMSFInterfaces {

type enumeration {

enum N20 ;

enum N21 ;

}

}

leaf-list UDMInterfaces {

type enumeration {

enum N8 ;

enum N10 ;

enum N13 ;

enum N21 ;

}

}

leaf-list UPFInterfaces {

type enumeration {

enum N4 ;

}

}

leaf-list ng-eNBInterfaces {

type enumeration {

enum NG-C ;

enum Xn-C ;

enum Uu ;

}

}

leaf-list gNB-CU-CPInterfaces {

type enumeration {

enum NG-C ;

enum Xn-C ;

enum Uu ;

enum F1-C ;

enum E1 ;

enum X2-C ;

}

}

leaf-list gNB-CU-UPInterfaces {

type enumeration {

enum E1 ;

}

}

leaf-list gNB-DUInterfaces {

type enumeration {

enum F1-C ;

}

}

}

leaf-list listOfNeTypes {

type enumeration {

enum MSC\_SERVER;

enum SGSN;

enum MGW;

enum GGSN;

enum RNC;

enum BM\_SC;

enum MME;

enum SGW;

enum PGW;

enum ENB;

enum EN\_GNB;

enum GNB\_CU\_CP;

enum GNB\_CU\_UP;

enum GNB\_DU;

}

description "Specifies the network element types where the trace should

be activated. The attribute is applicable only for Trace with

Signalling Based Trace activation. In case this attribute is not used,

it carries a null semantic";

reference "Clause 5.4 of 3GPP TS 32.422";

}

leaf PLMNTarget {

type string;

mandatory true;

description "Specifies which PLMN that the subscriber of the session to

be recorded uses as selected PLMN. PLMN Target might differ from the

PLMN specified in the Trace Reference";

reference "Clause 5.9b of 3GPP TS 32.422";

}

leaf traceReportingConsumerURI {

when '../traceReportingFormat = "STREAMING"';

type inet:uri;

mandatory true;

description "URI of the Streaming Trace data reporting MnS consumer

(a.k.a. streaming target).";

reference "Clause 5.9 of 3GPP TS 32.422";

}

leaf traceCollectionEntityIPAddress {

when '../traceReportingFormat = "FILE-BASED" or '

+'../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

type union {

type inet:uri;

type inet:ip-address;

}

mandatory true;

description "Specifies the address of the Trace Collection Entity when

the attribute traceReportingFormat is configured for the file-based

reporting. The attribute is applicable for both Trace and MDT.";

reference "Clause 5.9 of 3GPP TS 32.422";

}

leaf traceDepth {

when '../jobType = "TRACE\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type enumeration {

enum MINIMUM;

enum MEDIUM;

enum MAXIMUM;

enum VENDORMINIMUM;

enum VENDORMEDIUM;

enum VENDORMAXIMUM;

}

default MAXIMUM;

description "Specifies the trace depth. The attribute is applicable only

for Trace. In case this attribute is not used, it carries a null semantic.";

reference "Clause 5.3 of 3GPP TS 32.422";

}

list traceReference {

uses TraceReference;

key "mcc mnc traceId";

max-elements 1;

description "A globally unique identifier, which uniquely identifies the

Trace Session that is created by the TraceJob.

In case of shared network, it is the MCC and MNC of the Participating

Operator that request the trace session that shall be provided.

The attribute is applicable for both Trace and MDT.";

reference "Clause 5.7 of 3GPP TS 32.422";

}

leaf traceRecordingSessionReference {

type string;

description "An identifier, which identifies the Trace Recording Session.

The attribute is applicable for both Trace and MDT.";

reference "Clause 5.7 of 3GPP TS 32.422";

}

leaf traceReportingFormat {

type enumeration {

enum FILE-BASED;

enum STREAMING;

}

default FILE-BASED;

description "Specifies the trace reporting format - streaming trace

reporting or file-based trace reporting";

reference "Clause 5.11 of 3GPP TS 32.422";

}

list traceTarget {

key "targetIdType targetIdValue";

max-elements 1;

leaf targetIdType {

type enumeration {

enum IMSI;

enum IMEI;

enum IMEISV;

enum PUBLIC\_ID;

enum UTRAN\_CELL;

enum E-UTRAN\_CELL;

enum NG-RAN\_CELL;

enum ENB;

enum RNC;

enum GNB;

enum SUPI;

}

}

leaf targetIdValue {

type string;

}

description "Specifies the target object of the Trace and MDT. The

attribute is applicable for both Trace and MDT. This attribute

includes the ID type of the target as an enumeration and the ID value.

The traceTarget shall be PUBLIC\_ID in case of a Management Based

Activation is done to an SCSCFFunction (Serving Call Session Control Function)

or PCSCFFunction (Proxy Call Session Control Function)

The traceTarget shall be UTRAN\_CELL only in case of

UTRAN cell traffic trace function.

The traceTarget shall be E-UTRAN\_CELL only in case of E-UTRAN cell

traffic trace function.

The traceTarget shall be NG-RAN\_CELL only in case of E-UTRAN cell

traffic trace function.

The traceTarget shall be either IMSI, IMEI or

IMEISV if the Trace Session is activated to any of the following

ManagedEntity(ies):

- HssFunction

- MscServerFunction

- SgsnFunction

- GgsnFunction

- BmscFunction

- RncFunction

- MmeFunction

- ServingGWFunction

- PGWFunction

The traceTarget shall be either SUPI or IMEISV if the Trace Session

is activated to any of the following ManagedEntity(ies):

- AFFunction

- AMFFunction

- AUSFunction

- NEFFunction

- NRFFunction

- NSSFFunction

- PCFFunction

- SMFFunction

- UPFFunction

- UDMFunction

In case of signalling based MDT, the traceTarget attribute shall be

able to carry PUBLIC\_ID, IMSI, IMEI, IMEISV or SUPI.

