

ICS 35.200; 35.240.15; 35.240.40

English version

**Extensions for Financial Services (XFS) interface
specification Release 3.40 - Part 63: Identification Card
Device Class Interface - Migration from version 3.30 (CWA
16926) to Version 3.40 (this CWA) - Programmer's
Reference**

This CEN Workshop Agreement has been drafted and approved by a Workshop of representatives of interested parties, the constitution of which is indicated in the foreword of this Workshop Agreement.

The formal process followed by the Workshop in the development of this Workshop Agreement has been endorsed by the National Members of CEN but neither the National Members of CEN nor the CEN-CENELEC Management Centre can be held accountable for the technical content of this CEN Workshop Agreement or possible conflicts with standards or legislation.

This CEN Workshop Agreement can in no way be held as being an official standard developed by CEN and its Members.

This CEN Workshop Agreement is publicly available as a reference document from the CEN Members National Standard Bodies.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.



EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

Table of Contents

European Foreword.....	4
1. Migration Information.....	8
2. Identification Card Readers and Writers	9
2.1 Support for EMV Intelligent Contactless Card Readers.....	10
3. References	12
4. Info Commands	13
4.1 WFS_INF_IDC_STATUS.....	13
4.2 WFS_INF_IDC_CAPABILITIES	19
4.3 WFS_INF_IDC_FORM_LIST.....	24
4.4 WFS_INF_IDC_QUERY_FORM.....	25
4.5 WFS_INF_IDC_QUERY_IFM_IDENTIFIER.....	27
4.6 WFS_INF_IDC_EMVCLESS_QUERY_APPLICATIONS	28
5. Execute Commands	29
5.1 WFS_CMD_IDC_READ_TRACK.....	29
5.2 WFS_CMD_IDC_WRITE_TRACK	31
5.3 WFS_CMD_IDC_EJECT_CARD.....	33
5.4 WFS_CMD_IDC_RETAIN_CARD.....	35
5.5 WFS_CMD_IDC_RESET_COUNT	36
5.6 WFS_CMD_IDC_SETKEY	37
5.7 WFS_CMD_IDC_READ_RAW_DATA.....	38
5.8 WFS_CMD_IDC_WRITE_RAW_DATA	42
5.9 WFS_CMD_IDC_CHIP_IO	44
5.10 WFS_CMD_IDC_RESET.....	46
5.11 WFS_CMD_IDC_CHIP_POWER	47
5.12 WFS_CMD_IDC_PARSE_DATA	48
5.13 WFS_CMD_IDC_SET_GUIDANCE_LIGHT	49
5.14 WFS_CMD_IDC_POWER_SAVE_CONTROL	51
5.15 WFS_CMD_IDC_PARK_CARD	52
5.16 WFS_CMD_IDC_EMVCLESS_CONFIGURE	54
5.17 WFS_CMD_IDC_EMVCLESS_PERFORM_TRANSACTION	56
5.18 WFS_CMD_IDC_EMVCLESS_ISSUERUPDATE	62
5.19 WFS_CMD_IDC_SYNCHRONIZE_COMMAND	64
6. Events.....	65
6.1 WFS_EXEE_IDC_INVALIDTRACKDATA	65
6.2 WFS_EXEE_IDC_MEDIINSERTED	66
6.3 WFS_SRVE_IDC_MEDIAREMOVED	67

6.4	WFS_EXEE_IDC_MEDIARETAINED	68
6.5	WFS_EXEE_IDC_INVALIDMEDIA	69
6.6	WFS_SRVE_IDC_CARDACTION	70
6.7	WFS_USRE_IDC_RETAINBINTHRESHOLD	71
6.8	WFS_SRVE_IDC_MEDIADETECTED	72
6.9	WFS_SRVE_IDC_RETAINBINREMOVED	73
6.10	WFS_SRVE_IDC_RETAINBININSERTED	74
6.11	WFS_EXEE_IDC_INSERTCARD	75
6.12	WFS_SRVE_IDC_DEVICEPOSITION	76
6.13	WFS_SRVE_IDC_POWER_SAVE_CHANGE	77
6.14	WFS_EXEE_IDC_TRACKDETECTED	78
6.15	WFS_EXEE_IDC_EMVCLESSREADSTATUS	79
6.16	WFS_SRVE_IDC_MEDIARETAINED	80
7.	Form Description	81
8.	C-Header file	84
9.	Intelligent Contactless Card Sequence Diagrams	95
9.1	Single Tap Transaction Without Issuer Update Processing	96
9.2	Double Tap Transaction With Issuer Update Processing	97
9.3	Card Removed Before Completion	98
Appendix A.	Diagram Source	99

European Foreword

This CEN Workshop Agreement has been developed in accordance with the CEN-CENELEC Guide 29 “CEN/CENELEC Workshop Agreements – The way to rapid consensus” and with the relevant provisions of CEN/CENELEC Internal Regulations – Part 2. It was approved by a Workshop of representatives of interested parties on 2019-10-08, the constitution of which was supported by CEN following several public calls for participation, the first of which was made on 1998-06-24. However, this CEN Workshop Agreement does not necessarily include all relevant stakeholders.

