EMV®
Integrated Circuit Card Specifications for Payment Systems

Book 1

Application Independent ICC to Terminal Interface Requirements

Version 4.4
October 2022

The EMV® Specifications are provided “AS IS” without warranties of any kind, and EMVCo neither assumes nor accepts any liability for any errors or omissions contained in these Specifications. EMVCO DISCLAIMS ALL REPRESENTATIONS AND WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AS TO THESE SPECIFICATIONS.

EMVCo makes no representations or warranties with respect to intellectual property rights of any third parties in or in relation to the Specifications. EMVCo undertakes no responsibility to determine whether any implementation of these Specifications may violate, infringe, or otherwise exercise the patent, copyright, trademark, trade secret, know-how, or other intellectual property rights of third parties, and thus any person who implements any part of these Specifications should consult an intellectual property attorney before any such implementation.

Without limiting the foregoing, the Specifications may provide for the use of public key encryption and other technology, which may be the subject matter of patents in several countries. Any party seeking to implement these Specifications is solely responsible for determining whether its activities require a license to any such technology, including for patents on public key encryption technology. EMVCo shall not be liable under any theory for any party’s infringement of any intellectual property rights in connection with these Specifications.

Revision Log – Version 4.4

The following changes have been made to Book 1 since the publication of Version 4.3. Numbering and cross references in this version have been updated to reflect changes introduced by the published bulletins.

Incorporated changes described in the following Specification Bulletins:

Specification Bulletin no. 111: Clarification on Application Filtering

Specification Bulletin no. 151: Clarification on Cardholder Selection and Cardholder Confirmation

Specification Bulletin no. 175: Application Selection Registered Proprietary Data

Specification Bulletin no. 178, Third Edition: Tokenisation Data Objects – Payment Account Reference (PAR)

Specification Bulletin no. 231: Issuer Identification Number Extended (IINE)

Removed Part II, *Electromechanical Characteristics, Logical Interface, and Transmission Protocols*

The topics formerly addressed in Part II are included in *EMV Level 1 Specifications for Payment Systems, EMV Contact Interface Specification*.

Minor editorial clarifications and corrections, including those described in the following:

Specification Bulletin no. 175: Application Selection Registered Proprietary Data

Contents

Part V – Common Core Definitions

Tables

Figures

Part I

General

1 Scope

This document, the *Integrated Circuit Card (ICC) Specifications for Payment Systems – Book 1, Application Independent ICC to Terminal Interface Requirements*, describes the minimum functionality required of integrated circuit cards (ICCs) and terminals to ensure correct operation and interoperability independent of the application to be used. Additional proprietary functionality and features may be provided, but these are beyond the scope of this specification and interoperability cannot be guaranteed.

The *Integrated Circuit Card Specifications for Payment Systems* includes the following additional documents, all available on http://www.emvco.com:

- Book 2 – Security and Key Management
- Book 3 – Application Specification
- Book 4 – Cardholder, Attendant, and Acquirer Interface Requirements

1.1 Changes in Version 4.4

This release incorporates all relevant Specification Bulletins, Application Notes, amendments, etc. published up to the date of this release.

Part II, *Electromechanical Characteristics, Logical Interface, and Transmission Protocols*, was removed in Version 4.4. The topics formerly addressed in Part II are included in *EMV Level 1 Specifications for Payment Systems, EMV Contact Interface Specification*.

The Revision Log at the beginning of the Book provides additional detail about changes to this Book.

1.2 Structure

Book 1 consists of the following parts:

Part I – **General**

Part II – Removed in Version 4.4.

Part III – **Files, Commands, and Application Selection**

Part IV – **Annexes**

Part V – **Common Core Definitions**

Part I includes this introduction, as well as data applicable to all Books: normative references, definitions, abbreviations, notations, data element format convention, and terminology.

Part III defines data elements, files, and commands as they apply to the exchange of information between an ICC and a terminal. In particular it covers:

- Data elements and their mapping onto data objects.
- Structure and referencing of files.
- Structure and coding of messages between the ICC and the terminal to achieve application selection.

Part III also defines the application selection process from the standpoint of both the card and the terminal. The logical structure of data and files within the card that is required for the process is specified, as is the terminal logic using the card structure.

Part IV includes a data elements table specific to application selection, and example directory structures.

Part V defines an optional extension to be used when implementing the Common Core Definitions (CCD).

The Book also includes a revision log and an index.

1.3 Underlying Standards

This specification is based on the ISO/IEC 7816 series of standards and should be read in conjunction with those standards. However, if any of the provisions or definitions in this specification differ from those standards, the provisions herein shall take precedence.

1.4 Audience

This specification is intended for use by manufacturers of ICCs and terminals, system designers in payment systems, and financial institution staff responsible for implementing financial applications in ICCs.

2 Normative References

The following specifications and standards contain provisions that are referenced in these specifications. The latest version shall apply unless a publication date is explicitly stated.

EMV Contact Interface Specification	EMV Level 1 Specifications for Payment Systems, EMV Contact Interface Specification
EMV Tokenisation Framework	EMV Payment Tokenisation Specification – Technical Framework Framework specification for an interoperable Payment Tokenisation solution.
FIPS 202	SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions
IEEE P1363	Standard Specifications For Public-Key Cryptography
ISO 639-1	Codes for the representation of names of languages – Part 1: Alpha-2 Code **Note:** This standard is updated continuously by ISO. Additions/changes to ISO 639-1:1988: Codes for the Representation of Names of Languages are available on: http://www.loc.gov/standards/iso639-2/php/code_changes.php
ISO 3166	Codes for the representation of names of countries and their subdivisions
ISO 4217	Codes for the representation of currencies and funds
ISO/IEC 7812-1	Identification cards – Identification of issuers — Part 1: Numbering System
ISO/IEC 7813	Identification cards – Financial transaction cards
ISO/IEC 7816-4	Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange
ISO/IEC 7816-5	Identification cards — Integrated circuit cards — Part 5: Registration of application providers
ISO/IEC 7816-6	Identification cards – Integrated circuit cards – Part 6: Interindustry data elements for interchange

ISO/IEC 7816-11	Identification cards – Integrated circuit cards – Personal verification through biometric methods
ISO 8583:1987	Bank card originated messages – Interchange message specifications – Content for financial transactions
ISO 8583:1993	Financial transaction card originated messages – Interchange message specifications
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)
ISO/IEC 8859	Information processing – 8-bit single-byte coded graphic character sets
ISO 9362	Banking – Banking telecommunication messages – Bank identifier codes
ISO 9564-1	Financial services – Personal Identification Number (PIN) management and security – Part 1: Basic principles and requirements for PINs in card-based systems
ISO/IEC 9796-2	Information technology – Security techniques – Digital signature schemes giving message recovery – Part 2: Integer factorization based mechanisms
ISO/IEC 9797-1	Information technology – Security techniques – Message Authentication Codes – Part 1: Mechanisms using a block cipher
ISO/IEC 9797-2	Information technology – Security techniques – Message Authentication Codes (MACs) – Part 2: Mechanisms using a dedicated hash-function
ISO/IEC 10116	Information technology – Security techniques – Modes of operation for an *n*-bit block cipher
ISO/IEC 10118-3	Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions
ISO/IEC 11770-6	Information technology – Security techniques – Key management — Part 6: Key derivation
ISO 13616	Banking and related financial services – International bank account number (IBAN)

ISO/IEC 14888-3	Information technology – Security techniques – Digital signatures with appendix — Part 3: Discrete logarithm based mechanisms
ISO/IEC 15946-1	Information technology – Security techniques – Cryptographic techniques based on elliptic curves — Part 1: General
ISO/IEC 15946-5	Information technology – Security techniques – Cryptographic techniques based on elliptic curves — Part 5: Elliptic curve generation
ISO 16609	Banking – Requirements for message authentication using symmetric techniques
ISO/IEC 18031	Information technology – Security techniques – Random bit generation
ISO/IEC 18033-2	Information technology – Security techniques – Encryption algorithms – Part 2: Asymmetric ciphers
ISO/IEC 18033-3	Information technology – Security techniques – Encryption algorithms – Part 3: Block ciphers
ISO/IEC 19772	Information technology – Security techniques – Authenticated encryption
ISO/IEC 19785-3	Information technology – Common Biometric Exchange Formats Framework – Patron format specifications
ISO/IEC 19794	Information technology – Biometric data interchange formats
ISO/IEC 19794-2	Information technology – Biometric data interchange formats – Part 2: Finger minutiae data
SEC 1	Elliptic Curve Cryptography (available at http://www.secg.org)

3 Definitions

The following terms are used in one or more books of these specifications.

Application	The application protocol between the card and the terminal and its related set of data.
Application Authentication Cryptogram	An Application Cryptogram generated by the card when declining a transaction.
Application Cryptogram	A cryptogram generated by the card in response to a GENERATE AC command. See also: • Application Authentication Cryptogram • Authorisation Request Cryptogram • Transaction Certificate
Authentication	The provision of assurance of the claimed identity of an entity or of data origin.
Authorisation Request Cryptogram	An Application Cryptogram generated by the card when requesting online authorisation.
Authorisation Response Cryptogram	A cryptogram generated by the issuer in response to an Authorisation Request Cryptogram.
Biometric Data Block	A block of data with a specific format that contains information captured from a biometric capture device and that could be used as follows: • stored in the card as part of the biometric reference template • sent to the ICC in the data field of the PIN CHANGE/UNBLOCK command • sent to the ICC in the data field of the VERIFY command for offline biometric verification • sent online for verification The format of the BDB is outside the scope of this specification.

Biometric Reference Template	Biometric data stored in the card as reference. Data provided by a biometric capture device would be compared against the biometric reference template to determine a match.
Biometric Verification	The process of determining that the biometrics presented, such as finger, palm, iris, voice, or facial, are valid.
Byte	8 bits.
Card	A payment card as defined by a payment system.
Certificate	The public key and identity of an entity together with some other information, rendered unforgeable by signing with the private key of the certification authority which issued that certificate.
Certification Authority	Trusted third party that establishes a proof that links a public key and other relevant information to its owner.
Ciphertext	Enciphered information.
Combined DDA/Application Cryptogram Generation	A form of offline dynamic data authentication.
Command	A message sent by the terminal to the ICC that initiates an action and solicits a response from the ICC.
Command Chaining	A mechanism where consecutive command-response pairs can be chained.
Compromise	The breaching of secrecy or security.
Concatenation	Two elements are concatenated by appending the bytes from the second element to the end of the first. Bytes from each element are represented in the resulting string in the same sequence in which they were presented to the terminal by the ICC, that is, most significant byte first. Within each byte bits are ordered from most significant bit to least significant. A list of elements or objects may be concatenated by concatenating the first pair to form a new element, using that as the first element to concatenate with the next in the list, and so on.

