ETSI TS 103 280 V2.18.1 (2026-03)

TECHNICAL SPECIFICATION

Lawful Interception (LI);
Dictionary for common parameters

Reference
RTS/LI-00311

Keywords
dictionary, lawful interception, security

ETSI

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

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

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

Contents

Intellectual Property Rights

Essential patents

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

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

Trademarks

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

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

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Lawful Interception (LI).

The ASN.1, JSON Schema and XSD technical implementations are both available as an electronic attachment to the present document.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document defines a dictionary of parameters that are commonly used in multiple TC LI specifications. Aside from defining a dictionary, the present document aims to provide technical means for other specifications to use. It is encouraged to use the present document in the development of new specifications.

It is foreseen that regular maintenance of the present document is required. As such, release management requirements will be defined.

Before accepting any new common parameter, the present document will provide a set of requirements the parameter has to comply to in order to become a common parameter.

2 References

2.1 Normative references

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

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

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

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

[1] ETSI TS 102 232-1: "Lawful Interception (LI); Handover Interface and Service-Specific Details (SSD) for IP delivery; Part 1: Handover specification for IP delivery".

[2] W3C® Recommendation 5 April 2012: "W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes".

[3] Recommendation ITU-T X.680: "Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation".

[4] Recommendation ITU-T E.164: "The international public telecommunication numbering plan".

[5] Recommendation ITU-T E.212: "The international identification plan for public networks and subscriptions".

[6] ETSI TS 123 003: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; 5G; Numbering, addressing and identification (3GPP TS 23.003)".

[7] ETSI TS 102 657: "Lawful Interception (LI); Retained data handling; Handover interface for the request and delivery of retained data".

[8] IETF RFC 791: "Internet Protocol".

[9] IETF RFC 4632: "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan".

[10] IETF RFC 8200: "Internet Protocol, Version 6 (IPv6) Specification".

[11] IETF RFC 4291: "IP Version 6 Addressing Architecture".

[12] IETF RFC 9293: "Transmission Control Protocol (TCP)".

[13] IETF RFC 768: "User Datagram Protocol".

[14] IEEE 802.3™: "IEEE Standard for Ethernet".

[15] IETF RFC 5322: "Internet Message Format".

[16] WHATWG community: "HTML Living standard".

[17] IETF RFC 4122: "A Universally Unique IDentifier (UUID) URN Namespace".

[18] ISO 3166-1: "Codes for the representation of names of countries and their subdivisions — Part 1: Country codes".

[19] Void.

[20] ISO/IEC 7812-1:2017: "Identification cards — Identification of issuers — Part 1: Numbering system".

[21] IETF RFC 3261: "SIP: Session Initiation Protocol".

[22] IETF RFC 3966: "The tel URI for Telephone Numbers".

[23] DMA Technical Report 8350.2: "Department of Defense World Geodetic System 1984, Its Definition and Relationships With Local Geodetic Systems".

[24] ETSI TS 123 501: "5G; System architecture for the 5G System (5GS) (3GPP TS 23.501)".

[25] ETSI TS 133 501: "5G; Security architecture and procedures for 5G System (3GPP TS 33.501)".

[26] IETF RFC 7542: "The Network Access Identifier".

[27] ETSI TS 124 501: "5G; Non-Access-Stratum (NAS) protocol for 5G System (5GS); Stage 3 (3GPP TS 24.501)".

[28] ETSI TS 103 120: "Lawful Interception (LI); Interface for warrant information".

[29] W3C® Recommendation 16 August 2006: "Extensible Markup Language (XML) 1.1 (Second Edition)".

[30] IETF RFC 6530: "Overview and Framework for Internationalized Email".

[31] IETF RFC 9542: "IANA Considerations and IETF Protocol and Documentation Usage for IEEE 802 Parameters".

[32] ETSI TS 102 221: "Smart Cards; UICC-Terminal interface; Physical and logical characteristics".

[33] ETSI TS 129 571: "5G; 5G System; Common Data Types for Service Based Interfaces; Stage 3 (3GPP TS 29.571)".

[34] IANA: "Assigned Internet Protocol Numbers".

[35] IETF Draft draft-bhutton-json-schema-01: "JSON Schema: A Media Type for Describing JSON Documents".

[36] ECMA-262: "ECMAScript® 2023 Language Specification".

[37] IEEE 802.1Q™-2014: "IEEE Standard for Local and metropolitan area networks -- Bridges and Bridged Networks".

[38] ISO 3779:2009: "Road vehicles — Vehicle identification number (VIN) — Content and structure".

[39] GSMA SGP.02: "Remote Provisioning Architecture for Embedded UICC Technical Specification".

[40] IETF RFC 3508: "H.323 Uniform Resource Locator (URL) Scheme Registration".

2.2 Informative references

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

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

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

Not applicable.

3 Definition of terms, symbols and abbreviations

3.1 Terms

Void.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

3GPP	3rd Generation Partnership Project
AGL	Above Ground Level
AMSL	Above Mean Sea Level
APN	Access Point Name
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation One
CC	Content of Communication
CGI	Cell Global Identification
CI	Cell Identity
CIDR	Classless Inter-Domain Routing
CSP	Communications Service Provider
DNN	Data Network Name
ECGI	E-UTRAN Cell Global Identification
ECI	E-UTRAN Cell Identity
EID	eUICC IDentifier
EPV	Estimated Position error in Vertical (altitude)
EPX	Estimated Position error in X (longitude)
EPY	Estimated Position error in Y (latitude)
EUI	Extended Unique Identifier
eUICC	embedded Universal Integrated Circuit Card
E-UTRAN	Evolved Universal Terrestrial Radio Access Network
GML	Geography Markup Language
GNSS	Global Navigation Satellite System
GPSD	GPS Daemon
GPSI	Generic Public Subscription Identifier
HEX	HEXadecimal
HI	Handover Interface
HI1	Handover Interface port 1 (for administrative information)
HI2	Handover Interface port 2 (for Intercept Related Information)

HI3	Handover Interface port 3 (for Content of Communication)
IANA	Internet Assigned Numbers Authority
ICCID	Integrated Circuit Card IDentifier
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IEI	Information Element Identifier
IETF	Internet Engineering Task Force
IMEI	International Mobile station Equipment Identity
IMEISV	International Mobile station Equipment Identity and Software Version number
IMPI	IP Multimedia Private Identity
IMPU	IP Multimedia Public Identity
IMSI	International Mobile Subscriber Identity
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IRI	Intercept Related Information
ISO	International Organization for Standardization
ITU-T	International Telecommunication Union - Telecommunication
JSON	JavaScript Object Notation
LAC	Location Area Code
LDID	Lawful Disclosure IDentifier
LEA	Law Enforcement Agency
LIID	Lawful Interception IDentifier
LTE	Long-Term Evolution
MAC	Media Access Control
MCC	Mobile Country Code
MNC	Mobile Network Code
MSISDN	Mobile Station International Subscriber Directory Number
NAI	Network Access Identifier
NCGI	NR Cell Global Identification
NCI	NR Cell Identity
NR	New Radio
PEI	Permanent Equipment Identifier
RFC	Request For Comments
SIP	Session Initialization Protocol
SMF	Session Management Function
SUCI	Subscription Concealed Identifier
SUPI	Subscription Permanent Identifier
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UPF	User Plane Function
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UTC	Coordinated Universal Time
UUID	Universally Unique IDentifier
VIN	Vehicle Identification Number
VLAN	Virtual Local Area Network
VRF	Virtual Routing and Forwarding
WGS84	World Geodetic System 1984
XML	eXtended Markup Language
XSD	XML Schema Definition

4 Release management

This clause describes the release management requirements. The requirements are:

- The version of the present document is defined as <major>.<minor>.<patch>.
- The major version should be incremented when making a backwards incompatible change.

- The minor version should be incremented when adding backwards compatible functionality.
- The patch version should be incremented when fixing a backwards compatible bug.
- Once a major version has been incremented, the previous major version will be supported for 2 years after publication of the new version. Change requests issued to a version that is no longer supported will need to be issued for the latest supported major version.

5 Parameter requirements

5.0 Introduction

This clause describes the requirements a parameter should comply to in order to be specified as a common parameter.

5.1 Parameter attributes

Name

- The parameter should be assigned a unique name. The naming conventions used are described in clause 5.2.

Description

- A description of the parameter should be provided.

Usage guidance

- If there are circumstances in which additional usage guidance is applicable, use cases may be described in this attribute.

References to other specifications

- If the parameter is specified in another specification (such as an RFC), a reference to that specification shall be provided. If possible, the reference should point to the exact clause in the specification.

EXAMPLE: Specify one or more sample values of the parameter.

Technical means to define and validate the parameter

- If possible, provide a regular expression to specify the value that is accepted by this parameter. Implementations may be required to perform additional validation on the value. The regular expression is defined per clause 5.4. Define the parameter in the XSD [2] in clause 7.1. When converting a regular expression to an XSD [2] pattern, escape any XML [29] markup characters in the regular expression per XML [29], section 2.4 to create a valid XSD [2] pattern.
- Define the parameter in the ASN.1 [3] in clause 7.2.
- Define the parameter in the JSON Schema [35] in clause 7.3. Unless otherwise specified, the JSON definition shall be a translation of the XSD definition, following the translation given in annex C.