In case of management based Immediate MDT, the traceTarget attribute

shall be null value.

In case of management based Logged MDT, the traceTarget attribute

shall carry an eNB or a gNB or an RNC.

The Logged MDT should be initiated on the specified eNB/gNB/RNC in traceTarget.

In case of RLF reporting, or RCEF reporting, the traceTarget

attribute shall be null value.";

reference "3GPP TS 32.422";

}

leaf triggeringEvents {

when '../jobType = "TRACE\_ONLY" or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type enumeration {

enum MO\_MT\_CALLS ;

enum MO\_MT\_SMS ;

enum LU\_IMSIattach\_IMSIdetach ;

enum HANDOVER ;

enum SS ;

enum PDPcontext ;

enum RAU\_GPRSattach\_GPRSdetach ;

enum MBMScontext ;

enum CONTEXT ;

enum SIPsession\_StandaloneTransaction ;

enum MBMSactivation ;

enum UEinitiatedPDNconnectivityRequest ;

enum ServiceRequest ;

enum InitialAttach\_TAU\_Detach ;

enum UEinitiatedPDNdisconnection ;

enum BearerActivationModificationDeletion ;

enum Handover ;

enum PDNconnectionCreation ;

enum PDNconnectionTermination ;

enum Registration ;

enum UEderegistration ;

enum NetworkDeregistration ;

enum UEMobilityFromEPC ;

enum UEMobilityToEPC ;

enum PDUsessionEstablishment ;

enum PDUsessionModification ;

enum PDUsessionRelease ;

enum PDUsessionUPactivationDeactivation ;

enum MobilityBtw3gppAndN3gppTo5GC ;

enum MobilityFromEpc ;

enum AMpolicy ;

enum SMpolicy ;

enum Authorization ;

enum BDTpolicy ;

enum N4Session ;

enum UEauthentication ;

enum EventExposure ;

enum PFDmanagement ;

enum ParameterProvision ;

enum Trigger ;

enum NFmanagement ;

enum NFdiscovery ;

enum NSSelection ;

enum NSSAI ;

enum SMservice ;

enum UEcontext ;

enum SubscriberData ;

}

mandatory true;

description "Specifies the triggering event parameter of the trace session.

The attribute is applicable only for Trace. In case this attribute is

not used, it carries a null semantic.";

reference "Clause 5.1 of 3GPP TS 32.422";

}

leaf anonymizationOfMDTData {

when ../areaScope ;

type enumeration {

enum NO\_IDENTITY;

enum TAC\_OF\_IMEI;

}

default NO\_IDENTITY;

description "Specifies the level of anonymization for management based MDT.";

reference "Clause 5.10.12 of 3GPP TS 32.422";

}

list areaConfigurationForNeighCell {

when '../jobType = "LOGGED\_MDT\_ONLY"';

key "idx";

min-elements 1;

leaf idx { type uint32 ; }

description "It specifies the area for which UE is requested to perform

measurement logging for neighbour cells which have list of frequencies.

If it is not configured, the UE shall perform measurement logging for

all the neighbour cells.

Applicable only to Logged MDT in NR.";

reference "Clause 5.10.26 of 3GPP TS 32.422";

uses AreaConfigGrp;

}

grouping AreaConfigGrp {

description "Represents the AreaConfig dataType.

This <<dataType>> defines the area for which measurement logging should

be performed. It is described by a list of cells and a list of

frequencies.";

list freqInfo {

key arfcn;

min-elements 1;

max-elements 32;

description "It specifies the carrier frequency and bands used in

a cell.";

uses FreqInfoGrp ;

}

leaf-list pciList {

type uint32 {

range 0..1007;

}

min-elements 1;

max-elements 32;

description "List of neighbour cells subject for MDT scope.";

}

}

grouping FreqInfoGrp {

description "Represents the FreqInfo dataType.

This <<dataType>> defines the RF reference frequency and the frequency

operating bands used in a cell for a given direction (UL or DL) in FDD

or for both UL and DL directions in TDD";

leaf arfcn {

type uint32 {

range 0..3279165;

}

mandatory true;

description "RF Reference Frequency as defined in TS 38.104,

clause 5.4.2.1. The frequency provided identifies the absolute

frequency position of the reference resource block (Common RB 0)

of the carrier. Its lowest subcarrier is also known as Point A.";

}

leaf-list freqBands {

type uint32 {

range 1..1024;

}

min-elements 1;

description "List of NR frequency operating bands. Primary NR

Operating Band as defined in TS 38.104, clause 5.4.2.3.

The value 1 corresponds to n1, value 2 corresponds to NR operating

band n2, etc.";

}

}

list areaScope {

description "it specifies the area where data shall be collected";

key idx;

min-elements 0;

max-elements 1;

leaf idx {type string; }

uses AreaScopeGrp;

}

leaf collectionPeriodRRMLTE {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type enumeration {

enum 100ms;

enum 1000ms;

enum 1024ms;

enum 1280ms;

enum 2048ms;

enum 2560ms;

enum 5120ms;

enum 10000ms;

enum 10240ms;

enum 1min;

}

description "Specifies the collection period for collecting RRM configured

measurement samples for M3 in LTE. The attribute is applicable only

for Immediate MDT. In case this attribute is not used, it carries a

null semantic.";

reference "Clause 5.10.20 of 3GPP TS 32.422";

}

leaf collectionPeriodM6LTE {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type uint32 {

range "1024|2048|5120|10240";

}

units milliseconds;

description "Specifies the collection period for the Packet Delay

measurement (M6) for MDT taken by the eNB. The attribute is applicable

only for Immediate MDT. In case this attribute is not used,

it carries a null semantic.";

reference "Clause 5.10.32 of TS 32.422 ";

}

leaf collectionPeriodM7LTE {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type uint16 {

range 1..60 ;