Contact	A conducting element ensuring galvanic continuity between integrated circuit(s) and external interfacing equipment.
Cryptogram	Result of a cryptographic operation.
Cryptographic Algorithm	An algorithm that transforms data in order to hide or reveal its information content.
Data Integrity	The property that data has not been altered or destroyed in an unauthorised manner.
Decipherment	The reversal of a corresponding encipherment.
DEM1	A family of data encapsulation mechanisms defined in ISO/IEC 18033-2.
Digital Signature	An asymmetric cryptographic transformation of data that allows the recipient of the data to prove the origin and integrity of the data, and protect the sender and the recipient of the data against forgery by third parties, and the sender against forgery by the recipient.
Dynamic Data Authentication	A form of offline dynamic data authentication
Elliptic Curve Cryptography	Public key cryptography based on the algebraic structure of elliptic curves over finite fields.
Encipherment	The reversible transformation of data by a cryptographic algorithm to produce ciphertext.
Exclusive-OR	Binary addition with no carry, giving the following values: 0 + 0 = 0 0 + 1 = 1 1 + 0 = 1 1 + 1 = 0
Extended Data Authentication	A form of offline dynamic data authentication.
Facial Verification	The process of determining that the face presented is valid.
Financial Transaction	The act between a cardholder and a merchant or acquirer that results in the exchange of goods or services against payment.

Finger Verification	The process of determining that the finger presented is valid.
Function	A process accomplished by one or more commands and resultant actions that are used to perform all or part of a transaction.
Hash Function	A function that maps strings of bits to fixed-length strings of bits, satisfying the following two properties: • It is computationally infeasible to find for a given output an input which maps to this output. • It is computationally infeasible to find for a given input a second input that maps to the same output. Additionally, if the hash function is required to be collision-resistant, it must also satisfy the following property: • It is computationally infeasible to find any two distinct inputs that map to the same output.
Hash Result	The string of bits that is the output of a hash function.
I2OSP	An integer to octet string conversion primitive function defined in ISO/IEC 18033-2.
Integrated Circuit(s)	Electronic component(s) designed to perform processing and/or memory functions.
Integrated Circuit(s) Card	A card into which one or more integrated circuits are inserted to perform processing and memory functions.
Interface Device	That part of a terminal into which the ICC is inserted, including such mechanical and electrical devices as may be considered part of it.
Iris Verification	The process of determining that the iris presented is valid.
Issuer Action Code	Any of the following, which reflect the issuer-selected action to be taken upon analysis of the TVR: • Issuer Action Code – Default • Issuer Action Code – Denial • Issuer Action Code – Online

Kernel	The set of functions required to be present on every terminal implementing a specific interpreter. The kernel contains device drivers, interface routines, security and control functions, and the software for translating from the virtual machine language to the language used by the real machine. In other words, the kernel is the implementation of the virtual machine on a specific real machine.
Key	A sequence of symbols that controls the operation of a cryptographic transformation.
Key Introduction	The process of generating, distributing, and beginning use of a key pair.
Key Withdrawal	The process of removing a key from service as part of its revocation.
Keypad	Arrangement of numeric, command, and, where required, function and/or alphanumeric keys laid out in a specific manner.
Library	A set of high-level software functions with a published interface, providing general support for terminal programs and/or applications.
Logical Compromise	The compromise of a key through application of improved cryptanalytic techniques, increases in computing power, or combination of the two.
Magnetic Stripe	The stripe containing magnetically encoded information.
Message	A string of bytes sent by the terminal to the card or vice versa, excluding transmission-control characters.
Message Authentication Code	A symmetric cryptographic transformation of data that protects the sender and the recipient of the data against forgery by third parties.
Nibble	The four most significant or least significant bits of a byte.
Offline Data Encipherment	Offline encipherment of data, in particular for cardholder PIN and biometric data.
Padding	Appending extra bits to either side of a data string.
Palm Verification	The process of determining that the palm presented is valid.

Path	Concatenation of file identifiers without delimitation.
Payment System Environment	A logical construct within the ICC, the entry point to which is a Directory Definition File (DDF) named '1PAY.SYS.DDF01'. This DDF contains a Payment System Directory which in turn contains entries for one or more Application Definition Files (ADFs) which are formatted according to this specification.
Physical Compromise	The compromise of a key resulting from the fact that it has not been securely guarded, or a hardware security module has been stolen or accessed by unauthorised persons.
PIN Pad	Arrangement of numeric and command keys to be used for personal identification number (PIN) entry. Also known as a “PIN Entry Device” (PED).
Plaintext	Unenciphered information.
Potential Compromise	A condition where cryptanalytic techniques and/or computing power has advanced to the point that compromise of a key of a certain length is feasible or even likely.
Private Key	That key of an entity’s asymmetric key pair that should only be used by that entity. In the case of a digital signature scheme, the private key defines the signature function.
Public Key	That key of an entity’s asymmetric key pair that can be made public. In the case of a digital signature scheme, the public key defines the verification function.
Public Key Certificate	The public key information of an entity signed by the certification authority and thereby rendered unforgeable.
Response	A message returned by the ICC to the terminal after the processing of a command message received by the ICC.
RSA-KEM	A family of key encapsulation mechanisms defined in ISO/IEC 18033-2.
RSATransform	The RSA exponentiation that is used for encryption and decryption, and generating and verifying a signature.

Script	A command or a string of commands transmitted by the issuer to the terminal for the purpose of being sent serially to the ICC as commands.
Secret Key	A key used with symmetric cryptographic techniques and usable only by a set of specified entities.
Socket	An execution vector defined at a particular point in an application and assigned a unique number for reference.
Static Data Authentication	Offline static data authentication
Symmetric Cryptographic Technique	A cryptographic technique that uses the same secret key for both the originator's and recipient's transformation. Without knowledge of the secret key, it is computationally infeasible to compute either the originator's or the recipient's transformation.
Template	Value field of a constructed data object, defined to give a logical grouping of data objects.
Terminal	The device used in conjunction with the ICC at the point of transaction to perform a financial transaction. The terminal incorporates the interface device and may also include other components and interfaces such as host communications.
Terminal Action Code	Any of the following, which reflect the acquirer-selected action to be taken upon analysis of the TVR: • Terminal Action Code – Default • Terminal Action Code – Denial • Terminal Action Code – Online
Terminate Card Session	End the card session by deactivating the IFD contacts according to EMV Contact Interface Specification and displaying a message indicating that the ICC cannot be used to complete the transaction.
Terminate Transaction	Stop the current application and deactivate the card.
Transaction	An action taken by a terminal at the user's request. For a POS terminal, a transaction might be payment for goods, etc. A transaction selects among one or more applications as part of its processing flow.

Transaction Certificate	An Application Cryptogram generated by the card when accepting a transaction.
Virtual Machine	A theoretical microprocessor architecture that forms the basis for writing application programs in a specific interpreter software implementation.
Voice Verification	The process of determining that the voice presented is valid.

4 Abbreviations, Notations, Conventions, and Terminology

4.1 Abbreviations

a	Alphabetic (see Data Element Format Conventions, section 4.3)
AAC	Application Authentication Cryptogram
AAD	Additional Authenticated Data
AC	Application Cryptogram
ADF	Application Definition File
AEF	Application Elementary File
AES	Advanced Encryption Standard
AFL	Application File Locator
AID	Application Identifier
AIP	Application Interchange Profile
an	Alphanumeric (see section 4.3)
ans	Alphanumeric Special (see section 4.3)
APDU	Application Protocol Data Unit
API	Application Program Interface
ARC	Authorisation Response Code
ARPC	Authorisation Response Cryptogram
ARQC	Authorisation Request Cryptogram
ASI	Application Selection Indicator
ASN	Abstract Syntax Notation
ATC	Application Transaction Counter
ATM	Automated Teller Machine
ATR	Answer to Reset

AUC	Application Usage Control
b	Binary (see section 4.3)
BCD	Binary Coded Decimal
BDB	Biometric Data Block
BEK	Biometric Encryption Key
BER	Basic Encoding Rules (defined in ISO/IEC 8825-1)
BHT	Biometric Header Template
BIC	Bank Identifier Code
BIT	Biometric Information Template
BMK	Biometric MAC Key
CA	Certification Authority
CAD	Card Accepting Device
C-APDU	Command APDU
CBC	Cipher Block Chaining
CBEFF	Common Biometric Exchange Formats Framework
CCD	Common Core Definitions
CCI	Common Core Identifier
CCYYMMDD	Year (4 digits), Month, Day
CDA	Combined DDA/Application Cryptogram Generation
CDOL	Card Risk Management Data Object List
CID	Cryptogram Information Data
CLA	Class Byte of the Command Message
cn	Compressed Numeric (see section 4.3)
CPU	Central Processing Unit
CRL	Certificate Revocation List
CSU	Card Status Update
CV	Cryptogram Version
CV Rule	Cardholder Verification Rule

CVM	Cardholder Verification Method
CVR	Card Verification Results
DDA	Dynamic Data Authentication
DDF	Directory Definition File
DDOL	Dynamic Data Authentication Data Object List
DES	Data Encryption Standard
DF	Dedicated File
DIR	Directory
DOL	Data Object List
ECB	Electronic Code Book
EC-SDSA	Elliptic Curve Schnorr Digital Signature Algorithm
ECC	Elliptic Curve Cryptography
EF	Elementary File
EN	European Norm
FC	Format Code
FCI	File Control Information
Hex	Hexadecimal
HHMMSS	Hours, Minutes, Seconds
HMAC	Keyed-hash Message Authentication Code
I/O	Input/Output
IAC	Issuer Action Code (Denial, Default, Online)
IAD	Issuer Application Data
IBAN	International Bank Account Number
IC	Integrated Circuit
ICC	Integrated Circuit(s) Card
ICCD	Issuer Certified Card Data
IEC	International Electrotechnical Commission
IFD	Interface Device

IIN	Issuer Identification Number
IINE	Issuer Identification Number Extended
INS	Instruction Byte of Command Message
ISO	International Organization for Standardization
KD	Key Derivation
KDF	Key Derivation Function
K_M	Master Key
K_S	Session Key
L	Length
l.s.	Least Significant
Lc	Exact Length of Data Sent by the TAL in a Case 3 or 4 Command
LCOL	Lower Consecutive Offline Limit
L_{DD}	Length of the ICC Dynamic Data
Le	Maximum Length of Data Expected by the TAL in Response to a Case 2 or 4 Command
Lr	Length of Response Data Field
LRC	Longitudinal Redundancy Check
M	Mandatory
m.s.	Most Significant
MAC	Message Authentication Code
max.	Maximum
MF	Master File
MK	ICC Master Key for session key generation
MMDD	Month, Day
MMYY	Month, Year
n	Numeric (see section 4.3)
N_{CA}	Length of the Certification Authority Public Key Modulus