5.2 Parameter naming conventions

Allowed characters

- The following character classes are allowed: A-Z, a-z and 0-9.

Camel casing

- The name of the parameter is to be CamelCased, where the first character is uppercased. Any acronyms should be uppercased.

EXAMPLE:

- IPv4Address.
- SIPURI.
- EmailAddress.

5.3 Technology conventions

The used technologies defined in clause 7 may impose requirements that conflict with the requirements in clauses 5.1 and 5.2. In the case of a conflict and in exceptional cases, it is allowed to deviate from the requirements above.

5.4 Regular expression conventions

Regular expressions used for validation shall be limited to the regular expression capabilities supported by both XSD [2] patterns and ECMAScript regular expressions (see ECMA-262 [36], section 22.2.1, as used by JSON Schema [35] patterns).

Given the high disparity of regular expression implementations, the regular expressions should be limited to the following features (inspired by JSON Schema [35], section 6.4):

1) Individual Unicode characters. Unicode characters for XSD [2] need to be encoded using an appropriate XML [29] entity.
2) Character classes: "`[abc]`" (simple character classes), and "`[a-z]`" (range character classes).
3) Negated character classes: "`[^abc]`" (negated simple character classes), and "`[^a-z]`" (negated range character classes).
4) Simple quantifiers: "`*`" (zero or more occurrences), "`+`" (one or more occurrence), and "`?`" (zero or one occurrence).
5) Range quantifiers: "`{n}`" (exactly n occurrences), "`{n,m}`" (between n and m occurrences), and "`{n,}`" (at least n occurrences).
6) Grouping and alternation: "`(`" and "`)`" (simple grouping), and "`|`" (alternation).
7) Simple atoms: "`.`" (any character except new line and line feed).

As XSD [2] patterns are matched to the entire value, regular expressions shall not start with the anchor "`^`" or end with the anchor "`$`". When regular expressions are mapped to a JSON Schema [35] pattern per table C.2, the anchors are required.

6 Parameter dictionary

6.1 LIID

Name	LIID
Description	For each target identity related to an interception measure, the authorized CSP operator shall assign a special Lawful Interception IDentifier (LIID), which has been agreed between the LEA and the CSP. It is used within parameters of all HI interface ports. Using an indirect identification, pointing to a target identity makes it easier to keep the knowledge about a specific interception target limited within the authorized CSP operators and the handling agents at the LEA. The Lawful Interception IDentifier LIID is a component of the CC delivery procedure and of the IRI records. It shall be used within any information exchanged at the Handover Interfaces HI2 and HI3 for identification and correlation purposes. The LIID format shall consist of alphanumeric characters. It might for example, among other information, contain a lawful authorization reference number, and the date, when the lawful authorization was issued. The authorized CSP shall either enter a unique LIID for each target identity of the interception subject or as a national option a single LIID for multiple target identities all pertaining to the same interception subject. EXAMPLE: The interception subject has a telephony service with three telephone numbers. The CSP enters for each telephone number an own LIID, or optionally enters one LIID for all three telephone numbers. If more than one LEA intercepts the same target identity, there shall be unique LIIDs assigned, relating to each LEA.
Usage guidance	The LIID is defined as an OCTET STRING in ASN.1. This means it is possible to use binary octets or ASCII printable characters to express the LIID. To correctly handle this, the parameter accepts both variations.
References	ETSI TS 102 232-1 [1], clause 5.2.2.
Example	`ZZZ123 (ASCII printable LIID)` `46565527098f6bcd4621d373cade4e832627b4f6ff00ff00ff (Binary LIID, represented in HEX)`
Regular expression	`([!-~]{1,25})|([0-9a-f]{26,50})`
XSD	LIID, simpleType
ASN.1	LIID, OCTET STRING (SIZE(1..25))

6.2 UTCDateTime

Name	UTCDateTime
Description	A UTC timestamp with second precision
Usage guidance	-
References	W3C XML Schema Definition Language [2], section 3.3.7
Example	`2015-12-27T13:37:00Z`
Regular expression	`[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}Z`
XSD	UTCDateTime, simpleType
ASN.1	Not defined

6.3 UTCMicrosecondDateTime

Name	UTCMicrosecondDateTime
Description	A UTC timestamp with microsecond precision
Usage guidance	-
References	W3C XML Schema Definition Language [2], section 3.3.7
Example	`2015-12-27T13:37:00.012345Z`
Regular expression	`[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}\.[0-9]{6}Z`
XSD	UTCMicrosecondDateTime, simpleType
ASN.1	Not defined

6.4 QualifiedDateTime

Name	QualifiedDateTime
Description	A timestamp with second precision and timezone qualifier
Usage guidance	-
References	W3C XML Schema Definition Language 1.1 Part 2: Datatypes [2], section 3.3.7
Example XML	`2015-12-27T13:37:00+02:00`
Example ASN.1	`20151227133700+0200`
Regular expression	`[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}(Z|[+-][0-9]{2}:[0-9]{2})`
XSD	QualifiedDateTime, simpleType
ASN.1	QualifiedDateTime, GeneralizedTime Timestamps shall be provided with a timezone qualifier. The fractional part of a second shall not be present. Local time format shall not be used

6.5 QualifiedMicrosecondDateTime

Name	QualifiedMicrosecondDateTime
Description	A timestamp with microsecond precision and timezone qualifier
Usage guidance	-
References	W3C XML Schema Definition Language 1.1 Part 2: Datatypes [2], section 3.3.7
Example XML	`2015-12-27T13:37:00.012345+02:00`
Example ASN.1	`20151227133700.012345+0200`
Regular expression	`[0-9]{4}-[0-9]{2}-[0-9]{2}T[0-9]{2}:[0-9]{2}:[0-9]{2}\.[0-9]{6}(Z|[+-][0-9]{2}:[0-9]{2})`
XSD	QualifiedMicrosecondDateTime, simpleType
ASN.1	QualifiedMicrosecondDateTime, GeneralizedTime Timestamps shall be provided with a timezone qualifier. The fractional part of a second with no more than 6 digits shall be present. Local time format shall not be used.

6.6 InternationalE164

Name	InternationalE164
Description	E.164 Number in fully international format, written as decimal digits
Usage guidance	-
References	Recommendation ITU-T E.164 [4], clause 6
Example	`447700900123`
Regular expression	`[0-9]{1,15}`
XSD	InternationalE164, simpleType
ASN.1	NumericString (SIZE(1..15))

6.7 IMSI

Name	IMSI
Description	International Mobile Subscriber Identity, written as decimal digits
Usage guidance	-
References	Recommendation ITU-T E.212 [5], clause 6.1 3GPP TS 23.003 [6], clauses 2.2 and 2.3
Example	`999999123456789`
Regular expression	`[0-9]{6,15}`
XSD	IMSI, simpleType
ASN.1	NumericString (SIZE(6..15))

6.8 IMEI

Name	IMEI
Description	International Mobile station Equipment Identity, written as decimal digits without the Luhn check digit, annex B of ISO/IEC 7812-1 [20].
Usage guidance	To avoid implementation issues, the IMEI parameter explicitly excludes the Luhn check digit, annex B of ISO/IEC 7812-1 [20]. (See notes 1 and 2)
References	3GPP TS 23.003 [6], clause 6
Example	`00997123456789`
Regular expression	`[0-9]{14}`
XSD	IMEI, simpleType
ASN.1	NumericString (SIZE(14))
NOTE 1: ETSI TS 102 657 [7], clause E.3 identifies potential issues with the inclusion/exclusion of the check digit. As such, the IMEI parameter is explicitly specified without the check digit. NOTE 2: The IMEICheckDigit parameter can be used when the check digit is explicitly required.	

6.9 IMEICheckDigit

Name	IMEICheckDigit
Description	International Mobile station Equipment Identity, written as decimal digits with the Luhn check digit, annex B of ISO/IEC 7812-1 [20].
Usage guidance	-
References	3GPP TS 23.003 [6], clause 6
Example	`009971234567892`
Regular expression	`[0-9]{15}`
XSD	IMEICheckDigit, simpleType
ASN.1	NumericString (SIZE(15))

6.10 IMEISV

Name	IMEISV
Description	International Mobile station Equipment Identity and Software Version Number as defined in 3GPP TS 23.003 [6], clause 6.2.2, written as decimal digits including a software version number instead of a Luhn check digit.
Usage guidance	-
References	3GPP TS 23.003 [6], clause 6.2.2
Example	`0099712345678999`
Regular expression	`[0-9]{16}`
XSD	IMEISV, simpleType
ASN.1	NumericString (SIZE(16))

6.11 IPv4Address

Name	IPv4Address
Description	IPv4 address, written in dotted decimal notation
Usage guidance	The regular expression allows IPv4 addresses with leading zeroes, such as "010.010.010.010". Some programming APIs could incorrectly interpret a number which has a leading zero to be an octal number instead of a decimal number.
References	IETF RFC 791 [8]
Example	`192.0.2.1`
Regular expression	`((25[0-5]|2[0-4][0-9]|[01]?[0-9]?[0-9])\.){3}(25[0-5]|2[0-4][0-9]|[01]?[0-9]?[0-9])`
XSD	IPv4Address, simpleType
ASN.1	Not defined

