**3GPP TSG-SA3 Meeting #79-LI-e-b *s3i200732***

**Online, , 10th Nov 2020 - 12th Nov 2020**

|  |
| --- |
| *CR-Form-v12.1* |
| **CHANGE REQUEST** |
|  |
|  | **33.128** | **CR** | **0152** | **rev** | **-** | **Current version:** | **15.5.0** |  |
|  |
| *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 |  | Core Network |  |

|  |
| --- |
|  |
| ***Title:***  | Adding missing TAIList |
|  |  |
| ***Source to WG:*** | SA3-LI (National Technical Assistance) |
| ***Source to TSG:*** | SA3 |
|  |  |
| ***Work item code:*** | LI15 |  | ***Date:*** | 2020-11-11 |
|  |  |  |  |  |
| ***Category:*** | **F** |  | ***Release:*** | Rel-15 |
|  | *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:*** | Tracking area list is missing from initial registration IRI messages |
|  |  |
| ***Summary of change:*** | Adds a list of tracking areas to the initial registration IRI messages |
|  |  |
| ***Consequences if not approved:*** | Insufficient information in initial registraion IRI messages to meet LEA requirements. |
|  |  |
| ***Clauses affected:*** | 6.2.2.2.2, 6.2.2.2.5, Annex A |
|  |  |
|  | **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:*** |  |

 FIRST CHANGE

##### 6.2.2.2.2 Registration

The IRI-POI in the AMF shall generate an xIRI containing an AMFRegistration record when the IRI-POI present in the AMF detects that a UE matching one of the target identifiers provided via LI\_X1 has successfully registered to the 5GS via 3GPP NG-RAN or non-3GPP access. Accordingly, the IRI-POI in the AMF generates the xIRI when the following event is detected:

- AMF sends a N1: REGISTRATION ACCEPT message to the target UE and the UE 5G Mobility Management (5GMM) state within the AMF is changed to 5GMM-REGISTERED.

Table 6.2.2-1: Payload for AMFRegistration record

|  |  |  |
| --- | --- | --- |
| Field name | Description | M/C/O |
| registrationType | Specifies the type of registration, see TS 24.501 [13] clause 9.11.3.7. This is derived from the information received from the UE in the REGISTRATION REQUEST message. | M |
| registrationResult | Specifies the result of registration, see TS 24.501 [13] clause 9.11.3.6. | M |
| slice | Provide, if available, one or more of the following:- allowed NSSAI (see TS 24.501 [13] clause 9.11.3.37).- configured NSSAI (see TS 24.501 [13] clause 9.11.3.37).- rejected NSSAI (see TS 24.501 [13] clause 9.11.3.46).This is derived from the information sent to the UE in the REGISTRATION ACCEPT message. | C |
| sUPI | SUPI associated with the registration (see clause 6.2.2.4). | M |
| sUCI | SUCI used in the registration. | C |
| pEI | PEI provided by the UE during the registration, if available. | C |
| gPSI | GPSI obtained in the registration, if available as part of the subscription profile. | C |
| gUTI | 5G-GUTI provided as outcome of initial registration or used in other cases, see TS 24.501 [13] clause 5.5.1.2.2. | M |
| location | Location information determined by the network during the registration, if available.Encoded as a *userLocation* parameter (*location>locationInfo>userLocation*), see Annex A. | C |
| non3GPPAccessEndpoint | UE's local IP address used to reach the N3IWF, if available. IP addresses are given as 4 octets (for IPv4) or 16 octets (for IPv6) with the most significant octet first (network byte order). | C |
| fiveGSTAIList | List of tracking areas associated with the registration area within which the UE is current registered, see TS 24.501 [13], clause 9.11.3.4. (See NOTE 1) | C |
| NOTE 1: List shall be included each time there is a change to the registration area. |

 SECOND CHANGE

##### 6.2.2.2.5 Start of interception with registered UE

The IRI-POI in the AMF shall generate an xIRI containing an AMFStartOfInterceptionWithRegisteredUE record when the IRI-POI present in the AMF detects that interception is activated on a UE that has already been registered in the 5GS (see clause 6.2.2.4 on identity privacy). A UE is considered already registered to the 5GS when the 5GMM state for the access type (3GPP NG-RAN or non-3GPP access) for that UE is 5GMM-REGISTERED. Therefore, the IRI-POI present in the AMF shall generate the xIRI AMFStartOfInterceptionWithRegisteredUE record when it detects that a new interception for a UE is activated (i.e. provisioned by the LIPF) and the 5G mobility management state for the access type (3GPP NG-RAN or non-3GPP access) within the AMF for that UE is 5GMM-REGISTERED. If the UE is registered over both 3GPP NG-RAN and non-3GPP access, the IRI-POI present in the AMF shall generate an xIRI containing an AMFStartOfInterceptionWithRegisteredUE record for each access type.

Table 6.2.2-4: Payload for AMFStartOfInterceptionWithRegisteredUE record

|  |  |  |
| --- | --- | --- |
| Field name | Description | M/C/O |
| registrationResult | Specifies the result of registration, see TS 24.501 [13], clause 9.11.3.6. | M |
| registrationType | Specifies the type of registration, see TS 24.501 [13] clause 9.11.3.7, if available. | C |
| slice | Provide, if available, one or more of the following:- allowed NSSAI (see TS 24.501 [13] clause 9.11.3.37).- configured NSSAI (see TS 24.501 [13] clause 9.11.3.37).- rejected NSSAI (see TS 24.501 [13] clause 9.11.3.46).This is derived from the information that was sent to the UE in the REGISTRATION ACCEPT message. IRI-POI in AMF can include this information if and only if it retained the information that it had previously sent in the REGISTRATION ACCEPT message to the UE. | C |
| sUPI | SUPI associated with the registration (see clause 6.2.2.4). | M |
| sUCI | SUCI used in the registration, if available. | C |
| pEI | PEI provided by the UE during the registration, if available. | C |
| gPSI | GPSI obtained in the registration, if available as part of the subscription profile. | C |
| gUTI | 5G-GUTI provided as outcome of initial registration or used in other cases, see TS 24.501 [13], clause 5.5.1.2.2. | M |
| location | Location information, if available.Encoded as a *userLocation* parameter (*location>locationInfo>userLocation*), see Annex A. | C |
| non3GPPAccessEndpoint | UE's local IP address used to reach the N3IWF, if available. IP addresses are given as 4 octets (for IPv4) or 16 octets (for IPv6) with the most significant octet first (network byte order). | C |
| timeOfRegistration | Time at which the last registration occurred, if available. This is the time stamp when the REGISTRATION ACCEPT message is sent to the UE or (when applicable) when the REGISTRATION COMPLETE is received from the UE.Shall be given qualified with time zone information (i.e. as UTC or offset from UTC, not as local time). | C |
| fiveGSTAIList | List of tracking areas associated with the registration area within which the UE is current registered, see TS 24.501 [13], clause 9.11.3.4. (See NOTE 1) | C |
| NOTE 1: List shall be included each time there is a change to the registration area. |

