INTERNATIONAL STANDARD

ISO/IEC 8825-5

Fourth edition
2021-06

Information technology — ASN.1 encoding rules —

Part 5: Mapping W3C XML schema definitions into ASN.1

Technologies de l'information — Règles de codage ASN.1 —

Partie 5: Mappage en ASN.1 des définitions de schéma XML du W3C

Reference number
ISO/IEC 8825-5:2021(E)

Documents
Document Sharing Hub

© ISO/IEC 2021

Full document details: ISO_IEC 8825-5_2021

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2021

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub

Full document details: ISO_IEC 8825-5_2021

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs)

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems,* in collaboration with ITU-T. The identical text is published as ITU-T X.694 (02/2021).

This fourth edition cancels and replaces the third edition (ISO/IEC 8825-5:2015), which has been technically revised.

A list of all parts in the ISO/IEC 8825 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

Documents
Document Sharing Hub

Full document details: ISO_IEC 8825-5_2021

CONTENTS

		Page
1	Scope	1
2	Normative references	1
	2.1 Identical Recommendations | International Standards	1
	2.2 Additional references	2
3	Definitions	2
	3.1 Imported definitions	2
	3.2 Additional definitions	3
4	Abbreviations	3
5	Notation	3
6	Purpose and extent of standardization	3
7	Mapping XSD Schemas	4
8	Ignored schema components and properties	5
9	ASN.1 modules	6
10	Name conversion	6
	10.1 General	6
	10.2 Generating ASN.1 type definitions that are references to ASN.1 type assignments	7
	10.3 Generating identifiers and type reference names	7
	10.4 Order of the mapping	9
11	Mapping uses of XSD built-in types	10
12	Mapping facets	10
	12.1 The length, minLength, and maxLength facets	11
	12.2 The pattern facet	11
	12.3 The whiteSpace facet	11
	12.4 The enumeration facet	12
	12.5 Other facets	14
13	Mapping simple type definitions	14
14	Mapping element declarations	16
15	Mapping attribute declarations	17
16	Mapping values of simple type definitions	17
17	Mapping model group definitions	17
18	Mapping model groups	17
19	Mapping particles	18
20	Mapping complex type definitions	19
21	Mapping wildcards	20
22	Mapping attribute uses	22
23	Mapping uses of simple and complex type definitions (general case)	22
24	Mapping special uses of simple and complex type definitions (substitutable)	23
25	Mapping special uses of simple and complex type definitions (substitutable, nillable)	24
26	Mapping special uses of simple type definitions (nillable)	25
27	Mapping special uses of complex type definitions (nillable)	26
28	Mapping special uses of element declarations (head of element substitution group)	27
29	Generating special ASN.1 type assignments for types used in element declarations	27
30	Generating special ASN.1 type assignments for types belonging to a derivation hierarchy	29
31	Generating special ASN.1 type assignments for element substitution groups	29
Annex A – ASN.1 type definitions corresponding to XSD built-in types for the version 1 mapping		30
Annex B – ASN.1 type definitions corresponding to XSD built-in types for the version 2 mapping		33

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

Page

Annex C – Identification of the XSD module 37
Annex D – Examples of mappings 38
D.1 A Schema using simple type definitions 38
D.2 The corresponding ASN.1 definitions 39
D.3 Further examples 40
Annex E – Use of the mapping to provide binary encodings for W3C XML Schema 60
E.1 Encoding XSD Schemas 60
E.2 Transfer without using the XSD Schema for Schemas 60
E.3 Transfer using the XSD Schema for Schemas 60

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

Introduction

This Recommendation | International Standard specifies version 1 and version 2 of a mapping from a W3C XML Schema definition (an XSD Schema) into an Abstract Syntax Notation One (ASN.1) schema. The mappings can be applied to any XSD Schema. Both mappings specify the generation of one or more ASN.1 modules containing type definitions, together with ASN.1 XER encoding instructions. These are jointly described as an ASN.1 schema for Extensible Markup Language (XML) documents. This ASN.1 schema (produced by either version of the mapping), when used with the ASN.1 Extended XML Encoding Rules (EXTENDED-XER), can be used to generate and to validate the same set of W3C XML 1.0 documents as the original XSD Schema. The resulting ASN.1 types and encodings support the same semantic content as the XSD Schema. Thus ASN.1 tools can be used interchangeably with XSD tools for the generation and processing of the specified XML documents.

Other standardized ASN.1 encoding rules, such as the Distinguished Encoding Rules (DER) or the Packed Encoding Rules (PER), can be used in conjunction with this standardized mapping, but produce encodings for version 2 of the mapping that differ from (and are less verbose than) those produced by version 1 for XSD constructs involving dates and times or wildcards.

The combination of this Recommendation | International Standard with ASN.1 Encoding Rules provides fully standardized and vendor-independent compact and canonical binary encodings for data originally defined using an XSD Schema.

The ASN.1 schema provides a clear separation between the specification of the information content of messages (their abstract syntax) and the precise form of the XML document (e.g., use of attributes instead of elements). This results in both a clearer and generally a less verbose schema than the original XSD Schema.

Annex A forms an integral part of this Recommendation | International Standard, and is an ASN.1 module containing a set of ASN.1 type assignments that correspond to each of the XSD built-in types for version 1 of the mapping. Mappings of XSD Schemas into ASN.1 schemas either import the type reference names of those type assignments or include the type definitions in-line.

Annex B also forms an integral part of this Recommendation | International Standard and provides the ASN.1 module for version 2 of the mapping.

Annex C does not form an integral part of this Recommendation | International Standard, and summarizes the object identifier, OID internationalized resource identifier and object descriptor values assigned in this Recommendation | International Standard.

Annex D does not form an integral part of this Recommendation | International Standard, and gives examples of the mapping of XSD Schemas into ASN.1 schemas.

Annex E does not form an integral part of this Recommendation | International Standard, and describes the use of the mapping defined in this Recommendation | International Standard, in conjunction with standardized ASN.1 Encoding Rules, to provide compact and canonical encodings for data defined using an XSD Schema.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

Documents
Document Sharing Hub

Full document details: ISO_IEC 8825-5_2021

INTERNATIONAL STANDARD
ITU-T RECOMMENDATION

Information technology – ASN.1 encoding rules: Mapping W3C XML schema definitions into ASN.1

1 Scope

This Recommendation | International Standard specifies two versions of a mapping from any XSD Schema into an Abstract Syntax Notation One (ASN.1) schema. The ASN.1 schema for both versions support the same semantics and validate the same set of XML documents.

This Recommendation | International Standard specifies the final XER encoding instructions that are to be applied as part of the defined mapping to ASN.1 types, but does not specify which syntactic form is to be used for the specification of those final XER encoding instructions, or the order or manner of their assignment.

NOTE – Implementers of tools generating these mappings may choose any syntactic form or order of assignment that results in the specified final XER encoding instructions being applied. Examples in this Recommendation | International Standard generally use the type prefix form, but use of an XER Encoding Control Section may be preferred for the mapping of a complete XSD Schema, as a matter of style.

