



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

WORKSHOP AGREEMENT

CWA 13937-6

August 2000

ICS 35.240.40

J/eXtensions for Financial Services (J/XFS) for the Java Platform - Part
6: Printer Device Class Interface - Programmer's Reference

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

© 2000 CEN

All rights of exploitation in any form and by any means reserved world-wide for
CEN National Members

Ref. No CWA 13937-6:2000 E

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 1999-12-15/16 in Geneva and a subsequent electronic review by the Workshop participants, and the final version was sent to CEN for publication on 2000/06-21.

The specification is continuously reviewed and commented in the CEN/ISSS J/XFS Workshop. It is therefore expected that an update of the specification will be published in due time as a CWA, superseding this one. 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/iss/wkshp/j-xfs/cwa-updates>).

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 (iss@cenorm.be). To submit questions and comments for the J/XFS specifications, please contact the CEN/ISSS Secretariat (iss@cenorm.be) who will be forwarding them to the J/XFS Workshop.

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

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. All other trademarks are trademarks of their respective owners.

Contents

1	SCOPE	5
2	OVERVIEW	6
2.1	DESCRIPTION	6
2.2	CLASS HIERARCHY	7
2.3	CLASS AND INTERFACE SUMMARY	8
3	DEVICE BEHAVIOR	10
3.1	DEVICE OPEN()	10
4	CLASSES AND INTERFACES	11
4.1	ACCESS TO PROPERTIES	11
4.2	EXCEPTIONS	11
4.3	IJXFSPRINTERCONTROL	12
4.3.1	Summary	12
4.3.2	Properties	13
4.3.3	Methods	14
4.3.4	Events	21
4.4	IJXFSJECT	23
4.4.1	Summary	23
4.4.2	Properties	23
4.4.3	Methods	24
4.4.4	Events	25
4.5	IJXFSRETRACT	26
4.5.1	Summary	26
4.5.2	Properties	26
4.5.3	Methods	27
4.5.4	Events	28
4.6	IJXFSMEDIATURN	29
4.6.1	Summary	29
4.6.2	Properties	29
4.6.3	Methods	30
4.7	IJXFSREAD	31
4.7.1	Summary	31
4.7.2	Properties	31
4.7.3	Methods	32
4.7.4	Events	34
5	SUPPORT CLASSES	35
5.1	JXFSPTRCTRLMEDIACAPABILITY	36
5.1.1	Summary	36
5.1.2	Properties	36
5.1.3	Methods	37
5.2	JXFSPTRCTRLTURNCAPABILITY	39
5.2.1	Summary	39
5.2.2	Properties	39
5.2.3	Methods	39
5.3	JXFSPTREXTENTCAPABILITY	41
5.3.1	Summary	41
5.3.2	Properties	41
5.3.3	Methods	41
5.4	JXFSPTRFIELD	42
5.4.1	Summary	42
5.4.2	Properties	42
5.5	JXFSPTRFIELDFAILURE	44
5.5.1	Summary	44
5.5.2	Properties	44
5.6	JXFSPTRFORM	45
5.6.1	Summary	45
5.6.2	Properties	46
5.7	JXFSPTRFORMSCONFIG	48
5.7.1	Summary	48

5.7.2	Properties.....	49
5.8	JXFSPTRIMAGE.....	51
5.8.1	Summary	51
5.8.2	Properties.....	51
5.9	JXFSPTRMAXRETRACTCAPABILITY.....	52
5.9.1	Summary	52
5.9.2	Properties.....	52
5.10	JXFSPTRMAXSTACKERCAPABILITY.....	53
5.10.1	Summary	53
5.10.2	Properties.....	53
5.11	JXFSPTRMEDIA	54
5.11.1	Summary	54
5.11.2	Properties.....	54
5.12	JXFSPTRMEDIAEXTENTS.....	58
5.12.1	Summary	58
5.12.2	Properties.....	58
5.13	JXFSPTRREADFORMCAPABILITY	59
5.13.1	Summary	59
5.13.2	Properties.....	59
5.13.3	Methods	59
5.14	JXFSPTRREADIMAGECAPABILITY	61
5.14.1	Summary	61
5.14.2	Properties.....	61
5.14.3	Methods	62
5.15	JXFSPTRRETRACTCOUNT.....	63
5.15.1	Summary	63
5.15.2	Properties.....	63
5.15.3	Methods	63
5.16	JXFSPTRWRITEFORMCAPABILITY.....	64
5.16.1	Summary	64
5.16.2	Properties.....	64
5.16.3	Methods	64
6	STATUS EVENT CLASSES	66
6.1	JXFSPTHRESHOLDSTATUS.....	66
6.1.1	Summary	66
6.2	JXFSPTRLAMPSTATUS	68
6.2.1	Summary	68
6.2.2	Properties.....	68
6.2.3	Methods.....	69
6.3	JXFSMEDIASSTATUS	70
6.4	JXFSPTRSTATUS.....	71
6.4.1	Summary	71
6.4.2	Properties.....	71
6.4.3	Events	72
7	CODES	73
7.1	ERROR CODES	73
7.2	EXCEPTION CODES	74
7.3	STATUS CODES.....	74
7.4	CONSTANTS.....	76
7.5	OPERATION ID CODES	80
8	DEVICE SERVICE INTERFACE METHODS	81
8.1	FORM, FIELD AND MEDIA DEFINITIONS	82
9	INDEX.....	83

1 Scope

This document describes the printer device class 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 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 printers 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