 THIRD CHANGE

Annex A (normative): Structure of both the Internal and External Interfaces

TS33128Payloads

{itu-t(0) identified-organization(4) etsi(0) securityDomain(2) lawfulIntercept(2) threeGPP(4) ts33128(19) r15(15) version4(4)}

DEFINITIONS IMPLICIT TAGS EXTENSIBILITY IMPLIED ::=

BEGIN

-- =============

-- Relative OIDs

-- =============

xIRIPayloadOID RELATIVE-OID ::= {threeGPP(4) ts33128(19) r15(15) version3(3) xIRI(1)}

xCCPayloadOID RELATIVE-OID ::= {threeGPP(4) ts33128(19) r15(15) version1(1) xCC(2)}

iRIPayloadOID RELATIVE-OID ::= {threeGPP(4) ts33128(19) r15(15) version3(3) iRI(3)}

cCPayloadOID RELATIVE-OID ::= {threeGPP(4) ts33128(19) r15(15) version2(2) cC(4)}

lINotificationPayloadOID RELATIVE-OID ::= {threeGPP(4) ts33128(19) r15(15) version1(1) lINotification(5)}

-- ===============

-- X2 xIRI payload

-- ===============

XIRIPayload ::= SEQUENCE

{

 relativeOID [1] RELATIVE-OID,

 event [2] XIRIEvent

}

XIRIEvent ::= CHOICE

{

 -- Access and mobility related events, see clause 6.2.2

 registration [1] AMFRegistration,

 deregistration [2] AMFDeregistration,

 locationUpdate [3] AMFLocationUpdate,

 startOfInterceptionWithRegisteredUE [4] AMFStartOfInterceptionWithRegisteredUE,

 unsuccessfulAMProcedure [5] AMFUnsuccessfulProcedure,

 -- PDU session-related events, see clause 6.2.3

 pDUSessionEstablishment [6] SMFPDUSessionEstablishment,

 pDUSessionModification [7] SMFPDUSessionModification,

 pDUSessionRelease [8] SMFPDUSessionRelease,

 startOfInterceptionWithEstablishedPDUSession [9] SMFStartOfInterceptionWithEstablishedPDUSession,

 unsuccessfulSMProcedure [10] SMFUnsuccessfulProcedure,

 -- Subscriber-management related events, see clause 7.2.2

 servingSystemMessage [11] UDMServingSystemMessage,

 -- SMS-related events, see clause 6.2.5

 sMSMessage [12] SMSMessage,

 -- LALS-related events, see clause 7.3.3

 lALSReport [13] LALSReport,

 -- PDHR/PDSR-related events, see clause 6.2.3.4.1

 pDHeaderReport [14] PDHeaderReport,

 pDSummaryReport [15] PDSummaryReport

}

-- ==============

-- X3 xCC payload

-- ==============

-- No additional xCC payload definitions required in the present document.

-- ===============

-- HI2 IRI payload

-- ===============

IRIPayload ::= SEQUENCE

{

 relativeOID [1] RELATIVE-OID,

 event [2] IRIEvent,

 targetIdentifiers [3] SEQUENCE OF IRITargetIdentifier OPTIONAL

}

IRIEvent ::= CHOICE

{

 -- Registration-related events, see clause 6.2.2

 registration [1] AMFRegistration,

 deregistration [2] AMFDeregistration,

 locationUpdate [3] AMFLocationUpdate,

 startOfInterceptionWithRegisteredUE [4] AMFStartOfInterceptionWithRegisteredUE,

 unsuccessfulRegistrationProcedure [5] AMFUnsuccessfulProcedure,

 -- PDU session-related events, see clause 6.2.3

 pDUSessionEstablishment [6] SMFPDUSessionEstablishment,

 pDUSessionModification [7] SMFPDUSessionModification,

 pDUSessionRelease [8] SMFPDUSessionRelease,

 startOfInterceptionWithEstablishedPDUSession [9] SMFStartOfInterceptionWithEstablishedPDUSession,