There are different ways (syntactically) of assigning XER encoding instructions for use in EXTENDED-XER encodings (e.g., use of ASN.1 type prefix encoding instructions or use of an XER encoding control section). The choice of these syntactic forms is a matter of style and lies outside the scope of this Recommendation | International Standard.

2 Normative references

The following Recommendations | International Standards and W3C specifications contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations, International Standards and W3C specifications are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations, International Standards and W3C specifications listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations. The W3C maintains a list of currently valid W3C specifications. The reference to a document within this Recommendation | International Standard does not give it, as a stand-alone document, the status of a Recommendation or International Standard.

2.1 Identical Recommendations | International Standards

NOTE – The complete set of ASN.1 Recommendations | International Standards is listed in this clause, as these documents can all be applicable in particular uses of this Recommendation | International Standard. Where these are not directly referenced in the body of this Recommendation | International Standard, a † symbol is added to the reference.

- Recommendation ITU-T X.680 (2021) |ISO/IEC 8824-1:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*
- Recommendation ITU-T X.681 (2021) | ISO/IEC 8824-2:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*.
- Recommendation ITU-T X.682 (2021) | ISO/IEC 8824-3:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification*.
- Recommendation ITU-T X.683 (2021) | ISO/IEC 8824-4:2021, *Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications*.
- Recommendation ITU-T X.690 (2021) | ISO/IEC 8825-1:2021, *Information technology – ASN.1 encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*.
- Recommendation ITU-T X.691 (2021) | ISO/IEC 8825-2:2021, *Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)*.
- Recommendation ITU-T X.692 (2021) | ISO/IEC 8825-3:2021, *Information technology – ASN.1 encoding rules: Specification of Encoding Control Notation (ECN)*.
- Recommendation ITU-T X.693 (2021) | ISO/IEC 8825-4:2021, *Information technology – ASN.1 encoding rules: XML Encoding Rules (XER)*.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

– Recommendation ITU-T X.891 (2005) | ISO/IEC 24824-1:2007, *Information technology – Generic applications of ASN.1: Fast infoset.*

NOTE – The references above shall be interpreted as references to the identified Recommendations | International Standards together with all their published amendments and technical corrigenda.

2.2 Additional references

– ISO 8601:2019, *Date and time – Representation for information interchange – Part 1: Basic rules*.

– W3C XML 1.0:2008, *Extensible Markup Language (XML) 1.0* (Fifth Edition), W3C Recommendation, Copyright © [26 November 2008] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xml/.

– W3C XML Namespaces:1999, *Namespaces in XML*, W3C Recommendation, Copyright © [14 January 1999] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/1999/REC-xml-names-19990114.

– W3C XML Information Set:2004, XML *Information Set* (Second Edition), W3C Recommendation, Copyright © [4 February 2004] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xml-infoset/.

– W3C XML Schema:2004, *XML Schema Part 1: Structures* (Second Edition), W3C Recommendation, Copyright © [28 October 2004] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xmlschema-1/.

– W3C XML Schema:2004, *XML Schema Part 2: Datatypes* (Second Edition), W3C Recommendation, Copyright © [28 October 2004] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xmlschema-2/.

NOTE – When the reference "W3C XML Schema" is used in this Recommendation | International Standard, it refers to W3C XML Schema Part 1 and W3C XML Schema Part 2.

– IETF RFC 2396 (1998), *Uniform resource identifiers (URI): Generic syntax*.

– IETF RFC 3066 (2001), *Tags for the Identification of Languages*.

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply.

3.1 Imported definitions

3.1.1 This Recommendation | International Standard uses the terms defined in Rec. ITU-T X.680 | ISO/IEC 8824-1 and in Rec. ITU-T X.693 | ISO/IEC 8825-4.

NOTE – In particular, the terms "final encoding instructions", "type prefix" and "XER encoding control section" are defined in the Recommendations | International Standards mentioned in this clause.

3.1.2 This Recommendation | International Standard also uses the terms defined in W3C XML Schema and W3C XML Information Set.

NOTE 1 – It is believed that these terms do not conflict with the terms referenced in 3.1.1. If such a conflict occurs, the definition of the term in 3.1.1 applies.

NOTE 2 – In particular, the terms "schema component" is defined in W3C XML Schema, and the terms "element information item" and "attribute information item" are defined in W3C XML Information Set.

NOTE 3 – The terms "top-level **simple type definition**" and "top-level **complex type definition**" do not include XSD built-in types, when used in this Recommendation | International Standard.

© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

3.2 Additional definitions

For the purposes of this Recommendation | International Standard, the following additional definitions apply:

3.2.1 **XSD namespace**: A namespace with a uniform resource identifier of "http://www.w3.org/2001/XMLSchema".

3.2.2 **XSI namespace**: A namespace with a uniform resource identifier of "http://www.w3.org/2001/XMLSchema-instance".

3.2.3 **XML namespace**: A namespace with a uniform resource identifier of "http://www.w3.org/XML/1998/namespace".

4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply:

ASN.1	Abstract Syntax Notation One
BER	(ASN.1) Basic Encoding Rules
DER	(ASN.1) Distinguished Encoding Rules
OID	Object Identifier
PER	(ASN.1) Packed Encoding Rules
URI	(IETF) Uniform Resource Identifier
XER	(ASN.1) XML Encoding Rules
XML	(W3C) Extensible Markup Language
XSD	(W3C) XML Schema

5 Notation

5.1 This Recommendation | International Standard references the notation defined by Rec. ITU-T X.680 | ISO/IEC 8824-1, Rec. ITU-T X.682 | ISO/IEC 8824-3, W3C XML 1.0 and W3C XML Schema.

5.2 When it is necessary in the body of this Recommendation | International Standard to specify, either formally or in examples, the assignment of XER encoding instructions, the type prefix notation is generally used (but see 6.3 and 6.4). In Annex A, an XER encoding control section is used.

5.3 In this Recommendation | International Standard, **`bold Courier`** is used for ASN.1 notation and **bold Arial** is used for XSD notation and for XSD terms and concepts.

5.4 The XSD Schemas used in the examples in this Recommendation | International Standard use the prefix **xsd:** to identify the XSD namespace.

6 Purpose and extent of standardization

6.1 The mapping to ASN.1 that is specified in this Recommendation | International Standard ensures that:

a) any resulting ASN.1 modules generated by tools conforming to this Recommendation | International Standard (from the same XSD Schema) define the same (structured) abstract values;

b) all BASIC-XER, CXER, EXTENDED-XER, and binary encodings of that resulting ASN.1 specification will produce the same encodings (subject to encoder's options); and

c) all XML documents that conform to the source XSD Schema are valid EXTENDED-XER encodings of abstract values of that ASN.1 specification.

6.2 There are many aspects of an ASN.1 definition (such as the use of white-space, or of encoding control sections or type prefixes) that affect neither the abstract values being defined nor the XER or binary encodings of those values. Such aspects of the ASN.1 definition are generally not standardized in this Recommendation | International Standard.

