INTERNATIONAL STANDARD

ISO/IEC 19785-3

Third edition
2020-09

Information technology — Common Biometric Exchange Formats Framework —

Part 3: Patron format specifications

Technologies de l'information — Cadre de formats d'échange biométriques communs —

Partie 3: Spécifications de format d'usager

Reference number
ISO/IEC 19785-3:2020(E)

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

Contents

Page

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).

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 http://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.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 37, *Biometrics*.

This third edition cancels and replaces the second edition (ISO/IEC 19785-3:2015), which has been technically revised.

The main changes compared to the previous edition are as follows:

— Vocabulary and technical updates and corrections.

— Specification of new CBEFF patron formats for BioAPI 2.0, Presentation Attack Detection (PAD), new TLV-encoded patron format for ICCs or other tokens (with explicit tag allocation authority), and Self-identifying Tag-oriented simple and complex BIRs.

— Inclusion of conformance requirements in Annex B and coding specifications for tagged patron formats, such as TLV, XML, and JSON in Annex C.

— Some technical comments and clarifications on the Clause 11 patron format are introduced within Annex D (however, for legacy reasons, the technical content of the TLV-encoded patron format for use with smartcards or other tokens [with implicit tag allocation authority] in Clause 11, is presented as it was originally published in ISO/IEC 19785-3:2007).

— The permitted values for CBEFF_BDB_biometric_type and CBEFF_BDB_biometric_subtype have been harmonized across all patron formats (except for Clause 11, which is preserved for legacy reasons), using the common CBEFF-DATA-ELEMENTS definitions in 6.2.

— The CBEFF_version for the patron formats has been increased to (major 4, minor 0) to support newly defined CBEFF_BDB_biometric_type values from ISO/IEC 19785-1 (including Body Photography, Friction Ridge, and Thermal).

A list of all parts in the ISO/IEC 19785 series can be found on the ISO website.

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.

Introduction

Biometric-based authentication systems and applications are expected to support multiple biometric devices and multiple biometric data formats. The Common Biometric Exchange Formats Framework (CBEFF) promotes interoperability of biometric-based application programmes and systems developed by different vendors by facilitating biometric data interchange.

ISO/IEC 19785-1 defines the following items that enable standardized biometric data interchange:

a) a 3-part standardized structure for biometric information records (BIRs) consisting of:

   1) standardized biometric headers (SBHs),

   2) biometric data blocks (BDBs, which may be standardized or proprietary), and

   3) optional security blocks (SBs);

b) variations of the 3-part structure to support BIRs containing:

   1) only one SBH, at least one or more BDBs and possibly one SB (simple CBEFF BIRs),

   2) only one self-identifying SBH, at least one or more BDB and possibly one SB (self-identifying simple CBEFF BIRs), and

   3) more than one BDB along with some number of SBHs necessary to encode the BIR's structure and some number of SBs (complex CBEFF BIRs, multiple CBEFF BIRs).

c) a self-identifying concept, which may be applied to any variation of the 3-part structure for BIRs, using the 'SBIR' field as defined in ISO/IEC 19785-1;

d) more than 40 data elements and their associated abstract values that can be used in an SBH to describe attributes of a BDB within a BIR, as well as attributes of the BIR itself;

e) the concept of a CBEFF patron format (but ISO/IEC 19785-1 does not itself define any patron formats), which is a detailed specification of the structure and content of a particular, standardized BIR;

f) the concept of a CBEFF patron, which is a recognized standards organization that has registered with the Biometric Registration Authority (BRA) and declared its intention to define CBEFF patron format specifications;

g) the concept of the BRA, which is the mechanism by which unique identifiers are assigned to organizations (standards organizations, vendors and others) that create BDB formats and CBEFF patron formats (ISO/IEC 19785-2 defines the identification scheme for biometric objects and organizations registered by the BRA);

h) CBEFF data elements [see c) above] that support, within the SBH, the unique identifiers assigned by the BRA for biometric organizations, BDB formats, biometric products, capture devices, feature extraction algorithms, comparison algorithms, quality algorithms, compression algorithms, PAD mechanisms, patron formats (self-identifying or not), and SB formats.

Patron formats can be specified in other standards documents and registered with the BRA (see ISO/IEC 19785-2). For example, there is a registered patron format specified in ISO/IEC 19785-1. For a complete list of registered patron formats, consult the CBEFF Biometric Registration Authority website.

This document specifies a number of CBEFF patron formats that are considered to be of general utility in a variety of domains of use. Additional ISO/IEC JTC 1/SC 37 patron format specifications may be published as new clauses in future amendments to this document, or in other ISO/IEC JTC 1/SC 37 International Standards.

The CBEFF patron format type unambiguously identifies the CBEFF patron format within the scope of the CBEFF patron format owner. The CBEFF patron format type is unambiguous within the scope of an ASN.1 Object Identifier (see ISO/IEC 9834-1) that identifies the BRA (see ISO/IEC 19785-2). That ASN.1

Object Identifier (OID) is itself globally unambiguous within the scope of all ASN.1 OIDs, which forms a widely used global namespace.

NOTE ASN.1 OIDs are used by ITU-T, ITU-R, the UPU and many ISO and IEC Standards, to identify some IETF MIME types and for many other purposes. (These acronyms have not been spelled out, as the precise identification of these organizations is not relevant to this document.)

The combination of the BRA OID, the CBEFF patron format owner, and the CBEFF patron format type forms a larger ASN.1 OID that provides an unambiguous identification of the CBEFF patron format. This document specifies, for each CBEFF patron format that it defines, the ASN.1 OID that unambiguously identifies that CBEFF patron format.

New implementations should make use of the tag-oriented CBEFF version 4.0 patron formats, i.e. Clauses 16 and onwards.

It is also important to note that, for legacy reasons, the technical content of patron format type 5, TLV-encoded patron format for use with smartcards or other tokens (with implicit tag allocation authority) in Clause 11, is presented as it was originally published in ISO/IEC 19785-3:2007 (with minor editorial updates and technical comments introduced in Annex D). New implementations should use the explicit tag allocation authority alternative provided by the patron format included in Clause 19.

Information technology — Common Biometric Exchange Formats Framework —

Part 3:
Patron format specifications

1 Scope

This document specifies and publishes registered Common Biometric Exchange Formats Framework (CBEFF) patron formats defined by the CBEFF patron ISO/IEC JTC 1/SC 37, and specifies their registered CBEFF patron format types (see ISO/IEC 19785-1) and resulting full ASN.1 OIDs.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 2382-37, *Information technology — Vocabulary — Part 37: Biometrics*

ISO/IEC 7816-11, *Identification cards — Integrated circuit cards — Part 11: Personal verification through biometric methods*

ISO 8601 (all parts), *Data and time — Representations for information interchange*

ISO/IEC 8825-1, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER) — Part 1:*

ISO/IEC 8825-4, *Information technology — ASN.1 encoding rules: XML Encoding Rules (XER) — Part 4:*

ISO/IEC 10646, *Information technology — Universal coded character set (UCS)*

ISO/IEC 19785-1, *Information technology — Common Biometric Exchange Formats Framework — Part 1: Data element specification*

ISO/IEC 19785-2, *Information technology — Common Biometric Exchange Formats Framework — Part 2: Biometric registration authority*

ISO/IEC 24787, *Information technology — Identification cards — On-card biometric comparison*

ISO/IEC 30107-2, *Information technology — Biometric presentation attack detection — Part 2: Data formats*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 2382-37 and ISO/IEC 19785-1 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
biometric information template group
structure which allows two or more biometric information template instances to be nested

Note 1 to entry: The construction and use of biometric information template group Data Objects both for on-card and off-card biometric comparison are specified in ISO/IEC 7816-11.

3.2
Common Biometric Exchange Format Framework patron format structure type
CBEFF patron format structure type
indication of the CBEFF BIR structure used to define a given patron format, including whether the structure is self-identifying

Note 1 to entry: The possible structure types are defined in ISO/IEC 19785-1 and include:

— Simple CBEFF BIR;

— Complex CBEFF BIR;

— Multiple CBEFF BIR.

Note 2 to entry: ISO/IEC 19785-1 allows one or more BDBs to be included in the Simple CBEFF BIR structure type. However, unless otherwise indicated, all Simple CBEFF BIR structures defined in this document allow only a single BDB to be included.

3.3
TLV encoding
common form of encoding (with many variants) in which every field in the encoding has an assigned type (or tag) that is unambiguous in a certain context, a length determinant, and a value part that may contain further TLV components, nested to any depth

4 Symbols and abbreviated terms

For the purposes of this document, the symbols and abbreviated terms given in ISO/IEC 19785-1 and the following apply.

APDU	application protocol data unit, as defined in ISO/IEC 7816-4
ASN.1	Abstract Syntax Notation One
BER-TLV	Basic Encoding Rules – Tag Length Value
DO	data object, as defined in ISO/IEC 7816-4
ICC	integrated circuit card, as defined in ISO/IEC 7816-4
JSON	JavaScript Object Notation (as specified in RFC 8259 and in ISO/IEC 21778)
OID	object identifier
PAD	presentation attack detection
TLV	type (or tag), length, and value
UTF-8	Unicode (or Universal Coded Character Set) Tranformation Format – 8-bit
XML	eXtensible Markup Language
XSD	XML Schema Definition
W3C	World Wide Web Consortium

5 Conformance

Clauses 6 onwards of this document specify the encoding (semantics) that can form a valid instance of the CBEFF patron format that is defined in each clause, together with the ASN.1 OID.

If an implementation claims that it supports and conforms to a CBEFF patron format defined in this document, then it shall either be:

a) capable of generating at least one of the encodings specified for that CBEFF patron format, or

b) capable of decoding (determining the semantics of) or processing in any other way encoding specified for that CBEFF patron format.

Annex B of this document specifies a patron format conformance statement (PFCS) for each patron format defined in this document.

6 ASN.1 type definitions for CBEFF data elements and abstract values

6.1 General

This clause specifies in 6.2 an ASN.1 module **CBEFF-DATA-ELEMENTS** that defines types (see ITU-T Rec. X.680 and ISO/IEC 8824-1) for each of the CBEFF data elements. These type definitions are fully aligned with the abstract values of CBEFF data elements specified in ISO/IEC 19785-1 (conforming to CBEFF version "major(4) minor(0)"), and do not in themselves specify encodings of those abstract values. Encodings are determined by the patron formats specified in ASN.1 in this document.

All patron formats listed within this document that import definitions from the CBEFF-DATA-ELEMENTS {iso(1) standard(0) cbeff(19785) modules(0) types-for-cbeff-data-elements(1)} ASN.1 module utilize the definitions from subclause 6.2. Those patron formats do not use the definitions from any previous standard versions, even if the patron formats were initially defined in CBEFF version(s) before 4.0.

Any technical changes to a patron format, including updates to supported CBEFF data elements or abstract values, will result in a new major patron format version (this includes any technical changes made by using definitions from 6.2). However, if the technical structure of the format or abstract values have not changed, then no major version change will occur.

6.2 CBEFF data elements type definitions module

The following ASN.1 module can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en. The ASN.1 data element definitions listed in this section form a library, which can be referenced by any patron format using the ASN.1 IMPORT method.

```
CBEFF-DATA-ELEMENTS
{iso(1) standard(0) cbeff(19785) modules(0) types-for-cbeff-data-elements(1)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN
```

```
  /* The following elements do not need an individual definition, as they
   are either basic data types (BDBCreationDate vs DATE-TIME),
   or equal to other defined data types (e.g. Product vs. RegistryID):
  */
    NO ENCRYPTION = false
    ENCRYPTION = true.
    NO INTEGRITY = false, INTEGRITY = true.
    SubheaderCount ::= INTEGER (0..255)
    BDBCreationDate ::= DATE-TIME
        -- A patron format that uses this type shall specify
        -- its encoding for noValueAvailable
    BDBValidityPeriod ::= ValidityPeriod
    BIRCreationDate ::= DATE-TIME
      -- A patron format that uses this type shall specify
      -- its encoding for noValueAvailable
    Product ::= RegistryID
    PatronFormat ::= RegistryID
    BIRValidityPeriod ::= ValidityPeriod
    BiometricDataBlock ::= OCTET STRING
    SBFormat ::= RegistryID

  EncryptionOptions ::= BOOLEAN
     -- NO ENCRYPTION = false, ENCRYPTION = true.

  IntegrityOptions ::= BOOLEAN
     -- NO INTEGRITY = false, INTEGRITY = true.

  BiometricType ::= INTEGER
     {
     noValueAvailable                (0),
     multipleBiometricTypes          (1),
     face                            (2),
     voice                           (4),
     finger                          (8),
     iris                            (16),
     retina                          (32),
     handGeometry                    (64),
     signatureSign                   (128),
     keystroke                       (256),
     lipMovement                     (512),
     gait                            (4096),
     vein                            (8192),
     dna                             (16384),
     ear                             (32768),
     foot                            (65536),
     scent                           (131072),
     bodyPhotography                 (262144),
     frictionRidge                   (524288),
     thermal                         (1048576),
     presentationAttackData          (2097152),
     } (0..16777215)
     -- To include multiple biometric types in one instance of the
     -- BiometricType field, use the bitwise OR of each of the values.
     -- For example, for a BDB that will contain face and fingerprint
     -- use BiometricType = multipleBiometricTypes + face + finger (11)

BiometricSubtype ::= INTEGER
     {
```

```
noValueAvailable                (0),
left                            (1),
right                           (2),
thumb                           (4),
leftThumb                       (5),
rightThumb                      (6),
index                           (8),
leftIndex                       (9),
rightIndex                      (10),
middle                          (16),
leftMiddle                      (17),
rightMiddle                     (18),
ring                            (32),
leftRing                        (33),
rightRing                       (34),
little                          (64),
leftLittle                      (65),
rightLittle                     (66),
palm                            (132),
leftPalm                        (133),
rightPalm                       (134),
backOfHand                      (136),
leftBackOfHand                  (137),
rightBackOfHand                 (138),
wrist                           (144),
leftWrist                       (145),
rightWrist                      (146),

-- The following values are valid combinations
-- of biometric subtypes within a single byte field. The
-- values are the bitwise OR of other biometric subtype values.

leftAndRight                    (3),
bothThumbs                      (7),
bothIndex                       (11),
thumbIndex                      (12),
leftThumbIndex                  (13),
rightThumbIndex                 (14),
bothThumbIndex                  (15),
bothMiddle                      (19),
thumbMiddle                     (20),
leftThumbMiddle                 (21),
rightThumbMiddle                (22),
bothThumbMiddle                 (23),
indexMiddle                     (24),
leftIndexMiddle                 (25),
rightIndexMiddle                (26),
bothIndexMiddle                 (27),
thumbIndexMiddle                (28),
leftThumbIndexMiddle            (29),
rightThumbIndexMiddle           (30),
bothThumbIndexMiddle            (31),
bothRing                        (35),
thumbRing                       (36),
leftThumbRing                   (37),
rightThumbRing                  (38),
bothThumbRing                   (39),
ringIndex                       (40),
leftRingIndex                   (41),
rightRingIndex                  (42),
bothRingIndex                   (43),
thumbIndexRing                  (44),
leftThumbIndexRing              (45),
rightThumbIndexRing             (46),
bothThumbIndexRing              (47),
middleRing                      (48),
leftMiddleRing                  (49),
rightMiddleRing                 (50),
bothMiddleRing                  (51),
thumbMiddleRing                 (52),
leftThumbMiddleRing             (53),
```

```
    rightThumbMiddleRing            (54),
    bothThumbMiddleRing             (55),
    indexMiddleRing                 (56),
    leftIndexMiddleRing             (57),
    rightIndexMiddleRing            (58),
    bothIndexMiddleRing             (59),
    thumbIndexMiddleRing            (60),
    leftThumbIndexMiddleRing        (61),
    rightThumbIndexMiddleRing       (62),
    bothThumbIndexMiddleRing        (63),
    bothLittle                      (67),
    thumbLittle                     (68),
    leftThumbLittle                 (69),
    rightThumbLittle                (70),
    bothThumbLittle                 (71),
    indexLittle                     (72),
    leftIndexLittle                 (73),
    rightIndexLittle                (74),
    bothIndexLittle                 (75),
    thumbIndexLittle                (76),
    leftThumbIndexLittle            (77),
    rightThumbIndexLittle           (78),
    bothThumbIndexLittle            (79),
    middleLittle                    (80),
    leftMiddleLittle                (81),
    rightMiddleLittle               (82),
    bothMiddleLittle                (83),
    thumbMiddleLittle               (84),
    leftThumbMiddleLittle           (85),
    rightThumbMiddleLittle          (86),
    bothThumbMiddleLittle           (87),
    indexMiddleLittle               (88),
    leftIndexMiddleLittle           (89),
    rightIndexMiddleLittle          (90),
    bothIndexMiddleLittle           (91),
    thumbIndexMiddleLittle          (92),
    leftThumbIndexMiddleLittle      (93),
    rightThumbIndexMiddleLittle     (94),
    bothThumbIndexMiddleLittle      (95),
    ringLittle                      (96),
    leftRingLittle                  (97),
    rightRingLittle                 (98),
    bothRingLittle                  (99),
    thumbRingLittle                 (100),
    leftThumbRingLittle             (101),
    rightThumbRingLittle            (102),
    bothThumbRingLittle             (103),
    indexRingLittle                 (104),
    leftIndexRingLittle             (105),
    rightIndexRingLittle            (106),
    bothIndexRingLittle             (107),
    thumbIndexRingLittle            (108),
    leftThumbIndexRingLittle        (109),
    rightThumbIndexRingLittle       (110),
    bothThumbIndexRingLittle        (111),
    middleRingLittle                (112),
    leftMiddleRingLittle            (113),
    rightMiddleRingLittle           (114),
    bothMiddleRingLittle            (115),
    thumbMiddleRingLittle           (116),
    leftThumbMiddleRingLittle       (117),
    rightThumbMiddleRingLittle      (118),
    bothThumbMiddleRingLittle       (119),
    indexMiddleRingLittle           (120),  -- four finger "slap"
    leftIndexMiddleRingLittle       (121),
    rightIndexMiddleRingLittle      (122),
    bothIndexMiddleRingLittle       (123),
    allFingers                      (124),
    leftAllFingers                  (125),
    rightAllFingers                 (126),
    bothAllFingers                  (127),  -- tenprint
```

```
   bothPalms                    (135),
   bothBackOfHand               (139),
   palmBackOfHand               (140),
   leftPalmBackOfHand           (141),
   rightPalmBackOfHand          (142),
   bothPalmBackOfHand           (143),
   bothWrist                    (147),
   palmWrist                    (148),
   leftPalmWrist                (149),
   rightPalmWrist               (150),
   bothPalmWrist                (151),
   backOfHandWrist              (152),
   leftBackOfHandWrist          (153),
   rightBackOfHandWrist         (154),
   bothBackOfHandWrist          (155),
   palmBackOfHandWrist          (156),
   leftPalmBackOfHandWrist      (157),
   rightPalmBackOfHandWrist     (158),
   bothPalmBackOfHandWrist      (159)
   } (0 .. 255)
   -- For implementations that include multiple biometric subtypes
   -- in a single BiometricSubtype byte field, only those combinations
   -- listed above are defined. To specify combinations not defined
   -- above, multiple instances of the BiometricSubtype field are needed

ChallengeResponse ::= OCTET STRING (SIZE(0..65535))
     -- A patron format that uses this type shall specify
     -- its encoding for noValueAvailable

BDBIndex ::= OCTET STRING (SIZE(0..65535))
     -- A patron format that uses this type shall specify
     -- its encoding for noValueAvailable

ProcessedLevel ::= ENUMERATED
   {
   noValueAvailable  (0),
   raw               (1),
   intermediate      (2),
   processed         (3)
   }

Purpose ::= ENUMERATED
   {
   noValueAvailable  (0),
   verify            (1),
   identify          (2),
   enrol             (3),
   enrolVerify       (4),
   enrolIdentify     (5),
   audit             (6)
   }

QualityNotSupported::=  INTEGER (254)
NoValueAvailable::=     INTEGER (255)
Quality::= INTEGER (0..100 | QualityNotSupported | NoValueAvailable )

ConstructedQuality::= SEQUENCE
   {
   organization    [0]   INTEGER (0..65535),
   id              [1]   INTEGER (0..65535),
   score           [2]   Quality
   }

BDBFormat ::= SEQUENCE
   {
   organization      INTEGER (0..65535),
   id                INTEGER (0..65535)
   }

RegistryID ::= SEQUENCE
   {
```

```
    organization      INTEGER (1..65535),
    id                INTEGER (1..65535)
      -- id is a model number or an algorithm identifier, or anything
      -- identifying a product line, but never a serial number
    }

  Creator ::= UTF8String
       -- A patron format that uses this type shall specify
       -- its encoding for noValueAvailable

  BIRIndex ::= OCTET STRING (SIZE(0..65535))
       -- A patron format that uses this type shall specify
       -- its encoding for noValueAvailable

  Payload ::= OCTET STRING
       -- A patron format that uses this type shall specify
       -- its encoding for noValueAvailable

  ValidityPeriod ::= SEQUENCE
    {
    notValidBefore     DATE-TIME OPTIONAL,
    notValidAfter      DATE-TIME OPTIONAL
    }

  Version ::= SEQUENCE
    {
    major       INTEGER (0..255),
    minor       INTEGER (0..255)
    --noValueAvailable is encoded by {major 0, minor 0}
    }

END
```

7 Patron format specification: Simple bit-oriented patron format (deprecated)

7.1 Patron

ISO/IEC JTC 1/SC 37

NOTE This patron format is deprecated.

7.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

7.3 Patron format name

ISO/IEC JTC 1/SC 37 simple bit-oriented patron format

7.4 Patron format type

1 (0001Hex). This has been registered in accordance with ISO/IEC 19785-2.

7.5 ASN.1 OID for this patron format

{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) simple-bit-oriented(1)}

or, in XML value notation,

<OBJECT_IDENTIFIER>1.1.19785.0.257.1.1</OBJECT_IDENTIFIER>

7.6 Domain of use

This patron format has been deprecated due to a lack of adoption and technical issues with the definition of the patron format (it is not clear how to specify the BDB format owner or the BDB format type).

This patron format defines minimum length simple BIRs that may be of general utility in domains of use that wish to minimize data overhead in order to reduce storage or transfer bandwidth and processing costs at the expense of information content.

BIRs coded with this patron format shall be simple BIRs, not containing SB, and being able to accept loss of byte alignment.

As this patron format does not support the use of SB, CBEFF_BDB_encryption_options and CBEFF_BIR_integrity_options, are intrinsically considered set to, respectively, NO ENCRYPTION and NO INTEGRITY.

Only a single BDB is permitted for each BIR of this patron format.

7.7 Version identifier

This patron format specification has a version identifier of (major 0, minor 0).

7.8 CBEFF version

This specification conforms to CBEFF version (major 2, minor 0).

7.9 General

Patron Format Structure Type: Simple CBEFF BIR without self-identification.

Clause 7 defines a minimum simple bit-oriented patron format. The formal specification of the actual bits-on-the-line for this patron format is provided by a reference to the ASN.1 encoding rules.

The simple bit-oriented patron format is formally defined as the unaligned variant of the ASN.1 packed encoding rules (ASN.1 PER) defined in ISO/IEC 8825-2, applied to the **`MinimumElementsBitOriented`** type specified in 7.10.

An example of the encoding produced by an assignment of abstract values for this patron format, showing the size and encoding of each field of the SBH, is given in Table 1. The size of the SBH is **one octet** for BDB formats standardized by ISO/IEC JTC 1/SC 37, but can be greater for other BDB formats. The BDB length encoding is one octet if the BDB encoding is less than 128 bytes, making the total size of the BIR 2 octets plus the length of the BDB.

Table 1 — Minimal simple bit-oriented SBH for ISO/IEC JTC 1/SC 37 BDB format (1 byte)

BDB Format owner is SC 37 0	BDB Format type is <64 0	BDB Format type value
← 8 bits →		

7.10 ASN.1 Specification

The following ASN.1 module can be retrieved from http://standards.iso.org/iso-iec/19785/-3/ed-3/en.

```
CBEFF-SIMPLE-BIT-ORIENTED
{ iso(1) standard(0) cbeff(19785) modules(0)
minimumElementsWithDefaultsBitOriented(2)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

IMPORTS BiometricDataBlock
      FROM CBEFF-DATA-ELEMENTS

MinimumElementsBitOriented ::= SEQUENCE {

      /* This patron format contains only mandatory data elements
      and uses bit-level encoding for optimal use of encoding space.*/

      /* This patron format supports only the abstract values
      NO ENCRYPTION and NO INTEGRITY, which are encoded as zero length
      fields.*/

      format  SEQUENCE {
                owner    INTEGER (0.. 65535) DEFAULT 257,
                          -- 257 is the biometric organization
                          -- identifier of ISO/IEC JTC 1/SC 37
                type     INTEGER (0..63, ..., 64 .. 65535)},
      bdb               BiometricDataBlock (SIZE(0..127, ..., 128 .. MAX))}
END
```

8 Patron format specification: Simple byte-oriented patron format

8.1 Patron

ISO/IEC JTC 1/SC 37.

8.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

8.3 Patron format name

ISO/IEC JTC 1/SC 37 simple byte-oriented patron format.

8.4 Patron format type

2 (0002Hex). This has been registered in accordance with ISO/IEC 19785-2.

8.5 ASN.1 OID for this patron format

{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) simple-byte-oriented(2)}

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.2</OBJECT_IDENTIFIER>
```

8.6 Domain of use

Clause 8 provides the definition of a minimum byte-oriented patron format for simple BIR structures that may be of general utility in domains of use that wish to minimise the overhead of the SBH in order to reduce storage or transfer bandwidth and processing costs at the expense of information content, and that require byte alignment, and do not require ENCRYPTION and INTEGRITY. Only a single BDB is permitted for each BIR of this patron format.

8.7 Version identifier

This patron format does not include a version identifier and the version is considered to be (major 0, minor 0). Therefore, this patron format cannot be updated in the future.

8.8 CBEFF version

This specification conforms to CBEFF version (major 2, minor 0) onwards.

8.9 General

Patron Format Structure Type: Simple CBEFF BIR without self-identification.

Formally, the fields of this patron format are defined by application of the unaligned variant of the ASN.1 packed encoding rules defined in ISO/IEC 8825-2 to the presence bit-oriented patron format type specified in 8.10.

An example of the encoding produced by an assignment of abstract values for this patron format, showing the size and encoding of each field of the SBH, is given in Table 2. The size of the SBH for this patron format is fixed at 4 octets.

Table 2 — Simple byte-oriented SBH

BDB Format Owner (16 bits)	BDB Format Type (16 bits)

8.10 Patron format specification

An instance of a BIR shall contain the fields specified in Table 3, in exactly the same order and with no gaps between the fields. This patron format contains only mandatory data elements and uses byte-level encoding. This patron format supports only the abstract values NO ENCRYPTION and NO INTEGRITY, which are encoded as zero length fields.

Table 3 — Specification of simple byte-oriented patron format

CBEFF data element name	Field name	Length	Abstract values	Encoding
CBEFF_BDB_format_owner	Format owner	2 bytes	integer 1...65535	1...65535
CBEFF_BDB_format_type	Format type	2 bytes	integer 1...65535	1...65535
BDB	bdb	variable	See NOTE	See NOTE
NOTE The BDB format is not specified by this patron format specification.				

The following ASN.1 module can be retrieved from http://standards.iso.org/iso-iec/19785/-3/ed-3/en.

8.11 ASN.1 Specification

```
CBEFF-SIMPLE-BYTE-ORIENTED
{iso(1) standard(0) cbeff(19785) modules(0)
minimumElementsWithDefaultsByteOriented(3)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

IMPORTS BiometricDataBlock
      FROM CBEFF-DATA-ELEMENTS

MinimumElementsByteOriented ::= SEQUENCE {

      /* This patron format contains only mandatory data elements
      and uses byte-level encoding.*/

      /* This patron format supports only the abstract values
      NO ENCRYPTION and NO INTEGRITY, which are encoded as zero
      length fields.*/

      format   SEQUENCE {
         owner INTEGER (1..65535) DEFAULT 257,
          -- 257 is the biometric organization identifier of
          -- ISO/IEC JTC 1/SC 37
         type     INTEGER (1..63, ..., 64 .. 65535)},
         bdb      BiometricDataBlock (SIZE(0..127, ..., 128 .. MAX))}

END
```

9 Patron format specification: Presence byte-oriented patron format

9.1 Patron

ISO/IEC JTC 1/SC 37.

9.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

9.3 Patron format name

ISO/IEC JTC 1/SC 37 presence byte oriented patron format.

9.4 Patron format type

3 (0003Hex). This has been registered in accordance with ISO/IEC 19785-2.

9.5 ASN.1 OID for this patron format

```
{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc-sc37 (257)
patron-format (1) presence-byte-oriented (3)}}
```

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.3</OBJECT_IDENTIFIER>
```

9.6 Domain of use

This patron format defines the structure of an interoperable BIR that allows the coding of as many fields as requested from those defined for the SBH and has a byte-oriented encoding to ease implementations. Only a single BDB is permitted for each BIR of this patron format.

9.7 Version identifier

This patron format specification has a version identifier of 1.

9.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0) onwards.

9.9 General

Patron Format Structure Type: Simple CBEFF BIR without self-identification

This patron format supports the 12 fixed length, optional CBEFF data elements and their abstract values that are permitted in a simple BIR format (thus excluding CBEFF_BIR_patron_format_owner and CBEFF_BIR_patron_format_type), plus CBEFF_subheader_count, plus the patron-format-defined field bdbLength. It contains an initial bit map that identifies the optional fields present in an encoded SBH. The omission of these fields either encodes NO VALUE AVAILABLE, or encodes a specific default value (specified in the patron format specification) for the CBEFF data element. The presence bit map adds two octets, so if all optional fields have the default values, the size of this SBH would be 6 octets. This patron format does not support ENCRYPTION or INTEGRITY, and thus does not support CBEFF_SB_format_owner and CBEFF_SB_format_type.

9.10 Specification

Table 4 — Specification for presence byte-oriented patron format

CBEFF data element name	Field name	Length (octets)	Abstract values and encodings[a]
n/a: patron-defined data element	present	2	
	bioType-p	1 bit	PRESENT: b'1' or
	bioSubtype-p	1 bit	NOT PRESENT: b'0'
	bdbCreation-p	1 bit	for each
	procLevel-p	1 bit	
	product-p	1 bit	
	purpose-p	1 bit	
	quality-p	1 bit	
	bdbValidity-p	1 bit	
	birCreation-p	1 bit	
	birValidity-p	1 bit	
	patronHdrVers-p	1 bit	
	vers-p	1 bit	
	subHeadCt-p	1 bit	
	bdbLength-p	1 bit	
	reserved	2 bits	b'00'
	BDB-format	4	
CBEFF_BDB_format_owner	owner	2	Integer: 0...65535
CBEFF_BDB_format_type	type	2	Integer: 0...65535
CBEFF_BDB_encryption_options	n/a	0	NO ENCRYPTION
CBEFF_BDB_integrity_options	n/a	0	NO INTEGRITY

NOTE The length, content and encoding of the BDB are not defined by CBEFF nor by this CBEFF-defined patron format specification.