6.12 IPv4CIDR

Name	IPv4CIDR
Description	IPv4 CIDR, written in dotted decimal notation followed by CIDR notation
Usage guidance	The regular expression allows IPv4 addresses with leading zeroes, such as "010.010.010.010". Some programming APIs could incorrectly interpret a number which has a leading zero to be an octal number instead of a decimal number.
References	IETF RFC 791 [8] and IETF RFC 4632 [9]
Example	`192.0.2.0/24`
Regular expression	`((25[0-5]|2[0-4][0-9]|[01]?[0-9]?[0-9])\.){3}(25[0-5]|2[0-4][0-9]|[01]?[0-9]?[0-9])/([1-2]?[0-9]|3[0-2])`
XSD	IPv4CIDR, simpleType
ASN.1	Not defined

6.13 IPv6Address

Name	IPv6Address
Description	IPv6 address, written as eight groups of four hexadecimal digits separated by a colon.
Usage guidance	It is recognized that IPv6 address formatting has various options. To reduce complexity in technical implementations, the IPv6Address parameter restricts the address to the fully uncompressed representation of the IPv6 address.
References	IETF RFC 8200 [10]
Example	`2001:0db8:0000:0000:0000:0000:0000:0001`
Regular expression	`([0-9a-f]{4}:){7}([0-9a-f]{4})`
XSD	IPv6Address, simpleType
ASN.1	Not defined

6.14 IPv6CIDR

Name	IPv6CIDR
Description	IPv6 CIDR, written as eight groups of four hexadecimal digits separated by a colon followed by CIDR notation.
Usage guidance	See IPv6Address parameter for usage guidance.
References	IETF RFC 8200 [10], IETF RFC 4632 [9] and IETF RFC 4291 [11]
Example	`2001:0db8:0000:0000:0000:0000:0000:0000/48`
Regular expression	`([0-9a-f]{4}:){7}([0-9a-f]{4})/(([1-9][0-9]?)|(1[0-1][0-9])|(12[0-8]))`
XSD	IPv6CIDR, simpleType
ASN.1	Not defined

6.15 IPAddress

Name	IPAddress
Description	Either a IPv4Address parameter or IPv6Address parameter
Usage guidance	-
References	-
Example	XSD <IPAddress> <IPv4Address>192.0.2.1</IPv4Address> </IPAddress> or <IPAddress> <IPv6Address>2001:0db8:0000:0000:0000:0000:0000:0001</IPv6Address> </IPAddress>
Regular expression	-
XSD	IPAddress, complexType
ASN.1	Not defined

6.16 IPCIDR

Name	IPCIDR
Description	Either a IPv4CIDR parameter or IPv6CIDR parameter
Usage guidance	-
References	-
Example	XSD <IPCIDR> <IPv4CIDR>192.0.2.0/24</IPv4CIDR> </IPCIDR> or <IPCIDR> <IPv6CIDR>2001:0db8:0000:0000:0000:0000:0000:0000/48</IPv6CIDR> </IPCIDR>
Regular expression	-
XSD	IPCIDR, complexType
ASN.1	Not defined

6.17 TCPPort

Name	TCPPort
Description	TCP port, written in decimal notation
Usage guidance	-
References	IETF RFC 9293 [12]
Example	22
Regular expression	([1-9][0-9]{0,3}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}| 655[0-2][0-9]|6553[0-5])
XSD	TCPPort, simpleType
ASN.1	TCPPort, INTEGER (1..65535)

6.18 TCPPortRange

Name	TCPPortRange
Description	TCP port range, consists of a 'start' TCPPort parameter and an 'end' TCPPort parameter.
Usage guidance	The start and end values are inclusive, and the end value shall be equal to or greater than the start value.
References	-

Example	Regular expression
	`1024-2048`
	
	XSD
	`<TCPPortRange>`
	`  <start>1024</start>`
	`  <end>2048</end>`
	`</TCPPortRange>`

Regular expression	`([1-9][0-9]{0,3}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])-([1-9][0-9]{0,3}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])`
XSD	TCPPortRange, complexType
ASN.1	TCPPortRange, SEQUENCE

6.19 UDPPort

Name	UDPPort
Description	UDP port, written in decimal notation
Usage guidance	-
References	IETF RFC 768 [13]
Example	53
Regular expression	`([0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])`
XSD	UDPPort, simpleType
ASN.1	UDPPort, INTEGER (0..65535)

6.20 UDPPortRange

Name	UDPPortRange
Description	UDP port range, consists of a 'start' UDPPort parameter and an 'end' UDPPort parameter.
Usage guidance	The start and end values are inclusive, and the end value shall be equal to or greater than the start value.
References	-

Example	Regular expression
	`2048-4096`
	XSD
	`<UDPPortRange>`
	`  <start>2048</start>`
	`  <end>4096</end>`
	`</UDPPortRange>`

Regular expression	`([0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])-([0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])`
XSD	UDPPortRange, complexType
ASN.1	UDPPortRange, SEQUENCE

6.21 Port

Name	Port
Description	Either a TCPPort parameter or a UDPPort parameter
Usage guidance	-
References	-
Example	XSD <Port> <TCPPort>22</TCPPort> </Port>
Regular expression	([0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])
XSD	Port, complexType
ASN.1	Port, CHOICE

6.22 PortRange

Name	PortRange
Description	Either a TCPPortRange parameter or a UDPPortRange parameter
Usage guidance	The start and end values are inclusive.
References	-
Example	XSD <PortRange> <TCPPortRange> <start>2048</start> <end>4096</end> </TCPPortRange> </PortRange>
Regular expression	-
XSD	PortRange, complexType
ASN.1	PortRange, CHOICE

6.23 IPAddressPort

Name	IPAddressPort
Description	Combination of an IPAddress parameter and a Port parameter
Usage guidance	-
References	-
Example	XSD <IPAddressPort> <address> <IPv4Address>192.0.2.1</IPv4Address> </address> <port> <TCPPort>22</TCPPort> </port> </IPAddressPort>
Regular expression	-
XSD	IPAddressPort, complexType
ASN.1	Not defined

6.24 IPAddressPortRange

Name	IPAddressPortRange
Description	Combination of an IPAddress parameter and a PortRange parameter
Usage guidance	-
References	-
Example	XSD <IPAddressPortRange> <address> <IPv4Address>192.0.2.1</IPv4Address> </address> <portRange> <TCPPortRange> <start>2048</start> <end>4096</end> </TCPPortRange> </portRange> </IPAddressPortRange>
Regular expression	-
XSD	IPAddressPortRange, complexType
ASN.1	Not defined

6.25 MACAddress

Name	MACAddress
Description	MAC address, written as six groups of two hexadecimal digits separated by a colon
Usage guidance	-
References	IEEE 802.3 [14]
Example	00:00:5e:00:53:12
Regular expression	([a-f0-9]{2}:){5}[a-f0-9]{2}
XSD	MACAddress, simpleType
ASN.1	Not defined

6.26 EmailAddress

Name	EmailAddress
Description	E-mail address
Usage guidance	-
References	IETF RFC 5322 [15]
Example	john.doe@example.com
Regular expression	[a-zA-Z0-9.!#$%&'*+\/=?^_`{|}~-]+@[a-zA-Z0-9]([a-zA-Z0-9-]{0,61} [a-zA-Z0-9])?(\.[a-zA-Z0-9]([a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)* See note.
XSD	EmailAddress, simpleType
ASN.1	Not defined
NOTE: The regular expression above is sourced from the W3C HTML5 Recommendation [16].	

6.27 UUID

Name	UUID
Description	UUID
Usage guidance	-
References	IETF RFC 4122 [17]
Example	de305d54-75b4-431b-adb2-eb6b9e546013
Regular expression	[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}
XSD	UUID, simpleType
ASN.1	Not defined

6.28 ISOCountryCode

Name	ISOCountryCode
Description	An ISO 3166-1 [18] alpha-2 two-letter country code
Usage guidance	-
References	ISO 3166-1 [18] alpha-2
Example	`"NL"`
Regular expression	`[A-Z]{2}`
XSD	ISOCountryCode, simpleType
ASN.1	Not defined

6.29 ShortString

Name	ShortString
Description	A string with a maximum length of 255 characters
Usage guidance	-
References	-
Example	`string`
Regular expression	-
XSD	ShortString, simpleType
ASN.1	Not defined

6.30 LongString

Name	LongString
Description	A string with a maximum length of 65 535 characters
Usage guidance	-
References	-
Example	`string`
Regular expression	-
XSD	LongString, simpleType
ASN.1	Not defined

6.31 SIPURI

Name	SIPURI
Description	SIP URI
Usage guidance	-
References	IETF RFC 3261 [21], section 19.1
Example	`sip:user@example.com`
Regular expression	`sips?:[a-zA-Z0-9!#$&-;=?-\[\]_~%]+`
XSD	SIPURI, simple type
ASN.1	Not defined

6.32 TELURI

Name	TELURI
Description	TEL URI
Usage guidance	-
References	IETF RFC 3966 [22]
Example	`tel:+447700900000`
Regular expression	`tel:[a-zA-Z0-9!#$&-;=?-\[\]_~%]+`
XSD	TELURI, simple type
ASN.1	Not defined

6.33 WGS84CoordinateDecimal

Name	WGS84CoordinateDecimal
Description	A geographical latitude-longitude coordinate, referring to the WGS84 reference ellipsoid, given in decimal notation.
Usage guidance	-
References	DMA Technical Report 8350.2 [23] (for WGS84 definition itself, not for the syntax defined here)