6.3 There are many different ways in ASN.1 of assigning an XER encoding instruction to a type, including:

a) use of a type prefix for every encoding instruction to be assigned; or

b) use of an encoding control section, with a separate encoding instruction for each required assignment; or

c) use of an encoding control section, with a single encoding instruction making a global assignment, possibly supplemented by use of a negating encoding instruction for specific types.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

6.4 This Recommendation | International Standard specifies when a final XER encoding instruction shall be present, and uses the syntax of 6.3 a) in most of its examples. However, the use of the different options in 6.3 is not standardized, and conforming implementations of the mapping may choose any syntactic form (or a mixture of syntactic forms) for the assignment of final XER encoding instructions.

NOTE – The choice among these options does not affect the final binary or XML encodings.

6.5 A formal specification of the required mapping is not provided.

6.6 This Recommendation | International Standard is concerned only with the mapping of XSD Schemas that conform to W3C XML Schema.

NOTE – Such conformance can be either by the provision of one or more W3C XSD schema documents or by other means as specified in W3C XML Schema.

7 Mapping XSD Schemas

7.1 A mapping is based on a source XSD Schema, which is a set of schema components (see W3C XML Schema Part 1, 2.2). No particular representation of schema components or sets of schema components is required or assumed for the mapping, although it is expected that the source XSD Schema will usually be provided as one or more XML schema documents (see W3C XML Schema Part 1, 3.15.2).

NOTE 1 – The schema components represented in multiple XML schema documents become part of the same XSD Schema through the use of the **xsd:include, xsd:redefine,** and **xsd:import** element information items.

NOTE 2 – Since the mapping is defined in terms of schema components (and not in terms of their XML representation), it is not affected by details of the XML representation, such as the use of multiple schema documents linked by **xsd:include** and **xsd:redefine** element information items, the placement of element information items in one or another schema documents, the order of **xsd:attribute** element information items within a **xsd:complexType** element information item, and so on.

NOTE 3 – Two sets of schema documents that differ in many aspects but represent the same set of schema components generate the same set of ASN.1 type assignments, with the same final encoding instructions assigned to them and to their components to any depth.

7.2 The source XSD Schema shall meet all the constraints imposed by the XSD specification. If the source XSD Schema is represented (in part or all) as a set of XML schema documents, each schema document shall be valid according to the XSD Schema for Schemas (see W3C XML Schema Part 1, Appendix A).

7.3 One or more ASN.1 modules shall be generated for a source XSD Schema. The number of ASN.1 modules generated is an implementation option. Each ASN.1 module shall contain zero or more type assignments corresponding to top-level schema components (see 7.6), and zero or more special ASN.1 type assignments (see clauses 29, 30, and 31). The physical order of type assignments within each ASN.1 module is an implementation option. When multiple ASN.1 modules are generated, the way the generated type assignments are distributed across those ASN.1 modules is also an implementation option.

NOTE 1 – The inclusion in the same ASN.1 module of type assignments generated from XSD schema components with different **target namespace**s is permitted by this clause but not recommended. The preferred mapping is to generate one ASN.1 module per namespace whenever possible. It is also recommended that each special ASN.1 type assignment be inserted in the same ASN.1 module as its associated ASN.1 type assignment (see 29.5, 30.4, and 31.4).

NOTE 2 – The generation of ASN.1 type assignments (see 7.6 and 10.4) is not affected by the number of ASN.1 modules being generated (except for the possible use of "ExternalTypeReference" as specified in 10.2.2), nor by the way the generated type assignments are distributed across those modules, nor by the physical order of the type assignments within each module. In particular, the type reference names of those type assignments are the same whatever mapping style is used by the implementation.

NOTE 3 – A full description of the relationship between the namespace concept of W3C XML Namespaces and naming in ASN.1 is provided in Rec. ITU-T X.693 | ISO/IEC 8825-4, clause 16. Type reference names and identifiers defined in an ASN.1 module are assigned a namespace by means of a `NAMESPACE` encoding instruction, and otherwise do not have a namespace. The mapping generates `NAMESPACE` encoding instructions where needed.

7.4 All ASN.1 modules generated by the mapping shall contain (in the XER encoding control section) a **`GLOBAL-DEFAULTS MODIFIED-ENCODINGS`** encoding instruction and a **`GLOBAL-DEFAULTS CONTROL-NAMESPACE`** encoding instruction specifying the XSI namespace.

7.5 A source XSD Schema shall be processed as follows:

a) for each top-level **element declaration**, an ASN.1 type assignment shall be generated by applying clause 14 to the **element declaration**;

b) for each top-level **attribute declaration**, an ASN.1 type assignment shall be generated by applying clause 15 to the **attribute declaration**;

c) for each top-level **simple type definition**, an ASN.1 type assignment shall be generated by applying clause 13 to the **simple type definition**;

© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

d) for each top-level **complex type definition**, an ASN.1 type assignment shall be generated by applying clause 20 to the **complex type definition**;

e) for each **model group definition** whose **model group** has a **compositor** of **sequence** or **choice**, an ASN.1 type assignment shall be generated by applying clause 17 to the **model group definition**.

NOTE 1 – The remaining schema components of the source XSD schema will be processed as a result of mapping these schema components.

NOTE 2 – The order in which schema components are to be mapped is specified in 10.4. The order of the items of the foregoing list has no significance for the mapping.

7.6 Column 1 of Table 1 lists schema components. Column 2 gives the reference to the clause in W3C XML Schema that defines the schema component. Column 3 lists the clause that defines the mapping of those schema components into ASN.1.

Table 1 – Mapping of XSD schema components

XSD schema component	W3C XML Schema reference	Mapping defined by
attribute declaration	Part 1, 3.2	Clause 15
element declaration	Part 1, 3.3	Clause 14
complex type definition	Part 1, 3.4	Clause 20
attribute use	Part 1, 3.5	Clause 22
attribute group definition	Part 1, 3.6	*not mapped as such*
model group definition	Part 1, 3.7	Clause 17
model group	Part 1, 3.8	Clause 18
particle	Part 1, 3.9	Clause 19
wildcard	Part 1, 3.10	Clause 21
identity-constraint definition	Part 1, 3.11	*ignored by the mapping*
notation declaration	Part 1, 3.12	*ignored by the mapping*
annotation	Part 1, 3.13	*ignored by the mapping*
simple type definition	Part 1, 3.14	Clauses 11, 13
schema	Part 1, 3.15	Clause 9
ordered	Part 2, 4.2.2.1	*ignored by the mapping*
bounded	Part 2, 4.2.3.1	*ignored by the mapping*
cardinality	Part 2, 4.2.4.1	*ignored by the mapping*
numeric	Part 2, 4.2.5.1	*ignored by the mapping*
length	Part 2, 4.3.1.1	Clause 12
minLength	Part 2, 4.3.2.1	Clause 12
maxLength	Part 2, 4.3.3.1	Clause 12
pattern	Part 2, 4.3.4.1	Clause 12
enumeration	Part 2, 4.3.5.1	Clause 12
whiteSpace	Part 2, 4.3.6.1	Clause 12
maxInclusive	Part 2, 4.3.7.1	Clause 12
maxExclusive	Part 2, 4.3.8.1	Clause 12
minExclusive	Part 2, 4.3.9.1	Clause 12
minInclusive	Part 2, 4.3.10.1	Clause 12
totalDigits	Part 2, 4.3.11.1	Clause 12
fractionDigits	Part 2, 4.3.12.1	Clause 12

8 Ignored schema components and properties

8.1 The mapping shall ignore the schema components and properties that are listed in this clause.

8.2 All **annotation**s (see W3C XML Schema Part 1, 3.13) shall be ignored.

NOTE – All attribute information items in a schema document with names qualified with namespaces other than the XSD namespace (see W3C XML Schema Part 1, 3.13.1) are a property of **annotation**s, and are ignored.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

8.3 All **identity-constraint definition**s (see W3C XML Schema Part 1, 3.11) shall be ignored.

NOTE – The **identity-constraint definition** provides mechanisms for specifying referential constraints that can be required in a valid instance. ASN.1 currently has no concept of such constraints, and such constraints cannot be mapped into a formal ASN.1 specification, but they may be included as normative comments that are binding on an application implementation.

