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

WORKSHOP AGREEMENT

CWA 14050-2

November 2000

ICS 35.200; 35.240.15; 35.240.40

Extensions for Financial Services (XFS) interface specification -
Release 3.0 - Part 2: Service Classes Definition

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

Ref. No CWA 14050-2:2000 E

Table of Contents

Foreword

This CWA is revision 3.0 of the XFS interface specification.

The move from an XFS 2.0 specification (CWA 13449) to a 3.0 specification has been prompted by a series of factors.

Initially, there has been a technical imperative to extend the scope of the existing specification of the XFS Manager to include new devices, such as the Card Embossing Unit.

Similarly, there has also been pressure, through implementation experience and the advance of the Microsoft technology, to extend the functionality and capabilities of the existing devices covered by the specification.

Finally, it is also clear that our customers and the market are asking for an update to a specification, which is now over 2 years old. Increasing market acceptance and the need to meet this demand is driving the Workshop towards this release.

The clear direction of the CEN/ISSS XFS Workshop, therefore, is the delivery of a new Release 3.0 specification based on a C API. It will be delivered with the promise of the protection of technical investment for existing applications and the design to safeguard future developments.

The CEN/ISSS XFS Workshop gathers suppliers as well as banks and other financial service companies. A list of companies participating in this Workshop and in support of this CWA is available from the CEN/ISSS Secretariat.

This CWA was formally approved by the XFS Workshop meeting on 2000-10-18. The specification is continuously reviewed and commented in the CEN/ISSS Workshop on XFS. It is therefore expected that an update of the specification will be published in due time as a CWA, superseding this revision 3.0.

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 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 Class Interface - Programmer's Reference

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

Part 16: Application Programming Interface (API) - Service Provider Interface (SPI) - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 17: Printer Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 18: Identification Card Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 19: Cash Dispenser Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 20: PIN Keypad Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 21: Depository Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 22: Text Terminal Unit Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 23: Sensors and Indicators Unit Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 24: Camera Device Class Interface - Migration from Version 2.0 (see CWA 13449) to Version 3.0 (this CWA) - Programmer's Reference

Part 25: Identification Card Device Class Interface - PC/SC Integration Guidelines

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 http://www.cenorm.be/isss/Workshop/XFS.

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

Revision History:

2.00	November 11, 1996	Initial release of Service Class Document.
3.00	October 18, 2000	Addition of:
		Alarms (ALM)
		Card Embossing Unit (CEU)
		Cash In Module (CIM)
		Cheque Readers and Scanners (CHK)

1. Background to Release 3.0

The CEN XFS Workshop is a continuation of the Banking Solution Vendors Council workshop and maintains a technical commitment to the Win 32 API. However, the XFS Workshop has extended the franchise of multi vendor software by encouraging the participation of both banks and vendors to take part in the deliberations of the creation of an industry standard. This move towards opening the participation beyond the BSVC's original membership has been very succesful with a current membership level of more than 20 companies.

The fundamental aims of the XFS Workshop are to promote a clear and unambiguous specification for both service providers and application developers. This has been achieved to date by sub groups working electronically and quarterly meetings.

The move from an XFS 2.0 specification to a 3.0 specification has been prompted by a series of factors. Initially, there has been a technical imperative to extend the scope of the existing specification of the XFS Manager to include new devices, such as the Card Embossing Unit.

Similarly, there has also been pressure, through implementation experience and the advance of the Microsoft technology, to extend the functionality and capabilities of the existing devices covered by the specification.

Finally, it is also clear that our customers and the market are asking for an update to a specification, which is now over 2 years old. Increasing market acceptance and the need to meet this demand is driving the Workshop towards this release.

The clear direction of the XFS Workshop, therefore, is the delivery of a new Release 3.0 specification based on a C API. It will be delivered with the promise of the protection of technical investment for existing applications and the design to safeguard future developments.