Example	XSD
	`<WGS84CoordinateDecimal>`
	`  <latitude>N43.616000</latitude>`
	`  <longitude>E007.053000</longitude>`
	`</WGS84CoordinateDecimal>`

Regular expression	-
XSD	WGS84CoordinateDecimal, complexType
ASN.1	WGS84CoordinateDecimal, SEQUENCE

6.34 WGS84LatitudeDecimal

Name	WGS84LatitudeDecimal
Description	A geographical latitude, referring to the WGS84 reference ellipsoid, given in decimal notation.
Usage guidance	The latitude is given as two digits before the decimal point, left-padded with zero where necessary. The latitude is specific to six decimal places, right-padded with zero where necessary.
References	DMA Technical Report 8350.2 [23] (for WGS84 definition itself, not for the syntax defined here)
Example	`N43.616000`
Regular expression	`[NS][0-9]{2}\.[0-9]{6}`
XSD	WGS84LatitudeDecimal, simpleType
ASN.1	WGS84LatitudeDecimal, OCTET STRING (SIZE(10))

6.35 WGS84LongitudeDecimal

Name	WGS84LongitudeDecimal
Description	A geographical longitude, referring to the WGS84 reference ellipsoid, given in decimal notation.
Usage guidance	The longitude is given as three digits before the decimal point, left-padded with zero where necessary. The longitude is specific to six decimal places, right-padded with zero where necessary.
References	DMA Technical Report 8350.2 [23] (for WGS84 definition itself, not for the syntax defined here)
Example	`E007.053000`
Regular expression	`[EW][0-9]{3}\.[0-9]{6}`
XSD	WGS84LongitudeDecimal, simpleType
ASN.1	WGS84LongitudeDecimal, OCTET STRING (SIZE(11))

6.36 WGS84CoordinateAngular

Name	WGS84CoordinateAngular
Description	A geographical latitude-longitude coordinate, referring to the WGS84 reference ellipsoid, given in angular notation.
Usage guidance	-
References	DMA Technical Report 8350.2 [23] (for WGS84 definition itself, not for the syntax defined here)

Example	XSD
	`<WGS84CoordinateAngular>`
	`  <latitude>N433700.62</latitude>`
	`  <longitude>E0070310.42</longitude>`
	`</WGS84CoordinateAngular>`

Regular expression	-
XSD	WGS84CoordinateAngular, complexType
ASN.1	WGS84CoordinateAngular, SEQUENCE

6.37 WGS84LatitudeAngular

Name	WGS84LatitudeAngular
Description	A geographical latitude, referring to the WGS84 reference ellipsoid, given in angular notation.
Usage guidance	Values are specified as "XDDMMSS.ss", i.e. a concatenation of the following fixed-length values, each padded with zeroes where necessary: A one-character hemisphere indicator, "N" or "S". A two-digit value indicating degrees. A two-digit value indicating arc-minutes. A two-digit value indicating whole arc-seconds. A decimal point. A two-digit value indicating fractional arc-seconds.
References	DMA Technical Report 8350.2 [23] (for WGS84 definition itself, not for the syntax defined here)
Example	`N433700.62`
Regular expression	`[NS][0-9]{6}\.[0-9]{2}`
XSD	WGS84LatitudeAngular, simpleType
ASN.1	WGS84LatitudeAngular, OCTET STRING (SIZE(10))

6.38 WGS84LongitudeAngular

Name	WGS84LongitudeAngular
Description	A geographical longitude, referring to the WGS84 reference ellipsoid, given in angular notation.
Usage guidance	Values are specified as "XDDDMMSS.ss" i.e. a concatenation of the following fixed-length values, each padded with zeroes where necessary: A one-character hemisphere indicator, "E" or "W". A three-digit value indicating degrees. A two-digit value indicating arc-minutes. A two-digit value indicating whole arc-seconds. A decimal point. A two-digit value indicating fractional arc-seconds.
References	DMA Technical Report 8350.2 [23] (for WGS84 definition itself, not for the syntax defined here)
Example	`E0070310.42`
Regular expression	`[EW][0-9]{7}\.[0-9]{2}`
XSD	WGS84LongitudeAngular, simpleType
ASN.1	WGS84LongitudeAngular, OCTET STRING (SIZE(11))

6.39 SUPIIMSI

Name	SUPIIMSI
Description	Subscription Permanent Identifier as defined in 3GPP TS 23.501 [24], clause 5.9.2 in IMSI representation.
Usage guidance	In 3GPP Release 15 a SUPI may contain either an IMSI or an NAI, as defined in 3GPP TS 23.501 [24]. This representation is used for a SUPI in IMSI format.
References	3GPP TS 23.501 [24] 3GPP TS 23.003 [6], clause 2.2
Example	See definition of IMSI
Regular expression	
XSD	
ASN.1	

6.40 SUPINAI

Name	SUPINAI
Description	Subscription Permanent Identifier as defined in 3GPP TS 23.501 [24], clause 5.9.2 in NAI representation.
Usage guidance	In 3GPP Release 15 a SUPI may contain either an IMSI or an NAI, as defined in 3GPP TS 23.501 [24]. This representation is used for a SUPI in NAI format.
References	3GPP TS 23.501 [24] IETF RFC 7542 [26]
Example	See definition of NAI
Regular expression	
XSD	
ASN.1	

6.41 SUCI

Name	SUCI
Description	Subscription Concealed Identifier as defined in 3GPP TS 33.501 [25], clause 6.12.2.
Usage guidance	The structure of a SUCI is given in 3GPP TS 23.003 [6], clause 2.2B, and the IE encoding format is given in 3GPP TS 24.501 [27]. When the ASN.1 representation is used, the octets of the SUCI are provided as defined in 3GPP TS 24.501 [27], clause 9.11.3.4, with the 5GS Mobile identity IEI and length fields (i.e. octets 1, 2 and 3 in figure 9.11.3.4.3) omitted. When XSD or string representations are used, the same octets are provided but in hex-binary representation.
References	3GPP TS 33.501 [25] 3GPP TS 24.501 [27]
Example	`-`
Regular expression	`[0-9a-f]+`
XSD	SUCI, simpleType
ASN.1	SUCI, OCTET STRING

6.42 PEIIMEI

Name	PEIIMEI
Description	Permanent Equipment Identifier as defined in 3GPP TS 23.501 [24], clause 5.9.3 in IMEI representation, without the final check/spare digit.
Usage guidance	In 3GPP Release 15 a PEI may contain either an IMEI or an IMEISV as defined in 3GPP TS 23.501 [24]. This representation is used for IMEI format without the final check/spare digit.
References	3GPP TS 23.501 [24], clause 5.9.3 3GPP TS 23.003 [6], clause 6.2.1
Example	See definition of IMEI
Regular expression	
XSD	
ASN.1	

6.43 PEIIMEICheckDigit

Name	PEIIMEICheckDigit
Description	Permanent Equipment Identifier as defined in 3GPP TS 23.501 [24], clause 5.9.3 in IMEI representation with Luhn check digit.
Usage guidance	In 3GPP Release 15 a PEI may contain either an IMEI or an IMEISV as defined in 3GPP TS 23.501 [24]. This representation is used for IMEI format including the Luhn check digit.
References	3GPP TS 23.501 [24], clause 5.9.3 3GPP TS 23.003 [6], clause 6.2.1
Example	See definition of IMEICheckDigit
Regular expression	
XSD	
ASN.1	

6.44 PEIIMEISV

Name	PEIIMEISV
Description	Permanent Equipment Identifier as defined in 3GPP TS 23.501 [24], clause 5.9.3 in IMEISV representation.
Usage guidance	In 3GPP Release 15 a PEI may contain either an IMEI or an IMEISV as defined in 3GPP TS 23.501 [24]. This representation is used for IMEISV format written as decimal digits including a software version number instead of a Luhn check digit.
References	3GPP TS 23.501 [24] 3GPP TS 23.003 [6], clause 6.2.2
Example	See definition of IMEISV
Regular expression	
XSD	
ASN.1	

6.45 GPSIMSISDN

Name	GPSIMSISDN
Description	Generic Public Subscription Identifier as defined in 3GPP TS 23.501 [24], clause 5.9.8, and in 3GPP TS 23.003 [6], clause 28.8, in MSISDN representation.
Usage guidance	A GPSI may contain either an MSISDN, or an External Identifier given as an NAI following the rules given in 3GPP TS 23.003 [6], clause 19.7.2. This representation is used for an MSISDN following the format given in 3GPP TS 23.003 [6], clause 3.3.
References	3GPP TS 23.501 [24] 3GPP TS 23.003 [6]
Example	`447700900000`
Regular expression	`[0-9]{1,15}`
XSD	simpleType
ASN.1	NumericString (SIZE(1..15))

6.46 GPSINAI

Name	GPSINAI
Description	Generic Public Subscription Identifier as defined in 3GPP TS 23.501 [24], clause 5.9.8, and in 3GPP TS 23.003 [6], clause 28.8 in NAI representation.
Usage guidance	A GPSI may contain either an MSISDN, or an External Identifier given as an NAI following the rules given in 3GPP TS 23.003 [6], clause 19.7.2. This representation is used for an NAI following the format given in 3GPP TS 23.006 [6], clause 28.7.
References	3GPP TS 23.501 [24] 3GPP TS 23.003 [6] IETF RFC 7542 [26]
Example	See definition of NAI
Regular expression	
XSD	
ASN.1	