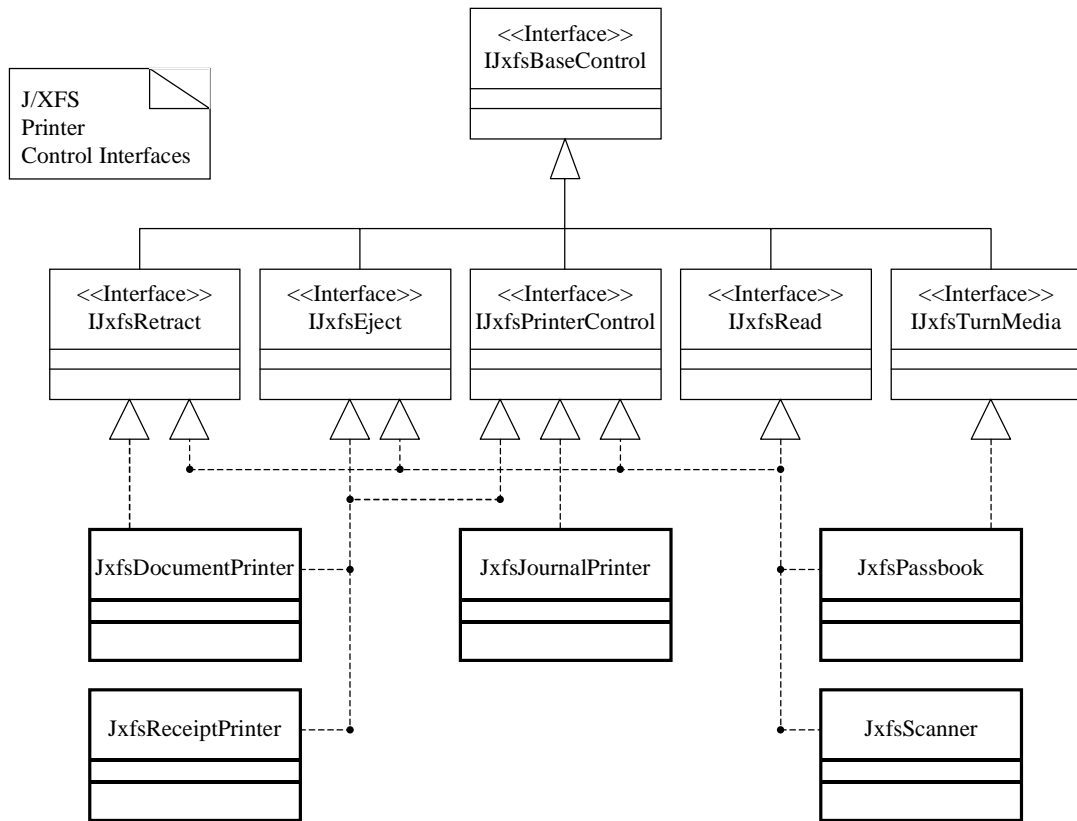
2.1 Description

The J/XFS Printer Device Support allows for the operation of the following categories of printers:

- **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 J/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 geometry of the book - i.e. the margins and centerfolds - presenting the simplest possible application interface while delivering the full range of functionality.
- **Document Printer**
Document printing is similar to receipt printing -- a set of fields are positioned on an inserted sheet of paper -- but the focus is on full-size forms. It should be noted that the J/XFS environment only implements the printing of text fields from the application. The electronic printing of the form image itself is not supported; but can be delivered as an added-value extension by the vendor. Statement printers belong to this category
- **Scanner**
The scanner device is able to scan any inserted printed or handwritten media. It may also be capable of printing.
The J/XFS definition supports automatic positioning of the inserted media, as well as read/write capability.

The J/XFS Printer Device Support uses the event driven model. The application will instantiate a J/XFS Printer Device Control Object and then call the defined I/O methods with passing data objects containing the parameters. When an I/O method is called, the J/XFS Printer Device Support 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 Printer Device Control Object for the various types of events it wishes to handle. If forms are being used then the J/XFS Printer Device Service will access the form indicated by the application via the published J/XFS configuration interface and use the form data to define positioning and presentation information for each of the fields on the document.

2.2 Class Hierarchy



2.3 Class and Interface Summary

The following classes and interfaces are used by the J/XFS Printer Device Controls. In order to support the definition of the different properties of the different printer devices (see Introduction), the J/XFS Printer Device Controls are defined in a class hierarchy.

Class or Interface	Name	Description	Extends / Implements
Interface	IJxfsBaseControl	Base interface for all device controls. Contains methods specific to all the device controls.	--
Class	JxfsBaseControl	Base class for all device controls. Implements the methods defined in the IJxfsBaseControl Interface. Contains the properties specific to all device controls.	Implements: IJxfsBaseControl
Interface	IJxfsPrinterControl	Base interface for all printer controls. Contains the methods specific to all the device controls for the printer device category.	Extends: IJxfsBaseControl
Interface	IJxfsEject	Interface that contains methods for the eject functionality of receipt printers, passbook printers, document printers and scanners.	Extends: IJxfsBaseControl
Interface	IJxfsMediaTurn	Interface that contains methods to turn media inside a printer	Extends: IJxfsBaseControl
Interface	IJxfsRetract	Interface that contains methods for the retract functionality of passbook printers, document printers and scanners.	Extends: IJxfsBaseControl
Interface	IJxfsRead	Interface that contains methods for the read functionality of scanners and passbook printers.	Extends: IJxfsBaseControl
Class	JxfsDocumentPrinter	Class for the Document Printer control	Implements: IJxfsPrinterControl IJxfsEject IJxfsRetract
Class	JxfsJournalPrinter	Class for the Journal Printer control	Implements: IJxfsPrinterControl

Class or Interface	Name	Description	Extends / Implements
Class	JxfsPassbookPrinter	Class for the Passbook Printer control.	Implements: IJxfsPrinterControl IJxfsEject IJxfsRetract IJxfsMediaTurn IJxfsRead
Class	JxfsReceiptPrinter	Class for the Receipt Printer control.	Implements: IJxfsPrinterControl IJxfsEject
Class	JxfsScanner	Class for the Scanner control.	Implements: IJxfsPrinterControl IJxfsEject IJxfsRetract IJxfsRead
Interface	IJxfsEventNotification	Includes one callback method per event type. The Device Service calls these methods to cause events to be delivered to the application.	--

3 Device behavior

3.1 Device open()

During the device open call the Device Service tries to access the connected device. This fails for the following circumstances:

JXFS_E_HARDWAREERROR	If the device could not be accessed. This may be that the device is not connected or broken.
JXFS_E_OPEN	The open was already done by this Device Control.

4 Classes and Interfaces

All operation methods return an identificationID. If a method cannot be processed a JxfsException is thrown.

After processing has taken place, an OutputComplete – Event is generated which contains detailed information about the status of the operation, i.e. if it failed or succeeded, and eventually additional data as a result.

The Constants, Error Codes, Exceptions, Status Codes and Support classes that are used in the methods are described in special chapters at the end of the documentation.

4.1 Access to properties

Please note the following when determining the meaning of a property's **Access**:

R	The property is read only.
W	The property is write only.
R/W	The property may be read or written.

To read or write a property the application must use the appropriate methods as defined in the JavaBeans specification.

getProperty

Syntax	Property <i>getProperty(void)</i> throws <i>JxfsException</i> ;
Description	Returns the requested property.
Parameter	None
Event	No additional events are generated.
Exceptions	Some possible JxfsException <i>value codes</i> . See section on JxfsExceptions for other JxfsException value codes. JXFS_E_CLOSED JXFS_E_REMOTE JXFS_E_UNREGISTERED

setProperty

Syntax	Property <i>setProperty(Property)</i> throws <i>JxfsException</i> ;
Description	Sets the requested property.
Parameter	None
Event	No additional events are generated.
Exceptions	Some possible JxfsException <i>value codes</i> . See section on JxfsExceptions for other JxfsException value codes. JXFS_E_CLOSED JXFS_E_PARAMETER_INVALID JXFS_E_REMOTE JXFS_E_UNREGISTERED

4.2 Exceptions

The methods described for the specific interfaces all can throw at least the following exceptions :

Exception	Value
<i>JXFSException</i>	JXFS_E_CLOSED
	JXFS_E_PARAMETER_INVALID
	JXFS_E_NOT_SUPPORTED
	JXFS_E_REMOTE
	JXFS_E_UNREGISTERED

Only if a method can throw additional exception this is explicitly mentioned.

4.3 IJxfsPrinterControl

The J/XFS Printer Device Control Subclass is defined in JxfsPrinterControl and is a subclass of JxfsBaseControl. Its interface is defined in IJxfsPrinterControl which is a subclass of IJxfsBaseControl. The intent of the J/XFS Printer Device Control object is to allow data and control to pass between the application and the device support code so that the associated device can be accessed.