The final text of this CEN Workshop Agreement was provided to CEN for publication on 2019-12-12.

The following organizations and individuals developed and approved this CEN Workshop Agreement:

- ATM Japan LTD
- AURIGA SPA
- BANK OF AMERICA
- CASHWAY TECHNOLOGY
- CHINAL ELECTRONIC FINANCIAL EQUIPMENT SYSTEM CO.
- CIMA SPA
- CLEAR2PAY SCOTLAND LIMITED
- DIEBOLD NIXDORF
- EASTERN COMMUNICATIONS CO. LTD – EASTCOM
- FINANZ INFORMATIK
- FUJITSU FRONTTECH LIMITED
- FUJITSU TECHNOLOGY
- GLORY LTD
- GRG BANKING EQUIPMENT HK CO LTD
- HESS CASH SYSTEMS GMBH & CO. KG
- HITACHI OMRON TS CORP.
- HYOSUNG TNS INC
- JIANGSU GUO GUANG ELECTRONIC INFORMATION TECHNOLOGY
- KAL
- KEBA AG
- NCR FSG
- NEC CORPORATION
- OKI ELECTRIC INDUSTRY SHENZHEN

- OKI ELECTRONIC INDUSTRY CO
- PERTO S/A
- REINER GMBH & CO KG
- SALZBURGER BANKEN SOFTWARE
- SIGMA SPA
- TEB
- ZIJIN FULCRUM TECHNOLOGY CO

It is possible that some elements of this CEN/CWA may be subject to patent rights. The CEN-CENELEC policy on patent rights is set out in CEN-CENELEC Guide 8 “Guidelines for Implementation of the Common IPR Policy on Patents (and other statutory intellectual property rights based on inventions)”. CEN shall not be held responsible for identifying any or all such patent rights.

The Workshop participants have made every effort to ensure the reliability and accuracy of the technical and non-technical content of CWA 16926-63, but this does not guarantee, either explicitly or implicitly, its correctness. Users of CWA 16926-63 should be aware that neither the Workshop participants, nor CEN can be held liable for damages or losses of any kind whatsoever which may arise from its application. Users of CWA 16926-63 do so on their own responsibility and at their own risk.

The CWA is published as a multi-part document, consisting of:

Part 1: Application Programming Interface (API) - Service Provider Interface (SPI) - Programmer's Reference

Part 2: Service Classes Definition - Programmer's Reference

Part 3: Printer and Scanning Device Class Interface - Programmer's Reference

Part 4: Identification Card Device Class Interface - Programmer's Reference

Part 5: Cash Dispenser Device Class Interface - Programmer's Reference

Part 6: PIN Keypad Device Class Interface - Programmer's Reference

Part 7: Check Reader/Scanner Device Class Interface - Programmer's Reference

Part 8: Depository Device Class Interface - Programmer's Reference

Part 9: Text Terminal Unit Device Class Interface - Programmer's Reference

Part 10: Sensors and Indicators Unit Device Class Interface - Programmer's Reference

Part 11: Vendor Dependent Mode Device Class Interface - Programmer's Reference

Part 12: Camera Device Class Interface - Programmer's Reference

Part 13: Alarm Device Class Interface - Programmer's Reference

Part 14: Card Embossing Unit Device Class Interface - Programmer's Reference

Part 15: Cash-In Module Device Class Interface - Programmer's Reference

Part 16: Card Dispenser Device Class Interface - Programmer's Reference

Part 17: Barcode Reader Device Class Interface - Programmer's Reference

Part 18: Item Processing Module Device Class Interface - Programmer's Reference

Part 19: Biometrics Device Class Interface - Programmer's Reference

Parts 20 - 28: Reserved for future use.

Parts 29 through 47 constitute an optional addendum to this CWA. They define the integration between the SNMP standard and the set of status and statistical information exported by the Service Providers.

Part 29: XFS MIB Architecture and SNMP Extensions - Programmer's Reference

Part 30: XFS MIB Device Specific Definitions - Printer Device Class

Part 31: XFS MIB Device Specific Definitions - Identification Card Device Class

Part 32: XFS MIB Device Specific Definitions - Cash Dispenser Device Class

Part 33: XFS MIB Device Specific Definitions - PIN Keypad Device Class

Part 34: XFS MIB Device Specific Definitions - Check Reader/Scanner Device Class

Part 35: XFS MIB Device Specific Definitions - Depository Device Class

Part 36: XFS MIB Device Specific Definitions - Text Terminal Unit Device Class

Part 37: XFS MIB Device Specific Definitions - Sensors and Indicators Unit Device Class

Part 38: XFS MIB Device Specific Definitions - Camera Device Class

Part 39: XFS MIB Device Specific Definitions - Alarm Device Class

Part 40: XFS MIB Device Specific Definitions - Card Embossing Unit Class

Part 41: XFS MIB Device Specific Definitions - Cash-In Module Device Class

Part 42: Reserved for future use.

