

ISO/IEC 8825-3:2021-06 (E)

Information technology - ASN.1 encoding rules - Part 3: Specification of Encoding Control Notation (ECN)

Contents		Page
Introduction		x
1	Scope	1
2	Normative references	1
2.1	Identical Recommendations International Standards	1
2.2	Additional references	2
3	Definitions	2
3.1	ASN.1 definitions	2
3.2	ECN-specific definitions	2
4	Abbreviations	5
5	Definition of ECN syntax	5
6	Encoding conventions and notation	5
7	The ECN character set	5
8	ECN lexical items	6
8.1	Encoding object references	6
8.2	Encoding object set references	6
8.3	Encoding class references	6
8.4	Reserved word items	7
8.5	Reserved encoding class name items	7
8.6	Non-ECN item	7
9	ECN Concepts	7
9.1	Encoding Control Notation (ECN) specifications	8
9.2	Encoding classes	8
9.3	Encoding structures	9
9.4	Encoding objects	9
9.5	Encoding object sets	9
9.6	Defining new encoding classes	9
9.7	Defining encoding objects	11
9.8	Differential encoding-decoding	11
9.9	Encoders options in encodings	12
9.10	Properties of encoding objects	12
9.11	Parameterization	12
9.12	Governors	13
9.13	General aspects of encodings	13
9.14	Identification of information elements	13
9.15	Reference fields and determinants	14
9.16	Replacement classes and structures	14
9.17	Mapping abstract values onto fields of encoding structures	15
9.18	Transforms and transform composites	15
9.19	Contents of Encoding Definition Modules	16
9.20	Contents of the Encoding Link Module	17
9.21	Defining encodings for primitive encoding classes	17
9.22	Application of encodings	19
9.23	Combined encoding object set	19
9.24	Application point	19
9.25	Conditional encodings	20
9.26	Other conditions for applying encodings	20
9.27	Encoding control for the open type	21
9.28	Changes to ASN.1 Recommendations International Standards	21
10	Identifying encoding classes, encoding objects, and encoding object sets	21
11	Encoding ASN.1 types	24
11.1	General	24

11.2	Built-in encoding classes used for implicitly generated encoding structures	24
11.3	Simplification and expansion of ASN.1 notation for encoding purposes.....	25
11.4	The implicitly generated encoding structure	27
12	The Encoding Link Module (ELM).....	27
12.1	Structure of the ELM	28
12.2	Encoding types	28
13	Application of encodings.....	29
13.1	General	29
13.2	The combined encoding object set and its application	29
14	The Encoding Definition Module (EDM)	31
15	The renames clause	33
15.1	Explicitly generated and exported structures	33
15.2	Name changes	34
15.3	Specifying the region for name changes	35
16	Encoding class assignments	36
16.1	General	36
16.2	Encoding structure definition	39
16.3	Alternative encoding structure	41
16.4	Repetition encoding structure.....	41
16.5	Concatenation encoding structure	42
17	Encoding object assignments	42
17.1	General	42
17.2	Encoding with a defined syntax	43
17.3	Encoding with encoding object sets	44
17.4	Encoding using value mappings.....	44
17.5	Encoding an encoding structure	45
17.6	Differential encoding-decoding.....	47
17.7	Encoding options.....	48
17.8	Non-ECN definition of encoding objects	49
18	Encoding object set assignments	49
18.1	General	49
18.2	Built-in encoding object sets	50
19	Mapping values	51
19.1	General	51
19.2	Mapping by explicit values	52
19.3	Mapping by matching fields.....	53
19.4	Mapping by #TRANSFORM encoding objects	54
19.5	Mapping by abstract value ordering.....	54
19.6	Mapping by value distribution	56
19.7	Mapping integer values to bits	57
20	Defining encoding objects using defined syntax	58
21	Types used in defined syntax specification	59
21.1	The Unit type	59
21.2	The EncodingSpaceSize type.....	59
21.3	The EncodingSpaceDetermination type.....	60
21.4	The UnusedBitsDetermination type.....	60
21.5	The OptionalityDetermination type	61
21.6	The AlternativeDetermination type	62
21.7	The RepetitionSpaceDetermination type.....	62
21.8	The Justification type	63
21.9	The Padding type	64
21.10	The Pattern and Non-Null-Pattern types	64
21.11	The RangeCondition type.....	65