NF	Norme Française
N_{FIELD}	Length of a finite field element
N_{HASH}	Output length of a hash function
N_I	Length of the Issuer Public Key Modulus
N_{IC}	Length of the ICC Public Key Modulus
NIST	National Institute for Standards and Technology
N_{PE}	Length of the ICC PIN Encipherment Public Key Modulus
N_{SIG}	Length of an ECC Digital Signature
O	Optional
O/S	Operating System
ODA	Offline Data Authentication
ODE	Offline Data Encipherment
P1	Parameter 1
P2	Parameter 2
PAN	Primary Account Number
PAR	Payment Account Reference
PC	Personal Computer
P_{CA}	Certification Authority Public Key
PDOL	Processing Options Data Object List
P_I	Issuer Public Key
P_{IC}	ICC Public Key
PIN	Personal Identification Number
PIX	Proprietary Application Identifier Extension
POS	Point of Service
pos.	Position
PSE	Payment System Environment
R-APDU	Response APDU

RFU	Reserved for Future Use
RID	Registered Application Provider Identifier
RSA	Rivest, Shamir, Adleman Algorithm
S_{CA}	Certification Authority Private Key
SDA	Static Data Authentication
SDAD	Signed Dynamic Application Data
SFI	Short File Identifier
SHA-1	Secure Hash Algorithm 1
SHA-2	Secure Hash Algorithm 2 (includes SHA-256 and SHA-512)
SHA-256	Secure Hash Algorithm 256
SHA-3	Secure Hash Algorithm 3
S_I	Issuer Private Key
S_{IC}	ICC Private Key
SK	Session Key
SW1	Status Byte One
SW2	Status Byte Two
TAA	Terminal Action Analysis
TAC	Terminal Action Code(s) (Default, Denial, Online)
TAL	Terminal Application Layer
TC	Transaction Certificate
TDOL	Transaction Certificate Data Object List
TLV	Tag Length Value
TPDU	Transport Protocol Data Unit
TSI	Transaction Status Information
TVR	Terminal Verification Results
UCOL	Upper Consecutive Offline Limit
UL	Underwriters Laboratories Incorporated
UN	Unpredictable Number

var.	Variable (see section 4.3)
XDA	Extended Data Authentication
YYMM	Year, Month
YYMMDD	Year, Month, Day

4.2 Notations

'0' to '9' and 'A' to 'F'	16 hexadecimal characters
xx	Any value
A := B	A is assigned the value of B
A = B	Value of A is equal to the value of B
A ≡ B mod n	Integers A and B are congruent modulo the integer n, that is, there exists an integer d such that $(A - B) = dn$
A mod n	The reduction of the integer A modulo the integer n, that is, the unique integer r, $0 \leq r < n$, for which there exists an integer d such that $A = dn + r$
A / n	The integer division of A by n, that is, the unique integer d for which there exists an integer r, $0 \leq r < n$, such that $A = dn + r$
Y := ALG(K)[X]	Encipherment of a data block X with a block cipher as specified in Book 2 section A1, using a secret key K
$X = ALG^{-1}(K)[Y]$	Decipherment of a data block Y with a block cipher as specified in Book 2 section A1, using a secret key K
$Y := Sign\ (S_K)[X]$	The signing of a data block X with an asymmetric reversible algorithm as specified in Book 2 section A2, using the private key S_K
$X = Recover(P_K)[Y]$	The recovery of the data block X with an asymmetric reversible algorithm as specified in Book 2 section A2, using the public key P_K
C := (A || B)	The concatenation of an n-bit number A and an m-bit number B, which is defined as $C = 2^m A + B$.
Leftmost	Applies to a sequence of bits, bytes, or digits and used interchangeably with the term “most significant”. If C = (A || B) as above, then A is the leftmost n bits of C.

Rightmost	Applies to a sequence of bits, bytes, or digits and used interchangeably with the term “least significant”. If C = (A || B) as above, then B is the rightmost m bits of C.
H := Hash[MSG]	Hashing of a message MSG of arbitrary length using a 160-bit hash function
X ⊕ Y	The symbol '⊕' denotes bit-wise exclusive-OR and is defined as follows: X ⊕ Y — The bit-wise exclusive-OR of the data blocks X and Y. If one data block is shorter than the other, then it is first padded to the left with sufficient binary zeros to make it the same length as the other.
MIN (x, y)	The smaller of values x and y.

4.3 Data Element Format Conventions

The EMV specifications use the following data element formats:

a	Alphabetic data elements contain a single character per byte. The permitted characters are alphabetic only (a to z and A to Z, upper and lower case).
an	Alphanumeric data elements contain a single character per byte. The permitted characters are alphabetic (a to z and A to Z, upper and lower case) and numeric (0 to 9). There is one exception: The permitted characters for Payment Account Reference are alphabetic *upper case* (A to Z) and numeric (0 to 9).
ans	Alphanumeric Special data elements contain a single character per byte. The permitted characters and their coding are shown in the Common Character Set table in Book 4 Annex B. There is one exception: The permitted characters for Application Preferred Name are the non-control characters defined in the ISO/IEC 8859 part designated in the Issuer Code Table Index associated with the Application Preferred Name.
b	These data elements consist of either unsigned binary numbers or bit combinations that are defined elsewhere in the specification. Binary example: The Application Transaction Counter (ATC) is defined as “b” with a length of two bytes. An ATC value of 19 is stored as Hex '00 13'. Bit combination example: Processing Options Data Object List (PDOL) is defined as “b” with the format shown in Book 3 section 5.4.
cn	Compressed numeric data elements consist of two numeric digits (having values in the range Hex '0'–'9') per byte. These data elements are left justified and padded with trailing hexadecimal 'F's. Example: The Application Primary Account Number (PAN) is defined as “cn” with a length of up to ten bytes. A value of 1234567890123 may be stored in the Application PAN as Hex '12 34 56 78 90 12 3F FF' with a length of 8.
n	Numeric data elements consist of two numeric digits (having values in the range Hex '0' – '9') per byte. These digits are right justified and padded with leading hexadecimal zeroes. Other specifications sometimes refer to this data format as Binary Coded Decimal (“BCD”) or unsigned packed. Example: Amount, Authorised (Numeric) is defined as “n 12” with a length of six bytes. A value of 12345 is stored in Amount, Authorised (Numeric) as Hex '00 00 00 01 23 45'.
var.	Variable data elements are variable length and may contain any bit combination. Additional information on the formats of specific variable data elements is available elsewhere.

4.4 Terminology

business agreement	An agreement reached between a payment system and its business partner(s).
proprietary	Not defined in this specification and/or outside the scope of this specification
shall	Denotes a mandatory requirement
should	Denotes a recommendation

Part II

Removed in Version 4.4

The topics formerly addressed in Part II are included in *EMV Contact Interface Specification*.

Part III

Files, Commands, and Application Selection

10 Files

An application in the ICC includes a set of items of information, often contained within files. These items of information may be accessible to the terminal after a successful application selection.

An item of information is called a data element. A data element is the smallest piece of information that may be identified by a name, a description of logical content, a format, and a coding.

It is up to the issuer to ensure that data in the card is of the correct format.

The data element directory defined in Annex B includes those data elements that may be used for application selection. Data elements used during application selection that are not specified in Annex B are outside the scope of these specifications.

10.1 File Structure

The file organisation applying to this specification is deduced from and complies with the basic organisations as defined in ISO/IEC 7816-4.

This section describes the file structure of applications conforming to this specification.

The files within the ICC are seen from the terminal as a tree structure. Every branch of the tree is an Application Definition File (ADF) or a Directory Definition File (DDF). An ADF is the entry point to one or more Application Elementary Files (AEFs). An ADF and its related data files are seen as being on the same branch of the tree. A DDF is an entry point to AEFs, ADFs, or other DDFs.

10.1.1 Application Definition Files

The tree structure of ADFs:

- Enables the attachment of data files to an application.
- Ensures the separation between applications.
- Allows access to the logical structure of an application by its selection.

An ADF is seen from the terminal as a file containing only data objects encapsulated in its file control information (FCI) as shown in Table 10.

10.1.2 Application Elementary Files

The structure and use of AEFs is application dependent. For the EMV Debit/Credit application, the files are described in Book 3.

10.1.3 Mapping of Files onto ISO/IEC 7816-4 File Structure

The following mapping onto ISO/IEC 7816-4 applies:

- A dedicated file (DF) as defined in ISO/IEC 7816-4 and containing a FCI is mapped onto an ADF or a DDF. It may give access to elementary files and DFs. The DF at the highest level of the card is the master file (MF).
- An elementary file (EF) as defined in ISO/IEC 7816-4 is mapped onto the AEF. An EF is never used as an entry point to another file.

If DFs are embedded, retrieval of the attached EF is transparent to this specification.

10.1.4 Directory Structure

When the Payment System Environment (PSE) as described in section 12.2.2 is present, the ICC shall maintain a directory structure for the list of applications within the PSE that the issuer wants to be selected by a directory. In that case, the applications are listed in a Payment System Directory file (DIR file), the location of which is indicated in the FCI of the PSE DDF.

The directory structure allows for retrieval of an application using its ADF Name.

The location of the DIR file shall be coded in the response message to the selection of the PSE (see the SELECT command).

The DIR file is an AEF (in other words, an EF) with a record structure according to this specification including the following data objects according to ISO/IEC 7816-4:

- One or more records that each contains one or more Application Templates (tag '61') containing an ADF directory entry, that is, DF Name (see Table 11).
- Optionally, other data objects present within a Directory Discretionary Template (tag '73'). The data objects contained in this template are outside the scope of this specification.

Directories are optional within an ICC, and when present there is no defined limit to the number of such directories that may exist. Each such directory is located by a directory SFI data object contained in the FCI of each DDF.

10.2 File Referencing

A file may be referred to by a name or a SFI depending on its type.

10.2.1 Referencing by Name

Any ADF or DDF in the card is referenced by its DF Name. A DF Name for an ADF corresponds to the AID or contains the AID as the beginning of the DF Name. Each DF Name shall be unique within a given card. A DF Name shall not be a substring of another DF Name on the card.

10.2.2 Referencing by SFI

SFIs are used for the selection of AEFs. Any AEF within a given application is referenced by a SFI coded on 5 bits in the range 1 to 30. The coding of the SFI is described in every command that uses it. A SFI shall be unique within an application.

11 Commands

11.1 Message Structure

Messages are transported between the terminal and the card according to the transmission protocol selected at the ATR (see *EMV Contact Interface Specification*). The terminal and the card shall also implement the physical, data link, and transport layers as defined in *EMV Contact Interface Specification*.

To run an application, an additional layer called application protocol is implemented in the terminal. It includes steps consisting of sending a command to the card, processing it in the card, and sending back the response to the terminal. All commands and responses referred to in the remainder of this Book are defined at the application layer.