 unsuccessfulSessionProcedure [10] SMFUnsuccessfulProcedure,

 -- Subscriber-management related events, see clause 7.2.2

 servingSystemMessage [11] UDMServingSystemMessage,

 -- SMS-related events, see clause 6.2.5

 sMSMessage [12] SMSMessage,

 -- LALS-related events, see clause 7.3.3

 lALSReport [13] LALSReport,

 -- PDHR/PDSR-related events, see clause 6.2.3.4.1

 pDHeaderReport [14] PDHeaderReport,

 pDSummaryReport [15] PDSummaryReport,

 -- MDF-related events, see clause 7.3.4

 mDFCellSiteReport [16] MDFCellSiteReport

}

IRITargetIdentifier ::= SEQUENCE

{

 identifier [1] TargetIdentifier,

 provenance [2] TargetIdentifierProvenance OPTIONAL

}

-- ==============

-- HI3 CC payload

-- ==============

CCPayload ::= SEQUENCE

{

 relativeOID [1] RELATIVE-OID,

 pDU [2] CCPDU

}

CCPDU ::= CHOICE

{

 uPFCCPDU [1] UPFCCPDU,

 extendedUPFCCPDU [2] ExtendedUPFCCPDU

}

-- ===========================

-- HI4 LI notification payload

-- ===========================

LINotificationPayload ::= SEQUENCE

{

 relativeOID [1] RELATIVE-OID,

 notification [2] LINotificationMessage

}

LINotificationMessage ::= CHOICE

{

 lINotification [1] LINotification

}

-- ==================

-- 5G AMF definitions

-- ==================

-- See clause 6.2.2.2.2 for details of this structure

AMFRegistration ::= SEQUENCE

{

 registrationType [1] AMFRegistrationType,

 registrationResult [2] AMFRegistrationResult,

 slice [3] Slice OPTIONAL,

 sUPI [4] SUPI,

 sUCI [5] SUCI OPTIONAL,

 pEI [6] PEI OPTIONAL,

 gPSI [7] GPSI OPTIONAL,

 gUTI [8] FiveGGUTI,

 location [9] Location OPTIONAL,

 non3GPPAccessEndpoint [10] UEEndpointAddress OPTIONAL,

 fiveGSTAIList [11] TAIList OPTIONAL

}

-- See clause 6.2.2.2.3 for details of this structure

AMFDeregistration ::= SEQUENCE

{

 deregistrationDirection [1] AMFDirection,

 accessType [2] AccessType,

 sUPI [3] SUPI OPTIONAL,

 sUCI [4] SUCI OPTIONAL,

 pEI [5] PEI OPTIONAL,

 gPSI [6] GPSI OPTIONAL,

 gUTI [7] FiveGGUTI OPTIONAL,

 cause [8] FiveGMMCause OPTIONAL,

 location [9] Location OPTIONAL

}

-- See clause 6.2.2.2.4 for details of this structure

AMFLocationUpdate ::= SEQUENCE

{

 sUPI [1] SUPI,

 sUCI [2] SUCI OPTIONAL,

 pEI [3] PEI OPTIONAL,

 gPSI [4] GPSI OPTIONAL,

 gUTI [5] FiveGGUTI OPTIONAL,

 location [6] Location

}

-- See clause 6.2.2.2.5 for details of this structure

AMFStartOfInterceptionWithRegisteredUE ::= SEQUENCE

{

 registrationResult [1] AMFRegistrationResult,

 registrationType [2] AMFRegistrationType OPTIONAL,

 slice [3] Slice OPTIONAL,

 sUPI [4] SUPI,

 sUCI [5] SUCI OPTIONAL,

 pEI [6] PEI OPTIONAL,

 gPSI [7] GPSI OPTIONAL,

 gUTI [8] FiveGGUTI,

 location [9] Location OPTIONAL,

 non3GPPAccessEndpoint [10] UEEndpointAddress OPTIONAL,

 timeOfRegistration [11] Timestamp OPTIONAL,

 fiveGSTAIList [12] TAIList OPTIONAL

}

-- See clause 6.2.2.2.6 for details of this structure

AMFUnsuccessfulProcedure ::= SEQUENCE

{

 failedProcedureType [1] AMFFailedProcedureType,

 failureCause [2] AMFFailureCause,

 requestedSlice [3] NSSAI OPTIONAL,

 sUPI [4] SUPI OPTIONAL,

 sUCI [5] SUCI OPTIONAL,

 pEI [6] PEI OPTIONAL,

 gPSI [7] GPSI OPTIONAL,

 gUTI [8] FiveGGUTI OPTIONAL,

 location [9] Location OPTIONAL

}

-- =================

-- 5G AMF parameters

-- =================

AMFID ::= SEQUENCE

{

 aMFRegionID [1] AMFRegionID,

 aMFSetID [2] AMFSetID,

 aMFPointer [3] AMFPointer

}

AMFDirection ::= ENUMERATED

{

 networkInitiated(1),

 uEInitiated(2)

}

AMFFailedProcedureType ::= ENUMERATED

{

 registration(1),

 sMS(2),

 pDUSessionEstablishment(3)

}

AMFFailureCause ::= CHOICE

{

 fiveGMMCause [1] FiveGMMCause,

 fiveGSMCause [2] FiveGSMCause

}

AMFPointer ::= INTEGER (0..63)

AMFRegistrationResult ::= ENUMERATED

{

 threeGPPAccess(1),

 nonThreeGPPAccess(2),

 threeGPPAndNonThreeGPPAccess(3)

}

AMFRegionID ::= INTEGER (0..255)

AMFRegistrationType ::= ENUMERATED

{

 initial(1),

 mobility(2),

 periodic(3),

 emergency(4)

}

AMFSetID ::= INTEGER (0..1023)