[a] Encodings are specified as binary integers unless otherwise indicated.

Table 4 *(continued)*

CBEFF data element name	Field name	Length (octets)	Abstract values and encodings[a]
CBEFF_BDB_biometric_type	bioType	3	*This encoding is a 3 octet bitmap. NO VALUE AVAILABLE is encoded as all 0 bits. If MULTIPLE BIOMETRIC TYPES is set, other bits may also be set to enumerate the types contained in the BDB.* Values can be found in 6.2 under the BiometricType ASN.1 definition
CBEFF_BDB_biometric_subtype	bioSubtype	1	*This encoding is a 1 octet bitmap.* Values can be found in 6.2 under the BiometricSubtype ASN.1 definition
CBEFF_BDB_creation_date	bdbCreation	7	NO VALUE AVAILABLE: seven 0 octets date/time value: fourteen BCD digits (YYYYMMDDHHMMSS)
CBEFF_BDB_processed_level	procLevel	1	NO VALUE AVAILABLE: 0 RAW: 1 INTERMEDIATE: 2 PROCESSED: 3 RESERVED: 4...15
CBEFF_BDB_product_owner CBEFF_BDB_product_type	product owner identifier	4 2 2	Field not present corresponds to "NO VALUE AVAILABLE" Integer: 1...65535 Integer: 1...65535
CBEFF_BDB_purpose	purpose	1	NO VALUE AVAILABLE: 0 VERIFY: 1 IDENTIFY: 2 ENROL: 3 ENROL FOR VERIFICATION ONLY: 4 ENROL FOR IDENTIFICATION ONLY: 5 AUDIT: 6 RESERVED: 7...15 (shall not be encoded)
CBEFF_BDB_quality	quality	1	NO VALUE AVAILABLE: 0 QUALITY NOT SUPPORTED - BY BDB CREATOR: 255 QUALITY SUPPORTED BY BDB-CREATOR BUT NOT SET: 254 INTEGER VALUE: 0...100
CBEFF_BDB_validity_period	bdbValidity	8	NO VALUE AVAILABLE: eight 0 octets pair of dates (not before, not after): sixteen BCD digits (YYYYMMDDYYYYMMDD)
CBEFF_BIR_creation_date	birCreation	7	NO VALUE AVAILABLE: seven 0 octets date/time value: fourteen BCD digits (YYYYMMDDHHMMSS)
CBEFF_BIR_validity_period	birValidity	8	NO VALUE AVAILABLE: eight 0 octets pair of dates (not before, not after): sixteen BCD digits (YYYYMMDDYYYYMMDD)
CBEFF_patron_header_version	patronHdrVers	1	INTEGER: 0
CBEFF_version	vers	1	Major '4' and Minor '0': 40_{Hex}
CBEFF_subheader_count	subHeadCt	1	Integer: 0

NOTE The length, content and encoding of the BDB are not defined by CBEFF nor by this CBEFF-defined patron format specification.

[a] Encodings are specified as binary integers unless otherwise indicated.

Table 4 *(continued)*

CBEFF data element name	Field name	Length (octets)	Abstract values and encodings[a]
n/a: patron-defined data element	bdbLength	4	Integer: 0...4294967295
BDB	bdb	See NOTE	See NOTE
NOTE The length, content and encoding of the BDB are not defined by CBEFF nor by this CBEFF-defined patron format specification.			
[a] Encodings are specified as binary integers unless otherwise indicated.			

9.11 ASN.1 Specification

The following ASN.1 module can be retrieved from http://standards.iso.org/iso-iec/19785/-3/ed-3/en.

```
    CBEFF-PRESENCE-BYTE-ORIENTED
    { iso(1) standard(0) cbeff(19785) modules(0) presence-byte-oriented(5)}
    DEFINITIONS
    AUTOMATIC TAGS ::=
    BEGIN

    IMPORTS BDBFormat, SubheaderCount, BiometricType, BiometricSubtype,
    BDBCreationDate, ProcessedLevel, Product, Purpose, Quality,
    BDBValidityPeriod, BIRCreationDate, BIRValidityPeriod, Version,
    BiometricDataBlock
    FROM CBEFF-DATA-ELEMENTS

    PresenceByteOriented ::= SEQUENCE {

/* This patron format contains all data elements but makes full use of default values to
allow omission of fields. A bit map is present at the start of the encoding to determine
which fields are present. */

    format BDBFormat,

    fieldPresence SEQUENCE {
        bioType BOOLEAN,
        bioSubtype BOOLEAN,
        bdbCreation BOOLEAN,
        procLevel BOOLEAN,
        bdbProduct BOOLEAN,
        bdbPurpose BOOLEAN,
        bdbQuality BOOLEAN,
        bdbValidity BOOLEAN,
        birCreation BOOLEAN,
        birValidity BOOLEAN,
        patronHdrVersion BOOLEAN,
        cbeffVers BOOLEAN,
        subHeaderCount BOOLEAN,
        bdbLength BOOLEAN,
        },

        bdbFormat BDBFormat,
        bioType BiometricType DEFAULT noValueAvailable,
        bioSubtype BiometricSubtype DEFAULT noValueAvailable,
        bdbCreation BDBCreationDate OPTIONAL,
        procLevel ProcessedLevel DEFAULT noValueAvailable,
        bdbProduct Product OPTIONAL,
        bdbPurpose Purpose DEFAULT noValueAvailable,
        bdbQuality Quality DEFAULT noValueAvailable,
        bdbValidity BDBValidityPeriod OPTIONAL,
        birCreation BIRCreationDate OPTIONAL,
        birValidity BIRValidityPeriod OPTIONAL,
        patronHdrVersion Version INTEGER (0..255) DEFAULT 1,
        cbeffVers Version DEFAULT {cbeff-major 4, cbeff-minor 0},
        subHeaderCount SubheaderCount DEFAULT 0,
        bdbLength INTEGER (0..4294967295) DEFAULT 0,
```

```
        bdb BiometricDataBlock
    }

END
```

10 Patron format specification: Presence bit-oriented patron format

10.1 Patron

ISO/IEC JTC 1/SC 37.

10.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

10.3 Patron format name

ISO/IEC JTC 1/SC 37 presence bit-oriented patron format

10.4 Patron format type

4 (0004Hex). This has been registered in accordance with ISO/IEC 19785-2.

10.5 ASN.1 OID for this patron format

{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) bit-oriented(4)}

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.4</OBJECT_IDENTIFIER>
```

10.6 Domain of use

This patron format defines the structure of an interoperable BIR that allows the coding of as many fields as requested from the ones defined for the SBH. Only a single BDB is permitted for each BIR of this patron format.

10.7 Version identifier

This patron format specification has a version identifier of (major 1, minor 0).

10.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0) onwards.

10.9 General

Patron Format Structure Type: Simple CBEFF BIR without self-identification.

This patron format was updated from the previous version to include field presence (bit map) and security block fields that were missing in the previous version.

The presence bit-oriented patron format uses fixed-width bit-wide fields that are just sufficient to encode the range of abstract values in version 2 of the CBEFF specification, with some reserved abstract values for future addition of abstract values to some CBEFF data elements. Formally, the fields of this patron format are defined by application of the unaligned variant of the ASN.1 packed encoding rules defined in ISO/IEC 8825-2 to the presence bit-oriented patron format type specified in 10.10. The

presence or absence of all optional fields is indicated by a presence field bit-map, but default values are used for omitted fields.

10.10 ASN.1 Specification

The following ASN.1 module can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en.

```
        CBEFF-PRESENCE-BIT-ORIENTED
        {iso(1) standard(0) cbeff(19785) modules(0)
        allElementsWithDefaultsBitOriented(4)}
        DEFINITIONS
        AUTOMATIC TAGS ::=
        BEGIN

        IMPORTS BDBFormat, EncryptionOptions, IntegrityOptions,
        SubheaderCount, BiometricType, BiometricSubtype, ChallengeResponse,
        BDBCreationDate, BDBIndex, ProcessedLevel, Product, Purpose, Quality,
        ValidityPeriod, BIRCreationDate, Creator, BIRIndex, PatronFormat,
        Payload, Version, SBFormat, BiometricDataBlock
        FROM CBEFF-DATA-ELEMENTS

        AllElementsWithDefaultsBitOriented ::= SEQUENCE {

/* This patron format contains all data elements but makes full use of default values to
allow omission of fields. It encodes each CBEFF data element into a fixed-width bit field.
A bit map is present at the start of the encoding to determine which fields are present.
*/

        format BDBFormat,

        fieldPresence SEQUENCE {
            encryptOpts BOOLEAN,
            integOpts BOOLEAN,
            subHeaderCount BOOLEAN,
            bdbLength BOOLEAN,
            bioType BOOLEAN,
            bioSubtype BOOLEAN,
            chaResp BOOLEAN,
            bdbCreation BOOLEAN,
            bdbIndex BOOLEAN,
            procLevel BOOLEAN,
            product BOOLEAN,
            purpose BOOLEAN,
            quality BOOLEAN,
            bdbValidity BOOLEAN,
            birCreation BOOLEAN,
            birCreator BOOLEAN,
            birIndex BOOLEAN,
            patronFormat BOOLEAN,
            payload BOOLEAN,
            birValidity BOOLEAN,
            patronHdrVersion BOOLEAN,
            sbFormat BOOLEAN,
            cbeffVers BOOLEAN,
            sb OCTET STRING BOOLEAN
            },


            encryptOpts EncryptionOptions DEFAULT FALSE,
            integOpts IntegrityOptions DEFAULT FALSE,
            subHeaderCount SubheaderCount DEFAULT 0,
            bdbLength INTEGER (0..4294967295) DEFAULT 0,
            bioType BiometricType DEFAULT noValueAvailable,
            bioSubtype BiometricSubtype DEFAULT noValueAvailable,
            chaResp ChallengeResponse OPTIONAL,
            bdbCreation BDBCreationDate OPTIONAL,
            bdbIndex BDBIndex OPTIONAL,
            procLevel ProcessedLevel DEFAULT noValueAvailable,
            product Product OPTIONAL,
            purpose Purpose DEFAULT noValueAvailable,
```

```
        quality Quality DEFAULT noValueAvailable,
        bdbValidity ValidityPeriod,
        birCreation BIRCreationDate OPTIONAL,
        birCreator Creator OPTIONAL,
        birIndex BIRIndex OPTIONAL,
        patronFormat PatronFormat OPTIONAL,
        payload Payload OPTIONAL,
        birValidity ValidityPeriod,
        patronHdrVersion Version DEFAULT {header-major 1, header-minor 0},
        sbFormat SBFormat OPTIONAL,
        cbeffVers Version DEFAULT {cbeff-major 4, cbeff-minor 0},
        bdb BiometricDataBlock,
        sb OCTET STRING (SIZE(0..4294967295)) OPTIONAL
    }

    END
```

11 Patron format specification: TLV-encoded patron format, for use with smartcards or other tokens (with implicit tag allocation authority)

11.1 Patron

ISO/IEC JTC 1/SC 37.

11.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

11.3 Patron format name

ISO/IEC JTC 1/SC 37 TLV-encoded patron format, for use with smartcards or other tokens

11.4 Patron format type

5 (0005Hex). This has been registered in accordance with ISO/IEC 19785-2.

11.5 ASN.1 OID for this patron format

{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) tlv-encoded(5)}

or, in XML value notation,

1.1.19785.0.257.1.5

11.6 Domain of Use

The domain of use is applications complying with ISO/IEC 7816-11 when implicit tag allocation authority is considered. This patron format is applicable for both on-card and off-card biometric comparison and it is maintained for legacy reasons.

For the recent applications complying with ISO/IEC 7816-11, the explicit tag allocation specification shall be followed, and in such a case, the patron format to be used shall be the one defined in Clause 19.

For any other kind of application that would like to use TLV coding, patron formats defined in Clauses 16 and 17 shall be used.

NOTE For legacy reasons the technical content of this clause is unchanged since Edition 1 (ISO/IEC 19785-3:2007). Also, in order to maintain the validity of other widely implemented documents referencing this patron format, it will be preserved in Clause 11 in this and all future versions of this document. Since technical and major editorial changes cannot be made to this patron format, Annex D (informative) provides additional clarifying information that may be helpful to the reader.

11.7 Version identifier

This patron format specification has a version identifier of (major 1, minor 1).

11.8 CBEFF version

This specification conforms to CBEFF version (major 2, minor 0).

11.9 General

11.9.1 Patron Format Structure Type: Complex CBEFF BIR without self-identification

NOTE This patron format structure type is considered to be Complex CBEFF BIR, because multiple Simple BIRs (Biometric Information Template DOs) that encapsulate BDB information (BDB Reference Data, BDB Reference Data Template DOs) with its own SBHs (Biometric Header Template DOs) can be included into the Biometric Information Template Group DO.

11.9.2 This clause specifies in 11.10:

a) the CBEFF patron format named "ISO/IEC JTC 1/SC 37 TLV-encoded patron format, for use with smart cards or other tokens";

b) a TLV encoding for a number of CBEFF data elements (together with distinct TLV tags) that can be used by any TLV-encoded application that needs to make reference to such data elements.

NOTE The particular form of TLV encoding specified in this clause is that provided by the ASN.1 Basic Encoding Rules (see ISO/IEC 8825-1).

11.9.3 ISO/IEC 7816-4 and ISO/IEC 7816-11 define the necessary APDU exchanges to support:

a) On-card biometric comparison

   1) Retrieval of information from a smartcard or other token prior to an external biometric verification process.

   2) Commands for performing a biometric verification on a smartcard.

   3) Strategies for enrollment (recording of information on a smartcard or other token).

   4) Security mechanisms for establishing a trusted channel between a smartcard and an external system.

b) Off-card biometric comparison

   1) Commands for retrieval of the biometric related data from a smartcard to be used in a service system, e.g. a cross border control system.

   2) Security mechanisms for the protection of the biometric related data.

11.9.4 This version of the CBEFF patron format named "ISO/IEC JTC 1/SC 37 TLV-encoded patron format, for use with smartcards or other tokens" is formally defined as the ASN.1 type **`BiometricInformationTemplate`** (see 11.10) encoded using the ASN.1 Basic Encoding Rules (see ISO/IEC 8825-1). A diagrammatic specification of this patron format is provided in 11.11.

NOTE The term Biometric Information Template is used in ISO/IEC 7816-11. This term corresponds to a (possibly partial) Biometric Information Record and is used in this clause where the relation with ISO/IEC 7816-11 is important.

11.9.5 The components of the ASN.1 type `BiometricHeaderTemplate` (with their context-specific tags, see 11.10), encoded using the ASN.1 Basic Encoding Rules are also defined as Data Objects for use in ISO/IEC 7816-11.

NOTE The term Biometric Header Template is used in ISO/IEC 7816-11, together with the abbreviation BHT. This term and the abbreviated term are synonymous with Standard Biometric Header and SBH, and are used in this clause where the relation with ISO/IEC 7816-11 is important.

11.9.6 The term Biometric Information Data Object is used in ISO/IEC 7816-11 to refer to the TLV-encoding of a specific ASN.1 type. Table 5 records the Biometric Information Data Objects that are defined in 11.10 for use in ISO/IEC 7816-11.

NOTE As ISO/IEC 7816-11 normally describes tags by their hex values, these are included here.

Table 5 — Biometric Information Data Objects with the Biometric Header Template Data Objects defined in 11.10

Biometric Information Data Object	ASN.1 type or component name (see 11.10)	Tag
Biometric Information Template	`BiometricInformationTemplate`	$7F60_{Hex}$
Biometric Information Template Group	`GroupBIT`	$7F61_{Hex}$
BDB Reference Data	`bdbReferenceData`	$5F2E_{Hex}$
BDB Reference Data Template	`bdbReferenceDataTemplate`	$7F2E_{Hex}$
BIR Payload (primitive / constructed)	`birPayLoad`	53_{Hex} / 73_{Hex}
For use only in Biometric Reference Data Template:		
BDB Reference Data with Standardised Format (Primitive)	`standardBDBReferenceData`	81_{Hex}
BDB Reference Data with Standardised Format (Constructed)	`standardBDBReferenceDataTemplate`	$A1_{Hex}$
BDB Reference Data with Proprietary Format (Primitive)	`proprietaryBDBReferenceData`	82_{Hex}
BDB Reference Data with Proprietary Format (Constructed)	`proprietaryBDBReferenceDataTemplate`	$A2_{Hex}$
For use only in Biometric Information Template:		
Biometric Header Template	`biometricHeaderTemplate`	$A1_{Hex}$
Algorithm Reference	`algorithmReference`	80_{Hex}
Reference Data Qualifier	`referenceDataQualifier`	83_{Hex}
For use only in Biometric Header Template (implicit tag allocation):		
Patron Header Version (as primitive)	`patronHeaderVersion`	80_{Hex}
BDB Biometric Type	`bdbBiometricType`	81_{Hex}
BDB Biometric Subtype	`bdbBiometricSubType`	82_{Hex}
BDB Creation Date	`bdbCreationDate`	83_{Hex}
BIR Creator	`birCreator`	84_{Hex}
BDB Validity Period (as primitive)	`bdbValidityPeriod`	85_{Hex}
BDB PID (as primitive)	`bdbPID`	86_{Hex}
BDB Format Owner	`bdbFormatOwner`	87_{Hex}
BDB Format Type	`bdbFormatType`	88_{Hex}
BIR Index	`birIndex`	90_{Hex}

Table 5 *(continued)*

Biometric Information Data Object	ASN.1 type or component name (see 11.10)	Tag
Comparison Algorithm Parameters (primitive / constructed)	`comparisonAlgParameters`	91_{Hex} / $B1_{Hex}$

11.9.7 CBEFF-compliance requires the NO VALUE AVAILABLE abstract value to be supported by this patron format for CBEFF data elements that are not fully supported. To meet this requirement, tags are assigned and reserved by this document in Table 6 for Data Objects appearing in the Biometric Header Template. These tag values shall not be assigned by any other tag authority to Data Objects that can appear in the Biometric Header Template. These Data Objects shall never appear in encodings of this version of this patron format, and their omission represents the NO VALUE AVAILABLE abstract value of the corresponding CBEFF data element.

NOTE Subsequent versions of this patron format can define these Data Objects to support the corresponding CBEFF data element.

Table 6 — Reserved tag values (implicit tag allocation)

ASN.1 tag value	Tag value	Corresponding CBEFF data element
`[19]`	93_{Hex}	CBEFF_BDB_challenge_response
`[20]`	94_{Hex}	CBEFF_BDB_index
`[21]`	95_{Hex}	CBEFF_BDB_processed_level
`[22]`	96_{Hex}	CBEFF_BDB_purpose
`[23]`	97_{Hex}	CBEFF_BDB_quality
`[24]`	98_{Hex}	CBEFF_BIR_creation_date
`[25]`	99_{Hex}	CBEFF_BIR_patron_format_owner
`[26]`	$9A_{Hex}$	CBEFF_BIR_patron_format_type
`[27]`	$9B_{Hex}$	CBEFF_BIR_validity_period (as primitive)
`[28]`	$9C_{Hex}$	CBEFF_version (as primitive)

11.10 ASN.1 specification

The ASN.1 module given in 11.10 can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en.

```
CBEFF-SMARTCARD-BIDO
-- The abbreviation BIDO is used for Biometric Information Data Object
{iso(1) standard(0) cbeff(19785) modules(0) types-for-smartcard(8)}
DEFINITIONS
IMPLICIT TAGS ::=

BEGIN

-- In all cases, omission of an optional component that represents a
-- CBEFF data element is the encoding of the NO VALUE AVAILABLE for
-- that data element.

PatronHeaderVersion    ::= OCTET STRING (SIZE(2))
        -- CBEFF_patron_header_version
        -- The first octet encodes the major version number
        -- The second octet encodes the minor version number

BiometricType          ::= OCTET STRING (SIZE(1..3))
        -- CBEFF_BDB_biometric_type
        -- The encoding of the abstract values in the value part of the
        -- TLV shall be the Recommended Encodings specified in Table 11.5.
        -- Note that this is different from the encoding of this type in
```

```
        -- clause 29.10 and in other patron formats specified in this part
        -- of ISO/IEC 19785.

BiometricSubType      ::= OCTET STRING (SIZE(1))
        -- CBEFF_BDB_biometric_subtype
        -- The encoding of the abstract values in the value part of the
        -- TLV shall be the Recommended Encodings specified in Table 11.6.
        -- Note that this is different from the encoding of this type in
        -- Table A.2 and in other patron formats specified in this part of
        -- ISO/IEC 19785.

BCDTime               ::= OCTET STRING (SIZE(7))
        -- BCD encoded timestamp with format ' YYYYMMDDHHMMSS'

Creator               ::= UTF8String
        -- CBEFF_BIR_creator

BCDDate               ::= OCTET STRING (SIZE(4))
        -- BCD encoded date with format 'YYYYMMDD'

BCDDatePeriod         := OCTET STRING (SIZE(8))
        -- Two concatenated BCD encoded dates with format YYYYMMDDYYYYMMDD

ProductID             ::= OCTET STRING (SIZE(4))
        -- CBEFF_BDB_product_owner in the first two octets
        -- CBEFF_BDB_product_type in the last two octets

FormatOwner           ::= OCTET STRING (SIZE(2))
        -- CBEFF_BDB_format_owner

FormatType            ::= OCTET STRING (SIZE(2))
        -- CBEFF_BDB_format_type

BIRIndex              ::= OCTET STRING
        -- CBEFF_BIR_index

BiometricInformationTemplate ::= [APPLICATION 96] SET {
        algorithmReference [0] OCTET STRING (SIZE(1)) OPTIONAL,
        -- A non-CBEFF data element - see ISO/IEC 7816-11
        referenceDataQualifier [3] OCTET STRING (SIZE(1)) OPTIONAL,
        -- A non-CBEFF data element - see ISO/IEC 7816-11
        biometricHeaderTemplate [1] BiometricHeaderTemplate,
        bdbReferenceData [APPLICATION 46] EXTERNAL OPTIONAL,
        -- A CBEFF BDB, mandatory for off-card-matching
        birPayLoad [APPLICATION 19] OCTET STRING OPTIONAL
        -- CBEFF_BIR_payload, contents defined by ISO/IEC 7816-11
        }

GroupBIT ::= [APPLICATION 97] SET OF BiometricInformationTemplate

BiometricHeaderTemplate ::= SET {
        patronHeaderVersion [0] PatronHeaderVersion
        DEFAULT'0101'H,
        -- The absence of this Data Object represents NO VALUE AVAILABLE
        bdbBiometricType [1] BiometricType OPTIONAL,
        bdbBiometricSubType [2] BiometricSubType OPTIONAL,
        -- Required to be absent unless bdbBiometricType is present
        bdbCreationDate [3] BCDTime OPTIONAL,
        -- CBEFF_BDB_creation_date
        birCreator [4] Creator OPTIONAL,
        bdbValidityPeriod [5] BCDDatePeriod OPTIONAL,
        bdbPID [6] ProductID OPTIONAL,
        bdbFormatOwner [7] FormatOwner,
        bdbFormatType [8] FormatType,
        birIndex [16] BIRIndex OPTIONAL,
        matchingAlgParameters [17] OCTET STRING OPTIONAL
        -- A non-CBEFF data element - see ISO/IEC 7816-11
        }
END
```

11.11 Tabular representations for information

11.11.1 The Biometric Information Template used for on-card biometric comparison

Table 7 shows the Biometric Information Template used for on-card biometric comparison. The Biometric Information Template has the following substructure:

a) Data Objects (with tags 80_{Hex} and 83_{Hex}) as defined in ISO/IEC 7816-11, containing values relevant for interindustry commands used for biometric verification as defined in ISO/IEC 7816-4;

b) Biometric Header Template (BHT) with tag $A1_{Hex}$ as defined in ISO/IEC 7816-11. The tag allocation authority for the Data Objects nested in the BHT is ISO/IEC JTC 1/SC 37 (default tag allocation authority);

NOTE The ASN.1 OID for this tag allocation authority is `{iso(1) standard(1) cbeff(19785) part(3) tag-allocation(1) clause-11(0)}`.

c) Data Objects for CBEFF data elements that are relevant for on-card biometric comparison;

d) Biometric Data Objects which are on-card biometric comparison specific and are defined in this clause (the biometric comparison algorithm parameters Data Object with tag 91_{Hex} or $B1_{Hex}$).

Table 7 — The Biometric Information Template used for on-card biometric comparison (with implicit tag allocation authority)

Tag	Length	Value					Presence
$7F60_{Hex}$	Var.	Biometric Information Template					
		Tag	**Length**	**Value**			
		80_{Hex}	1	Algorithm reference for use in the VERIFY / EXT. AUTHENTICATE / MANAGE SE command as defined in ISO/IEC 7816-4			Optional
		83_{Hex}	1	Reference data qualifier for use in the VERIFY / EXT. AUTHENTICATE / MANAGE SE command as defined in ISO/IEC 7816-4			Optional
		$A1_{Hex}$	Var.	Biometric Header Template (BHT) Tag allocation authority: ISO/IEC JTC 1/SC 37			Mandatory
				Tag	**Length**	**Value**	
				80_{Hex}	2	CBEFF_patron_header_version (default 0101_{Hex})	Mandatory (if absent, the default value applies)
				90_{Hex}	Var.	CBEFF_BIR_index, unique identifier used for referencing this biometric data set in an application context outside the card	Optional
				81_{Hex}	1-3	CBEFF_BDB_biometric_type, see Table 9.	Optional
				82_{Hex}	1	CBEFF_BDB_biometric_subtype, see Table 10.	Optional, use with biometric type only
				83_{Hex}	7	CBEFF_BDB_creation_date, creation date and time of biometric reference data: fourteen BCD digits (YYYYMMDDHHMMSS)	Optional
				84_{Hex}	Var.	CBEFF_BIR_creator	Optional
				85_{Hex}	8	CBEFF_BDB_validity_period, a pair of dates (not before, not after): sixteen BCD digits (YYYYMMDDYYYYMMDD)	Optional

NOTE 1 In Table 7 the biometric data block as defined in ISO/IEC 19785-1 is not present, since the biometric reference data are stored separately in the card and not in this Biometric Information Template. The biometric verification data (format owner and format type indicated in the Data Objects with tag '87' and '88') have to be presented to the card using e.g. a ISO/IEC 7816 VERIFY command.

NOTE 2 In Table 7 no payload is present, since usually access to a payload, if used by the application, is granted after successful completion of the biometric verification. The payload may be retrieved using ISO/IEC 7816 interindustry access commands like GET DATA or READ BINARY.

NOTE 3 The outside world uses format owner / format type for identifying the required structure for the verification data. The matching algorithm in the card, which is able to process the verification data according to format owner / format type, is addressed by the algorithm reference, if the algorithm reference Data Object is present.

NOTE 4 The biometric comparison algorithm Data Object provides any special parameters of an on-card biometric comparison algorithm implementation, e.g. maximum number of minutiae expected in the biometric verification data. The content of this Data Object is defined by the BDB format owner (see e.g. ISO/IEC 19794-2).

Table 7 *(continued)*

Tag	Length	Value					Presence
				86_{Hex}	4	CBEFF_BDB_product_owner, CBEFF_BDB_product_type. Concatenation of product owner and product type, identifying the product that created the biometric reference data	Optional
				87_{Hex}	2	CBEFF_BDB_format_owner, format owner of the biometric verification data, value assigned by the biometrics registration authority, see NOTE 1 and NOTE 3	Mandatory
				88_{Hex}	2	CBEFF_BDB_format_type, format type of biometric verification data, specified by format owner, see NOTE 1 and NOTE 3	Mandatory
				91_{Hex} / $B1_{Hex}$	Var.	Biometric comparison algorithm parameters (primitive / constructed), see NOTE 4	Optional

NOTE 1 In Table 7 the biometric data block as defined in ISO/IEC 19785-1 is not present, since the biometric reference data are stored separately in the card and not in this Biometric Information Template. The biometric verification data (format owner and format type indicated in the Data Objects with tag '87' and '88') have to be presented to the card using e.g. a ISO/IEC 7816 VERIFY command.

NOTE 2 In Table 7 no payload is present, since usually access to a payload, if used by the application, is granted after successful completion of the biometric verification. The payload may be retrieved using ISO/IEC 7816 interindustry access commands like GET DATA or READ BINARY.

NOTE 3 The outside world uses format owner / format type for identifying the required structure for the verification data. The matching algorithm in the card, which is able to process the verification data according to format owner / format type, is addressed by the algorithm reference, if the algorithm reference Data Object is present.

NOTE 4 The biometric comparison algorithm Data Object provides any special parameters of an on-card biometric comparison algorithm implementation, e.g. maximum number of minutiae expected in the biometric verification data. The content of this Data Object is defined by the BDB format owner (see e.g. ISO/IEC 19794-2).

The use of the Biometric Information Template shown in Table 7, which is intended to be retrieved prior to a biometric user verification, and the verification process itself with the related interindustry commands and their security requirements, are specified in ISO/IEC 7816-11 for the case of implicit tag allocation.

11.11.2 The Biometric Information Template used for off-card biometric comparison

Table 8 shows the Biometric Information Template used for off-card biometric comparison. The Biometric Information Template has the following substructure:

a) Biometric Header Template (BHT) with tag $A1_{Hex}$ as defined in ISO/IEC 7816-11. The tag allocation authority for the Data Objects nested in the BHT is ISO/IEC JTC 1/SC 37 (default tag allocation authority);

   NOTE The ASN.1 OID for this tag allocation authority is `{iso(1) standard(1) cbeff(19785) part(3) tag-allocation(1) clause-11(0)}`.

b) Biometric Data Objects for CBEFF data elements that are relevant for off-card biometric comparison;

c) A biometric reference Data Object (primitive or constructed) with tags as defined in ISO/IEC 7816-11, representing the CBEFF Biometric Data Block (BDB);

d) Payload as the content of an optional Data Object with tag 53_{Hex} or 73_{Hex} as defined in ISO/IEC 7816-6.

The use of the data structure shown in Table 8 is not restricted to smartcards. The data structure may also be used in other types of cards, e.g. magnetic stripe cards, optical memory cards or cards with 2-dimensional barcodes.

Table 8 — The Biometric Information Template used for off-card biometric comparison (with implicit tag allocation authority)

Tag	Length	Value					Presence
$7F60_{Hex}$	var.	Biometric Information Template					
		Tag	Length	Value			
		$A1_{Hex}$	var.	Biometric Header Template (BHT) Tag allocation authority: ISO/IEC JTC 1/SC 37			Mandatory
				Tag	**Length**	**Value**	
				80_{Hex}	2	CBEFF_patron_header_version (default '0101')	Mandatory (if absent, the default value applies)
				81_{Hex}	1-3	CBEFF_BDB_biometric_type, see Table 9.	Optional
				82_{Hex}	1	CBEFF_BDB_biometric_subtype, see Table 10.	Optional, use together with biometric type only
				83_{Hex}	7	CBEFF_BDB_creation_date creation date and time of biometric reference data: fourteen BCD digits (YYYYMMDDHHMMSS)	Optional
				84_{Hex}	var.	CBEFF_BIR_creator	Optional
				85_{Hex}	8	CBEFF_BDB_validity_period, a pair of dates (not before, not after): sixteen BCD digits (YYYYMMDDYYYYMMDD)	Optional
				86_{Hex}	4	CBEFF_BDB_product owner, CBEFF_BDB_product_type. Concatenation of product owner and product type, identifying the product that created the biometric reference data	Optional
				87_{Hex}	2	CBEFF_BDB_format_owner, format owner of the biometric reference data, value assigned by the biometrics registration authority	Mandatory
				88_{Hex}	2	CBEFF_BDB_format_type, format type of biometric reference data, specified by format owner	Mandatory
				90_{Hex}	var.	CBEFF_BIR_index, unique identifier used for referencing this biometric data set in an application context outside the card	Optional
		$5F2E_{Hex}$ / $7F2E_{Hex}$	var.	CBEFF BDB, biometric reference data (primitive / constructed); see note			Mandatory
		53_{Hex} / 73_{Hex}	var.	CBEFF_BIR_payload, optional data for payload (primitive / constructed)			Optional

NOTE The structure of the biometric reference data in the case of constructed allows the integration of a biometric challenge for user prompting (e.g. a phrase to be spoken) and a concatenation of biometric data with standardized and proprietary structure. The respective Data Objects are specified in ISO/IEC 7816-11.

11.12 The Biometric Information Template Group Data Object

Several Biometric Information Templates may be nested in a Biometric Information Template Group (tag $7F61_{Hex}$ as defined in ISO/IEC 7816-11). The construction and use of Biometric Information Template Group Data Objects both for on-card and off-card biometric comparison are specified in ISO/IEC 7816-11.

11.13 Abstract values and encodings for biometric type and subtype

Table 9 — CBEFF_BDB_biometric_type abstract values and encodings

Abstract value name	Value encoding	Subtype applies
NO VALUE AVAILABLE	000000_{Hex}	
MULTIPLE BIOMETRIC TYPES	01_{Hex}	
FACE	02_{Hex}	
VOICE	04_{Hex}	
FINGER	08_{Hex}	x
IRIS	10_{Hex}	x
RETINA	20_{Hex}	x
HAND GEOMETRY	40_{Hex}	x
SIGNATURE/SIGN	80_{Hex}	
KEYSTROKE	0100_{Hex}	
LIP MOVEMENT	0200_{Hex}	
THERMAL FACE	0400_{Hex}	
THERMAL HAND	0800_{Hex}	
GAIT	1000_{Hex}	
BODY ODOR	2000_{Hex}	
DNA	4000_{Hex}	
EAR	8000_{Hex}	x
FINGER GEOMETRY	010000_{Hex}	
PALM GEOMETRY	020000_{Hex}	
VEIN PATTERN	040000_{Hex}	
FOOT PRINT	080000_{Hex}	
NOTE Only the least significant bytes (omitting leading zeros) can become part of the Biometric Header Template.		

Table 10 — CBEFF_BDB_biometric_subtype abstract values and encodings

b8 b7 b6 b5 b4 b3 b2 b1	Biometric Subtype
0 0 0 0 0 0 0 0	NO INFORMATION GIVEN
0 1	RIGHT
1 0	LEFT
0 0 0	NO MEANING
0 0 1	THUMB
0 1 0	INDEX FINGER
0 1 1	MIDDLE FINGER
1 0 0	RING FINGER
1 0 1	LITTLE FINGER
1	ALWAYS SET
x x	RESERVED FOR FUTURE USE
NOTE This optional field denotes which sample of the biometric type is to be presented (e.g. right index finger). This field is only useful in conjunction with the field CBEFF_BDB_biometric_type in cases denoted in Table 9.	

11.14 Patron format conformance statement

11.14.1 Identifying information

Required information	Patron format reference
Patron name	See 11.1
Patron identifier	See 11.2
Patron format name	See 11.3
Patron format identifier	See 11.4
Patron format ASN.1 OID	See 11.5
Domain of use description	See 11.6
Patron format version	See 11.7
CBEFF version	See 11.8

11.14.2 CBEFF-defined data elements and abstract values

CBEFF data element name	Mandatory/ optional	Patron format field name	Abstract values specified?	Encodings specified?
CBEFF_patron_header_version	Mandatory	CBEFF_patron_header_version	Yes	Yes
CBEFF_BIR_index	Optional	CBEFF_BIR_index	Yes	Yes
CBEFF_BDB_biometric_type	Optional	CBEFF_BDB_biometric_type	Yes	Yes
CBEFF_BDB_biometric_subtype	Optional	CBEFF_BDB_biometric_subtype	Yes	Yes
CBEFF_BDB_creation_date	Optional	CBEFF_BDB_creation_date	Yes	Yes
CBEFF_BIR_creator	Optional	CBEFF_BIR_creator	Yes	Yes
CBEFF_BDB_validity_period	Optional	CBEFF_BDB_validity_period	Yes	Yes
CBEFF_BDB_product_owner	Optional	CBEFF_BDB_product_owner	Yes	Yes
CBEFF_BDB_product_type	Optional	CBEFF_BDB_product_type	Yes	Yes
CBEFF_BDB_format_owner	Mandatory	CBEFF_BDB_format_owner	Yes	Yes
CBEFF_BDB_format_type	Mandatory	CBEFF_BDB_format_type	Yes	Yes
CBEFF_BIR_payload	Optional	CBEFF_BIR_payload	Yes	Yes
BDB (on-card biometric comparison)	n/a	See Note 1 of Table 7	n/a	n/a
BDB (off-card biometric comparison)	Mandatory	BDB	n/a	n/a

11.14.3 Patron-defined data elements and abstract values

Patron format data element	Mandatory/ optional	Patron format field name	Abstract values specified?	Encodings specified?
Algorithm reference	Optional	Same	Yes	Yes
Reference data qualifier	Optional	Same	Yes	Yes
Biometric comparison algorithm parameters	Optional	Same	Yes	Yes

12 Patron format specification: Complex patron format

12.1 Patron

ISO/IEC JTC 1/SC 37.

12.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

12.3 Patron format name

ISO/IEC JTC 1/SC 37 complex patron format.

12.4 Patron format type

6 (0006Hex). This has been registered in accordance with ISO/IEC 19785-2.

12.5 ASN.1 OID for this patron format