4.3.1 Summary

Property	Type	Access	Initialized after
compound	boolean	R	
ctrlMediaCapability	JxfsPtrCtrlMedia Capability	R	
extentCapability	JxfsPtrExtentCapability	R	
formsConfig	JxfsPtrFormsConfig	R/W	
ptrStatus	JxfsPtrStatus	R	
writeFormCapability	JxfsPtrWriteForm Capability	R	

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
<i>setProperty</i>	<i>Property</i>	
ctrlMedia	identificationID	
isCompound	boolean	
getFormList	identificationID	
mediaExtents	identificationID	
getMediaList	identificationID	
printForm	identificationID	
printRawData	identificationID	
getFieldDescription	identificationID	
getFormDescription	identificationID	
getMediaDescription	identificationID	
resetPrinter	identificationID	

Event	May occur after
StatusEvent JXFS_S_PTR_DEVICE JXFS_S_PTR_MEDIA JXFS_S_PTR_PAPER JXFS_S_PTR_TONER	
IntermediateEvent JXFS_I_PTR_MEDIA_NOT_PRESENT JXFS_I_PTR_MEDIA_INSERTED JXFS_E_PTR_FIELD_FAILURE	<i>printForm()</i> , <i>printRawData()</i> , <i>mediaExtents()</i> <i>printForm()</i> , <i>printRawData()</i> , <i>mediaExtents()</i> <i>printForm()</i>
OCPtrFieldInfoEvent	<i>getFieldDescription()</i>
OCPtrFormInfoEvent	<i>getFormDescription()</i>
OCPtrFormListEvent	<i>getFormList()</i>
OCPtrMediaExtentsEvent	<i>mediaExtent()</i>
OCPtrMediaInfoEvent	<i>getMediaDescription()</i>
OCPtrMediaListEvent	<i>getMediaList()</i>
OCPtrRawDataEvent	<i>printRawData()</i>

Event	May occur after
OperationCompleteEvent JXFS_O_PTR_CTRL_MEDIA JXFS_O_PTR_RESET_PRINTER JXFS_O_PTR_WRITE_FORM_DATA JXFS_O_PTR_WRITE_RAW_DATA	<i>ctrlMedia()</i> <i>resetPrinter()</i> <i>printForm()</i> <i>printRawData()</i>

4.3.2 Properties

compound (R)

Type	<i>boolean</i>
Initial Value	false
Description	specifies whether the device is a compound device

ptrStatus (R)

Type	<i>JxfsPtrStatus</i>
Initial Value	<i>aJxfsPtrStatus</i> (for initial values see <i>JxfsPtrStatus</i>)
Description	see <i>JxfsPtrStatus</i>
Event	If the value of this property changes, the Device Service will send all registered StatusListeners a StatusEvent with one of the following status values :
	Value
	JXFS_S_PTR_DEVICE
	JXFS_S_PTR_MEDIA
	JXFS_S_PTR_PAPER
	JXFS_S_PTR_TONER

See description of the support classes for all other properties

4.3.3 Methods

ctrlMedia

Syntax	<i>identificationID ctrlMedia(int mediaControl) throws JxfsException;</i>																
Description	This command is used to control a form drawn in by the device (e.g. after reading or in case of termination of an application request).																
Parameter	<table border="0"> <thead> <tr> <th style="text-align: left;">Type</th> <th style="text-align: left;">Name</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td><i>int</i></td> <td>mediaControl</td> <td>Specifies the manner in which the media should be handled, as a combination of the following flags : JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP</td> </tr> </tbody> </table>	Type	Name	Meaning	<i>int</i>	mediaControl	Specifies the manner in which the media should be handled, as a combination of the following flags : JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP										
Type	Name	Meaning															
<i>int</i>	mediaControl	Specifies the manner in which the media should be handled, as a combination of the following flags : JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP															
Exceptions	No additional exceptions generated.																
Events	<p>Additional Events can be generated :</p> <p>OperationCompleteEvent When a <i>ctrlMedia()</i> operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Field</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_PTR_CTRL_MEDIA</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table> <p>StatusEvent When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Field</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td><i>status</i></td> <td>JXFS_S_PTR_MEDIA</td> </tr> <tr> <td><i>details</i></td> <td>JxfsMediaStatus</td> </tr> </tbody> </table>	Field	Value	<i>operationID</i>	JXFS_O_PTR_CTRL_MEDIA	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT	<i>data</i>	none	Field	Value	<i>status</i>	JXFS_S_PTR_MEDIA	<i>details</i>	JxfsMediaStatus
Field	Value																
<i>operationID</i>	JXFS_O_PTR_CTRL_MEDIA																
<i>identificationID</i>	The corresponding ID																
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT																
<i>data</i>	none																
Field	Value																
<i>status</i>	JXFS_S_PTR_MEDIA																
<i>details</i>	JxfsMediaStatus																

getFieldDescription

Syntax	<i>identificationID getFieldDescription(String[] fieldNames, String formName) throws JxfsException;</i>									
Description	This method is used to retrieve details of the definition of a single or all fields on a specified form. <i>fieldNames</i> and <i>formName</i> will be used to define the fields whose definitions are requested.									
Parameter	<table border="0"> <thead> <tr> <th style="text-align: left;">Type</th> <th style="text-align: left;">Name</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>String[]</td> <td>fieldNames</td> <td>Names of the requested fields.</td> </tr> <tr> <td>String</td> <td>formName</td> <td>Name of the requested form.</td> </tr> </tbody> </table>	Type	Name	Meaning	String[]	fieldNames	Names of the requested fields.	String	formName	Name of the requested form.
Type	Name	Meaning								
String[]	fieldNames	Names of the requested fields.								
String	formName	Name of the requested form.								
Exceptions	No additional exceptions generated.									
Events	<p>Additional Events can be generated :</p> <p>OCPtrFieldInfoEvent When a <i>getFieldDescription()</i> operation is completed an OCPtrFieldInfoEvent event will be sent by J/XFS Printer Device Control to the registered OCPtrFieldInfoListener.</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Field</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FORM_NOT_FOUND</td> </tr> <tr> <td><i>data</i></td> <td>String[] aFormsList List of the forms available on the printer</td> </tr> </tbody> </table>	Field	Value	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FORM_NOT_FOUND	<i>data</i>	String[] aFormsList List of the forms available on the printer	
Field	Value									
<i>identificationID</i>	The corresponding ID									
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FORM_NOT_FOUND									
<i>data</i>	String[] aFormsList List of the forms available on the printer									

getFormDescription

Syntax	<i>identificationID getFormDescription(String formName) throws JxfsException;</i>		
Description	This method is used to retrieve details of the definition of a specified form. <i>formName</i> will be used to define the form whose definition is requested.		
Parameter	Type	Name	Meaning
	String	formName	Name of the requested form.
Exceptions	No additional exceptions generated.		
Events	Additional Events can be generated : OCPtrFormInfoEvent When a <i>getFormDescription()</i> operation is completed an OCPtrFormInfoEvent event will be sent by J/XFS Printer Device Control to the registered OCPtrFormInfoListener.		
	Field	Value	
	<i>identificationID</i>	The corresponding ID	
	<i>result</i>	JXFS_RC_SUCCESSFUL	
		JXFS_E_PTR_FORM_NOT_FOUND	
	<i>data</i>	JxfsPtrForm aJxfsPtrForm	Description of the requested form.

