**3GPP TSG- Meeting # *rev1***

**, , -**

|  |
| --- |
| *CR-Form-v12.1* |
| **CHANGE REQUEST** |
|  |
|  |  | **CR** |  | **rev** |  | **Current version:** |  |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network | **X** | Core Network |  |

|  |
| --- |
|  |
| ***Title:***  | Rel-17 CR 28.541 Update RANSliceSubnetProfile attributes |
|  |  |
| ***Source to WG:*** |  |
| ***Source to TSG:*** | S5 |
|  |  |
| ***Work item code:*** |  |  | ***Date:*** |  |
|  |  |  |  |  |
| ***Category:*** |  |  | ***Release:*** |  |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-15 (Release 15)Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)* |
|  |  |
| ***Reason for change:*** | dLThptPerSliceSubnet and uLThptPerSliceSubnet attributes are included in TopSliceSubnetProfile and CNSliceSubnetProfile, but missed in RANSliceSubnetProfile.According to clause 6.3.25, TopSliceSubnetProfile includes an aggregated list of the attributes from RANSliceSubnetProfile and CNSliceSubnetProfile. To comply with this definition, it is therefore needed to add dLThptPerSliceSubnet and uLThptPerSliceSubnet attributes to RANSliceSubnetProfile.  |
|  |  |
| ***Summary of change:*** | * Add dLThptPerSliceSubnet and uLThptPerSliceSubnet attributes to RANSliceSubnetProfile.
* Mirror these changes in stage 3.
 |
|  |  |
| ***Consequences if not approved:*** | May lead to incorrect implementation |
|  |  |
| ***Clauses affected:*** | 6.3.24.2, J.4.3, N.2.5 |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  | **X** |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  | **X** |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  | **X** |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** |  |

|  |
| --- |
| **1st change** |

### 6.3.24 RANSliceSubnetProfile<<dataType>>

#### 6.3.24.1 Definition

This data type represents the requirements for RAN slice profile.

Editor's NOTE 1: Whether the attributes of RANSliceSubnetProfile need to be modelled by one IOC or more than one IOC is FFS.

Editor's NOTE 2: Whether RANSliceSubnetProfile is an IOC or dataType is FFS.

#### 6.3.24.2 Attributes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute name | S | isReadable | isWritable | isInvariant | isNotifyable |
|  |  |  |  |  |  |
| coverageAreaTAList | O | T | T | F | T |
| uEMobilityLevel | O | T | T | F | T |
| resourceSharingLevel | O | T | T | F | T |
| maxNumberofUEs | O | T | T | F | T |
| activityFactor | O | T | T | F | T |
| dLThptPerSliceSubnet | O | T | T | F | T |
| dLThptPerUE | O | T | T | F | T |
| uLThptPerSliceSubnet | O | T | T | F | T |
| uLThptPerUE | O | T | T | F | T |
| uESpeed | O | T | T | F | T |
| reliability | O | T | T | F | T |
| nROperatingBands | O | T | T | F | T |
| serviceType | O | T | T | F | T |
| dLLatency | O | T | T | F | T |
| uLLatency | O | T | T | F | T |
| delayTolerance | O | T | T | F | T |
| sliceSimultaneousUse | O | T | T | F | T |
| dLMaxPktSize | O | T | T | F | T |
| uLMaxPktSize | O | T | T | F | T |
| energyEfficiency | O | T | T | F | T |
| termDensity | O | T | T | F | T |
| survivalTime | O | T | T | F | T |
| dLDeterministicComm | O | T | T | F | T |
| uLDeterministicComm | O | T | T | F | T |
| positioning | O | T | T | F | T |
| synchronicity | O | T | T | F | T |

#### 6.3.24.3 Attribute constraints

None.

#### 6.3.24.4 Notifications

The subclause 6.5 of the <<IOC>> using this <<dataType>> as one of its attributes, shall be applicable.

|  |
| --- |
| **2nd change** |

Annex J (normative):
OpenAPI definition of the Slice NRM

# J.1 General

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

The Information Service (IS) of the NR NRM is defined in clause 6.

Mapping rules to produce the OpenAPI definition based on the IS are defined in 3GPP TS 32.160 [47].

# J.2 Void

# J.3 Void

# J.4 Solution Set (SS) definitions

## J.4.1 Void

## J.4.2 Void

## J.4.3 OpenAPI document "sliceNrm.yaml"

openapi: 3.0.1

info:

 title: Slice NRM

 version: 17.4.0

 description: >-

 OAS 3.0.1 specification of the Slice NRM

 @ 2020, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).

 All rights reserved.

externalDocs:

 description: 3GPP TS 28.541; 5G NRM, Slice NRM

 url: http://www.3gpp.org/ftp/Specs/archive/28\_series/28.541/

paths: {}

components:

 schemas:

#------------ Type definitions ---------------------------------------------------

 Float:

 type: number

 format: float

 MobilityLevel:

 type: string

 enum:

 - STATIONARY

 - NOMADIC

 - RESTRICTED MOBILITY

 - FULLY MOBILITY

 SynAvailability:

 type: string

 enum:

 - NOT SUPPORTED

 - BETWEEN BS AND UE

 - BETWEEN BS AND UE & UE AND UE

 PositioningAvailability:

 type: array

 items:

 type: string

 enum:

 - CIDE-CID

 - OTDOA

 - RF FINGERPRINTING

 - AECID

 - HYBRID POSITIONING

 - NET-RTK

 Predictionfrequency:

 type: string

 enum:

 - PERSEC

 - PERMIN

 - PERHOUR

 SharingLevel:

 type: string

 enum:

 - SHARED

 - NON-SHARED

 NetworkSliceSharingIndicator:

 type: string

 enum:

 - SHARED

 - NON-SHARED

 ServiceType:

 type: string

 enum:

 - eMBB

 - RLLC

 - MIoT

 - V2X

 SliceSimultaneousUse:

 type: string

 enum:

 - ZERO

 - ONE

 - TWO

 - THREE

 - FOUR

 Category:

 type: string

 enum:

 - CHARACTER

 - SCALABILITY

 Tagging:

 type: array

 items:

 type: string

 enum:

 - PERFORMANCE

 - FUNCTION

 - OPERATION

 Exposure:

 type: string

 enum:

 - API

 - KPI

 ServAttrCom:

 type: object

 properties:

 category:

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

 tagging:

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

 exposure:

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

 Support:

 type: string

 enum:

 - NOT SUPPORTED

 - SUPPORTED

 DelayTolerance:

 type: object

 properties:

 servAttrCom:

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

 support:

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

 DeterministicComm:

 type: object

 properties:

 servAttrCom:

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

 availability:

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

 periodicityList:

 type: string

 XLThpt:

 type: object

 properties:

 servAttrCom:

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

 guaThpt:

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

 maxThpt:

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

 MaxPktSize:

 type: object

 properties:

 servAttrCom:

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

 maxsize:

 type: integer

 MaxNumberofPDUSessions:

 type: object

 properties:

 servAttrCom:

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

 nOofPDUSessions:

 type: integer

 KPIMonitoring:

 type: object

 properties:

 servAttrCom:

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

 kPIList:

 type: string

 NBIoT:

 type: object

 properties:

 servAttrCom:

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

 support:

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

 RadioSpectrum:

 type: object

 properties:

 servAttrCom:

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

 nROperatingBands:

 type: string

 Synchronicity:

 type: object

 properties:

 servAttrCom:

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

 availability:

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

 accuracy:

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

 SynchronicityRANSubnet:

 type: object

 properties:

 availability:

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

 accuracy:

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

 Positioning:

 type: object

 properties:

 servAttrCom:

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

 availability:

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

 predictionfrequency:

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

 accuracy:

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

 PositioningRANSubnet:

 type: object

 properties:

 availability:

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

 predictionfrequency:

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

 accuracy:

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

 UserMgmtOpen:

 type: object

 properties:

 servAttrCom:

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

 support:

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

 V2XCommModels:

 type: object

 properties:

 servAttrCom:

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

 v2XMode:

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

 TermDensity:

 type: object

 properties:

 servAttrCom:

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

 density:

 type: integer

 NsInfo:

 type: object

 properties:

 nsInstanceId:

 type: string

 nsName:

 type: string

 EmbbEEPerfReq:

 type: integer

 UrllcEEPerfReq:

 type: integer

 MIoTEEPerfReq:

 type: object

 properties:

 KpiType:

 type: string

 enum:

 - MAXREGSUBS

 - MEANACTIVEUES

 Req:

 type: integer

 EEPerfReq:

 oneOf:

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

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

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

 EnergyEfficiency:

 type: object

 properties:

 servAttrCom:

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

 performance:

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

 NSSAASupport:

 type: object

 properties:

 servAttrCom:

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

 support:

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

SecFunc:

 type: object

 properties:

 secFunId:

 type: string

 secFunType:

 type: string

 secRules:

 type: array

 items:

 type: string

N6Protection:

 type: object

 properties:

 servAttrCom:

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

 secFuncList:

 type: array

 items:

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

 CNSliceSubnetProfile:

 type: object

 properties:

 maxNumberofUEs:

 type: integer

 dLLatency:

 type: integer

 uLLatency:

 type: integer

 dLThptPerSliceSubnet:

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

 dLThptPerUE:

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

 uLThptPerSliceSubnet:

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

 uLThptPerUE:

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

 maxNumberOfPDUSessions:

 type: integer

 coverageAreaTAList:

 type: integer

 resourceSharingLevel:

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

 dLMaxPktSize:

 type: integer

 uLMaxPktSize:

 type: integer

 delayTolerance:

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

 synchronicity:

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

 sliceSimultaneousUse:

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

 reliability:

 type: string

 energyEfficiency:

 type: integer

 dLDeterministicComm:

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

 uLDeterministicComm:

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

 survivalTime:

 type: string

 nssaaSupport:

 $ref: '#/components/schemas/NSSAASupport’

 n6Protection:

 $ref: '#/components/schemas/N6Protection'

 RANSliceSubnetProfile:

 type: object

 properties:

 coverageAreaTAList:

 type: integer

 dLLatency:

 type: integer

 uLLatency:

 type: integer

 uEMobilityLevel:

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

 resourceSharingLevel:

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

 maxNumberofUEs:

 type: integer

 activityFactor:

 type: integer

 dLThptPerSliceSubnet:

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

 dLThptPerUE:

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

 uLThptPerSliceSubnet:

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

 uLThptPerUE:

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

 uESpeed:

 type: integer

 reliability:

 type: string

 serviceType:

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

 dLMaxPktSize:

 type: integer

 uLMaxPktSize:

 type: integer

 nROperatingBands:

 type: string

 delayTolerance:

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

 positioning:

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

 sliceSimultaneousUse:

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

 energyEfficiency:

 type: integer

 termDensity:

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

 survivalTime:

 type: string

 synchronicity:

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

 dLDeterministicComm:

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

 uLDeterministicComm:

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

 TopSliceSubnetProfile:

 type: object

 properties:

 dLLatency:

 type: integer

 uLLatency:

 type: integer

 maxNumberofUEs:

 type: integer

 dLThptPerSliceSubnet:

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

 dLThptPerUE:

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

 uLThptPerSliceSubnet:

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

 uLThptPerUE:

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

 dLMaxPktSize:

 type: integer

 uLMaxPktSize:

 type: integer

 maxNumberOfPDUSessions:

 type: integer

 nROperatingBands:

 type: string

 sliceSimultaneousUse:

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

 energyEfficiency:

 type: integer

 synchronicity:

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

 delayTolerance:

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

 positioning:

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

 termDensity:

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

 activityFactor:

 type: integer

 coverageAreaTAList:

 type: integer

 resourceSharingLevel:

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

 uEMobilityLevel:

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

 uESpeed:

 type: integer

 reliability:

 type: string

 serviceType:

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

 dLDeterministicComm:

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

 uLDeterministicComm:

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

 survivalTime:

 type: string

 ServiceProfile:

 type: object

 properties:

 serviceProfileId:

 type: string

 plmnInfoList:

 $ref: 'nrNrm.yaml#/components/schemas/PlmnInfoList'

 maxNumberofUEs:

 type: number

 dLLatency:

 type: number

 uLLatency:

 type: number

 uEMobilityLevel:

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

 sst:

 $ref: 'nrNrm.yaml#/components/schemas/Sst'

 networkSliceSharingIndicator:

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

 availability:

 type: number

 delayTolerance:

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

 dLDeterministicComm:

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

 uLDeterministicComm:

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

 dLThptPerSlice:

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

 dLThptPerUE:

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

 uLThptPerSlice:

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

 uLThptPerUE:

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

 dLMaxPktSize:

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

 uLMaxPktSize:

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

 maxNumberofPDUSessions:

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

 kPIMonitoring:

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

 nBIoT:

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

 radioSpectrum:

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

 synchronicity:

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

 positioning:

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

 userMgmtOpen:

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

 v2XModels:

 $ref: '#/components/schemas/V2XCommModels'

 coverageArea:

 type: string

 termDensity:

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

 activityFactor:

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

 uESpeed:

 type: integer

 jitter:

 type: integer

 survivalTime:

 type: string

 reliability:

 type: string

 maxDLDataVolume:

 type: string

 maxULDataVolume:

 type: string

 sliceSimultaneousUse:

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

 energyEfficiency:

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

 nssaaSupport:

 $ref: '#/components/schemas/NSSAASupport’

 n6Protection:

 $ref: '#/components/schemas/N6Protection'

 SliceProfile:

 type: object

 properties:

 serviceProfileId:

 type: string

 plmnInfoList:

 $ref: 'nrNrm.yaml#/components/schemas/PlmnInfoList'

 cNSliceSubnetProfile:

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

 rANSliceSubnetProfile:

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

 topSliceSubnetProfile:

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

 IpAddress:

 oneOf:

 - $ref: 'genericNrm.yaml#/components/schemas/Ipv4Addr'

 - $ref: 'genericNrm.yaml#/components/schemas/Ipv6Addr'

 LogicInterfaceInfo:

 type: object

 properties:

 logicalInterfceType:

 type: string

 enum:

 - VLAN

 - MPLS

 - Segment

 logicalInterfceId:

 type: string

 ServiceProfileList:

 type: array

 items:

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

 SliceProfileList:

 type: array

 items:

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

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

 SubNetwork-Single:

 allOf:

 - $ref: 'genericNrm.yaml#/components/schemas/Top'

 - type: object

 properties:

 attributes:

 allOf:

 - $ref: 'genericNrm.yaml#/components/schemas/SubNetwork-Attr'

 - $ref: 'genericNrm.yaml#/components/schemas/SubNetwork-ncO'

 - type: object

 properties:

 SubNetwork:

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

 NetworkSlice:

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

 NetworkSliceSubnet:

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

 EP\_Transport:

 $ref: '#/components/schemas/EP\_Transport-Multiple'

 NetworkSlice-Single:

 allOf:

 - $ref: 'genericNrm.yaml#/components/schemas/Top'

 - type: object

 properties:

 attributes:

 allOf:

 - type: object

 properties:

 networkSliceSubnetRef:

 $ref: 'genericNrm.yaml#/components/schemas/Dn'

 operationalState:

 $ref: 'genericNrm.yaml#/components/schemas/OperationalState'

 administrativeState:

 $ref: 'genericNrm.yaml#/components/schemas/AdministrativeState'

 serviceProfileList:

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

 NetworkSliceSubnet-Single:

 allOf:

 - $ref: 'genericNrm.yaml#/components/schemas/Top'

 - type: object

 properties:

 attributes:

 allOf:

 - type: object

 properties:

 managedFunctionRefList:

 $ref: 'genericNrm.yaml#/components/schemas/DnList'

 networkSliceSubnetRefList:

 $ref: 'genericNrm.yaml#/components/schemas/DnList'

 operationalState:

 $ref: 'genericNrm.yaml#/components/schemas/OperationalState'

 administrativeState:

 $ref: 'genericNrm.yaml#/components/schemas/AdministrativeState'

 nsInfo:

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

 sliceProfileList:

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

 epTransportRefList:

 $ref: 'genericNrm.yaml#/components/schemas/DnList'

 priorityLabel:

 type: integer

 networkSliceSubnetType:

 type: string

 enum:

 - TopSliceSubnet

 - RANSliceSubnet

 - CNSliceSubnet

 EP\_Transport-Single:

 allOf:

 - $ref: 'genericNrm.yaml#/components/schemas/Top'

 - type: object

 properties:

 attributes:

 type: object

 properties:

 ipAddress:

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

 logicInterfaceInfo:

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

 nextHopInfo:

 type: string

 qosProfile:

 type: string

 epApplicationRefs:

 $ref: 'genericNrm.yaml#/components/schemas/DnList'

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

 SubNetwork-Multiple:

 type: array

 items:

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

 NetworkSlice-Multiple:

 type: array

 items:

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

 NetworkSliceSubnet-Multiple:

 type: array

 items:

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

 EP\_Transport-Multiple:

 type: array

 items:

 $ref: '#/components/schemas/EP\_Transport-Single'

#------------ Definitions in TS 28.541 for TS 28.532 -----------------------------

 resources-sliceNrm:

 oneOf:

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

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

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

 - $ref: '#/components/schemas/EP\_Transport-Single'

|  |
| --- |
| **3rd change** |

Annex N (normative):
YANG definition of the Slice NRM

# N.1 General

This annex contains the YANG definitions for the Slice NRM in YANG format.

The Information Service (IS) of the Slice NRM is defined in clause 6.

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

# N.2 Modules

## N.2.1 module \_3gpp-ns-nrm-networkslice.yang

<CODE BEGINS>

module \_3gpp-ns-nrm-networkslice {

 yang-version 1.1;

 namespace urn:3gpp:sa5:\_3gpp-ns-nrm-networkslice;

 prefix ns3gpp;

 import \_3gpp-ns-nrm-networkslicesubnet { prefix nss3gpp; }

 import \_3gpp-common-subnetwork { prefix subnet3gpp; }

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

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

 include \_3gpp-ns-nrm-serviceprofile;

 organization "3GPP SA5";

 contact

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

 description "A network slice instance in a 5G network.";

 reference "3GPP TS 28.541

 Management and orchestration;

 5G Network Resource Model (NRM);

 Information model definitions for network slice NRM (chapter 6)

 ";

 revision 2020-06-02 {

 reference "CR-0485, CR-0508";

 }

 revision 2020-02-19 {

 description "Introduction of YANG definitions for network slice NRM";

 reference "CR-0458";

 }

 grouping NetworkSliceGrp {

 uses subnet3gpp:SubNetworkGrp; // Inherits from SubNetwork

 leaf operationalState {

 description "The operational state of the network slice instance.

 It describes whether or not the resource is physically installed

 and working.";

 config false;

 type types3gpp:OperationalState;

 }

 leaf administrativeState {

 description "The administrative state of the network slice instance.

 It describes the permission to use or prohibition against

 using the instance, imposed through the OAM services.";

 type types3gpp:AdministrativeState;

 }

 list serviceProfileList {

 description "A list of service profiles supported by the network

 slice instance.";

 key serviceProfileId;

 uses ServiceProfileGrp;

 }

 leaf networkSliceSubnetRef {

 type leafref {

 path /nss3gpp:NetworkSliceSubnet/nss3gpp:id;

 }

 description "The NetworkSliceSubnet that the NetworkSlice is

 associated with.";

 }

 }

 list NetworkSlice {

 description "Represents the properties of a network slice instance in

 a 5G network.";

 key id;

 container attributes {

 uses NetworkSliceGrp;

 }

 uses top3gpp:Top\_Grp;

 }

}

<CODE ENDS>

## N.2.2 module \_3gpp-ns-nrm-networkslicesubnet.yang

<CODE BEGINS>

module \_3gpp-ns-nrm-networkslicesubnet {

 yang-version 1.1;

 namespace urn:3gpp:sa5:\_3gpp-ns-nrm-networkslicesubnet;

 prefix nss3gpp;

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

 import \_3gpp-common-subnetwork { prefix subnet3gpp; }

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

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

 // import \_3gpp-ns-nrm-common { prefix ns3cmn; }

 include \_3gpp-ns-nrm-sliceprofile;

 organization "3GPP SA5";

 contact

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

 description "This IOC represents the properties of a network slice subnet

 instance in a 5G network.";

 reference "3GPP TS 28.541

 Management and orchestration;

 5G Network Resource Model (NRM);

 Information model definitions for network slice NRM (chapter 6)

 ";

 revision 2021-05-05 {

 description "replace perfReq with 3 new datatypes xxxSliceSubnetProfile";

 reference "CR-0485";

 }

 revision 2020-02-19 {

 description "Introduction of YANG definitions for network slice NRM";

 reference "CR-0458";

 }

 revision 2019-06-07 {

 description "initial revision";

 reference "Based on

 3GPP TS 28.541 V15.X.XX";

 }

 feature MeasurementsUnderNetworkSliceSubnet {

 description "The MeasurementSubtree shall be contained under

 NetworkSliceSubnet.";

 }

 typedef ETSI-GS-NFV-Identifier {

 type string;

 reference "ETSI GS NFV-IFA 013";

 }

 grouping EPTransportGrp {

 leaf ipAddress {

 description "This parameter specifies the IP address assigned to a

 logical transport interface/endpoint. It can be an IPv4 address

 (See RFC 791) or an IPv6 address (See RFC 2373).";

 mandatory true;

 type string;

 }

 leaf logicInterfaceId {

 description "This parameter specifies the identify of a logical

 transport interface. It could be VLAN ID (See IEEE 802.1Q),

 MPLS Tag or Segment ID.";

 mandatory true;

 type string;

 }

 leaf-list nextHopInfo {

 description "This parameter is used to identify ingress transport

 node. Each node can be identified by any of combination of IP

 address of next-hop router of transport network, system name,

 port name, IP management address of transport nodes.";

 type string;

 }

 leaf-list qosProfile {

 description "This parameter specifies reference to QoS Profile for

 a logical transport interface. A QoS profile includes a set of

 parameters which are locally provisioned on both sides of a logical

 transport interface.";

 type string;

 }

 leaf-list epApplicationRef {

 description "This parameter specifies a list of application level

 EPs associated with the logical transport interface.";

 min-elements 1;

 type types3gpp:DistinguishedName;

 }

 uses top3gpp:Top\_Grp;

 }

 grouping NsInfoGrp {

 description "The NsInfo of the NS instance corresponding to the network

 slice subnet instance.";

 //suport condition: It shall be supported if the NSS instance is

 //realized in the virtualized environment.