6.47 NAI

Name	NAI
Description	Network Access Identifier following the format given in IETF RFC 7542 [26]
Usage guidance	In general an NAI will take the form "username@realm".
References	IETF RFC 7542 [26]
Example	`user@homerealm.example.net`
Regular expression	-
XSD	NAI, simpleType
ASN.1	NAI, UTF8String

6.48 LDID

Name	LDID
Description	For each Lawful Disclosure request, the LEA shall assign a unique Lawful Disclosure Identifier. A CSP can then use this identifier to uniquely identify a request and when sending a response, include this identifier for correlation on the LEA side. The structure of the LDID is compatible with RequestID parameter as defined in ETSI TS 102 657 [7]. It is defined as a "-"-separated string that contains: 1) A two letter country code as per ISO 3166-1 [18]. 2) A unique identifier that identifies the LEA. 3) A request identifier that is unique within the scope of country code and LEA identifier.
Usage guidance	-
References	ETSI TS 103 120 [28], clause 8.3
Example	`NL-03112345-123`
Regular expression	`[A-Z]{2}-.+-.+`
XSD	LDID, simpleType
ASN.1	-

6.49 InternationalizedEmailAddress

Name	InternationalizedEmailAddress
Description	Internationalized E-mail address
Usage guidance	-
References	IETF RFC 6530 [30]
Example	`Όνομα.παραδείγματος@example.com`
Regular expression	`.+@.+`
XSD	InternationalizedIEmailAddress, simpleType
ASN.1	Not defined

6.50 EUI64

Name	EUI64
Description	EUI64 written as eight groups of two hexadecimal digits separated by a colon
Usage guidance	Colons are being used as separators instead of dashes as used in IETF RFC 9542 [31]. Also, lowercase is used in the expression instead of uppercase.
References	IEEE 802.3 [14] and IETF RFC 9542 [31]
Example	`02:00:5e:10:00:00:00:00`
Regular expression	`([a-f0-9]{2}:){7}[a-f0-9]{2}`
XSD	EUI64, simpleType
ASN.1	Not defined

6.51 CGI

Name	CGI
Description	Cell Global Identification
Usage guidance	MCC (3 digits), MNC (2-3 digits), LAC (16 bits, hex) and CI (16 bits, hex), separated by "-"
References	3GPP TS 29.571 [33], clauses 5.4.2 and 5.4.4.54
Example	`001-01-12ab-34de`
Regular expression	`[0-9]{3}-[0-9]{2,3}-[a-f0-9]{4}-[a-f0-9]{4}`
XSD	CGI
ASN.1	Not defined

6.52 ECGI

Name	ECGI
Description	E-UTRAN Cell Global Identification
Usage guidance	MCC (3 digits), MNC (2-3 digits), ECI (28 bits, hex), separated by "-"
References	3GPP TS 29.571 [33], clauses 5.4.2 and 5.4.4.5
Example	`001-01-5db6007`
Regular expression	`[0-9]{3}-[0-9]{2,3}-[a-f0-9]{7}`
XSD	ECGI
ASN.1	Not defined

6.53 NCGI

Name	NCGI
Description	NR Cell Global Identifier
Usage guidance	MCC (3 digits), MNC (2-3 digits), NCI (36 bits, hex), separated by "-"
References	3GPP TS 29.571 [33], clauses 5.4.2 and 5.4.4.6
Example	`001-01-225db6007`
Regular expression	`[0-9]{3}-[0-9]{2,3}-[a-f0-9]{9}`
XSD	NCGI
ASN.1	Not defined

6.54 ICCID

Name	ICCID
Description	Integrated circuit card identifier
Usage guidance	19 or 20 digits
References	ETSI TS 102 221 [32]
Example	-
Regular expression	`[0-9]{19,20}`
XSD	ICCID
ASN.1	Not defined

6.55 IPProtocol

Name	IPProtocol
Description	Assigned Internet Protocol Number
Usage guidance	0 to 255
References	IANA [34] "Assigned Internet Protocol Numbers" [34]
Example	6
Regular expression	`[01]?[0-9]?[0-9]|2[0-4][0-9]|25[0-5]`
XSD	IPProtocol, simpleType
ASN.1	IPProtocol, INTEGER (0..255)

6.56 VLANID

Name	VLANID
Description	VLAN ID, written in decimal notation
Usage guidance	-
References	IEEE 802.1Q [37]
Example	22
Regular expression	`[0-9]|[0-9]{2}|[0-3]?[0-9]{3}|40[0-8][0-9]|409[0-5]`
XSD	VLANID, simpleType
ASN.1	VLANID, INTEGER (0..4095)

6.57 VIN

Name	VIN
Description	Vehicle Identification Number, given as 17 alphanumeric characters excluding the letters I, O and Q
Usage guidance	-
References	ISO 3779 [38]
Example	`1G9Y817H34LSP7298`
Regular expression	`[A-HJ-NPR-Z0-9]{17}`
XSD	VIN, simpleType
ASN.1	Not defined

6.58 ServiceAccessIdentifier

Name	ServiceAccessIdentifier
Description	Identifies a user within the context of a service
Usage guidance	This parameter should be used to uniquely identify a user of a service.
References	-
Example	`1234567890`
Regular expression	-
XSD	ServiceAccessIdentifier, simpleType
ASN.1	ServiceAccessIdentifier, UTF8String

6.59 EUICCID

Name	EUICCID
Description	Embedded Universal Integrated Circuit Card Identifier
Usage guidance	32 digits
References	GSMA SGP.02 [39], clause 2.2.2
Example	`89049032123451234512345678901235`
Regular expression	`[0-9]{32}`
XSD	EUICCID
ASN.1	Not defined

6.60 APN

Name	APN
Description	String representing an APN as defined in clause 9 of 3GPP TS 23.003 [6]; it shall contain either an APN Network Identifier, or a full APN with both the Network Identifier and Operator Identifier as specified in 3GPP TS 23.003 [6], clauses 9.1.1 and 9.1.2. It shall be coded as string in which the labels are separated by dots (e.g. "Label1.Label2.Label3").
Usage guidance	Identifies the name of a gateway between a mobile network and another computer network, frequently the public Internet.
References	3GPP TS 23.003 [6], clauses 9.1.1 and 9.1.2
Example	`province1.mnc99.mcc999.gprs`
Regular expression	Not defined
XSD	APN, simpleType
ASN.1	Not defined

6.61 DNN

Name	DNN
Description	String representing a DNN as defined in clause 9 of 3GPP TS 23.003 [6]; it shall contain either a DNN Network Identifier, or a full DNN with both the Network Identifier and Operator Identifier as specified in 3GPP TS 23.003 [6], clauses 9.1.1 and 9.1.2. It shall be coded as string in which the labels are separated by dots (e.g. "Label1.Label2.Label3").
Usage guidance	In 5G, the DNN (Data Network Name) is the counterpart of APN in LTE. It is used to identify and route traffic to a specific network slice. The DNN is a reference to a data network, it may be used e.g. to select SMF or UPF.
References	3GPP TS 23.003 [6], clauses 9.1.1 and 9.1.2
Example	`province1.mnc99.mcc999`
Regular expression	Not defined
XSD	DNN, simpleType
ASN.1	Not defined

6.62 H323URI

Name	H323URI
Description	H323 URI
Usage guidance	-
References	IETF RFC 3508 [40]
Example	`h323:user@domain.org`
Regular expression	`h323:[a-zA-Z0-9!#$&-;=?-\[\]_~%]+`
XSD	H323URI, simpleType
ASN.1	Not defined

6.63 IMPU

Name	IMPU
Description	IP Multimedia Public Identity (IMPU), as per 3GPP 23.003 [6], clause 13.4, in the form of a SIP URI or TEL URI.
Usage guidance	-
References	3GPP TS 23.003 [6]
Example	`sip:user@example.com`
Regular expression	`[a-zA-Z0-9!#$&-;=?-\[\]_~%]+`
XSD	IMPU, simpleType
ASN.1	Not defined

6.64 IMPI

Name	IMPI
Description	IP Multimedia Private Identity (IMPI) as defined in 3GPP TS 23.003 [6], clause 13.3, in the form of a Network Access Identifier (NAI).
Usage guidance	-
References	6.47
Example	user@homerealm.example.net
Regular expression	-
XSD	NAI
ASN.1	Not defined

6.65 VRF

Name	VRF
Description	Identifies a VRF (Virtual Routing and Forwarding) instance within the context of a network.
Usage guidance	This parameter should be used to uniquely identify a VRF within a network. A VRF name should be provided.
References	-
Example	VRF-NAME
Regular expression	-
XSD	VRF, simpleType
ASN.1	Not defined

6.66 Percentage

Name	Percentage
Description	A percentage value expressed as an integer between 0 and 100 inclusive. Used to represent confidence levels quality metrics or other percentage-based measurements.
Usage guidance	This parameter represents a percentage value where 0 indicates 0 % and 100 indicates 100 %. When this parameter is used as a confidence metric in location reporting, a value of 0 indicates that the location system has no confidence in the associated location value reported.
References	-
Example	95
Regular expression	([0-9]|[1-9][0-9]|100)
XSD	Percentage, simpleType
ASN.1	Percentage, INTEGER (0..100)