}

description "It specifies the collection period for the Data Volume (M6)

and Throughput measurements (M7) for UMTS MDT taken by RNC. The

attribute is applicable only for Immediate MDT. In case this attribute

is not used, it carries a null semantic.";

reference "Clause 5.10.22 of TS 32.422 .";

}

leaf collectionPeriodRRMUMTS {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type uint32 {

range "100|250|500|1000|2000|"

+"3000|4000|6000";

}

units milliseconds;

description "Specifies the collection period for collecting RRM configured

measurement samples for M3, M4, M5 in UMTS. The attribute is applicable

only for Immediate MDT. In case this attribute is not used, it carries

a null semantic";

reference "Clause 5.10.21 of 3GPP TS 32.422";

}

leaf collectionPeriodRRMNR {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type uint32 {

range "1024|2048|5120|10240|60000";

}

units milliseconds;

description "Specifies the collection period for collecting RRM

configured measurement samples for M4, M5 in NR. The attribute is

applicable only for Immediate MDT. In case this attribute is not

used, it carries a null semantic.";

reference "Clause 5.10.30 of 3GPP TS 32.422";

}

leaf collectionPeriodM6NR {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type enumeration {

enum 120ms;

enum 240ms;

enum 480ms;

enum 640ms;

enum 1024ms;

enum 2048ms;

enum 5120ms;

enum 10240ms;

enum 20480ms;

enum 40960ms;

enum 1min;

enum 6min;

enum 12min;

enum 30min;

}

description "It specifies the collection period for the Packet Delay

measurement (M6) for NR MDT taken by the gNB. The attribute is

applicable only for Immediate MDT. In case this attribute is not used,

it carries a null semantic.";

reference "clause 5.10.34 of TS 32.422";

}

leaf collectionPeriodM7NR {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type uint32 {

range "1..60";

}

description "It specifies the collection period for the Packet Loss Rate

measurement (M7) for NR MDT taken by the gNB. The attribute is

applicable only for Immediate MDT. In case this attribute is not used,

it carries a null semantic.";

reference "clause 5.10.35 of TS 32.422";

}

leaf eventListForTriggeredMeasurement {

when '../jobType = "LOGGED\_MDT\_ONLY"';

type enumeration {

enum OUT\_OF\_COVERAGE ;

enum A2\_EVENT ;

}

mandatory true;

description "Specifies event types for event triggered measurement in the

case of logged NR MDT. Each trace session may configure at most one

event. The UE shall perform logging of measurements only upon certain

condition being fulfilled:

- Out of coverage.

- A2 event.";

reference "Clause 5.10.28 of 3GPP TS 32.422";

}

leaf eventThreshold {

type int64;

description "Specifies the threshold which should trigger the reporting

in case A2 event reporting in LTE or 1F/1l event in UMTS. The attribute

is applicable only for Immediate MDT and when reportingTrigger is

configured for A2 event in LTE or 1F event or 1l event in UMTS. In

case this attribute is not used, it carries a null semantic.";

reference "Clauses 5.10.7 and 5.10.7a of 3GPP TS 32.422";

}

leaf listOfMeasurements {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"';

type int64;

mandatory true;

description "It specifies the UE measurements that shall be collected in

an Immediate MDT job. The attribute is applicable only for Immediate MDT.

In case this attribute is not used, it carries a null semantic.";

reference "Clause 5.10.3 of 3GPP TS 32.422";

}

leaf loggingDuration {

when '../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

type uint32 {

range "600|1200|2400|3600|5400|7200";

}

units seconds;

mandatory true;

description "Specifies how long the MDT configuration is valid at the

UE in case of Logged MDT. The attribute is applicable only for

Logged MDT and Logged MBSFN MDT. In case this attribute is not used, it

carries a null semantic.";

reference "Clause 5.10.9 of 3GPP TS 32.422";

}

leaf loggingInterval {

when '../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

type enumeration {

enum 320ms;

enum 640ms;

enum 1280ms;

enum 2560ms;

enum 5120ms;

enum 10240ms;

enum 20480ms;

enum 30720ms;

enum 40960ms;

enum 61440ms;

enum INFINITY;

}

mandatory true;

description "Specifies the periodicity for Logged MDT. The attribute is

applicable only for Logged MDT and Logged MBSFN MDT. In case this

attribute is not used, it carries a null semantic";

reference "Clause 5.10.8 of 3GPP TS 32.422";

}

leaf eventThresholdL1 {

when '../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

type uint32 {

range "0..127";

}

description "It specifies the threshold which should trigger

the reporting in case of event based reporting of logged NR MDT.

The attribute is applicable only for Logged MDT and when reportType

is configured for event triggered reporting and when

EventListForTriggeredMeasurement is configured for L1 event.

In case this attribute is not used, it carries a null semantic.";

reference "Clause 5.10.36 of TS 32.422";

}

leaf hysteresisl1 {

when '../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

type uint32 {

range "0..30";

}

description "It specifies the hysteresis used within the entry and leave

condition of the L1 event based reporting of logged NR MDT.

The attribute is applicable only for Logged MDT, when reportType

is configured for event triggered reporting and when

eventListForTriggeredMeasurement is configured for L1 event.

In case this attribute is not used, it carries a null semantic.";

reference "Clause 5.10.37 of TS 32.422";

}

leaf timeToTriggerL1 {

when '../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

type int32 {

range 0|40|64|80|100|128|160|256|320|480|512|640|1024|1280|2560|5120;

}

units milliseconds;

description "It specifies the threshold which should trigger

the reporting in case of event based reporting of logged NR MDT.

The attribute is applicable only for Logged MDT, when reportType

is configured for event triggered reporting and when

EventListForTriggeredMeasurement is configured for L1 event.

In case this attribute is not used, it carries a null semantic.";

reference "Clause 5.10.38 of TS 32.422";

}

grouping MbsfnAreaGrp {

description "Represents the MbsfnArea dataType.