getFormList

Syntax	<i>identificationID formList(void) throws JxfsException;</i>		
Description	This method is used retrieve a list of the names of the form definitions available on the printer.		
Parameter	None		
Exceptions	No additional exceptions generated.		
Events	Additional Events can be generated : OCPtrFormListEvent When a <i>getFormList()</i> operation is completed an OCPtrFormListEvent event will be sent by J/XFS Printer Device Control to all registered OCPtrFormListListeners.		
	Field	Value	
	<i>identificationID</i>	The corresponding ID	
	<i>result</i>	JXFS_RC_SUCCESSFUL	
		JXFS_E_PTR_NOFORMS	
	<i>data</i>	String[] aFormsList	List of the forms available on the printer

getMediaDescription

Syntax	<i>identificationID getMediaDescription(String mediaName) throws JxfsException;</i>		
Description	This method is used to retrieve details of the definition of a specified media. <i>mediaName</i> will be used to define the media whose definition is desired.		
Parameter	Type	Name	Meaning
	String	mediaName	Name of the requested media.
Exceptions	No additional exceptions generated.		
Events	Additional Events can be generated : OCPtrMediaInfoEvent When a <i>queryMedia()</i> operation is completed an OCPtrMediaInfoEvent event will be sent by J/XFS Printer Device Control to the registered OCPtrMediaInfoListener.		
	Field	Value	
	<i>identificationID</i>	The corresponding ID	
	<i>result</i>	JXFS_RC_SUCCESSFUL	
		JXFS_E_PTR_MEDIA_NOT_FOUND	
	<i>data</i>	JxfsPtrMedia aJxfsPtrMedia	Description of the requested media.

getMediaList

Syntax	<i>identificationID getMediaList(void) throws JxfsException;</i>
Description	This method is used retrieve a list of names of the media definitions available on the printer.
Parameter	None
Exceptions	No additional exceptions generated.
Events	Additional Events can be generated : OCPtrMediaListEvent When a <i>mediaList()</i> operation is completed an OCPtrMediaListEvent event will be sent by J/XFS Printer Device Control to all registered OCPtrMediaListListeners.
Field	Value
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NOMEDIAS
<i>data</i>	String[] aFormsList List of the forms available on the printer

mediaExtents

Syntax	<i>identificationID mediaExtents(void) throws JxfsException;</i>
Description	This method is used to get the extents of the media inserted in the printer. The extents will be based on the values of <i>formsConfig.base</i> , <i>formsConfig.unitX</i> and <i>formsConfig.unitY</i> . If no media is present the printer waits endless or until cancelled by the application.
Parameter	None
Exceptions	No additional exceptions generated.
Events	Additional Events can be generated : OCPtrMediaExtentEvent When a <i>mediaExtents()</i> operation is completed an OCPtrMediaExtentEvent event will be sent by J/XFS Printer Device Control to all registered OCPtrMediaExtentListeners.
Field	Value
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	<i>JxfsPtrMediaExtents</i> aJxfsPtrMediaExtents The extents of the inserted media.

IntermediateEvent

If no media is present the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

Field	Value
<i>operationID</i>	JXFS_O_PTR_MEDIA_EXTENTS
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

IntermediateEvent

If media is inserted and the operation can continue the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

Field	Value
<i>operationID</i>	JXFS_O_PTR_MEDIA_EXTENTS
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_MEDIA_INSERTED
<i>data</i>	none

StatusEvent

When the status of the media changes a StatusEvent is sent to all

registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_MEDIA
<i>details</i>	JxfsMediaStatus

printForm

Syntax

identificationID printForm(String formName, String mediaName, String[] fieldWriteData) throws JxfsException;

Description

This method prints the form with the name *formName* using the description of the media defined by *mediaName*. After a successful completion of this output operation, an *OperationCompleteEvent* is issued to inform the application of the results. If no media is present the printer waits endless or until cancelled by the application.

Parameter

Type	Name	Meaning
String	formName	Name of the form to be printed.
String	mediaName	Name of the media to be used for printing.
String[]	fieldWriteData	An array of " <i><FieldName>=<FieldValue></i> " strings. If the field is an index field, then the syntax of the field is instead " <i><FieldName><index>=<FieldValue></i> " where <i><index></i> indicates the zero based element of the index field.

Exceptions Events

No additional exceptions generated.
Additional Events can be generated :

OperationCompleteEvent

When a *printForms()* operation is completed an *OperationCompleteEvent* will be sent by J/XFS Printer Device Control to all registered *OperationCompleteListeners* with the status containing the following data

<i>operationID</i>	JXFS_O_PTR_WRITE_FORM_DATA
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_ERROR JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FIELD_SPEC_FAILURE JXFS_E_PTR_FORM_INVALID JXFS_E_PTR_FORM_NOT_FOUND JXFS_E_PTR_MEDIA_INVALID JXFS_E_PTR_MEDIA_NOT_FOUND JXFS_E_PTR_MEDIA_OVERFLOW JXFS_E_PTR_MEDIA_SKEWED JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

IntermediateEvent

If no media is present the J/XFS Printer Device Control will send an *IntermediateEvent* to all registered *IntermediateListeners* with the following data :

<i>operationID</i>	JXFS_O_PTR_WRITE_FORM_DATA
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

IntermediateEvent

If media is inserted and the operation can continue the J/XFS Printer Device Control will send an *IntermediateEvent* to all registered *IntermediateListeners* with the following data :

<i>operationID</i>	JXFS_O_PTR_WRITE_FORM_DATA
<i>identificationID</i>	The corresponding ID

reason JXFS_I_PTR_MEDIA_INSERTED
data none

IntermediateEvent

If a field error occurs during printing the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

operationID JXFS_O_PTR_WRITE_FORM_DATA
identificationID The corresponding ID
reason JXFS_E_PTR_FIELD_FAILURE
data JxfsFieldFailure

StatusEvent

When the status of the printer's paper supply changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_PAPER
<i>details</i>	JxfsPaperStatus

StatusEvent

When the status of the printer's toner supply changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_TONER
<i>details</i>	JxfsTonerStatus

StatusEvent

When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_MEDIA
<i>details</i>	JxfsMediaStatus

printRawData

Syntax	<i>identificationID printRawData(byte[] rawData, boolean inputData) throws JxfsException;</i>		
Description	This command is used to send raw data (a byte string of device dependent data) to the physical device. If no media is present the printer waits endless or until cancelled by the application. After a successful completion of this output operation without any input data (see parameter inputData), an <i>OperationCompleteEvent</i> is issued to inform the application of the results. If input data are expected (see parameter inputData) and can be sent to the client this will be done via a <i>RawDataEvent</i> event.		
Parameter	Type	Name	Meaning
	byte[]	rawData	Raw data to be sent to the printer.
	boolean	inputData	Indicates whether input data from the printer is expected in response to sending the raw data.
Exceptions	No additional exceptions generated.		

Events

Additional Events can be generated :

OCPtrRawDataEvent

When a *printRawData()* operation is completed an OCPtrRawDataEvent will be sent by J/XFS Printer Device Control to all registered OCRawDataListeners with the following data:

<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	byte[] inputData Input data sent by the printer

OperationCompleteEvent

When a *printRawData()* operation without expected input data is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data :

<i>operationID</i>	JXFS_O_PTR_WRITE_RAW_DATA
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

IntermediateEvent

If no media is present the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

<i>operationID</i>	JXFS_O_PTR_WRITE_RAW_DATA
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

IntermediateEvent

If media is inserted and the operation can continue the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

<i>operationID</i>	JXFS_O_PTR_WRITE_RAW_DATA
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_MEDIA_INSERTED
<i>data</i>	none

StatusEvent

When the status of the printer's paper supply changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_PAPER
<i>details</i>	JxfsPaperStatus

StatusEvent

When the status of the printer's toner supply changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_TONER
<i>details</i>	JxfsTonerStatus

StatusEvent

When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_MEDIA
<i>details</i>	JxfsMediaStatus

resetPrinter

Syntax
Description
Parameter
Exceptions
Events

identificationID **resetPrinter(void) throws JxfsException;**

The printer tries to reset.