The command message sent from the application layer and the response message returned by the card to the application layer are called Application Protocol Data Units (APDU). A specific response corresponds to a specific command. These are referred to as APDU command-response pairs. In an APDU command-response pair, the command message and the response message may contain data.

This section describes the structure of the APDU command-response pairs necessary to the application protocols defined in this specification. This Book describes only those commands necessary to the functioning of application selection. All other commands shall be implemented as required by specific applications, but shall conform to the APDU structures (formats) defined in Book 3 Part II.

11.1.1 Command APDU Format

The command APDU consists of a mandatory header of four bytes followed by a conditional body of variable length, as shown in Figure 1:

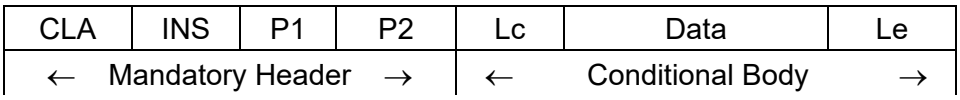

Figure 1: Command APDU Structure

The number of data bytes sent in the command APDU is denoted by Lc (length of command data field).

The maximum number of data bytes expected in the response APDU is denoted by Le (length of expected data). When Le is present and contains the value zero, the maximum number of data bytes available (≤ 256) is requested. READ RECORD and SELECT commands issued during application selection and all case 2 and case 4 commands issued according to Book 3 shall have Le = '00'.

The content of a command APDU message is as shown in Table 1:

Code	Description	Length
CLA	Class of instruction	1
INS	Instruction code	1
P1	Instruction parameter 1	1
P2	Instruction parameter 2	1
Lc	Number of bytes present in command data field	0 or 1
Data	String of data bytes sent in command (= Lc)	var.
Le	Maximum number of data bytes expected in data field of response	0 or 1

Table 1: Command APDU Content

The different cases of command APDU structure are described in *EMV Contact Interface Specification*.

11.1.2 Response APDU Format

The response APDU format consists of a conditional body of variable length followed by a mandatory trailer of two bytes, as shown in Figure 2:

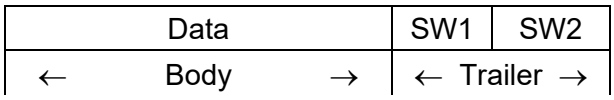

Figure 2: Response APDU Structure

The number of data bytes received in the response APDU is denoted by Lr (length of response data field). Lr is not returned by the transport layer. The application layer may rely on the object oriented structure of the response message data field to calculate Lr if needed.

The trailer indicates in two bytes the processing state of the command as returned by the transport layer.

The content of a response APDU message is as shown in Table 2:

Code	Description	Length
Data	String of data bytes received in response	var(= Lr)
SW1	Command processing status	1
SW2	Command processing qualifier	1

Table 2: Response APDU Content

11.2 READ RECORD Command-Response APDUs

11.2.1 Definition and Scope

The READ RECORD command reads a file record in a linear file.

The response from the ICC consists of returning the record.

11.2.2 Command Message

The READ RECORD command message is coded according to Table 3:

Code	Value
CLA	'00'
INS	'B2'
P1	Record number
P2	Reference control parameter (see Table 4)
Lc	Not present
Data	Not present
Le	'00'

Table 3: READ RECORD Command Message

Table 4 defines the reference control parameter of the command message:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x	x				SFI
					1	0	0	P1 is a record number

Table 4: READ RECORD Command Reference Control Parameter

11.2.3 Data Field Sent in the Command Message

The data field of the command message is not present.

11.2.4 Data Field Returned in the Response Message

The data field of the response message of any successful READ RECORD command contains the record read. Records read during application selection are directory records which are formatted as in section 12.2.3. The format of records read during application processing is application dependent.

11.2.5 Processing State Returned in the Response Message

'9000' indicates a successful execution of the command.

11.3 SELECT Command-Response APDUs

11.3.1 Definition and Scope

The SELECT command is used to select the PSE, a DDF, or an ADF corresponding to the submitted file name or AID. The selection of an application is described in section 12.

A successful execution of the command sets the path to the PSE, DDF, or ADF.

Subsequent commands apply to AEFs associated with the selected PSE, DDF, or ADF using SFIs.

The response from the ICC consists of returning the FCI.

11.3.2 Command Message

The SELECT command message is coded according to Table 5:

Code	Value
CLA	'00'
INS	'A4'
P1	Reference control parameter (see Table 6)
P2	Selection options (see Table 7)
Lc	'05'–'10'
Data	File name
Le	'00'

Table 5: SELECT Command Message

Table 6 defines the reference control parameter of the SELECT command message:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0	0				
					1			Select by name
						0	0	

Table 6: SELECT Command Reference Control Parameter

Table 7 defines the selection options P2 of the SELECT command message:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
						0	0	First or only occurrence
						1	0	Next occurrence

Table 7: SELECT Command Options Parameter

11.3.3 Data Field Sent in the Command Message

The data field of the command message contains the PSE name or the AID to be selected. During final selection of the application to be used, the data field shall be the DF Name if List of AIDs selection has been used, or ADF Name if PSE selection has been used.

11.3.4 Data Field Returned in the Response Message

The data field of the response message contains the FCI specific to the selected PSE, DDF, or ADF. The tags defined in Table 8, Table 9, and Table 10 apply to this specification. No additional data elements shall be present in the FCI template (tag '6F') returned in the response to the SELECT command other than those contained in template 'BF0C'. Padding is not allowed within the FCI returned by the card. Terminals shall accept and correctly parse an FCI containing padding unless the FCI would be rejected due to other errors.

Data elements present in templates '6F' and/or 'BF0C' that are not expected or understood by the terminal because the terminal does not support any issuer-specific processing shall be ignored.

Table 8 defines the FCI returned by a successful selection of the PSE:

Tag	Value			Presence
'6F'	FCI Template			M
	'84'	DF Name		M
	'A5'	FCI Proprietary Template		M
		'88'	SFI of the Directory Elementary File	M
		'5F2D'	Language Preference	O
		'9F11'	Issuer Code Table Index	O
		'BF0C'	FCI Issuer Discretionary Data	O
			'XXXX' (Tag constructed according to Book 3 Annex B)	1 or more additional proprietary data elements from an application provider, issuer, or IC card supplier, or EMV-defined tags that are specifically allocated to 'BF0C'	O

Table 8: SELECT Response Message Data Field (FCI) of the PSE

Table 9 defines the FCI returned by a successful selection of a DDF:

Tag	Value			Presence
'6F'	FCI Template			M
	'84'	DF Name		M
	'A5'	FCI Proprietary Template		M
		'88'	SFI of the Directory Elementary File	M
		'BF0C'	FCI Issuer Discretionary Data	O
			'XXXX' (Tag constructed according to Book 3 Annex B)	1 or more additional proprietary data elements from an application provider, issuer, or IC card supplier, or EMV-defined tags that are specifically allocated to 'BF0C'	O

Table 9: SELECT Response Message Data Field (FCI) of a DDF

Table 10 defines the FCI returned by a successful selection of an ADF:

Tag	Value				Presence
'6F'	FCI Template				M
	'84'	DF Name			M
	'A5'	FCI Proprietary Template			M
		'50'	Application Label		M
		'87'	Application Priority Indicator		O
		'9F38'	PDOL		O
		'5F2D'	Language Preference		O
		'9F11'	Issuer Code Table Index		O
		'9F12'	Application Preferred Name		O
		'BF0C'	FCI Issuer Discretionary Data		O
			'9F4D'	Log Entry	O
			'9F0A'	Application Selection Registered Proprietary Data	O
			'XXXX' (Tag constructed according to Book 3 Annex B)	1 or more additional proprietary data elements from an application provider, issuer, or IC card supplier, or EMV-defined tags that are specifically allocated to 'BF0C'	O

Table 10: SELECT Response Message Data Field (FCI) of an ADF

11.3.5 Processing State Returned in the Response Message

'9000' indicates a successful execution of the command.

ICC support for the selection of a DF using only a partial DF Name is not mandatory. However, if the ICC does support partial name selection, it shall comply with the following:

- If, after a DF has been successfully selected, the terminal repeats the SELECT command having P2 set to the Next Occurrence option (see Table 7) and with the same partial DF Name, the card shall select a different DF matching the partial name, if another such DF exists.
- Repeated issuing of the same command with no intervening application level commands shall retrieve all such files, but shall retrieve no file twice.
- After all matching DFs have been selected, repeating the same command again shall result in no file being selected, and the card shall respond with SW1 SW2 = '6A82' (file not found).

12 Application Selection

12.1 Overview of Application Selection

During an EMV card session as defined in *EMV Contact Interface Specification*, application selection using the commands and techniques described in sections 11 and 12 shall be the first process performed immediately after contact activation/reset of the card and prior to the first application function. If a proprietary processing session (including any proprietary application selection method) is performed immediately before or after an EMV card session, there is no requirement to remove/reinsert the card between the sessions. However, if proprietary processing occurs before the EMV card session, the card contacts shall be deactivated before starting the EMV card session.

This section describes the application selection process from the standpoint of both the card and the terminal. It specifies the logical structure of data and files within the card that are required for the process, and then describes the terminal logic using the card structure.

It is not recommended that the ICC and the terminal use implicit selection as defined in ISO 7816, as it is not useful in an interchange environment. If used, it shall be performed outside the EMV card session as defined in *EMV Contact Interface Specification*.

The application selection process described in this section is the process by which the terminal uses data in the ICC according to protocols defined herein to determine the terminal program and the ICC application to be used in processing a transaction. The process is described in two steps:

1. Create a list of ICC applications that are supported by the terminal. (This list is referred to below using the name ‘candidate list’.) This process is described in section 12.3.
2. Select the application to be run from the list generated above. This process is described in section 12.4.

This section of the specification describes the necessary information in the card and two terminal selection algorithms that yield the correct results. Other terminal selection algorithms that yield the same results are permitted in place of the selection algorithms described here.

A payment system application is comprised of the following:

- A set of files in the ICC providing data customised by the issuer
- Data in the terminal provided by the acquirer or the merchant
- An application protocol agreed upon by both the ICC and the terminal

Applications supported by the terminal are identified by AIDs conforming to ISO/IEC 7816-4. Applications supported by the ICC are identified by DF Names conforming to ISO/IEC 7816-4. For further details see section 12.2.1 below).

The techniques chosen by the payment systems and described herein are designed to meet the following key objectives:

- Ability to work with ICCs with a wide range of capabilities.
- Ability for terminals with a wide range of capabilities to work with all ICCs supporting payment system applications according to this specification.
- Conformance with ISO standards.
- Ability of ICCs to support multiple applications, not all of which need to be payment system applications.
- Ability for ICCs to provide multiple sets of applications to be supported by a single terminal program. (For example, a card may contain multiple credit/debit applications, each representing a different type or level of service or a different account).
- As far as possible, provide the capability for applications conforming with this specification to co-reside on cards with presently existing applications.
- Minimum overhead in storage and processing.
- Ability for the issuer to optimise the selection process.