This <<dataType>> defines a MBSFN area. It is composed of the MBSFN Area

identifier and the carrier frequency (EARFCN).";

leaf mbsfnAreaId {

type uint32 {

range 1..max;

}

mandatory true;

description "MBSFN Area Identifier";

}

leaf earfcn{

type uint32 {

range 1..max;

}

mandatory true;

description "Carrier Frequency";

}

}

list mbsfnAreaList {

when '../jobType = "LOGGED\_MBSFN\_MDT"';

key "mbsfnAreaId earfcn";

max-elements 8;

description "The MBSFN Area consists of a MBSFN Area ID and Carrier

Frequency (EARFCN). The target MBSFN area List can have up to 8 entries.

This parameter is applicable only if the job type is Logged MBSFN MDT.";

reference "Clause 5.10.25 of 3GPP TS 32.422";

uses MbsfnAreaGrp;

}

leaf measurementPeriodLTE {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type enumeration {

enum 1024ms;

enum 2048ms;

enum 5120ms;

enum 10240ms;

enum 1min;

}

mandatory true;

description "It specifies the measurement period for the Data Volume (M4) and

Scheduled IP throughput measurements (M5) for LTE MDT taken by the eNB.

The attribute is applicable only for Immediate MDT. In case this

attribute is not used, it carries a null semantic.";

reference "Clause 5.10.23 of 3GPP TS 32.422";

}

leaf measurementPeriodUMTS {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ 'or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type uint32 {

range "1000|2000|3000|4000|6000|8000|12000|16000|20000|"

+"24000|28000|32000|64000";

}

units milliseconds;

mandatory true;

description "It specifies the measurement period for the Data Volume (M6) and

Throughput measurements (M7) for UMTS MDT taken by RNC.

The attribute is applicable only for Immediate MDT. In case this

attribute is not used, it carries a null semantic.";

reference "Clause 5.10.22 of 3GPP TS 32.422";

}

leaf measurementQuantity {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type enumeration {

enum CPICH\_ECNO;

enum CPICH\_RSCP;

enum PATHLOSS;

}

description "It specifies the measurements that are collected in an MDT

job for a UMTS MDT configured for event triggered reporting.";

reference "Clause 5.10.15 of 3GPP TS 32.422";

}

leaf eventThresholdUphUMTS {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type uint16 {

range 0..31 ;

}

description "It specifies the threshold which should trigger

the reporting in case of event-triggered periodic reporting for M4

(UE power headroom measurement) in UMTS. In case this attribute is

not used, it carries a null semantic.";

reference "Clause 5.10.39 of 3GPP TS 32.422";

}

list plmnList {

when '../jobType = "LOGGED\_MDT\_ONLY"';

key "mcc mnc";

uses types3gpp:PLMNId;

max-elements 16;

description "It indicates the PLMNs where measurement collection, status

indication and log reporting is allowed.";

reference "Clause 5.10.24 of 3GPP TS 32.422";

}

leaf positioningMethod {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' or ../jobType = "IMMEDIATE\_MDT\_AND\_TRACE"';

type enumeration {

enum GNSS;

enum E-CELL\_ID;

}

mandatory true;

description "It specifies what positioning method should be used in the

MDT job.";

reference "Clause 5.10.19 of 3GPP TS 32.422";

}

leaf reportAmount {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' and ../reportingTrigger = "PERIODICAL"';

type enumeration {

enum 1ms;

enum 4ms;

enum 8ms;

enum 16ms;

enum 32ms;

enum 64ms;

enum INFINITY;

}

description "It specifies the number of measurement reports that shall be

taken for periodic reporting while the UE is in connected.

The attribute is applicable only for Immediate MDT and when

reportingTrigger is configured for periodical measurements. In

case this attribute is not used, it carries a null semantic.";

reference "Clause 5.10.6 of 3GPP TS 32.422";

}

leaf reportingTrigger {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"';

type enumeration {

enum PERIODICAL;

enum A2\_FOR\_LTE;

enum 1F\_FOR\_UMTS;

enum 1I\_FOR\_UMTS\_MCPS\_TDD;

enum A2\_TRIGGERED\_PERIODIC\_FOR\_LTE;

enum ALL\_CONFIGURED\_RRM\_FOR\_LTE;

enum ALL\_CONFIGURED\_RRM\_FOR\_UMTS;

}

description "It specifies whether periodic or event based measurements

should be collected.

The attribute is applicable only for Immediate MDT and when the

ListOfMeasurements is configured for M1 (for both UMTS and LTE)

or M2 (only for UMTS). In case this attribute is not used, it carries

a null semantic.";

reference "Clause 5.10.4 of 3GPP TS 32.422";

}

leaf reportInterval {

when '../jobType = "IMMEDIATE\_MDT\_ONLY"'

+ ' and ../reportingTrigger = "PERIODICAL"';

type uint32 {

range "120|240|250|480|500|640|1000|1024|2000|2048|3000|4000|"

+"5120|6000|8000|10240|12000|16000|20000|"

+"24000|28000|32000|60000|64000|"

+"360000|720000|1800000|3600000";

}

units milliseconds;

mandatory true;

description "It specifies the interval between the periodical measurements

that shall be taken when the UE is in connected mode.

The attribute is applicable only for Immediate MDT and when

reportingTrigger is configured for periodical measurements. In case

this attribute is not used, it carries a null semantic.";

reference "5.10.5 of 3GPP TS 32.422";

}

leaf reportType {

when '../jobType = "LOGGED\_MDT\_ONLY"';

type enumeration {

enum PERIODICAL;

enum EVENT\_TRIGGERED;

}

mandatory true;

description "It specifies report type for logged NR MDT";

reference "Clause 5.10.27 of 3GPP TS 32.422";

}

leaf sensorInformation {

type enumeration {

enum BAROMETRIC\_PRESSURE;

enum UE\_SPEED;

enum UE\_ORIENTATION;

}

description "It specifies which sensor information shall be included in

logged NR MDT and immediate NR MDT measurement if they are available.