```
{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257)
patron-format(1) complex(6)}
```

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.6</OBJECT_IDENTIFIER>
```

12.6 Domain of use

This clause provides a definition of a patron format that may be of general utility to applications that need to carry one or more BIRs (of either the same or different patron formats) in a single complex BIR structure, with explicit identification of the patron format(s) being used.

12.7 Version identifier

This patron format specification has a version identifier of 2.

NOTE The patron format version was increased from 1 to 2 because the QUALITY NOT SUPPORTED BY BDB CREATOR and QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET values were modified. Additionally, the CBEFF_biometric_type and CBEFF_biometric_subtype values were changed.

12.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0) onwards.

12.9 General

12.9.1 Patron Format Structure Type: Complex CBEFF BIR without self-identification

12.9.2 This patron format supports all the mandatory and optional data elements specified in ISO/IEC 19785-1 except the following ones: capture device owner and identifier, feature extraction algorithm owner and identifier, biometric comparison algorithm owner and identifier, quality algorithm owner and identifier, compression algorithm owner and identifier and PAD mechanism vendor and identifier. It can support either a simple BIR or a complex BIR structure where each intermediate node or leaf of the structure is itself a BIR (called a "child BIR") and can be represented in any patron format.

12.9.3 The patron format of each child BIR is explicitly identified in its parent by a pair: patron format owner / patron format type, and can be either this patron format (in which case the child BIR may itself

have children), or a different patron format (in which case the child BIR is considered a leaf of this patron format although it may be a complex BIR in its own regard).

12.9.4 Most fields in this patron format are optional. The presence of each optional field is encoded as a single bit of a 24-bit field ("fieldPresence") at the beginning of the format, which has one bit for each optional field defined in the patron format. The bit value '1' in a given position of that field means that the corresponding field is present in the BIR instance.

12.9.5 All character strings and octet strings are preceded by a length prefix, which can be one, two, or four octets long, as specified for each field.

12.9.6 All integer values, including lengths, are encoded in big-endian order.

12.9.7 Dates and date intervals are encoded as character strings in a way conforming to the ISO 8601 series.

12.9.8 An instance of a BIR or child BIR contains either a BDB or one or more BIR children, but never contains both.

12.10 Specification

An instance of a BIR shall contain the fields specified below, in exactly the same order and with no gaps between the fields.

Table 11 — Specifications for complex patron format

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
The following fields shall occur at most once			
CBEFF_patron_header_version	patronHeaderVersion	1, mandatory	2
CBEFF_version	cbeffVersion	1, mandatory	Major '4' and Minor '0': 40_{Hex} (64)

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

Table 11 *(continued)*

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
not a standard CBEFF data element	fieldPresence	3, mandatory	A 24-bit field containing one bit for each optional field in the patron format. The bit value '1' means that the corresponding field is present in the BIR instance. Bit position (1=most significant, 24=least significant) and corresponding optional field: 1 bdbFormat Owner & Type 2 bdbEncryption 3 bdbBiometricType 4 bdbBiometricSubtype 5 bdbChallengeResponse 6 bdbCreationDate 7 bdbIndex 8 bdbProcessedLevel 9 bdbProduct Owner & Type 10 bdbPurpose 11 bdbQuality 12 bdbValidityPeriod 13 birCreationDate 14 birCreator 15 birIndex 16 birPayload 17 birValidityPeriod 18 sbFormat Owner & Type 19 bdb 20 sb 21..24 not used (shall be '0')
CBEFF_BDB_format_owner	bdbFormatOwner	2, mandatory if a BDB is present, optional if a BDB is not present	0..65535
CBEFF_BDB_format_type	bdbFormatType	2, mandatory if a BDB is present, optional if a BDB is not present	0..65535
CBEFF_BDB_encryption_options	bdbEncryption	1; mandatory if a BDB is present, otherwise required to be absent	NO ENCRYPTION: 0 ENCRYPTION: 1
CBEFF_BIR_integrity_options	birIntegrity	1, mandatory	NO INTEGRITY: 0 INTEGRITY: 1

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

Table 11 *(continued)*

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
CBEFF_BDB_biometric_type	bdbBiometricType	3	*This encoding is a 3 octet bitmap. NO VALUE AVAILABLE is encoded as all 0 bits. If MULTIPLE BIOMETRIC TYPES is set, other bits may also be set to enumerate the types contained in the BDB.* Values can be found in 6.2 under the BiometricType ASN.1 definition
CBEFF_BDB_biometric_subtype	bdbBiometricSubtype	1	*This encoding is a 1 octet bitmap.* Values can be found in 6.2 under the BiometricSubtype ASN.1 definition
CBEFF_BDB_challenge_response	bdbChallengeResponse	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets)
CBEFF_BDB_creation_date	bdbCreationDate	1 + 8..15	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters) The string shall represent a date (or date and a time of the day)[c]
CBEFF_BDB_index	bdbIndex	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets) Shall not appear in any BIR in which numChildren is not 00_{Hex}
CBEFF_BDB_processed_level	bdbProcessedLevel	1	RAW: 1 INTERMEDIATE: 2 PROCESSED: 3
CBEFF_BDB_product_owner	bdbProductOwner	2	1..65535; field not present corresponds to “NO VALUE AVAILABLE”
CBEFF_BDB_product_type	bdbProductType	2	1..65535; field not present corresponds to “NO VALUE AVAILABLE”
CBEFF_BDB_purpose	bdbPurpose	1	VERIFY: 1 IDENTIFY: 2 ENROL: 3 ENROL FOR VERIFICATION ONLY: 4 ENROL FOR IDENTIFICATION ONLY: 5 AUDIT: 6
CBEFF_BDB_quality	bdbQuality	1	QUALITY NOT SUPPORTED BY BDB CREATOR: 254 QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET: 255 INTEGER VALUE: 0 – 100
CBEFF_BDB_validity_period	bdbValidityPeriod	1 + 17..31	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters) The string shall represent an interval of two dates (or date and time of the day)[d]

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

Table 11 *(continued)*

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
CBEFF_BIR_creation_date	birCreationDate	1 + 8..15	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters) The string shall represent a date (or date and a time of the day)[c]
CBEFF_BIR_creator	birCreator	2 + 0..65535	Variable-length ISO/IEC 10646 character string, encoded in UTF-8, and preceded by a 16-bit integer field containing the length of the UTF-8 encoding (octets)
CBEFF_BIR_index	birIndex	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets) Shall not inherit its value from any other level BIR
CBEFF_BIR_payload	birPayload	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets) Shall not inherit its value from any other level BIR
CBEFF_BIR_validity_period	birValidityPeriod	1 + 17..31	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters) The string shall represent an interval of two dates (or date and time of the day)[b]
CBEFF_SB_format_owner	sbFormatOwner	2	1..65535
CBEFF_SB_format_type	sbFormatType	2	1..65535
BDB	bdb	4 + 0..4294967295	Variable-length octet string, preceded by a 32-bit integer field containing the length (octets) If this field is present in a BIR instance (as indicated in bit 19 of the field *fieldPresence*), then no child BIRs shall be included (*numChildren* shall have the value 0). Otherwise, at least one child BIR shall be included (*numChildren* shall have a value greater than 0) NOTE The content and encoding of the BDB are not specified by CBEFF nor by this patron format specification
CBEFF_subheader_count	numChildren	1, mandatory	0..255
The following 3 fields shall occur as a group as many times as specified in the field numChildren (0..255)			
CBEFF_BIR_patron_format_owner	childBirPatronFormatOwner	2, mandatory if no BDB is present, otherwise required to be absent	1..65535
CBEFF_BIR_patron_format_type	childBirPatronFormatType	2, mandatory if no BDB is present, otherwise required to be absent	1..65535

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

Table 11 *(continued)*

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
not a standard CBEFF data element	childBir	4 + 0..4294967295, mandatory if no BDB is present, otherwise required to be absent	Variable-length octet string, preceded by a 32-bit integer field containing the length (octets)[e]
The following field shall occur at most once			
SB	sb	4 + 0..4294967295	Variable-length octet string, preceded by a 32-bit integer field containing the length (octets).

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

12.11 Illustrative examples

Table 12 — Simple BIR (one BDB)

Field name	Length	Abstract value	Encoding
patronHeaderVersion	1	2	02_{Hex}
cbeffVersion	1	Major 4, Minor 0	40_{Hex}
fieldPresence	3	bdbFormatOwner and Type bdbEncryption bdbBiometricType bdbQuality bdb	$E02020_{Hex}$
bdbFormatOwner	2	ISO/IEC JTC 1/SC 37	257 (0101_{Hex})
bdbFormatType	2	Face image	0008_{Hex}
bdbEncryption	1	NO ENCRYPTION	00_{Hex}
birIntegrity	1	NO INTEGRITY	00_{Hex}
bdbBiometricType	3	FACE-IMAGE	400000_{Hex}
bdbQuality	1	75/100	$4B_{Hex}$
bdb	4 + 4096	octet string	00001000_{Hex} + 4096 octets
numChildren	1	zero	00_{Hex}

Table 13 — Complex BIR fields and abstract values corresponding to ISO/IEC 19785-1, Figure 2

1.	patronHeaderVersion = 2	*(beginning of the root header BIR)*
2.	cbeffVersion = 4:0	
3.	fieldPresence = sbFormatOwner/Type	
4.	birIntegrity = INTEGRITY	*(integrity is applied to the entire complex BIR via the SB on line 90)*
5.	sbFormatOwner = *a security vendor*	
6.	sbFormatType = *that vendor's security block format*	*(see the final SB on line 90)*
7.	numChildren = 2	
8.	childBirPatronFormatOwner = SC 37	
9.	childBirPatronFormatType = 6 (*this format*)	
10.	►	*(denotes the beginning of the next BIR)*
11.	patronHeaderVersion = 2	
12.	cbeffVersion = 4:0	
13.	fieldPresence = bdbBiometricType	
14.	birIntegrity = NO INTEGRITY	
15.	bdbBiometricType = FINGER	*(the next 3 BIRs inherit this value)*
16.	numChildren = 3	
17.	childBirPatronFormatOwner = SC 37	
18.	childBirPatronFormatType = 6 (this format)	
19.	►	
20.	patronHeaderVersion = 2	
21.	cbeffVersion = 4:0	
22.	fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; bdb	

Table 13 *(continued)*

bdbFormatOwner = SC 37
bdbFormatType = *a standardized BDB format*
bdbEncryption = NO ENCRYPTION
birIntegrity = NO INTEGRITY
bdbBiometricSubtype = LEFT INDEX FINGER
bdb
numChildren=0
►
patronHeaderVersion = 2
cbeffVersion = 4:0
fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; bdb
bdbFormatOwner = *vendor ABC*
bdbFormatType = *non standard BDB format A*
bdbEncryption = NO ENCRYPTION
birIntegrity = NO INTEGRITY
bdbBiometricSubtype = LEFT MIDDLE FINGER
bdb
numChildren=0
►
patronHeaderVersion = 2
cbeffVersion = 4:0
fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; bdb
bdbFormatOwner = *vendor XYZ*
bdbFormatType = *non standard BDB format B*
bdbEncryption = NO ENCRYPTION
birIntegrity = NO INTEGRITY
bdbBiometricSubtype = LEFT RING FINGER
bdb
numChildren=0
►
patronHeaderVersion = 2
cbeffVersion = 4:0
fieldPresence = bdbBiometricType
birIntegrity = NO INTEGRITY
bdbBiometricType = IRIS *(the next 2 BIRs inherit this type)*
numChildren = 2
childBirPatronFormatOwner = SC 37
childBirPatronFormatType = 6 (*this format*)
►
patronHeaderVersion = 2
cbeffVersion = 4:0
fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype;
sbFormatOwner/Type; bdb; sb

Table 13 *(continued)*

65.	bdbFormatOwner = SC 37	
66.	bdbFormatType = *an iris format*	
67.	bdbEncryption = ENCRYPTION	
68.	birIntegrity = NO INTEGRITY	
69.	bdbBiometricSubtype = LEFT	
70.	sbFormatOwner = *an encryption vendor*	
71.	sbFormatType = *a security block format*	*(see SB on line 74)*
72.	bdb	
73.	numChildren=0	
74.	sb	*(see SB format identifier on lines 70-71)*
75.	►	
76.	patronHeaderVersion = 2	
77.	cbeffVersion = 4:0	
78.	fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; sbFormatOwner/Type; bdb; sb	
79.	bdbFormatOwner = *vendor PQR*	
80.	bdbFormatType = *vendor's format C*	
81.	bdbEncryption = ENCRYPTION	
82.	birIntegrity = NO INTEGRITY	
83.	bdbBiometricSubtype = RIGHT	
84.	sbFormatOwner = *an encryption vendor*	
85.	sbFormatType = *a security block format*	*(see SB on line 88)*
86.	bdb	
87.	numChildren=0	
88.	sb	*(see SB format identifier on lines 84-85)*
89.	►	
90.	sb	*(see SB format identifier in root header on line 6)*

Table 14 — BIR wrapped in an enveloping BIR

Field name	Length	Abstract value	Encoding
patronHeaderVersion	1	2	02_{Hex}
cbeffVersion	1	Major 4, Minor 0	40_{Hex}
fieldPresence	3	all optional fields absent in the enveloping BIR	000000_{Hex}
birIntegrity	1	NO INTEGRITY	00_{Hex}
numChildren	1	one child *(the enveloped BIR)*	01_{Hex}
childBirPatronFormatOwner)	2	patron format owner of the enveloped BIR	*Variable*
childBirPatronFormatType	2	patron format type of the enveloped BIR	*Variable*
childBir (length of the child BIR)	4	length of the enveloped BIR	*Variable*
childBir (octets of the child BIR	*variable*	octets of the enveloped BIR	*Variable*

Table 14 shows how the Complex patron format specified in this clause can be used as a simple envelope around a BIR of an arbitrary patron format in order to provide identification of its format and specify its length. When using the Complex patron format in this way, the portion of the enveloping BIR preceding

the enveloped BIR can be thought of as a fixed-length prefix to the enveloped BIR. Since all the optional fields of the enveloping BIR are absent, the length of the prefix is only 15 octets, given by:

a) 7 octets with the fixed values 01200000000001_{Hex}; plus

b) 4 octets containing the patron format owner and type of the enveloped BIR; plus

c) 4 octets containing the length of the enveloped BIR.

12.12 ASN.1 definition (provided for illustrative purposes only)

The following ASN.1 specification provides an abstract description of the patron format and can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en. This ASN.1 specification may be useful to some readers of this document, but it is not intended to provide an alternative specification of the encodings of this patron format.

```
CBEFF-COMPLEX-PATRON-FORMAT
{ iso(1) standard(0) cbeff(19785) modules(0) complex-BIR(6)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

IMPORTS BiometricType, BiometricSubtype, Product
FROM CBEFF-DATA-ELEMENTS

BIR ::= SEQUENCE {
    patronHeaderVersion INTEGER(0..255),
    cbeffVersion INTEGER(0..255),

    fieldPresence SEQUENCE {
    bdbFormat BOOLEAN,
    bdbEncryption BOOLEAN,
    bdbBiometricType BOOLEAN,
    bdbBiometricSubtype BOOLEAN,
    bdbChallengeResponse BOOLEAN,
    bdbCreationDate BOOLEAN,
    bdbIndex BOOLEAN,
    bdbProcessedLevel BOOLEAN,
    bdbProduct BOOLEAN,
    bdbPurpose BOOLEAN,
    bdbQuality BOOLEAN,
    bdbValidityPeriod BOOLEAN,
    birCreationDate BOOLEAN,
    birCreator BOOLEAN,
    birIndex BOOLEAN,
    birValidityPeriod BOOLEAN,
    sbFormat BOOLEAN,
    bdb BOOLEAN,
    children BOOLEAN,
    sb BOOLEAN
    },

    bdbFormat SEQUENCE {
        bdbFormatOwner INTEGER(0..65535),
        bdbFormatType INTEGER(0..65535)
    } OPTIONAL,

    bdbEncryption INTEGER(0..255) OPTIONAL,
    birIntegrity INTEGER(0..255),
    bdbBiometricType BiometricType DEFAULT noValueAvailable,
    bdbBiometricSubtype BiometricSubtype DEFAULT noValueAvailable,
    bdbChallengeResponse OCTET STRING (SIZE(0..65535)) OPTIONAL,
    bdbCreationDate OCTET STRING (SIZE(8..15)) OPTIONAL,
    bdbIndex OCTET STRING (SIZE(0..65535)) OPTIONAL,
    bdbProcessedLevel INTEGER(0..255) OPTIONAL,

    bdbProduct Product OPTIONAL,
    bdbPurpose INTEGER(0..255) OPTIONAL,
```

```
        bdbQuality INTEGER(0..255) OPTIONAL,
        bdbValidityPeriod OCTET STRING (SIZE(15..31)) OPTIONAL,
        birCreationDate OCTET STRING (SIZE(8..15)) OPTIONAL,
        birCreator OCTET STRING (SIZE(0..65535)) OPTIONAL,
        birIndex OCTET STRING (SIZE(0..65535)) OPTIONAL,
        birPayload OCTET STRING (SIZE(0..65535)) OPTIONAL,
        birValidityPeriod OCTET STRING (SIZE(15..31)) OPTIONAL,

        sbFormat SEQUENCE {
           sbFormatOwner INTEGER(0..65535),
           sbFormatType INTEGER(0..65535)
        } OPTIONAL,

        bdb OCTET STRING (SIZE(0..4294967295)) OPTIONAL,

        children SEQUENCE (SIZE(0..255)) OF
            child SEQUENCE {
                childBirPatronFormat SEQUENCE {
                    childBirPatronFormatOwner INTEGER(0..65535),
                    childBirPatronFormatType INTEGER(0..65535)
                },
                childBir OCTET STRING (SIZE(0..4294967295))
            }
        ,
        sb OCTET STRING (SIZE(0..4294967295)) OPTIONAL
    }
    END
```

13 Patron format specification: XML-full patron format (with additional data elements)

13.1 Patron

ISO/IEC JTC 1/SC 37.

13.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

13.3 Patron format name

ISO/IEC JTC 1/SC 37 XML-full patron format.

13.4 Patron format type

11 (000BHex). This has been registered in accordance with ISO/IEC 19785-2.