None

No additional exceptions generated.

Additional Events can be generated :

OperationCompleteEvent

When a *resetPrinter()* operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:

<i>operationID</i>	JXFS_O_PTR_RESET_PRINTER
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

StatusEvent

When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_MEDIA
<i>details</i>	JxfsMediaStatus

4.3.4 Events

OCPtrFieldInfoEvent

Interface	<i>jxfs.events.OCPtrFieldInfoListener</i>		
Method	OCPtrFieldInfoOccurred(OCPtrFieldInfoEvent e);		
Description	Issued to indicate that a <i>getFieldDescription()</i> operation has completed.		
Properties	Type	Name	Meaning
	int	identificationID	The corresponding ID
	int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FORM_NOT_FOUND
	JxfsPtrField[]	data	Description of the requested fields

OCPtrFormInfoEvent

Interface	<i>jxfs.events.OCPtrFormInfoListener</i>		
Method	OCPtrFormInfoOccurred(OCPtrFormInfoEvent e);		
Description	Issued to indicate that a <i>getFormDescription()</i> operation has completed.		
Properties	Type	Name	Meaning
	int	identificationID	The corresponding ID
	int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FORM_NOT_FOUND
	JxfsPtrForm	data	Description of the requested form.

OCPtrFormListEvent

Interface	<i>jxfs.events.OCPtrFormListListener</i>		
Method	OCPtrFormListOccurred(OCPtrFormListEvent e);		
Description	Issued to indicate that a <i>formList()</i> operation has completed.		
Properties	Type	Name	Meaning
	int	identificationID	The corresponding ID
	int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NOFORMS
	String[]	data	List of the forms available on the printer

OCPtrMediaExtentEvent

Interface	<i>jxfs.events.OCPtrMediaExtentListener</i>		
Method	OCPtrMediaExtentOccurred(OCPtrMediaExtentEvent e);		
Description	Issued to indicate that a <i>mediaExtent()</i> operation has completed.		
Properties	Type	Name	Meaning
	int	identificationID	The corresponding ID
	int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_MEDIA_NOT_PRESENT
	JxfsPtrMediaE xtents	data	The extents of the inserted media.

OCPtrMediaInfoEvent

Interface	<i>jxfs.events.OCPtrMediaInfoListener</i>		
Method	OCPtrMediaInfoOccurred(OCPtrMediaInfoEvent e);		
Description	Issued to indicate that a <i>queryMedia()</i> operation has completed.		
Properties	Type	Name	Meaning
	int	identificationID	The corresponding ID
	int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_MEDIA_NOT_FOUND
	JxfsPtrMedia	data	Description of the requested media.

OCPtrMediaListEvent

Interface	<i>jxfs.events.MediaListListener</i>		
Method	OCPtrMediaListOccurred(OCPtrMediaListEvent e);		
Description	Issued to indicate that a <i>mediaList()</i> operation has completed.		
Properties	Type	Name	Meaning
	int	identificationID	The corresponding ID
	int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NOMEDIAS
	String[]	data	List of the media available on the printer

OCPtrRawDataEvent

Interface	<i>jxfs.events.RawDataListener</i>		
Method	OCPtrRawDataOccurred(OCPtrRawDataEvent e);		
Description	Issued to notify the client that a <i>printRawData()</i> command has completed with sending the expected input data.		
Properties	Type	Name	Meaning
	int	identificationID	The corresponding ID
	int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_MEDIA_NOT_PRESENT
	byte[]	data	input data sent by the printer

4.4 IJxfsEject

4.4.1 Summary

Property	Type	Access	Initialized after
inkStatus	JxfsThresholdStatus	R	
maxStackerCapability	JxfsPtrMaxStackerCapability	R	

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
ejectMedia	identificationID	<i>printRaw()</i> , <i>printForm()</i> , <i>readForm()</i> , <i>readImage()</i>
prepareEject	identificationID	<i>printRaw()</i> , <i>printForm()</i> , <i>readForm()</i> , <i>readImage()</i>

Event	May occur after
StatusEvent JXFS_S_PTR_INK JXFS_S_PTR_MEDIA	<i>ejectMedia()</i> , <i>prepareEject()</i> <i>ejectMedia()</i>
IntermediateEvent JXFS_I_PTR_MEDIA_TAKEN	<i>ejectMedia()</i> (media taken before successful retract)
OperationCompleteEvent JXFS_O_PTR_EJECT_MEDIA	<i>ejectMedia()</i>

4.4.2 Properties

See description of the support classes for the properties.

4.4.3 Methods

ejectMedia

Syntax	<i>identificationID ejectMedia(int mediaControl) throws JxfsException;</i>																								
Description	<p>This command is used to eject a form.</p> <p>The eject operation completes when the media is moved to the exit slot and an OperationCompleteEvent is sent.</p> <p>An IntermediateEvent is generated when the media has been taken by the user.</p>																								
Parameter	<table border="0"> <thead> <tr> <th style="text-align: left;">Type</th> <th style="text-align: left;">Name</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td><i>int</i></td> <td>mediaControl</td> <td> <p>Specifies the manner in which the media should be handled before ejecting, as a combination of the following flags :</p> <p>JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_STACK JXFS_PTR_CTRL_STAMP</p> </td> </tr> </tbody> </table>	Type	Name	Meaning	<i>int</i>	mediaControl	<p>Specifies the manner in which the media should be handled before ejecting, as a combination of the following flags :</p> <p>JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_STACK JXFS_PTR_CTRL_STAMP</p>																		
Type	Name	Meaning																							
<i>int</i>	mediaControl	<p>Specifies the manner in which the media should be handled before ejecting, as a combination of the following flags :</p> <p>JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_STACK JXFS_PTR_CTRL_STAMP</p>																							
Exceptions	No additional exceptions generated.																								
Events	<p>Additional Events can be generated :</p> <p>OperationCompleteEvent When a <i>ctrlMedia()</i> operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Field</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_PTR_EJECT_MEDIA</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table> <p>IntermediateEvent If the ejected media is taken the J/XFS Printer Device Control will send an IntermediateEvent after the OperationCompleteEvent to the registered IntermediateListener with the following value :</p> <table border="0"> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_PTR_EJECT_MEDIA</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>reason</i></td> <td>JXFS_I_PTR_MEDIA_TAKEN</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table> <p>StatusEvent When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Field</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td><i>status</i></td> <td>JXFS_S_PTR_MEDIA</td> </tr> <tr> <td><i>details</i></td> <td>JxfsMediaStatus</td> </tr> </tbody> </table>	Field	Value	<i>operationID</i>	JXFS_O_PTR_EJECT_MEDIA	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT	<i>data</i>	none	<i>operationID</i>	JXFS_O_PTR_EJECT_MEDIA	<i>identificationID</i>	The corresponding ID	<i>reason</i>	JXFS_I_PTR_MEDIA_TAKEN	<i>data</i>	none	Field	Value	<i>status</i>	JXFS_S_PTR_MEDIA	<i>details</i>	JxfsMediaStatus
Field	Value																								
<i>operationID</i>	JXFS_O_PTR_EJECT_MEDIA																								
<i>identificationID</i>	The corresponding ID																								
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT																								
<i>data</i>	none																								
<i>operationID</i>	JXFS_O_PTR_EJECT_MEDIA																								
<i>identificationID</i>	The corresponding ID																								
<i>reason</i>	JXFS_I_PTR_MEDIA_TAKEN																								
<i>data</i>	none																								
Field	Value																								
<i>status</i>	JXFS_S_PTR_MEDIA																								
<i>details</i>	JxfsMediaStatus																								