21.12	The Comparison type.....	65
21.13	The SizeRangeCondition type	66
21.14	The ReversalSpecification type	66
21.15	The ResultSize type	67
21.16	The HandleValueSet type.....	67
21.17	The IntegerMapping type	68
22	Commonly used encoding property groups.....	68
22.1	Replacement specification.....	68
22.1.1	Encoding properties, syntax and purpose.....	68
22.1.2	Specification restrictions	69
22.1.3	Encoder actions	70
22.1.4	Decoder actions.....	70
22.2	Pre-alignment and padding specification	71
22.2.1	Encoding properties, syntax and purpose.....	71
22.2.2	Specification constraints	71
22.2.3	Encoder actions	71
22.2.4	Decoder actions.....	71
22.3	Start pointer specification.....	72
22.3.1	Encoding properties, syntax and purpose.....	72
22.3.2	Specification constraints	72
22.3.3	Encoder actions	72
22.3.4	Decoder actions.....	72
22.4	Encoding space specification	73
22.4.1	Encoding properties, syntax and purpose.....	73
22.4.2	Specification restrictions	73
22.4.3	Encoder actions	74
22.4.4	Decoder actions.....	74
22.5	Optionality determination	75
22.5.1	Encoding properties, syntax and purpose.....	75
22.5.2	Specification restrictions	75
22.5.3	Encoder actions	76
22.5.4	Decoder actions.....	76
22.6	Alternative determination.....	77
22.6.1	Encoding properties, syntax and purpose.....	77
22.6.2	Specification restrictions	77
22.6.3	Encoder actions	77
22.6.4	Decoder actions.....	78
22.7	Repetition space specification.....	78
22.7.1	Encoding properties, syntax and purpose.....	78
22.7.2	Specification constraints	79
22.7.3	Encoder actions	80
22.7.4	Decoder actions.....	80
22.8	Value padding and justification.....	81
22.8.1	Encoding properties, syntax, and purpose.....	81
22.8.2	Specification restrictions	82
22.8.3	Encoder actions	82
22.8.4	Decoder actions.....	83
22.9	Identification handle specification	83
22.9.1	Encoding properties, syntax and purpose.....	83
22.9.2	Specification constraints	84
22.9.3	Encoders actions.....	84
22.9.4	Decoders actions	84
22.10	Concatenation specification	84
22.10.1	Encoding properties, syntax and purpose.....	84
22.10.2	Specification constraints	85
22.10.3	Encoder actions	85
22.10.4	Decoder actions.....	85
22.11	Contained type encoding specification.....	85
22.11.1	Encoding properties, syntax and purpose.....	85
22.11.2	Encoder actions	86

22.11.3	Decoder actions	86
22.12	Bit reversal specification	86
22.12.1	Encoding properties, syntax, and purpose	86
22.12.2	Specification constraints	86
22.12.3	Encoder actions	87
22.12.4	Decoder actions	87
23	Defined syntax specification for bit-field and constructor classes	87
23.1	Defining encoding objects for classes in the alternatives category	87
23.1.1	The defined syntax	87
23.1.2	Purpose and restrictions	88
23.1.3	Encoder actions	88
23.1.4	Decoder actions	88
23.2	Defining encoding objects for classes in the bitstring category	88
23.2.1	The defined syntax	88
23.2.2	Model for the encoding of classes in the bitstring category	89
23.2.3	Purpose and restrictions	90
23.2.4	Encoder actions	90
23.2.5	Decoder actions	91
23.3	Defining encoding objects for classes in the boolean category	91
23.3.1	The defined syntax	91
23.3.2	Purpose and restrictions	92
23.3.3	Encoder actions	93
23.3.4	Decoder actions	93
23.4	Defining encoding objects for classes in the characterstring category	93
23.4.1	The defined syntax	93
23.4.2	Model for the encoding of classes in the characterstring category	94
23.4.3	Purpose and restrictions	94
23.4.4	Encoder actions	95
23.4.5	Decoder actions	95
23.5	Defining encoding objects for classes in the concatenation category	95
23.5.1	The defined syntax	95
23.5.2	Purpose and restrictions	97
23.5.3	Encoder actions	97
23.5.4	Decoder actions	98
23.6	Defining encoding objects for classes in the integer category	98
23.6.1	The defined syntax	98
23.6.2	Purpose and restrictions	98
23.6.3	Encoder actions	98
23.6.4	Decoder actions	98
23.7	Defining encoding objects for the #CONDITIONAL-INT class	98
23.7.1	The defined syntax	98
23.7.2	Purpose and restrictions	100
23.7.3	Encoder actions	101
23.7.4	Decoder actions	102
23.8	Defining encoding objects for classes in the null category	102
23.8.1	The defined syntax	102
23.8.2	Purpose and restrictions	103
23.8.3	Encoder actions	103
23.8.4	Decoder actions	104
23.9	Defining encoding objects for classes in the octetstring category	104
23.9.1	The defined syntax	104
23.9.2	Model for the encoding of classes in the octetstring category	105
23.9.3	Purpose and restrictions	105
23.9.4	Encoder actions	106
23.9.5	Decoder actions	106
23.10	Defining encoding objects for classes in the open type category	106
23.10.1	The defined syntax	106
23.10.2	Model for the encoding of classes in the open type category	107
23.10.3	Purpose and restrictions	108
23.10.4	Encoder actions	108
23.10.5	Decoder actions	108