13.5 ASN.1 OID for this patron format

{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) xml-full(11)}

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.11</OBJECT_IDENTIFIER>
```

NOTE ISO/IEC 19785-3:2015 included patron format type 11 (xml-full) from ISO/IEC 19785-3:2007/Amd 1:2010 Clause 15, but incorrectly listed the patron format type as 7. Patron format type 7 was already assigned to the patron format (xml) defined in ISO/IEC 19785-3:2007 Clause 13. Therefore, the patron format type has been restored to 11 (xml-full) in this document.

13.6 Domain of use

This clause specifies a patron format based on XML that may be of general utility to applications that need to carry one or more BIRs in a simple or complex BIR structure and benefit from the use of XML over a binary format.

This patron format is kept for legacy reasons. For any new implementation that would like to use XML coding, Clauses 16 and/or 17 shall be used.

13.7 Version identifier

This patron format specification has a version identifier of (major 3, minor 0).

13.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0) onwards.

13.9 General

13.9.1 Patron Format Structure Type: Complex CBEFF BIR without self-identification

13.9.2 This patron format is based on W3C XML Schema, Parts 1[13] and 2[14]. It supports all the mandatory and optional data elements specified in ISO/IEC 19785-1:2007. It can support either a simple BIR or a complex BIR structure where each intermediate node or leaf of the structure is itself a BIR (called a "child BIR").

13.9.3 Most fields in this patron format are optional. Some mandatory and optional fields are represented by XML elements, others are represented by attributes of XML elements. The presence of an optional field in a BIR is signaled by simply including the corresponding element or attribute and its absence is signaled by simply omitting the corresponding element or attribute.

13.9.4 Special encodings are specified for integers (see 13.26), octet strings (see 13.27), and date and time-of-the-day abstract values (see 13.28).

13.9.5 An instance of a BIR or child BIR contains either a BDB or one or more BIR children, but never contains both.

13.9.6 An extension mechanism is specified, which enables the inclusion of application-specific data (not standardized) within a BIR or child BIR (see 13.11.1.6).

13.10 Specification

13.10.1 In the rest of this clause, the terms "element" and "attribute" are used with the meaning of "XML element" and "XML attribute", respectively.

13.10.2 The namespace with the name `"http://standards.iso.org/iso-iec/19785/-3/ed-3/en"` is called the patron format namespace of this patron format.

13.10.3 All elements defined in this patron format have the patron format namespace name. All attribute names are unqualified.

13.10.4 An instance of a BIR shall be represented as a `<BIR>` element (see 13.11).

13.10.5 The `<BIR>` element may be the root of an XML document, but this is not required.

13.10.6 The portion of the XML document consisting of the **<BIR>** element and its whole content shall be valid according to the XML schema provided in 13.30, the ASN.1 module given in 13.31, and the ASN.1 XML Encoding Rules (XER) defined in ISO/IEC 8825-4.

NOTE 1 Validity according to that XML schema does not imply that the **<BIR>** element satisfies all the requirements in the normative text of this specification, as there are some requirements that cannot be (or are not) formally expressed in the XML schema.

When the **<BIR>** element is the root of an XML document, the UTF-8 character encoding is recommended for the XML document, because it will usually produce a smaller encoding.

13.10.7 The abstract value NO VALUE AVAILABLE, for any CBEFF data element that supports this abstract value, shall be encoded as the omission of the corresponding element or attribute both in the **<BIR>** element and in all of its ancestor **<BIR>** elements.

NOTE The inheritance mechanism specified in 13.14.2.1, 13.15.2.1 and 13.24.2.1 causes a data element of a BIR to inherit an abstract value (different from NO VALUE AVAILABLE) from its closest ancestor **<BIR>** element that contains that element or attribute when the **<BIR>** element in question does not contain it. If any **<BIR>** element in a hierarchy of **<BIR>** elements specifies an abstract value for a given data element, that abstract value can be overridden by a different abstract value in any of its descendant **<BIR>** elements, but the overriding abstract value can never be NO VALUE AVAILABLE.

13.11 Element <BIR>

13.11.1 Syntax

13.11.1.1 This element shall have no attributes, and shall have a content consisting of the following (in order):

a) an optional **<Version>** element (see 13.12);

b) an optional **<CBEFFVersion>** element (see 13.13);

c) zero or more application-specific elements;

d) a mandatory **<BIRInfo>** element (see 13.14);

e) an optional **<BDBInfo>** element (see 13.15);

f) an optional **<SBInfo>** element (see 13.24);

g) zero or more **<BIR>** elements (see 13.11);

h) either an optional **<BDB>** element that shall contain a valid representation of an octet string (see 13.27), or an optional **<bdbX>** element that shall contain a valid XML string;

i) an optional **<SB>** element — the content of this element shall be a valid representation of an octet string (see 13.27)

13.11.1.2 The **<BDB>** or **<bdbX>** element shall not be present if one or more child **<BIR>** elements are present, and shall be present if no child **<BIR>** elements are present.

13.11.1.3 The **<SB>** element shall be absent unless its presence is required by 13.14.2.2 or permitted by 13.15.2.3.

13.11.1.4 If the **<BDB>** or **<bdbX>** element is present, then the **<BDBInfo>** element shall also be present.

13.11.1.5 If the **<SB>** element is present, then the **<SBInfo>** element shall also be present.

13.11.1.6 The number of application-specific elements and their name, namespace name, attributes, and content are not defined in this patron format specification. However, the namespace name of those elements shall be different from the patron format namespace name (see 13.10.2).

13.11.2 Semantics

13.11.2.1 This element is either a complex or a simple BIR, depending on which child elements are present. If a child **<BDB>** or **<bdbX>** element is present, this element is a simple BIR. If one or more child **<BIR>** elements are present, this element is a complex BIR.

13.11.2.2 The elements **<Version>**, **<CBEFFVersion>**, **<BIRInfo>**, **<BDBInfo>**, and **<SBInfo>** and their content form the standard biometric header of the BIR.

13.11.2.3 The **<Version>** element (if present) carries the major and minor version number of this patron format.

13.11.2.4 The **<CBEFFVersion>** element (if present) carries the major and minor version number of the CBEFF standard.

13.11.2.5 Each **<BIR>** element is a whole BIR (of the same patron format) that is a child BIR of the BIR.

13.11.2.6 The **<BDB>** or **<bdbX>** element (if present) carries the biometric data block (BDB) of the BIR.

NOTE A **<BDB>** or **<bdbX>** element and a **<BIR>** element cannot coexist as children of the same **<BIR>** element (see 13.11.1.2).

13.11.2.7 The **<SB>** element (if present) carries the security block (SB) of the BIR.

NOTE A **<SB>** element can coexist with either a **<BIR>** element or a **<BDB>** or **<bdbX>** element that is a child of the same **<BIR>** element.

13.11.2.8 The **<BIRInfo>** element carries information about both the BIR and (possibly) about its descendant BIRs (if the **<BIR>** element has one or more child **<BIR>** elements), as specified in 13.14.2.1.

13.11.2.9 The **<BDBInfo>** element (if present) carries information about either the BDB of the BIR (if the **<BIR>** element has a child **<BDB>** or **<bdbX>** element) or about the BDBs of the descendant BIRs that have a child **<BDB>** or **<bdbX>** element (if the **<BIR>** element has one or more child **<BIR>** elements), as specified in 13.15.2.1.

13.11.2.10 The **<SBInfo>** element (if present) carries information about either the SB of the BIR (if the **<BIR>** element has a child **<SB>** element) or about the SBs of the descendant BIRs that have a child **<SB>** element (if the **<BIR>** element has one or more child **<BIR>** elements but no child **<SB>** element), as specified in 13.24.2.1.

13.12 Element <Version>

13.12.1 Syntax

This element shall have contents consisting of the following (in order):

a) a required **<Major>** element — the value of this element shall be a valid representation of an integer in the range 0 to 15 (see 13.26);

b) a required **<Minor>** element — the value of this element shall be a valid representation of an integer in the range 0 to 15 (see 13.26).

13.12.2 Semantics

13.12.2.1 This element represents the data element CBEFF_patron_header_version, and carries the (major and minor) version number of the patron format. The number assigned to this version of the patron format is major 3, minor 0.

13.12.2.2 The **<Major>** element represents the major version number (3 in this version).

13.12.2.3 The **<Minor>** element represents the minor version number (0 in this version).

13.12.2.4 If this element is not present, the values **Major="3" Minor="0"** are implied.

13.12.2.5 A child **<BIR>** element shall have the same (major and minor) version number as its parent **<BIR>** element.

NOTE This implies that the **<Version>** element, if present in a child **<BIR>** element, has to carry the same values as the **<Version>** element in the parent **<BIR>** element. This is equivalent to omitting the **<Version>** element. Therefore, this element is normally omitted in child **<BIR>** elements.

13.13 Element <CBEFFVersion>

13.13.1 Syntax

This element shall have content consisting of the following (in order):

a) a required **<Major>** element — the value of this element shall be a valid representation of an integer in the range 0 to15 (see 13.26);

b) a required **<Minor>** element — the value of this element shall be a valid representation of an integer in the range 0 to 15 (see 13.26).

13.13.2 Semantics

13.13.2.1 This element represents the data element CBEFF_version, and carries the version number of the CBEFF standard supported by this patron format. The number assigned to the version of CBEFF supported by this patron format is Major=4, Minor=0.

13.13.2.2 The **<Major>** element represents the major version number (4 in this version).

13.13.2.3 The **<Minor>** element represents the minor version number (0 in this version).

13.13.2.4 If this element is not present, the values **Major="4" Minor="0"** are implied.

13.13.2.5 A child **<BIR>** element shall have the same CBEFF version number (major and minor) as its parent **<BIR>** element.

NOTE Thus, the **<CBEFFVersion>** element is normally omitted from all child **<BIR>** elements, as it would be redundant.

13.14 Element <BIRInfo>

13.14.1 Syntax

13.14.1.1 This element shall have a content consisting of the following (in order):

a) an optional **<Creator>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) an optional **<Index>** element — the content of this element shall be a valid representation of a universally unique identifier (see 13.29), and shall not inherit its value from any other level BIR;

c) an optional **<Payload>** element — the content of this element shall be a valid representation of an octet string, and shall not inherit its value from any other level BIR.

d) a required **<Integrity>** element — the value of this element shall be one of the character strings in the third cell of the corresponding row of Table 15;

e) an optional **<CreationDate>** element — the value of this element shall be a valid representation of a date and time of the day (see 13.28);

f) an optional **<NotValidBefore>** element — the value of this element shall be a valid representation of a date and time of the day (see 13.28);

g) an optional **<NotValidAfter>** element — the value of this element shall be a valid representation of a date and time of the day (see 13.28).

13.14.2 Semantics

13.14.2.1 The **<BIRInfo>** element carries information about the BIR. In addition, if the BIR has one or more child BIRs (the **<BIR>** element has one or more child **<BIR>** elements), the information carried by the attributes and child elements of the **<BIRInfo>** element is inherited by those child BIRs except where overridden by a corresponding attribute or child element of the **<BIRInfo>** element of a child BIR. The information inherited by a BIR applies to that BIR, and (if the BIR has itself child BIRs) is further inherited by its child BIRs in the same way (and so on recursively).

NOTE Since the **Integrity** element is required and the **<BIRInfo>** element is mandatory in all **<BIR>** elements, inheritance of the **Integrity** element can never occur.

13.14.2.2 The **Integrity** element indicates whether integrity information about this BIR is provided within the security block (SB) of the BIR (the child **<SB>** element of the parent **<BIR>** element of this **<BIRInfo>** element).

NOTE This information can consist of a digital signature or MAC, a reference to a key or certificate, an encrypted key (with or without a reference to the key used to encrypt that key), or other parameters of the digital signing (or MAC) process.

13.14.2.3 If the value of the **<Integrity>** element is **"true"**, then the parent **<BIR>** element of this **<BIRInfo>** element shall have a child **<SB>** element.

13.14.2.4 Table 15 specifies the correspondence between the child elements of this element and CBEFF data elements, and specifies the supported abstract values and their encodings (see also 13.10.7).

NOTE This element contains all CBEFF data elements whose names begin with "CBEFF_BIR_".

Table 15 — BIR information

CBEFF data element name	XML element	Supported abstract values and encodings
CBEFF_BIR_creator	<Creator>	All ISO/IEC 10646 character strings are supported. The character string shall be encoded as the string itself.
CBEFF_BIR_index	<Index>	All well-formed UUIDs are supported. The UUIDs shall be encoded as specified in 13.29. Shall not inherit its value from any other BIR level.
CBEFF_BIR_payload	<Payload>	All octet strings are supported. The octet strings shall be encoded as specified in 13.27. Shall not inherit its value from any other BIR level.

CBEFF_BIR_integrity_options	<Integrity>	The following abstract values are supported.
		The abstract values shall be encoded as shown below.
		NO INTEGRITY:
		`"false"`
		INTEGRITY:
		`"true"`

CBEFF_BIR_creation_date	<CreationDate>	All date and time-of-the-day abstract values permitted by CBEFF are supported. The abstract values shall be encoded as specified in 13.28.
CBEFF_BIR_validity_period (lower end)	<NotValidBefore>	All date and time-of-the-day abstract values permitted by CBEFF are supported. The abstract values shall be encoded as specified in 13.28.
CBEFF_BIR_validity_period (upper end)	<NotValidAfter>	All date and time-of-the-day abstract values permitted by CBEFF are supported. The abstract values shall be encoded as specified in 13.28.

13.15 Element <BDBInfo>

13.15.1 Syntax

13.15.1.1 This element shall have a content consisting of the following (in order):

a) an optional **<ChallengeResponse>** element — the content of this element shall be a valid representation of an octet string (see 13.27);

b) an optional **<Index>** element — the content of this element shall be a valid representation of a universally unique identifier (see 13.29).

c) an optional **<Format>** element (see 13.16);

d) an optional **<Encryption>** element — the value of this element shall be one of the character strings in the third cell of the corresponding row of Table 16;

e) an optional **<CreationDate>** element — the value of this element shall be a valid representation of a date and time of the day (see 13.28);

f) an optional **<NotValidBefore>** element — the value of this element shall be a valid representation of a date and time of the day (see 13.28);

g) an optional **<NotValidAfter>** element — the value of this element shall be a valid representation of a date and time of the day (see 13.28);

h) an optional **<Type>** element — the value of this element shall be one of the character strings in the third cell of the corresponding row of Table 16;

i) an optional **<Subtype>** element — the value of this element shall be one of the character strings in the third cell of the corresponding row of Table 16;

j) an optional **<Level>** element — the value of this element shall be one of the character strings in the third cell of the corresponding row of Table 16;

k) an optional **<Product>** element (see 13.17);

l) an optional **<CaptureDevice>** element (see 13.18);

m) an optional **<FeatureExtractionAlgorithm>** element (see 13.19);

n) an optional **<ComparisonAlgorithm>** element (see 13.20);

o) an optional **<CompressionAlgorithm>** element (see 13.21);

p) an optional **<Purpose>** element — the value of this element shall be one of the character strings in the third cell of the corresponding row of Table 16;

q) an optional **<Quality>** element (see 13.22).

13.15.1.3 If the parent **<BIR>** element has a child **<BDB>** element, then the **<Encryption>** element shall be present in this **<BDBInfo>** element unless it is present in the child **<BDBInfo>** element of an ancestor **<BIR>** element (see also 13.11.1.4).

13.15.1.4 If the parent **<BIR>** element has a child **<BDB>** element, then the **<FormatOwner>** element shall be present in this **<BDBInfo>** element unless it is present in the child **<BDBInfo>** element of an ancestor **<BIR>** element (see also 13.11.1.4).

13.15.1.5 If the parent **<BIR>** element has a child **<BDB>** element, then the **<FormatType>** element shall be present in this **<BDBInfo>** element unless it is present in the child **<BDBInfo>** element of an ancestor **<BIR>** element (see also 13.11.1.4).

The ancestor **<BIR>** elements mentioned in the last three subclauses above need not be the same.

13.15.2 Semantics

13.15.2.1 If the BIR has a BDB (the **<BIR>** element has a child **<BDB>** element), then the **<BDBInfo>** element carries information about that BDB. Otherwise, the information carried by the attributes and child elements of the **<BDBInfo>** element is inherited by all the BIRs that are children of the BIR except where overridden by a corresponding attribute or child element of the **<BDBInfo>** element of a child BIR. The information inherited by a BIR with a BDB applies to that BDB, whereas the information inherited by a BIR that has itself child BIRs is further inherited by all the BIRs that are children of the BIR in the same way (and so on recursively).

13.15.2.2 If the BIR has a BDB and encryption is applied to that BDB (either by including the **encryption** attribute with the value **"true"** in the **<BDBInfo>** element or by having the BIR inherit that attribute value from its parent BIR), then the BDB in the **<BDB>** element shall be encrypted.

13.15.2.3 If the BDB of a BIR is encrypted, information about the encryption process may be provided within the security block (SB) of that BIR (the child **<SB>** element of the parent **<BIR>** element of this **<BIRInfo>** element).

NOTE This information can consist of a reference to an encryption key, an encrypted key (with or without a reference to the key used to encrypt that key), or other parameters of the encryption process.

13.15.2.4 Table 16 specifies the correspondence between the child elements of this element and CBEFF data elements, and specifies the supported abstract values and their encodings (see also 13.10.7).

NOTE This element contains all CBEFF data elements whose name begins with "CBEFF_BDB_".

Table 16 — BDB information

CBEFF data element name	XML element	Supported abstract values and encodings
CBEFF_BDB_format	<Format>	See 13.16

CBEFF_BDB_encryption_options	<Encryption>	The following abstract values are supported.
		The abstract values shall be encoded as shown below.
		NO ENCRYPTION:
		`"false"`
		ENCRYPTION:
		`"true"`

CBEFF_BDB_creation_date	<CreationDate>	All date and time-of-the-day abstract values permitted by CBEFF are supported. The abstract values shall be encoded as specified in 13.28.
CBEFF_BDB_validity_period (lower end)	<NotValidBefore>	All date and time-of-the-day abstract values permitted by CBEFF are supported. The abstract values shall be encoded as specified in 13.28.
CBEFF_BDB_index	<Index>	All well-formed UUIDs are supported. The UUIDs shall be encoded as specified in 13.29. Shall appear only in BIRs that have a BDB.
CBEFF_BDB_challenge_response	<ChallengeResponse>	All octet strings are supported. The octet strings shall be encoded as specified in 13.27. Shall appear only in BIRs that have a BDB.
CBEFF_BDB_validity_period (upper end)	<NotValidAfter>	All date and time-of-the-day abstract values permitted by CBEFF are supported. The abstract values shall be encoded as specified in 13.28.

Table 16 *(continued)*

CBEFF data element name	XML element	Supported abstract values and encodings
CBEFF_BDB_biometric_type	<Type>	The following abstract values and all their unordered combinations are supported. A single abstract value shall be encoded as the corresponding string shown below. A combination of two or more abstract values shall be encoded as a list of elements (see XML Schema). SCENT: **"Scent"** DNA: **"DNA"** EAR: **"Ear"** FACE: **"Face"** FINGER: **"Finger"** FOOT: **"Foot"** VEIN: **"Vein"** HAND GEOMETRY: **"HandGeometry"** IRIS: **"Iris"** RETINA: **"Retina"** VOICE: **"Voice"** GAIT: **"Gait"** KEYSTROKE: **"Keystroke"** LIP MOVEMENT: **"LipMovement"** SIGNATURE OR SIGN: **"SignatureSign"** BODY PHOTOGRAPHY: **"BodyPhotography"** FRICTION RIDGE: **"FrictionRidge"** THERMAL: **"Thermal"** PRESENTATION ATTACK DATA **"PresentationAttack"**

Table 16 *(continued)*

CBEFF data element name	XML element	Supported abstract values and encodings
CBEFF_BDB_biometric_subtype	<Subtype>	The following abstract values are supported. The abstract values shall be encoded as shown below. A combination of two or more abstract values shall be encoded as a list of elements (see XML Schema). LEFT: **"Left"** RIGHT: **"Right"** THUMB: **"Thumb"** LEFT THUMB: **"LeftThumb"** RIGHT THUMB: **"RightThumb"** INDEX FINGER: **"IndexFinger"** LEFT INDEX FINGER: **"LeftIndexFinger"** RIGHT INDEX FINGER: **"RightIndexFinger"** MIDDLE FINGER: **"MiddleFinger"** LEFT MIDDLE FINGER: **"LeftMiddleFinger"** RightMIDDLE FINGER: **"RightMiddleFinger"** RING FINGER: **"RingFinger"** LEFT RING FINGER: **"LeftRingFinger"** RIGHT RING FINGER: **"RightRingFinger"** LITTLE FINGER: **"LittleFinger"** LEFT LITTLE FINGER: **"LeftLittleFinger"** RIGHT LITTLE FINGER: **"RightLittleFinger"** PALM: **"Palm"** LEFT PALM: **"LeftPalm"** RIGHT PALM: **"RightPalm"** BACK OF HAND: **"BackOfHand"** LEFT BACK OF HAND: **"LeftBackOfHand"** RIGHT BACK OF HAND: **"RightBackOfHand"** WRIST: **"Wrist"** LEFT WRIST: **"LeftWrist"** RIGHT WRIST: **"RightWrist"**

Table 16 *(continued)*

CBEFF data element name	XML element	Supported abstract values and encodings
CBEFF_BDB_processed_level	<Level>	The following abstract values are supported. The abstract values shall be encoded as shown below. RAW: **"Raw"** INTERMEDIATE: **"Intermediate"** PROCESSED: **"Processed"**
CBEFF_BDB_product	<Product>	See 13.17
CBEFF_BDB_capture_device	<CaptureDevice>	See 13.18
CBEFF_BDB_feature_extraction_algorithm	<FeatureExtractionAlgorithm>	See 13.19
CBEFF_BDB_comparison_algorithm	<ComparisonAlgorithm>	See 13.20
CBEFF_BDB_compression_algorithm	<CompressionAlgorithm>	See 13.21
CBEFF_BDB_purpose	<Purpose>	The following abstract values are supported. The abstract values shall be encoded as shown below. VERIFY: **"Verify"** IDENTIFY: **"Identify"** ENROL: **"Enrol"** ENROL FOR VERIFICATION ONLY: **"EnrolVerify"** ENROL FOR IDENTIFICATION ONLY: **"EnrolIdentify"** AUDIT: **"Audit"**
CBEFF_BDB_quality	<Quality>	See 13.22

13.16 Element <Format> of BDBInfoType

13.16.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required **<Organization>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required **<Type>** element — the content of this element shall be a string of ISO/IEC 10646 characters.

13.16.2 Semantics

13.16.2.1 This element represents the CBEFF_BDB_format_owner and CBEFF_BDB_format_type data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.16.2.2 The **<Organization>** element represents the organization/owner/vendor code as specified in the registry.

13.16.2.3 The **<Type>** element represents the type code as specified in the registry.

13.17 Element <Product>

13.17.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required **<Organization>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required **<Type>** element — the content of this element shall be a string of ISO/IEC 10646 characters.

13.17.2 Semantics

13.17.2.1 This element represents the CBEFF_BDB_product_owner and CBEFF_BDB_product_type data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.17.2.2 The **<Organization>** element represents the organization/owner/vendor code as specified in the registry.

13.17.2.3 The **<Type>** element represents the type code as specified in the registry.

13.18 Element <CaptureDevice>

13.18.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required **<Organization>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required **<Type>** element — the content of this element shall be a string of ISO/IEC 10646 characters.

13.18.2 Semantics

13.18.2.1 This element represents the CBEFF_BDB_capture_device_owner and CBEFF_BDB_capture_device_type data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.18.2.2 The **<Organization>** element represents the organization/owner/vendor code as specified in the registry.

13.18.2.3 The **<Type>** element represents the type code as specified in the registry.

13.19 Element <FeatureExtractionAlgorithm>

13.19.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required **<Organization>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required `<Type>` element — the content of this element shall be a string of ISO/IEC 10646 characters;

13.19.2 Semantics

13.19.2.1 This element represents the CBEFF_BDB_feature_extraction_algorithm_owner and CBEFF_BDB_feature_extraction_algorithm_type data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.19.2.2 The `<Organization>` element represents the organization/owner/vendor code as specified in the registry.

13.19.2.3 The `<Type>` element represents the type code as specified in the registry.

13.20 Element <ComparisonAlgorithm>

13.20.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required `<Organization>` element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required `<Type>` element — the content of this element shall be a string of ISO/IEC 10646 characters;

13.20.2 Semantics

13.20.2.1 This element represents the CBEFF_BDB_comparison_algorithm_owner and CBEFF_BDB_comparison_algorithm_type data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.20.2.2 The `<Organization>` element represents the organization/owner/vendor code as specified in the registry.

13.20.2.3 The `<Type>` element represents the type code as specified in the registry.

13.21 Element <CompressionAlgorithm>

13.21.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required `<Organization>` element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required `<Type>` element — the content of this element shall be a string of ISO/IEC 10646 characters;

13.21.2 Semantics

13.21.2.1 This element represents the CBEFF_BDB_compression_algorithm_owner and CBEFF_BDB_compression_algorithm_type data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.21.2.2 The **<Organization>** element represents the organization/owner/vendor code as specified in the registry.

13.21.2.3 The **<Type>** element represents the type code as specified in the registry.

13.22 Element <Quality>

13.22.1 Syntax

Absence of this element indicates that quality is not reported by the appropriate quality algorithm provider. If present, this element shall have the contents consisting of the following (in order):

a) a required **<Algorithm>** element (see 13.23).

b) either a **<Score>** element shall be a valid representation of an integer in the range 0 to 100 (see 13.26), or a **<QualityCalculationFailed>** element if there was an error during the calculation of a quality score. The **<QualityCalculationFailed>** element must either be empty or shall be a string of ISO/IEC 10646 characters.

13.22.2 Semantics

13.22.2.1 The **<Score>** element, if present, represents the CBEFF_BDB_quality data element and carries an integer score in the range of 0 to 100 specifying the quality score calculated by a biometric system.

13.22.2.2 The **<QualityCalculatedFailed>** element, if present, denotes that the quality calculation was unsuccessful for some reason. This element can either be empty or carry a message of why the calculation failed.

13.23 Element <Algorithm>

13.23.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required **<Organization>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required **<Type>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

13.23.2 Semantics

13.23.2.1 This element represents the CBEFF_BDB_quality_algorithm_owner and CBEFF_BDB_quality_algorithm data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.23.2.2 The **<Organization>** element represents the organization/owner/vendor code as specified in the registry.

13.23.2.3 The **<Type>** element represents the type code as specified in the registry.

13.24 Element <SBInfo>

13.24.1 Syntax

13.24.1.1 The contents of this element shall be limited to an optional **<Format>** element (see 13.25).

13.24.1.2 If the parent **<BIR>** element has a child **<SB>** element, then the **<Format>** element shall be present in this **<SBInfo>** element unless it is present in the child **<SBInfo>** element of an ancestor **<BIR>** element (see also 13.11.1.5).

The ancestor **<BIR>** elements mentioned in the last two subclauses above need not be the same. When the parent **<BIR>** element has a child **<SB>** element and one omits both children of the **<SBInfo>** element, the **<SBInfo>** element will have no attributes and an empty content. Omission of the **<SBInfo>** element is not allowed in this case (see 13.11.1.5).

13.24.2 Semantics

13.24.2.1 If the BIR has an SB (the **<BIR>** element has a child **<SB>** element), then the **<SBInfo>** element carries information about that SB. In addition, if the BIR has one or more child BIRs (the **<BIR>** element has one or more child **<BIR>** elements), the information carried by the child element of the **<SBInfo>** element is inherited by those child BIRs except where overridden by a corresponding child element of the **<SBInfo>** element of a child BIR. The information inherited by a BIR with an SB applies to that SB, and (if the BIR has itself child BIRs) is further inherited by its child BIRs in the same way (and so on recursively).

13.25 Element <Format> of SBInfoType

13.25.1 Syntax

If present, this element shall have the contents consisting of the following (in order):

a) a required **<Organization>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

b) a required **<Type>** element — the content of this element shall be a string of ISO/IEC 10646 characters;

13.25.2 Semantics

13.25.2.1 This element represents the CBEFF_SB_format_owner and CBEFF_SB_format_type data elements, and carries the organization code and type code, respectively, as registered with the Biometric Registry Authority.

13.25.2.2 The **<Organization>** element represents the organization/owner/vendor code as specified in the registry.

13.25.2.3 The **<Type>** element represents the type code as specified in the registry.

13.26 Representation of Integers

13.26.1 A non-negative integer shall be represented as a string of one or more ISO/IEC 10646 characters in the range DIGIT ZERO to DIGIT NINE (`"0"` to `"9"`) in decimal notation.

13.26.2 A negative integer shall be represented as the corresponding positive integer, preceded by a HYPHEN-MINUS character (`"-"`).

13.26.3 Arbitrary whitespace is allowed before and after the encoding but is forbidden inside the encoding.

13.27 Representation of Octet Strings

13.27.1 An octet string shall be represented as a string of the following ISO/IEC 10646 characters:

a) LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER Z;

b) LATIN SMALL LETTER A to LATIN SMALL LETTER Z;

c) DIGIT ZERO to DIGIT NINE;

d) PLUS SIGN;

e) SOLIDUS;

f) EQUALS SIGN.

forming the Base64 encoding of the octet string (see IETF RFC 2045), with all whitespace removed.

13.27.2 Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the encoding.

13.28 Representation of Date and Time of the Day

13.28.1 A date and time of the day shall be represented as a string of ISO/IEC 10646 characters in the following format, which conforms to the ISO 8601 series.

13.28.2 The encoding shall be the concatenation of all the following components (in order):

a) the "year" component, consisting of the year encoded in four digits (`"2000"` to `"2999"`);

b) the hyphen character "-";

c) the "month" component, consisting of the month encoded in two digits (`"01"` to `"12"`);

d) the hyphen character "-";

e) the "day" component, consisting of the day encoded in two digits (`"01"` to `"31"`);

f) the letter `"T"`;

g) the "hour" component, consisting of the hour encoded in two digits (`"00"` to `"23"`);

h) the colon character ":";

i) the "minute" component, consisting of the minute encoded in two digits (`"00"` to `"59"`);

j) the colon character ":";

k) the "second" component, consisting of the second encoded in two digits (`"00"` to `"59"`);

l) the letter `"Z"`.

13.28.3 The "year", "month", "day", "hour", "minute", and "second" components shall be present.

13.28.4 The letter `"T"` shall be present.

13.28.5 The letter `"Z"` shall be present whether or not the "hour" component is present.

NOTE This letter indicates that the date and time of the day are UTC.

13.28.6 Arbitrary whitespace is allowed before and after the encoding, but is forbidden inside the encoding.

13.29 Representation of Universally Unique Identifiers

13.29.1 The following subclauses describe the same representation of a UUID as is specified in ISO/IEC 9834-8:2014, Clause 8. An example of such a representation is: f81d4fae-7dec-11d0-a765-00a0c91e6bf6

13.29.2 A universally unique identifier (UUID) shall be represented as a string of ISO/IEC 10646 characters. Each string shall contain exactly 36 characters from the union of the following sets:

a) DIGIT ZERO to DIGIT NINE (`"0"` to `"9"`), each representing a hexadecimal digit 0 through 9;

b) LATIN CAPITAL LETTER A to LATIN CAPITAL LETTER F (`"A"` to `"F"`), each representing a hexadecimal digit A through F;

c) LATIN SMALL LETTER A to LATIN SMALL LETTER F (`"a"` to `"f"`), each representing a hexadecimal digit A through F; and

d) HYPHEN-MINUS (`"-"`).

13.29.3 Each of the positions 9, 14, 19, and 24 of an encoding shall contain a character from set (d). The other 32 positions shall contain characters from sets (a) through (c).

13.29.4 Arbitrary whitespace is allowed before and after the encoding but is forbidden inside the encoding.

13.30 XML schema of the patron format

The following XML schema can be retrieved from http://standards.iso.org/iso-iec/19785/-3/ed-3/en.

```
<?xml version="1.0" encoding="utf-8"?>

<!--
THE SCHEMA IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SCHEMA OR THE USE OR OTHER
DEALINGS IN THE SCHEMA.
-->

<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://standards.iso.org/iso-iec/19785/-3/ed-3/en"
  targetNamespace="http://standards.iso.org/iso-iec/19785/-3/ed-3/en"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:element name="BIR" type="BIRType"/>

  <xs:complexType name="BIRType">
        <xs:sequence>
              <xs:element name="Version" type="VersionType"
minOccurs="0"/>
              <xs:element name="CBEFFVersion" type="VersionType"
minOccurs="0"/>
```

```
                <xs:any namespace="##other" processContents="skip"
minOccurs="0" maxOccurs="unbounded"/>
                <xs:element name="BIRInfo" type="BIRInfoType"/>
                <xs:element name="BDBInfo" type="BDBInfoType" minOccurs="0"/>
                <xs:element name="SBInfo" type="SBInfoType" minOccurs="0"/>
                 <xs:element name="BIR" type="BIRType" minOccurs="0"
maxOccurs="unbounded"/>
                <xs:element name="BDB" type="xs:base64Binary" minOccurs="0"/>
                <xs:element name="SB" type="xs:base64Binary" minOccurs="0"/>
        </xs:sequence>
  </xs:complexType>

  <xs:complexType name="VersionType">
        <xs:sequence>
                <xs:element name="Major" type="xs:unsignedInt"/>
                <xs:element name="Minor" type="xs:unsignedInt"/>
        </xs:sequence>
  </xs:complexType>

  <xs:complexType name="BIRInfoType">
        <xs:sequence>
                <xs:element name="Creator" type="xs:string" minOccurs="0"/>
                <xs:element name="Index" type="UUIDType" minOccurs="0"/>
                <xs:element name="Payload" type="xs:base64Binary"
minOccurs="0"/>
                <xs:element name="Integrity" type="xs:boolean"/>
                <xs:element name="CreationDate" type="xs:dateTime"
minOccurs="0"/>
                <xs:element name="NotValidBefore" type="xs:dateTime"
minOccurs="0"/>
                <xs:element name="NotValidAfter" type="xs:dateTime"
minOccurs="0"/>
        </xs:sequence>
  </xs:complexType>

  <xs:complexType name="BDBInfoType">
        <xs:sequence>
                <xs:element name="ChallengeResponse" type="xs:base64Binary"
minOccurs="0"/>
                <xs:element name="Index" type="UUIDType" minOccurs="0"/>
                <xs:element name="Format" type="RegistryIDType" minOccurs="0"/>
                <xs:element name="Encryption" type="xs:boolean" minOccurs="0"/>
                <xs:element name="CreationDate" type="xs:dateTime"
minOccurs="0"/>
                <xs:element name="NotValidBefore" type="xs:dateTime"
minOccurs="0"/>
                <xs:element name="NotValidAfter" type="xs:dateTime"
minOccurs="0"/>
                <xs:element name="Type" type="MultipleTypesType"
minOccurs="0"/>
                <xs:element name="Subtype" type="SubtypeType"
minOccurs="0"/>
                <xs:element name="Level" type="ProcessedLevelType"
minOccurs="0"/>
                <xs:element name="Product" type="RegistryIDType"
minOccurs="0"/>
                <xs:element name="CaptureDevice" type="RegistryIDType"
minOccurs="0"/>
                <xs:element name="FeatureExtractionAlgorithm"
type="RegistryIDType" minOccurs="0"/>
                <xs:element name="ComparisonAlgorithm" type="RegistryIDType"
minOccurs="0"/>
                <xs:element name="CompressionAlgorithm" type="RegistryIDType"
minOccurs="0"/>
                <xs:element name="Purpose" type="PurposeType" minOccurs="0"/>
                <xs:element name="Quality" type="QualityType" minOccurs="0"/>
        </xs:sequence>
  </xs:complexType>

  <xs:complexType name="RegistryIDType">
        <xs:sequence>
```

```
                <xs:element name="Organization" type="xs:string"/>
                <xs:element name="Type" type="xs:string"/>
          </xs:sequence>
    </xs:complexType>

    <xs:complexType name="SBInfoType">
          <xs:sequence>
                <xs:element name="Format" type="RegistryIDType" minOccurs="0"/>
          </xs:sequence>
    </xs:complexType>

    <xs:simpleType name="QualityScoreType">
          <xs:restriction base="xs:unsignedInt">
                <xs:maxInclusive value="100"/>
          </xs:restriction>
    </xs:simpleType>

    <xs:complexType name="QualityType">
          <xs:sequence>
             <xs:element name="Algorithm" type="RegistryIDType"/>
             <xs:choice>
                  <xs:element name="Score" type="QualityScoreType"/>
                  <xs:element name="QualityCalculationFailed"
type="xs:string"/>
          </xs:choice>
       </xs:sequence>
</xs:complexType>

<xs:simpleType name="SingleTypeType">
     <xs:restriction base="xs:string">
          <xs:enumeration value="Scent"/>
          <xs:enumeration value="DNA"/>
          <xs:enumeration value="Ear "/>
          <xs:enumeration value="Face"/>
          <xs:enumeration value="Finger"/>
          <xs:enumeration value="Foot"/>
          <xs:enumeration value="HandGeometry"/>
          <xs:enumeration value="Vein"/>
          <xs:enumeration value="Iris"/>
          <xs:enumeration value="Retina"/>
          <xs:enumeration value="Voice"/>
          <xs:enumeration value="Gait"/>
          <xs:enumeration value="Keystroke"/>
          <xs:enumeration value="LipMovement"/>
          <xs:enumeration value="SignatureSign"/>
          <xs:enumeration value="BodyPhotography"/>
          <xs:enumeration value="FrictionRidge"/>
          <xs:enumeration value="Thermal"/>
          <xs:enumeration value="PresentationAttack"/>
     </xs:restriction>
</xs:simpleType>

<xs:simpleType name="MultipleTypesType">
     <xs:list itemType="SingleTypeType"/>
</xs:simpleType>

<xs:simpleType name="SingleAnySubtypeType">
     <xs:restriction base="xs:string">
          <xs:enumeration value="Left"/>
          <xs:enumeration value="Right"/>
          <xs:enumeration value="Thumb"/>
          <xs:enumeration value="LeftThumb"/>
          <xs:enumeration value="RightThumb"/>
          <xs:enumeration value="IndexFinger"/>
          <xs:enumeration value="LeftIndexFinger"/>
          <xs:enumeration value="RightIndexFinger"/>
          <xs:enumeration value="MiddleFinger"/>
          <xs:enumeration value="LeftMiddleFinger"/>
          <xs:enumeration value="RightMiddleFinger"/>
          <xs:enumeration value="RingFinger"/>
          <xs:enumeration value="LeftRingFinger"/>
```

```
                <xs:enumeration value="RightRingFinger"/>
                <xs:enumeration value="LittleFinger"/>
                <xs:enumeration value="LeftLittleFinger"/>
                <xs:enumeration value="RightLittleFinger"/>
                <xs:enumeration value="Palm"/>
                <xs:enumeration value="LeftPalm"/>
                <xs:enumeration value="RightPalm"/>
                <xs:enumeration value="BackOfHand"/>
                <xs:enumeration value="LeftBackOfHand"/>
                <xs:enumeration value="RightBackOfHand"/>
                <xs:enumeration value="Wrist"/>
                <xs:enumeration value="LeftWrist"/>
                <xs:enumeration value="RightWrist"/>
          </xs:restriction>
     </xs:simpleType>

     <xs:simpleType name="SubtypeType">
          <xs:list itemType="SingleAnySubtypeType"/>
     </xs:simpleType>

     <xs:simpleType name="ProcessedLevelType">
          <xs:restriction base="xs:string">
                <xs:enumeration value="Raw"/>
                <xs:enumeration value="Intermediate"/>
                <xs:enumeration value="Processed"/>
          </xs:restriction>
     </xs:simpleType>

     <xs:simpleType name="PurposeType">
          <xs:restriction base="xs:string">
                <xs:enumeration value="Verify"/>
                <xs:enumeration value="Identify"/>
                <xs:enumeration value="Enrol"/>
                <xs:enumeration value="EnrolVerify"/>
                <xs:enumeration value="EnrolIdentify"/>
                <xs:enumeration value="Audit"/>
          </xs:restriction>
     </xs:simpleType>

     <xs:simpleType name="UUIDType">
          <xs:restriction base="xs:string">
                <xs:pattern value="[a-fA-F0-9]{8}\-([a-fA-F0-9]{4}\-){3}
     [a-fA-F0-9]{12}"/>
          </xs:restriction>
     </xs:simpleType>
  </xs:schema>
```

NOTE NO VALUE AVAILABLE is encoded by the absence of optional fields in the XML encoding. There is little value in, for example, having the following string appear in a record: <level> no value available <level>.

13.31 ASN.1 schema of the patron format

The following ASN.1 module can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en.