6.67 AltitudeMeters

Name	AltitudeMeters
Description	Altitude measurement expressed in meters as an integer value. This parameter is used to specify the numeric altitude value in either the AGL or AMSL reference system. When using AMSL, the range supports altitudes from 1 000 meters below sea level to 50 000 meters above sea level (stratosphere).
Usage guidance	This parameter represents altitude in meters. When using AMSL, negative values indicate positions below sea level. The maximum value of 50 000 meters accommodates high-altitude vehicles and stratospheric measurements. This parameter is used as a component in the Altitude choice type to specify the numeric altitude value for either altitudeAboveGroundLevel or altitudeAboveMeanSeaLevel reference systems. When used with AGL the value represents height above local ground elevation. When used with AMSL the value represents height above mean sea level.
References	-
Example	1234
Regular expression	-?[0-9]+
XSD	AltitudeMeters, simpleType
ASN.1	AltitudeMeters, INTEGER (-1000..50000)

6.68 Altitude

Name	Altitude
Description	A choice type that specifies altitude measurement with an explicit reference system. Supports two altitude reference systems: altitude Above Ground Level (AGL) and altitude Above Mean Sea Level (AMSL).
Usage guidance	This parameter allows the sender to explicitly indicate which altitude reference system is being used. AGL measurements are relative to the local ground elevation and are commonly used in aviation. AMSL measurements are relative to mean sea level and provide a consistent global reference. The choice of reference system shall match the source of the altitude data.
References	-
Example	XSD <Altitude> <altitudeAboveMeanSeaLevel>1234</altitudeAboveMeanSeaLevel> </Altitude>
Regular expression	-
XSD	Altitude, complexType
ASN.1	Altitude, CHOICE

6.69 UncertaintyMeters

Name	UncertaintyMeters
Description	Uncertainty or error estimate expressed in meters as a positive integer value. Used to represent the estimated accuracy or precision of position measurements.
Usage guidance	This parameter represents the uncertainty radius or error estimate in meters for position measurements. The minimum value is 1 meter. There is no maximum value constraint allowing representation of any uncertainty magnitude. This parameter is used in location shapes (e.g. sphere ellipsoid) to indicate the precision of the reported position.
References	-
Example	6
Regular expression	[1-9][0-9]*
XSD	UncertaintyMeters, simpleType
ASN.1	UncertaintyMeters, INTEGER (1..MAX)

6.70 GNSSLocation

Name	GNSSLocation
Description	A sequence type that represents a complete GNSS location report including both the geographic shape with uncertainty and the fix mode quality indicator.
Usage guidance	This parameter provides a comprehensive GNSS location report suitable for satellite-based positioning systems. The location field contains the geographic shape (point sphere or ellipsoid) with optional confidence percentage. This structure is designed to accommodate rich location data from GNSS receivers including uncertainty metrics and fix quality information.
References	-
Example	XSD <GNSSLocation> <location> <shape> <xyEllipsoid> <centerPoint> <wGS84Point> <location> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </location> <altitude> <altitudeAboveMeanSeaLevel>1234</altitudeAboveMeanSeaLevel> </altitude> </wGS84Point> </centerPoint> <uncertaintyLongitude>6</uncertaintyLongitude> <uncertaintyLatitude>5</uncertaintyLatitude> <uncertaintyAltitude>3</uncertaintyAltitude> </xyEllipsoid> </shape> <confidence>95</confidence> </location> <fixMode>fix3D</fixMode> </GNSSLocation>
Regular expression	-
XSD	GNSSLocation, complexType
ASN.1	GNSSLocation, SEQUENCE

6.71 WGS84Location

Name	WGS84Location
Description	A sequence type that represents a WGS84 coordinate position with optional altitude information. Supports both decimal and angular coordinate formats.
Usage guidance	This parameter provides a WGS84 geographic coordinate (latitude and longitude) with optional altitude. The location field is a choice between decimal format (e.g. N43.616000) and angular format (e.g. N431000.00). The altitude field is optional and when present specifies whether the altitude is above ground level or above mean sea level. This structure is used as the center point in geographic shapes and provides the fundamental coordinate representation.
References	WGS84CoordinateDecimal and WGS84CoordinateAngular parameters
Example	XSD <WGS84Location> <location> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </location> <altitude> <altitudeAboveMeanSeaLevel>1234</altitudeAboveMeanSeaLevel> </altitude> </WGS84Location>
Regular expression	-
XSD	WGS84Location, complexType
ASN.1	WGS84Location, SEQUENCE

6.72 WGS84Coordinate

Name	WGS84Coordinate
Description	A choice type that allows selection between decimal and angular WGS84 coordinate formats for representing latitude and longitude.
Usage guidance	This parameter provides flexibility in coordinate representation by supporting both decimal format (e.g. N43.616000 W108.504000) and angular format (e.g. N431000.00 W1083024.00). Both formats represent the same geographic position using the WGS84 datum.
References	WGS84CoordinateDecimal and WGS84CoordinateAngular parameters
Example	XSD <WGS84Coordinate> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </WGS84Coordinate>
Regular expression	-
XSD	WGS84Coordinate, complexType
ASN.1	WGS84Coordinate, CHOICE

6.73 GeoShape

Name	GeoShape
Description	A sequence type that combines a geographic shape (point sphere or ellipsoid) with an optional confidence percentage. Provides a complete representation of a geographic location with uncertainty and quality metrics.
Usage guidance	This parameter encapsulates a geographic shape along with an optional confidence metric. The shape field specifies the geometric representation (point for precise location sphere for circular uncertainty or ellipsoid for directional uncertainty). The confidence field when present indicates the statistical confidence level (0 - 100 %) associated with the location measurement. This structure enables rich location reporting that includes both spatial uncertainty and measurement quality information.
References	-
Example	XSD <GeoShape> <shape> <xyEllipsoid> <centerPoint> <wGS84Point> <location> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </location> <altitude> <altitudeAboveMeanSeaLevel>1234</altitudeAboveMeanSeaLevel> </altitude> </wGS84Point> </centerPoint> <uncertaintyLongitude>6</uncertaintyLongitude> <uncertaintyLatitude>5</uncertaintyLatitude> <uncertaintyAltitude>3</uncertaintyAltitude> </xyEllipsoid> </shape> <confidence>95</confidence> </GeoShape>
Regular expression	-
XSD	GeoShape, complexType
ASN.1	GeoShape, SEQUENCE

6.74 GMLShape

Name	GMLShape
Description	A choice type inspired by GML that allows selection between different geometric shapes for representing geographic locations: point sphere or XY ellipsoid.
Usage guidance	This parameter provides flexibility in representing geographic locations with varying levels of precision and uncertainty. A point represents a precise location without explicit uncertainty. A sphere represents a location with uniform circular uncertainty (equal in all horizontal directions). An XY ellipsoid represents a location with directional uncertainty that may differ in longitude and latitude directions plus optional vertical uncertainty. The choice of shape should match the characteristics of the positioning system and the available uncertainty information.
References	-
Example	XSD <GMLShape> <xyEllipsoid> <centerPoint> <wGS84Point> <location> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </location> <altitude> <altitudeAboveMeanSeaLevel>1234</altitudeAboveMeanSeaLevel> </altitude> </wGS84Point> </centerPoint> <uncertaintyLongitude>6</uncertaintyLongitude> <uncertaintyLatitude>5</uncertaintyLatitude> <uncertaintyAltitude>3</uncertaintyAltitude> </xyEllipsoid> </GMLShape>
Regular expression	-
XSD	GMLShape, complexType
ASN.1	GMLShape, CHOICE

6.75 GeoPoint

Name	GeoPoint
Description	A choice type that represents a geographic point location. Currently supports WGS84-based point locations with optional altitude.
Usage guidance	This parameter represents a precise geographic point without explicit uncertainty information. The point contains a WGS84 coordinate (latitude and longitude in either decimal or angular format) and optional altitude. This structure is used both as a standalone point shape and as the center point for sphere and ellipsoid shapes. The choice structure allows for future extension to support additional coordinate reference systems.
References	-
Example	<GeoPoint> <wGS84Point> <location> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </location> <altitude> <altitudeAboveMeanSeaLevel>1234</altitudeAboveMeanSeaLevel> </altitude> </wGS84Point> </GeoPoint>
Regular expression	-
XSD	GeoPoint, complexType
ASN.1	GeoPoint, CHOICE

6.76 GeoSphere

Name	GeoSphere
Description	A sequence type that represents a spherical geographic region defined by a center point and a radius. Used to indicate a location with uniform circular uncertainty.
Usage guidance	This parameter represents a geographic location where the uncertainty is equal in all directions forming a circle around the center point. The centerPoint specifies the geographic coordinates (and optional altitude) of the sphere's center. The radiusMeters specifies the uncertainty radius in meters. This shape is appropriate when the positioning system provides a single horizontal uncertainty value or when the uncertainty is known to be approximately equal in all directions.
References	-
Example	XSD <GeoSphere> <centerPoint> <wGS84Point> <location> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </location> </wGS84Point> </centerPoint> <radiusMeters>10</radiusMeters> </GeoSphere>
Regular expression	-
XSD	GeoSphere, complexType
ASN.1	GeoSphere, SEQUENCE