-- ==================

-- 5G SMF definitions

-- ==================

-- See clause 6.2.3.2.2 for details of this structure

SMFPDUSessionEstablishment ::= SEQUENCE

{

 sUPI [1] SUPI OPTIONAL,

 sUPIUnauthenticated [2] SUPIUnauthenticatedIndication OPTIONAL,

 pEI [3] PEI OPTIONAL,

 gPSI [4] GPSI OPTIONAL,

 pDUSessionID [5] PDUSessionID,

 gTPTunnelID [6] FTEID,

 pDUSessionType [7] PDUSessionType,

 sNSSAI [8] SNSSAI OPTIONAL,

 uEEndpoint [9] SEQUENCE OF UEEndpointAddress OPTIONAL,

 non3GPPAccessEndpoint [10] UEEndpointAddress OPTIONAL,

 location [11] Location OPTIONAL,

 dNN [12] DNN,

 aMFID [13] AMFID OPTIONAL,

 hSMFURI [14] HSMFURI OPTIONAL,

 requestType [15] FiveGSMRequestType,

 accessType [16] AccessType OPTIONAL,

 rATType [17] RATType OPTIONAL,

 sMPDUDNRequest [18] SMPDUDNRequest OPTIONAL

}

-- See clause 6.2.3.2.3 for details of this structure

SMFPDUSessionModification ::= SEQUENCE

{

 sUPI [1] SUPI OPTIONAL,

 sUPIUnauthenticated [2] SUPIUnauthenticatedIndication OPTIONAL,

 pEI [3] PEI OPTIONAL,

 gPSI [4] GPSI OPTIONAL,

 sNSSAI [5] SNSSAI OPTIONAL,

 non3GPPAccessEndpoint [6] UEEndpointAddress OPTIONAL,

 location [7] Location OPTIONAL,

 requestType [8] FiveGSMRequestType,

 accessType [9] AccessType OPTIONAL,

 rATType [10] RATType OPTIONAL

}

-- See clause 6.2.3.2.4 for details of this structure

SMFPDUSessionRelease ::= SEQUENCE

{

 sUPI [1] SUPI,

 pEI [2] PEI OPTIONAL,

 gPSI [3] GPSI OPTIONAL,

 pDUSessionID [4] PDUSessionID,

 timeOfFirstPacket [5] Timestamp OPTIONAL,

 timeOfLastPacket [6] Timestamp OPTIONAL,

 uplinkVolume [7] INTEGER OPTIONAL,

 downlinkVolume [8] INTEGER OPTIONAL,

 location [9] Location OPTIONAL

}

-- See clause 6.2.3.2.5 for details of this structure

SMFStartOfInterceptionWithEstablishedPDUSession ::= SEQUENCE

{

 sUPI [1] SUPI OPTIONAL,

 sUPIUnauthenticated [2] SUPIUnauthenticatedIndication OPTIONAL,

 pEI [3] PEI OPTIONAL,

 gPSI [4] GPSI OPTIONAL,

 pDUSessionID [5] PDUSessionID,

 gTPTunnelID [6] FTEID,

 pDUSessionType [7] PDUSessionType,

 sNSSAI [8] SNSSAI OPTIONAL,

 uEEndpoint [9] SEQUENCE OF UEEndpointAddress,

 non3GPPAccessEndpoint [10] UEEndpointAddress OPTIONAL,

 location [11] Location OPTIONAL,

 dNN [12] DNN,

 aMFID [13] AMFID OPTIONAL,

 hSMFURI [14] HSMFURI OPTIONAL,

 requestType [15] FiveGSMRequestType,

 accessType [16] AccessType OPTIONAL,

 rATType [17] RATType OPTIONAL,

 sMPDUDNRequest [18] SMPDUDNRequest OPTIONAL

}

-- See clause 6.2.3.2.6 for details of this structure

SMFUnsuccessfulProcedure ::= SEQUENCE

{

 failedProcedureType [1] SMFFailedProcedureType,

 failureCause [2] FiveGSMCause,

 initiator [3] Initiator,

 requestedSlice [4] NSSAI OPTIONAL,

 sUPI [5] SUPI OPTIONAL,

 sUPIUnauthenticated [6] SUPIUnauthenticatedIndication OPTIONAL,

 pEI [7] PEI OPTIONAL,

 gPSI [8] GPSI OPTIONAL,

 pDUSessionID [9] PDUSessionID OPTIONAL,

 uEEndpoint [10] SEQUENCE OF UEEndpointAddress OPTIONAL,

 non3GPPAccessEndpoint [11] UEEndpointAddress OPTIONAL,

 dNN [12] DNN OPTIONAL,

 aMFID [13] AMFID OPTIONAL,

 hSMFURI [14] HSMFURI OPTIONAL,

 requestType [15] FiveGSMRequestType OPTIONAL,

 accessType [16] AccessType OPTIONAL,

 rATType [17] RATType OPTIONAL,

 sMPDUDNRequest [18] SMPDUDNRequest OPTIONAL,

 location [19] Location OPTIONAL

}

-- =================

-- 5G SMF parameters

-- =================

SMFFailedProcedureType ::= ENUMERATED

{

 pDUSessionEstablishment(1),

 pDUSessionModification(2),

 pDUSessionRelease(3)

}

-- ==================

-- 5G UPF definitions

-- ==================

UPFCCPDU ::= OCTET STRING

-- See clause 6.2.3.8 for the details of this structure

ExtendedUPFCCPDU ::= SEQUENCE

{

 payload [1] UPFCCPDUPayload,

 qFI [2] QFI OPTIONAL

}

-- =================

-- 5G UPF parameters

-- =================

UPFCCPDUPayload ::= CHOICE

{

 uPFIPCC [1] OCTET STRING,

 uPFEthernetCC [2] OCTET STRING,

 uPFUnstructuredCC [3] OCTET STRING

}

QFI ::= INTEGER (0..63)

-- ==================

-- 5G UDM definitions

-- ==================

UDMServingSystemMessage ::= SEQUENCE

{

 sUPI [1] SUPI,

 pEI [2] PEI OPTIONAL,

 gPSI [3] GPSI OPTIONAL,

 gUAMI [4] GUAMI OPTIONAL,

 gUMMEI [5] GUMMEI OPTIONAL,

 pLMNID [6] PLMNID OPTIONAL,

 servingSystemMethod [7] UDMServingSystemMethod

}

-- =================

-- 5G UDM parameters

-- =================

UDMServingSystemMethod ::= ENUMERATED

{

 amf3GPPAccessRegistration(0),

 amfNon3GPPAccessRegistration(1),

 unknown(2)

}

-- ===================

-- 5G SMSF definitions

-- ===================

-- See clause 6.2.5.3 for details of this structure

SMSMessage ::= SEQUENCE

{

 originatingSMSParty [1] SMSParty,

 terminatingSMSParty [2] SMSParty,

 direction [3] Direction,

 transferStatus [4] SMSTransferStatus,

 otherMessage [5] SMSOtherMessageIndication OPTIONAL,

 location [6] Location OPTIONAL,

 peerNFAddress [7] SMSNFAddress OPTIONAL,

 peerNFType [8] SMSNFType OPTIONAL,

 sMSTPDUData [9] SMSTPDUData OPTIONAL

}

-- ==================

-- 5G SMSF parameters

-- ==================

SMSParty ::= SEQUENCE

{

 sUPI [1] SUPI OPTIONAL,

 pEI [2] PEI OPTIONAL,

 gPSI [3] GPSI OPTIONAL

}

SMSTransferStatus ::= ENUMERATED

{

 transferSucceeded(1),

 transferFailed(2),

 undefined(3)

}

SMSOtherMessageIndication ::= BOOLEAN

SMSNFAddress ::= CHOICE

{

 iPAddress [1] IPAddress,

 e164Number [2] E164Number

}

SMSNFType ::= ENUMERATED

{

 sMSGMSC(1),

 iWMSC(2),

 sMSRouter(3)

}

SMSTPDUData ::= CHOICE

{

 sMSTPDU [1] SMSTPDU

}

SMSTPDU ::= OCTET STRING (SIZE(1..270))

-- ===================

-- 5G LALS definitions

-- ===================

LALSReport ::= SEQUENCE

{

 sUPI [1] SUPI OPTIONAL,

 pEI [2] PEI OPTIONAL,

 gPSI [3] GPSI OPTIONAL,

 location [4] Location OPTIONAL

}

-- =====================

-- PDHR/PDSR definitions

-- =====================

PDHeaderReport ::= SEQUENCE

{

 pDUSessionID [1] PDUSessionID,

 sourceIPAddress [2] IPAddress,

 sourcePort [3] PortNumber OPTIONAL,

 destinationIPAddress [4] IPAddress,

 destinationPort [5] PortNumber OPTIONAL,

 nextLayerProtocol [6] NextLayerProtocol,

 iPv6flowLabel [7] IPv6FlowLabel OPTIONAL,

 direction [8] Direction,

 packetSize [9] INTEGER

}

PDSummaryReport ::= SEQUENCE

{

 pDUSessionID [1] PDUSessionID,

 sourceIPAddress [2] IPAddress,

 sourcePort [3] PortNumber OPTIONAL,

 destinationIPAddress [4] IPAddress,

 destinationPort [5] PortNumber OPTIONAL,

 nextLayerProtocol [6] NextLayerProtocol,

 iPv6flowLabel [7] IPv6FlowLabel OPTIONAL,

 direction [8] Direction,

 pDSRSummaryTrigger [9] PDSRSummaryTrigger,

 firstPacketTimestamp [10] Timestamp,

 lastPacketTimestamp [11] Timestamp,

 packetCount [12] INTEGER,

 byteCount [13] INTEGER

}

-- ====================

-- PDHR/PDSR parameters

-- ====================

PDSRSummaryTrigger ::= ENUMERATED

{

 timerExpiry(1),

 packetCount(2),

 byteCount(3)