```
XML-FULL-PATRON-FORMAT
{iso(1) standard(0) cbeff(19785) modules(0) xml-full-patron-format(11)}
DEFINITIONS
XER INSTRUCTIONS
AUTOMATIC TAGS ::=
BEGIN
IMPORTS
    BiometricDataBlock, Version, RegistryID, ProcessedLevel, Purpose,
    SecurityBlock
FROM CBEFF-DATA-ELEMENTS
    SBType
FROM SB-DATA-ELEMENTS;


BIR ::= SEQUENCE {
       version       Version OPTIONAL,
```

```
        cBEFFVersion  Version OPTIONAL,
        elemlist      [UNTAGGED] SEQUENCE OF elem UTF8String (CONSTRAINED BY
                   {/* Shall conform to the "AnyElementFormat" specified
                       in ITU-T Rec. X.693 | ISO/IEC 8825-4, clause 19 */}),
        bIRInfo       BIRInfoType,
        bDBInfo       BDBInfoType OPTIONAL,
        sBInfo        SBInfoType OPTIONAL,
        birlist       [UNTAGGED] SEQUENCE OF bIR BIR,
        bDB           [BASE64] BiometricDataBlock OPTIONAL,
        sB            [BASE64] SBType OPTIONAL
}

BIRInfoType ::= SEQUENCE {
        creator          UTF8String OPTIONAL,
        index            UUID OPTIONAL,
        payload          [BASE64] OCTET STRING OPTIONAL,
        integrity        BOOLEAN,
        creationDate     DATE-TIME OPTIONAL,
        notValidBefore   DATE-TIME OPTIONAL,
        notValidAfter    DATE-TIME OPTIONAL
}

BDBInfoType ::= SEQUENCE {
        challengeResponse [BASE64] OCTET STRING OPTIONAL,
        index UUID OPTIONAL,
        format              RegistryID OPTIONAL,
        encryption          BOOLEAN OPTIONAL,
        creationDate        DATE-TIME OPTIONAL,
        notValidBefore      DATE-TIME OPTIONAL,
        notValidAfter       DATE-TIME OPTIONAL,
        type                [LIST] SEQUENCE OF SingleTypeType OPTIONAL,
        subtype             Subtype OPTIONAL,
        level               ProcessedLevel OPTIONAL,
        product             RegistryID OPTIONAL,
        captureDevice       RegistryID OPTIONAL,
        featureExtractionAlgorithm RegistryID OPTIONAL,
        comparisonAlgorithm  RegistryID OPTIONAL,
        compressionAlgorithm RegistryID OPTIONAL,
        purpose             Purpose OPTIONAL,
        quality             Quality OPTIONAL
}

SBInfoType ::= SEQUENCE {
        format       RegistryID OPTIONAL
}

Quality ::= SEQUENCE {
        algorithm       RegistryID,
        qualityScoreType QualityScoreType
}

QualityScoreType ::= CHOICE {
        score INTEGER (0..100),
        qualityCalculationFailed UTF8String
}

SingleTypeType ::= ENUMERATED {
        scent,
        dNA,
        ear,
        face,
        finger,
        foot,
        handGeometry,
        vein,
        iris,
        retina,
        voice,
        gait,
        keystroke,
        lipMovement,
```

```
        signatureSign,
        bodyPhotography,
        frictionRidge,
        thermal,
    presentationAttack
}

SingleAnySubtypeType ::= ENUMERATED {
              left,
              right,
              thumb,
              leftThumb,
              rightThumb,
              indexFinger,
              leftIndex,
              rightIndex,
              middleFinger,
              leftMiddle,
              rightMiddle,
              ringFinger,
              leftRing,
              rightRing,
              littleFinger,
              leftLittle,
              rightLittle,
              palm,
              leftPalm,
              rightPalm,
              backOfHand,
              leftBackOfHand,
              rightBackOfHand,
              wrist,
              leftWrist,
              rightWrist
}

Subtype ::= [LIST] SEQUENCE OF SingleAnySubtypeType

UUID ::= IA5String (SIZE(36)) (PATTERN "[a-fA-F0-9]#(8)\-([a-fA-F0-9]#(4)\-)#(3)
[a-fA-F0-9]{12}")

ENCODING-CONTROL XER
GLOBAL-DEFAULTS MODIFIED-ENCODINGS
GLOBAL-DEFAULTS CONTROL-NAMESPACE
"http://www.w3.org/2001/XMLSchema-instance" PREFIX "xsi"
NAMESPACE ALL, ALL IN ALL AS "iso-iec.jtc1.sc37.common" PREFIX "ns"
NOT NAMESPACE ALL IN BIRInfoType, ALL IN BDBInfoType, ALL IN SBInfoType
NAMESPACE index, payload IN BIRInfoType, index, challengeResponse IN BDBInfoType
AS "iso-iec.jtc1.sc37.common" PREFIX "ns"
TEXT BDBInfoType.type.*:ALL, SingleAnySubtypeType:ALL,
BDBInfoType.level:ALL, BDBInfoType.purpose:ALL
ANY-ELEMENT BIR.elemlist.elem EXCEPT ABSENT
"iso-iec.jtc1.sc37.common"
TEXT SingleAnySubtypeType:ALL AS CAPITALIZED
TEXT Processed:ALL AS CAPITALIZED
TEXT PURPOSE:ALL AS CAPITALIZED
NAME AS CAPITALIZED
NAME BIR AS UPPERCASED
END
```

Imported elements shall be defined prior to compiling the ASN.1 specification. For illustrative purposes the following example is provided. Implementers shall refer to the relevant standards where the elements are defined.

```
/*
-- These definitions shall be provided by the SB format provider.
-- In case of being primitive data, these data elements shall be of type
-- OCTET STRING.
-- In case of constructed data, this/these data elements' complete
-- definition/s should be provided with the data element type names given here.
--
-- THE DATA DEFINED BELOW IS AN EXAMPLE FOR ILLUSTRATIVE PURPOSES ONLY
--
SB-DATA-ELEMENTS
{iso(1) standard(0) cbeff(19785) part(3)} -- or whatever OID is registered
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN
   SBType ::= OCTET STRING (SIZE(0..4294967295)) -- if primitive (comment line
                                                 -- if not used)
/*
   SBType ::= SET                  -- if constructed, to be defined by the
                                   -- SB data format provider (comment block
                                   -- if not used)
      {
      tagAllocationAuthority  [APPLICATION 24]      CHOICE
         {
         oid                                        OBJECT IDENTIFIER,
         country               [APPLICATION 1]      OCTET STRING,
         issuerID              [APPLICATION 2]      OCTET STRING,
         application           [APPLICATION 15]     OCTET STRING
         }DEFAULT oid: {1 0 24787 2018},   -- or whatever OID is registered

      sbData                   [APPLICATION 16]   SET
         {
         field0   [0] OCTET STRING (SIZE(1)) OPTIONAL,
         field1   [1] OCTET STRING (SIZE(1)) OPTIONAL,
         field2   [2] OCTET STRING (SIZE(1)) OPTIONAL
         }
      }
*/

END
*/
```

13.32 An example of a simple BIR in XML encoding (complying with the ASN.1 schema, the XSD schema, and the normative textual description)

```
<?xml version="1.0" encoding="utf-8"?>
<BIR xmlns="http://standards.iso.org/iso-iec/19785/-3/ed-3/en">
	<Version>
		<Major>3</Major>
		<Minor>0</Minor>
	</Version>
	<CBEFFVersion>
		<Major>4</Major>
		<Minor>0</Minor>
	</CBEFFVersion>
	<BIRInfo>
		<Creator>ABCDE</Creator>
		<Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>
		<Payload>UjBsR09EbGhjZ0dTQUxNQUFBUUNBRU1tQ1p0dU1GUXhEUzhi
</Payload>
		<Integrity>true</Integrity>
		<CreationDate>2004-03-02T15:03:15Z</CreationDate>
		<NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
		<NotValidAfter>2004-03-03T15:00:00Z</NotValidAfter>
	</BIRInfo>
	<BDBInfo>
<ChallengeResponse>dTQUxNQUFBUUNBRU1tQ1UjBsR09EbGhjZ0p0dU1GUXhEU
zhi</ChallengeResponse>
<Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>
<Format>
	<Organization>51</Organization>
	<Type>99</Type>
</Format>
<Encryption>true</Encryption>
<CreationDate>2004-03-02T15:00:00Z</CreationDate>
<NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
<NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
<Type>Iris</Type>
<Subtype>Left</Subtype>
<Level>Processed</Level>
<Product>
	<Organization>16</Organization>
	<Type>2</Type>
</Product>
	<Purpose>Verify</Purpose>
	<Quality>
		<Algorithm>
			<Organization>4</Organization>
			<Type>9</Type>
		</Algorithm>
		<Score>100</Score>
	</Quality>
</BDBInfo>
<SBInfo>
	<Format>
		<Organization>51</Organization>
		<Type>99</Type>
	</Format>
</SBInfo>
<BDB>Q1UjBsR09EbGhjZ0p0dU1GUXhEUzhidTQUxNQUFBUUNBRU1t</BDB>
<SB>1tQ1UjBsR09EbGhjZ0p0dU1GUXhEUzhidTQUxNQUFBUUNBRU</SB>
</BIR>
```

13.33 An example of a complex BIR in XML encoding (complying with the ASN.1 schema, the XSD schema, and the normative textual description)

```
<?xml version="1.0" encoding="utf-8"?>
<BIR xmlns="http://standards.iso.org/iso-iec/19785/-3/ed-3/en">
	<Version>
	    <Major>3</Major>
	    <Minor>0</Minor>
```

```
            </Version>
            <CBEFFVersion>
                <Major>4</Major>
                <Minor>0</Minor>
            </CBEFFVersion>
            <BIRInfo>
                <Creator>ABCDE</Creator>
                <Index>86CA3100-43F3-0D23-A941-7871E519A00E</Index>
                <Payload>UjBsR09EbGhjZ0dTQUxNQUFBUUNBRU1tQ1p0dU1GUXhEUzhi</Payload>
                <Integrity>true</Integrity>
                <CreationDate>2004-03-02T15:03:15Z</CreationDate>
                <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
                <NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
            </BIRInfo>
            <SBInfo>
                <Format>
                    <Organization>51</Organization>
                    <Type>99</Type>
                </Format>
            </SBInfo>
            <BIR>
                <BIRInfo>
                    <Creator>ABCDE</Creator>
                    <Index>310086CA-43F3-0D23-A941-7871E519A00E</Index>
                    <Payload>09EbUjBsRGhjZ0dTQUxNQUFBUUNBRU1tQ1p0dU1GUXhEUzhi
</Payload>
                    <Integrity>false</Integrity>
                    <CreationDate>2004-03-02T00:00:00Z</CreationDate>
                    <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
                    <NotValidAfter>2004-03-02T15:33:00Z</NotValidAfter>
                </BIRInfo>
                <BDBInfo>
                    <ChallengeResponse>O2Rmc2RmZHNmZzM0NmVydGZnZmQ=
</ChallengeResponse>
                    <Index>310086CA-43F3-0D23-A941-7871E519A00E</Index>
                    <Format>
                        <Organization>51</Organization>
                        <Type>99</Type>
                    </Format>
                    <Encryption>true</Encryption>
                    <CreationDate>2004-03-02T15:00:00Z</CreationDate>
                    <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
                    <NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
                    <Type>Iris</Type>
                    <Subtype>Left</Subtype>
                    <Level>Processed</Level>
                    <Product>
                        <Organization>16</Organization>
                        <Type>2</Type>
                    </Product>
                    <Purpose>Verify</Purpose>
                    <Quality>
                        <Algorithm>
                            <Organization>4</Organization>
                            <Type>9</Type>
                        </Algorithm>
                        <Score>100</Score>
                    </Quality>
                </BDBInfo>
                <SBInfo>
                    <Format>
                        <Organization>51</Organization>
                        <Type>99</Type>
                    </Format>
                </SBInfo>
                <BDB>VGhpcyBpcyBhbiBJU08gc3RhbmRhcmQ=</BDB>
                <SB>U2VjdXJpdHkgQmxvY2s=</SB>
            </BIR>
            <BIR>
                <BIRInfo>
                    <Creator>ABCDE</Creator>
```

```
                <Index>00130224-0D23-1193-BEAD-7871E519A00E</Index>
                <Payload>UGF5bG9hZCBnb2VzIGhlcmU=</Payload>
                <Integrity>true</Integrity>
                <CreationDate>2004-03-02T15:00:00Z</CreationDate>
                <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
                <NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
            </BIRInfo>
            <BDBInfo>
                <ChallengeResponse>Q2hhbGxlbmdlUmVzcG9uc2UgZ29lcyBoZXJl
</ChallengeResponse>
                <Index>00130224-0D23-1193-BEAD-7871E519A00E</Index>
                <Format>
                    <Organization>51</Organization>
                    <Type>88</Type>
                </Format>
                <Type>Iris</Type>
                <Level>Processed</Level>
                <Product>
                    <Organization>51</Organization>
                    <Type>88</Type>
                </Product>
                <Purpose>Enrol</Purpose>
            </BDBInfo>
            <SBInfo>
                <Format>
                    <Organization>51</Organization>
                    <Type>99</Type>
                </Format>
            </SBInfo>
            <BIR>
                <BIRInfo>
                    <Integrity>false</Integrity>
                </BIRInfo>
                <BDBInfo>
                    <Encryption>true</Encryption>
                    <CreationDate>2004-03-02T15:00:00Z</CreationDate>
                    <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
                    <NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
                    <Subtype>Left</Subtype>
                    <Quality>
                        <Algorithm>
                            <Organization>4</Organization>
                            <Type>9</Type>
                        </Algorithm>
                        <Score>90</Score>
                    </Quality>
                </BDBInfo>
                <SBInfo>
                    <Format>
                        <Organization>51</Organization>
                        <Type>99</Type>
                    </Format>
                </SBInfo>
                <BDB>QmlvbWV0cmljRGF0YUJsb2NrIGdvZXMgaGVyZQ==</BDB>
                <SB>U2VjdXJpdHkgQmxvY2s=</SB>
            </BIR>
            <BIR>
                <BIRInfo>
                    <Integrity>false</Integrity>
                </BIRInfo>
                <BDBInfo>
                    <Encryption>true</Encryption>
                    <CreationDate>2004-03-02T15:00:00Z</CreationDate>
                    <NotValidBefore>2004-03-02T15:00:00Z</NotValidBefore>
                    <NotValidAfter>2004-03-02T15:00:00Z</NotValidAfter>
                    <Subtype>Right</Subtype>
                    <Quality>
                        <Algorithm>
                            <Organization>4</Organization>
                            <Type>9</Type>
                        </Algorithm>
```

```
                        <QualityCalculationFailed />
                    </Quality>
                </BDBInfo>
                <SBInfo>
                    <Format>
                        <Organization>51</Organization>
                        <Type>99</Type>
                    </Format>
                </SBInfo>
                <BDB>QmlvbWV0cmljRGF0YUJsb2NrIGdvZXMgaGVyZQ==</BDB>
                <SB>QSBTQiBzaG91bGQgZ29lcyBoZXJl</SB>
            </BIR>
            <SB>QW5vdGhlciBTQiBzaG91bGQgZ29lcyBoZXJl</SB>
        </BIR>
        <SB>QSBmaW5hbCBTQiBzaG91bGQgZ29lcyBoZXJl</SB>
    </BIR>
```

14 Patron format specification: full-complex patron format

14.1 Patron

ISO/IEC JTC 1/SC 37-

14.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

14.3 Patron format name

ISO/IEC JTC 1/SC 37 full-complex patron format

14.4 Patron format type

10 (000AHex). This has been registered in accordance with ISO/IEC 19785-2.

14.5 ASN.1 OID for this patron format

```
{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257)
patron-format(1) full-complex(10)}
```

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.10</OBJECT_IDENTIFIER>
```

14.6 Domain of use

This clause provides a definition of a patron format that may be of general utility to applications that need to carry one or more BIRs (of either the same or different patron formats) in a single complex BIR structure, with explicit identification of the patron format(s) being used.

This patron format is identical to that specified in Clause 12 except for the addition of twelve new data elements (capture device owner and identifier, feature extraction algorithm owner and identifier, comparison algorithm owner and identifier, quality algorithm owner and identifier, PAD mechanism and PAD mechanism vendor, and compression algorithm owner and identifier) and a "fieldPresence" field that is one byte longer.

14.7 Version identifier

This patron format specification has a version identifier of 2.

NOTE The patron format version was increased from 1 to 2 because the QUALITY NOT SUPPORTED BY BDB CREATOR and QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET values were modified. Additionally, the CBEFF_biometric_type and CBEFF_biometric_subtype values were changed.

14.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0).

14.9 General

14.9.1 Patron Format Structure Type: Complex CBEFF BIR without self-identification.

14.9.2 This patron format supports all the mandatory and optional data elements specified in ISO/IEC 19785-1. It can support either a simple BIR or a complex BIR structure where each intermediate node or leaf of the structure is itself a BIR (called a "child BIR") and can be represented in any patron format.

14.9.3 The patron format of each child BIR is explicitly identified in its parent by a pair: patron format owner / patron format type, and can be either this patron format (in which case the child BIR may itself have children), or a different patron format (in which case the child BIR is considered a leaf of this patron format although it may be a complex BIR in its own regard).

14.9.4 Most fields in this patron format are optional. The presence of each optional field is encoded as a single bit of a 32-bit field ("fieldPresence") at the beginning of the format, which has one bit for each optional field defined in the patron format. The bit value '1' in a given position of that field means that the corresponding field is present in the BIR instance.

14.9.5 All character strings and octet strings are preceded by a length prefix, which can be one, two, or four octets long, as specified for each field.

14.9.6 All integer values, including lengths, are encoded in big-endian order.

14.9.7 Dates and date intervals are encoded as character strings in a way conforming to the ISO 8601 series.

14.9.8 An instance of a BIR or child BIR contains either a BDB or one or more BIR children, but never contains both.

14.10 Specification

An instance of a BIR shall contain the fields specified below, in exactly the same order and with no gaps between the fields.

Table 17 — Specifications for full-complex patron format

CBEFF data element name	Field name	Length (in octets) and optionality[a,d]	Abstract values and encodings[b]
The following fields shall occur at most once			
CBEFF_patron_header_version	patronHeaderVersion	1, mandatory	2
CBEFF_version	cbeffVersion	1, mandatory	Major '4' and Minor '0': 40_{Hex} (64)
not a standard CBEFF data element	fieldPresence	4, mandatory	A 32-bit field containing one bit for each optional field in the patron format. The bit value '1' means that the corresponding field is present in the BIR instance. Bit position (1=most significant, 32=least significant) and corresponding optional field: 1 bdbFormat Owner & Type 2 bdbEncryption 3 bdbBiometricType 4 bdbBiometricSubtype 5 bdbChallengeResponse 6 bdbCreationDate 7 bdbIndex 8 bdbProcessedLevel 9 bdbProduct Owner & Type 10 bdbCaptureDevice Owner & Type 11 bdbFeatureExtAlg Owner & Type 12 bdbComparisonAlg Owner & Type 13 bdbQualityAlg Owner & Type 14 bdbCompressionAlg Owner & Type 15 bdbPADMechanism & Vendor

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

[d] All fields are optional unless otherwise indicated.

Table 17 *(continued)*

CBEFF data element name	Field name	Length (in octets) and optionality[a,d]	Abstract values and encodings[b]
			16 bdbPurpose 17 bdbQuality 18 bdbValidityPeriod 19 birCreationDate 20 birCreator 21 birIndex 22 birPayload 23 birValidityPeriod 24 sbFormat Owner & Type 25 bdb 26 sb 27..32 not used (shall be '0')
CBEFF_BDB_format_owner	bdbFormatOwner	2, mandatory if a BDB is present, optional if a BDB is not present.	1..65535; field not present corresponds to "NO VALUE AVAILABLE
CBEFF_BDB_format_type	bdbFormatType	2, mandatory if a BDB is present, optional if a BDB is not present.	1..65535; field not present corresponds to"NO VALUE AVAILABLE"
CBEFF_BDB_encryption_options	bdbEncryption	1; mandatory if a BDB is present, otherwise required to be absent.	NO ENCRYPTION: 0 ENCRYPTION: 1
CBEFF_BIR_integrity_options	birIntegrity	1, mandatory	NO INTEGRITY: 0 INTEGRITY: 1
CBEFF_BDB_biometric_type	bdbBiometricType	3	*This encoding is a 3 octet bitmap. NO VALUE AVAILABLE is encoded as all 0 bits. If MULTIPLE BIOMETRIC TYPES is set, other bits may also be set to enumerate the types contained in the BDB.* Values can be found in 6.2 under the BiometricType ASN.1 definition.
CBEFF_BDB_biometric_subtype	bdbBiometricSubtype	1	*This encoding is a 1 octet bitmap.* Values can be found in 6.2 under the BiometricSubtype ASN.1 definition.

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

[d] All fields are optional unless otherwise indicated.

Table 17 *(continued)*

CBEFF data element name	Field name	Length (in octets) and optionality[a,d]	Abstract values and encodings[b]
CBEFF_BDB_challenge_response	bdbChallengeResponse	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets).
CBEFF_BDB_creation_date	bdbCreationDate	1 + 8..15	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters). The string shall represent a date (or date and a time of the day)[c].
CBEFF_BDB_index	bdbIndex	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets). Shall not appear in any BIR in which numChildren is not x'00'.
CBEFF_BDB_processed_level	bdbProcessedLevel	1	RAW: 1 INTERMEDIATE: 2 PROCESSED: 3
CBEFF_BDB_product_owner	bdbProductOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_product_type	bdbProductType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_capture_device_owner	bdbCaptureDeviceOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_capture_device_type	bdbCaptureDeviceType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_feature_extraction_algorithm_owner	bdbFeatureExtAlgOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_feature_extraction_algorithm_type	bdbFeatureExtAlgType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_comparison_algorithm_owner	bdbComparisonAlgOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_comparison_algorithm_type	bdbComparisonAlgType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_quality_algorithm_owner	bdbQualityAlgOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_quality_algorithm_type	bdbQualityAlgType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_compression_algorithm_owner	bdbCompressionAlgOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_compression_algorithm_type	bdbCompressionAlgType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_PAD_mechanism_vendor	bdbPADMechanismOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
CBEFF_BDB_PAD_mechanism	bdbPADMechanismType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

[d] All fields are optional unless otherwise indicated.

Table 17 *(continued)*

CBEFF data element name	Field name	Length (in octets) and optionality[a,d]	Abstract values and encodings[b]
CBEFF_BDB_purpose	bdbPurpose	1	VERIFY: 1 IDENTIFY: 2 ENROL: 3 ENROL FOR VERIFICATION ONLY: 4 ENROL FOR IDENTIFICATION ONLY: 5 AUDIT: 6
CBEFF_BDB_quality	bdbQuality	1	QUALITY NOT SUPPORTED BY BDB CREATOR: 254 QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET: 255 INTEGER VALUE: 0 – 100
CBEFF_BDB_validity_period	bdbValidityPeriod	1 + 17..31	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters). The string shall represent an interval of two dates (or date and time of the day)[d].
CBEFF_BIR_creation_date	birCreationDate	1 + 8..15	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters). The string shall represent a date (or date and a time of the day)[c].
CBEFF_BIR_creator	birCreator	2 + 0..65535	Variable-length ISO/IEC 10646 character string, encoded in UTF-8, and preceded by a 16-bit integer field containing the length of the UTF-8 encoding (octets).
CBEFF_BIR_index	birIndex	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets). Shall not inherit its value from any other level BIR.
CBEFF_BIR_payload	birPayload	2 + 0..65535	Variable-length octet string, preceded by a 16-bit integer field containing the length (octets). Shall not inherit its value from any other level BIR.
CBEFF_BIR_validity_period	birValidityPeriod	1 + 17..31	Variable-length ASCII character string, preceded by an 8-bit integer field containing the length (characters). The string shall represent an interval of two dates (or date and time of the day)[d].
CBEFF_SB_format_owner	sbFormatOwner	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

[d] All fields are optional unless otherwise indicated.

Table 17 *(continued)*

CBEFF data element name	Field name	Length (in octets) and optionality[a,d]	Abstract values and encodings[b]
CBEFF_SB_format_type	sbFormatType	2	1..65535; field not present corresponds to "NO VALUE AVAILABLE"
BDB	bdb	4 + 0..4294967295	Variable-length octet string, preceded by a 32-bit integer field containing the length (octets). If this field is present in a BIR instance (as indicated in bit 24 of the field *fieldPresence*), then no child BIRs shall be included (*numChildren* shall have the value 0). Otherwise, at least one child BIR shall be included (*numChildren* shall have a value greater than 0). NOTE The content and encoding of the BDB are not specified by CBEFF nor by this patron format specification.
CBEFF_subheader_count	numChildren	1, mandatory	0..255
The following 3 fields shall occur as a group as many times as specified in the field numChildren (0..255)			
CBEFF_BIR_patron_format_owner	childBirPatronFormatOwner	2, mandatory if no BDB is present, otherwise required to be absent.	0..65535, where 0 = "NO VALUE AVAILABLE
CBEFF_BIR_patron_format_type	childBirPatronFormatType	2, mandatory if no BDB is present, otherwise required to be absent.	0..65535, where 0 = "NO VALUE AVAILABLE
not a standard CBEFF data element	childBir	4 + 0..4294967295, mandatory if no BDB is present, otherwise required to be absent.	Variable-length octet string, preceded by a 32-bit integer field containing the length (octets)[e].
The following field shall occur at most once			
SB	sb	4 + 0..4294967295	Variable-length octet string, preceded by a 32-bit integer field containing the length (octets).

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted.

Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits.

Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR consists of either: 1) an SBH, BDB, optional SB, and numChildren value of zero, or 2) an SBH, no BDB, and numChildren value greater than zero.

[d] All fields are optional unless otherwise indicated.