The following sensor measurement can be included or excluded for

the UE.";

reference "Clause 5.10.29 of 3GPP TS 32.422";

}

leaf traceCollectionEntityID {

when '../jobType = "LOGGED\_MDT\_ONLY" or ../jobType = "LOGGED\_MBSFN\_MDT"';

type uint8;

mandatory true;

description "It specifies the TCE Id which is sent to the UE in

Logged MDT.";

reference "Clause 5.10.11 of 3GPP TS 32.422";

}

}

grouping AreaScopeGrp {

description "Represents the AreaScope dataType.

This <<dataType>> defines the area scope of MDT.

The Area Scope parameter in LTE and NR is either:

- list of Cells, identified by E-UTRAN-CGI or NG-RAN CGI.

Maximum 32 CGI can be defined.

- list of Tracking Area, identified by TAC.

Maximum of 8 TAC can be defined.

- list of Tracking Area Identity, identified by TAC with

associated plmn-Identity perTAC-List containing the

PLMN identity for each TAC. Maximum of 8 TAI can be defined.";

choice AreaScopeChoice {

leaf-list eutraCellIdList {

type string;

min-elements 1;

max-elements 32;

description "List of E-UTRAN cells identified by E-UTRAN-CGI";

}

leaf-list nrCellIdList {

type string;

min-elements 1;

max-elements 32;

description "List of NR cells identified by NG-RAN CGI";

}

leaf-list tacList {

type types3gpp:Tac;

min-elements 1;

max-elements 8;

description "Tracking Area Code list";

}

list taiList {

description "Tracking Area Identity list";

key idx;

min-elements 1;

max-elements 8;

leaf idx { type string; }

uses types3gpp:TaiGrp;

}

}

}

grouping TraceSubtree {

description "Contains classes that manage Tracing.

Should be used in all classes (or classes inheriting from)

- SubNetwork

- ManagedElement

- ManagedFunction

If a YANG module wants to augment these classes/list/groupings they must

augment all user classes!";

list TraceJob {

description "Represents the Trace Control and Configuration parameters of a

particular Trace Job (see TS 32.421 and TS 32.422 for details).

It can be name-contained by SubNetwork, ManagedElement, ManagedFunction

or NetworkSliceSubnet.

To activate Trace Jobs, a MnS consumer has to create TraceJob object

instances on the MnS producer. A MnS consumer can activate a Trace Job

for another MnS consumer since it is not required the value of

traceCollectionEntityAddress or streamingTraceConsumerUri to be his

own.

When a MnS consumer wishes to deactivate a Trace Job, the MnS consumer

shall delete the corresponding TraceJob instance.

For details of management Trace Job activation/deactivation see clause

4.1.1.1.2 of TS 32.422.

The attribute traceReference specifies a globally unique ID and

identifies a Trace session. One Trace Session may be activated to

multiple Network Elements. The traceReference is populated by the

consumer that makes the request for a Trace Session.

The attribute jobType specifies the kind of data to collect. Dependent

on the selected type various parameters shall be available. The

attributes jobType, traceReference,

traceCollectionEntityAddress and traceReportingFormat are mandatory

for all job types. If streaming reporting is selected for

traceReportingFormat, streamingTraceConsumerURI shall be present

additionally. The attribute PLMNTarget shall be present if trace

activation method is management based.

For the different job types the attributes are differentiated as follows:

- In case of TRACE\_ONLY additionally the following attributes shall be

available: listOfNeTypes, traceDepth, traceTarget and

triggeringEvents.

For this case the optional attribute listOfInterfaces allows to

specify the interfaces to be recorded.

- In case of IMMEDIATE\_MDT\_ONLY additionally the following attributes

shall be available:

- anonymizationOfMDTData,

- listOfMeasurements,

- collectionPeriodRRMUMTS (conditional for M4 and M5 in UMTS),

- measurementPeriodUMTS (conditional for M6 and M7 in UMTS),

- collectionPeriodRRMLTE (conditional for M3 in LTE),

- measurementPeriodLTE (conditional for M4 and M5 in LTE),

- collectionPeriodM6LTE (conditional for M6 in LTE),

- collectionPeriodM7LTE (conditional for M7 in LTE),

- collectionPeriodRRMNR (conditional for M4 and M5 in NR),

- collectionPeriodM6NR (conditional for M6 in NR),

- collectionPeriodM7NR (conditional for M7 in NR),

- beamLevelMeasurement (conditional for M1 in NR),

- reportInterval (conditional for M1 in LTE or NR and M1/M2 in UMTS),

- reportAmount (conditional for M1 in LTE or NR and M1/M2 in UMTS),

- reportingTrigger (conditional for M1 in LTE or NR and M1/M2 in UMTS),

- eventThreshold (conditional for A2 event reporting or A2 event triggered periodic reporting),

- measurementQuantity (conditional for 1F event reporting).

- excessPacketDelayThresholds (conditional for M6 UL measurement in NR).

For this case the optional attribute areaScope defines the area scope

of MDT, which is specified in clause 5.10.2 of TS 32.44. Additionally, the optional

attributes positioningMethod, sensorInformation allow to

specify the positioning methods to use or the sensor information to

include.

- In case of IMMEDIATE\_MDT\_AND\_TRACE both additional attributes of

TRACE\_ONLY and IMMEDIATE\_MDT\_ONLY shall apply.

- In case of LOGGED\_MDT\_ONLY additionally the following attributes

shall be available: traceTarget, anonymizationOfData,

traceCollectionEntityID, loggingInterval,

loggingDuration, reportType,

eventListForTriggeredMeasurements.