}

-- ===========================

-- LI Notification definitions

-- ===========================

LINotification ::= SEQUENCE

{

 notificationType [1] LINotificationType,

 appliedTargetID [2] TargetIdentifier OPTIONAL,

 appliedDeliveryInformation [3] SEQUENCE OF LIAppliedDeliveryInformation OPTIONAL,

 appliedStartTime [4] Timestamp OPTIONAL,

 appliedEndTime [5] Timestamp OPTIONAL

}

-- ==========================

-- LI Notification parameters

-- ==========================

LINotificationType ::= ENUMERATED

{

 activation(1),

 deactivation(2),

 modification(3)

}

LIAppliedDeliveryInformation ::= SEQUENCE

{

 hI2DeliveryIPAddress [1] IPAddress OPTIONAL,

 hI2DeliveryPortNumber [2] PortNumber OPTIONAL,

 hI3DeliveryIPAddress [3] IPAddress OPTIONAL,

 hI3DeliveryPortNumber [4] PortNumber OPTIONAL

}

-- ===============

-- MDF definitions

-- ===============

MDFCellSiteReport ::= SEQUENCE OF CellInformation

-- =================

-- Common Parameters

-- =================

AccessType ::= ENUMERATED

{

 threeGPPAccess(1),

 nonThreeGPPAccess(2),

 threeGPPandNonThreeGPPAccess(3)

}

Direction ::= ENUMERATED

{

 fromTarget(1),

 toTarget(2)

}

DNN ::= UTF8String

E164Number ::= NumericString (SIZE(1..15))

FiveGGUTI ::= SEQUENCE

{

 mCC [1] MCC,

 mNC [2] MNC,

 aMFRegionID [3] AMFRegionID,

 aMFSetID [4] AMFSetID,

 aMFPointer [5] AMFPointer,

 fiveGTMSI [6] FiveGTMSI

}

FiveGMMCause ::= INTEGER (0..255)

FiveGSMRequestType ::= ENUMERATED

{

 initialRequest(1),

 existingPDUSession(2),

 initialEmergencyRequest(3),

 existingEmergencyPDUSession(4),

 modificationRequest(5),

 reserved(6)

}

FiveGSMCause ::= INTEGER (0..255)

FiveGTMSI ::= INTEGER (0..4294967295)

FTEID ::= SEQUENCE

{

 tEID [1] INTEGER (0.. 4294967295),

 iPv4Address [2] IPv4Address OPTIONAL,

 iPv6Address [3] IPv6Address OPTIONAL

}

GPSI ::= CHOICE

{

 mSISDN [1] MSISDN,

 nAI [2] NAI

}

GUAMI ::= SEQUENCE

{

 aMFID [1] AMFID,

 pLMNID [2] PLMNID

}

GUMMEI ::= SEQUENCE

{

 mMEID [1] MMEID,

 mCC [2] MCC,

 mNC [3] MNC

}

HomeNetworkPublicKeyID ::= OCTET STRING

HSMFURI ::= UTF8String

IMEI ::= NumericString (SIZE(14))

IMEISV ::= NumericString (SIZE(16))

IMSI ::= NumericString (SIZE(6..15))

Initiator ::= ENUMERATED

{

 uE(1),

 network(2),

 unknown(3)

}

IPAddress ::= CHOICE

{

 iPv4Address [1] IPv4Address,

 iPv6Address [2] IPv6Address

}

IPv4Address ::= OCTET STRING (SIZE(4))

IPv6Address ::= OCTET STRING (SIZE(16))

IPv6FlowLabel ::= INTEGER(0..1048575)

MACAddress ::= OCTET STRING (SIZE(6))

MCC ::= NumericString (SIZE(3))

MNC ::= NumericString (SIZE(2..3))

MMEID ::= SEQUENCE

{

 mMEGI [1] MMEGI,

 mMEC [2] MMEC

}

MMEC ::= NumericString

MMEGI ::= NumericString

MSISDN ::= NumericString (SIZE(1..15))

NAI ::= UTF8String

NextLayerProtocol ::= INTEGER(0..255)

NSSAI ::= SEQUENCE OF SNSSAI

PLMNID ::= SEQUENCE

{

 mCC [1] MCC,

 mNC [2] MNC

}

PDUSessionID ::= INTEGER (0..255)

PDUSessionType ::= ENUMERATED

{

 iPv4(1),

 iPv6(2),

 iPv4v6(3),

 unstructured(4),

 ethernet(5)

}

PEI ::= CHOICE

{

 iMEI [1] IMEI,

 iMEISV [2] IMEISV

}

PortNumber ::= INTEGER(0..65535)

ProtectionSchemeID ::= INTEGER (0..15)

RATType ::= ENUMERATED

{

 nR(1),

 eUTRA(2),

 wLAN(3),

 virtual(4)

}

RejectedNSSAI ::= SEQUENCE OF RejectedSNSSAI

RejectedSNSSAI ::= SEQUENCE

{

 causeValue [1] RejectedSliceCauseValue,

 sNSSAI [2] SNSSAI

}

RejectedSliceCauseValue ::= INTEGER (0..255)

RoutingIndicator ::= INTEGER (0..9999)