14.11 Illustrative examples

Table 18 — Simple BIR (one BDB)

Field name	Length	Abstract value	Encoding
patronHeaderVersion	1	2	02_{Hex}
cbeffVersion	1	Major 4, Minor 0	40_{Hex}
fieldPresence	4	bdbFormatOwner and Type bdbEncryption bdbBiometricType bdbQuality bdb	$E0200020_{Hex}$
bdbFormatOwner	2	ISO/IEC JTC 1/SC 37	257 (0101_{Hex})
bdbFormatType	2	Face image	0008_{Hex}
bdbEncryption	1	NO ENCRYPTION	00_{Hex}
birIntegrity	1	NO INTEGRITY	00_{Hex}
bdbBiometricType	3	FACE-IMAGE	000002_{Hex}
bdbQuality	1	75/100	$4B_{Hex}$
Bdb	4 + 4096	octet string	00001000_{Hex} + 4096 octets
numChildren	1	zero	00_{Hex}

Table 19 — Complex BIR fields and abstract values corresponding to ISO/IEC 19785-1, Figure 2

1. patronHeaderVersion = 2 *(beginning of the root header BIR)*
2. cbeffVersion = 4:0
3. fieldPresence = sbFormatOwner/Type
4. birIntegrity = INTEGRITY *(integrity is applied to the entire complex BIR via the SB on line 90)*
5. sbFormatOwner = *a security vendor*
6. sbFormatType = *that vendor's security block format* *(see the final SB on line 90)*
7. numChildren = 2
8. childBirPatronFormatOwner = SC 37
9. childBirPatronFormatType = 10 *(this format)*
10. ► *(denotes the beginning of the next BIR)*
11. patronHeaderVersion = 2
12. cbeffVersion = 4:0
13. fieldPresence = bdbBiometricType
14. birIntegrity = NO INTEGRITY
15. bdbBiometricType = FINGER *(the next 3 BIRs inherit this value)*
16. numChildren = 3
17. childBirPatronFormatOwner = SC 37
18. childBirPatronFormatType = 10 (*this format*)
19. ►
20. patronHeaderVersion = 2
21. cbeffVersion = 4:0
22. fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; bdb
23. bdbFormatOwner = SC 37
24. bdbFormatType = *a standardized BDB format*

Table 19 *(continued)*

25. bdbEncryption = NO ENCRYPTION
26. birIntegrity = NO INTEGRITY
27. bdbBiometricSubtype = LEFT INDEX FINGER
28. Bdb
29. numChildren=0
30. ►
31. patronHeaderVersion = 2
32. cbeffVersion = 4:0
33. fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; bdb
34. bdbFormatOwner = *vendor ABC*
35. bdbFormatType = *non standard BDB format A*
36. bdbEncryption = NO ENCRYPTION
37. birIntegrity = NO INTEGRITY
38. bdbBiometricSubtype = LEFT MIDDLE FINGER
39. Bdb
40. numChildren=0
41. ►
42. patronHeaderVersion = 2
43. cbeffVersion = 4:0
44. fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; bdb
45. bdbFormatOwner = *vendor XYZ*
46. bdbFormatType = *non standard BDB format B*
47. bdbEncryption = NO ENCRYPTION
48. birIntegrity = NO INTEGRITY
49. bdbBiometricSubtype = LEFT RING FINGER
50. Bdb
51. numChildren=0
52. ►
53. patronHeaderVersion = 2
54. cbeffVersion = 4:0
55. fieldPresence = bdbBiometricType
56. birIntegrity = NO INTEGRITY
57. bdbBiometricType = IRIS *(the next 2 BIRs inherit this type)*
58. numChildren = 2
59. childBirPatronFormatOwner = SC 37
60. childBirPatronFormatType = 10 (*this format*)
61. ►
62. patronHeaderVersion = 2
63. cbeffVersion = 4:0
64. fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; sbFormatOwner/Type; bdb; sb
65. bdbFormatOwner = SC 37
66. bdbFormatType = *an iris format*

Table 19 *(continued)*

67.	bdbEncryption = ENCRYPTION	
68.	birIntegrity = NO INTEGRITY	
69.	bdbBiometricSubtype = LEFT	
70.	sbFormatOwner = *an encryption vendor*	
71.	sbFormatType = *a security block format*	*(see SB on line 74)*
72.	bdb	
73.	numChildren=0	
74.	sb *(see SB format identifier on lines 70-71)*	
75.	►	
76.	patronHeaderVersion = 2	
77.	cbeffVersion = 4:0	
78.	fieldPresence = bdbFormatOwner/Type; bdbEncryption; bdbBiometricSubtype; sbFormatOwner/Type; bdb; sb	
79.	bdbFormatOwner = *vendor PQR*	
80.	bdbFormatType = *vendor's format C*	
81.	bdbEncryption = ENCRYPTION	
82.	birIntegrity = NO INTEGRITY	
83.	bdbBiometricSubtype = RIGHT	
84.	sbFormatOwner = *an encryption vendor*	
85.	sbFormatType = *a security block format*	*(see SB on line 88)*
86.	Bdb	
87.	numChildren=0	
88.	Sb	*(see SB format identifier on lines 84-85)*
89.	►	
90.	Sb	*(see SB format identifier in root header on line 6)*

Table 20 — BIR wrapped in an enveloping BIR

Field Name	Length	Abstract value	Encoding
patronHeaderVersion	1	2	02_{Hex}
cbeffVersion	1	Major 4, Minor 0	40_{Hex}
fieldPresence	4	all optional fields absent in the enveloping BIR	00000000_{Hex}
birIntegrity	1	NO INTEGRITY	00_{Hex}
numChildren	1	one child *(the enveloped BIR)*	01_{Hex}
childBirPatronFormatOwner)	2	patron format owner of the enveloped BIR	*variable*
childBirPatronFormatType	2	patron format type of the enveloped BIR	*variable*
childBir (length of the child BIR)	4	length of the enveloped BIR	*variable*
childBir (octets of the child BIR	*variable*	octets of the enveloped BIR	*variable*

Table 20 shows how the Complex patron format specified in this clause can be used as a simple envelope around a BIR of an arbitrary patron format in order to provide identification of its format and specify its length. When using the Complex patron format in this way, the portion of the enveloping BIR preceding the enveloped BIR can be thought of as a fixed-length prefix to the enveloped BIR. Since all the optional fields of the enveloping BIR are absent, the length of the prefix is only 16 octets, given by:

a) 8 octets with the fixed values 0120000000000001_{Hex}; plus;

b) 4 octets containing the patron format owner and type of the enveloped BIR; plus;

c) 4 octets containing the length of the enveloped BIR.

14.12 ASN.1 definition (provided for illustrative purposes only)

The following ASN.1 specification provides an abstract description of the patron format and can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en. This ASN.1 specification may be useful to some readers of this document. It is not intended to provide an alternative specification of the encodings of this patron format.

```
CBEFF-FULL-COMPLEX-PATRON-FORMAT
{iso(1) standard(0) cbeff(19785) modules(0) full-complex(10)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

  IMPORTS
    RegistryID, BDBFormat, BiometricType, BiometricSubtype,
    ChallengeResponse, BDBIndex, ProcessedLevel, Product, Purpose,
    Quality, Creator, BIRIndex, Payload
          FROM CBEFF-DATA-ELEMENTS;

  BIR ::= SEQUENCE
    {
    patronHeaderVersion INTEGER(0..255),
    cbeffVersion        INTEGER(0..255),
    fieldPresence       SEQUENCE
      {
      bdbFormat              BOOLEAN,
      bdbEncryption          BOOLEAN,
      bdbBiometricType       BOOLEAN,
      bdbBiometricSubtype    BOOLEAN,
      bdbChallengeResponse   BOOLEAN,
      bdbCreationDate        BOOLEAN,
      bdbIndex               BOOLEAN,
      bdbProcessedLevel      BOOLEAN,
      bdbProduct             BOOLEAN,
      bdbCaptureDevice       BOOLEAN,
      bdbFeatureExtAlg       BOOLEAN,
      bdbComparisonAlg       BOOLEAN,
      bdbQualityAlg          BOOLEAN,
      bdbCompressionAlg      BOOLEAN,
      bdbPurpose             BOOLEAN,
      bdbQuality             BOOLEAN,
      bdbValidityPeriod      BOOLEAN,
      birCreationDate        BOOLEAN,
      birCreator             BOOLEAN,
      birIndex               BOOLEAN,
      birValidityPeriod      BOOLEAN,
      sbFormat               BOOLEAN,
      bdb                    BOOLEAN,
      children               BOOLEAN,
      sb                     BOOLEAN
      },
    bdbFormat              BDBFormat OPTIONAL,
    bdbEncryption          INTEGER(0..255) OPTIONAL,
    birIntegrity           INTEGER(0..255),
    bdbBiometricType       BiometricType DEFAULT noValueAvailable,
    bdbBiometricSubtype    BiometricSubtype DEFAULT noValueAvailable,
    bdbChallengeResponse   ChallengeResponse OPTIONAL,
    bdbCreationDate        OCTET STRING (SIZE(8..15)) OPTIONAL,
    bdbIndex               BDBIndex OPTIONAL,
    bdbProcessedLevel      ProcessedLevel OPTIONAL,
    bdbProduct             Product OPTIONAL,
    bdbCaptureDevice       RegistryID OPTIONAL,
    bdbFeatureExtAlg       RegistryID OPTIONAL,
    bdbComparisonAlg       RegistryID OPTIONAL,
```

```
        bdbQualityAlg           RegistryID OPTIONAL,
        bdbCompressionAlg       RegistryID OPTIONAL,
        bdbPurpose              Purpose OPTIONAL,
        bdbQuality              Quality OPTIONAL,
        bdbValidityPeriod       OCTET STRING (SIZE(15..31)) OPTIONAL,
        birCreationDate         OCTET STRING (SIZE(8..15)) OPTIONAL,
        birCreator              Creator OPTIONAL,
        birIndex                BIRIndex OPTIONAL,
        birPayload              Payload OPTIONAL,
        birValidityPeriod       OCTET STRING (SIZE(15..31)) OPTIONAL,
        sbFormat                RegistryID OPTIONAL,
        bdb                     OCTET STRING (SIZE(0..4294967295)) OPTIONAL,
        children                SEQUENCE (SIZE(0..255)) OF
          child SEQUENCE
            {
            childBirPatronFormat    RegistryID,
            childBir                OCTET STRING (SIZE(0..4294967295))
            },
        sb OCTET STRING (SIZE(0..4294967295)) OPTIONAL
        }
    END
```

15 Patron format specification: Biometric application programming interface (BioAPI 2.0)

This patron format is to be used in the context of BioAPI 2.0, and is therefore defined in ISO/IEC 19784-1 as the ISO/IEC JTC 1/SC 37 BioAPI patron format with ASN.1 identifier {iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) bioAPI (8)}}.

16 Patron format specification: Self-identifying Tag-oriented Simple BIR

16.1 Patron

ISO/IEC JTC 1/SC 37.

This patron format was developed from the Simple BIR defined for Object-Oriented BioAPI (see ISO/IEC 30106-1), but it is specified and generalized here for use in any environment, even outside the scope of ISO/IEC 30106-1. The patron format shall be coded in a tagged data format. Specifications for the encoding in a selection of popular tagged formats are given in Annex C.

16.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

16.3 Patron format name

ISO/IEC JTC 1/SC 37 self-identifying-tag-oriented-simple-bir patron format

16.4 Patron format type

12 (000CHex). This has been registered in accordance with ISO/IEC 19785-2.

16.5 ASN.1 OID for this patron format

{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) self-id-tagged-simple(12)}

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.12</OBJECT_IDENTIFIER>
```


16.6 Domain of use

This clause provides a definition of a patron format that may be of general utility to applications that need to carry BIRs in a complete simple BIR structure, with optional support for the Multiple CBEFF BIR structure (see 16.9). Regardless of whether the simple BIR or Multiple CBEFF BIR structure is used, only a single BDB may be included any BIR of this patron format.

16.7 Version identifier

This patron format specification has a version identifier of (major 1, minor 0).

16.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0).

16.9 General

16.9.1 Patron Format Structure Type: Simple CBEFF BIR with self-identification and optional support for Multiple CBEFF BIR. This patron format includes optional CBEFF data elements needed to implement Multiple CBEFF BIR structures. Specifically, this patron format includes an optional CBEFF_BIR_pointer field which may be used to point to subsequent BIRs.

When subsequent BIRs are included in this patron format, a linear structure shall be used such that each subsequent BIR points to the next, and the sequence is terminated by a BIR where the CBEFF_BIR_pointer is not included or has a value of 0. This is also known as the end-to-end structure and is depicted in ISO/IEC 19785-1:2020 Figure 3. Since this patron format is tag-oriented and does not support memory location pointers, the CBEFF_BIR_pointer will indicate the location of the next BIR using a URL or file path.

However, if the CBEFF_BIR_pointer is not included or has a value of 0, this patron format represents a Simple CBEFF BIR. The inclusion of the optional CBEFF_BIR_pointer field is intended to provide flexibility in the use of this patron format. If a Simple CBEFF BIR meets implementation needs, then the Multiple CBEFF BIR concept can be ignored.

16.9.2 This patron format supports many of the mandatory and optional data elements specified in ISO/IEC 19785-1. For a complete list of supported fields, see Table 21.

16.9.3 In order to allow future backwards interoperability, this patron format shall be coded with a tagged data format (e.g. TLV, XML or JSON). Most fields in this patron format are optional. Those optional parameters, if absent, will not include their relevant tags in the data structure.

Tagged data formats can be parsed and processed without using the overall BIR length value. Therefore, this patron format does not include a BIR length field.

16.9.4 All character strings and octet strings are preceded by a length prefix, which can be one, two, or four octets long, as specified for each field. The tagged data format chosen will adapt such a length prefix according to the best practices within such data format.

EXAMPLE In TLV data coding, the length will be part of the L sub-field, and not the V sub-field.

16.9.5 All integer values, including lengths, are encoded in big-endian order.

16.9.6 Dates and date intervals are encoded as character strings in a way conforming to the ISO 8601 series.

16.9.7 The BDB field is mandatory in this patron format.

16.10 Specification

An instance of a BIR shall contain the fields specified below. The order of the fields is not fixed for this tag-oriented format.

Table 21 — Specifications for self-identifying, tag-oriented simple BIR patron format

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
The following fields shall occur at most once			
SBIR	SBir	container, mandatory	Container of the whole data format (53424952_{Hex}, i.e.'SBIR" ASCII)
The following fields shall be within SBIR container and shall occur at most once			
CBEFF_BIR_self_id_owner	BirSelfIdOrganization	2, mandatory	0101_{Hex}
CBEFF_BIR_self_id_type	BirSelfIdId	2, mandatory	$000C_{Hex}$
CBEFF_version	cbeffVersion	2, mandatory	Major '4' and Minor '0': 0400_{Hex}
CBEFF_patron_header_version	patronHeaderVersion	2, mandatory	Major '1' and Minor '0': 0100_{Hex}
The following are fields related to the BIR	birInfo	container, mandatory	
CBEFF_BIR_integrity_options	birIntegrity	1, mandatory	NO INTEGRITY: 0 INTEGRITY: 1
CBEFF_BIR_creation_date	birCreationDate	8..15, optional	Variable-length ASCII character string. The string shall represent a date (or date and a time of the day)[a].
CBEFF_BIR_validity_period	birValidityPeriod	17..31, optional	Variable-length ASCII character string. The string shall represent an interval of two dates (or date and time of the day)[b].
CBEFF_BIR_creator	birCreator	0..65535, optional	Variable-length ISO/IEC 10646 character string, encoded in UTF-8.
CBEFF_BIR_index	birIndex	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.
CBEFF_BIR_payload	birPayload	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.
CBEFF_BIR_pointer	birPointer	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR. NO SUSEQUENT BIR is represented by birPointer value = 0 or birPointer not present.
The following are fields related to the BDB	bdbInfo	container, mandatory	
CBEFF_BDB_format_owner	bdbFormatOrganization	2, mandatory	0..65535, where 0 = "NO VALUE AVAILABLE
CBEFF_BDB_format_type	bdbFormatId	2, mandatory	0..65535, where 0 = "NO VALUE AVAILABLE

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits. Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

Table 21 *(continued)*

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
CBEFF_BDB_encryption_options	bdbEncryption	1; mandatory	NO ENCRYPTION: 0 ENCRYPTION: 1
CBEFF_BDB_biometric_type	bdbBiometricType	3, optional	Values can be found in 6.2 under the BiometricType ASN.1 definition.
CBEFF_BDB_biometric_subtype	bdbBiometricSubtype	1, optional	Values can be found in 6.2 under the BiometricSubtype ASN.1 definition.
CBEFF_BDB_processed_level	bdbProcessedLevel	1, optional	RAW: 1 INTERMEDIATE: 2 PROCESSED: 3
CBEFF_BDB_purpose	bdbPurpose	1, optional	VERIFY: 1 IDENTIFY: 2 ENROL: 3 ENROL FOR VERIFICATION ONLY: 4 ENROL FOR IDENTIFICATION ONLY: 5 AUDIT: 6
CBEFF_BDB_creation_date	bdbCreationDate	8..15, optional	Variable-length ASCII character string. The string shall represent a date (or date and a time of the day)[a].
CBEFF_BDB_validity_period	bdbValidityPeriod	17..31, optional	Variable-length ASCII character string. The string shall represent an interval of two dates (or date and time of the day)[b].
CBEFF_BDB_challenge_response	bdbChallengeResponse	0..65535, optional	Variable-length octet string.
CBEFF_BDB_index	bdbIndex	0..65535, optional	Variable-length octet string.
CBEFF_BDB_product_owner	bdbProductOrganization	2, optional	1..65535
CBEFF_BDB_product_type	bdbProductId	2, optional	1..65535
CBEFF_BDB_capture_device_owner	bdbCaptureDeviceOrganization	2, optional	1..65535
CBEFF_BDB_capture_device_type	bdbCaptureDeviceId	2, optional	1..65535
CBEFF_BDB_feature_extraction_algorithm_owner	bdbFeatureExtractionAlgorithmOrganization	2, optional	1..65535
CBEFF_BDB_feature_extraction_algorithm_owner_type	bdbFeatureExtractionAlgorithmId	2, optional	1..65535
CBEFF_BDB_comparison_algorithm_owner	bdbComparisonAlgorithmOrganization	2, optional	1..65535
CBEFF_BDB_comparison_algorithm_type	bdbComparisonAlgorithmId	2, optional	1..65535

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits. Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

Table 21 *(continued)*

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
CBEFF_BDB_compression_algorithm_owner	bdbCompressionAlgorithmOrganization	2, optional	1..65535
CBEFF_BDB_compression_algorithm_type	bdbCompressionAlgorithmId	2, optional	1..65535
CBEFF_BDB_quality	bdbConstructedQuality	5, optional	Created as a constructed element of bdbQuality, bdbQualityAlgoritmOrganization, bdbQualityAlgorithmId
CBEFF_BDB_quality	bdbQuality	1, optional (use bdbConstructedQuality)	QUALITY NOT SUPPORTED BY BDB CREATOR: 254 QUALITY SUPPORTED BY BDB CREATOR BUT NOT SET: 255 (FAILED) INTEGER VALUE: 0 – 100
CBEFF_BDB_quality_algorithm_owner	bdbQualityAlgoritmOrganization	2, optional (use bdbConstructedQuality)	1..65535
CBEFF_BDB_quality_algorihtm_type	bdbQualityAlgorithmId	2, optional (use bdbConstructedQuality)	1..65535
CBEFF_BDB_PAD_mechanism_vendor	bdbPadMechanismVendor	2, optional	1..65535
CBEFF_BDB_PAD_mechanism	bdbPadMechanismId	2, optional	1..65535
The following are conditional fields related to the SB, mandatory only if a SB is included in the BIR	sbInfo	container, conditional	
CBEFF_SB_format_owner	sbFormatOrganization	2, conditional	1..65535
CBEFF_SB_format_type	sbFormatId	2, conditional	1..65535
The following field shall occur only once			
BDB	bdb	0..4294967295	Variable-length octet string.
The following field is optional, and if it exists, it shall only be once			
SB	sb	0..4294967295, conditional	Variable-length octet string.

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits. Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

16.11 Illustrative examples

Illustrative examples of tag-oriented patron formats are included in C.6. The child BIR example uses the format of the Self-identifying Tag-oriented Simple BIR patron format.

16.12 ASN.1 definition

The following ASN.1 specification provides an abstract description of the patron format and can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en. This ASN.1 specification may be useful to some readers of this part of ISO/IEC 19785. It is not intended to provide an alternative specification of the encodings of this patron format.

```
SELF-IDENTIFYING-TAG-ORIENTED-SIMPLE-BIR-PATRON-FORMAT
{iso(1) standard(0) cbeff(19785) modules(0) self-id-tagged-simple(12)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

  IMPORTS
   BDBFormat, BiometricType, BiometricSubtype, ProcessedLevel, Product, Purpose,
   Quality, ValidityPeriod, Version, RegistryID, EncryptionOptions,IntegrityOptions,
   Creator, BIRIndex, Payload, BDBIndex, ChallengeResponse
  FROM CBEFF-DATA-ELEMENTS
     -- The following data elements are defined by the BDB format provider
     -- (e.g. ISO/IEC 19794-x)
     CardholderChallengeType, StandardBDBType, ProprietaryBDBType
  FROM BDB-DATA-ELEMENTS
     -- The following data elements are defined by the BDB format provider
     -- (e.g. ISO/IEC 19785-4)
     SBType
  FROM SB-DATA-ELEMENTS;

  SBIR ::= [APPLICATION 96] IMPLICIT SEQUENCE {
    birSelfId            [6]   RegistryID,
    cbeffVersion         [28]  Version,
    patronHeaderVersion  [0]   Version,
    birInfo              [7]   SET
      {
      birIntegrity             [0]   IntegrityOptions,
      birCreationDate          [24]  DATE-TIME OPTIONAL,
      birValidityPeriod        [23]  ValidityPeriod OPTIONAL,
      birCreator               [4]   Creator OPTIONAL,
      birIndex                 [16]  BIRIndex OPTIONAL,
      birPayload               [1]   Payload OPTIONAL,
      birPointer               [2]   OCTET STRING (SIZE(0..65535)) OPTIONAL
      },
    bdbInfo              [3]   SET
      {
      bdbFormat                [7]   BDBFormat,
      bdbEncryption            [0]   EncryptionOptions,
      bdbBiometric             [1]   SEQUENCE
        {
        bdbBiometricType           [1] BiometricType,
        bdbBiometricSubtype        [2] BiometricSubtype OPTIONAL
        } OPTIONAL,
      bdbProcessedLevel        [21]  ProcessedLevel OPTIONAL,
      bdbPurpose               [22]  Purpose OPTIONAL,
      bdbCreationDate          [3]   DATE-TIME OPTIONAL,
      bdbValidityPeriod        [5]   ValidityPeriod OPTIONAL,
      bdbChallengeResponse     [19]  ChallengeResponse OPTIONAL,
      bdbIndex                 [20]  BDBIndex OPTIONAL,
      bdbProduct               [6]   Product OPTIONAL,
      bdbCaptureDevice         [2]   RegistryID OPTIONAL,
      bdbFeatureExtractionAlg  [4]   RegistryID OPTIONAL,
      bdbComparisonAlg         [9]   RegistryID OPTIONAL,
      bdbCompressionAlg        [10]  RegistryID OPTIONAL,
      bdbQuality               [23]  ConstructedQuality OPTIONAL,
      bdbPADMechanism          [29]  RegistryID OPTIONAL
      },
```

```
    sbInfo              [4]   SET
      {
      sbFormat                [0]   RegistryID
      } OPTIONAL,
    bdbData                         [APPLICATION 46] SET
      {
      challengeCardholder   [0]             CardholderChallengeType OPTIONAL,
                                            -- defined in BDB-DATA-ELEMENTS
      standardBDB           [1]             StandardBDBType OPTIONAL,
                                            -- defined in BDB-DATA-ELEMENTS
      proprietaryBDB        [2]             ProprietaryBDBType OPTIONAL,
                                            -- defined in BDB-DATA-ELEMENTS
      referenceQualifier    [3]             OCTET STRING (SIZE(1)) OPTIONAL,
      biometricData         [APPLICATION 46] OCTET STRING OPTIONAL
      } OPTIONAL
    sb                  [18]  OCTET STRING (SIZE(0..4294967295)) OPTIONAL
    }

END
```

Imported elements shall be defined prior to compiling the ASN.1 specification. For illustrative purposes the following example is provided. Implementers shall refer to the relevant standards where the elements are defined.

```
/*
-- These definitions shall be provided by the BDB format provider.
-- In case of being primitive data, these data elements shall be of type
-- OCTET STRING.
-- In case of constructed data, this/these data elements' complete
-- definiton should be provided with the data element names given here.
--
-- THE DATA DEFINED BELOW IS AN EXAMPLE FOR ILLUSTRATIVE PURPOSES ONLY
--


BDB-DATA-ELEMENTS
{iso(1) standard(0) 19794(19794) 2014(2014)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

  CardholderChallengeType ::= OCTET STRING -- if primitive (comment line
                                           -- if not used)
/*
  CardholderChallengeType ::= SET    -- if constructed, to be defined by the
                                     -- BDB data format provider (comment
                                     -- block if not used)
```

```
    {
    tagAllocationAuthority [APPLICATION 24] CHOICE
      {
      oid                                    OBJECT IDENTIFIER,
      country                [APPLICATION 1]  OCTET STRING,
      issuerID               [APPLICATION 2]  OCTET STRING,
      application            [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018},

    biometricData       [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/
  StandardBDBType ::= OCTET STRING -- if primitive (comment line
                                   -- if not used)
/*

  StandardBDBType ::= SET   -- if constructed, to be defined by the
                            -- BDB data format provider (comment block
                            -- if not used)
    {
    tagAllocationAuthority [APPLICATION 24] CHOICE
      {
      oid                                    OBJECT IDENTIFIER,
      country                [APPLICATION 1]  OCTET STRING,
      issuerID               [APPLICATION 2]  OCTET STRING,
      application            [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018},

    biometricData       [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/
```

```
  ProprietaryBDBType ::= OCTET STRING    -- if primitive (comment line
                                         -- if not used)
/*
  ProprietaryBDBType ::= SET     -- if constructed, to be defined by the
                                 -- BDB data format provider (comment block
                                 -- if not used)
    {
    tagAllocationAuthority [APPLICATION 24] CHOICE
      {
      oid                                  OBJECT IDENTIFIER,
      country              [APPLICATION 1]  OCTET STRING,
      issuerID             [APPLICATION 2]  OCTET STRING,
      application          [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018},

    biometricData           [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/

END
*/
```

```
/*
-- These definitions shall be provided by the SB format provider.
-- In case of being primitive data, these data elements shall be of type
-- OCTET STRING.
-- In case of constructed data, this/these data elements' complete
-- definition/s should be provided with the data element type names given here.
--
-- THE DATA DEFINED BELOW IS AN EXAMPLE FOR ILLUSTRATIVE PURPOSES ONLY
--
SB-DATA-ELEMENTS
{iso(1) standard(0) cbeff(19785) part(3)} -- or whatever OID is registered
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN
```

```
  SBType ::= OCTET STRING (SIZE(0..4294967295)) -- if primitive (comment line
                                                -- if not used)
/*
  SBType ::= SET                -- if constructed, to be defined by the
                                -- SB data format provider (comment block
                                -- if not used)
    {
    tagAllocationAuthority   [APPLICATION 24] CHOICE
      {
      oid                                     OBJECT IDENTIFIER,
      country                  [APPLICATION 1]  OCTET STRING,
      issuerID                 [APPLICATION 2]  OCTET STRING,
      application              [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018}, -- or whatever OID is registered

    sbData                   [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/

END
*/
```


17 Patron format specification: Self-identifying, Tag-oriented Complex BIR

17.1 Patron

ISO/IEC JTC 1/SC 37.

This patron format was developed from the Complex BIR defined for its use in Object-Oriented BioAPI (see ISO/IEC 30106-1:2016, 6.3), but it is specified and generalized here for use in any other environment, even outside the scope of ISO/IEC 30106-1. The patron format is defined to be coded in any tagged data format. Specifications for the encoding in popular tagged formats are given in Annex C.

17.2 Patron format owner

257 (0101_{Hex}). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

17.3 Patron format name

```
ISO/IEC JTC 1/SC 37 self-identifying-tag-oriented-complex-bir patron format
```

17.4 Patron format type

`13 (000DHex)`. This has been registered in accordance with ISO/IEC 19785-2.

17.5 ASN.1 OID for this patron format

```
{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257)
patron-format(1) self-id-tagged-complex(13)}
```

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.13 </OBJECT_IDENTIFIER>
```

17.6 Domain of use

This clause provides a definition of a patron format that may be of general utility to applications that need to carry more than one BDB (of either the same or different patron formats) in a single complex BIR structure, with explicit identification of each of the patron format(s) being used. This patron format also includes optional support for the Multiple CBEFF BIR structure.

The Self-Identifying Tag-Oriented Simple BIR patron format, defined in Clause 16, is referenced for the coding of child BIRs capable of containing BDBs.

17.7 Version identifier

This patron format specification has a version identified with major 1 and minor 0.

17.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0).

17.9 General

17.9.1 Patron Format Structure Type: Complex CBEFF BIR with self-identification and optional support for Multiple CBEFF BIR.

This patron format includes optional CBEFF data elements needed to implement multiple CBEFF BIR structures. Specifically, this patron format includes an optional CBEFF_BIR_pointer field which may be used to point to subsequent BIRs.

When subsequent BIRs are included in this patron format, a linear structure shall be used such that each subsequent BIR points to the next, and the sequence is terminated by a BIR where the CBEFF_BIR_pointer is not included or has a value of 0. This is also known as the end-to-end structure and is depicted in ISO/IEC 19785-1:2020 Figure 3. Since this patron format is tag-oriented and does not support memory location pointers, the CBEFF_BIR_pointer will indicate the location of the next BIR using a URL or file path.

However, if the CBEFF_BIR_pointer is not included or has a value of 0, this patron format represents a Complex CBEFF BIR.

NOTE The inclusion of the optional CBEFF_BIR_pointer field is intended to provide flexibility in the use of this patron format. If a Complex CBEFF BIR meets implementation needs, then the Multiple CBEFF BIR concept can be ingored.

17.9.2 This patron format supports some of the mandatory and optional data elements specified in ISO/IEC 19785-1. For a full list of allowed fields, refer to Table 22. It supports complex BIR structures (and optional Multiple BIR structures) where each intermediate node or leaf of the structure is itself a BIR (called a "child BIR"), represented by a self-identifying Simple BIR structure defined in Clause 16 or another self-identifying Complex BIR as defined in Clause 17 (this clause).

17.9.3 The patron format of each child BIR is explicitly identified in its own BIR structure, in its mandatory pair: patron format owner / patron format type.

17.9.4 Some fields in this patron format are optional. In order to allow future backwards interoperability, this patron format shall be coded with a tagged data format (e.g. TLV, XML or JSON). Those optional parameters, if absent, will not include their relevant tags in the data structure.

Tagged data formats can be parsed and processed without using the overall BIR length value. Therefore, this patron format does not include a BIR length field.

17.9.5 All character strings and octet strings are preceded by a length prefix, which can be one, two, or four octets long, as specified for each field. The tagged data format chosen will adapt such a length prefix according to the best practices within such data format.

EXAMPLE In TLV data coding, the length is be part of the L sub-field, and not the V sub-field.