The set of data that the ICC contains in support of a given application is defined by an ADF selected by the terminal using a SELECT command and an Application File Locator (AFL) returned by the ICC in response to a GET PROCESSING OPTIONS command.

12.2 Data in the ICC Used for Application Selection

12.2.1 Coding of Payment System Application Identifier

The structure of AIDs, ADF Names and DF Names is according to ISO/IEC 7816-4 and consists of two parts:

1. A Registered Application Provider Identifier (RID) of 5 bytes, unique to an application provider and assigned according to ISO/IEC 7816-4.
2. An optional field assigned by the application provider of up to 11 bytes. This field is known as a Proprietary Application Identifier Extension (PIX) and may contain any 0–11 byte value specified by the provider. The meaning of this field is defined only for the specific RID and need not be unique across different RIDs.

Additional ADFs defined under the control of other application providers may be present in the ICC but shall avoid duplicating the range of RIDs assigned to payment systems. Compliance with ISO/IEC 7816-4 will assure this avoidance.

12.2.2 Structure of the PSE

The PSE is accessed via a DDF with the name '1PAY.SYS.DDF01'. The presence of this DDF in the ICC is optional but, if present, it shall comply with this specification. If it is present, this DDF is mapped onto a DF within the card, which may or may not be the MF, and shall contain a Payment System Directory. The FCI of this DDF shall contain at least the information defined for all DDFs in section 11 and, optionally, the Language Preference (tag '5F2D') and the Issuer Code Table Index (tag '9F11').

The Language Preference and Issuer Code Table Index are optional data objects that may occur in two places: the FCI of the PSE and the FCI of ADF files. If either of these data elements is present in one location but not the other, the terminal shall use the data element that is present. If either data element is present in both locations but has different values in the two locations, the terminal may use either value.[1]

The directory attached to the PSE DDF contains entries for ADFs that are formatted according to this specification, although the applications defined by those ADFs may or may not conform to this specification.

The directory is not required to have entries for all ADFs in the card. However, if the PSE exists, only applications that are revealed by reading the directory can be assured of international interoperability.

See Annex C for examples of the internal logic structure of an ICC containing the PSE.

[1] A terminal building a candidate list using the process described in section 12.3.2 will encounter the values specified in the FCI of the PSE and will not see the values specified in the FCI of the ADF until the application to be run has been chosen. A terminal building the candidate list using the process described in section 12.3.3 will encounter the values specified in the FCI of the ADFs. To ensure consistent interface to the cardholder, the values must be the same.

12.2.3 Coding of a Payment System Directory

A Payment System Directory is a linear EF file identified by a SFI in the range 1 to 10. The SFI for the Payment System Directory AEF is contained in the FCI of the DDF to which the directory is attached. The Payment System Directory is read using the READ RECORD command as defined in section 11.

A record may have several directory entries, but a directory entry shall always be encapsulated in its entirety in a single record. Each record in the Payment System Directory is a constructed data object, and the value field is comprised of one or more directory entries as described below in Table 11:

Tag '70'	Data Length (L)	Tag '61'	Length of directory entry 1	Directory entry 1 (ADF)	...	Tag '61'	Length of directory entry n	Directory entry n (ADF)

Table 11: Payment System Directory Record Format

Payment Systems Directory records shall not contain any entries for DDFs. If the terminal encounters a directory entry for a DDF in one of these records, it may ignore it or may optionally process the DDF, but any such processing is outside the scope of EMV.

Each entry in a Payment System Directory is the value field of an Application Template (tag '61') and contains the information according to Table 12. No additional data elements shall be present in the Payment System Directory Record (tag '70') other than those contained in template '73'.

Data elements present in the Payment System Directory Record, template '61', or template '73' that are not expected or understood by the terminal because the terminal does not support any issuer-specific processing, shall be ignored.

Tag	Length	Value		Presence
'4F'	5–16	ADF Name		M
'50'	1–16	Application Label		M
'9F12'	1–16	Application Preferred Name		O
'87'	1	Application Priority Indicator (see Table 13)		O
'73'	var.	Directory Discretionary Template		O [2]
	'9F0A'	var.	Application Selection Registered Proprietary Data	O
	'XXXX' (Tag constructed according to Book 3 Annex B)	var.	1 or more additional proprietary data elements from an application provider, issuer, or IC card supplier, or EMV-defined tags that are specifically allocated to template '73'	O

Table 12: ADF Directory Entry Format

b8	b7–b5	b4–b1	Definition
1			Application cannot be selected without confirmation by the cardholder
0			Application may be selected without confirmation by the cardholder
	xxx		RFU
		0000	No priority assigned
		xxxx (except 0000)	Order in which the application is to be listed or selected, ranging from 1–15, with 1 being highest priority

Table 13: Format of Application Priority Indicator

[2] Other data objects not relevant to this specification may appear in this constructed data object.

12.2.4 Error Handling for FCI Response Data

The data elements Application Label, Application Preferred Name, Issuer Code Table Index, and Language Preference are present for the convenience of the cardholder and are not critical to the successful processing of application selection. If these data elements are present in the FCI, the issuer is responsible for their correct encoding.

If the Application Label data element is not present in the FCI of an ADF, the terminal shall not terminate the card session but shall proceed with application selection.

Terminals shall not enforce the correct formatting of these data elements. If Application Preferred Name or Application Label contains a character that is not valid for the defined format, the terminal shall display the character if it is able to, or if the terminal is unable to display the invalid character, it should omit the character or substitute a space character or any other appropriate character. Otherwise, if the terminal detects format errors in any of these data elements, the terminal shall disregard these errors and act as if the response provided by the card did not contain these data elements. More specifically, the terminal shall not terminate the card session but shall proceed with application selection.

If the terminal does not understand the value in Issuer Code Table Index or Language Preference, it shall treat the data element as not present.

12.3 Building the Candidate List

A terminal shall always support application selection using the List of AIDs method as described in section 12.3.3. A terminal may additionally support application selection using the PSE method as described in section 12.3.2. If the card contains no PSE, the procedure described in section 12.3.3 must be followed.

The terminal may know other ways that are not described in this section to locate proprietary applications. This is permitted as long as all interoperable applications can be located in the ICC using the techniques described here.

If allowed by business agreement between the affected parties, specific applications may be eliminated from consideration either during or after building the candidate list.

12.3.1 Matching Terminal Applications to ICC Applications

The terminal shall maintain a list of applications supported by the terminal identified by their AIDs. The terminal determines which applications in the ICC are supported by comparing the AIDs for applications supported by the terminal with the DF Names[3] of applications supported by the ICC.

For each of the AIDs within the list of applications supported by the terminal, the terminal shall use the Application Selection Indicator (ASI) as an indicator of the matching criterion to use.

- If the ASI indicates that the terminal supports the ICC application only when the AID in the terminal has the same length and value as the DF Name then this limits the ICC to at most one matching ADF.
- If the ASI indicates that the terminal supports the ICC application when the DF Name begins with the entire AID kept within the terminal then this allows the ICC to have multiple ADFs matching the terminal AID by adding unique information to the DF Name used by each of the ADFs.

If the ICC does not support partial name selection as described in section 11.3.5, the DF Name of the ADF must be an exact match with the terminal AID.

If the ICC supports partial name selection as described in section 11.3.5 and has multiple ADFs supported by a single terminal AID, all of the matching DF Names must be distinguished by adding unique data to the PIX. All of the matching DF Names shall be longer than the corresponding terminal AID.

[3] Depending upon the method used to build the candidate list, the names in the list will be ADF Names found in directory entries if the PSE selection method is used or DF Names found in the FCIs returned to SELECT commands if the List of AIDs method is used. For readability in this section, the term DF Name is used in place of either.

12.3.2 Using the PSE

If a terminal chooses to support application selection using the PSE method, it shall follow the procedure described in this section to determine the applications supported by the card. Figure 3 is a flow diagram of the logic described here.

The terminal performs the following steps:

1. The terminal begins by selecting the PSE using a SELECT command as described in section 11 using a file name of '1PAY.SYS.DDF01'. This establishes the PSE and makes the Payment System Directory accessible.

   If the card is blocked or the SELECT command is not supported (both conditions represented by SW1 SW2 = '6A81'), the terminal terminates the session.

   If there is no PSE in the ICC, the ICC shall return '6A82' ('File not found') in response to the SELECT command for the PSE. In this case, the terminal shall use the List of AIDs method described in section 12.3.3.

   If the PSE is blocked, the ICC shall return '6283'. In this case, the terminal shall use the List of AIDs method described in section 12.3.3.

   If the ICC returns SW1 SW2 = '9000', the terminal proceeds to step 2.

   If the card returns any other value in SW1 SW2, the terminal shall use the List of AIDs method described in section 12.3.3.

   If any error, including a SW1 SW2 different from '90 00' or '6A 83', occurs in steps 2 through 4, the terminal shall clear the candidate list and restart the application selection process using the List of AIDs method described in section 12.3.3 to find the matching applications.

2. The terminal uses the Directory SFI from the FCI returned and reads all the records in the Payment System Directory beginning with record number 1 and continuing with successive records until the card returns SW1 SW2 = '6A83', which indicates that the record number requested does not exist. (The card shall return '6A83' if the record number in the READ RECORD command is greater than the number of the last record in the file). If the card returns SW1 SW2 = '6A83' in response to a READ RECORD for record number 1 for the Payment System Directory, no directory entries exist, and step 5 (below) applies.

   For each record in the Payment System Directory, the terminal begins with the first directory entry and processes each directory entry in turn as described in steps 3 and 4. If there are no directory entries in the record, the terminal proceeds to the next directory record.

3. If the ADF Name in the directory entry matches one of the applications supported by the terminal as defined in section 12.3.1, the application joins the candidate list for final application selection under control of the ASI maintained in the terminal for that AID.

   The ASI indicates whether the AID in the terminal shall match exactly (both in length and name) or need only partially match the associated ADF Name in the directory entry (tag '4F').

   The application is added to the candidate list in either of the following cases:

   - the ADF Name in the directory entry retrieved is an exact match, or
   - the ASI for the AID in the terminal indicates that a partial match is allowed.

   The application is not added to the candidate list if the ADF Name in the directory entry retrieved is not an exact match and the ASI for the AID in the terminal indicates that an exact match is required.

4. When the terminal finishes processing all entries in the last record of the Payment System Directory, all ADFs that can be found by this procedure have been determined. The search and the candidate list are complete. If at least one matching ADF Name was found, the terminal continues processing as described in section 12.4.