SchemeOutput ::= OCTET STRING

Slice ::= SEQUENCE

{

 allowedNSSAI [1] NSSAI OPTIONAL,

 configuredNSSAI [2] NSSAI OPTIONAL,

 rejectedNSSAI [3] RejectedNSSAI OPTIONAL

}

SMPDUDNRequest ::= OCTET STRING

SNSSAI ::= SEQUENCE

{

 sliceServiceType [1] INTEGER (0..255),

 sliceDifferentiator [2] OCTET STRING (SIZE(3)) OPTIONAL

}

SUCI ::= SEQUENCE

{

 mCC [1] MCC,

 mNC [2] MNC,

 routingIndicator [3] RoutingIndicator,

 protectionSchemeID [4] ProtectionSchemeID,

 homeNetworkPublicKeyID [5] HomeNetworkPublicKeyID,

 schemeOutput [6] SchemeOutput

}

SUPI ::= CHOICE

{

 iMSI [1] IMSI,

 nAI [2] NAI

}

SUPIUnauthenticatedIndication ::= BOOLEAN

TargetIdentifier ::= CHOICE

{

 sUPI [1] SUPI,

 iMSI [2] IMSI,

 pEI [3] PEI,

 iMEI [4] IMEI,

 gPSI [5] GPSI,

 mISDN [6] MSISDN,

 nAI [7] NAI,

 iPv4Address [8] IPv4Address,

 iPv6Address [9] IPv6Address,

 ethernetAddress [10] MACAddress

}

TargetIdentifierProvenance ::= ENUMERATED

{

 lEAProvided(1),

 observed(2),

 matchedOn(3),

 other(4)

}

Timestamp ::= GeneralizedTime

UEEndpointAddress ::= CHOICE

{

 iPv4Address [1] IPv4Address,

 iPv6Address [2] IPv6Address,

 ethernetAddress [3] MACAddress

}

-- ===================

-- Location parameters

-- ===================

Location ::= SEQUENCE

{

 locationInfo [1] LocationInfo OPTIONAL,

 positioningInfo [2] PositioningInfo OPTIONAL,

 locationPresenceReport [3] LocationPresenceReport OPTIONAL

}

CellSiteInformation ::= SEQUENCE

{

 geographicalCoordinates [1] GeographicalCoordinates,

 azimuth [2] INTEGER (0..359) OPTIONAL,

 operatorSpecificInformation [3] UTF8String OPTIONAL

}

-- TS 29.518 [22], clause 6.4.6.2.6

LocationInfo ::= SEQUENCE

{

 userLocation [1] UserLocation OPTIONAL,

 currentLoc [2] BOOLEAN OPTIONAL,

 geoInfo [3] GeographicArea OPTIONAL,

 rATType [4] RATType OPTIONAL,

 timeZone [5] TimeZone OPTIONAL,

 additionalCellIDs [6] SEQUENCE OF CellInformation OPTIONAL

}

-- TS 29.571 [17], clause 5.4.4.7

UserLocation ::= SEQUENCE

{

 eUTRALocation [1] EUTRALocation OPTIONAL,

 nRLocation [2] NRLocation OPTIONAL,

 n3GALocation [3] N3GALocation OPTIONAL

}

-- TS 29.571 [17], clause 5.4.4.8

EUTRALocation ::= SEQUENCE

{

 tAI [1] TAI,

 eCGI [2] ECGI,

 ageOfLocatonInfo [3] INTEGER OPTIONAL,

 uELocationTimestamp [4] Timestamp OPTIONAL,

 geographicalInformation [5] UTF8String OPTIONAL,

 geodeticInformation [6] UTF8String OPTIONAL,

 globalNGENbID [7] GlobalRANNodeID OPTIONAL,

 cellSiteInformation [8] CellSiteInformation OPTIONAL

}

-- TS 29.571 [17], clause 5.4.4.9

NRLocation ::= SEQUENCE

{

 tAI [1] TAI,

 nCGI [2] NCGI,

 ageOfLocatonInfo [3] INTEGER OPTIONAL,

 uELocationTimestamp [4] Timestamp OPTIONAL,

 geographicalInformation [5] UTF8String OPTIONAL,

 geodeticInformation [6] UTF8String OPTIONAL,

 globalGNbID [7] GlobalRANNodeID OPTIONAL,

 cellSiteInformation [8] CellSiteInformation OPTIONAL

}

-- TS 29.571 [17], clause 5.4.4.10

N3GALocation ::= SEQUENCE

{

 tAI [1] TAI OPTIONAL,

 n3IWFID [2] N3IWFIDNGAP OPTIONAL,

 uEIPAddr [3] IPAddr OPTIONAL,

 portNumber [4] INTEGER OPTIONAL

}

-- TS 38.413 [23], clause 9.3.2.4

IPAddr ::= SEQUENCE

{

 iPv4Addr [1] IPv4Address OPTIONAL,

 iPv6Addr [2] IPv6Address OPTIONAL

}

-- TS 29.571 [17], clause 5.4.4.28

GlobalRANNodeID ::= SEQUENCE

{

 pLMNID [1] PLMNID,

 aNNodeID [2] CHOICE

 {

 n3IWFID [1] N3IWFIDSBI,

 gNbID [2] GNbID,

 nGENbID [3] NGENbID

 }

}

-- TS 38.413 [23], clause 9.3.1.6

GNbID ::= BIT STRING(SIZE(22..32))

-- TS 29.571 [17], clause 5.4.4.4

TAI ::= SEQUENCE

{

 pLMNID [1] PLMNID,

 tAC [2] TAC

}

TAIList ::= SEQUENCE OF TAI

-- TS 29.571 [17], clause 5.4.4.5

ECGI ::= SEQUENCE

{

 pLMNID [1] PLMNID,

 eUTRACellID [2] EUTRACellID

}

-- TS 29.571 [17], clause 5.4.4.6

NCGI ::= SEQUENCE

{

 pLMNID [1] PLMNID,

 nRCellID [2] NRCellID

}

RANCGI ::= CHOICE

{

 eCGI [1] ECGI,

 nCGI [2] NCGI

}

CellInformation ::= SEQUENCE

{

 rANCGI [1] RANCGI,

 cellSiteinformation [2] CellSiteInformation OPTIONAL,

 timeOfLocation [3] Timestamp OPTIONAL

}

-- TS 38.413 [23], clause 9.3.1.57

N3IWFIDNGAP ::= BIT STRING (SIZE(16))

-- TS 29.571 [17], clause 5.4.4.28

N3IWFIDSBI ::= UTF8String

-- TS 29.571 [17], table 5.4.2-1

TAC ::= OCTET STRING (SIZE(2..3))

-- TS 38.413 [23], clause 9.3.1.9

EUTRACellID ::= BIT STRING (SIZE(28))

-- TS 38.413 [23], clause 9.3.1.7

NRCellID ::= BIT STRING (SIZE(36))

-- TS 38.413 [23], clause 9.3.1.8

NGENbID ::= CHOICE

{

 macroNGENbID [1] BIT STRING (SIZE(20)),

 shortMacroNGENbID [2] BIT STRING (SIZE(18)),

 longMacroNGENbID [3] BIT STRING (SIZE(21))