2. Service Classes

The following classes of services have been defined for the second version of the XFS Extensions for Financial Services to include the data and methods needed for the support of self-service, unattended, operations:

- Printers
- Identification Card Units
- Cash Dispensers
- Personal Identification Number Keypads (PIN pads)
- Depository Units
- Text Terminal Units
- Sensors and Indicators Units
- Vendor Dependent Mode
- Cameras
- Card Embossing Units
- Alarms
- Cash In Modules

The following sections detail for each of the service classes defined for this version of the WOSA Extensions for Financial Services:

- the standard values to be used as *class* attribute in the configuration information;
- the unique number assigned to each service class;
- the types of devices defined and supported by the service class specifications.

The table below summarises the unique attributes of each service class:

Service Class	Class Name	Class Identifier	Reference
Printers	PTR	1	CWA NNNNN- 3
Identification Card Units	IDC	2	CWA NNNNN- 4
Cash Dispensers	CDM	3	CWA NNNNN- 5
PIN pads	PIN	4	CWA NNNNN- 6
Check Readers and Scanners	CHK	5	CWA NNNNN- 7
Depository Units	DEP	6	CWA NNNNN- 8
Text Terminal Units	TTU	7	CWA NNNNN- 9
Sensors and Indicators Units	SIU	8	CWA NNNNN- 10
Vendor Dependent Mode	VDM	9	CWA NNNNN- 11
Cameras	CAM	10	CWA NNNNN- 12
Alarms	ALM	11	CWA NNNNN- 13
Card Embossing Units	CEU	12	CWA NNNNN- 14
Cash In Modules	CIM	13	CWA NNNNN- 15

2.1   Printers (PTR)

Class Name   PTR

Class Identifier WFS_SERVICE_CLASS_PTR = 1

The XFS printer service defines and supports five types of banking printers through a common interface:

- **Receipt Printer**
  The receipt printer is used to print cut sheet documents. It may or may not require insert or eject operations, and often includes an operator identification device, e.g., Teller A and Teller B lights, for shared operation.

- **Journal Printer**
  The journal is a continuous form device used to record a hardcopy audit trail of transactions, and for certain report printing requirements.

- **Passbook Printer**
  The passbook device is physically and functionally the most complex printer. The XFS definition supports automatic positioning of the book, as well as read/write capability for an optional integrated magnetic stripe. The implementation also manages the book geometry - i.e. the margins and centerfolds - presenting the simplest possible application interface while delivering the full range of functionality.

  Some passbook devices also support the dispensing of new passbooks from up to four passbook paper sources (upper, aux, aux2, lower). Some passbook devices may also be able to place a full passbook in a parking station, print the new passbook and return both to the customer. Passbooks can only be dispensed or moved from the parking station if there is no other media in the print position or in the entry/exit slot.

- **Document Printer**
  Document printing is similar to receipt printing -- a set of fields are positioned on one or more inserted sheets of paper -- but the focus is on full-size forms. It should be noted that the XFS environment supports the printing of text and graphic fields from the application. The electronic printing of the form image (the template portion of the form which is usually pre-printed with dot-matrix style printers) may also be printed by the application.

- **Scanner Printer**
  The scanner printer is a device incorporating both the capabilities to scan inserted documents and to print on them. These devices may have more than one area where documents may be retained.

Additional hardware components, like scanners, stripe readers, OCR readers, and stamps, normally attached directly to the printer are also controlled through this interface.

The specification refers to the terms paper and media. When the term paper is used this refers to paper that is situated in a paper supply attached to the device. The term media is used for media that is inserted by the customer (e.g. check and other material that is scanned) or that is issued to the customer (e.g. a receipt or statement). That means that a journal printer has only paper and scanners have only media. Receipt, document printers and also passbook printers with white passbook dispensing capability have both. As soon as the paper gets printed it becomes media.

2.2   Identification Card Units (IDC)

Class Name   IDC

Class Identifier WFS_SERVICE_CLASS_IDC = 2

This specification describes the functions provided by a generic identification card reader/writer service (IDC).