 // Otherwise this attribute shall be absent.

 reference "ETSI GS NFV-IFA 013 clause 8.3.3.2.2, which can be found at

 https://www.etsi.org/deliver/etsi\_gs/NFV-IFA/001\_099/013

 /03.04.01\_60/gs\_NFV-IFA013v030401p.pdf page 123-124";

 leaf nSInstanceId {

 description "Uniquely identifies the NS instance.";

 config false;

 type ETSI-GS-NFV-Identifier;

 }

 leaf nsName {

 description "Human readable name of the NS instance.";

 type string;

 config false;

 }

 leaf description {

 description "Human readable description of the NS instance.";

 config false;

 type string;

 }

 }

 grouping NetworkSliceSubnetGrp {

 uses subnet3gpp:SubNetworkGrp;

 uses EPTransportGrp;

 leaf operationalState {

 description "The operational state of the network slice instance.

 It describes whether or not the resource is physically installed

 and working.";

 mandatory true;

 config false;

 type types3gpp:OperationalState;

 }

 leaf administrativeState {

 description "The administrative state of the network slice instance.

 It describes the permission to use or prohibition against

 using the instance, imposed through the OAM services.";

 mandatory true;

 type types3gpp:AdministrativeState;

 }

 list nsInfo {

 description "This list represents the properties of network service

 information corresponding to the network slice subnet instance.";

 reference "ETSI GS NFV-IFA 013 clause 8.3.3.2.2";

 config false;

 key nSInstanceId;

 max-elements 1;

 uses NsInfoGrp;

 }

 list sliceProfileList {

 description "List of SliceProfiles supported by the network slice

 subnet instance. All members of the list, instances of SliceProfile,

 shall contain the same datatype representing slice profile requirements:

 TopSliceSubnetProfile, RANSliceSubnetProfile or CNSliceSubnetProfile.

 Members of the list may contain TopSliceSubnetProfile datatype

 only when this attribute (sliceProfileList) belongs to

 a NetworkSliceSubnet that is directly referenced by a NetworkSlice";

 key sliceProfileId;

 uses SliceProfileGrp;

 }

 list managedFunctionRef {

 description "The managed functions that the NetworkSliceSubnet is

 associated with.";

 key aggregatedManagedFunction;

 leaf aggregatedManagedFunction {

 type instance-identifier;

 }

 }

 leaf-list networkSliceSubnetRef {

 type leafref {

 path /NetworkSliceSubnet/id;

 }

 description "Lists the NetworkSliceSubnet instances associated with

 this NetworkSliceSubnet.";

 }

 }

 list NetworkSliceSubnet {

 description "Represents the properties of a network slice subnet

 instance in a 5G network.";

 key id;

 container attributes {

 uses NetworkSliceSubnetGrp;

 leaf-list parents {

 description "Reference to direct parent NetworkSliceSubnet

 instances.

 If NetworkSliceSubnets form a containment hierarchy this is

 modeled using references between the child NetworkSliceSubnet

 and the parent NetworkSliceSubnet.

 This reference MUST NOT be present for the top level

 NetworkSliceSubnet and MUST be present for other

 NetworkSliceSubnets.";

 type leafref {

 path "/NetworkSliceSubnet/id";

 }

 }

 leaf-list containedChildren {

 description "Reference to all directly contained NetworkSliceSubnet

 instances. If NetworkSliceSubnets form a containment hierarchy

 this is modeled using references between the child

 NetworkSliceSubnet and the parent NetworkSliceSubnet.";

 type leafref {

 path "/NetworkSliceSubnet/id";

 }

 }

 }

 uses top3gpp:Top\_Grp;

 uses meas3gpp:MeasurementSubtree {

 if-feature MeasurementsUnderNetworkSliceSubnet;

 }

 }

}

<CODE ENDS>

## N.2.3 Void

## N.2.4 module \_3gpp-ns-nrm-serviceprofile.yang

<CODE BEGINS>

submodule \_3gpp-ns-nrm-serviceprofile {

 yang-version 1.1;

 belongs-to \_3gpp-ns-nrm-networkslice { prefix ns3gpp; }

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

 import \_3gpp-5g-common-yang-types { prefix types5g3gpp; }

 import \_3gpp-ns-nrm-common { prefix ns3cmn; }

 organization "3GPP SA5";

 contact

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

 description "A network slice instance in a 5G network.";

 reference "3GPP TS 28.541

 Management and orchestration;

 5G Network Resource Model (NRM);

 Information model definitions for network slice NRM (chapter 6)

 ";

 revision 2020-06-02 {

 reference "CR-0485, CR-0508";

 }

 revision 2020-02-19 {

 description "Introduction of YANG definitions for network slice NRM";

 reference "CR-0458";

 }

 revision 2019-06-23 {

 description "Initial revision";

 reference "3GPP TS 28.541 V15.X.XX";

 }

 typedef availability-percentage {

 description "

 Percentage value of the amount of time the end-to-end communication

 service is delivered according to an agreed QoS, divided by the amount

 of time the system is expected to deliver the end-to-end service

 according to the specification in a specific area.";

 reference "3GPP TS 22.261 3.1";

 type decimal64 {

 fraction-digits 4; // E.g. 99.9999

 range 0..100;

 }

 }

 typedef V2XMode-enum {

 type enumeration {

 enum NOT\_SUPPORTED;

 enum SUPPORTED\_BY\_NR;

 }

 }

 grouping ServiceProfileGrp {

 leaf serviceProfileId {

 description "Service profile identifier.";

 type types3gpp:DistinguishedName;

 }

 list sNSSAIList {

 description "The S-NSSAI list to be supported by the new NSI to be

 created or the existing NSI to be re-used.";

 min-elements 1;

 key idx;

 unique "sst sd";

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses types5g3gpp:SNssai;

 }

 list pLMNIdList {

 description "List of PLMN IDs.";

 min-elements 1;

 key "mcc mnc";

 ordered-by user;

 uses types3gpp:PLMNId;

 }

 leaf maxNumberofUEs {

 description "The maximum number of UEs that may simultaneously

 access the network slice instance.";

 mandatory true;

 type uint64;

 }

 leaf-list coverageArea {

 min-elements 1;

 description "A list of TrackingAreas where the NSI can be selected.";

 type types3gpp:Tac;

 }

 leaf latency {

 description "The packet transmission latency (milliseconds) through

 the RAN, CN, and TN part of 5G network, used to evaluate utilization

 performance of the end-to-end network slice instance.";

 reference "3GPP TS 28.554 clause 6.3.1";

 mandatory true;

 type uint16;

 units milliseconds;

 }

 leaf uEMobilityLevel {

 description "The mobility level of UE accessing the network slice

 instance.";

 reference "3GPP TS 22.261 clause 6.2.1";

 type types3gpp:UeMobilityLevel;

 }

 leaf resourceSharingLevel {

 description "Specifies whether the resources to be allocated to the

 network slice instance may be shared with another network slice

 instance(s).";

 type types3gpp:ResourceSharingLevel;

 }

 //Stage2 issue: The sNSSAIList above specifies one or potentially

 // several sST objects for the service profile.