For this case the optional attribute areaScope defines the area

scope of MDT, which is specified in clause 5.10.2 of TS 32.422.

Additionaly, the optional attribute PLMNList allows to specify the PLMNs where measurement collection,

status indication and log reporting is allowed, the optional attribute

areaConfigurationForNeighCell allows to specify the area for

which UE is requested to perform measurements logging for neighbour

cells which have list of frequencies and the optional attribute

SensorInformation allows to specify the sensor information to

include.

- In case of RLF\_REPORT\_ONLY and RCEF\_REPORT\_ONLY additionally the

attribute traceTarget shall be available.

- In case of LOGGED\_MBSFN\_MDT additionally the following attributes

shall be available: anonymizationOfData, loggingInterval,

loggingDuration, msbnfAreaList.

Creation and deletion of TraceJob instances by MnS consumers is optional;

when not supported, the TraceJob instances may be created and deleted by

the system or be pre-installed.";

key id;

uses top3gpp:Top\_Grp ;

container attributes {

uses TraceJobGrp ;

}

}

}

}

<CODE ENDS>

# D.3 Void

# D.4 Mount information

If the class ManagedElement and the underlying hierarchy is contained under a SubNetwork all YANG modules containing IOCs that can be contained under the ManagedElement directly or under other IOCs contained by the ManagedElement and the YANG module for ManagedElement itself shall be mounted at the mountpoint "children-of-SubNetwork" in the YANG module \_3gpp-common-subnetwork.

See IETF RFC 8528 [16] that describes the mechanism that adds the schema trees defined by a set of YANG modules onto a mount point defined in the schema tree in another YANG module.