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

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
Watermark	Sweden
Chip (contacted)	ISO 7816
Chip (contactless)	ISO 10536.

National standards like Transac for France are not considered, but can be easily included via the forms mechanism.

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.

2.3   Cash Dispensers (CDM)

Class Name   CDM

Class Identifier WFS_SERVICE_CLASS_CDM = 3

This specification describes the functionality of a XFS compliant Cash Dispenser Module (CDM) service provider.

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

This specification covers the dispensing of items. An "item" is defined as any media that can be dispensed and includes coupons, documents, bills and coins. However, if coins and bills are both to be dispensed separate service providers must be implemented for each.

All currency parameters in this specification are expressed as a quantity of underlined minimum dispense units, as defined in the description of the WFS_INF_CDM_CURRENCY_EXP command.

There are two types of CDM: Self-Service CDM and Teller CDM. A Self-Service CDM operates in an automated environment, while a Teller CDM has an operator present. The functionality provided by the following commands is only applicable to a Teller CDM:

WFS_CMD_CDM_SET_TELLER_INFO
WFS_INF_CDM_ TELLER_INFO

It is possible for the CDM to be part of a compound device with the Cash In Module (CIM). This CIM\CDM combination is referred to throughout this specification as a "Cash Recycler".

If the device is a Cash Recycler then, if cash unit exchanges are required on both interfaces, the exchanges cannot be performed concurrently. An exchange on one interface must be complete (the WFS_CMD_CDM_END_EXCHANGE must have completed) before an exchange can start on the other interface. The WFS_ERR_CDM_EXCHANGEACTIVE error code will be returned if the correct sequence is not adhered to. If the device has recycle units of multiple currencies and/or denominations, then the CDM interface should be used for exchange operations involving these cash units.

The Cash-Out cash unit counts will be available through the CDM interface and the Cash-In cash unit counts will be available through the CIM interface. Counts for recycle cash units are available through both interfaces. The event WFS_SRVE_CDM_COUNTS_CHANGED will be posted if an operation on the CIM interface effects the recycle cash unit counts which are available through the CDM interface.

2.4   Personal Identification Number Keypads (PIN)

Class Name   PIN

Class Identifier WFS_SERVICE_CLASS_PIN = 4

This document describes the application program interface for personal identification number keypads (PIN pads) and other encryption/decryption devices.

This document describes the general interface for the following functions:
- Administration of encryption devices
- Loading of encryption keys
- Encryption / decryption
- Entering Personal Identification Numbers (PINs)
- PIN verification
- PIN block generation (encrypted PIN)
- Clear text data handling
- Function key handling
- PIN presentation to chipcard
- Read and write safety critical Terminal Data from/to HSM
- HSM and Chipcard Authentication

If the PIN Pad device has local display capability, display handling should be handled using the Text Terminal Unit (TTU) interface.

The adoption of this specification does not imply the adoption of a specific security standard.

> **Important Notes:**
> - This revision of this specification does not define key management procedures; key management is vendor-specific.
> - Key space management is customer-specific, and is therefore handled by vendor-specific mechanisms.
> - Only numeric PIN pads are handled in this specification.

This specification also supports the Hardware Security Module (HSM), which is necessary for the German ZKA Electronic Purse transactions. Furthermore the HSM stores terminal specific data.
This data will be compared against the message data fields (Sent and Received ISO8583 messages) prior to HSM-MAC generation/verification. HSM-MACs are generated/verified only if the message fields match the data stored.

Keys used for cryptographic HSM functions are stored separate from other keys. This must be considered when importing keys.

This version of PinPad complies to the current ZKA specification 3.0. It supports loading and unloading against card account for both card types (Type 0and Type 1) of the ZKA electronic purse. It also covers the necessary functionality for 'Loading against other legal tender'.