8.4 All **notation declaration**s (see W3C XML Schema Part 1, 3.12) shall be ignored.

8.5 All schema components that are the **fundamental facet**s (**ordered**, **bounded**, **cardinality**, **numeric**) of **simple type definition**s (see W3C XML Schema Part 2, 4.2) shall be ignored.

8.6 The properties **identity-constraint definition**s, **substitution group exclusion**s and **disallowed substitution**s of **element declaration**s shall be ignored.

8.7 The properties **final**, **abstract**, and **prohibited substitutions** of **complex type definition**s shall be ignored.

8.8 The property **process contents** of **wildcard**s shall be ignored.

NOTE – There is no support in ASN.1 for any action other than **skip**.

8.9 The properties **fundamental facet**s and **final** of **simple type definition**s shall be ignored.

8.10 All **value constraint**s that are present on any **element declaration**s or **attribute declaration**s whose **type definition** is either **xsd:QName** or a **simple type definition** derived from **xsd:QName** or **xsd:NOTATION** shall be ignored.

8.11 All **attribute group definition**s shall be ignored.

NOTE – The **attribute use**s in an **attribute group definition** become part of the **attribute use**s of the **complex type definition**s whose XML representation contains a reference to the **attribute group definition**.

9 ASN.1 modules

9.1 The mapping of an XSD Schema generates one or more ASN.1 modules (see 7.3).

9.2 The ASN.1 "ModuleIdentifier" (see Rec. ITU-T X.680 | ISO/IEC 8824-1, clause 13) to be generated by the mapping is not standardized. Where **`IMPORTS`** statements are used, the ASN.1 module names and module identifiers in the **`IMPORTS`** statements shall be those generated for the ASN.1 modules generated by the mapping.

NOTE – The choice of "ModuleIdentifier" does not affect the encodings in any of the standard encoding rules.

9.3 The ASN.1 modules shall have a "TagDefault" of **`AUTOMATIC TAGS`**.

9.4 In each ASN.1 module generated by a version 1 mapping, there shall be an ASN.1 **`IMPORTS`** statement importing the ASN.1 type reference names in the module named **`XSD {joint-iso-itu-t asn1(1) specification(0) modules(0) xsd-module(2) version1(1)}`** specified in Annex A that are referenced in the ASN.1 module.

9.5 In each ASN.1 module generated by a Version 2 mapping, there shall be an ASN.1 **`IMPORTS`** statement importing the ASN.1 type reference names in the module named **`XSD {joint-iso-itu-t asn1(1) specification(0) modules(0) xsd-module(2) version2(2)}`** specified in Annex B that are referenced in the generated ASN.1 module.

NOTE – The term "XSD module" in this Recommendation | International Standard refers to the module defined in Annex A (version 1 mapping) or in Annex B (version 2 mapping), according to the version of the mapping.

9.6 The **`IMPORTS`** statement shall also import the ASN.1 type reference names of type assignments that have been placed (as a result of the mapping) in other ASN.1 modules but are referenced in this ASN.1 module.

9.7 There shall be no **`EXPORTS`** statement.

NOTE – This means that all ASN.1 type reference names in the ASN.1 module can be imported into other modules.

10 Name conversion

10.1 General

10.1.1 This Recommendation | International Standard specifies the generation of:

a) ASN.1 type reference names corresponding to the **name**s of **model group definition**s, top-level **element declaration**s, top-level **attribute declaration**s, top-level **complex type definition**s, and top-level **simple type definition**s;

b) ASN.1 identifiers corresponding to the **name**s of top-level **element declaration**s, top-level **attribute declaration**s, local **element declaration**s, and local **attribute declaration**s;

© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

c) ASN.1 identifiers for the mapping of certain **simple type definition**s with an **enumeration** facet (see 12.4.1 and 12.4.2);

d) ASN.1 type reference names of special type assignments (see clauses 29, 30, and 31); and

e) ASN.1 identifiers of certain sequence components introduced by the mapping (see clause 20).

10.1.2 All of these ASN.1 names are generated by applying 10.3 either to the **name** of the corresponding schema component, or to a member of the **value** of an **enumeration** facet, or to a specified character string, as specified in the relevant clauses of this Recommendation | International Standard.

10.2 Generating ASN.1 type definitions that are references to ASN.1 type assignments

10.2.1 This subclause applies as explicitly invoked by other clauses of this Recommendation | International Standard to generate an ASN.1 type (a "DefinedType") definition that is a reference to an ASN.1 type assignment.

10.2.2 If a "DefinedType" is to be inserted in an ASN.1 module (M, say) other than the ASN.1 module where the referenced ASN.1 type assignment is being inserted, then the "DefinedType" shall be either a "typereference" or an "ExternalTypeReference" for that type assignment, as an implementation option. Otherwise, it shall be a "typereference" for that type assignment.

NOTE – All ASN.1 "typereference"s created by the mapping are unique for any legal input schema, so a type defined in another ASN.1 module does not need to be an "ExternalTypeReference".

10.3 Generating identifiers and type reference names

10.3.1 This subclause applies as explicitly invoked by other clauses of this Recommendation | International Standard to generate an ASN.1 type reference name or identifier.

10.3.2 **Name**s of **attribute declaration**s, **element declaration**s, **model group definition**s, top-level **simple type definition**s, and top-level **complex type definition**s can be identical to ASN.1 reserved words or can contain characters not allowed in ASN.1 identifiers or in ASN.1 type reference names. In addition, there are cases in which ASN.1 names are required to be distinct where the **name**s of the corresponding XSD schema components (from which the ASN.1 names are mapped) are allowed to be identical.