}

-- TS 29.518 [22], clause 6.4.6.2.3

PositioningInfo ::= SEQUENCE

{

 positionInfo [1] LocationData OPTIONAL,

 rawMLPResponse [2] RawMLPResponse OPTIONAL

}

RawMLPResponse ::= CHOICE

{

 -- The following parameter contains a copy of unparsed XML code of the

 -- MLP response message, i.e. the entire XML document containing

 -- a <slia> (described in OMA-TS-MLP-V3\_5-20181211-C [20], clause 5.2.3.2.2) or

 -- a <slirep> (described in OMA-TS-MLP-V3\_5-20181211-C [20], clause 5.2.3.2.3) MLP message.

 mLPPositionData [1] UTF8String,

 -- OMA MLP result id, defined in OMA-TS-MLP-V3\_5-20181211-C [20], Clause 5.4

 mLPErrorCode [2] INTEGER (1..699)

}

-- TS 29.572 [24], clause 6.1.6.2.3

LocationData ::= SEQUENCE

{

 locationEstimate [1] GeographicArea,

 accuracyFulfilmentIndicator [2] AccuracyFulfilmentIndicator OPTIONAL,

 ageOfLocationEstimate [3] AgeOfLocationEstimate OPTIONAL,

 velocityEstimate [4] VelocityEstimate OPTIONAL,

 civicAddress [5] CivicAddress OPTIONAL,

 positioningDataList [6] SET OF PositioningMethodAndUsage OPTIONAL,

 gNSSPositioningDataList [7] SET OF GNSSPositioningMethodAndUsage OPTIONAL,

 eCGI [8] ECGI OPTIONAL,

 nCGI [9] NCGI OPTIONAL,

 altitude [10] Altitude OPTIONAL,

 barometricPressure [11] BarometricPressure OPTIONAL

}

-- TS 29.518 [22], clause 6.2.6.2.5

LocationPresenceReport ::= SEQUENCE

{

 type [1] AMFEventType,

 timestamp [2] Timestamp,

 areaList [3] SET OF AMFEventArea OPTIONAL,

 timeZone [4] TimeZone OPTIONAL,

 accessTypes [5] SET OF AccessType OPTIONAL,

 rMInfoList [6] SET OF RMInfo OPTIONAL,

 cMInfoList [7] SET OF CMInfo OPTIONAL,

 reachability [8] UEReachability OPTIONAL,

 location [9] UserLocation OPTIONAL,

 additionalCellIDs [10] SEQUENCE OF CellInformation OPTIONAL

}

-- TS 29.518 [22], clause 6.2.6.3.3

AMFEventType ::= ENUMERATED

{

 locationReport(1),

 presenceInAOIReport(2)

}

-- TS 29.518 [22], clause 6.2.6.2.16

AMFEventArea ::= SEQUENCE

{

 presenceInfo [1] PresenceInfo OPTIONAL,

 lADNInfo [2] LADNInfo OPTIONAL

}

-- TS 29.571 [17], clause 5.4.4.27

PresenceInfo ::= SEQUENCE

{

 presenceState [1] PresenceState OPTIONAL,

 trackingAreaList [2] SET OF TAI OPTIONAL,

 eCGIList [3] SET OF ECGI OPTIONAL,

 nCGIList [4] SET OF NCGI OPTIONAL,

 globalRANNodeIDList [5] SET OF GlobalRANNodeID OPTIONAL

}

-- TS 29.518 [22], clause 6.2.6.2.17

LADNInfo ::= SEQUENCE

{

 lADN [1] UTF8String,

 presence [2] PresenceState OPTIONAL

}

-- TS 29.571 [17], clause 5.4.3.20

PresenceState ::= ENUMERATED

{

 inArea(1),

 outOfArea(2),

 unknown(3),

 inactive(4)

}

-- TS 29.518 [22], clause 6.2.6.2.8

RMInfo ::= SEQUENCE

{

 rMState [1] RMState,

 accessType [2] AccessType

}

-- TS 29.518 [22], clause 6.2.6.2.9

CMInfo ::= SEQUENCE

{

 cMState [1] CMState,

 accessType [2] AccessType

}

-- TS 29.518 [22], clause 6.2.6.3.7

UEReachability ::= ENUMERATED

{

 unreachable(1),

 reachable(2),

 regulatoryOnly(3)

}

-- TS 29.518 [22], clause 6.2.6.3.9

RMState ::= ENUMERATED

{

 registered(1),

 deregistered(2)

}

-- TS 29.518 [22], clause 6.2.6.3.10

CMState ::= ENUMERATED

{

 idle(1),

 connected(2)

}

-- TS 29.572 [24], clause 6.1.6.2.5

GeographicArea ::= CHOICE

{

 point [1] Point,

 pointUncertaintyCircle [2] PointUncertaintyCircle,

 pointUncertaintyEllipse [3] PointUncertaintyEllipse,

 polygon [4] Polygon,

 pointAltitude [5] PointAltitude,

 pointAltitudeUncertainty [6] PointAltitudeUncertainty,

 ellipsoidArc [7] EllipsoidArc

}

-- TS 29.572 [24], clause 6.1.6.3.12

AccuracyFulfilmentIndicator ::= ENUMERATED

{

 requestedAccuracyFulfilled(1),

 requestedAccuracyNotFulfilled(2)

}

-- TS 29.572 [24], clause

VelocityEstimate ::= CHOICE

{

 horVelocity [1] HorizontalVelocity,

 horWithVertVelocity [2] HorizontalWithVerticalVelocity,

 horVelocityWithUncertainty [3] HorizontalVelocityWithUncertainty,

 horWithVertVelocityAndUncertainty [4] HorizontalWithVerticalVelocityAndUncertainty