Key values are passed to the API as binary hexadecimal values, for example:
        0123456789ABCDEF = 0x01 0x23 0x45 0x67 0x89 0xAB 0xCD 0xEF

2.5   Check Readers and Scanners (CHK)

Class Name   **CHK**

Class Identifier   **WFS_SERVICE_CLASS_CHK = 5**

This specification describes the XFS service class of check readers and scanners. Check image scanners are treated as a special case of check readers, i.e., image-enabled instances of the latter. This class includes devices with a range of features, from small hand-held read-only devices through which checks are manually swiped one at a time, to desktop units which automatically feed the check one at a time; recording the MICR data and check image, and endorse or encode the check.

In the U.S., checks are always encoded in magnetic ink for reading by Magnetic Ink Character Recognition (MICR), and a single font is always used. In Europe some countries use MICR and some use Optical Character Recognition (OCR) character sets, with different fonts, for their checks.

In all countries, typical fields found encoded on a check include the bank ID number and the account number. Part of the processing done by the bank is to also encode the amount on the check, usually done by having an operator enter the handwritten or typewritten face amount on a numeric keypad.

This service class is currently defined only for attended branch service

2.6   Depository Unit (DEP)

Class Name   **DEP**

Class Identifier   **WFS_SERVICE_CLASS_DEP = 6**

A Depository is used for the acceptance and deposit of media into the device or terminal. There are two main types of depository: an envelope depository for the deposit of media in envelopes and a night safe depository for the deposit of bags containing bulk media.

An envelope depository accepts media, prints on the media and deposits the media into a holding container or bin. Some envelope depositories offer the capability to dispense an envelope to the customer at the start of a transaction. The customer takes this envelope, fills in the deposit media, possibly inscribes it and puts it into the deposit slot. The envelope is then accepted, printed and transported into a deposit container.

The envelope dispense mechanism may be part of the envelope depository device mechanism with the same entry/exit slot or it may be a separate mechanism with separate entry/exit slot.

Envelopes dispensed and not taken by the customer can be retracted back into the device. When the dispenser is a separate mechanism the envelope is retracted back into the dispenser container. When the dispenser is a common mechanism the envelope is retracted into the depository container.

A night safe depository normally only logs the deposit of a bag and does not print on the media.

2.7   Text Terminal Unit (TTU)

Class Name   **TTU**

Class Identifier   **WFS_SERVICE_CLASS_TTU = 7**

This specification describes the functions provided by a generic Text Terminal Unit (TTU) service. A Text Terminal Unit is a text i/o device, which applies both to ATM operator panels and to displays incorporated in devices such as PIN pads and printers. This service allows for the following categories of functions:

- Forms oriented input and output
- Direct display output

- Keyboard input
- LED settings and control

All position indexes are zero based, where column zero, row zero is the top-leftmost position.

If the device has no shift key, the WFS_CMD_TTU_READ_FORM and WFS_CMD_TTU_READ commands will return only upper case letters. If the device has a shift key, these commands return upper and lower case letters as governed by the user's use of the shift key.

2.8   Sensors and Indicators Units (SIU)

Class Name   **SIU**

Class Identifier   WFS_SERVICE_CLASS_SIU = 8

This specification describes the functions provided by a generic Sensors and Indicators Unit service. This service allows for the operation of the following categories of ports:

- Door sensors, such as cabinet, safe or vandal shield doors;
- Alarm sensors, such as tamper, seismic or heat sensors;
- Generic sensors, such as proximity or ambient light sensors;
- Key switch sensors, such as the ATM operator switch;
- Lamp/sign indicators, such as fascia light or audio indicators;
- Auxiliary indicators.

In self-service devices, the sensors and indicators unit is capable of dealing with external sensors, such as door switches, locks, alarms and proximity sensors, as well as external indicators, such as turning on lamps or heating.

2.9   Vendor Dependent Mode (VDM)

Class Name   **VDM**

Class Identifier   WFS_SERVICE_CLASS_VDM = 9

This specification describes the functionality of the services provided by the Vendor Dependent Mode (VDM).