Annex E (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Change history | | | | | | | |
| Date | TSG # | TSG Doc. | CR | Rev | Subject/Comment | Old | New |
| 2012-12 |  |  |  |  | New version after approval | 2.0.0 | 11.0.0 |
| 2013-06 | SA#60 | SP-130304 | 002 | 2 | Correction of XML schema | 11.0.0 | 11.1.0 |
| 2014-06 | SA#64 | SP-140332 | 003 | 1 | upgrade XSD | 11.1.0 | 11.2.0 |
| SP-140358 | 004 | - | remove the feature support statements |
| 2014-09 | SA#65 | SP-140560 | 005 | - | Update the link from Solution Set to Information Service due to the end of Release 12 | 11.2.0 | 12.0.0 |
| 2015-12 | SA#70 | SP-150691 | 006 | 1 | Add missing id attribute | 12.0.0 | 12.1.0 |
| 2016-01 |  |  |  |  | Upgrade to Rel-13 (MCC) | 12.1.0 | 13.0.0 |
| 2016-03 | SA#71 | SP-160031 | 010 | 1 | Make the XML schema well formed | 13.0.0 | 13.1.0 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2016-06 | SA#72 | SP-160407 | 0011 | - | F | Update the link from IRP Solution Set to IRP Information Service | 13.2.0 |
| 2017-03 | SA#75 | - | - | - |  | Promotion to Release 14 without technical change | 14.0.0 |
| 2017-06 | SA#76 | SP-170510 | 0015 | 2 | B | Modifications to align with IS to support Configuration Management for mobile networks that include virtualized network functions | 14.1.0 |
| 2018-03 | SA#79 | SP-180060 | 0016 | 1 | B | Add attribute peeParametersList to Solution Set definitions | 15.0.0 |
| 2018-12 | SA#82 | SP-181042 | 0018 | 1 | F | Update NRM root IOCs Solution Set to support priority | 15.1.0 |
| 2019-03 | SA#83 | SP-190121 | 0020 | 1 | F | Update Generic NRM Solution Set to support JSON | 15.2.0 |
| 2019-06 | SA#84 | SP-190371 | 0021 | - | B | Add IOCs for threshold monitoring control | 16.0.0 |
| 2019-09 | SA#85 | SP-190745 | 0026 | 1 | F | generate JSON definition for generic NRM based on new style guideline | 16.1.0 |
| 2019-09 | SA#85 | SP-190744 | 0027 | - | A | Add IDL XML YANG solutions | 16.1.0 |
| 2019-09 | SA#85 | SP-190751 | 0029 | - | A | Correct references and remove not need abbreviations | 16.1.0 |
| 2019-12 | SA#86 | SP-191166 | 0031 | 1 | F | Correct XML solution set for generic NRM | 16.2.0 |
| 2019-12 | SA#86 | SP-191166 | 0035 | - | B | Updates to YANG SS | 16.2.0 |
| 2019-12 | SA#86 | SP-191173 | 0037 | 1 | A | Add the definition of attribute measurementsList | 16.2.0 |
| 2019-12 | SA#86 | SP-191166 | 0039 | - | B | Add heartbeat control NRM fragment - Stage 3 | 16.2.0 |
| 2019-12 | SA#86 | SP-191166 | 0040 | - | B | Add notification subscription control NRM fragment - Stage 3 | 16.2.0 |
| 2020-03 | SA#87E | SP-200163 | 0041 | 2 | B | Add configurable KPI control NRM | 16.3.0 |
| 2020-03 | SA#87E | SP-200163 | 0042 | - | B | Add configurable FM - YANG Solution | 16.3.0 |
| 2020-03 | SA#87E | SP-200230 | 0043 | 1 | F | Add OpenAPI definitions required by the ProvMnS | 16.3.0 |
| 2020-03 | SA#87E | SP-200169 | 0045 |  | F | Correct errors in yang solution set | 16.3.0 |
| 2020-03 | SA#87E |  |  |  |  | Correction in the implementation of CR0041 | 16.3.1 |
| 2020-03 | SA#87E |  |  |  |  | Correction of implementation | 16.3.2 |
| 2020-07 | SA#88E | SP-200490 | 0046 | 2 | B | Add OpenAPI definitions for the FM control fragment | 16.4.0 |
| 2020-07 | SA#88E | SP-200489 | 0047 | - | F | Correct OpenAPI definition for notificationTypes | 16.4.0 |
| 2020-07 | SA#88E | SP-200483 | 0079 | 2 | B | Add trace control NRM fragment stage 3 | 16.4.0 |
| 2020-07 | SA#88E | SP-200484 | 0080 | - | D | Fix inconsistent formatting | 16.4.0 |
| 2020-07 | SA#88E | SP-200493 | 0081 | - | B | Stage3 add the NRM fragment for SON management | 16.4.0 |
| 2020-07 | SA#88E | SP-200485 | 0082 | - | F | Update the definition of SNssai | 16.4.0 |
| 2020-07 | SA#88E | SP-200490 | 0084 | - | F | Update ManagedElement YANG moduel | 16.4.0 |
| 2020-07 | SA#88E | SP-200596 | 0085 | 1 | F | Update Nrm YANG | 16.4.0 |
| 2020-07 | SA#88E | SP-200490 | 0087 | 2 | F | Update PM control fragment (OpenAPI definitions) | 16.4.0 |
| 2020-07 | SA#88E | SP-200490 | 0088 | - | F | Clarify usage of the VsDataContainer (OpenAPI definitions) | 16.4.0 |
| 2020-07 | SA#88E | SP-200490 | 0089 | - | F | Add common data definitions (OpenAPI definitions) | 16.4.0 |
| 2020-07 | SA#88E | SP-200490 | 0091 | - | F | Update FM control fragment (YANG definitions) | 16.4.0 |
| 2020-07 | SA#88E | SP-200490 | 0092 | - | F | Update PM Control fragment (YANG definitions) | 16.4.0 |
| 2020-07 | SA#88E | SP-200490 | 0093 | 1 | F | Correct genericNRM definition in XML solution | 16.4.0 |
| 2020-09 | SA#89e | SP-200729 | 0095 | - | F | Correction of YANG errors | 16.5.0 |
| 2020-09 | SA#89e | SP-200727 | 0101 | 1 | A | Clean-up definitions and references | 16.5.0 |
| 2020-09 | SA#89e | SP-200729 | 0102 | - | B | YANG SS for Trace Control | 16.5.0 |
| 2020-09 | SA#89e | SP-200724 | 0103 | - | F | Add missing definitions to comDefs.yaml (OpenAPI definitions) | 16.5.0 |
| 2020-09 | SA#89e | SP-200724 | 0104 | - | F | Correct various smaller errors (e.g. validation errors) in genericNRM.yaml (OpenAPI definitions) | 16.5.0 |
| 2020-09 | SA#89e | SP-200729 | 0105 | 1 | F | Correct ThresholdMonitor definition (OpenAPI definitions) | 16.5.0 |
| 2020-09 | SA#89e | SP-200729 | 0106 | - | F | Update HeartbeatControl YANG definition | 16.5.0 |
| 2020-09 | SA#89e | SP-200729 | 0107 | - | F | Update ThresholdMonitor YANG definition | 16.5.0 |
| 2020-12 | SA#90e | SP-201057 | 0108 | - | F | Correction of NRM YANG errors | 16.6.0 |
| 2020-12 | SA#90e | SP-201063 | 0109 | 1 | F | Add new MDT specific parameter collection period for NR aligning with 28.622 for stage 3 | 16.6.0 |
| 2020-12 | SA#90e | SP-201057 | 0110 | - | F | Remove thresholdLevel attribute from ThresholdMonitor (OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0111 | 1 | F | Correct and add types in comDefs.yaml (OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0112 | 1 | F | Use comDefs.yaml instead of local definitions in genericNrm.yaml (OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201057 | 0113 | 1 | F | Update attribute perfMetricJobGroupId. | 16.6.0 |