17.9.6 All integer values, including lengths, are encoded in big-endian order.

17.9.7 Dates and date intervals are encoded as character strings in a way conforming to the ISO 8601 series.

17.9.8 An instance of a BIR contains one or more BIR children, but never contains a BDB from itself.

17.10 Specification

An instance of a BIR shall contain the fields specified below.

The structure of the embedded Child BIR shall follow the same specification as that of the Self-identifying Tag-oriented Simple BIR (Clause 16, i.e. patron format type 12), or it may be another instance of the patron format specified in this clause.

Table 22 — Specifications for self-identifying, tag-oriented complex BIR patron format

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and encodings[b]
The following fields shall occur at most once			
SBIR	ComplexSBIR	container, mandatory	Container of the whole data format
The following fields shall be within SBIR container and shall occur at most once			
CBEFF_BIR_self_id_owner	BirSelfIdOrganization	2, mandatory	0101_{Hex}
CBEFF_BIR_self_id_type	BirSelfIdId	2, mandatory	$000D_{Hex}$
CBEFF_version	cbeffVersion	2, mandatory	Major '4' and Minor '0': 0400_{Hex}
CBEFF_patron_header_version	patronHeaderVersion	2, mandatory	Major '1' and Minor '0': 0100_{Hex}
CBEFF_subheader_count	numChildren	1, mandatory	1..255
The following are fields related to the BIR	birInfo	container, mandatory	
CBEFF_BIR_integrity_options	birIntegrity	1, mandatory	NO INTEGRITY: 0 INTEGRITY: 1
CBEFF_BIR_creation_date	birCreationDate	8..15, optional	Variable-length ASCII character string. The string shall represent a date (or date and a time of the day)[a].
CBEFF_BIR_validity_period	birValidityPeriod	17..31, optional	Variable-length ASCII character string. The string shall represent an interval of two dates (or date and time of the day)[b].
CBEFF_BIR_creator	birCreator	0..65535, optional	Variable-length ISO/IEC 10646 character string, encoded in UTF-8.
CBEFF_BIR_index	birIndex	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.
CBEFF_BIR_payload	birPayload	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.
CBEFF_BIR_pointer	birPointer	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.
The following are conditional fields related to the SB, mandatory only if a SB is included in the BIR	sbInfo	container, conditional	
CBEFF_SB_format_owner	sbFormatOrganization	2, conditional	1..65535
CBEFF_SB_format_type	sbFormatId	2, conditional	1..65535
The following field shall occur as many times as specified in the field numChildren (1..255)			
not a standard CBEFF data element	childBir	4 + 0..4294967295, mandatory	Coded either as a complex BIR (i.e. this patron format) or as a simple BIR (child BIR), in whatever patron that identifies at the beginning its patron format owner and type and its length (e.g. the SimpleBIR patron format defined in Clause 16)
The following field shall occur at most once			
SB	sb	4 + 0..4294967295, optional	Variable-length octet string, preceded by a 32-bit integer field containing the length (octets).

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits. Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR conforming to this patron format consists of an SBH with numChildren value greater than 0 and does not directly contain a BDB or SB. BDB and SB information may be contained in child BIRs conforming to the self-identifying tag-oriented simple BIR format defined in Clause 16.

17.11 Illustrative example

Additional illustrative examples of tag-oriented patron formats are included in C.6.

17.12 ASN.1 definition

The following ASN.1 specification provides an abstract description of the patron format and can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en. This ASN.1 specification may be useful to some readers of this part of ISO/IEC 19785. It is not intended to provide an alternative specification of the encodings of this patron format.

```
SELF-IDENTIFYING-TAG-ORIENTED-COMPLEX-BIR-PATRON-FORMAT
{iso(1) standard(0) cbeff(19785) modules(0) self-id-tagged-complex(13)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

  IMPORTS
    ValidityPeriod, Version, RegistryID, IntegrityOptions, Creator,
    BIRIndex, Payload
  FROM CBEFF-DATA-ELEMENTS
    SBIR
  FROM SELF-IDENTIFYING-TAG-ORIENTED-SIMPLE-BIR-PATRON-FORMAT
    -- The following data elements are defined by the BDB format provider
    -- (e.g. ISO/IEC 19785-4)
    SBType
  FROM SB-DATA-ELEMENTS;


  ComplexSBIR ::=   [APPLICATION 97]   IMPLICIT SEQUENCE {
    birSelfId           [6]  RegistryID,
    cbeffVersion        [28] Version,
    patronHeaderVersion [0]  Version,
    numChildren         [5]  INTEGER (1..255),
    birInfo             [7]  SET
    {
      birIntegrity           [0]  IntegrityOptions,
      birCreationDate        [24] DATE-TIME OPTIONAL,
      birValidityPeriod      [27] ValidityPeriod OPTIONAL,
      birCreator             [4]  Creator OPTIONAL,
      birIndex               [16] BIRIndex OPTIONAL,
      birPayload             [1] Payload OPTIONAL,
      birPointer             [2] OCTET STRING (SIZE(0..65535)) OPTIONAL
    },
    sbInfo              [4]   SET
    {
      sbFormat               [0]  RegistryID
    } OPTIONAL,
    children            [1]   SEQUENCE (SIZE(1..255)) OF
      childBir CHOICE
      {
        simpleBIR      [APPLICATION 96] SBIR,        -- As per clause 16
        complexBIR     [APPLICATION 97] ComplexSBIR  -- As per clause 17
      },
    sb                  [18] SBType OPTIONAL
  }

END
```

Imported elements shall be defined prior to compiling the ASN.1 specification. For illustrative purposes the following example is provided. Implementers shall refer to the relevant standards where the elements are defined.

```
/*
-- These definitions shall be provided by the SB format provider.
-- In case of being primitive data, these data elements shall be of type
-- OCTET STRING.
-- In case of constructed data, this/these data elements' complete
-- definition/s should be provided with the data element type names given here.
--
-- THE DATA DEFINED BELOW IS AN EXAMPLE FOR ILLUSTRATIVE PURPOSES ONLY
--
SB-DATA-ELEMENTS
{iso(1) standard(0) cbeff(19785) part(3)} -- or whatever OID is registered
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

  SBType ::= OCTET STRING (SIZE(0..4294967295)) -- if primitive (comment line
                                                -- if not used)
/*
  SBType ::= SET               -- if constructed, to be defined by the
                               -- SB data format provider (comment block
                               -- if not used)
    {
    tagAllocationAuthority  [APPLICATION 24] CHOICE
      {
      oid                                    OBJECT IDENTIFIER,
      country                [APPLICATION 1]  OCTET STRING,
      issuerID               [APPLICATION 2]  OCTET STRING,
      application            [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018}, -- or whatever OID is registered

    sbData                  [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/

END
*/
```

18 Patron format specification: PAD patron format (Linking BDB with PAD Sample

18.1 Patron

ISO/IEC JTC 1/SC 37.

18.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

18.3 Patron format name

ISO/IEC JTC 1/SC 37 PAD patron format.

18.4 Patron format type

14 (000EHex). This has been registered in accordance with ISO/IEC 19785-2.

18.5 ASN.1 OID for this patron format

{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257) patron-format(1) pad(14)}

or, in XML value notation,

<OBJECT_IDENTIFIER>1.1.19785.0.257.1.14</OBJECT_IDENTIFIER>

18.6 Domain of use

This clause provides a definition of a patron format that links PAD information with a relevant BDB. In order to perform such a functionality, a Self-identifying Tag-oriented Complex BIR patron (Clause 17) is taken as a basis, implementing a self-identifying CBEFF BIR structure. This complex BIR will include at least one simple BIR with a BDB, and another simple BIR containing the PAD information as specified in ISO/IEC 30107-2. The two simple BIRs shall use the Self-identifying Tag-oriented Simple BIR patron format (Clause 16).

18.7 Version identifier

This patron format specification has a version identifier of major 1 and minor 0.

18.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0).

18.9 General

18.9.1 Patron Format Structure Type: Complex CBEFF BIR with self-identification.

18.9.2 This patron is based in the byte and tag-oriented complex BIR patron format (Clause 17).

18.9.3 In order to allow future backwards interoperability, this patron format shall be coded with a tagged data format (e.g. TLV, XML or JSON). Optional parameters, if absent, will not include their relevant tags in the data structure.

Tagged data formats can be parsed and processed without using the overall BIR length value. Therefore, this patron format does not include a BIR length field.

18.9.4 All character strings and octet strings are preceded by a length prefix, which can be one, two, or four octets long, as specified for each field. The tagged data format chosen will adapt such a length prefix according to the best practices within such data format.

EXAMPLE In TLV data coding, the length will be part of the L sub-field, and not into the V sub-field.

18.9.5 All integer values, including lengths, are encoded in big-endian order.

18.9.6 Dates and date intervals are encoded as character strings in a way conforming to the ISO 8601 series.

18.9.7 An instance of a BIR shall contain at least 2 child BIRs, one with the biometric probe data, and another one with the PAD data.

18.10 Specification

An instance of a BIR shall contain the fields specified below.

The structure of the embedded Child BIR shall follow the same specification as that of the Self-identifying Tag-oriented Simple BIR (Clause 16, i.e. patron format type 13).

Table 23 — Specifications for PAD bir

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and Encodings[b]
The following fields shall occur at most once			
SBIR	PadSBIR	container, mandatory	Container of the whole data format
The following fields shall be within ComplexBIR container and shall occur at most once			
CBEFF_BIR_self_id_owner	BirSelfIdOrganization	2, mandatory	0101_{Hex}
CBEFF_BIR_self_id_type	BirSelfIdId	2, mandatory	$000D_{Hex}$
CBEFF_version	cbeffVersion	2, mandatory	Major '4' and Minor '0': 0400_{Hex}
CBEFF_patron_header_version	patronHeaderVersion	2, mandatory	Major '1' and Minor '0': 0100_{Hex}
CBEFF_subheader_count	numChildren	1, mandatory	2..255
The following are fields related to the BIR	birInfo	container, mandatory	
CBEFF_BIR_integrity_options	birIntegrity	1, mandatory	NO INTEGRITY: 0 INTEGRITY: 1
CBEFF_BIR_creation_date	birCreationDate	8..15, optional	Variable-length ASCII character string. The string shall represent a date (or date and a time of the day)[a].
CBEFF_BIR_validity_period	birValidityPeriod	17..31, optional	Variable-length ASCII character string. The string shall represent an interval of two dates (or date and time of the day)[b].
CBEFF_BIR_creator	birCreator	0..65535, optional	Variable-length ISO/IEC 10646 character string, encoded in UTF-8.
CBEFF_BIR_index	birIndex	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.
CBEFF_BIR_payload	birPayload	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.
CBEFF_BIR_pointer	birPointer	0..65535, optional	Variable-length octet string. Shall not inherit its value from any other level BIR.

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits. Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR conforming to this patron format consists of an SBH with numChildren value greater than or equal to 2 and does not directly contain a BDB or SB. At least one child BIR must contain a BDB and another child BIR must contain PAD information.

Table 23 *(continued)*

CBEFF data element name	Field name	Length and optionality[a]	Abstract values and Encodings[b]
The following are conditional fields related to the SB, mandatory only if a SB is included in the BIR	sbInfo	container, conditional	
CBEFF_SB_format_owner	sbFormatOrganization	2, conditional	1..65535
CBEFF_SB_format_type	sbFormatId	2, conditional	1..65535
The following field shall occur a number of times between 1 and numChildren-1			
Not a standard CBEFF data element	biometricdataBir	15..4294967295, mandatory	Coded either as a complex BIR (i.e., Clause 17) or as a simple BIR (i.e., Clause 16), The content of such BIRs shall be biometric samples. (The minimum biometricdataBir length is equal to the minimum length of mandatory fields for a Simple BIR in Clause 16)
The following field shall occur the number of times resulting of substracting the number of biometricdataBirs from numChildren			
Not a standard CBEFF data element	padBir	0..4294967295, mandatory	Coded either as a complex BIR (i.e., clause 17) or as a simple BIR (i.e., clause 16), The content of such BIRs shall be presentation attack detection (i.e., ISO/IEC 30107-2).
The following field shall occur at most once			
SB	sb	0..4294967295, optional	Variable-length octet string.

[a] The date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. Examples: 20050103, 20050106T11, 20050106T1230, and 20050106T145504.

[b] Each date shall be represented in the ISO 8601 series basic format YYYYMMDDTHHMMSS, where the last 2, the last 4, or the last 7 characters may be omitted. The two dates shall be separated by a SOLIDUS ("/") character, and shall have the same number of digits. Examples: 20050103/20060103, 20050106T11/20050306T11, and 20050106T113300/20050306T113259.

[c] A BIR conforming to this patron format consists of an SBH with numChildren value greater than or equal to 2 and does not directly contain a BDB or SB. At least one child BIR must contain a BDB and another child BIR must contain PAD information.

18.11 Illustrative example

Table 24 — Byte-oriented complex BIR with byte-oriented simple BIR children

1. sBir = ComplexSBIR
2. BirSelfIdOrganization = 257 (ISO/IEC JTC 1/SC 37)
3. BirSelfIdId = 13
4. cbeffVersion = 4.0
5. patronHeaderVersion = 1.0
6. numChildren = 2
7. birInfo
8. ► *(denotes the beginning of the BIR Info)*
9. birIntegrity = INTEGRITY *(integrity is applied to the entire complex BIR via the SB on line 50)*
10. sbInfo
11. ► *(denotes the beginning of the SB Info)*
12. sbFormatOrganization = *a security vendor*
13. sbFormatId = *that vendor's security block format (see the final SB on line 50)*
14. ► *(denotes the beginning of child BIR 1 of 2 containing the BDB)*
15. sBir = SBIR
16. BirSelfIdOrganization = SC 37
17. BirSelfIdId = 12
18. cbeffVersion = 4.0
19. patronHeaderVersion = 1.1
20. numChildren=0
21. birInfo
22. ► *(denotes the beginning of the BIR Info)*
23. birIntegrity = NO INTEGRITY
24. bdbInfo
25. ► *(denotes the beginning of the BDB Info)*
26. bdbFormatOrganization = SC 37
27. bdbFormatId = *a standardized BDB format*
28. bdbEncryption = NO ENCRYPTION
29. bdbBiometricType = FINGER
30. bdbBiometricSubtype = LEFT INDEX FINGER
31. bdb
32. ► *(denotes the beginning of child BIR 2 of 2 containing PAD data)*
33. sBir = SBIR
34. BirSelfIdOrganization = SC 37
35. BirSelfIdId = 12
36. cbeffVersion = 4.0
37. patronHeaderVersion = 1.1
38. numChildren = 0
39. birInfo

Table 24 *(continued)*

40. ► *(denotes the beginning of the BIR Info)*
41. birIntegrity = NO INTEGRITY
42. bdbInfo
43. ► *(denotes the beginning of the BDB Info)*
44. bdbFormatOrganization = SC37
45. bdbFormatId = PAD information (according to *ISO/IEC 30107-2*)
46. bdbEncryption = NO ENCRYPTION
47. bdbBiometricType = FINGER *(must be same as line 29)*
48. bdbBiometricSubtype = LEFT INDEX FINGER *(must be same as line 30)*
49. bdb
50. sb *(see SB format identifier in root header on lines 12-13)*

18.12 ASN.1 definition

The following ASN.1 specification provides an abstract description of the patron format and can be retrieved from https://standards.iso.org/iso-iec/19785/-3/ed-3/en. This ASN.1 specification may be useful to some readers of this part of ISO/IEC 19785. It is not intended to provide an alternative specification of the encodings of this patron format.

```
BDB-AND-PAD-BIR-PATRON-FORMAT
{iso(1) standard(0) cbeff(19785) modules(0) padBIR(14)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

  IMPORTS
    IntegrityOptions, ValidityPeriod, Version, RegistryID
  FROM CBEFF-DATA-ELEMENTS
    SBIR
  FROM SELF-IDENTIFYING-TAG-ORIENTED-SIMPLE-BIR-PATRON-FORMAT
    ComplexSBIR
  FROM SELF-IDENTIFYING-TAG-ORIENTED-COMPLEX-BIR-PATRON-FORMAT
     -- The following data elements are defined by the BDB format provider
     -- (e.g. ISO/IEC 19785-4)
     SBType
  FROM SB-DATA-ELEMENTS;

  PadSBIR ::= [APPLICATION 97] IMPLICIT SEQUENCE {
    birSelfId             [6]  RegistryID,
    cbeffVersion          [28] Version,
    patronHeaderVersion   [0]  Version,
    numChildren           [5]  INTEGER (2..255),
    birInfo               [7]  SET {
      birIntegrity             [0]  IntegrityOptions,
      birCreationDate          [24] DATE-TIME OPTIONAL,
      birValidityPeriod        [27] ValidityPeriod OPTIONAL,
      birCreator               [4]  OCTET STRING (SIZE(0..65535)) OPTIONAL,
      birIndex                 [16] OCTET STRING (SIZE(0..65535)) OPTIONAL,
      birPayload               [1]  OCTET STRING (SIZE(0..65535)) OPTIONAL,
      birPointer               [2]  OCTET STRING (SIZE(0..65535)) OPTIONAL
      },
    sbInfo                [4] SET {
      sbFormat                 [0] RegistryID
      } OPTIONAL,
    biometricdataBir      [1] SEQUENCE (SIZE(1..numChildren-1)) OF
      childBir CHOICE {
        simpleBIR          [APPLICATION 96] SBIR,       -- As per clause 16
        complexBIR         [APPLICATION 97] ComplexSBIR -- As per clause 17
        },
    padBir                [9] SEQUENCE (SIZE(1..numChildren-1)) OF
      childBir CHOICE {
```

```
            simpleBIR          [APPLICATION 96] SBIR,       -- As per clause 16
            complexBIR         [APPLICATION 97] ComplexSBIR -- As per clause 17
            },
        sb                     [18] SBType OPTIONAL
    }

END
```

Imported elements shall be defined prior to compiling the ASN.1 specification. For illustrative purposes the following example is provided. Implementers shall refer to the relevant standards where the elements are defined.

```
/*
-- These definitions shall be provided by the SB format provider.
-- In case of being primitive data, these data elements shall be of type
-- OCTET STRING.
-- In case of constructed data, this/these data elements' complete
-- definition/s should be provided with the data element type names given
   here.
--
-- THE DATA DEFINED BELOW IS AN EXAMPLE FOR ILLUSTRATIVE PURPOSES ONLY
--
SB-DATA-ELEMENTS
{iso(1) standard(0) cbeff(19785) part(3)} -- or whatever OID is registered
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

  SBType ::= OCTET STRING (SIZE(0..4294967295)) -- if primitive (comment line
                                                -- if not used)
/*
  SBType ::= SET              -- if constructed, to be defined by the
                              -- SB data format provider (comment block
                              -- if not used)
  {
  tagAllocationAuthority   [APPLICATION 24] CHOICE
    {
    oid                                     OBJECT IDENTIFIER,
    country                 [APPLICATION 1]  OCTET STRING,
    issuerID                [APPLICATION 2]  OCTET STRING,
    application              [APPLICATION 15] OCTET STRING
    }DEFAULT oid: {1 0 24787 2018}, -- or whatever OID is registered

  sbData                 [APPLICATION 16] SET
    {
    field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
    field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
    field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
    }
  }
*/

END
*/
```

19 Patron format specification: TLV-encoded patron format for ICCs and other tokens (with explicit tag allocation authority)

19.1 Patron

ISO/IEC JTC 1/SC 37.

19.2 Patron format owner

257 (0101Hex). The BRA has allocated this identifier for ISO/IEC JTC 1/SC 37.

19.3 Patron format name

ISO/IEC JTC 1/SC 37 TLV-encoded patron format for ICCs with explicit tag allocation authority

19.4 Patron format type

15 (000FHex). This has been registered in accordance with ISO/IEC 19785-2.

19.5 ASN.1 OID for this patron format

```
{iso(1) registration-authority(1) cbeff(19785) biometric-organization(0) jtc1-sc37(257)
patron-format(1) tlv-encoded-explicit(15)}
```

or, in XML value notation,

```
<OBJECT_IDENTIFIER>1.1.19785.0.257.1.15</OBJECT_IDENTIFIER>
```

19.6 Domain of Use

The domain of use is applications complying with ISO/IEC 7816-11 with explicit tag allocation authority. This patron format is applicable for both on-card and off-card biometric comparison and should be used for new applications using ICCs. Older version applications may decide to be compliant to the patron format defined in Clause 11.

For any other kind of application that would like to use TLV coding, patron formats defined in Clauses 16 and 17 shall be used.

19.7 Version identifier

This patron format specification has a version identifier of (major 1, minor 0).

19.8 CBEFF version

This specification conforms to CBEFF version (major 4, minor 0).

19.9 General

19.9.1 Patron Format Structure Type: Complex CBEFF BIR without self-identification

NOTE This patron format structure type is considered to be Complex CBEFF BIR, because multiple Simple BIRs (Biometric Information Template DOs) that encapsulate BDB information (BDB Reference Data, BDB Reference Data Template DOs) with their own SBHs (Biometric Header Template DOs) can be included in the Biometric Information Template Group DO.

19.9.2 This clause specifies in 19.10:

a) the CBEFF patron format named "ISO/IEC JTC 1/SC 37 TLV-encoded patron format for ICCs with explicit tag allocation authority", and

b) a TLV encoding for CBEFF data elements and patron-defined data elements (together with distinct TLV tags) that can be used by any TLV-encoded application that needs to make reference to such data elements.

NOTE The particular form of TLV encoding specified in this clause is that provided by the ASN.1 Distinguished Encoding Rules (see ISO/IEC 8825-1). See ISO/IEC 7816-4 for additional BER-TLV data objects encoding provisions applied to ICCs.

19.9.3 ISO/IEC 7816-4 and ISO/IEC 7816-11 define the necessary APDU exchanges to support:

a) On-card biometric comparison

1) Retrieval of biometric information from a smartcard or other token for valid biometric probe construction prior to biometric verification process on ICC.

2) Commands for performing a biometric probe verification on a smartcard.

3) Commands for biometric reference and biometric information enrolment (recording of information on a smartcard or other token).

4) Security mechanisms for establishing a trusted channel between a smartcard and an external system.

b) Off-card biometric comparison

1) Commands for retrieval of the biometric reference data from a smartcard to be used in an external service system.

2) Security mechanisms for the protection of the biometric reference data.

19.9.4 This version of the CBEFF patron format named "ISO/IEC JTC 1/SC 37 TLV-encoded patron format for ICCs with explicit tag allocation authority" is formally defined as the ASN.1 type **`BiometricInformationTemplateExplicit`** (see 19.10) encoded using the ASN.1 Distinguished Encoding Rules (see ISO/IEC 8825-1). A diagrammatic specification of this patron format is provided in 19.11.

NOTE The term Biometric Information Template is used in ISO/IEC 7816-11. This term corresponds to a (possibly partial) Biometric Information Record and is used in this clause where the relation with ISO/IEC 7816-11 is important.

19.9.5 The components of the ASN.1 type **`BiometricHeaderTemplateExplicit`** (with their context-specific tags, see 19.10), encoded using the ASN.1 Basic Encoding Rules are also defined as Data Objects for use in ISO/IEC 7816-11.

NOTE The term Biometric Header Template is used in ISO/IEC 7816-11, together with the abbreviation BHT. This term and the abbreviated term are synonymous with Standard Biometric Header and SBH, and are used in this clause where the relation with ISO/IEC 7816-11 is important.

19.9.6 The term Biometric Information Data Object is used in ISO/IEC 7816-11 to refer to the TLV-encoding of a specific ASN.1 type. Table 25 records the Biometric Information Data Objects that are defined in 19.10 for use in ISO/IEC 7816-11.

NOTE As ISO/IEC 7816-11 normally describes tags by their hex values, these are included here.

Table 25 — Biometric information DOs specified by the compatible tag allocation authority with the Biometric Header Template Data Objects defined in 19.10

Biometric information Data Object	ASN.1 type or component name (see 19.10)	Tag
Biometric Information Template	**BiometricInformationTemplateExplicit**	$7F60_{Hex}$
Biometric Information Template Group	**GroupBIT**	$7F61_{Hex}$
BDB Data Template	**bdbReferenceData**	$5F2E_{Hex}$
BDB Data Template Template	**bdbReferenceDataTemplate**	$7F2E_{Hex}$
BIR Payload (primitive / constructed)	**birPayLoad**	53_{Hex} / 73_{Hex}
For use only in Biometric Data Template:		
BDB Data Template with Standardised Format (Primitive)	**standardBDBReferenceData**	81_{Hex}
BDB Data Template with Standardised Format (Constructed)	**standardBDBReferenceDataTemplate**	$A1_{Hex}$
BDB Data Template with Proprietary Format (Primitive)	**proprietaryBDBReferenceData**	82_{Hex}
BDB Data Template with Proprietary Format (Constructed)	**proprietaryBDBReferenceDataTemplate**	$A2_{Hex}$
For use only in Biometric Information Template:		
Biometric Header Template	**biometricHeaderTemplateExplicit**	$A1_{Hex}$
Algorithm Reference	**algorithmReference**	80_{Hex}
Data Qualifier	**referenceDataQualifier**	83_{Hex}
For use only in Biometric Header Template (i.e. within DO'A1'):		
Tag Allocation Authority	**tagAllocationAuthority**	78_{Hex}
Biometric Information Data Object	**biometricInformationDO**	70_{Hex}
For use only in Biometric Information DOs template (i.e. within DO'70'):		
Patron Header Version (as primitive)	**patronHeaderVersion**	80_{Hex}
BDB Biometric Type	**bdbBiometricType**	81_{Hex}
BDB Biometric Subtype	**bdbBiometricSubType**	82_{Hex}
BDB Creation Date	**bdbCreationDate**	83_{Hex}
BIR Creator	**birCreator**	84_{Hex}
BDB Validity Period (as primitive)	**bdbValidityPeriod**	85_{Hex}
BDB PID (as primitive)	**bdbPID**	86_{Hex}
BDB Format Owner (as constructed)	**bdbFormat**	$A7_{Hex}$
BDB Format Owner (as primitive)	**bdbFormatOwner**	87_{Hex}
BDB Format Type (as primitive)	**bdbFormatType**	88_{Hex}
BIR Index	**birIndex**	90_{Hex}
Biometric Comparison Parameters (primitive / constructed)	**comparisonParameters**	91_{Hex} / $B1_{Hex}$
Biometric Functionality Information (primitive / constructed)	**functionalityInformation**	92_{Hex} / $B2_{Hex}$

19.9.7 CBEFF-compliance requires the NO VALUE AVAILABLE abstract value to be supported by this patron format for CBEFF data elements that are not fully supported. To meet this requirement, tags are assigned and reserved by this document in Table 26 for Data Objects appearing in the Biometric Header Template. These tag values shall not be assigned by any other tag authority to Data Objects that can appear in the Biometric Header Template. These Data Objects shall never appear in encodings of this version of this patron format, and their omission represents the NO VALUE AVAILABLE abstract value of the corresponding CBEFF data element.

NOTE Subsequent versions of this patron format can define these Data Objects to support the corresponding CBEFF data element.