23.11	Defining encoding objects for classes in the optionality category	109
23.11.1	The defined syntax	109
23.11.2	Purpose and restrictions	109
23.11.3	Encoder actions	110
23.11.4	Decoder actions	110
23.12	Defining encoding objects for classes in the pad category	110
23.12.1	The defined syntax	110
23.12.2	Purpose and restrictions	111
23.12.3	Encoder actions	111
23.12.4	Decoder actions	111
23.13	Defining encoding objects for classes in the repetition category	112
23.13.1	The defined syntax	112
23.13.2	Purpose and restrictions	112
23.13.3	Encoder actions	112
23.13.4	Decoder actions	112
23.14	Defining encoding objects for the #CONDITIONAL-REPETITION class	112
23.14.1	The defined syntax	112
23.14.2	Purpose and restrictions	114
23.14.3	Encoder actions	114
23.14.4	Decoder actions	115
23.15	Defining encoding objects for classes in the tag category	115
23.15.1	The defined syntax	115
23.15.2	Purpose and restrictions	116
23.15.3	Encoder actions	117
23.15.4	Decoder actions	117
23.16	Defining encoding objects for classes in the other categories	117
24	Defined syntax specification for the #TRANSFORM encoding class	118
24.1	Summary of encoding properties and defined syntax	118
24.2	Source and target of transforms	120
24.3	The int-to-int transform	121
24.4	The bool-to-bool transform	122
24.5	The bool-to-int transform	122
24.6	The int-to-bool transform	123
24.7	The int-to-chars transform	123
24.8	The int-to-bits transform	124
24.9	The bits-to-int transform	125
24.10	The char-to-bits transform	125
24.11	The bits-to-char transform	127
24.12	The bit-to-bits transform	128
24.13	The bits-to-bits transform	128
24.14	The chars-to-composite-char transform	129
24.15	The bits-to-composite-bits transform	129
24.16	The octets-to-composite-bits transform	130
24.17	The composite-char-to-chars transform	130
24.18	The composite-bits-to-bits transform	130
24.19	The composite-bits-to-octets transform	130
25	Complete encodings and the #OUTER class	131
25.1	Encoding properties, syntax and purpose for the #OUTER class	131
25.2	Encoder actions for #OUTER	132
25.3	Decoder actions for #OUTER	132
Annex A	Addendum to Rec. ITU-T X.680 ISO/IEC 8824-1	133
A.1	Exports and imports clauses	133
A.2	Addition of REFERENCE	133
A.3	Notation for character string values	134
Annex B	Addendum to Rec. ITU-T X.681 ISO/IEC 8824-2	135
B.1	Definitions	135
B.2	Additional lexical items	135