10.3.3 The following transformations shall be applied, in order, to each character string being mapped to an ASN.1 name, where each transformation (except the first) is applied to the result of the previous transformation:

– the characters " " (SPACE), "`.`" (FULL STOP), and "_" (LOW LINE) shall all be replaced by a "-" (HYPHEN-MINUS); and

– any character except "A" to "Z" (LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER Z), "a" to "z" (LATIN SMALL LETTER A to LATIN SMALL LETTER Z), "0" to "9" (DIGIT ZERO to DIGIT NINE), and "-" (HYPHEN-MINUS) shall be removed; and

– a sequence of two or more HYPHEN-MINUS characters shall be replaced with a single HYPHEN-MINUS; and

– HYPHEN-MINUS characters occurring at the beginning or at the end of the name shall be removed; and

– if a character string that is to be used as a type reference name starts with a lower-case letter, the first letter shall be capitalized (converted to upper-case); if it starts with a digit (DIGIT ZERO to DIGIT NINE), it shall be prefixed with an "**X**" (LATIN CAPITAL LETTER X) character; and

– if a character string that is to be used as an identifier starts with an upper-case letter, the first letter shall be uncapitalized (converted to lower-case); if it starts with a digit (DIGIT ZERO to DIGIT NINE), it shall be prefixed with an "**x**" (LATIN SMALL LETTER X) character; and

– if a character string that is to be used as a type reference name is empty, it shall be replaced by "**X**" (LATIN CAPITAL LETTER X); and

– if a character string that is to be used as an identifier is empty, it shall be replaced by "**x**" (LATIN SMALL LETTER X).

10.3.4 Depending on the kind of name being generated, one of the three following subclauses applies.

10.3.4.1 If the name being generated is the type reference name of an ASN.1 type assignment and the character string generated by 10.3.3 is identical to:

a) the type reference name of another ASN.1 type assignment previously (see 10.4) generated by the mapping (in any ASN.1 module); or

b) the type reference name of a type assignment in the **XSD** module (see Annex A); or

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

c) one of the reserved words specified in Rec. ITU-T X.680 | ISO/IEC 8824-1, 12.38,

then a suffix shall be appended to the character string generated by 10.3.3. The suffix shall consist of a HYPHEN-MINUS followed by the canonical lexical representation (see W3C XML Schema Part 2, 2.3.1) of an integer. This integer shall be the least positive integer such that the new name is different from the type reference name of any other ASN.1 type assignment previously generated (in any ASN.1 module).

NOTE – As a consequence of this rule, all type reference names defined in an ASN.1 specification generated from a source XSD schema (including the standardized type references defined in the **XSD** module) will be unique within that ASN.1 specification. This allows maximum flexibility in the way that the generated ASN.1 type assignments are distributed across multiple ASN.1 modules (see 7.3).

10.3.4.2 If the name being generated is the identifier of a component of a sequence, set, or choice type, and the character string generated by 10.3.3 is identical to the identifier of a previously generated component of the same sequence, set, or choice type, then a suffix shall be appended to the character string generated by 10.3.3. The suffix shall consist of a HYPHEN-MINUS followed by the canonical lexical representation (see W3C XML Schema Part 2, 2.3.1) of an integer. This integer shall be the least positive integer such that the new identifier is different from the identifier of any previously generated component of that sequence, set, or choice type.

10.3.4.3 If the name being generated is the "identifier" in an "EnumerationItem" of an enumerated type, and the character string generated by 10.3.3 is identical to the "identifier" in another "EnumerationItem" previously generated in the same enumerated type, then a suffix shall be appended to the character string generated by 10.3.3. The suffix shall consist of a HYPHEN-MINUS followed by the canonical lexical representation (see W3C XML Schema Part 2, 2.3.1) of an integer. This integer shall be the least positive integer such that the new identifier is different from the "identifier" in any other "EnumerationItem" already present in that ASN.1 enumerated type.

10.3.5 For an ASN.1 type reference name (or identifier) that is generated by applying this subclause 10.3 to the **name** of an **element declaration**, **attribute declaration**, top-level **complex type definition** or top-level **simple type definition**, if the type reference name (or identifier) generated is different from the **name**, a final **NAME** encoding instruction shall be assigned to the ASN.1 type assignment with that type reference name (or to the component with that identifier) as specified in the three following subclauses.

10.3.5.1 If the only difference is the case of the first letter (which is upper case in the type reference name and lower case in the **name**), then the "Keyword" in the **NAME** encoding instruction shall be **UNCAPITALIZED**.

10.3.5.2 If the only difference is the case of the first letter (which is lower case in the identifier and upper case in the **name**), then the "Keyword" in the **NAME** encoding instruction shall be **CAPITALIZED**.

10.3.5.3 Otherwise, the "NewName" in the **NAME** encoding instruction shall be the **name**.

EXAMPLE – The top-level **complex type definition**:

```
<xsd:complexType name="COMPONENTS">
        <xsd:sequence>
                <xsd:element name="Elem" type="xsd:boolean"/>
                <xsd:element name="elem" type="xsd:integer"/>
                <xsd:element name="Elem-1" type="xsd:boolean"/>
                <xsd:element name="elem-1" type="xsd:integer"/>
        </xsd:sequence>
</xsd:complexType>
```

is mapped to the ASN.1 type assignment:

```
COMPONENTS-1 ::= [NAME AS "COMPONENTS"] SEQUENCE {
             elem     [NAME AS CAPITALIZED] BOOLEAN,
             elem-1   [NAME AS "elem"] INTEGER,
             elem-1-1 [NAME AS "Elem-1"] BOOLEAN,
             elem-1-2 [NAME AS "elem-1"] INTEGER }
```

10.3.6 For an ASN.1 type reference name (or identifier) that is generated by applying this subclause 10.3 to the **name** of an **element declaration**, **attribute declaration**, top-level **complex type definition** or user-defined top-level **simple type definition**, if the **target namespace** of the schema component is not **absent**, then a final **NAMESPACE** encoding instruction shall be assigned to the ASN.1 type assignment with that type reference name (or to the named type with that identifier) and shall specify the **target namespace** of the schema component.

10.3.7 For an ASN.1 identifier that is generated by this subclause 10.3 for the mapping of a **simple type definition** with an **enumeration** facet where the identifier generated is different from the corresponding member of the **value** of the **enumeration** facet, a final **TEXT** encoding instruction shall be assigned to the ASN.1 enumerated type, with qualifying information specifying the "identifier" in the "EnumerationItem" of the enumerated type. One of the two following subclauses applies.

© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

10.3.7.1 If the only difference is the case of the first letter (which is lower case in the identifier and upper case in the member of the **value** of the **enumeration** facet), then the "Keyword" in the **`TEXT`** encoding instruction shall be **`CAPITALIZED`**.

10.3.7.2 Otherwise, the "NewName" in the **`TEXT`** encoding instruction shall be the member of the **value** of the **enumeration** facet.

10.4 Order of the mapping

10.4.1 An order is imposed on the top-level schema components of the source XSD Schema on which the mapping is performed. This applies to **model group definition**s, top-level **complex type definition**s, top-level **simple type definition**s, top-level **attribute declaration**s, and top-level **element declaration**s.

NOTE – Other top-level schema components are not mapped to ASN.1, and XSD built-in types are mapped in a special way.

10.4.2 The order is specified in the three following subclauses.

10.4.2.1 Top-level schema components shall first be ordered by their **target namespace**, with the **absent** namespace preceding all namespace names specified in the XSD schema in ascending lexicographical order.

10.4.2.2 Within each target namespace, top-level schema components shall be divided into four sets ordered as follows:

a) **element declaration**s;

b) **attribute declaration**s;

c) **complex type definition**s and **simple type definition**s;

d) **model group definition**s.

10.4.2.3 Within each set (see 10.4.2.2), schema components shall be ordered by **name** in ascending lexicographical order.