 // How do they relate?

 leaf sST {

 description "Specifies the slice/service type. See 3GPP TS 23.501

 for defined values.";

 mandatory true;

 type uint32;

 reference "3GPP TS 23.501 5.15.2.2";

 }

 leaf availability {

 description "The availability requirement for a network slice

 instance, expressed as a percentage.";

 type availability-percentage;

 }

 list delayTolerance {

 description "An attribute specifies the properties of service delivery

 flexibility, especially for the vertical services that are not

 chasing a high system performance.";

 reference "TS 22.104 clause 4.3";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf support {

 description "An attribute specifies whether or not the network

 slice supports service delivery flexibility, especially for the

 vertical services that are not chasing a high system performance.";

 type ns3cmn:Support-enum;

 }

 }

 list deterministicComm {

 //Stage2 issue: deterministicComm is not defined in 28.541 chapter 6,

 // but I guess deterministicComm is meant

 description "This list represents the properties of the deterministic

 communication for periodic user traffic. Periodic traffic refers to the

 type of traffic with periodic transmissions.";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf availability {

 //Stage2 issue: Defined differently in 28.541 chapter 6, but XML

 // uses DeterministicCommAvailability

 config false;

 type ns3cmn:DeterminCommAvailability;

 }

 leaf periodicityList {

 //Stage2 issue: Not defined in 28.541 chapter 6. XML and YAML

 // says "string".

 type string;

 }

 }

 list dLThptPerSlice {

 description "This attribute defines achievable data rate of the

 network slice in downlink that is available ubiquitously across

 the coverage area of the slice";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:XLThptGrp;

 }

 list dLThptPerUE {

 description "This attribute defines data rate supported by the network

 slice per UE";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:XLThptGrp;

 }

 list uLThptPerSlice {

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 description "This attribute defines achievable data rate of the

 network slice in uplink that is available ubiquitously across

 the coverage area of the slice";

 uses ns3cmn:XLThptGrp;

 }

 list uLThptPerUE {

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 description "This attribute defines data rate supported by the

 network slice per UE";

 uses ns3cmn:XLThptGrp;

 }

 list maxPktSize {

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 description "This parameter specifies the maximum packet size

 supported by the network slice";

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf maxSize {

 //Stage2 issue: Not defined in 28.541, guessing integer bytes

 type uint32;

 units bytes;

 }

 }

 list maxNumberofPDUSessions {

 description "Represents the maximum number of

 concurrent PDU sessions supported by the network slice";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf nOofPDUSessions {

 //Stage2 issue: Not defined in 28.541, guessing integer

 type uint32;

 }

 }

 list kPIMonitoring {

 description "Represents performance monitoring";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf kPIList {

 //Stage2 issue: Data format not specified, low interoperability

 description "An attribute specifies the name list of KQIs and KPIs

 available for performance monitoring";

 type string;

 }

 }

 list userMgmtOpen {

 description "An attribute specifies whether or not the network slice

 supports the capability for the NSC to manage their users or groups

 of users' network services and corresponding requirements.";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf support {

 type ns3cmn:Support-enum;

 }

 }

 list v2XCommModels {

 description "An attribute specifies whether or not the V2X

 communication mode is supported by the network slice.";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf v2XMode {

 type V2XMode-enum;

 }

 }

 list termDensity {

 description "An attribute specifies the overall user density over

 the coverage area of the network slice";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf density {

 type uint32;

 units users/km2;

 }

 }

 leaf activityFactor {

 //Stage2 issue: This is modeled as writable/config true in 28.542,

 // but that does not appear to match the description

 description "An attribute specifies the percentage value of the

 amount of simultaneous active UEs to the total number of UEs where

 active means the UEs are exchanging data with the network";

 reference "TS 22.261 Table 7.1-1";

 type decimal64 {

 fraction-digits 1;

 }

 }

 leaf uESpeed {

 //Stage2 issue: This is modeled as writable/config true in 28.542,

 // but that does not appear to match the description

 description "An attribute specifies the maximum speed (in km/hour)

 supported by the network slice at which a defined QoS can be

 achieved";

 type uint32;

 units km/h;

 }

 leaf jitter {

 //Stage2 issue: This is modeled as writable/config true in 28.542,

 // but that does not appear to match the description

 description "An attribute specifies the deviation from the desired

 value to the actual value when assessing time parameters";

 reference "TS 22.104 clause C.4.1";

 type uint32;

 units microseconds;

 }

 leaf survivalTime {

 description "An attribute specifies the time that an application

 consuming a communication service may continue without an

 anticipated message.";

 reference "TS 22.104 clause 5";

 type string;

 }

 leaf reliability {

 description "An attribute specifies in the context of network layer

 packet transmissions, percentage value of the amount of sent

 network layer packets successfully delivered to a given system

 entity within the time constraint required by the targeted service,

 divided by the total number of sent network layer packets.";

 reference "TS 22.261, TS 22.104";

 type string;

 }

 leaf maxDLDataVolume {

 //Stage2 issue: Not defined in 28.541. XML and YAML says "string"

 type string;

 }

 leaf maxULDataVolume {

 //Stage2 issue: Not defined in 28.541. XML and YAML says "string"

 type string;

 }

 list nBIoT {

 description "An attribute specifies whether NB-IoT is supported in

 the RAN in the network slice";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn:ServAttrComGrp;

 }

 leaf support {

 description "An attribute specifies whether NB-IoT is supported

 in the RAN in the network slice";

 type ns3cmn:Support-enum;

 }

 }

 }

}

<CODE ENDS>

## N.2.5 module \_3gpp-ns-nrm-sliceprofile.yang

<CODE BEGINS>

submodule \_3gpp-ns-nrm-sliceprofile {

 yang-version 1.1;

 belongs-to \_3gpp-ns-nrm-networkslicesubnet { prefix nss3gpp; }

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

 import \_3gpp-5g-common-yang-types { prefix types5g3gpp; }

 // import \_3gpp-ns-nrm-networkslice { prefix ns3gpp; }

 import \_3gpp-ns-nrm-common { prefix ns3cmn3gpp; }

 organization "3GPP SA5";

 contact

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

 description "Represents the properties of network slice subnet related

 requirement that should be supported by the network slice subnet

 instance in a 5G network.";

 reference "3GPP TS 28.541

 Management and orchestration;

 5G Network Resource Model (NRM);

 Information model definitions for network slice NRM (chapter 6)

 ";

 revision 2021-07-16 { reference CR-0566 ; }

 revision 2021-05-05 {

 description "replace perfReq with 3 new datatypes xxxSliceSubnetProfile";

 reference "CR-0485";

 }

 revision 2020-02-19 {

 description "Introduction of YANG definitions for network slice NRM";

 reference "CR-0458";

 }

 revision 2019-05-27 {

 description "initial revision.";

 reference "Based on

 3GPP TS 28.541 V15.X.XX";

 }

 typedef SliceSimultaneousUse-enum {

 type enumeration {

 enum ZERO;

 enum ONE;

 enum TWO;

 enum THREE;

 enum FOUR;

 }

 }

 typedef ServiceType-enum {

 type enumeration {

 enum eMBB;

 enum URLLC;

 enum MIoT;

 enum V2X;

 }

 }

 grouping PositioningGrp {

 description "Represents positioning support.";

 reference "Clause 3.4.20 of GSMA NG.116 ";

 uses ns3cmn3gpp:ServAttrComGrp ;

 leaf-list availability {

 type enumeration {

 enum CIDE\_CID ;

 enum OTDOA;

 enum RF\_FINGERPRINTING;

 enum AECID;

 enum HYBRID\_POSITIONING;

 enum NET\_RTK;

 }

 min-elements 1;

 config false;

 description "Specifies if this attribute is provided by the RAN domain

 of the network slice and contains a list of positioning methods

 provided by the RAN domain. If the list is empty this attribute is

 not available in the RAN domain and the other parameters might be

 ignored, see NG.116. Values allowed: are

 CIDE-CID (LTE and NR), OTDOA (LTE and NR), RF fingerprinting, AECID,

 Hybrid positioning, NET-RTK.";

 }

 leaf predictionfrequency {

 type enumeration {

 enum PERSEC;

 enum PERMIN;

 enum PERHOUR;

 }

 mandatory true;

 description "Specifies how often location information is provided.

