

## Annex B Rules for BER-TLV Data Objects

As defined in ISO/IEC 8825, a BER-TLV data object consists of 2-3 consecutive fields:

- The tag field (T) consists of one or more consecutive bytes. It indicates a class, a type, and a number (see Table 35). The tag field of the data objects described in this specification is coded on **one or two bytes.**
- The length field (L) consists of one or more consecutive bytes. It indicates the length of the following field. The length field of the data objects described in this specification which are transmitted over the card-terminal interface is **coded on one or two bytes.**

**Note:** Three length bytes may be used if needed for templates '71' and '72' and tag '86' (to express length greater than 255 bytes), as they are not transmitted over the card-terminal interface.

- The value field (V) indicates the value of the data object. If L = '00', the value field is not present.

A BER-TLV data object belongs to one of the following two categories:

- A primitive data object where the value field contains a data element for financial transaction interchange.
- A constructed data object where the value field contains one or more primitive or constructed data objects. The value field of a constructed data object is called a template.

The coding of BER-TLV data objects is defined as follows.

## B1 Coding of the Tag Field of BER-TLV Data Objects

Table 35 describes the first byte of the tag field of a BER-TLV data object:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							Universal class
0	1							Application class
1	0							Context-specific class
1	1							Private class
		0						Primitive data object
		1						Constructed data object
			1	1	1	1	1	See subsequent bytes
			Any other value <31					Tag number

**Table 35: Tag Field Structure (First Byte) BER-TLV**

According to ISO/IEC 8825, Table 36 defines the coding rules of the subsequent bytes of a BER-TLV tag when tag numbers  $\geq 31$  are used (that is, bits b5 - b1 of the first byte equal '11111').

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Another byte follows
0								Last tag byte
	Any value > 0							(Part of) tag number

**Table 36: Tag Field Structure (Subsequent Bytes) BER-TLV**

Before, between, or after TLV-coded data objects, '00' or 'FF' bytes without any meaning may occur (for example, due to erased or modified TLV-coded data objects).

The tag field of a BER-TLV data object is coded according to the following rules:

- The following application class templates defined in ISO/IEC 7816 apply: '61' and '6F'.
- The following range of application class templates is defined in Part II: '70' to '7F'. The meaning is then specific to the context of an application according to this specification. Tags '78', '79', '7D', and '7E' are defined in ISO/IEC 7816-6 and are not used in this specification.
- The application class data objects defined in ISO/IEC 7816 and described in Part II are used according to the ISO/IEC 7816 definition.
- Context-specific class data objects are defined in the context of this specification or in the context of the template in which they appear.
- The coding of primitive context-specific class data objects in the ranges '80' to '9E' and '9F00' to '9F4F' is reserved for this specification.
- The coding of primitive context-specific class data objects in the range '9F50' to '9F7F' is reserved for the payment systems.
- The coding of primitive and constructed private class data objects is left to the discretion of the issuer.

## B2 Coding of the Length Field of BER-TLV Data Objects

When bit **b8** of the most significant byte of the length field is set to 0, the length field consists of only one byte. Bits b7 to b1 code the number of bytes of the value field. The length field is within the range 1 to 127.

When bit b8 of the most significant byte of the length field is set to 1, the subsequent bits **b7 to b1** of the most significant byte **code the number of subsequent bytes in the length field**. The subsequent bytes code an integer representing the number of bytes in the value field. Two bytes are necessary to express **up to 255 bytes in the value field**.

### B3 Coding of the Value Field of Data Objects

A data element is the value field (V) of a primitive BER-TLV data object. A data element is the smallest data field that receives an identifier (a tag).

A primitive data object is structured as illustrated in Figure 16:

Tag (T)	Length (L)	Value (V)
---------	------------	-----------

Figure 16: Primitive BER-TLV Data Object (Data Element)

A constructed BER-TLV data object consists of a tag, a length, and a value field composed of one or more BER-TLV data objects. A record in an AEF governed by this specification is a constructed BER-TLV data object. A constructed data object is structured as illustrated in Figure 17:

Tag (T)	Length (L)	Primitive or constructed BER-TLV data object number 1	...	Primitive or constructed BER-TLV data object number n
------------	---------------	---	-----	---

Figure 17: Constructed BER-TLV Data Object