B.2.1	Ordered value list field references.....	135
B.2.2	Ordered encoding object list field references.....	135
B.2.3	Encoding class field references.....	135
B.3	Addition of "ENCODING-CLASS".....	135
B.4	FieldSpec additions.....	135
B.5	Fixed-type ordered value list field spec.....	136
B.6	Fixed-class encoding object field spec.....	136
B.7	Variable-class encoding object field spec.....	136
B.8	Fixed-class encoding object set field spec.....	137
B.9	Fixed-class ordered encoding object list field spec.....	137
B.10	Encoding class field spec.....	137
B.11	Ordered value list notation.....	137
B.12	Ordered encoding object list notation.....	137
B.13	Primitive field names.....	138
B.14	Additional reserved words.....	138
B.15	Definition of encoding objects.....	139
B.16	Additions to "Setting".....	139
B.17	Encoding class field type.....	139
Annex C	Addendum to Rec. ITU-T X.683 ISO/IEC 8824-4.....	141
C.1	Parameterized assignments.....	141
C.2	Parameterized encoding assignments.....	141
C.3	Referencing parameterized definitions.....	142
C.4	Actual parameter list.....	142
Annex D	Examples.....	144
D.1	General examples.....	144
D.1.1	An encoding object for a boolean type.....	144
D.1.2	An encoding object for an integer type.....	145
D.1.3	Another encoding object for an integer type.....	145
D.1.4	An encoding object for an integer type with holes.....	145
D.1.5	A more complex encoding object for an integer type.....	145
D.1.6	Positive integers encoded in BCD.....	146
D.1.7	An encoding object of class #BITS.....	147
D.1.8	An encoding object for an octetstring type.....	148
D.1.9	An encoding object for a character string type.....	148
D.1.10	Mapping character values to bit values.....	148
D.1.11	An encoding object for a sequence type.....	149
D.1.12	An encoding object for a choice type.....	149
D.1.13	Encoding a bitstring containing another encoding.....	150
D.1.14	An encoding object set.....	150
D.1.15	ASN.1 definitions.....	151
D.1.16	EDM definitions.....	151
D.1.17	ELM definitions.....	151
D.2	Specialization examples.....	152
D.2.1	Encoding by distributing values to an alternative encoding structure.....	152
D.2.2	Encoding by mapping ordered abstract values to an alternative encoding structure.....	152
D.2.3	Compression of non-continuous value ranges.....	153
D.2.4	Compression of non-continuous value ranges using a transform.....	153
D.2.5	Compression of an unevenly distributed value set by mapping ordered abstract values.....	154
D.2.6	Presence of an optional component depending on the value of another component.....	154
D.2.7	The presence of an optional component depends on some external condition.....	155
D.2.8	A variable length list.....	155
D.2.9	Equal length lists.....	156
D.2.10	Uneven choice alternative probabilities.....	157
D.2.11	A version 1 message.....	158
D.2.12	The encoding object set.....	159
D.2.13	ASN.1 definitions.....	159
D.2.14	EDM definitions.....	160
D.2.15	ELM definitions.....	160
D.3	Explicitly generated structure examples.....	160

D.3.1	Sequence with optional components defined by a pointer	161
D.3.2	Addition of a boolean type as a presence determinant	161
D.3.3	Sequence with optional components identified by a unique tag and delimited by a length field	163
D.3.4	Sequence-of type with a count	164
D.3.5	Encoding object sets.....	164
D.3.6	ASN.1 definitions.....	165
D.3.7	EDM definitions.....	165
D.3.8	ELM definitions	165
D.4	A more-bit encoding example	166
D.4.1	Description of the problem.....	166
D.4.2	Use of ASN.1 to provide the more-bit determinant.....	166
D.4.3	Use of value mappings to provide the more-bit determinant	167
D.4.4	Use of the replacement mechanism to provide the more-bit determinant	168
D.5	Legacy protocol specified with tabular notation	168
D.5.1	Introduction	168
D.5.2	Encoding definition for the top-level message structure	170
D.5.3	Encoding definition for a message structure	170
D.5.4	Encoding for the sequence type "B".....	171
D.5.5	Encoding for an octet-aligned sequence-of type with a length determinant.....	171
D.5.6	Encoding for an octet-aligned sequence-of type which continues to the end of the PDU.....	171
D.5.7	EDM definitions.....	171
D.5.8	ELM definitions	172
Annex E	Support for Huffman encodings	173
Annex F	Additional information on the Encoding Control Notation (ECN).....	175
Annex G	Summary of the ECN notation	176