}

-- TS 29.572 [24], clause 6.1.6.2.14

CivicAddress ::= SEQUENCE

{

 country [1] UTF8String,

 a1 [2] UTF8String OPTIONAL,

 a2 [3] UTF8String OPTIONAL,

 a3 [4] UTF8String OPTIONAL,

 a4 [5] UTF8String OPTIONAL,

 a5 [6] UTF8String OPTIONAL,

 a6 [7] UTF8String OPTIONAL,

 prd [8] UTF8String OPTIONAL,

 pod [9] UTF8String OPTIONAL,

 sts [10] UTF8String OPTIONAL,

 hno [11] UTF8String OPTIONAL,

 hns [12] UTF8String OPTIONAL,

 lmk [13] UTF8String OPTIONAL,

 loc [14] UTF8String OPTIONAL,

 nam [15] UTF8String OPTIONAL,

 pc [16] UTF8String OPTIONAL,

 bld [17] UTF8String OPTIONAL,

 unit [18] UTF8String OPTIONAL,

 flr [19] UTF8String OPTIONAL,

 room [20] UTF8String OPTIONAL,

 plc [21] UTF8String OPTIONAL,

 pcn [22] UTF8String OPTIONAL,

 pobox [23] UTF8String OPTIONAL,

 addcode [24] UTF8String OPTIONAL,

 seat [25] UTF8String OPTIONAL,

 rd [26] UTF8String OPTIONAL,

 rdsec [27] UTF8String OPTIONAL,

 rdbr [28] UTF8String OPTIONAL,

 rdsubbr [29] UTF8String OPTIONAL

}

-- TS 29.572 [24], clause 6.1.6.2.15

PositioningMethodAndUsage ::= SEQUENCE

{

 method [1] PositioningMethod,

 mode [2] PositioningMode,

 usage [3] Usage

}

-- TS 29.572 [24], clause 6.1.6.2.16

GNSSPositioningMethodAndUsage ::= SEQUENCE

{

 mode [1] PositioningMode,

 gNSS [2] GNSSID,

 usage [3] Usage

}

-- TS 29.572 [24], clause 6.1.6.2.6

Point ::= SEQUENCE

{

 geographicalCoordinates [1] GeographicalCoordinates

}

-- TS 29.572 [24], clause 6.1.6.2.7

PointUncertaintyCircle ::= SEQUENCE

{

 geographicalCoordinates [1] GeographicalCoordinates,

 uncertainty [2] Uncertainty

}

-- TS 29.572 [24], clause 6.1.6.2.8

PointUncertaintyEllipse ::= SEQUENCE

{

 geographicalCoordinates [1] GeographicalCoordinates,

 uncertainty [2] UncertaintyEllipse,

 confidence [3] Confidence

}

-- TS 29.572 [24], clause 6.1.6.2.9

Polygon ::= SEQUENCE

{

 pointList [1] SET SIZE (3..15) OF GeographicalCoordinates

}

-- TS 29.572 [24], clause 6.1.6.2.10

PointAltitude ::= SEQUENCE

{

 point [1] GeographicalCoordinates,

 altitude [2] Altitude

}

-- TS 29.572 [24], clause 6.1.6.2.11

PointAltitudeUncertainty ::= SEQUENCE

{

 point [1] GeographicalCoordinates,

 altitude [2] Altitude,

 uncertaintyEllipse [3] UncertaintyEllipse,

 uncertaintyAltitude [4] Uncertainty,

 confidence [5] Confidence

}

-- TS 29.572 [24], clause 6.1.6.2.12

EllipsoidArc ::= SEQUENCE

{

 point [1] GeographicalCoordinates,

 innerRadius [2] InnerRadius,

 uncertaintyRadius [3] Uncertainty,

 offsetAngle [4] Angle,

 includedAngle [5] Angle,

 confidence [6] Confidence

}

-- TS 29.572 [24], clause 6.1.6.2.4

GeographicalCoordinates ::= SEQUENCE

{

 latitude [1] UTF8String,

 longitude [2] UTF8String

}

-- TS 29.572 [24], clause 6.1.6.2.22

UncertaintyEllipse ::= SEQUENCE

{

 semiMajor [1] Uncertainty,

 semiMinor [2] Uncertainty,

 orientationMajor [3] Orientation

}

-- TS 29.572 [24], clause 6.1.6.2.18

HorizontalVelocity ::= SEQUENCE

{

 hSpeed [1] HorizontalSpeed,

 bearing [2] Angle

}

-- TS 29.572 [24], clause 6.1.6.2.19

HorizontalWithVerticalVelocity ::= SEQUENCE

{

 hSpeed [1] HorizontalSpeed,

 bearing [2] Angle,

 vSpeed [3] VerticalSpeed,

 vDirection [4] VerticalDirection

}

-- TS 29.572 [24], clause 6.1.6.2.20

HorizontalVelocityWithUncertainty ::= SEQUENCE

{

 hSpeed [1] HorizontalSpeed,

 bearing [2] Angle,

 uncertainty [3] SpeedUncertainty

}

-- TS 29.572 [24], clause 6.1.6.2.21

HorizontalWithVerticalVelocityAndUncertainty ::= SEQUENCE

{

 hspeed [1] HorizontalSpeed,

 bearing [2] Angle,

 vSpeed [3] VerticalSpeed,

 vDirection [4] VerticalDirection,

 hUncertainty [5] SpeedUncertainty,

 vUncertainty [6] SpeedUncertainty

}

-- The following types are described in TS 29.572 [24], table 6.1.6.3.2-1

Altitude ::= UTF8String

Angle ::= INTEGER (0..360)

Uncertainty ::= INTEGER (0..127)

Orientation ::= INTEGER (0..180)

Confidence ::= INTEGER (0..100)

InnerRadius ::= INTEGER (0..65535)

AgeOfLocationEstimate ::= INTEGER (0..32767)

HorizontalSpeed ::= UTF8String

VerticalSpeed ::= UTF8String

SpeedUncertainty ::= UTF8String

BarometricPressure ::= INTEGER (30000..155000)

-- TS 29.572 [24], clause 6.1.6.3.13

VerticalDirection ::= ENUMERATED

{

 upward(1),

 downward(2)

}

-- TS 29.572 [24], clause 6.1.6.3.6

PositioningMethod ::= ENUMERATED

{

 cellID(1),

 eCID(2),

 oTDOA(3),

 barometricPresure(4),

 wLAN(5),

 bluetooth(6),

 mBS(7)

}

-- TS 29.572 [24], clause 6.1.6.3.7

PositioningMode ::= ENUMERATED

{

 uEBased(1),

 uEAssisted(2),

 conventional(3)

}

-- TS 29.572 [24], clause 6.1.6.3.8

GNSSID ::= ENUMERATED

{

 gPS(1),

 galileo(2),

 sBAS(3),

 modernizedGPS(4),

 qZSS(5),

 gLONASS(6)

}

-- TS 29.572 [24], clause 6.1.6.3.9

Usage ::= ENUMERATED

{

 unsuccess(1),

 successResultsNotUsed(2),

 successResultsUsedToVerifyLocation(3),

 successResultsUsedToGenerateLocation(4),

 successMethodNotDetermined(5)

}

-- TS 29.571 [17], table 5.2.2-1

TimeZone ::= UTF8String

END

 END OF CHANGES