In all device classes there needs to be some method of going into a vendor specific mode to allow for capabilities which go beyond the scope of the current XFS specifications. A typical usage of such a mode might be to handle some configuration or diagnostic type of function or perhaps perform some 'off-line' testing of the device. These functions are normally available on Self-Service devices in a mode traditionally referred to as Maintenance Mode or Supervisor Mode and usually require operator intervention. It is those vendor-specific functions not covered by (and not required to be covered by) XFS Service Providers that will be available once the device is in Vendor-Dependent mode.

This service provides the mechanism for switching to and from Vendor Dependent Mode. The VDM Service Provider can be seen as the central point through which all Enter and Exit VDM requests are synchronised.

Entry into, or exit from, Vendor Dependent Mode can be initiated either by an application or by the VDM Service Provider itself. If initiated by an application, then this application needs to issue the appropriate command to request entry or exit. If initiated by the VDM Service Provider i.e. some vendor dependent switch, then these request commands are in-appropriate and not issued.

Once the entry request has been made, all registered applications will be notified of the entry request by an event message. These applications must attempt to close all open sessions with XFS Service Providers as soon as possible and then issue an acknowledgement command to the VDM Service Provider when ready. Once all applications have acknowledged, the VDM Service Provider will issue event messages to these applications to indicate that the System is in Vendor Dependent Mode.

Similarly, once the exit request has been made all registered applications will be notified of the exit request by an event message. These applications must then issue an acknowledgement command to the VDM Service Provider immediately. Once all applications have acknowledged, the VDM Service Provider will issue event messages to these applications to indicate that the System has exited from Vendor Dependent Mode.

Thus, XFS compliant applications that do not need the system to be in Vendor Dependent Mode, must comply with the following:

- Every XFS application should open a session with the VDM ServiceProvider passing a valid ApplId and then register for all VDM entry and exit notices.

- Before opening any session with any other XFS Service Provider, check the status of the VDM Service Provider. If Vendor Dependent Mode is not "Inactive", do not open a session.

- When getting a VDM entry notice, close all open sessions with all other XFS Service Providers as soon as possible and issue an acknowledgement for the entry to VDM.

- When getting a VDM exit notice, acknowledge at once.

- When getting a VDM exited notice, re-open any required sessions with other XFS Service Providers.

This is mandatory for self-service but optional for branch.

2.10 Cameras (CAM)

Class Name   **CAM**

Class Identifier **WFS_SERVICE_CLASS_CAM = 10**

The XFS camera service defines the functions provided by banking camera systems.

Banking camera systems usually consist of a recorder, a video mixer and one or more cameras. If there are several cameras, each camera focuses a special place within the self-service area (eg. the room, the customer or the cash tray). By using the video mixer it can be decided, which of the cameras should take the next photo. Furthermore data can be given to be inserted in the photo (eg. date, time or bankcode).

If there is only one camera that can switch to take photos from different positions, it is presented by the service provider as a set of cameras, one for each of its possible positions.

2.11 Alarms (ALM)

Class Name   **ALM**

Class Identifier **WFS_SERVICE_CLASS_ALM = 11**

The XFS alarms service defines the functions provided by unauthorized access alarm units.

The Alarm device class is provided as a separate service due to the need to set or reset an Alarm when one or more logical services associated with an attended CDM or unattended (self-service) device are locked. Because logical services can be locked by the application the Alarm is implemented in a separate device class to ensure that a set (trigger) or reset operation can be performed at any time.

The Alarm device class can be part of a compound device, as in the case of many CDMs or self-service terminals, or may be separate physical alarms.

2.12 Card Embossing Unit (CEU)

Class Name   **CEU**

Class Identifier   **WFS_SERVICE_CLASS_CEU = 12**

The XFS card embossing unit service defines the functions provided by retail branch resident card embossing units.