5. If steps 1 through 4 yield no directory entries that match applications supported by the terminal, the terminal shall use the list of AIDs method described in section 12.3.3 to find a match.

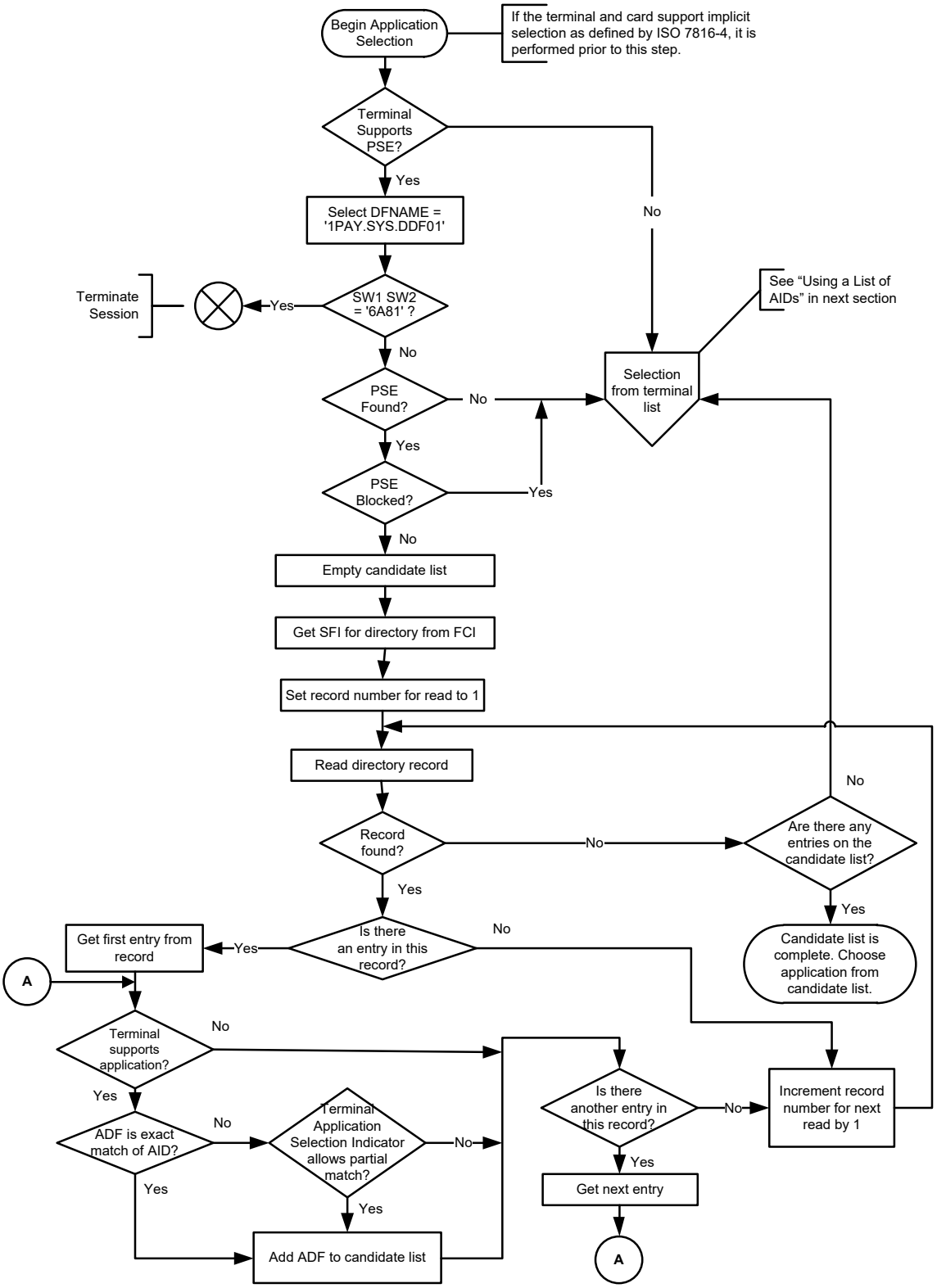

Figure 3: Terminal Logic Using Directories

12.3.3 Using a List of AIDs

If either the card or the terminal does not support the PSE method or if the terminal is unable to find a matching application using the Payment System Directory selection method, the terminal shall use a list of AIDs that it supports to build the candidate list. Figure 4 is a flow diagram of the logic described here.

The terminal performs the following steps:

1. The terminal issues a SELECT command using the first AID[4] in the terminal list as the file name.
2. If the SELECT command fails because the card is blocked or the command is not supported by the ICC (SW1 SW2 = '6A81'), the terminal terminates the card session.
3. If the SELECT command is successful (SW1 SW2 = '9000' or '6283'), the terminal compares the AID with the DF Name field returned in the FCI. The DF Name will either be identical to the AID (including the length), or the DF Name will start with the AID but will be longer. If the names are identical, the terminal proceeds with step 4. If the DF Name is longer, the card processed the command as a partial name selection, and the terminal proceeds to step 6.

   If the card returns any other status or if mandatory data is missing from the SELECT response or if the FCI contains formatting errors not described in Section 12.2.4, the terminal proceeds to Step 5 without adding the DF Name to the candidate list.
4. If the SELECT command is successful (SW1 SW2 = '9000'), the terminal adds the DF Name and other information[5] from the FCI to the candidate list and proceeds to step 5. If the application is blocked (SW1 SW2 = '6283'), the terminal proceeds to step 5 without adding the DF Name to the candidate list.
5. The terminal issues another SELECT command using the next AID in its list and returns to step 3. If there are no more AIDs in the list, the candidate list is complete, and the terminal proceeds as specified in section 12.4.
6. Along with the AID list, the terminal keeps an Application Selection Indicator that indicates whether the card may have multiple occurrences of the application within the card. The terminal checks this indicator. If the indicator says that an exact match (in both length and name) is required, the terminal does not add the DF Name and other information from the FCI to the candidate list, but proceeds to step 5.

[4] To assist in a clear understanding of the process described in this section, it is necessary to distinguish between the application identifier kept in the terminal and the application identifier kept in the ICC. As can be seen in section 12.3.1, these might not be identical even for matching applications. In this procedure, the term AID is used for the application identifier kept in the terminal, and DF Name is used for the application identifier in the card.

[5] The Application Label and Application Preferred Name must also be saved if the cardholder will be provided a list during final selection. The DF Name and the Application Priority Indicator will be required in any case.

If multiple occurrences are permitted, the partial name match is sufficient. If the application is not blocked (SW1 SW2 = '9000'), the terminal adds the DF Name and other information from the FCI to the candidate list and proceeds to step 7.

If multiple occurrences are permitted but the application is blocked (SW1 SW2 ≠ '9000'), the terminal proceeds to step 7 without adding the DF Name or other information from the FCI to the candidate list.

7. The terminal repeats the SELECT command using the same command data as before, but changes P2 in the command to '02' (Select Next). If the ICC returns SW1 SW2 = '9000', '62xx', or '63xx', the terminal returns to step 3. If it returns a different SW1 SW2, the terminal goes to step 5.

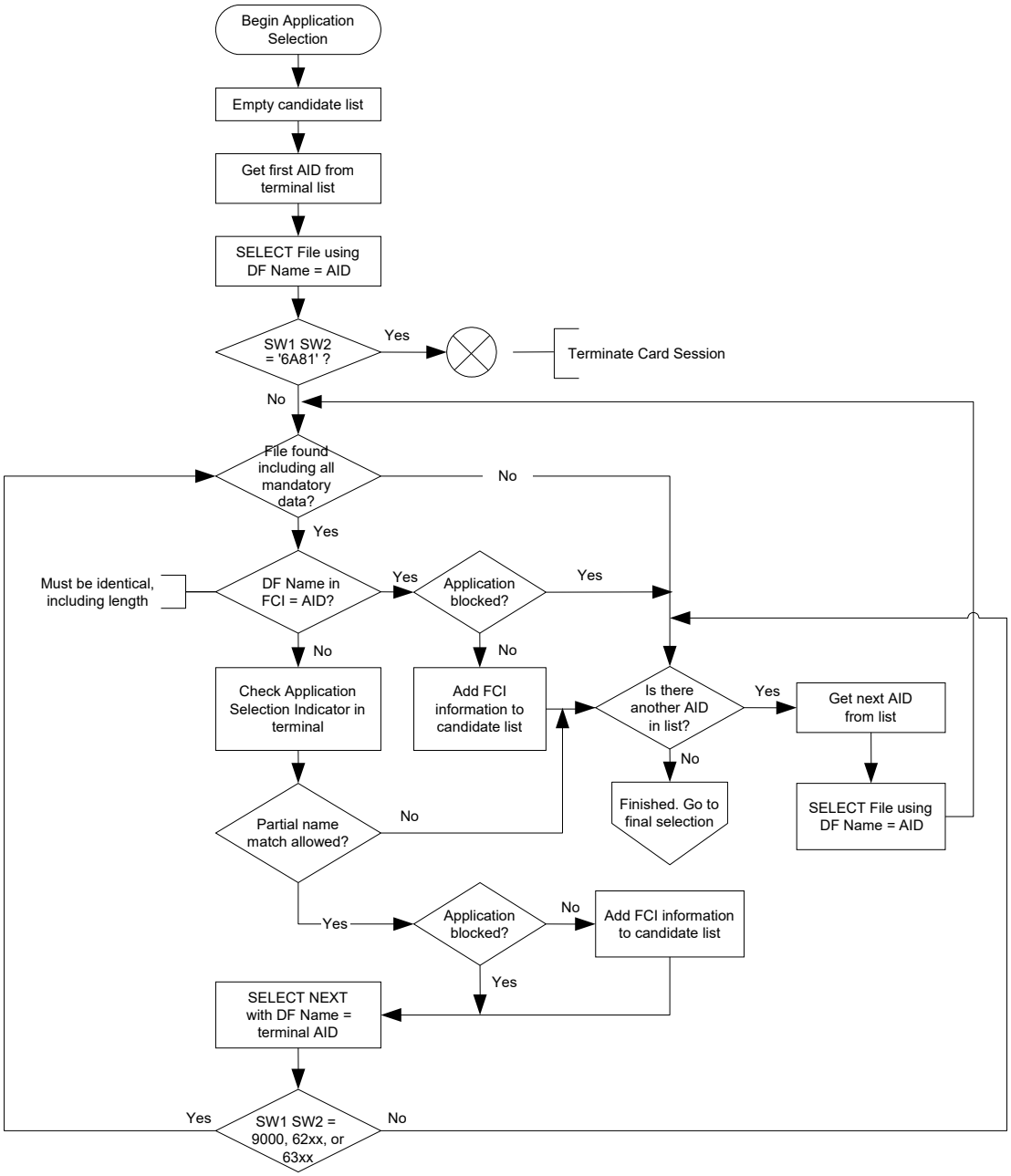

Figure 4: Using the List of AIDs in the Terminal

12.4 Final Selection

The terminal should support the ability to allow the cardholder to select an application or to confirm the application proposed by the terminal.