StatusEvent

When the status of the ink supply changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_INK
<i>details</i>	JxfsThresholdStatus

prepareEject

Syntax

identificationID prepareEject(int mediaControl) throws JxfsException;

Description

This command is used to prepare the ejecting of a form. The operation completes when the media is handled in the way defined by mediaControl. Then an OperationCompleteEvent is sent.

Parameter

Type	Name	Meaning
<i>int</i>	mediaControl	Specifies the manner in which the media should be prepared for ejecting, as a combination of the following flags : JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_PERFORATE JXFS_PTR_CTRL_STACK JXFS_PTR_CTRL_STAMP

Exceptions

No additional exceptions generated.

Events

Additional Events can be generated :

OperationCompleteEvent

When a *prepareEject()* operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:

Field	Value
<i>operationID</i>	JXFS_O_PTR_PREPARE_EJECT
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

StatusEvent

When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_MEDIA
<i>details</i>	JxfsMediaStatus

StatusEvent

When the status of the ink supply changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_INK
<i>details</i>	JxfsThresholdStatus

4.4.4 Events

No additional class specific events.

4.5 IJxfsRetract

4.5.1 Summary

Property	Type	Access	Initialized after
inkStatus	JxfsThresholdStatus	R	
maxRetractCapability	JxfsPtrMaxRetract Capability	R	
retractBinStatus	JxfsThresholdStatus	R	
retractCount	JxfsPtrRetractCount	R/W	

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
<i>setProperty</i>	<i>Property</i>	
retractMedia	identificationID	

Event	May occur after
StatusEvent JXFS_S_PTR_INK JXFS_S_PTR_RETRACT_BIN JXFS_S_PTR_RETRACT_COUNT	<i>retractMedia()</i> <i>retractMedia()</i> <i>retractMedia()</i>
IntermediateEvent JXFS_I_PTR_MEDIA_TAKEN	<i>retractMedia()</i> (media taken before successful retract)
OperationCompleteEvent JXFS_O_PTR_RETRACT_MEDIA	<i>retractMedia()</i>

4.5.2 Properties

See description of the support classes for the properties.

4.5.3 Methods

retractMedia

Syntax	<i>identificationID retractMedia(int mediacontrol) throws JxfsException;</i>																								
Description	<p>This command is used to control a form drawn in by the device (e.g. after reading or in case of termination of an application request). If an eject operation is specified, it completes when the media is moved to the exit slot and an OperationCompleteEvent is sent. An AsynchronousEvent is generated when the media has been taken by the user.</p>																								
Parameter	<table border="0"> <thead> <tr> <th style="text-align: left;">Type</th> <th style="text-align: left;">Name</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td><i>int</i></td> <td>mediaControl</td> <td> <p>Specifies the manner in which the media should be handled before retracting, as a combination of the following flags :</p> <p>JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_STAMP</p> </td> </tr> </tbody> </table>	Type	Name	Meaning	<i>int</i>	mediaControl	<p>Specifies the manner in which the media should be handled before retracting, as a combination of the following flags :</p> <p>JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_STAMP</p>																		
Type	Name	Meaning																							
<i>int</i>	mediaControl	<p>Specifies the manner in which the media should be handled before retracting, as a combination of the following flags :</p> <p>JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_STAMP</p>																							
Exceptions	No additional exceptions generated.																								
Events	<p>Additional Events can be generated :</p> <p>OperationCompleteEvent When a <i>retractMedia()</i> operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Field</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_PTR_RETRACT_MEDIA</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table> <p>IntermediateEvent If the ejected media is taken before the retract succeeded the J/XFS Printer Device Control will send an IntermediateEvent to the registered IntermediateListener with the following value :</p> <table border="0"> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_PTR_RETRACT_MEDIA</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>reason</i></td> <td>JXFS_I_PTR_MEDIA_TAKEN</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table> <p>StatusEvent When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Field</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td><i>status</i></td> <td>JXFS_S_PTR_MEDIA</td> </tr> <tr> <td><i>details</i></td> <td>JxfsMediaStatus</td> </tr> </tbody> </table>	Field	Value	<i>operationID</i>	JXFS_O_PTR_RETRACT_MEDIA	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT	<i>data</i>	none	<i>operationID</i>	JXFS_O_PTR_RETRACT_MEDIA	<i>identificationID</i>	The corresponding ID	<i>reason</i>	JXFS_I_PTR_MEDIA_TAKEN	<i>data</i>	none	Field	Value	<i>status</i>	JXFS_S_PTR_MEDIA	<i>details</i>	JxfsMediaStatus
Field	Value																								
<i>operationID</i>	JXFS_O_PTR_RETRACT_MEDIA																								
<i>identificationID</i>	The corresponding ID																								
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FLUSH_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT																								
<i>data</i>	none																								
<i>operationID</i>	JXFS_O_PTR_RETRACT_MEDIA																								
<i>identificationID</i>	The corresponding ID																								
<i>reason</i>	JXFS_I_PTR_MEDIA_TAKEN																								
<i>data</i>	none																								
Field	Value																								
<i>status</i>	JXFS_S_PTR_MEDIA																								
<i>details</i>	JxfsMediaStatus																								

StatusEvent

When the status of the ink supply changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_INK
<i>details</i>	JxfsThresholdStatus

4.5.4 Events

No additional class specific events.

4.6 IxfsMediaTurn

4.6.1 Summary

Property	Type	Access	Initialized after
ctrlTurnCapability	JxfsPtrCtrlTurnCapability	R	

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
atpBackward	identificationID	
atpForward	identificationID	
turnMedia	identificationID	

Event	May occur after
OperationCompleteEvent JXFS_O_PTR_ATP_BACKWARD JXFS_O_PTR_ATP_FORWARD JXFS_O_PTR_TURN_MEDIA	<i>atpBackward()</i> <i>atpForward()</i> <i>turnMedia()</i>

4.6.2 Properties

See description of the support classes for the properties.