| 2020-12 | SA#90e | SP-201057 | 0114 | - | F | Remove value handling from the granularityPeriod description | 16.6.0 |
| 2020-12 | SA#90e | SP-201088 | 0115 | - | F | Correct and add types in comDefs.yaml (OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201063 | 0117 |  | F | Correct trace target parameter for trace control in stage 3 | 16.6.0 |
| 2020-12 | SA#90e | SP-201089 | 0118 | 1 | F | Remove incorrect S-NSSAI definition from YANG SS | 16.6.0 |
| 2021-03 | SA#91e | SP-210146 | 0121 | - | F | Fix compilation errors | 16.7.0 |
| 2021-03 | SA#91e | SP-210153 | 0125 | - | F | YANG compilation error and missing stage 2 corrections | 16.7.0 |
| 2021-06 | SA#92e | SP-210406 | 0119 | 2 | F | Replace legacy IRPAgent with MnsAgent (OpenAPI definition) | 16.8.0 |
| 2021-06 | SA#92e | SP-210397 | 0127 | 1 | F | Correction of Trace/MDT related parameters (OpenAPI definition) | 16.8.0 |
| 2021-06 | SA#92e | SP-210397 | 0128 | 1 | F | Align Trace/MDT related parameters to TS 32.422 (OpenAPI definition) | 16.8.0 |
| 2021-06 | SA#92e | SP-210406 | 0129 | 1 | F | Clean up regarding common data types (OpenAPI definition) | 16.8.0 |
| 2021-06 | SA#92e | SP-210411 | 0130 | - | F | Correct definition of additionalInformation (YANG) | 16.8.0 |
| 2021-09 | SA#93e | SP-210886 | 0131 | 1 | F | Replace local data type definition for notificationFilter by common filter definition | 16.9.0 |
| 2021-09 | SA#93e | SP-210886 | 0132 | 1 | F | Correct data type of notificationId (YANG definitions) | 16.9.0 |
| 2021-09 | SA#93e | SP-210886 | 0133 | 1 | F | Clarify resource id is required and nullable (OpenAPI definitions) | 16.9.0 |
| 2021-09 | SA#93e | SP-210865 | 0134 | - | F | Correction and clarification of reporting in TraceJob (stage3) | 16.9.0 |
| 2021-09 | SA#93e | SP-210865 | 0135 | - | F | Adaptation and cleanup of Trace/MDT related parameters (stage3) | 16.9.0 |
| 2021-09 | SA#93e | SP-210871 | 0136 | - | F | YANG updates to correct YANG merging problems | 16.9.0 |
| 2021-09 | SA#93e | SP-210867 | 0137 | 1 | F | Correction of YANG Solution set | 16.9.0 |
| 2021-12 | SA#94e | SP-211475 | 0139 | 1 | F | Correction of YANG Solution set | 16.10.0 |
| 2021-12 | SA#94e | SP-211458 | 0142 | - | F | Introduce missing IEs for HSS and UDM Trace Record | 16.10.0 |
| 2022-06 | SA#96 | SP-220498 | 0158 | - | F | Stage 3 Yang fix for 3GPP Common Trace | 16.11.0 |
| 2022-06 | SA#96 | SP-220498 | 0160 | 1 | F | OpenAPI file name and dependence change for comDefs.yaml | 16.11.0 |
| 2022-06 | SA#96 | SP-220498 | 0161 | 1 | F | OpenAPI file name and dependence change for genericNrm.yaml | 16.11.0 |
| 2022-06 | SA#96 | SP-220498 | 0165 | 1 | F | yaml indentation correction for comDefs.yaml | 16.11.0 |
| 2022-06 | SA#96 | SP-220516 | 0167 | - | F | Alignment of attribute names of TraceJob IOC to TS 32.422 (stage 3) | 16.11.0 |
| 2022-06 | SA#96 | SP-220516 | 0173 | - | F | Alignment of attribute values of attribute tjMDTReportInterval to TS 32.422, TS 38.413 and TS 38.423 | 16.11.0 |
| 2022-06 | SA#96 |  |  |  |  | Correction in the implementation in OPENAPI annexes | 16.11.1 |
| 2022-06 | SA#96 |  |  |  |  | Correction in the implementation in OPENAPI annex C4.3 | 16.11.2 |
| 2022-09 | SA#97e | SP-220853 | 0179 | - | F | YANG Corrections | 16.12.0 |
| 2022-09 | SA#97e | SP-220859 | 0181 | 1 | F | Adding missing interface for SMF | 16.12.0 |
| 2022-09 | SA#97e | SP-220864 | 0187 | - | F | Correction of attribute names according to Upper Camel Case Convention and WKA | 16.12.0 |
| 2022-09 | SA#97e |  |  |  |  | Alignment of FORGE with Annex C.4.3 | 16.12.1 |
| 2022-12 | SA#98e | SP-221173 | 0199 | 1 | A | Adding YANG begin and End markers | 16.13.0 |
| 2022-12 | SA#98e | SP-221170 | 0216 | - | F | Add missing attribute properties to YANG | 16.13.0 |
| 2023-03 | SA#99 | SP-230199 | 0221 | 1 | F | Fix IpAddr stage 3 definition | 16.14.0 |
| 2023-03 | SA#99 | SP-230200 | 0228 | - | A | Missing Mount information | 16.14.0 |
| 2023-03 | SA#99 | SP-230210 | 0231 | 1 | F | Correcting traceRecordingSessionReference property (stage3) | 16.14.0 |
| 2023-03 | SA#99 | SP-230210 | 0235 |  | F | Correct Trace IOC attribute names (stage3, yang) | 16.14.0 |
| 2023-03 | SA#99 | SP-230208 | 0238 | 1 | F | Clarify reporting and monitoring period usage in SupportedPerfMetricGroup datatype (stage3) | 16.14.0 |
| 2023-03 | SA#99 |  |  |  |  | Corrections in Annex D to align with FORGE | 16.14.1 |
| 2023-09 | SA#101 | SP-230942 | 0258 | 1 | F | Rel-16 CR 28.623 Clarify HeartbeatControl IOC definition (stage3, yang) | 16.15.0 |
| 2023-12 | SA#102 | SP-231488 | 0292 | 2 | F | Rel-16 CR TS 28.623 Align N38 in SMF with TS23.501 | 16.16.0 |
| 2023-12 | SA#102 | SP-231488 | 0300 | - | F | Rel-16 CR TS28.623 Adding N16 and N16a into module\_3gpp-common-trace.yang | 16.16.0 |
| 2024-09 | SA#105 | SP-241170 | 0395 | - | F | Rel-16 CR 28.623 Cleanup of TraceJob | 16.17.0 |
| 2024-09 | SA#105 | SP-241170 | 0399 | 1 | F | Rel-16 CR 28.623 Correction of TraceJob attributes MBSFN Area List and Area Configuration For Neighboring Cells (stage 3) | 16.17.0 |
| 2024-12 | SA#106 | SP-241636 | 0446 | 1 | F | Rel-16 CR 28.623 Correction for "NR Measurent Type" for "GenericNrm" | 16.18.0 |
| 2024-12 | SA#106 | SP-241636 | 0468 | 1 | F | Rel-16 CR TS 28.623 Correction to AreaScope (stage 3, YANG) | 16.18.0 |
| 2024-12 | SA#106 | SP-241646 | 0482 | 1 | A | Rel-16 CR TS 28.623 Add information for IRP based solutions | 16.18.0 |
| 2025-03 | SA#107 | SP-250154 | 0501 | 1 | F | Rel-16 CR TS 28.623 Correct Trace-MDT (YANG) | 16.19.0 |
| 2025-06 | SA#108 | SP-250555 | 0533 | 1 | F | Rel-16 CR TS 28.623 Corrections on MDT PLMN List | 16.20.0 |