- Cardholder selection is a function that allows the cardholder to choose the application they want to use when two or more applications are mutually supported by the card and terminal.
- Cardholder confirmation is a mechanism that allows issuers/cardholders the ability to identify applications on the card that the terminal shall not be allowed to select automatically without explicit cardholder confirmation of some kind.

These two features are covered by a single implementation or configuration option. These specifications allow for the following:

- The terminal supports both cardholder selection and cardholder confirmation.
- The terminal supports neither cardholder selection nor cardholder confirmation.

When the terminal displays an application to the cardholder, it shall display:

- the Application Preferred Name, if present and if the Issuer Code Table Index indicating the part of ISO/IEC 8859 to use is present and supported by the terminal (as indicated in Additional Terminal Capabilities)
- otherwise the Application Label, if present, by using the common character set of ISO/IEC 8859 (see Book 4 Annex B)

Once the terminal determines the list of mutually supported applications, it proceeds as follows:

1. If there are no mutually supported applications, the transaction is terminated.
2. If there is only one mutually supported application, the terminal checks b8 of the card's Application Priority Indicator for that application if present.
    - If b8 = 0, the terminal selects the application.
    - If b8 = 1 and the terminal provides for confirmation by the cardholder, the terminal requests confirmation and selects the application if the cardholder approves. If the terminal does not provide for confirmation by the cardholder, or if the terminal requests confirmation and the cardholder does not approve, the terminal terminates the session.
3. If there is more than one mutually supported application, then:
    - If the terminal supports the ability to allow the cardholder to select an application the terminal shall offer a selection to the cardholder as described in step 4.
    - If the terminal does not support the ability to allow the cardholder to select an application, the terminal makes the selection itself as described in step 5.

    Step 4 is the preferred method.
4. When the terminal offers a selection to the cardholder, then:
    - Applications where the card's Application Priority Indicator is present shall be presented in priority sequence as indicated by the Application Priority Indicator with the highest priority application offered first. Where the same priority is assigned to multiple applications, the terminal may present these applications in its own preferred order or in the order encountered in the card.
    - For applications where the card's Application Priority Indicator is not present, the terminal may present these applications in its own preferred order or in the order encountered in the card.
5. When the terminal does not offer a selection to the cardholder, then the terminal shall select the highest priority application that does not require cardholder confirmation (that is, the card's Application Priority Indicator b8 = 0) from the list of mutually supported applications. The application's priority is determined using the same rules as described in step 4.

Once the application to be run is determined by the terminal or by the cardholder, the application shall be selected. A SELECT command coded according to section 11 shall be issued by the terminal for the application using the ADF Name field (if a directory was read) or the DF Name field from the FCI (if the List of AIDs method was used) found during the building of the candidate list.

On successful selection of the application to be used (SW1 SW2 = '9000' returned in response to the final SELECT command, the response contains no format errors other than those described in section 12.2.4, and the AID used in the final SELECT command exactly matches the DF Name (tag '84') returned by the ICC in the FCI), the terminal shall set the value of the terminal data element Application Identifier (AID) – terminal (tag '9F06') to the same value as the DF Name (tag '84') returned in the FCI. If transaction processing is to be continued according to Book 3, this shall be done prior to issuance of the GET PROCESSING OPTIONS command.

If the command returns other than '9000' in SW1 SW2 or the SELECT response contains format errors other than those described in section 12.2.4, the application shall be removed from the candidate list, and processing shall resume at step 1.

If the cardholder selects or confirms the selection of an application that is subsequently removed from the candidate list due to its being blocked or for any other reason, no application is to be selected without cardholder confirmation.

If no application can be selected, the terminal shall terminate the transaction.

In any case, the terminal shall inform the cardholder of the action taken (that is, by using the messages defined in Book 4 section 11.2), if appropriate.

12.5 Application Selection Registered Proprietary Data

Usage of the Application Selection Registered Proprietary Data (ASRPD) received from the ICC is optional and proprietary.

If Application Selection processing does not use ASRPD,
then Application Selection processing shall ignore instances of '9F0A' in Directory Entries (Tag '61') and continue processing as if the data was not present.

If Application Selection processing uses ASRPD,
then Application Selection processing interprets the value field to recover all the Proprietary Data Identifiers:

- **If** the value field of the ASRPD is not correctly formatted (ID L V, ID L V, … as defined below),
  then the Application Selection processing shall ignore this instance of the ASRPD and continue processing as if the data was not present. Note that no assumption can be made on the IDs already registered by EMVCo nor on the format of the value fields of the Proprietary Data Identifiers and as a consequence the value field of the ASRPD is considered to be incorrectly formatted only if a length problem is detected.
- **If** the value field of the ASRPD is correctly formatted,
  then proprietary functionality may be activated for the recognised Proprietary Data Identifiers.
- Application Selection processing is not required to keep track of the Proprietary Data Identifiers defined by EMVCo, therefore unrecognised Proprietary Data Identifiers shall be ignored.

The coding of the ASRPD is as follows:

The value field of the ASRPD object follows the following format:
ID1, L1, V1, ID2, L2, V2,…

Where

- ID is a two byte Proprietary Data Identifier. Proprietary Data Identifiers are registered by EMVCo.
- L is the length of the value field coded in 1 byte (0 to 255).
- V is the value field. Its content is proprietary and format is out of scope of EMVCo.

Part IV

Annexes

Annex A Removed in Version 4.4

Annex A was deleted when Part II was removed in Version 4.4.

Annex B Data Elements Table

Table 14 defines those data elements that may be used for application selection and their mapping onto data objects and files.[6] Table 15 lists the data elements in tag sequence.

The characters used in the "Format" column are described in section 4.3, Data Element Format Convention.

B1 Data Elements by Name

Table 14: Data Elements Table

Name	Description		Source	Format	Template	Tag	Length
Application Dedicated File (ADF) Name	Identifies the application as described in ISO/IEC 7816-4		ICC	b	'61'	'4F'	5–16
Application Identifier (AID) - terminal	Identifies the application as described in ISO/IEC 7816-4		Terminal	b	—	'9F06'	5–16

[6] Book 3 Annex A provides a complete data elements table, defining all data elements that may be used for financial transaction interchange and their mapping onto data objects and files.

Name	Description		Source	Format	Template	Tag	Length
Application Label	Mnemonic associated with the AID according to ISO/IEC 7816-4		ICC	ans with the special character limited to space	'61' or 'A5'	'50'	1–16
Application Preferred Name	Preferred mnemonic associated with the AID		ICC	ans (see section 4.3)	'61' or 'A5'	'9F12'	1–16
Application Priority Indicator	Indicates the priority of a given application or group of applications in a directory		ICC	b	'61' or 'A5'	'87'	1
Application Selection Indicator	For an application in the ICC to be supported by an application in the terminal, the Application Selection Indicator indicates whether the associated AID in the terminal must match the AID in the card exactly, including the length of the AID, or only up to the length of the AID in the terminal There is only one Application Selection Indicator per AID supported by the terminal		Terminal	At the discretion of the terminal. The data is not sent across the interface	—	—	See Format

Name	Description		Source	Format	Template	Tag	Length
Application Selection Registered Proprietary Data (ASRPD)	Proprietary data allowing for proprietary processing during application selection. Proprietary data is identified using Proprietary Data Identifiers that are managed by EMVCo and their usage by the Application Selection processing is according to their intended usage, as agreed by EMVCo during registration		Card	b, also see section 12.5	'73' 'BF0C'	'9F0A'	var.
Application Template	Contains one or more data objects relevant to an application directory entry according to ISO/IEC 7816-4		ICC	b	'70'	'61'	var. up to 252
Bank Identifier Code (BIC)	Uniquely identifies a bank as defined in ISO 9362.		ICC	var.	'BF0C' or '73'	'5F54'	8 or 11
Dedicated File (DF) Name	Identifies the name of the DF as described in ISO/IEC 7816-4		ICC	b	'6F'	'84'	5–16
Directory Definition File (DDF) Name	Identifies the name of a DF associated with a directory		ICC	b	'61'	'9D'	5–16
Directory Discretionary Template	Issuer discretionary part of the directory according to ISO/IEC 7816-4		ICC	var.	'61'	'73'	var. up to 252
File Control Information (FCI) Issuer Discretionary Data	Issuer discretionary part of the FCI		ICC	var.	'A5'	'BF0C'	var. up to 222

Name	Description		Source	Format	Template	Tag	Length
File Control Information (FCI) Proprietary Template	Identifies the data object proprietary to this specification in the FCI template according to ISO/IEC 7816-4		ICC	var.	'6F'	'A5'	var.
File Control Information (FCI) Template	Identifies the FCI template according to ISO/IEC 7816-4		ICC	var.	—	'6F'	var. up to 252
International Bank Account Number (IBAN)	Uniquely identifies the account of a customer at a financial institution as defined in ISO 13616.		ICC	var.	'BF0C' or '73'	'5F53'	Var. up to 34
Issuer Code Table Index	Indicates the code table according to ISO/IEC 8859 for displaying the Application Preferred Name		ICC	n 2	'A5'	'9F11'	1
Issuer Country Code (alpha2 format)	Indicates the country of the issuer as defined in ISO 3166 (using a 2 character alphabetic code)		ICC	a 2	'BF0C' or '73'	'5F55'	2
Issuer Country Code (alpha3 format)	Indicates the country of the issuer as defined in ISO 3166 (using a 3 character alphabetic code)		ICC	a 3	'BF0C' or '73'	'5F56'	3
Issuer Identification Number (IIN)	The number that identifies the major industry and the card issuer and that forms the first part of the Primary Account Number (PAN)		ICC	n 6	'BF0C' or '73'	'42'	3

Name	Description		Source	Format	Template	Tag	Length
Issuer Identification Number Extended (IINE)	The number that identifies the major industry and the card issuer and that forms the first part (6 or 8-digits) of the Primary Account Number (PAN). While the first 6-digits of the IINE (tag '9F0C') and IIN (tag '42') are the same and there is no need to have both data objects on the card, cards may have both the IIN and IINE data objects present.		ICC	n 6 or 8	'BF0C' or '73'	'9F0C'	var. 3 or 4
Issuer URL	The URL provides the location of the issuer's Library Server on the Internet		ICC	ans	'BF0C' or '73'	'5F50'	var.
Language Preference	1–4 languages stored in order of preference, each represented by 2 alphabetical characters according to ISO 639 **Note:** EMVCo strongly recommends that cards be personalised with data element '5F2D' coded in lowercase, but that terminals accept the data element whether it is coded in upper or lower case.		ICC	an 2	'A5'	'5F2D'	2–8
Log Entry	Provides the SFI of the Transaction Log file and its number of records		ICC	b	'BF0C' or '73'	'9F4D'	2

Name	Description		Source	Format	Template	Tag	Length
Processing Options Data Object List (PDOL)	Contains a list of terminal resident data objects (tags and lengths) needed by the ICC in processing the GET PROCESSING OPTIONS command		ICC	b	'A5'	'9F38'	var.
READ RECORD Response Message Template	Contains the contents of the record read. (Mandatory for SFIs 1-10. Response messages for SFIs 11-30 are outside the scope of EMV, but may use template '70'.)		ICC	var.	—	'70'	var. up to 252
Short File Identifier (SFI)	Identifies the AEF referenced in commands related to a given ADF or DDF. It is a binary data object having a value in the range 1 – 30 and with the three high order bits set to zero.		ICC	b	'A5'	'88'	1

When the length defined for the data object is greater than the length of the actual data, the following rules apply:

- A data element in format n is right justified and padded with leading hexadecimal zeroes.
- A data element in format a, an, or ans is left justified and padded with trailing hexadecimal zeroes.

