

**CEN**

**CWA 14923-3**

**WORKSHOP**

May 2004

**AGREEMENT**

---

ICS 35.240.40

Supersedes CWA 13937-3:2003

English version

**J/eXtensions for Financial Services (J/XFS) for the Java  
Platform - Part 3: Magnetic Stripe & Chip Card Device Class  
Interface - 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 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, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.



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

**Management Centre: rue de Stassart, 36 B-1050 Brussels**

## Contents

<b>CONTENTS .....</b>	<b>2</b>
<b>FOREWORD .....</b>	<b>4</b>
<b>HISTORY .....</b>	<b>5</b>
<b>1 SCOPE .....</b>	<b>6</b>
<b>2 OVERVIEW .....</b>	<b>7</b>
2.1 DESCRIPTION .....	7
2.1.1 <i>Magnetic Stripe Device</i> .....	7
2.1.2 <i>Chip Card Device</i> .....	7
2.2 CLASS HIERARCHY .....	9
2.3 CLASSES AND INTERFACES .....	10
2.4 SUPPORT CLASSES .....	11
<b>3 DEVICE BEHAVIOR .....</b>	<b>12</b>
3.1 DEVICE OPEN() .....	12
3.2 HANDLING OF NULL PARAMETERS .....	12
<b>4 CLASSES AND INTERFACES .....</b>	<b>13</b>
4.1 ACCESS TO PROPERTIES .....	13
4.2 EXCEPTIONS .....	13
4.3 IJXFSMAGSTRIPECONTROL .....	14
4.3.1 <i>Introduction</i> .....	14
4.3.2 <i>Properties</i> .....	14
4.3.3 <i>Methods</i> .....	15
4.4 IJXFSCHIPCARDCONTROL .....	19
4.4.1 <i>Introduction</i> .....	19
4.4.2 <i>Properties</i> .....	19
4.4.3 <i>Methods</i> .....	20
4.5 IJXFSMOTORIZEDCARD .....	23
4.5.1 <i>Introduction</i> .....	23
4.5.2 <i>Properties</i> .....	23
4.5.3 <i>Methods</i> .....	25
4.6 IJXFSMSDSECURE .....	27
4.6.1 <i>Introduction</i> .....	27
4.6.2 <i>Properties</i> .....	27
4.6.3 <i>Methods</i> .....	28
<b>5 SUPPORT CLASSES .....</b>	<b>31</b>
5.1 JXFSMSDTRACKS .....	31
5.1.1 <i>Properties</i> .....	31
5.1.2 <i>Methods</i> .....	32
5.2 JXFSMSDTRACKSELECTION .....	33
5.2.1 <i>Properties</i> .....	33
5.2.2 <i>Methods</i> .....	33
5.3 JXFSMSDREADDATA .....	34
5.3.1 <i>Properties</i> .....	34
5.3.2 <i>Methods</i> .....	35
5.4 JXFSCCDDATA .....	36
5.4.1 <i>Properties</i> .....	36
5.4.2 <i>Methods</i> .....	36
5.5 JXFSMSDWMDATA .....	37
5.5.1 <i>Properties</i> .....	37
5.5.2 <i>Methods</i> .....	37
5.6 JXFSMSDSECUREMODE .....	38

5.6.1	Properties .....	38
5.6.2	Methods .....	38
5.7	JXFSMSDREADDATASECURE .....	39
5.7.1	Properties .....	39
5.7.2	Methods .....	40
<b>6</b>	<b>CODES .....</b>	<b>41</b>
6.1	ERROR CODES .....	41
6.2	STATUS CODES .....	41
6.3	OPERATION CODES .....	42
6.4	CONSTANTS .....	42
<b>7</b>	<b>DEVICE SERVICE INTERFACE METHODS .....</b>	<b>44</b>
<b>8</b>	<b>APPENDIX A: CEN/ISSS WORKSHOP 14923:2004 CORE MEMBERS : .....</b>	<b>45</b>

## Foreword

This CWA contains the specifications that define the J/eXtensions for Financial Services (J/XFS) for the Java™ Platform, as developed by the J/XFS Forum and endorsed by the CEN/ISSS J/XFS Workshop. J/XFS provides an API for Java applications which need to access financial devices. It is hardware independent and, by using 100% pure Java, also operating system independent.

The CEN/ISSS J/XFS Workshop gathers suppliers (among others the J/XFS Forum members), service providers 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. The specification was agreed upon by the J/XFS Workshop Meeting of 2002-09-25/26 in Barcelona and a subsequent electronic review by the Workshop participants, and the final version was sent to CEN for publication on 2002-12-06.

The specification is continuously reviewed and commented in the CEN/ISSS J/XFS Workshop. The information published in this CWA is furnished for informational purposes only. CEN/ISSS makes no warranty expressed or implied, with respect to this document. Updates of the specification will be available from the CEN/ISSS J/XFS Workshop public web pages pending their integration in a new version of the CWA (see: <http://www.cenorm.be/cenorm/businessdomains/businessdomains/informationssystemstandardizationsystem/applying+technologies/j-xfs+workshop/index.asp>).