Part 43: XFS MIB Device Specific Definitions - Vendor Dependent Mode Device Class

Part 44: XFS MIB Application Management

Part 45: XFS MIB Device Specific Definitions - Card Dispenser Device Class

Part 46: XFS MIB Device Specific Definitions - Barcode Reader Device Class

Part 47: XFS MIB Device Specific Definitions - Item Processing Module Device Class

Part 48: XFS MIB Device Specific Definitions - Biometrics Device Class

Parts 49 - 60 are reserved for future use.

Part 61: Application Programming Interface (API) - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Service Provider Interface (SPI) - Programmer's Reference

Part 62: Printer and Scanning Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 63: Identification Card Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 64: Cash Dispenser Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 65: PIN Keypad Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 66: Check Reader/Scanner Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 67: Depository Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 68: Text Terminal Unit Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 69: Sensors and Indicators Unit Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 70: Vendor Dependent Mode Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 71: Camera Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 72: Alarm Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) -

Programmer's Reference

Part 73: Card Embossing Unit Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 74: Cash-In Module Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 75: Card Dispenser Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 76: Barcode Reader Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

Part 77: Item Processing Module Device Class Interface - Migration from Version 3.30 (CWA 16926) to Version 3.40 (this CWA) - Programmer's Reference

In addition to these Programmer's Reference specifications, the reader of this CWA is also referred to a complementary document, called Release Notes. The Release Notes contain clarifications and explanations on the CWA specifications, which are not requiring functional changes. The current version of the Release Notes is available online from: https://www.cen.eu/work/Sectors/Digital_society/Pages/WSXFS.aspx.

The information in this document represents the Workshop's current views on the issues discussed as of the date of publication. It is provided for informational purposes only and is subject to change without notice. CEN makes no warranty, express or implied, with respect to this document.

1. Migration Information

XFS 3.40 has been designed to minimize backwards compatibility issues. This document highlights the changes made to the IDC device class between version 3.30 and 3.40, by highlighting the additions and deletions to the text.

2. Identification Card Readers and Writers

This section describes the functions provided by a generic identification card reader/writer service (IDC). These descriptions include definitions of the service-specific commands that can be issued, using the **WFSAsyncExecute**, **WFSGetInfo** and **WFSAsyncGetInfo** functions.

This service allows for the operation of the following categories of units:

- motor driven card reader/writer
- pull through card reader (writing facilities only partially included)
- dip reader
- contactless chip card readers
- permanent chip card readers (each chip is accessed through a unique logical service)

Some motor driven card reader/writers have parking stations inside and can place identification cards there. Once a card is in its parking station another card can be accepted by the card reader. Cards may only be moved out of a parking station if there is no other card present in the media read/write position, the chip I/O position, the transport, or the entry/exit slot.

The following tracks/chips and the corresponding international standards are taken into account in this document:

- | | |
|------------------------------|------------------------------------|
| • Track 1 | ISO 7811 |
| • Track 2 | ISO 7811 |
| • Track 3 | ISO 7811 / ISO 4909 |
| • Cash Transfer Card Track 1 | (JIS I: 8 bits/char) Japan |
| • Cash Transfer Card Track 3 | (JIS I: 8 bits/char) Japan |
| • Front Track 1 | (JIS II) Japan |
| • Watermark | Sweden |
| • Chip (contacted) | ISO 7816 |
| • Chip (contactless) | ISO 10536, ISO 14443 and ISO 18092 |

National standards like Transac for France are not considered, but can be easily included via the forms mechanism (see Section 7, Form Definition).

In addition to the pure reading of the tracks mentioned above, security boxes can be used via this service to check the data of writable tracks for manipulation. These boxes (such as CIM or MM) are sensor-equipped devices that are able to check some other information on the card and compare it with the track data.

Persistent values are maintained through power failures, open sessions, close session and system resets.

When the service controls a permanently connected chip card, **WFS_ERR_UNSUPP_COMMAND** will be returned to all commands except **WFS_INF_IDC_STATUS**, **WFS_INF_IDC_CAPABILITIES**, **WFS_CMD_IDC_CHIP_POWER**, **WFS_CMD_IDC_CHIP_IO** and **WFS_CMD_IDC_RESET**.

The following defines the roles and responsibilities of an application within EMV: A distinction needs to be made between EMV Contact support and EMV Contactless support.