When data is moved from one entity to another (for example, card to terminal), it shall always be passed in order from high order to low order, regardless of how it is internally stored. The same rule applies when concatenating data.

B2 Data Elements by Tag

Name	Template	Tag
Issuer Identification Number (IIN)	'BF0C' or '73'	'42'
Application Dedicated File (ADF) Name	'61'	'4F'
Application Label	'61' or 'A5'	'50'
Language Preference	'A5'	'5F2D'
Issuer URL	'BF0C' or '73'	'5F50'
International Bank Account Number (IBAN)	'BF0C' or '73'	'5F53'
Bank Identifier Code (BIC)	'BF0C' or '73'	'5F54'
Issuer Country Code (alpha2 format)	'BF0C' or '73'	'5F55'
Issuer Country Code (alpha3 format)	'BF0C' or '73'	'5F56'
Application Template	'70'	'61'
File Control Information (FCI) Template	—	'6F'
READ RECORD Response Message Template	—	'70'
Directory Discretionary Template	'61'	'73'
Dedicated File (DF) Name	'6F'	'84'
Application Priority Indicator	'61' or 'A5'	'87'
Short File Identifier (SFI)	'A5'	'88'
Directory Definition File (DDF) Name	'61'	'9D'
Application Identifier (AID) - terminal	—	'9F06'
Application Selection Registered Proprietary Data (ASRPD)	'73'	'9F0A'
Issuer Identification Number Extended (IINE)	'BF0C' or '73'	'9F0C'
Issuer Code Table Index	'A5'	'9F11'
Application Preferred Name	'61' or 'A5'	'9F12'
Processing Options Data Object List (PDOL)	'A5'	'9F38'
Log Entry	'BF0C' or '73'	'9F4D'
File Control Information (FCI) Proprietary Template	'6F'	'A5'
File Control Information (FCI) Issuer Discretionary Data	'A5'	'BF0C'

Table 15: Data Elements Tags

Annex C Examples of Directory Structures

This annex illustrates some possible logical ICC file structures.

C1 Single Application Card

Figure 5 illustrates a single application card with only a single level directory. In this example, the MF (with file identification of '3F00', as defined by ISO/IEC 7816-4) acts as the only DDF in the card. The MF shall be given the unique payment system's name assigned to the first level DDF as defined in section 12.2, and the FCI of the MF shall contain the SFI data object.

'DIR A' in this example may or may not be the ISO DIR file, but it shall conform to this specification, including the requirement that it has a SFI in the range 1 to 10.

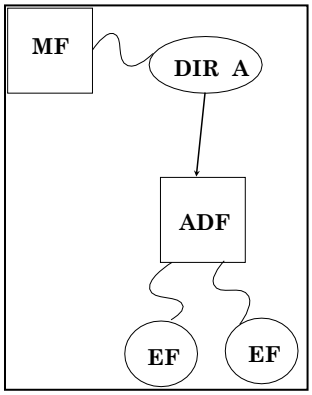

Figure 5: Simplest Card Structure Single Application

C2 Single Level Directory

Figure 6 gives an example of a multi-application card with a single directory. In this example, the root file (MF) does not support an application complying with this specification, and no restrictions are placed on the function of the MF. According to ISO/IEC 7816-4, a DIR file may be present but is not used by the application selection algorithm defined in section 12. Also note that the directory does not have entries for all ADFs (ADF2 to ADF5), as ADF5 is omitted. ADF5 can be selected only by a terminal that 'knows' ADF5 may exist in the card. The manner in which the terminal finds ADF5 is outside the scope of this specification.

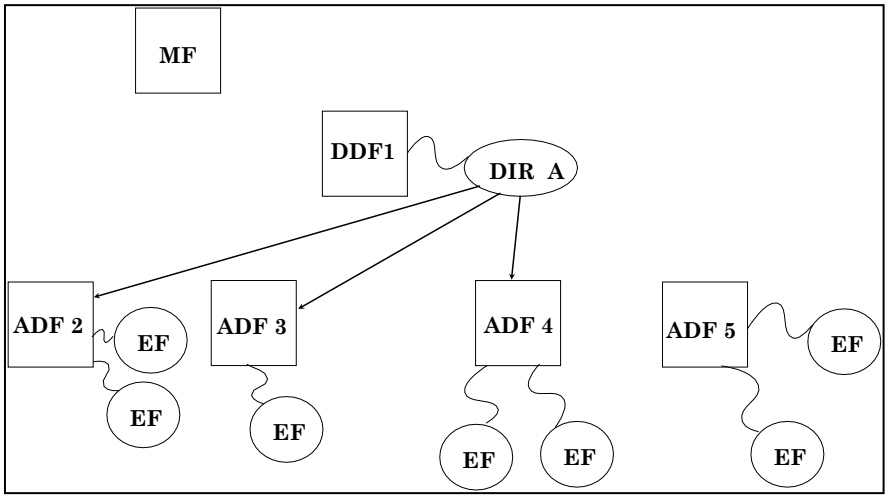

Figure 6: Single Level Directory

C3 Multi-Level Directory

Figure 7 is an example of a multi-application card with an *n* level directory structure. The first level directory ('DIR A') has entries for 2 ADFs – ADF3 and ADF4 – and a single DDF – DDF2. The directory attached to DDF2 ('DIR B') has entries for two ADFs – ADF21 and ADF22 – and a single DDF – DDF6. DDF5 has no entry in the root directory and can be found only by a terminal that 'knows' of the existence of DDF5. The manner in which the terminal finds and selects DDF5 is outside the scope of this specification, but the directory attached to DF5 ('DIR C') may conform to this specification, and, if found by the terminal, may lead the terminal to ADFs such as DF51, DF52, and DF53. DIR D, attached to DDF6, is a third level directory and points to four files (not shown), which may be either ADFs or more DDFs.

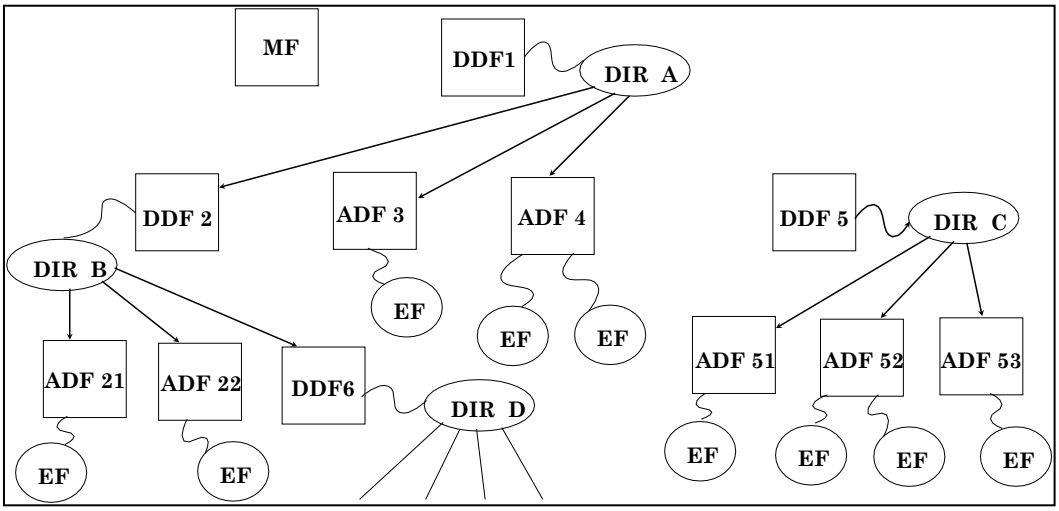

Figure 7: Third Level Directory

C4 Coding of Proprietary Directories

EMV does not mandate the formats for proprietary directories that may be present on the card in addition to the Payment System Directory. Directories following the structure defined in section 12.2 can be accessed using the methods described in this section.

These directories can have the same format as shown in Table 11 for the Payment System Directory Record Format, and can in addition to the one or more ADF Directory Entries also contain one or more DDF Directory Entries. If present, these DDF Directory entries can have the format as shown in Table 16 below.

Tag	Length	Value		Presence
'9D'	5–16	DDF Name		M
'73'	var.	Directory Discretionary Template		O [7]
	'XXXX' (Tag constructed according to Book 3 Annex B)	var.	1 or more additional proprietary data elements from an application provider, issuer, or IC card supplier, or EMV-defined tags that are specifically allocated to template '73'	O

Table 16: Example of a DDF Directory Entry Format

[7] Other data objects not relevant to this specification may appear in this constructed data object.

Part V

Common Core Definitions

Common Core Definitions

This Part describes an optional extension to this Book, to be used when implementing the Common Core Definitions (CCD). It is to be used in conjunction with Books 2, 3, and 4, including the Common Core Definitions part of Books 2 and 3.

These Common Core Definitions specify a minimum common set of card application implementation options, card application behaviours, and data element definitions sufficient to accomplish an EMV transaction. Terminals certified to be compliant with the existing EMV specifications will, without change, accept cards implemented according to the Common Core Definitions, since the Common Core Definitions are supported within the existing EMV requirements. To be compliant with the Common Core Definitions, an implementation shall implement all the additional requirements in the Common Core Definitions sections of all affected Books.

Changed Sections

Each section heading below refers to the section in this Book to which the additional requirements apply. The text defines requirements for a common core implementation, in addition to the requirements already specified in the referenced section of EMV.

Part III - Files, Commands, and Application Selection

11 Commands

11.3 SELECT Command-Response APDUs

11.3.5 Processing State Returned in the Response Message

The ICC shall support partial name selection and shall accept SELECT command messages containing at least the 5 high-order bytes of the DF Name (that is, the RID). Select Next functionality shall be supported.

Index

L

M

N

P

R

S

T

U