Embossing card units are generally viewed by XFS as compound devices with the following capabilities and features:

- Embossing of magnetic stripe card/ smart card.
- Reading/encoding magnetic stripe tracks 1, 2, and 3.
- Reading/writing smart card.
- LCD display/ keypad input.

The XFS services supporting the various embossing card unit components are outlined as follows:

- Embossing of magnetic stripe card/ smart card – Card Embossing Unit (CEU) service.
- Reading/encoding magnetic stripe tracks 1, 2, and 3 – ID Card (IDC) service, however when combined encoding/ embossing is performed the CEU service class is used.
- Reading/writing smart cards - ID Card (IDC) service, however when combined writing smart card/ embossing is performed the CEU service class is used.
- LCD display/ keypad input – Text Terminal Unit (TTU) service.

2.13 Cash In Module (CIM)

Class Name   **CIM**

Class Identifier   **WFS_SERVICE_CLASS_CIM = 13**

This specification describes the functionality of a XFS compliant Cash In Module (CIM) service provider.

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

This specification covers the acceptance of items. An "item" is defined as any media that can be accepted and includes coupons, documents, bills and coins. However, if coins and bills are both to be accepted separate service providers must be implemented for each.

All currency parameters in this specification are expressed as a quantity of <u>minimum dispense units</u>, as defined in the description of the WFS_INF_CIM_CURRENCY_EXP command.

There are two types of CIM: Self-Service CIM and Teller CIM. A Self-Service CIM operates in an automated environment, while a Teller CIM has an operator present. The functionality provided by the following commands is only applicable to a Teller CIM:

WFS_CMD_CIM_SET_TELLER_INFO
WFS_INF_CIM_SET_TELLER_INFO

It is possible for the CIM to be part of a compound device with the Cash Dispenser Module (CDM). This CIM\CDM combination is referred to throughout this specification as a "Cash Recycler".

If the device is a Cash Recycler then, if cash unit exchanges are required on both interfaces, the exchanges cannot be performed concurrently. An exchange on one interface must be complete (the WFS_CMD_CIM_END_EXCHANGE must have completed) before an exchange can start on the other interface. The WFS_ERR_CIM_EXCHANGEACTIVE error code will be returned if the correct sequence is not adhered

to. If the device has recycle units of multiple currencies and/or denominations, then the CIM interface should be used for exchange operations involving these cash units.

The Cash-Out cash unit counts will be available through the CDM interface and the Cash-In cash unit counts will be available through the CIM interface. Counts for recycle cash units are available through both interfaces. The event WFS_SRVE_CIM_COUNTS_CHANGED will be posted if an operation on the CDM interface effects the recycle cash unit counts which are available through the CIM interface.

3. Planned Enhancements and Extensions

This section describes functions and facilities that are not fully defined in this version of the XFS Extensions for Financial Services specification; modifications and complete definitions will be supplied in later versions. Vendor and user input is encouraged on these functions and facilities, as well as suggestions as to additional functionality.

XFS currently includes specifications for access to the key classes of financial peripherals, as listed above. These existing specifications will be extended and enhanced based on vendor and user experience with them. The following classes of devices or services, and others that customers and vendors request, may be evaluated for inclusion in future versions of this specification:
- Hologram readers
- Point-to-point audio/video-conferencing
- Biometrics authentication hardware, e.g. signature capture devices, fingerprint capture devices, ...

Also to be considered for future versions of XFS are other types of services, such as financial transaction messaging and management, as well as related services for financial networks such as network and systems management and security. As with the current specification, all these capabilities will be specified for access from the familiar, consistent Microsoft Windows user interface and programming environments.

Please submit comments and questions to
    xfs-helpdesk@cenorm.be

Or contact
    Luc Van den Berghe
    CEN/ISSS Workshop Manager
    Rue de Stassart 36
    B-1050 Brussels
    Luc.vandenberghe@cenorm.be
    Tel: + 32 2 55 00 957