When defining an EMV Contact implementation

- EMV Level 2 interaction is handled above the XFS API
- EMV Level 1 interaction is handled below the XFS API

All EMV status information that is defined as a Level 1 responsibility in the EMV specification should be handled below the XFS API.

EMVCo grants EMV Level 1 Approvals to contact IFMs and EMVCo Level 2 Approvals to Application Kernels.

When defining an EMV Contactless implementation

The responsibilities will depend on the type of EMV Contactless Product being implemented.

There are different EMVCo defined product types, they can be found in the EMVCo Type Approval – Contactless Product – Administrative Process document.

- In this specification when referring to the Contactless Product Type – Intelligent Card Reader :

The following must be included and handled below the XFS API:

- An EMVCo Approved Level 1 Contactless PCD
- Entry Point and POS System Architecture according to Book A and B
- EMV Kernels according to Book C1 to C7 (minimum one kernel needs to be supported)

The Network Interface & the Consumer, Merchant Interfaces will be managed above the XFS API.

2.1 Support for EMV Intelligent Contactless Card Readers

In relation to contactless transactions, the terminology used in this document is based on the EMV Contactless Specifications for Payment Systems, see the **Error! Reference source not found.** section.

There are a number of types of payment systems (or EMV) compliant contactless card readers, from the intelligent reader device; where the reader device handles most of the transaction processing and only returns the result, to a transparent card reader; where the contactless card reader device provides a generic communication channel to the card without having any in-built transaction processing capabilities.

A contactless payment system transaction can be performed in two different ways, magnetic stripe emulation; where the data returned from the chip is formatted as if it was read from the magnetic stripe, and EMV-like; where, in a similar way to a contact EMV transaction, the chip returns a full set of BER-TLV (Basic Encoding Rules-Tag Length Value) data. Each payment system defines when each type, or profile, is used for a transaction, but it is usually dependent on both the configuration of the terminal and contactless card being tapped.

This document will use “magnetic stripe emulation” and “EMV-like” to identify the two profiles of contactless transactions.

Support for a generic contactless communication channel to the card is provided via the WFS_CMD_IDC_CHIP_IO command. This is suitable for use with a transparent contactless card reader or with an intelligent contactless card reader device operating in a pass through mode.

The WFS_CMD_IDC_READ_RAW_DATA command can be used with an intelligent contactless card reader device to provide magnetic track emulation transactions. Only magnetic track emulation transactions can be supported using this command.

When using an intelligent contactless card reader to support both EMV-like and magnetic track emulation transactions a number of commands are required. The WFS_CMD_IDC_EMVCLESS_CONFIGURE command allows the exchange of data to configure the reader for card acceptance and the WFS_CMD_IDC_EMVCLESS_PERFORM_TRANSACTION command enables the reader and performs the transaction with the card when it is tapped. In most cases all the transaction steps involving the card are completed within the initial card tap. Section **Error! Reference source not found.**, Appendix provides a sequence diagram showing the expected IDC command sequences, as well as the cardholder and application actions when performing a contactless card based transaction.

Some contactless payment systems allow a 2nd tap of the contactless card. For example a 2nd tap can be used to process authorization data received from the host. In the case of issuer update data this second tap is performed via the WFS_CMD_IDC_EMVCLESS_ISSUERUPDATE command. Section 9, Appendix provides a sequence diagram showing the expected IDC command sequences, as well as the cardholder and application actions. The WFS_INF_IDC_EMVCLESS_QUERY_APPLICATIONS and WFS_CMD_IDC_EMVCLESS_CONFIGURE commands specified later in this document refer to the EMV terminology “Application Identifier (AID) - Kernel Combinations”. A detailed explanation can be found in Reference [2] and Reference [3] documents.

This document refers to BER-TLV tags. These are defined by each individual payment systems and contain the data exchanged between the application, contactless card and an intelligent contactless card reader. They are used to configure and prepare the intelligent contactless card reader for a transaction and are also part of the data that is returned by the reader on completion of the cards tap.

Based on the applicable payment system the application is expected to know which tags are required to be configured, what values to use for the tags and how to interpret the tags returned. Intelligent readers are expected to

know the BER-TLV tag definitions supported per payment system application. The tags provided in this document are examples of the types of tags applicable to each command. They are not intended to be a definite list.

This document is a preview generated by EVS

3. References

- | |
|--|
| 1. XFS Application Programming Interface (API)/Service Provider Interface (SPI), Programmer's Reference Revision 3.40. |
| 2. EMVCo Integrated Circuit Card Specifications for Payment Systems Version 4.3 |
| 3. EMVCo Contactless Specifications for Payment Systems, Version 2.4 |
| 4. EMVCo Contactless Type Approval Administrative Process Version 2.4 |