 This parameter simply defines how often the customer is allowed to

 request location information. This is not related to the time it

 takes to determine the location, which is a characteristic of the

 positioning method.

 If leaf-list availability is empty, the value has no meaning.";

 reference "NG.116";

 }

 leaf accuracy {

 type decimal64 {

 fraction-digits 2;

 }

 units meter;

 mandatory true;

 description "Specifies the accuracy of the location information.

 Accuracy depends on the respective positioning solution applied in the

 RAN domain of the network slice.";

 reference "NG.116";

 }

 }

 grouping TopSliceSubnetProfileGrp {

 leaf-list coverageArea {

 min-elements 1;

 description "A list of TrackingAreas where the NSI can be selected.";

 type types3gpp:Tac;

 }

 leaf latency {

 description "The packet transmission latency (milliseconds) through

 the RAN, CN, and TN part of 5G network, used to evaluate

 utilization performance of the end-to-end network slice instance.";

 reference "3GPP TS 28.554 clause 6.3.1";

 //optional support

 mandatory true;

 type uint16;

 units milliseconds;

 }

 leaf maxNumberofUEs {

 description "Specifies the maximum number of UEs may simultaneously

 access the network slice instance.";

 //optional support

 mandatory true;

 type uint64;

 }

 list dLThptPerSliceSubnet {

 description "This attribute defines achievable data rate of the

 network slice subnet in downlink that is available ubiquitously

 across the coverage area of the slice";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list dLThptPerUE {

 description "This attribute defines data rate supported by the

 network slice per UE, refer NG.116.";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list uLThptPerSliceSubnet {

 description "This attribute defines achievable data rate of the

 network slice subnet in uplink that is available ubiquitously

 across the coverage area of the slice";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list uLThptPerUE {

 description "This attribute defines data rate supported by the

 network slice per UE, refer NG.116";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list maxPktSize {

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 description "This parameter specifies the maximum packet size

 supported by the network slice";

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf maxSize {

 //Stage2 issue: Not defined in 28.541, guessing integer bytes

 type uint32;

 units bytes;

 }

 }

 list maxNumberofPDUSessions {

 description "Represents the maximum number of

 concurrent PDU sessions supported by the network slice";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf nOofPDUSessions {

 //Stage2 issue: Not defined in 28.541, guessing integer

 type uint32;

 }

 }

 leaf sliceSimultaneousUse {

 description "This attribute describes whether a network slice

 can be simultaneously used by a device together with other

 network slices and if so, with which other classes of network slices.";

 type SliceSimultaneousUse-enum;

 }

 list delayTolerance {

 description "An attribute specifies the properties of service delivery

 flexibility, especially for the vertical services that are not

 chasing a high system performance.";

 reference "TS 22.104 clause 4.3";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf support {

 description "An attribute specifies whether or not the network

 slice supports service delivery flexibility, especially for the

 vertical services that are not chasing a high system performance.";

 type ns3cmn3gpp:Support-enum;

 }

 }

 list termDensity {

 description "An attribute specifies the overall user density over

 the coverage area of the network slice";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf density {

 type uint32;

 units users/km2;

 }

 }

 leaf activityFactor {

 //Stage2 issue: This is modeled as writable/config true in 28.542,

 // but that does not appear to match the description

 description "An attribute specifies the percentage value of the

 amount of simultaneous active UEs to the total number of UEs where

 active means the UEs are exchanging data with the network";

 reference "TS 22.261 Table 7.1-1";

 type decimal64 {

 fraction-digits 1;

 }

 }

 leaf-list coverageAreaTAList {

 description "A list of TrackingAreas where the NSI can be selected.";

 //optional support

 min-elements 1;

 type types3gpp:Tac;

 }

 leaf uEMobilityLevel {

 description "The mobility level of UE accessing the network slice

 instance.";

 //optional support

 type types3gpp:UeMobilityLevel;

 }

 leaf resourceSharingLevel {

 description "Specifies whether the resources to be allocated to the

 network slice subnet instance may be shared with another network

 slice subnet instance(s).";

 //optional support

 type types3gpp:ResourceSharingLevel;

 }

 leaf uESpeed {

 //Stage2 issue: This is modeled as writable/config true in 28.542,

 // but that does not appear to match the description

 description "An attribute specifies the maximum speed (in km/hour)

 supported by the network slice at which a defined QoS can be

 achieved";

 type uint32;

 units km/h;

 }

 leaf reliability {

 description "An attribute specifies in the context of network layer

 packet transmissions, percentage value of the amount of sent

 network layer packets successfully delivered to a given system

 entity within the time constraint required by the targeted service,

 divided by the total number of sent network layer packets.";

 reference "TS 22.261, TS 22.104";

 type string;

 }

 leaf serviceType {

 description "An attribute specifies the standardized network slice type.

 allowedValues: eMBB, URLLC, MIoT, V2X.";

 type ServiceType-enum;

 }

 list deterministicComm {

 //Stage2 issue: deterministicComm is not defined in 28.541 chapter 6,

 // but I guess determinComm is meant

 description "This list represents the properties of the deterministic

 communication for periodic user traffic. Periodic traffic refers to the

 type of traffic with periodic transmissions.";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf availability {

 //Stage2 issue: Defined differently in 28.541 chapter 6, but XML

 // uses DeterminCommAvailability

 config false;

 type ns3cmn3gpp:DeterminCommAvailability;

 }

 leaf periodicityList {

 //Stage2 issue: Not defined in 28.541 chapter 6. XML and YAML

 // says "string".

 type string;

 }

 }

 leaf survivalTime {

 description "An attribute specifies the time that an application

 consuming a communication service may continue without an

 anticipated message.";

 reference "TS 22.104 clause 5";

 type string;

 }

 list positioning {

 key predictionfrequency;

 min-elements 1;

 max-elements 1;

 description "Specifies whether the network slice provides

 geo-localization methods or supporting methods";

 reference "Clause 3.4.20 of NG.116";

 uses PositioningGrp;

 }

 }

 grouping CNSliceSubnetProfileGrp {

 leaf-list coverageArea {

 min-elements 1;

 description "A list of TrackingAreas where the NSI can be selected.";

 type types3gpp:Tac;

 }

 leaf latency {

 description "The packet transmission latency (milliseconds) through

 the RAN, CN, and TN part of 5G network, used to evaluate

 utilization performance of the end-to-end network slice instance.";

 reference "3GPP TS 28.554 clause 6.3.1";