4.6.3 Methods

atpBackward

Syntax
Description
Parameter
Exceptions
Events

identificationID atpBackward() throws JxfsException;
This command is used to turn the page of the passbook backward.
none
No additional exceptions generated.
Additional Events can be generated :
OperationCompleteEvent
When a *atpBackward()* operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:

Field	Value
<i>operationID</i>	JXFS_O_PTR_ATP_BACKWARD
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

atpForward

Syntax
Description
Parameter
Exceptions
Events

identificationID atpForward() throws JxfsException;
This command is used to turn the page of the passbook forward.
none
No additional exceptions generated.
Additional Events can be generated :
OperationCompleteEvent
When a *atpForward()* operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:

Field	Value
<i>operationID</i>	JXFS_O_PTR_ATP_FORWARD
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

turnMedia

Syntax
Description
Parameter
Exceptions
Events

identificationID turnMedia() throws JxfsException;
This command is used to turn the inserted media.
none
No additional exceptions generated.
Additional Events can be generated :
OperationCompleteEvent
When a *turnMedia()* operation is completed an OperationCompleteEvent will be sent by J/XFS Printer Device Control to all registered OperationCompleteListeners with the following data:

Field	Value
<i>operationID</i>	JXFS_O_PTR_TURN_MEDIA
<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_MEDIA_TURN_FAIL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

4.7 IxfsRead

4.7.1 Summary

Property	Type	Access	Initialized after
lampStatus	JxfsPtrLampStatus	R	
readFormCapability	JxfsPtrReadForm Capability	R	
readImageCapability	JxfsPtrReadImage Capability	R	

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
readForm	identificationID	
readImage	identificationID	

Event	May occur after
StatusEvent JXFS_S_PTR_LAMP	<i>readForm()</i> , <i>readImage()</i>
IntermediateEvent JXFS_I_PTR_NO_MEDIA_PRESENT	<i>readForm()</i> , <i>readImage()</i>
OCPtrReadDataEvent	<i>readForm()</i>
OCPtrReadImageEvent	<i>readImage()</i>

4.7.2 Properties

See description of the support classes for the properties.

4.7.3 Methods

readForm

Syntax *identificationID readForm(formName) throws JxfsException;*
Description This method reads the form with the name *formName*, using the appropriate information of *formsConfig*. After a successful completion of this input operation, a *ReadDataEvent* is issued to inform the application of the results. If no media is present the printer waits endlessly or until cancelled by the application.

Parameter

Type	Name	Meaning
String	formName	Name of the form to be read.

Exceptions No additional exceptions generated.
Events Additional Events can be generated :

OCPtrReadDataEvent

When a *readForm()* operation is completed an *OCPtrReadDataEvent* will be sent by J/XFS Printer Device Control to all registered *OCPtrReadDataListeners* with the fields that have been read.

<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_ERROR JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FIELD_SPEC_FAILURE JXFS_E_PTR_FORM_INVALID JXFS_E_PTR_FORM_NOT_FOUND JXFS_E_PTR_MEDIA_INVALID JXFS_E_PTR_MEDIA_NOT_FOUND JXFS_E_PTR_MEDIA_OVERFLOW JXFS_E_PTR_MEDIA_SKEWED JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	String [] readData Set to an array of "<FieldName>=<FieldValue>" strings.

IntermediateEvent

If no media is present the J/XFS Printer Device Control will send an *IntermediateEvent* to all registered *IntermediateListeners* with the following data :

<i>operationID</i>	JXFS_O_PTR_READ_FORM_DATA
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

IntermediateEvent

If media is inserted and the operation can continue the J/XFS Printer Device Control will send an *IntermediateEvent* to all registered *IntermediateListeners* with the following data :

<i>operationID</i>	JXFS_O_PTR_READ_FORM_DATA
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_MEDIA_INSERTED
<i>data</i>	none

IntermediateEvent

If a field error occurs during printing the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

<i>operationID</i>	JXFS_O_PTR_READ_FORM_DATA
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_E_PTR_FIELD_FAILURE
<i>data</i>	JxfsFieldFailure

StatusEvent

When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
<i>status</i>	JXFS_S_PTR_MEDIA
<i>details</i>	JxfsMediaStatus

readImage

Syntax Description

identificationID **readImage()** throws *JxfsException*;

This method is used to retrieve image data from the current media.

After a successful completion of this input operation, a *ReadImageEvent* is issued to inform the application of the results. If no media is present the printer waits endlessly or until cancelled by the application.

Parameter Exceptions Events

None

No additional exceptions generated.

Additional Events can be generated :

OCPtrReadImageEvent

When a *readImage()* operation is completed an OCPtrReadImageEvent will be sent by J/XFS Printer Device Control to all registered OCPtrReadImageListeners with the fields that have been read.

<i>identificationID</i>	The corresponding ID
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_PTR_NO_MEDIA_PRESENT
<i>data</i>	JxfsPtrImage readData Set to an array of "<FieldName>=<FieldValue>" strings.

IntermediateEvent

If no media is present the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

<i>operationID</i>	JXFS_O_PTR_READ_IMAGE
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_NO_MEDIA_PRESENT
<i>data</i>	none

IntermediateEvent

If media is inserted and the operation can continue the J/XFS Printer Device Control will send an IntermediateEvent to all registered IntermediateListeners with the following data :

<i>operationID</i>	JXFS_O_PTR_READ_IMAGE
<i>identificationID</i>	The corresponding ID
<i>reason</i>	JXFS_I_PTR_MEDIA_INSERTED
<i>data</i>	none

StatusEvent

When the status of the media changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field	Value
-------	-------

status JXFS_S_PTR_MEDIA
details JxfsMediaStatus

StatusEvent

When the status of the imaging lamp changes a StatusEvent is sent to all registered StatusEventListeners with the following data :

Field **Value**
status JXFS_S_PTR_LAMP
details JxfsPtrLampStatus

4.7.4 Events

OCPtrReadDataEvent

Interface
Method
Description

jxfs.events.OCPtrReadDataListener

OCPtrReadDataOccurred(OCPtrReadDataEvent *e*);

Issued to notify the client that a *readForm()* command has completed with sending the expected input data.

Properties

Type	Name	Meaning
int	identification ID	The corresponding ID
int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FORM_NOT_FOUND
String[]	data	Set to an array of " <i><FieldName>=<FieldValue></i> " strings. If the field is an index field, then the syntax of the field is instead " <i><FieldName><index>=<FieldValue></i> " where <i><index></i> indicates the zero-based element of the index field.

OCPtrReadImageEvent

Interface
Method
Description

jxfs.events.OCPtrReadImageListener

OCPtrReadImageOccurred(OCPtrReadImageEvent *e*);

Issued to notify the client that a *readImage()* command has completed with sending the expected input data.

Properties

Type	Name	Meaning
int	identification ID	The corresponding ID
int	result	JXFS_RC_SUCCESSFUL JXFS_E_PTR_FIELD_NOT_FOUND JXFS_E_PTR_FORM_NOT_FOUND
JxfsPtrImage	data	Image data from the current media.

5 Support Classes

Summary

Class	Description
JxfsPtrCtrlMediaCapability	Specifies Control Media capabilities.
JxfsPtrCtrlTurnCapability	Specifies if the printer is able to turn media.
JxfsPtrExtentCapability	Specifies if the printer is able to measure the extent of the media.
JxfsPtrField	Specifies the description of a field.
JxfsPtrFieldFailure	Specifies the failure that occurred during field processing at readForm or writeForm
JxfsPtrForm	Specifies the description of a form.
JxfsPtrFormsConfig	Specifies the configuration necessary to print a form.
JxfsPtrImage	Specifies the data of the read image.
JxfsPtrMaxRetractCapability	Specifies maximum retract capabilities.
JxfsPtrMaxStackerCapability	Specifies maximum stacker capabilities.
JxfsPtrMedia	Specifies the description of a media.
JxfsPtrReadFormCapability	Specifies read form capabilities.
JxfsPtrReadImageCapability	Specifies read image capabilities.
JxfsPtrWriteFormCapability	Specifies write form capabilities.
JxfsPtrRetractCount	Specifies retract count .