10.4.3 Two sets of ASN.1 type assignments are generated by the mapping:

a) one set of ASN.1 type assignments (generated by clauses 13, 14, 15, 17, and 20) correspond directly to top-level schema components, and their type reference names are derived from the name of the schema component with no suffix appended;

b) another set of ASN.1 type assignments (generated by clauses 29, 30, and 31) correspond to special uses of top-level schema components, and their type reference names are generated from the name of the schema component followed by a suffix and (in some cases) by a post-suffix.

NOTE − For each top-level schema component in the source XSD Schema, at most one ASN.1 type assignment in the set in 10.4.3 a) can be generated, but multiple ASN.1 type assignments in the set in 10.4.3 b) can be generated.

10.4.4 ASN.1 type assignments in the set in 10.4.3 a) shall be generated in the order of the corresponding XSD schema components (see 10.4.1), and shall all be generated before any type assignments in 10.4.3 b) are generated.

10.4.5 ASN.1 type assignments in 10.4.3 b) shall be generated in the following order:

a) given two top-level schema components SC1 and SC2, where SC1 precedes SC2 in the order specified in 10.4.1, all the ASN.1 type assignments corresponding to SC1 (if any) shall be generated before any type assignments corresponding to SC2 are generated;

b) within each set of type assignments corresponding to any given schema component, type assignments shall be generated in an order based on the suffix specified in clauses 29 to 31, as follows:

1) suffix "**`-nillable`**";

2) suffix "**`-nillable-default-`**";

3) suffix "**`-nillable-fixed-`**";

4) suffix "**`-derivations`**";

5) suffix "**`-deriv-default-`**";

6) suffix "**`-deriv-fixed-`**";

7) suffix "**`-deriv-nillable`**";

8) suffix "**`-deriv-nillable-default-`**";

9) suffix "**`-deriv-nillable-fixed-`**";

10) suffix "**`-group`**";

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

c) for items 2, 3, 5, 6, 8, and 9 of b), within each set of type assignments corresponding to any given schema component and any given suffix, type assignments shall be generated in ascending lexicographical order of the post-suffix specified in clause 29 (if any).

11 Mapping uses of XSD built-in types

11.1 This clause applies as explicitly invoked by other clauses of this Recommendation | International Standard to generate an ASN.1 type definition corresponding to the use of an XSD built-in type.

NOTE – All XSD built-in types are **simple type definition**s with the exception of **xsd:anyType**, which is a **complex type definition**.

11.2 A use of an XSD built-in type shall be mapped to an ASN.1 type definition in accordance with Table 2, which gives the ASN.1 type definition to be used. The notation "XSD.Name" indicates that the ASN.1 type definition shall be the ASN.1 type definition (a "DefinedType") generated by applying 10.2 to the corresponding ASN.1 type assignment present in the **`XSD {joint-iso-itu-t asn1(1) specification(0) modules(0) xsd-module(2) version1(1)}`** module (version 1 mapping – see Annex A) or the **`XSD {joint-iso-itu-t asn1(1) specification(0) modules(0) xsd-module(2) version2(2)}`** module (version 2 mapping – see Annex B).

Table 2 – ASN.1 type definitions corresponding to uses of XSD built-in types

XSD built-in type	ASN.1 type definition	XSD built-in type	ASN.1 type definition
anyURI	`XSD.AnyURI`	Int	`XSD.Int`
anySimpleType	`XSD.AnySimpleType`	Integer	`INTEGER`
anyType	`XSD.AnyType` or `XSD.AnyType-nillable` (see 11.3)	language	`XSD.Language`
base64Binary	`[BASE64] OCTET STRING`	long	`XSD.Long`
boolean	`BOOLEAN`	Name	`XSD.Name`
byte	`INTEGER (-128..127)`	NCName	`XSD.NCName`
date	`XSD.Date`	negativeInteger	`INTEGER (MIN..-1)`
dateTime	`XSD.DateTime`	NMTOKEN	`XSD.NMTOKEN`
decimal	`XSD.Decimal`	NMTOKENS	`XSD.NMTOKENS`
double	`XSD.Double`	nonNegativeInteger	`INTEGER (0..MAX)`
duration	`XSD.Duration`	nonPositiveInteger	`INTEGER (MIN..0)`
ENTITIES	`XSD.ENTITIES`	normalizedString	`XSD.NormalizedString`
ENTITY	`XSD.ENTITY`	NOTATION	`XSD.NOTATION`
float	`XSD.Float`	positiveInteger	`INTEGER (1..MAX)`
gDay	`XSD.GDay`	QName	`XSD.QName`
gMonth	`XSD.GMonth`	short	`XSD.Short`
gMonthDay	`XSD.GMonthDay`	string	`XSD.String`
gYear	`XSD.GYear`	time	`XSD.Time`
gYearMonth	`XSD.GYearMonth`	token	`XSD.Token`
hexBinary	`OCTET STRING`	unsignedByte	`INTEGER (0..255)`
ID	`XSD.ID`	unsignedInt	`XSD.UnsignedInt`
IDREF	`XSD.IDREF`	unsignedLong	`XSD.UnsignedLong`
IDREFS	`XSD.IDREFS`	unsignedShort	`XSD.UnsignedShort`

11.3 A use of **xsd:anyType** as the **type definition** of an **element declaration** that is not **nillable** shall be mapped to **`XSD.AnyType`**. A use of **xsd:anyType** as the **type definition** of an **element declaration** that is **nillable** shall be mapped to **`XSD.AnyType-nillable`**.

12 Mapping facetS

This clause applies as explicitly invoked by other clauses of this Recommendation | International Standard to map a **facet** of a **simple type definition**. A **facet** of a **simple type definition** STD is mapped to an ASN.1 constraint applied to the ASN.1 type definition corresponding to the STD, unless the STD has an **enumeration** facet that is being mapped to an ASN.1

© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

"Enumeration" (see 12.4.1 and 12.4.2). In this case, no ASN.1 constraint is generated from the facet (see 12.1.2, 12.2.1, 12.3.1, and 12.5.1).

12.1 The length, minLength, and maxLength facets

12.1.1 The **length**, **minLength**, and **maxLength** facets shall be ignored for the XSD built-in types **xsd:QName** and **xsd:NOTATION** and for any **simple type definition** derived from these by restriction.

12.1.2 If a **length**, **minLength**, or **maxLength** facet belongs to a **simple type definition** that has also an **enumeration** facet being mapped to an ASN.1 "Enumeration" (see 12.4.1 and 12.4.2), then no "EnumerationItem"s shall be included in the "Enumeration" for the members (if any) of the **value** of the **enumeration** facet that do not satisfy the **length**, **minLength**, or **maxLength** facet.

12.1.3 Otherwise, the **length**, **minLength**, and **maxLength** facets of the **simple type definition** shall be mapped to an ASN.1 size constraint according to Table 3.