The J/XFS specifications are now further developed in the CEN/ISSS J/XFS Workshop. CEN/ISSS Workshops are open to all interested parties offering to contribute. Parties interested in participating should contact the CEN/ISSS Secretariat ([issss@cenorm.be](mailto:issss@cenorm.be)). To submit questions and comments for the J/XFS specifications, please contact the J/XFS Workshop Secretariat hosted in CEN/ISSS ([jxfs-helpdesk@cenorm.be](mailto:jxfs-helpdesk@cenorm.be)).

Questions and comments can also be submitted to the members of the J/XFS Forum, who are all CEN/ISSS J/XFS Workshop members, through the J/XFS Forum web-site <http://www.jxfs.com>

This CWA is composed of the following parts:

- Part 1: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Base Architecture - Programmer's Reference
- Part 2: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Pin Keypad Device Class Interface - Programmer's Reference
- Part 3: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Magnetic Stripe & Chip Card Device Class Interface - Programmer's Reference
- Part 4: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Text Input/Output Device Class Interface - Programmer's Reference
- Part 5: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Cash Dispenser, Recycler and ATM Interface - Programmer's Reference
- Part 6: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Printer Device Class Interface - Programmer's Reference
- Part 7: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Alarm Device - Programmer's Reference
- Part 8: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Sensors and Indicators Unit Device Class Interface - Programmer's Reference
- Part 9: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Depository Device Class Interface - Programmer's Reference
- Part 10: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Check Reader/Scanner Device Class Interface - Programmer's Reference
- Part 11: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Camera Specification - Programmer's Reference
- Part 12: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Vendor Dependant Mode Specification - Programmer's Reference

CWA 14923-3:2004 replaces CWA 13937-3:2003 and should be read in conjunction with CWA 13937-3:2000, which contains the previous release of the J/XFS specification

Note: Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. The Java Trademark Guidelines are currently available on the web at [http://java.sun.com/nav/business/trademark\\_guidelines.html](http://java.sun.com/nav/business/trademark_guidelines.html). All other trademarks are trademarks of their respective owners.

## HISTORY

The main differences to the previous CWA 13937:2000 are:

- Modified readata method description
- Modified ejectCard method, status event added
- Modified retainCard method, status event added
- Corrected some typing errors
- Added missing clarification on the writeData method
- Removed the JXFS\_E\_CLAIMED exception
- Removed “media taken” as a code for an intermediate event, at section 6.3
- Added JXFS\_S\_MEDIA\_STATUS events at the ejectCard and retainCard methods of the motorized card interface.
- Added class hierarchy diagram
- Modified the Description of the readData method of the IJxfsMagStripeControl interface, relating to the magnetic pre-head detection.
- Added paragraph describing handling of null parameters
- Changed from lowercase “j” to uppercase “J” in all interface names starting with “IJxfs...”

## 1 Scope

This document describes the Magnetic Stripe Device (MSD) as well as Chip Card Device (CCD) classes based on the basic architecture of J/XFS which is similar to the JavaPOS architecture. It is event driven and asynchronous.

Three basic levels are defined in JavaPOS. For J/XFS this model is extended by a communication layer, which provides device communication that allows distribution of applications and devices within a network. So we have the following layers in J/XFS :

- Application
- Device Control and Device Manager
- Device Communication
- Device Service

Application developers program against control objects and the Device Manager which reside in the Device Control layer. This is the usual interface between applications and J/XFS devices. Device Control objects access the Device Manager to find an associated Device Service. Device Service objects provide the functionality to access the real device (i.e. like a device driver).

During application startup the Device Manager is responsible for locating the desired Device Service object and attaching this to the requesting Device Control object. Location and/or routing information for the Device Manager reside in a central repository.

To support Magnetic Stripe devices and Chip Card devices the basic Device Control structure is extended with various properties and methods specific to this device which are described on the following pages.

## 2 Overview

### 2.1 Description

This document describes the J/XFS support classes for both Magnetic Stripe devices (MSD) as well as Chip Card devices (CCD).

As well as the rest of J/XFS device controls, J/XFS Magnetic Stripe and J/XFS Chip Card devices use the event driven model and the same behavioral model. Therefore, in the case of a Magnetic Stripe device, the application will instantiate a J/XFS Magnetic Stripe Device Control Object and then use the available methods to do I/O. When an I/O method is called, the J/XFS Magnetic Stripe Device Service will attempt to process the requested I/O. If the request is invalid or an exception is encountered, the application will be notified by a J/XFS exception. Completion of the request will be reported by an event. Thus the application must register itself with the J/XFS Magnetic Stripe Device Control Object for the various types of events it wishes to handle.

The same model applies to all J/XFS device controls and, in particular, to the Chip Card Device control.

#### 2.1.1 Magnetic Stripe Device

The J/XFS Magnetic Stripe Reader/Encoder Device Support allows for the operation of devices with magnetic stripe read/write capabilities. Following are typical devices with such a capability:

- motor driven card reader/writer
- pull through card reader/writer
- dip card reader/writer

The following tracks 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

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.

Leds handling will be defined based on initialization configuration so no reference to them is made in this document.

Handling of *watermark* is also considered.

#### 2.1.2 Chip Card Device

The J/XFS Chip Card Device Support allows for the operation of devices with chip access capabilities. Following are typical devices with such a capability:

- motor driven chip card devices.
- dip chip card devices.