6.77 XYEllipsoid

Name	XYEllipsoid
Description	A sequence type that represents an ellipsoidal geographic region with directional uncertainty in the horizontal plane (longitude and latitude) and optional vertical uncertainty. The ellipsoid is always aligned with the WGS84 coordinate axes (orientation 0).
Usage guidance	This parameter represents a geographic location where the uncertainty differs in the longitude (X) and latitude (Y) directions forming an ellipse in the horizontal plane. The centerPoint specifies the geographic coordinates (and optional altitude) of the ellipsoid's center. The uncertaintyLongitude and uncertaintyLatitude specify the uncertainty in meters along the longitude and latitude axes respectively. The optional uncertaintyAltitude specifies vertical uncertainty in meters. This shape is appropriate for GNSS receivers and other positioning systems that report directional uncertainty values (e.g. EPX EPY EPV from GPSD). The XY prefix indicates the ellipsoid is aligned with coordinate axes rather than rotated.
References	-
Example	XSD <XYEllipsoid> <centerPoint> <wGS84Point> <location> <wGS84CoordinateDecimal> <latitude>N43.616000</latitude> <longitude>W108.504000</longitude> </wGS84CoordinateDecimal> </location> <altitude> <altitudeAboveMeanSeaLevel>1234</altitudeAboveMeanSeaLevel> </altitude> </wGS84Point> </centerPoint> <uncertaintyLongitude>6</uncertaintyLongitude> <uncertaintyLatitude>5</uncertaintyLatitude> <uncertaintyAltitude>3</uncertaintyAltitude> </XYEllipsoid>
Regular expression	-
XSD	XYEllipsoid, complexType
ASN.1	XYEllipsoid, SEQUENCE

6.78 FixMode

Name	FixMode
Description	An enumerated type that indicates the quality and type of GNSS position fix. Specifies whether a valid position fix is available and whether it includes altitude information.
Usage guidance	This parameter indicates the quality of the GNSS position fix. The value 'unknown' (0) indicates the fix mode is not known or not applicable. The value 'noFix' (1) indicates no valid position fix is available. The value 'fix2D' (2) indicates a valid 2-dimensional fix with latitude and longitude but no altitude. The value 'fix3D' (3) indicates a valid 3-dimensional fix with latitude longitude and altitude. This parameter helps receivers understand the reliability and completeness of the position data.
References	-
Example	XSD <FixMode>fix3D</FixMode>
Regular expression	-
XSD	FixMode, simpleType
ASN.1	FixMode, ENUMERATED

7 Technical implementation

7.1 XSD

The XSD definition named "TS_103_280_v021601.xsd" is contained in archive ts_103280v021801p0.zip which accompanies the present document.

The targetNamespace of the XSD is set to 'http://uri.etsi.org/03280/common/2017/07'. The XSD version is set to 2.16.1.

The targetNamespace shall be increased in the event of a major release as defined in clause 4 and the requirement to do so. The year in the targetNamespace shall be set to the year and month of publication of the major release.

The XSD version shall be increased according to the versioning scheme as defined in clause 4. A change to the present document shall not necessarily lead to a new XSD version. The XSD version shall only be increased when a change to the XSD is required, as such the version of the present document and the XSD version may differ.

As the XSD version is not part of the targetNamespace, an implementation should take into account that the appropriate version is used when importing the XSD.

7.2 ASN.1

The ASN.1 definition named "TS_103_280_v020701.asn1" is contained in archive ts_103280v021801p0.zip which accompanies the present document.

The ASN.1 object identifier is defined as itu-t(0) identified-organization(4) etsi(0) common-parameters(3280) version271(271).

The ASN.1 version shall be increased according to the versioning scheme as defined in clause 4. A change to the present document shall not necessarily lead to a new ASN.1 version. The ASN.1 version shall only be increased when a change to the ASN.1 is required, as such the version of the present document and the ASN.1 version may differ.

7.3 JSON

The JSON Schema definition named "TS_103_280_v021601.schema.json" is contained in archive ts_103280v021801p0.zip which accompanies the present document.

Unless otherwise specified, each definition in the JSON Schema shall be the result of transforming the equivalent definition in the XSD schema (see clause 7.1 and annex A) according to the translation rules given in annex C. In the event of a discrepancy then each schema shall be considered authoritative with respect to its relevant encoding.

Annex A (normative): XSD definition

The XSD definition is available as an archive which accompanies the present document (see clause 7.1).

Annex B (normative): ASN.1 definition

The ASN.1 definition is available as an archive which accompanies the present document (see clause 7.2).

Annex C (informative): XSD to JSON schema translation

C.1 Overview

This annex gives a translation for converting XSD schema definitions to JSON Schema definitions [35].

C.2 General translation rules

For specifications which have multiple XSD schemas, each XSD schema is translated into a JSON Schema using the procedures given in this annex.

The ID of a translated JSON Schema is mapped from the relevant XSD namespace. The precise mapping is set by the relevant specification.

C.3 Translation of simple types

C.3.1 Translation rules

If the simple type is a restriction of an XSD native type, then the translation in clause C.3.2 applies.

If the simple type is a restriction of any other simple type, then the translation in clause C.3.3 applies.

C.3.2 Restrictions of XSD native simple types

XSD native simple types are mapped according to table C.1.

Table C.1: Mapping of XSD native simple types

XSD type	JSON Schema definition
string	{"type" : "string"}
normalizedString	{"type" : "string"}
dateTime	{"type" : "string"}
Token	{"type" : "string"}
anyURI	{"type" : "string"}
Integer	{"type" : "integer"}
nonNegativeInteger	{"type" : "integer", "minimum" : 0}
positiveInteger	{"type" : "integer", "minimum" : 1}
Boolean	{"type" : "boolean" }
hexBinary	{"type" : "string", "pattern" : "^([a-fA-F0-9]{2})*$"}
base64Binary	{"type" : "string", "pattern" : "^[A-Za-z0-9+\/]*={0,3}$"}
anyType	{}

If the simple type is a restriction of an XSD native simple type not described in table C.1, the mapping is undefined.

If the simple type has additional facets, they are mapped according to table C.2.

Table C.2: Mapping of facets

XSD facet	JSON Schema definition
<xs:pattern value="{pattern}"/>	"pattern" : "^{pattern}$" See note.
<xs:maxLength="{length}"/>	"maxLength" : {length}
<xs:minLength="{length}"/>	"minLength" : {length}
<xs:minInclusive="{value}"/>	"minimum" : {value}
<xs:minExclusive="{value}"/>	"exclusiveMinimum" : {value}
<xs:maxInclusive="{value}"/>	"maximum" : {value}
<xs:maxExclusive="{value}"/>	"exclusiveMaximum" : {value}
NOTE: The JSON Schema requires the "^" and "$" anchors.	

If the simple type has additional facets not described in table C.2, the mapping is undefined.

C.3.3 Restrictions of other simple types

If the simple type has no additional facets on the base simple type, then the JSON definition is a direct reference to the base simple type.

If the simple type has additional facets, then the JSON Schema definition is an "allOf" union of the following:

- A reference to the base simple type.
- A JSON Schema definition obtained by the mapping of the root XSD native type according to table C.1 and the mapping of each facet according to table C.2.

C.4 Translation of complex types

C.4.1 Translation rules

If the complex type is a sequence, then the rules in clause C.4.2 apply.

If the complex type is a choice, then the rules in clause C.4.3 apply.

C.4.2 Translation of sequences

The "type" field of the JSON Schema definition is set to "object".

The "properties" dictionary of the JSON definition contains an entry for each element in the XSD sequence. For each entry, the key is set to the name of the element in the XSD sequence. If the element is defined in any namespace other than the target namespace of an instance document (i.e. the namespace of the root schema), then the key of the entry in the JSON definition shall include a namespace prefix. The appropriate namespace prefix shall be given by the relevant specification.

NOTE: The namespace prefix is required to enable round-trip translation of instance documents between JSON and XML (see annex D).

The value of that entry is set as follows:

- If the effective maxOccurs fact of the element is greater than 1, the entry is a definition with the "type" field set to "array" and the "items" field set to the appropriate definition (see below). In this case:
  - If the effective maxOccurs value of the element not "unbounded", that value is used to set the "maxItems" field of the definition.
  - If the effective minOccurs value of the element is greater than zero, that value is used to set the "minItems" field of the definition.

- Otherwise, the entry is set to the appropriate definition. If the effective minOccurs value of the element is 1, then the key is added to the "required" list of the JSON Schema definition.
- The appropriate definition is defined as follows:
  - If the type of the element is an XSD native type described in table C.1, then the appropriate definition is the mapping of that native type as described in table C.1.
  - If the type of the element is an XSD native type not described in table C.1, then the appropriate definition is undefined.
  - Otherwise, the appropriate definition is a reference to the JSON Schema definition created by the mapping of the referenced XSD type.

If the sequence contains an anonymous choice group, then the JSON Schema definition is set to an "allOf" union of the following:

- A JSON Schema definition of the non-choice elements, constructed using the rules above.
- A "one of" alternation containing a list of definitions obtained from mapping the inner choice elements according to clause C.4.3.

If the sequence contains more than one anonymous choice group, or if the anonymous choice group is not the last element in the sequence, then the mapping is undefined.