Table 3 – ASN.1 size constraints corresponding to the length, minLength, and maxLength facets

XSD facet	ASN.1 size constraint
length=*value*	`(SIZE(`*value*`))`
minLength=*min*	`(SIZE(`*min* `.. MAX))`
maxLength=*max*	`(SIZE(0 ..` *max*`))`
minLength=*min* **maxLength**=*max*	`(SIZE(`*min* `..` *max*`))`

12.2 The pattern facet

12.2.1 If a **pattern** facet belongs to a **simple type definition** that has also an **enumeration** facet being mapped to an ASN.1 "Enumeration" (see 12.4.1 and 12.4.2), then no "EnumerationItem"s shall be included in the "Enumeration" for the members (if any) of the **value** of the **enumeration** facet that do not satisfy the **pattern** facet.

12.2.2 Otherwise, the **pattern** facet shall be mapped to a user-defined constraint. One of the two following subclauses applies.

12.2.2.1 If the **value** of the **pattern** facet is a single regular expression, the user-defined constraint shall be:

```
(CONSTRAINED BY {/* XML representation of the XSD pattern "xyz" */})
```

where "xyz" is the XML representation of the **value** of the **pattern** facet, except that if the substring "***/**" appears in the **value** of the **pattern** facet, it shall be replaced by the character string "***/**".

12.2.2.2 If the **value** of the **pattern** facet is a conjunction of unions of regular expressions (the general case), the user-defined constraint is not specified (but see 12.5.4).

12.3 The whiteSpace facet

12.3.1 If a **whiteSpace** facet with a **value** of **replace** or **collapse** belongs to a **simple type definition** that has also an **enumeration** facet being mapped to an ASN.1 "Enumeration" (see 12.4.1 and 12.4.2), then the three following subclauses apply.

12.3.1.1 No "EnumerationItem"s shall be included in the "Enumeration" for the members (if any) of the **value** of the **enumeration** facet that contain any of the characters HORIZONTAL TABULATION, NEWLINE or CARRIAGE RETURN, or (in the case of **collapse**) contain leading, trailing, or multiple consecutive SPACE characters.

12.3.1.2 If the **value** of the **whiteSpace** facet is **replace** and a final **`TEXT`** encoding instruction with qualifying information is being assigned to the ASN.1 type definition, then a final **`WHITESPACE REPLACE`** encoding instruction shall also be assigned to it.

12.3.1.3 If the **value** of the **whiteSpace** facet is **collapse** and a final **`TEXT`** encoding instruction with qualifying information is being assigned to the ASN.1 type definition, then a final **`WHITESPACE COLLAPSE`** encoding instruction shall also be assigned to it.

12.3.2 Otherwise, at most one of the three following subclauses applies:

12.3.2.1 If the **value** of the **whiteSpace** facet is **preserve**, then the **whiteSpace** facet shall be ignored.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

12.3.2.2 If the **value** of the **whiteSpace** facet is **replace** and the ASN.1 type definition corresponding to the **simple type definition** is an ASN.1 restricted character string type, then a permitted alphabet constraint shall be added to the ASN.1 type definition to remove HORIZONTAL TABULATION, NEWLINE, and CARRIAGE RETURN characters. A final **WHITESPACE REPLACE** encoding instruction shall be assigned to the ASN.1 type definition. The following or an equivalent permitted alphabet constraint shall be used:

```
(FROM ({0, 0, 0, 32} .. {0, 16, 255, 255}))
```

12.3.2.3 If the **value** of the **whiteSpace** facet is **collapse** and the ASN.1 type definition corresponding to the **simple type definition** is an ASN.1 restricted character string type, then both a permitted alphabet constraint as specified in 12.3.2.2 and a pattern constraint that forbids leading, trailing, and multiple consecutive SPACE characters shall be added to the ASN.1 type definition. A final **WHITESPACE COLLAPSE** encoding instruction shall be assigned to the ASN.1 type definition. The following or an equivalent pattern constraint shall be used:

```
(PATTERN "([^ ]([^ ]| [^ ])*)?")
```

12.4 The enumeration facet

12.4.1 An **enumeration** facet belonging to a **simple type definition** with a **variety** of **atomic** that is derived by restriction (directly or indirectly) from **xsd:string** shall not be mapped to an ASN.1 constraint. Instead, the facet shall be mapped to the "Enumeration" of the ASN.1 enumerated type corresponding to the **simple type definition** (see 13.4) as specified in the four following subclauses.

12.4.1.1 For each member of the **value** of the **enumeration** facet, an "EnumerationItem" that is an "identifier" shall be added to the "Enumeration" (subject to 12.1.2, 12.2.1, 12.3.1, and 12.5.1).

12.4.1.2 Each "identifier" shall be generated by applying 10.3 to the corresponding member of the **value** of the **enumeration** facet.

12.4.1.3 The members of the **value** of the **enumeration** facet shall be mapped in ascending lexicographical order and any duplicate members shall be discarded.

12.4.1.4 If the **simple type definition** has a **whiteSpace** facet with the value **preserve** or **replace**, then the enumerated type shall be assigned at least one final **TEXT** encoding instruction with qualifying information indicating one or more of the "EnumerationItem"s, and without "TextToBeUsed".

NOTE – An important example of this is a restriction of **xsd:string** with an **enumeration** facet, which has **whiteSpace preserve** by default.

12.4.2 An **enumeration** facet belonging to a **simple type definition** with a **variety** of **atomic** that is derived by restriction (directly or indirectly) from **xsd:integer** shall not be mapped to an ASN.1 constraint. Instead, the facet shall be mapped to the "Enumeration" of the ASN.1 enumerated type corresponding to the **simple type definition** (see 13.5) as specified in the three following subclauses.

12.4.2.1 For each member of the **value** of the **enumeration** facet, an "EnumerationItem" that is a "NamedNumber" shall be added to the "Enumeration" (subject to 12.1.2, 12.2.1, 12.3.1, and 12.5.1).

12.4.2.2 The "identifier" in each "NamedNumber" shall be generated by concatenating the character string "**int**" with the canonical lexical representation (see W3C XML Schema Part 2, 2.3.1) of the corresponding member of the **value** of the **enumeration** facet. The "SignedNumber" in the "NamedNumber" shall be the ASN.1 value notation for the member (an integer number).

12.4.2.3 The members of the **value** of the **enumeration** facet shall be mapped in ascending numerical order and any duplicate members shall be discarded.

12.4.3 Any other **enumeration** facet shall be mapped to an ASN.1 constraint that is either a single value or a union of single values corresponding to the members of the **value** of the **enumeration**.

NOTE – The **enumeration** facet applies to the value space of the **base type definition**. Therefore, for an **enumeration** of the XSD built-in types **xsd:QName** or **xsd:NOTATION**, the value of the **uri** component of the **[USE-QNAME] SEQUENCE** produced as a single value ASN.1 constraint is determined, in the XML representation of an XSD Schema, by the namespace declarations whose scope includes the **xsd:QName** or **xsd:NOTATION**, and by the prefix (if any) of the **xsd:QName** or **xsd:NOTATION**.