 //optional support

 mandatory true;

 type uint16;

 units milliseconds;

 }

 leaf maxNumberofUEs {

 description "Specifies the maximum number of UEs may simultaneously

 access the network slice instance.";

 //optional support

 mandatory true;

 type uint64;

 }

 list dLThptPerSliceSubnet {

 description "This attribute defines achievable data rate of the

 network slice subnet in downlink that is available ubiquitously

 across the coverage area of the slice";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list dLThptPerUE {

 description "This attribute defines data rate supported by the

 network slice per UE, refer NG.116.";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list uLThptPerSliceSubnet {

 description "This attribute defines achievable data rate of the

 network slice subnet in uplink that is available ubiquitously

 across the coverage area of the slice";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list uLThptPerUE {

 description "This attribute defines data rate supported by the

 network slice per UE, refer NG.116";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list maxPktSize {

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 description "This parameter specifies the maximum packet size

 supported by the network slice";

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf maxSize {

 //Stage2 issue: Not defined in 28.541, guessing integer bytes

 type uint32;

 units bytes;

 }

 }

 list maxNumberofPDUSessions {

 description "Represents the maximum number of

 concurrent PDU sessions supported by the network slice";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf nOofPDUSessions {

 //Stage2 issue: Not defined in 28.541, guessing integer

 type uint32;

 }

 }

 leaf sliceSimultaneousUse {

 description "This attribute describes whether a network slice

 can be simultaneously used by a device together with other

 network slices and if so, with which other classes of network slices.";

 type SliceSimultaneousUse-enum;

 }

 list delayTolerance {

 description "An attribute specifies the properties of service delivery

 flexibility, especially for the vertical services that are not

 chasing a high system performance.";

 reference "TS 22.104 clause 4.3";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf support {

 description "An attribute specifies whether or not the network

 slice supports service delivery flexibility, especially for the

 vertical services that are not chasing a high system performance.";

 type ns3cmn3gpp:Support-enum;

 }

 }

 leaf-list coverageAreaTAList {

 description "A list of TrackingAreas where the NSI can be selected.";

 //optional support

 min-elements 1;

 type types3gpp:Tac;

 }

 leaf resourceSharingLevel {

 description "Specifies whether the resources to be allocated to the

 network slice subnet instance may be shared with another network

 slice subnet instance(s).";

 //optional support

 type types3gpp:ResourceSharingLevel;

 }

 list deterministicComm {

 //Stage2 issue: deterministicComm is not defined in 28.541 chapter 6,

 // but I guess determinComm is meant

 description "This list represents the properties of the deterministic

 communication for periodic user traffic. Periodic traffic refers to the

 type of traffic with periodic transmissions.";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf availability {

 //Stage2 issue: Defined differently in 28.541 chapter 6, but XML

 // uses DeterminCommAvailability

 config false;

 type ns3cmn3gpp:DeterminCommAvailability;

 }

 leaf periodicityList {

 //Stage2 issue: Not defined in 28.541 chapter 6. XML and YAML

 // says "string".

 type string;

 }

 }

 }

 grouping PositioningRANSubnetGrp {

 description "Represents positioning support in RAN domain";

 leaf-list availability {

 type enumeration {

 enum CIDE\_CID ;

 enum OTDOA;

 enum RF\_FINGERPRINTING;

 enum AECID;

 enum HYBRID\_POSITIONING;

 enum NET\_RTK;

 }

 config false;

 description "Specifies if this attribute is provided by the RAN domain

 of the network slice and contains a list of positioning methods

 provided by the RAN domain. If the list is empty this attribute is

 not available in the RAN domain and the other parameters might be

 ignored, see NG.116. Values allowed: are

 CIDE-CID (LTE and NR), OTDOA (LTE and NR), RF fingerprinting, AECID,

 Hybrid positioning, NET-RTK.";

 }

 leaf predictionfrequency {

 type enumeration {

 enum PERSEC;

 enum PERMIN;

 enum PERHOUR;

 }

 mandatory true;

 description "Specifies how often location information is provided.

 This parameter simply defines how often the customer is allowed to

 request location information. This is not related to the time it

 takes to determine the location, which is a characteristic of the

 positioning method.

 If leaf-list availability is empty, the value has no meaning.";

 reference "NG.116";

 }

 leaf accuracy {

 type decimal64 {

 fraction-digits 2;

 }

 units meter;

 mandatory true;

 description "Specifies the accuracy of the location information.

 Accuracy depends on the respective positioning solution applied in the

 RAN domain of the network slice.";

 reference "NG.116";

 }

 }

 grouping RANSliceSubnetProfileGrp {

 description "Represents the RANSliceSubnetProfile datatype";

 leaf latency {

 description "The packet transmission latency (milliseconds) through

 the RAN, CN, and TN part of 5G network, used to evaluate

 utilization performance of the end-to-end network slice instance.";

 reference "3GPP TS 28.554 clause 6.3.1";

 //optional support

 mandatory true;

 type uint16;

 units milliseconds;

 }

 leaf maxNumberofUEs {

 description "Specifies the maximum number of UEs may simultaneously

 access the network slice instance.";

 //optional support

 mandatory true;

 type uint64;

 }

 list dLThptPerSliceSubnet {

 description "This attribute defines achievable data rate of the

 network slice subnet in downlink that is available ubiquitously

 across the coverage area of the slice";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list dLThptPerUE {

 description "This attribute defines data rate supported by the

 network slice per UE, refer NG.116.";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list uLThptPerSliceSubnet {

 description "This attribute defines achievable data rate of the

 network slice subnet in uplink that is available ubiquitously

 across the coverage area of the slice";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list uLThptPerUE {

 description "This attribute defines data rate supported by the

 network slice per UE, refer NG.116";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:XLThptGrp;

 }

 list maxPktSize {

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 description "This parameter specifies the maximum packet size

 supported by the network slice";

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf maxSize {

 //Stage2 issue: Not defined in 28.541, guessing integer bytes

 type uint32;

 units bytes;

 }

 }

 list delayTolerance {

 description "An attribute specifies the properties of service delivery

 flexibility, especially for the vertical services that are not

 chasing a high system performance.";

 reference "TS 22.104 clause 4.3";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf support {

 description "An attribute specifies whether or not the network

 slice supports service delivery flexibility, especially for the

 vertical services that are not chasing a high system performance.";

 type ns3cmn3gpp:Support-enum;

 }

 }

 leaf sliceSimultaneousUse {

 description "This attribute describes whether a network slice

 can be simultaneously used by a device together with other

 network slices and if so, with which other classes of network slices.";

 type SliceSimultaneousUse-enum;

 }

 list termDensity {

 description "An attribute specifies the overall user density over

 the coverage area of the network slice";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf density {

 type uint32;

 units users/km2;

 }

 }

 leaf activityFactor {

 //Stage2 issue: This is modeled as writable/config true in 28.542,

 // but that does not appear to match the description

 description "An attribute specifies the percentage value of the

 amount of simultaneous active UEs to the total number of UEs where

 active means the UEs are exchanging data with the network";

 reference "TS 22.261 Table 7.1-1";

 type decimal64 {

 fraction-digits 1;

 }

 }

 leaf-list coverageAreaTAList {

 description "A list of TrackingAreas where the NSI can be selected.";

 //optional support

 min-elements 1;

 type types3gpp:Tac;

 }

 leaf uEMobilityLevel {

 description "The mobility level of UE accessing the network slice

 instance.";