If the sequence contains anything other than elements and at most one anonymous choice group, the mapping of the complex type is undefined.

The "additionalProperties" field of the JSON definition shall be set to "false" unless the definition is an "allOf" union as described above. In this case, the "additionalProperties" field shall be left unset, and instead the "unevaluatedProperties" field shall be set to "false".

If the complex type is itself an extension of another complex type, then the "parameters" dictionary includes mappings of the elements in both that complex type and all ancestor types. The "parameters" dictionary also contains an additional entry with a key of "@xsi:type" and a definition as given in table C.3.

Table C.3: Additional @xsi:type property

Field	Value
type	String
enum	The fully-qualified name of the XSD type referenced in the xsi:type attribute of the complexType definition

C.4.3 Translation of choices

The definition consists of a "oneOf" alternation containing one entry for each choice in the original complexType.

The entry for each element consists of a JSON Schema definition constructed as follows:

- The "type" field is set to "object".
- The "properties" dictionary shall consist of a single key-value pair. The key is set to the name of the element. The value is set to the appropriate definition following the same rules for sequence elements (see clause C.4.2).
- The "required" list consists of a single entry, set to the value of the single key in the "properties" dictionary.
- If the choice contains anything other than elements, the mapping is undefined.

Annex D (normative): Translation of instance documents

D.1 Overview

Annex D gives a translation for converting XML instance documents to JSON and vice-versa.

D.2 Root element translation

The root XML element is translated to a JSON dictionary and vice versa.

NOTE: This means that translating from JSON to XML requires knowing the name of the root XML element in order to reconstruct it, as this information is not present in the JSON document.

D.3 General element translation

Elements that do not fall under the scope of clause D.4 (lists) or clause D.5 (attributes) are translated as follows.

An XML element is translated to an entry in a JSON dictionary, with the key set to the name of the XML element and the value set to the translated value of the XML element. If the content is itself structured, then the translation occurs recursively. Otherwise (i.e. primitive data such as strings or integers) the content is copied across as-is, preserving their type.

NOTE: Implementers should take care to ensure that the type is preserved especially for integer and Boolean values, and that such values are not accidentally translated into string values in JSON documents. Doing so will result in an instance document that fails the translated JSON schema validation.

Conversely, each element of a JSON dictionary is converted to an XML element whose name is set to the key in the JSON dictionary, and whose contents are set to the translation of the value in the JSON dictionary.

D.4 List element translation

List elements are those which are represented in JSON as an array, following the translation rules given in clause C.4.2 above. According to these translation rules, list elements are repeated XML elements which have the same name. The contents of each element are then translated to JSON and given as an element of a JSON array. The JSON array is then given as an element whose key is set to the repeated XML element name.

Conversely, JSON arrays are translated to repeated instances of an XML element, each of which having the name is set to the key of the element that contains the JSON array, and the contents set to the translated value of the JSON array element.

Empty XML lists (i.e. the absence of an XML element whose maxOccurs is greater than 1) are translated into an empty JSON array. Conversely, empty JSON arrays are translated into XML by simply not providing any XML elements.

D.5 Attribute translation

XML attributes are translated to JSON elements. The key of the JSON element is set to the name of the attributed prefixed by a "@" character. The value of the element is set to the value of the attribute. Conversely, JSON elements that begin with the "@" character are translated as attributes of the relevant XML element.

NOTE: This, together with the use of prefixes in the schema translation (see clause C.4.2) allows namespace information to be round-tripped between XML and JSON instance documents.

Annex E (informative): Change history

Status of the present document Dictionary for common parameters		
TC LI approval date	**Version**	**Remarks**
June 2015	V1.1.1	First publication of the TS after approval by ETSI TC LI#39
August 2016	V1.1.2	Revision for a minor editorial correction
January 2017	V1.2.1	Included Change Requests agreed by LI#42: CR001r1, LI(16)P42024r1 (Cat D) Addition of XSD annex to ETSI TS 103 280 CR002r1, LI(16)P420r1 (Cat B) ASN.1 definitions in ETSI TS 103 280
June 2017	V2.1.1	Included Change Requests: CR003r1 (agreed by LI#43), LI(16)P43009r1 (Cat F) Short IMSI CR005 (agreed by LI#45), LI(17)P45025 (Cat B) Addition of SIP URI and TEL URI to common definitions CR006r1 (agreed by LI#45), LI(17)P45026r1 (Cat B) Addition of ASN.1 definitions to ETSI TS 103 280
June 2018	V2.2.1	Included Change Requests: CR007r3 (agreed by LI#48), LI(18)P48008r3 (Cat B) Clarification to UTC time parameters and addition of WGS84 Location Parameters CR008 (agreed by LI#48), LI(18)P48020 (Cat D) Correction of the Regular Expression contained in the Definition of EmailAddress
March 2019	V2.3.1	Included Change Requests: CR009r3 (agreed by LI#50), LI(19)P50011r3 (Cat B) Addition of 5G identifiers to common parameters
October 2019	V2.4.1	Included Change Requests: CR011 (agreed by LI#52), LI(19)P52024 (Cat D) Additional ASN.1 example for QualifiedDateTime and QualifiedMicrosecondDateTime CR012 (agreed by LI#52), LI(19)P52031r1 (Cat B) Addition of LDID
February 2021	V2.5.1	Included Change Requests agreed by ETSI TC LI#56e: CR014r1, LI(21)P56012r1 (Cat F) Regex anchors and section headings CR015r3, LI(21)P56022r3 (Cat F) Correction of SUCI user guidance
June 2021	V2.6.1	Included Change Request agreed by ETSI TC LI#57e: CR016, LI(21)P57015r1 (Cat B) Addition of InternationalizedEmailAddress type
November 2021	V2.7.1	Included Change Requests agreed by ETSI TC LI#58e: CR017, LI(21)P58035r1 (Cat F) Better examples CR018, LI(21)P58024r3 (Cat B) Addition of EUI64
February 2022	V2.8.1	Included Change Request agreed by ETSI TC LI#59e: CR019, LI(22)P59027r2 (Cat B) Addition of CGI, ECGI, NCGI and ICCID
November 2022	V2.9.1	Included Change Request agreed by ETSI TC LI#61: CR022, LI(22)P61048 (Cat B) Addition of IPProtocol parameter
June 2023	V2.10.1	Included Change Request agreed by ETSI TC LI#63: CR023, LI(23)P63033r2 (Cat B) JSON encoding
December 2023	V2.11.1	Included Change Requests agreed by ETSI TC LI#64: CR024, LI(23)P64016r1 (Cat C) Regular expression clarifications and editorial fixes CR025, LI(23)P64017r5 (Cat B) Add VLAN types
February 2024	V2.12.1	Included Change Requests agreed by ETSI TC LI#65: CR026, LI(23)P65017r3 (Cat B) Addition of VIN CR027, LI(23)P65023r1 (Cat C) Adding a service access identifier

Status of the present document Dictionary for common parameters		
TC LI approval date	**Version**	**Remarks**
July 2024	V2.13.1	Included Change Requests agreed by ETSI TC LI#66: CR028, LI(24)P66014r1 (Cat B) Addition of the eUICC identifier (EID) CR029, LI(24)P66021r1 (Cat F) Adding $schema keyword to JSON schema CR030, LI(24)P66038r3 (Cat C) Adding APN and DNN
November 2024	V2.14.1	Included Change Request agreed by ETSI TC LI#67: CR031r1, LI(24)P67024r1 (Cat F) JSON translation clarifications
February 2025	V2.15.1	Included Change Requests agreed by ETSI TC LI#68: CR032, LI(25)P68006r3 (Cat F) Correction of references in TS 103 280 CR033, LI(25)P68008r1 (Cat C) Addition of H323-URI, IMPU and IMPI to the dictionary parameters CR034, LI(25)P68018r1 (Cat C) Closing JSON schemas
June 2025	V2.16.1	Included Change Request agreed by ETSI TC LI#69: CR035, LI(25)P69022 (Cat F) Corrections to regular expression conventions
October 2025	V2.17.1	Included Change Request agreed by ETSI TC LI#70: CR037, LI(25)70050r2 (Cat B) Addition of VRF parameter
January 2026	V2.18.1	Included Change Requests agreed by ETSI TC LI#71: CR038, LI(26)71009r1 (Cat F) IPv4 regex corrections CR040, LI(26)71037r4 (Cat B) Extended WGS84 Coordinate Types

History

Version	Date	Status
V1.1.1	August 2015	Publication
V1.1.2	August 2015	Publication
V1.2.1	August 2016	Publication
V2.1.1	August 2017	Publication
V2.2.1	September 2018	Publication
V2.3.1	April 2019	Publication
V2.4.1	December 2019	Publication
V2.5.1	March 2021	Publication
V2.6.1	July 2021	Publication
V2.7.1	November 2021	Publication
V2.8.1	April 2022	Publication
V2.9.1	December 2022	Publication
V2.10.1	August 2023	Publication
V2.11.1	January 2024	Publication
V2.12.1	April 2024	Publication
V2.13.1	July 2024	Publication
V2.14.1	December 2024	Publication
V2.15.1	May 2025	Publication
V2.16.1	August 2025	Publication
V2.17.1	November 2025	Publication
V2.18.1	March 2026	Publication