Table 26 — Reserved tag values in DO'70'

ASN.1 tag value	Tag value	Corresponding CBEFF data elements
[5]	$A5_{Hex}$	CBEFF_BDB_validity_period (as constructed)
[9]	$A9_{Hex}$	CBEFF_BDB_comparison_algorithm
[10]	AA_{Hex}	CBEFF_BDB_compression_algorithm
[11]	AB_{Hex}	CBEFF_BDB_quality_algorithm
[12]	AC_{Hex}	CBEFF_BDB_capture_device
[13]	AD_{Hex}	CBEFF_BDB_PAD_mechanisms (as constructed)
[14]	AE_{Hex}	CBEFF_BDB_feature_extraction_algorithm
[19]	93_{Hex}	CBEFF_BDB_challenge_response
[20]	94_{Hex}	CBEFF_BDB_index
[21]	95_{Hex}	CBEFF_BDB_processed_level
[22]	96_{Hex}	CBEFF_BDB_purpose
[23]	97_{Hex} / $B7_{Hex}$	CBEFF_BDB_quality
[24]	98_{Hex}	CBEFF_BIR_creation_date
[25]	99_{Hex}	CBEFF_BIR_patron_format_owner
[26]	$9A_{Hex}$	CBEFF_BIR_patron_format_type
[27]	$9B_{Hex}$	CBEFF_BIR_validity_period (as primitive)
[28]	$9C_{Hex}$	CBEFF_version (as primitive)

19.10 ASN.1 specification

The ASN.1 module given in 19.10 can be retrieved from http://standards.iso.org/iso-iec/19785/-3/ed-3/en.

```
CBEFF-SMARTCARD-BIDO-EXPLICIT
-- The abbreviation BIDO is used for Biometric Information Data Object
{iso(1) standard(0) cbeff(19785) modules(0) tlv-encoded-explicit(15)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

  IMPORTS
     Quality, RegistryID, BiometricType, BiometricSubtype, Creator, BIRIndex
  FROM CBEFF-DATA-ELEMENTS
     -- The following data elements are defined in ISO/IEC 24787
     OCCcomparisonParameters, OCCfunctionalityInformation,
     OCCBiometricInformation
  FROM OCC-DATA-ELEMENTS
     -- The following data elements are defined by the BDB format provider
     -- (e.g. ISO/IEC 19794-x)
     CardholderChallengeType, StandardBDBType, ProprietaryBDBType
  FROM BDB-DATA-ELEMENTS;


  -- In all cases, omission of an optional component that represents a
  -- CBEFF data element is the encoding of the NO VALUE AVAILABLE for
  -- that data element.

  PatronHeaderVersion    ::= OCTET STRING (SIZE(2))
  -- CBEFF_patron_header_version
  -- The first octet encodes the major version number
  -- The second octet encodes the minor version number

  BCDTime                ::= OCTET STRING (SIZE(7))
  -- BCD encoded timestamp with format 'YYYYMMDDHHMMSS'

  BCDDatePeriod          ::= OCTET STRING (SIZE(8))
  -- Two concatenated BCD encoded dates with format YYYYMMDDYYYYMMDD
```

```
  -- First date shall always be earlier or equal to the second date

  ProductID             ::= OCTET STRING (SIZE(4))
  -- CBEFF_BDB_product_owner in the first two octets
  -- CBEFF_BDB_product_type in the last two octets

  FormatOwner           ::= OCTET STRING (SIZE(2))
  -- CBEFF_BDB_format_owner

  FormatType            ::= OCTET STRING (SIZE(2))
  -- CBEFF_BDB_format_type

  BiometricHeaderTemplateExplicit ::= SET
    {
    tagAllocationAuthority   [APPLICATION 24] CHOICE
      {
      oid                                     OBJECT IDENTIFIER,
      country               [APPLICATION 1]   OCTET STRING,
      issuerID              [APPLICATION 2]   OCTET STRING,
      application           [APPLICATION 15]  OCTET STRING
      } DEFAULT oid: {1 0 19785 3 1 19},

    biometricInformationDO   [APPLICATION 16] SET
      {
      patronHeaderVersion       [0]  PatronHeaderVersion DEFAULT'0100'H,
      bdbBiometricType          [1]  BiometricType OPTIONAL,
      bdbBiometricSubType       [2]  BiometricSubtype OPTIONAL,
        -- Required to be absent unless bdbBiometricType is present
      bdbCreationDate           [3]  BCDTime OPTIONAL,
        -- CBEFF_BDB_creation_date
      birCreator                [4]  Creator OPTIONAL,
      bdbValidityPeriod         [5]  BCDDatePeriod OPTIONAL,
      bdbPID                    [6]  ProductID OPTIONAL,
      bdbFormatOwner            [7]  FormatOwner,
      bdbFormatType             [8]  FormatType,
      birIndex                  [16] BIRIndex OPTIONAL
      }
    }

  BiometricInformationTemplateExplicit ::= [APPLICATION 96] SET
    {
    algorithmReference                  [0] OCTET STRING (SIZE(1)) OPTIONAL,
      -- A non-CBEFF data element, for on-card comparison - see ISO/IEC 24787
    referenceDataQualifier              [3] OCTET STRING (SIZE(1)) OPTIONAL,
      -- A non-CBEFF data element, for on-card comparison - see ISO/IEC 24787
      biometricHeaderTemplateExplicit [1] BiometricHeaderTemplateExplicit,
      occBiometricInformation           [2] OCCBiometricInformation OPTIONAL,
      bdbData                           [APPLICATION 46] SET
        -- A CBEFF BDB, mandatory for off-card-comparison
        {
        challengeCardholder [0]             CardholderChallengeType OPTIONAL,
                                            -- defined in BDB-DATA-ELEMENTS
        standardBDB         [1]             StandardBDBType OPTIONAL,
                                            -- defined in BDB-DATA-ELEMENTS
        proprietaryBDB      [2]             ProprietaryBDBType OPTIONAL,
                                            -- defined in BDB-DATA-ELEMENTS
        referenceQualifier  [3]             OCTET STRING (SIZE(1)) OPTIONAL,
        biometricData       [APPLICATION 46] OCTET STRING OPTIONAL -- primitive DO only
        } OPTIONAL
        -- A CBEFF BDB, mandatory for off-card-comparison

         birPayLoad [APPLICATION 19] OCTET STRING OPTIONAL
         -- CBEFF_BIR_payload, contents defined by ISO/IEC 24787
        }

  GroupBIT ::= [APPLICATION 97] SET OF BiometricInformationTemplateExplicit

END
```

Imported elements shall be defined prior to compiling the ASN.1 specification. For illustrative purposes the following example is provided. Implementers shall refer to the relevant standards where the elements are defined.

```
/*
-- These definitions can be found in ISO/IEC 24787.
-- They are copied here to ease understanding.
-- In case of discrepancy, the information in ISO/IEC 24787 is the one
-- to be used.
--
-- THE DATA DEFINED BELOW IS AN EXAMPLE FOR ILLUSTRATIVE PURPOSES ONLY
--
OCC-DATA-ELEMENTS
{iso(1) standard(0) 24787(24787) 2018(2018)}
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN
IMPORTS
    Quality, RegistryID
FROM CBEFF-DATA-ELEMENTS

    OCCcomparisonParameters ::= SET
      {
      minimumMaximumProbeSize       [1]  INTEGER OPTIONAL,
      orderingFeatures              [2]  OCTET STRING OPTIONAL,
      featureHandlingIndicator      [3]  OCTET STRING OPTIONAL,
      alignmentInformation          [4]  OCTET STRING OPTIONAL,
      minimumVerificationQuality    [7]  OCTET STRING OPTIONAL,
      verificationTypes             [16] INTEGER (0..255) OPTIONAL,
      estimationTime                [17] INTEGER (0..65535) OPTIONAL
      }

    OCCfunctionalityInformation ::= SET
      {
      maximumProbeSize              [0]  INTEGER (0..16777215)OPTIONAL,
      maximumReferenceSize          [1]  INTEGER (0..16777215)OPTIONAL,
      supportedNumberReferences     [2]  INTEGER (0..255) OPTIONAL,
      enrolmentPossibility          [3]  INTEGER (0..255) OPTIONAL,
      minimumVerificationQuality    [5]  Quality OPTIONAL,
      minimumQualityRequirements    [7]  OCTET STRING OPTIONAL,
      proprietaryData               [15] OCTET STRING OPTIONAL,
      verificationTypes             [16] OCTET STRING OPTIONAL,
      comparisonAlgID               [4]  RegistryID OPTIONAL
      }
```

```
   OCCBiometricInformation ::= SET
     {
     tagAllocationAuthority   [APPLICATION 24] CHOICE
       {
       oid                                          OBJECT IDENTIFIER,
       country                      [APPLICATION 1]  OCTET STRING,
       issuerID                     [APPLICATION 2]  OCTET STRING,
       application                  [APPLICATION 15] OCTET STRING
       }DEFAULT oid: {1 0 24787 2018},

     biometricInformationDO   [APPLICATION 16] SET
       {
       comparisonParameters      [17] OCCcomparisonParameters OPTIONAL,
       functionalityInformation  [18] OCCfunctionalityInformation OPTIONAL
       }
     }

END
*/
```

```
/*
-- These definitions shall be provided by the BDB format provider.
-- In case of being primitive data, these data elements shall be of type
-- OCTET STRING.
-- In case of constructed data, this/these data elements' complete
-- definiton should be provided with the data element names given here.
--
-- THE DATA DEFINED BELOW IS AN EXAMPLE FOR ILLUSTRATIVE PURPOSES ONLY
--
BDB-DATA-ELEMENTS
{iso(1) standard(0) 39794(39794) 2020(2020)} -- or whatever OID is registered
DEFINITIONS
AUTOMATIC TAGS ::=
BEGIN

  CardholderChallengeType ::= OCTET STRING -- if primitive (comment line
                                           -- if not used)
/*
```

```
  CardholderChallengeType ::= SET     -- if constructed, to be defined by the
                                      -- BDB data format provider (comment
                                      -- block if not used)
    {
    tagAllocationAuthority   [APPLICATION 24] CHOICE
      {
      oid                                     OBJECT IDENTIFIER,
      country                  [APPLICATION 1]  OCTET STRING,
      issuerID                 [APPLICATION 2]  OCTET STRING,
      application              [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018},

    biometricData [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/
  StandardBDBType ::= OCTET STRING -- if primitive (comment line
                                   -- if not used)

/*
  StandardBDBType ::= SET     -- if constructed, to be defined by the
                              -- BDB data format provider (comment block
                              -- if not used)
    {
    tagAllocationAuthority   [APPLICATION 24] CHOICE
      {
      oid                                     OBJECT IDENTIFIER,
      country                  [APPLICATION 1]  OCTET STRING,
      issuerID                 [APPLICATION 2]  OCTET STRING,
      application              [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018},

    biometricData              [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/
```

```
  ProprietaryBDBType ::= OCTET STRING -- if primitive (comment line
                                      -- if not used)

/*
  ProprietaryBDBType ::= SET      -- if constructed, to be defined by the
                                  -- BDB data format provider (comment block
                                  -- if not used)
    {
    tagAllocationAuthority   [APPLICATION 24] CHOICE
      {
      oid                                          OBJECT IDENTIFIER,
      country                    [APPLICATION 1]   OCTET STRING,
      issuerID                   [APPLICATION 2]   OCTET STRING,
      application                [APPLICATION 15] OCTET STRING
      }DEFAULT oid: {1 0 24787 2018},         -- or whatever OID is registered

    biometricData   [APPLICATION 16] SET
      {
      field0 [0] OCTET STRING (SIZE(1)) OPTIONAL,
      field1 [1] OCTET STRING (SIZE(1)) OPTIONAL,
      field2 [2] OCTET STRING (SIZE(1)) OPTIONAL
      }
    }
*/

END
*/
```


19.11 Tabular representations for information

19.11.1 The Biometric Information Template used for on-card biometric comparison

Table 27 shows the Biometric Information Template (with explicit tag allocation authority coding) used for on-card biometric comparison. The Biometric Information Template has the following substructure:

a) Data Objects (with tags 80_{Hex} and 83_{Hex}) containing values relevant for interindustry commands used for biometric verification as defined in ISO/IEC 7816-4;

b) Biometric Header Template (BHT) with tag $A1_{Hex}$. The tag allocation authority for the Data Objects nested in the DO 70_{Hex} within BHT is provided in the preceding DO with tag 78_{Hex};

   NOTE The ASN.1 OID for the appropriate tag allocation authority is defined within Table 27.

c) Data Objects for CBEFF data elements that are relevant for on-card biometric comparison;

d) Biometric Data Objects which are on-card biometric comparison specific and are defined in this clause and respective clauses of ISO/IEC 7816-11 and ISO/IEC 24787 standards (the biometric comparison parameters DO with tag $B1_{Hex}$, and biometric functionality information DO with tag $B2_{Hex}$).

Table 27 — The Biometric Information Template used for on-card biometric comparison (with explicit tag allocation authority)

Tag	Length	Value						Presence
$7F60_{Hex}$	Var.	Biometric Information Template						
		Tag	**Length**	**Value**				
		80_{Hex}	1	Algorithm reference for use in the VERIFY / EXT. AUTHENTICATE / MANAGE SE command as defined in ISO/IEC 7816-4				Optional
		83_{Hex}	1	Reference data qualifier for use in the VERIFY / EXT. AUTHENTICATE / MANAGE SE command as defined in ISO/IEC 7816-4				Optional
		$A1_{Hex}$	Var.	Biometric Header Template (BHT)				Mandatory
				Tag	**Length**	**Value**		
				78_{Hex}	Var	Compatible tag allocation authority		Mandatory if 'A1' is present
						Tag	**Length**	**Value**
						06_{Hex}	Var.	Object identifier (OID, encoding specified in ISO/IEC 8825-1) By default use ISO/IEC 19785-3 as the tag allocation authority for the BHT
				70_{Hex}	Var	Biometric information DOs specified by the tag allocation authority		Optional
						Tag	**Length**	**Value**
						80_{Hex}	2	CBEFF_patron_header_version (default 0100_{Hex})	Mandatory (if absent, the default value applies)
						90_{Hex}	Var.	CBEFF_BIR_index, unique identifier used for referencing this biometric data set in an application context outside the card	Optional
						81_{Hex}	1-3	CBEFF_BDB_biometric_type, see Table 29.	Optional
						82_{Hex}	1	CBEFF_BDB_biometric_subtype, see Table 30.	Optional, use with biometric type only

NOTE 1 In Table 27 the biometric data block as defined in ISO/IEC 19785-1 is not present when Biometric Information Template is read from the card before the on-card comparison. The biometric verification data (in the format as indicated within format owner and format type the Data Objects with tag 87_{Hex} and 88_{Hex} respectively, or in the constructed manner with $A7_{Hex}$) are presented to the card using, e.g. an ISO/IEC 7816 VERIFY or PERFORM BIOMETRIC OPERATION (COMPARE BIOMETRIC PROBE) command.

NOTE 2 In Table 27 no payload is present, since usually access to a payload, if used by the application, is granted after successful completion of the biometric verification. The payload can be retrieved using ISO/IEC 7816 interindustry access commands like GET DATA or READ BINARY.

NOTE 3 The outside world uses format owner / format type for identifying the required structure for the verification data. The biometric comparison algorithm in the card, which is able to process the verification data according to format owner / format type, is addressed by the algorithm reference, if the algorithm reference DO is present.

NOTE 4 If this patron format is used within the context of ISO/IEC 24787, then this DO is mandatory with the specifications for tag allocation authority and data objects defined in such standard.

Table 27 *(continued)*

Tag	Length	Value							Presence
						83_{Hex}	7	CBEFF_BDB_creation_date, creation date and time of biometric reference data: fourteen BCD digits (YYYYMMDDHHMMSS)	Optional
						84_{Hex}	Var.	CBEFF_BIR_creator	Optional
						85_{Hex}	8	CBEFF_BDB_validity_period, a pair of dates (not before, not after): sixteen BCD digits (YYYYMMDDYYYYMMDD)	Optional
						86_{Hex}	4	CBEFF_BDB_product_owner, CBEFF_BDB_product_type. Concatenation of product owner and product type, identifying the product that created the biometric reference data	Optional
						$A7_{Hex}$	8	CBEFF_BDB_format (as constructed)	Mandatory, if not in primitive mode (87_{Hex} and 88_{Hex})
						87_{Hex}	2	CBEFF_BDB_format_owner, format owner of the biometric verification data, value assigned by the biometrics registration authority, see NOTE 1 and NOTE 3	Mandatory, if $A7_{Hex}$ is not present
						88_{Hex}	2	CBEFF_BDB_format_type, format type of biometric verification data, specified by format owner, see NOTE 1 and NOTE 3	Mandatory, if $A7_{Hex}$ is not present
		$A2_{Hex}$	Var.	Biometric information DOs in addition to the BHT (see NOTE 4)					Mandatory
				Tag	**Length**	**Value**			
				78_{Hex}	Var	Compatible tag allocation authority (e.g. 1.0.24787.2018 for ISO/IEC 24787)			Mandatory if 'A2' is present
						Tag	**Length**	**Value**	
						06_{Hex}	Var.	Object identifier (OID, encoding specified in ISO/IEC 8825-1)	
				70_{Hex}	Var	Biometric information DOs specified by the tag allocation authority			Optional
						91_{Hex} / $B1_{Hex}$	Var.	In case the compatible tag allocation authority is ISO/IEC 24787, Biometric comparison parameters (primitive / constructed), as defined in ISO/IEC 24787	Optional

NOTE 1 In Table 27 the biometric data block as defined in ISO/IEC 19785-1 is not present when Biometric Information Template is read from the card before the on-card comparison. The biometric verification data (in the format as indicated within format owner and format type the Data Objects with tag 87_{Hex} and 88_{Hex} respectively, or in the constructed manner with $A7_{Hex}$) are presented to the card using, e.g. an ISO/IEC 7816 VERIFY or PERFORM BIOMETRIC OPERATION (COMPARE BIOMETRIC PROBE) command.

NOTE 2 In Table 27 no payload is present, since usually access to a payload, if used by the application, is granted after successful completion of the biometric verification. The payload can be retrieved using ISO/IEC 7816 interindustry access commands like GET DATA or READ BINARY.

NOTE 3 The outside world uses format owner / format type for identifying the required structure for the verification data. The biometric comparison algorithm in the card, which is able to process the verification data according to format owner / format type, is addressed by the algorithm reference, if the algorithm reference DO is present.

NOTE 4 If this patron format is used within the context of ISO/IEC 24787, then this DO is mandatory with the specifications for tag allocation authority and data objects defined in such standard.

Table 27 *(continued)*

Tag	Length	Value							Presence
						92_{Hex} / $B2_{Hex}$	Var.	In case the compatible tag allocation authority is ISO/IEC 24787, Biometric functionality information (primitive / constructed), as defined in ISO/IEC 24787	Optional
		$5F2E_{Hex}$ / $7F2E_{Hex}$	var.	Biometric data (primitive / constructed); see note 1					Optional

NOTE 1 In Table 27 the biometric data block as defined in ISO/IEC 19785-1 is not present when Biometric Information Template is read from the card before the on-card comparison. The biometric verification data (in the format as indicated within format owner and format type the Data Objects with tag 87_{Hex} and 88_{Hex} respectively, or in the constructed manner with $A7_{Hex}$) are presented to the card using, e.g. an ISO/IEC 7816 VERIFY or PERFORM BIOMETRIC OPERATION (COMPARE BIOMETRIC PROBE) command.

NOTE 2 In Table 27 no payload is present, since usually access to a payload, if used by the application, is granted after successful completion of the biometric verification. The payload can be retrieved using ISO/IEC 7816 interindustry access commands like GET DATA or READ BINARY.

NOTE 3 The outside world uses format owner / format type for identifying the required structure for the verification data. The biometric comparison algorithm in the card, which is able to process the verification data according to format owner / format type, is addressed by the algorithm reference if the algorithm reference DO is present.

NOTE 4 If this patron format is used within the context of ISO/IEC 24787, then this DO is mandatory with the specifications for tag allocation authority and data objects defined in such standard.

The use of the Biometric Information Template (with explicit tag allocation authority coding) shown in Table 27, which is intended to be retrieved prior to a subject biometric verification, and the verification process itself with the related interindustry commands and their security requirements, are specified in ISO/IEC 7816-11.

19.11.2 The Biometric Information Template used for off-card biometric comparison

Table 28 shows the Biometric Information Template (with explicit tag allocation authority coding) used for off-card biometric comparison. The Biometric Information Template has the following substructure:

a) Biometric Header Template (BHT) with tag $A1_{Hex}$. The tag allocation authority for the Data Objects nested in the BHT is provided in tag 78_{Hex};

b) Biometric Data Objects for CBEFF data elements that are relevant for off-card biometric comparison;

c) A biometric reference Data Object (primitive or constructed) with tags representing the CBEFF Biometric Data Block (BDB);

d) Payload as the content of an optional Data Object for discretionary data with tag 53_{Hex} or 73_{Hex} as defined in ISO/IEC 7816-6.

The use of the data structure shown in Table 28 is not restricted to smartcards. The data structure may also be used in other types of cards, e.g. magnetic stripe cards, optical memory cards or cards with 2-dimensional barcodes.

Table 28 — The Biometric Information Template used for off-card biometric comparison (with explicit tag allocation authority)

Tag	Length	Value							Presence
$7F60_{Hex}$	Var.	Biometric Information Template							
		Tag	**Length**	**Value**					
		$A1_{Hex}$	Var.	Biometric Header Template (BHT)					Mandatory
				Tag	**Length**	**Value**			
				78_{Hex}	Var	Compatible tag allocation authority			Mandatory if 'A1' is present
						Tag	**Length**	**Value**	
						06_{Hex}	Var.	Object identifier (OID, encoding specified in ISO/IEC 8825-1)	
				70_{Hex}	Var	Biometric information DOs specified by the tag allocation authority			Optional
						Tag	**Length**	**Value**	
						80_{Hex}	2	CBEFF_patron_header_version (default 0100_{Hex})	Mandatory (if absent, the default value applies)
						90_{Hex}	Var.	CBEFF_BIR_index, unique identifier used for referencing this biometric data set in an application context outside the card	Optional
						81_{Hex}	1-3	CBEFF_BDB_biometric_type, see Table 29.	Optional
						82_{Hex}	1	CBEFF_BDB_biometric_subtype, see Table 30.	Optional, use with biometric type only
						83_{Hex}	7	CBEFF_BDB_creation_date, creation date and time of biometric reference data: fourteen BCD digits (YYYYMMDDHHMMSS)	Optional
						84_{Hex}	Var.	CBEFF_BIR_creator	Optional
						85_{Hex}	8	CBEFF_BDB_validity_period, a pair of dates (not before, not after): sixteen BCD digits (YYYYMMDDYYYYMMDD)	Optional
						86_{Hex}	4	CBEFF_BDB_product_owner, CBEFF_BDB_product_type. Concatenation of product owner and product type, identifying the product that created the biometric reference data	Optional
						$A7_{Hex}$	8	CBEFF_BDB_format (as constructed)	Mandatory, if not in primitive mode (87_{Hex} and 88_{Hex})
						87_{Hex}	2	CBEFF_BDB_format_owner, format owner of the biometric verification data, value assigned by the biometrics registration authority, see NOTE	Mandatory, if $A7_{Hex}$ is not present

NOTE The use of a constructed Data Object with tag $7F2E_{Hex}$ for biometric reference data encoding allows the integration of a biometric challenge Data Objects for user prompting (e.g. a phrase to be spoken) and a concatenation of biometric data with standardized and proprietary format. The respective Data Objects are specified in ISO/IEC 7816-11.

Table 28 *(continued)*

Tag	Length	Value							Presence
						88_{Hex}	2	CBEFF_BDB_format_type, format type of biometric verification data, specified by format owner, see NOTE	Mandatory, if $A7_{Hex}$ is not present
		$5F2E_{Hex}$ / $7F2E_{Hex}$	var.	CBEFF BDB, biometric reference data (primitive / constructed); see NOTE					Mandatory
		53_{Hex} / 73_{Hex}	var.	CBEFF_BIR_payload, optional data for payload (primitive / constructed)					Optional
NOTE The use of a constructed Data Object with tag $7F2E_{Hex}$ for biometric reference data encoding allows the integration of a biometric challenge Data Objects for user prompting (e.g. a phrase to be spoken) and a concatenation of biometric data with standardized and proprietary format. The respective Data Objects are specified in ISO/IEC 7816-11.									

19.12 The Biometric Information Template Group Data Object

Several Biometric Information Templates may be nested in a Biometric Information Template Group (tag $7F61_{Hex}$ as defined in ISO/IEC 7816-11). The construction and use of Biometric Information Template Group Data Objects both for on-card and off-card biometric comparison are specified in ISO/IEC 7816-11.

19.13 Abstract values and encodings for biometric type and subtype

Values for CBEFF_biometeric_type and CBEFF_biometric_subtype are defined in 6.2 under the BiometricType ASN.1 definition and the BiometricSubtype ASN.1 definition.

19.14 Illustrative examples

Examples on the use of this patron format with explicit tag allocation authority can be found in ISO/IEC 24787 and in ISO/IEC 7816-11.

Annex A
(informative)

Guidelines on the specification of patron formats

A.1 General

A.1.1 This document requires that a CBEFF patron format be defined that encodes some or all of the values of some or all of the CBEFF data elements.

A.1.2 For CBEFF data elements defined as mandatory, there is a requirement to encode at least one of the specified abstract values, possibly by a null encoding.

For mandatory CBEFF data elements, the value NO VALUE AVAILABLE shall only be used if it is specified as a valid value in ISO/IEC 19785-1 (see ISO/IEC 19785-1:2020, 7.1.8).

A.1.3 For CBEFF data elements defined as optional, the patron format has 3 options:

1) The data element can be supported only for one abstract value (which may be the NO VALUE AVAILABLE abstract value). In this case, a null encoding is used (zero bits).

2) The CBEFF data element can be encoded as a mandatory field in the patron format.

   i) The format would then require NO VALUE AVAILABLE to be encoded as an explicit encoding of that field.

   ii) The format may, but need not, define encodings of that field for other CBEFF-defined abstract values.

   iii) The format may, but need not, define encodings of that field for additional abstract values defined by the format.

3) The CBEFF data element can be encoded as an optional field in the patron format.

   i) The abstract value NO VALUE AVAILABLE would then normally be encoded by the absence of the optional field, but any other abstract value can be chosen as the default.

   ii) The format may, but need not, define encodings of that field for other CBEFF-defined abstract values.

   iii) The format may, but need not, define encodings of that field for additional abstract values defined by the format.

   iv) The format shall specify how the presence or absence of this optional field is determined.

A.1.4 Patrons that are defining a new patron format should give due consideration to inclusion of optional CBEFF data elements and abstract values that may be required for the successful processing of BDB formats that are expected to be associated with the new patron format.

A.2 Basic encoding mechanisms

A.2.1 Where only a single value of a CBEFF data element (for example, the NO VALUE AVAILABLE or NO ENCRYPTION abstract values) is to be supported, this can be encoded using zero bits — that is, by not

having a field for that data element. This would be the normal support for optional CBEFF data elements that are to be supported only with the NO VALUE AVAILABLE value.

A.2.2 The simplest form of encoding is a fixed-length field (not necessarily a multiple of eight bits) with a distinct encoding for each supported value. Any further encodings of the fixed length field can either be reserved or can be used to carry values that are not CBEFF data element values, but are determined by the CBEFF patron.

A.2.3 If there are encodings of the fixed-length field that do not correspond to CBEFF data element values or to values defined by the CBEFF patron, then the patron format would normally specify that such encodings should not be generated, and should be treated as NO VALUE AVAILABLE on reading. (This is equivalent to saying that these encodings are reserved for future use.)

A.2.4 Another common form of encoding is to use a single "presence bit" to indicate whether a field is present. If absent, then one of the values being supported is implied. (This will usually be the NO VALUE AVAILABLE value, but can in principle be any CBEFF-defined or patron-defined value). If present, the field would again be fixed length, and would encode the remaining supported values. This can generalize into what is called a Huffman encoding, in which one bit is used to represent the most commonly used value, and an increasing number of bits are used to represent increasingly uncommon values.

A.3 Octet-alignment

A.3.1 There is no requirement for the fields of a patron format to be an integral multiple of eight bits, nor to start on an octet boundary.

A.3.2 Decisions to include padding bits, or to use octet alignment in the patron format specification, are taken by the CBEFF patron.

A.3.3 If padding bits are included, it will be normal to require that they be set to a fixed value when generated (typically zero), and are ignored on reading.

A.4 Length fields

A.4.1 All CBEFF patron formats are required to be an integral number of octets in the complete encoding, but are not required to be self-delimiting.

A.4.2 Self-delimiting means that the end of an encoding can always be determined using only knowledge of the encoding itself. This can be achieved by the use of one or more length fields, or by particular terminating patterns, or by a combination of these and other mechanisms, or by specifying the patron format as being of fixed length. Normally, however, a patron format encoding (a BIR) will be included in a database, a data structure or object, or a message between systems, where the delimitation of the BIR will be provided by the containing mechanism.

A.5 CBEFF data element values and patron format fields

A.5.1 There is no requirement for a one-to-one correspondence between CBEFF data elements and patron format fields.

A.5.2 For example, if five values of one CBEFF data element and three values of a second CBEFF data element are to be supported, this requires fifteen distinct encodings that can, if desired, be provided as the values of a single 4-bit field, rather than as the values of two separate 3-bit and 2-bit fields. This would be a decision taken by the CBEFF patron when defining the CBEFF patron format.

A.6 Historically used encodings of certain abstract values

Earlier, non-ISO versions of CBEFF specified abstract values and encodings for the data elements equivalent to CBEFF_BDB_biometric_type and CBEFF_BDB_biometric_subtype. Patron formats requiring compatibility with these encodings should specify them as described in Table A.1 and Table A.2.

Table A.1 — Historical abstract values and encodings for biometric type

Abstract value	Encoding (hex)
Multiple Biometrics Used	000001_{Hex}
Facial Features	000002_{Hex}
Voice	000004_{Hex}
Fingerprint	000008_{Hex}
Iris	000010_{Hex}
Retina	000020_{Hex}
Hand Geometry	000040_{Hex}
Signature Dynamics	000080_{Hex}
Keystroke Dynamics	000100_{Hex}
Lip Movement	000200_{Hex}
Thermal Face Image	000400_{Hex}
Thermal Hand Image	000800_{Hex}
Gait	001000_{Hex}
Body Odour	002000_{Hex}
DNA	004000_{Hex}
Ear Shape	008000_{Hex}
Finger Geometry	010000_{Hex}
Palm Print	020000_{Hex}
Vein Pattern	040000_{Hex}
Foot Print	080000_{Hex}

Table A.2 — Historical abstract values and encodings for biometric subtype

b8 b7 b6 b5 b4 b3 b2 b1	Biometric subtype
0 0 0 0 0 0 0 0	No information given
0 1	Right
1 0	Left
0 0 0	No meaning
0 0 1	Thumb
0 1 0	Index finger
0 1 1	Middle finger
1 0 0	Ring finger
1 0 1	Little finger
x x x	Reserved for future use

A.7 Variable length data elements

A.7.1 CBEFF defines certain data elements (for example, CBEFF_BDB_challenge_response and CBEFF_BIR_creator) that can only be encoded as variable length encodings if all abstract values are to be supported. These data elements support abstract values that are strings of transparent octets, or of ISO/IEC 10646 characters.

A.7.2 Patron formats should ensure that when variable length fields are encoded in SBHs it is possible to easily determine the location of the BDB. This may require a field that encodes the length of the entire SBH.

A.8 Security Blocks

A.8.1 Earlier, non-ISO versions of CBEFF defined abstract values of MAC and SIGNATURE for the INTEGRITY data element. Those versions of CBEFF defined an unstructured "signature block", requiring that the specific message authentication code or digital signature algorithm be inferred from the domain of use. In addition, if encryption was used, those versions of CBEFF did not provide a standard way of conveying the parameters of the encryption (such as algorithms, key references, or session keys) in the BIR.

A.8.2 This version of CBEFF defines the security block (SB) as the top level of a structure containing security information for the BIR, which may include BDB encryption information (algorithms and parameters), BIR integrity information (algorithms, parameters, and a signature or MAC), or both.

A.8.3 CBEFF also defines data elements for an SB format owner and an SB format identifier, which are supported by the CBEFF Registration Authority and together provide a universally unique identification for any SB format, analogous with BDB format identifiers. Each SB format owner / SB format identifier pair identifies the specification of the information in the SB of a BIR and the format of the SB. For example, if an SB format supports the encryption of a BDB using any of several specified encryption algorithms, that SB format would specify encodings of the SB such that processes that need to decrypt a BDB would be able to determine which algorithm (with what parameters) and which key to use. Similarly, if an SB format supports the use of MACs to insure the integrity of a BIR, that SB format would specify the SB such that a processor could use information encoded in the SB to verify the BIR's integrity.

Annex B
(informative)

Conformance of the defined patron formats

B.1 Introduction

This informative annex collects information for stating conformance of the patron formats defined in this document. This annex constitutes a patron format conformance statement (PFCS) as specified by ISO/IEC 19785-1, Annex A.

B.2 Identifying information

The following table reflects where identifying information (required by ISO/IEC 19785-1, A.2.1) can be found in each of the patron formats.

Table B.1 — Sources for conformance information

Required Information	Patron format reference	
	Clause X	**E.g.: Clause 13**
i) Patron name	See X.1	See 13.1
ii) Patron identifier (Patron format owner)	See X.2	See 13.2
iii) Patron format name	See X.3	See 13.3
iv) Patron format identifier (Patron format type)	See X.4	See 13.4
v) Patron format ASN.1 OID	See X.5	See 13.5
vi) Domain of use description	See X.6	See 13.6
vii) Patron format version (Version identifier)	See X.7	See 13.7
viii) CBEFF version	See X.8	See 13.8
ix) CBEFF patron format structure type	See X.9	See 13.9
x) Supported CBEFF-defined data elements and abstract values	See Table B.2	See Table B.2
xi) Patron-defined data elements and abstract values	See Clause B.4	See Clause B.4
NOTE For Clause 15 — Biometric application programming interface (BioAPI 2.0) — the patron format is defined in ISO/IEC 19784-1.		

B.3 CBEFF-defined data elements and abstract values

Table B.2 indicates which CBEFF-defined data elements are implemented in each of the patron formats. ISO/IEC 19785-1 requires that the following fields shall be mandatory and included in all patron format BIR structures that contain BDBs:

— CBEFF_BDB_format_owner

— CBEFF_BDB_format_type