 //optional support

 type types3gpp:UeMobilityLevel;

 }

 leaf resourceSharingLevel {

 description "Specifies whether the resources to be allocated to the

 network slice subnet instance may be shared with another network

 slice subnet instance(s).";

 //optional support

 type types3gpp:ResourceSharingLevel;

 }

 leaf uESpeed {

 //Stage2 issue: This is modeled as writable/config true in 28.542,

 // but that does not appear to match the description

 description "An attribute specifies the maximum speed (in km/hour)

 supported by the network slice at which a defined QoS can be

 achieved";

 type uint32;

 units km/h;

 }

 leaf reliability {

 description "An attribute specifies in the context of network layer

 packet transmissions, percentage value of the amount of sent

 network layer packets successfully delivered to a given system

 entity within the time constraint required by the targeted service,

 divided by the total number of sent network layer packets.";

 reference "TS 22.261, TS 22.104";

 type string;

 }

 leaf serviceType {

 description "An attribute specifies the standardized network slice type.

 allowedValues: eMBB, URLLC, MIoT, V2X.";

 type ServiceType-enum;

 }

 list deterministicComm {

 //Stage2 issue: deterministicComm is not defined in 28.541 chapter 6,

 // but I guess determinComm is meant

 description "This list represents the properties of the deterministic

 communication for periodic user traffic. Periodic traffic refers to the

 type of traffic with periodic transmissions.";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ns3cmn3gpp:ServAttrComGrp;

 }

 leaf availability {

 //Stage2 issue: Defined differently in 28.541 chapter 6, but XML

 // uses DeterminCommAvailability

 config false;

 type ns3cmn3gpp:DeterminCommAvailability;

 }

 leaf periodicityList {

 //Stage2 issue: Not defined in 28.541 chapter 6. XML and YAML

 // says "string".

 type string;

 }

 }

 leaf survivalTime {

 description "An attribute specifies the time that an application

 consuming a communication service may continue without an

 anticipated message.";

 reference "TS 22.104 clause 5";

 type string;

 }

 list positioning {

 min-elements 1;

 max-elements 1;

 description "Specifies whether the RAN domain of the network slice

 provides geo-localization methods or supporting methods.";

 reference "Clause 3.4.20 of NG.116 [50].";

 uses PositioningRANSubnetGrp;

 }

 }

 grouping SliceProfileGrp {

 leaf sliceProfileId {

 description "A unique identifier of the property of network slice

 subnet related requirement should be supported by the network

 slice subnet instance.";

 type types3gpp:DistinguishedName;

 }

 list sNSSAIList {

 description "List of S-NSSAIs the managed object is capable of

 supporting. (Single Network Slice Selection Assistance Information)

 An S-NSSAI has an SST (Slice/Service type) and an optional SD

 (Slice Differentiator) field.";

 key idx;

 unique "sst sd";

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses types5g3gpp:SNssai;

 }

 list pLMNIdList {

 description "List of at most six entries of PLMN Identifiers, but at

 least one (the primary PLMN Id). The PLMN Identifier is composed

 of a Mobile Country Code (MCC) and a Mobile Network Code (MNC).";

 min-elements 1;

 max-elements 6;

 key "mcc mnc";

 ordered-by user;

 uses types3gpp:PLMNId;

 }

 leaf maxNumberofUEs {

 description "Specifies the maximum number of UEs may simultaneously

 access the network slice instance.";

 //optional support

 mandatory true;

 type uint64;

 }

 leaf-list coverageAreaTAList {

 description "A list of TrackingAreas where the NSI can be selected.";

 //optional support

 min-elements 1;

 type types3gpp:Tac;

 }

 leaf latency {

 description "The packet transmission latency (milliseconds) through

 the RAN, CN, and TN part of 5G network, used to evaluate

 utilization performance of the end-to-end network slice instance.";

 reference "3GPP TS 28.554 clause 6.3.1";

 //optional support

 mandatory true;

 type uint16;

 units milliseconds;

 }

 leaf uEMobilityLevel {

 description "The mobility level of UE accessing the network slice

 instance.";

 //optional support

 type types3gpp:UeMobilityLevel;

 }

 leaf resourceSharingLevel {

 description "Specifies whether the resources to be allocated to the

 network slice subnet instance may be shared with another network

 slice subnet instance(s).";

 //optional support

 type types3gpp:ResourceSharingLevel;

 }

 list CNSliceSubnetProfile {

 description " This represents the requirements for the top slice associated with the

 network slice. ";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses TopSliceSubnetProfileGrp;

 }

 list RANSliceSubnetProfile {

 description " This represents the requirements for the top slice associated with the

 network slice. ";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses TopSliceSubnetProfileGrp;

 }

 list TopSliceSubnetProfile {

 description " This represents the requirements for the top slice associated with the

 network slice. ";

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses TopSliceSubnetProfileGrp;

 }

 }

}

<CODE ENDS>

## N.2.6 module \_3gpp-ns-nrm-common.yang

<CODE BEGINS>

module \_3gpp-ns-nrm-common {

 yang-version 1.1;

 namespace urn:3gpp:sa5:\_3gpp-ns-nrm-common;

 prefix ns3cmn3gpp;

 // import \_3gpp-common-subnetwork { prefix subnet3gpp; }

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

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

 organization "3GPP SA5";

 contact

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

 description "Common network slice definitions";

 reference "3GPP TS 28.541

 Management and orchestration;

 5G Network Resource Model (NRM);

 Information model definitions for network slice NRM (chapter 6)

 ";

 revision 2021-07-16 { reference CR-0566 ; }

 revision 2021-05-17 {

 description "Introduction of Common Data types";

 reference "CR-0485";

 }

 grouping XLThptGrp {

 list servAttrCom {

 description "This list represents the common properties of service

 requirement related attributes.";

 reference "GSMA NG.116 corresponding to Attribute categories,

 tagging and exposure";

 config false;

 key idx;

 max-elements 1;

 leaf idx {

 description "Synthetic index for the element.";

 type uint32;

 }

 uses ServAttrComGrp;

 }

 leaf guaThpt {

 description "This attribute describes the guaranteed data rate.";

 type uint64;

 units kbits/s;

 }

 leaf maxThpt {

 description "This attribute describes the maximum data rate.";

 type uint64;

 units kbits/s;

 }

 }

 typedef Tagging-enum {

 type enumeration {

 enum performance;

 enum function;

 enum operation;

 }

 }

 typedef Exposure-enum {

 type enumeration {

 enum API;

 enum KPI;

 }

 }

 typedef Category-enum {

 type enumeration {

 enum character;

 enum scalability;

 }

 }

 typedef Support-enum {

 type enumeration {

 enum NOT\_SUPPORTED;

 enum SUPPORTED;

 }

 }

 grouping ServAttrComGrp {

 leaf category {

 description "This attribute specifies the category of a service

 requirement/attribute of GST";

 type Category-enum;

 config false;

 }

 leaf-list tagging {

 description "This attribute specifies the tagging of a service

 requirement/attribute of GST in character category";

 when "../category = 'character'";

 type Tagging-enum;

 config false;

 }

 leaf exposure {

 description "This attribute specifies exposure mode of a service

 requirement/attribute of GST";

 type Exposure-enum;

 config false;

 }

 }

 typedef DeterminCommAvailability {

 type Support-enum;

 }

}

<CODE ENDS>

|  |
| --- |
| **End of changes** |