EXAMPLE 1 – The following represents a top-level **simple type definition** that is a restriction of **xsd:string** with an **enumeration** facet:

```
<xsd:simpleType name="state">
        <xsd:restriction base="xsd:string">
                <xsd:enumeration value="off"/>
                <xsd:enumeration value="on"/>
        </xsd:restriction>
</xsd:simpleType>
```


© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

It is mapped to the ASN.1 type assignment:

```
State ::= [NAME AS UNCAPITALIZED] ENUMERATED {off, on}
```

EXAMPLE 2 – The following represents a top-level **simple type definition** that is a restriction of **xsd:integer** with an **enumeration** facet:

```
<xsd:simpleType name="integer-0-5-10">
    <xsd:restriction base="xsd:integer">
        <xsd:enumeration value="0"/>
        <xsd:enumeration value="5"/>
        <xsd:enumeration value="10"/>
    </xsd:restriction>
</xsd:simpleType>
```

It is mapped to the ASN.1 type assignment:

```
Integer-0-5-10 ::= [NAME AS UNCAPITALIZED] [USE-NUMBER] ENUMERATED {int0(0), int5(5),
int10(10)}
```

EXAMPLE 3 – The following represents a top-level **simple type definition** that is a restriction of **xsd:integer** with a **minInclusive** and a **maxInclusive** facet:

```
<xsd:simpleType name="integer-1-10">
    <xsd:restriction base="xsd:integer">
        <xsd:minInclusive value="1"/>
        <xsd:maxInclusive value="10"/>
    </xsd:restriction>
</xsd:simpleType>
```

It is mapped to the ASN.1 type assignment:

```
Integer-1-10 ::= [NAME AS UNCAPITALIZED] INTEGER(1..10)
```

EXAMPLE 4 – The following represents a top-level **simple type definition** that is a restriction (with a **minExclusive** facet) of another **simple type definition**, derived by restriction from **xsd:integer** with the addition of a **minInclusive** and a **maxInclusive** facet:

```
<xsd:simpleType name="multiple-of-4">
    <xsd:restriction>
        <xsd:simpleType>
            <xsd:restriction base="xsd:integer">
                <xsd:minInclusive value="1"/>
                <xsd:maxInclusive value="10"/>
            </xsd:restriction>
        </xsd:simpleType>
        <xsd:minExclusive value="5"/>
    </xsd:restriction>
</xsd:simpleType>
```

It is mapped to the ASN.1 type assignment:

```
Multiple-of-4 ::= [NAME AS UNCAPITALIZED] INTEGER(5<..10)
```

EXAMPLE 5 – The following represents a top-level **simple type definition** that is a restriction (with a **minLength** and a **maxLength** facet) of another **simple type definition**, derived by restriction from **xsd:string** with the addition of an **enumeration** facet:

```
<xsd:simpleType name="color">
    <xsd:restriction>
        <xsd:simpleType>
            <xsd:restriction base="xsd:string">
                <xsd:enumeration value="white"/>
                <xsd:enumeration value="black"/>
                <xsd:enumeration value="red"/>
            </xsd:restriction>
        </xsd:simpleType>
        <xsd:minLength value="2"/>
        <xsd:maxLength value="4"/>
    </xsd:restriction>
</xsd:simpleType>
```

It is mapped to the ASN.1 type assignment:

```
Color ::= [NAME AS UNCAPITALIZED] ENUMERATED {red}
```

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021

12.5 Other facets

12.5.1 If a **totalDigits**, **fractionDigits**, **maxInclusive**, **maxExclusive**, **minExclusive**, or **minInclusive** facet belongs to a **simple type definition** that has also an **enumeration** facet being mapped to an ASN.1 "Enumeration" (see 12.4.1 and 12.4.2), then no "EnumerationItem"s shall be included in the "Enumeration" for the members (if any) of the **value** of the **enumeration** facet that do not satisfy the **totalDigits**, **fractionDigits**, **maxInclusive**, **maxExclusive**, **minExclusive**, or **minInclusive** facet.

12.5.2 If a **maxInclusive**, **maxExclusive**, **minExclusive**, or **minInclusive** facet belongs to a **simple type definition** without an **enumeration** facet or with an **enumeration** facet which is not being mapped to an ASN.1 "Enumeration" (see 12.4.1 and 12.4.2), then one of the two following subclauses applies:

12.5.2.1 If the **simple type definition** is derived by restriction (directly or indirectly) from an XSD built-in date or time type (**xsd:date**, **xsd:dateTime**, **xsd:duration**, **xsd:gDay**, **xsd:gMonth**, **xsd:gYear**, **xsd:gYearMonth**, **xsd:gMonthDay**, or **xsd:time**), then the **maxInclusive**, **maxExclusive**, **minExclusive**, and **minInclusive** facets of the **simple type definition** shall be mapped to an ASN.1 user-defined constraint (see 12.5.4).

12.5.2.2 Otherwise, the **maxInclusive**, **maxExclusive**, **minExclusive** and **minInclusive** facets of the **simple type definition** shall be mapped to an ASN.1 value range or single value constraint in accordance with Table 4.

Table 4 – ASN.1 constraints corresponding to the maxInclusive, maxExclusive, minExclusive, and minInclusive facets

XSD facet	ASN.1 constraint
maxInclusive=*ub*	(**MIN** .. *ub*)
maxExclusive=*ub*	(**MIN** .. < *ub*)
minExclusive=*lb*	(*lb* <.. **MAX**)
minInclusive=*lb*	(*lb* .. **MAX**)
minInclusive=*ub* **maxInclusive=***lb*	(*lb* .. *ub*)
minInclusive=*v* **maxInclusive=***v*	(*v*)
minInclusive=*ub* **maxExclusive=***lb*	(*lb* ..< *ub*)
minExclusive=*ub* **maxInclusive=***lb*	(*lb* <.. *ub*)
minExclusive=*ub* **maxExclusive=***lb*	(*lb* <..< *ub*)

12.5.3 If a **totalDigits** or **fractionDigits** facet belongs to a **simple type definition** without an **enumeration** facet or with an **enumeration** facet which is not mapped to an ASN.1 "Enumeration" (see 12.4.1 and 12.4.2), then the **totalDigits** and **fractionDigits** facets of the **simple type definition** shall be mapped to a user-defined constraint (see 12.5.4).

12.5.4 When a facet is mapped to an ASN.1 user-defined constraint, it is recommended that the facet and its **value** appear in an ASN.1 comment in the user-defined constraint. The precise form of the user-defined constraint is not specified.

13 Mapping simple type definitions

13.1 This clause applies as explicitly invoked by other clauses of this Recommendation | International Standard to generate an ASN.1 type assignment or ASN.1 type definition corresponding to a **simple type definition**.

NOTE – This clause is not invoked for **simple type definition**s that are XSD built-in types.

13.2 A top-level **simple type definition** shall be mapped to an ASN.1 type assignment. The "typereference" in the "TypeAssignment" shall be generated by applying 10.3 to the **name** of the **simple type definition** and the "Type" in the "TypeAssignment" shall be an ASN.1 type definition as specified in subclauses 13.4 to 13.9.

13.3 An anonymous **simple type definition** shall be mapped to an ASN.1 type definition as specified in subclauses 13.4 to 13.9.

13.4 For a **simple type definition** with a **variety** of **atomic** with an **enumeration** facet that is derived by restriction (directly or indirectly) from **xsd:string**, the ASN.1 type definition shall be an ASN.1 enumerated type whose "Enumeration" shall be generated as specified in 12.4.1.

© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: ISO_IEC 8825-5_2021