5.1 JxfsPtrCtrlMediaCapability

This class specifies the control media capabilities of the printer.

5.1.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
ctrlMediaCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrCtrlMediaCapability	ctrlMediaCapability	int

Method	Return	Meaning
<i>getProperty</i>	<i>Property</i>	
isCtrlAlarmSupported	boolean	
isCtrlStampSupported	boolean	
isCtrlCutSupported	boolean	
isCtrlEjectSupported	boolean	
isCtrlFlushSupported	boolean	
isCtrlPartialCutSupported	boolean	
isCtrlPerforateSupported	boolean	
isCtrlRetractSupported	boolean	
isCtrlSkipSupported	boolean	
isCtrlStackSupported	boolean	

Event	May occur after
none	

5.1.2 Properties

ctrlMediaCapability (R)

Type	<i>int</i>
Initial Value	0
Description	Specifies the manner in which media can be controlled, as a combination of the following bit flags: JXFS_PTR_CTRL_ALARM JXFS_PTR_CTRL_STAMP JXFS_PTR_CTRL_CUT JXFS_PTR_CTRL_EJECT JXFS_PTR_CTRL_FLUSH JXFS_PTR_CTRL_PARTIALCUT JXFS_PTR_CTRL_PERFORATE JXFS_PTR_CTRL_RETRACT JXFS_PTR_CTRL_SKIP JXFS_PTR_CTRL_STACK

5.1.3 Methods

isCtrlAlarmSupported

Syntax	<i>boolean isCtrlAlarmSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to issue an alarm (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_ALARM).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlStampSupported

Syntax	<i>boolean isCtrlStampSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to stamp on the media (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_STAMP).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlCutSupported

Syntax	<i>boolean isCtrlCutSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to cut the media (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_CUT).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlEjectSupported

Syntax	<i>boolean isCtrlEjectSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to eject the media (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_EJECT).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlFlushSupported

Syntax	<i>boolean isCtrlFlushSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to store data internally and then print it after a flush (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_FLUSH).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlPartialCutSupported

Syntax	<i>boolean isCtrlPartialCutSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to cut the media partially (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_PARTIALCUT).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlPerforateSupported

Syntax	<i>boolean isCtrlPerforateSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to perforate the media (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_PERFORATE).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlRetractSupported

Syntax	<i>boolean isCtrlRetractSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to retract the media (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_RETRACT).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlSkipSupported

Syntax	<i>boolean isCtrlSkipSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to skip the media to the next mark (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_SKIP).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlStackSupported

Syntax	<i>boolean isCtrlStackSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to stack the media (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_STACK).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

5.2 JxfsPtrCtrlTurnCapability

This class specifies the turn media capabilities of the printer.

5.2.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
ctrlTurnMediaCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrCtrlTurnCapability	ctrlTurnMediaCapability	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
isCtrlATPBackwardSupported	boolean	
isCtrlATPForwardSupported	boolean	
isCtrlMediaTurnSupported	boolean	

Event	May occur after
none	

5.2.2 Properties

ctrlTurnMediaCapability (R)

Type	<i>int</i>
Initial Value	0
Description	Specifies the manner in which media can be controlled, as a combination of the following bit flags: JXFS_PTR_CTRL_ATP_BACKWARD JXFS_PTR_CTRL_ATP_FORWARD JXFS_PTR_CTRL_TURNMEDIA

5.2.3 Methods

isCtrlATPBackwardSupported

Syntax	<i>boolean isCtrlBackwardSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to turn one page backward (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_ATP_BACKWARD).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlATPForwardSupported

Syntax	<i>boolean isCtrlATPForwardSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to turn one page forward (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_ATP_FORWARD).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isCtrlTurnMediaSupported

Syntax	<i>boolean isCtrlTurnMediaSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to turn the media (the <i>ctrlMediaCapability</i> property contains the value JXFS_PTR_CTRL_TURNMEDIA).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

5.3 JxfsPtrExtentCapability

This class specifies the extent capability of the printer.

5.3.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
extentCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrExtentCapability	extentCapability	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
isExtHorizontalSupported	boolean	
isExtVerticalSupported	boolean	

Event	May occur after
none	

5.3.2 Properties

extentCapability (R)

Type	<i>int</i>
Initial Value	0
Description	Specifies whether the device is able to measure the inserted media. Depending on the device capability <i>extentCapability</i> will be set as a combination of the following values:
Value	Meaning
JXFS_PTR_EXT_HORIZONTAL	Device has horizontal size detection capability.
JXFS_PTR_EXT_VERTICAL	Device has vertical size detection capability.

5.3.3 Methods

isExtHorizontalSupported

Syntax	<i>boolean isExtHorizontalSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer is able to measure the horizontal size of the inserted media (the <i>extentCapability</i> property contains the value JXFS_PTR_EXT_HORIZONTAL)
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isExtVerticalSupported

Syntax	<i>boolean isExtVerticalSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer is able to measure the vertical size of the inserted media (the <i>extentCapability</i> property contains the value JXFS_PTR_EXT_VERTICAL)
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

5.4 JxfsPtrField

The JxfsPtrField class contains the properties of a field on a specified form.

5.4.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
access	int	R	
classType	int	R	
fieldName	String	R	
format	String	R	
indexCount	int	R	
initialValue	String	R	
overflow	int	R	
type	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrField	access	int
	classType	int
	fieldName	String
	format	String
	indexCount	int
	initialValue	String
	overflow	int
	type	String

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.4.2 Properties

access (R)

Type	<i>int</i>						
Initial Value	0						
Description	Indicates whether the field is to be used for input, output or both and can be a combination of the following values:						
	<table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_PTR_FRM_ACCESS_READ</td> <td>Field is used for input.</td> </tr> <tr> <td>JXFS_PTR_FRM_ACCESS_WRITE</td> <td>Field is used for output.</td> </tr> </tbody> </table>	Value	Meaning	JXFS_PTR_FRM_ACCESS_READ	Field is used for input.	JXFS_PTR_FRM_ACCESS_WRITE	Field is used for output.
Value	Meaning						
JXFS_PTR_FRM_ACCESS_READ	Field is used for input.						
JXFS_PTR_FRM_ACCESS_WRITE	Field is used for output.						

classType (R)

Type	<i>int</i>								
Initial Value	JXFS_PTR_FRM_CLASS_OPTIONAL								
Description	Indicates the class of the field and can be one of the following values:								
	<table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_PTR_FRM_CLASS_STATIC</td> <td>Field data cannot be set by the application.</td> </tr> <tr> <td>JXFS_PTR_FRM_CLASS_OPTIONAL</td> <td>Field data can be set by the application.</td> </tr> <tr> <td>JXFS_PTR_FRM_CLASS_REQUIRED</td> <td>Field data must be set by the application.</td> </tr> </tbody> </table>	Value	Meaning	JXFS_PTR_FRM_CLASS_STATIC	Field data cannot be set by the application.	JXFS_PTR_FRM_CLASS_OPTIONAL	Field data can be set by the application.	JXFS_PTR_FRM_CLASS_REQUIRED	Field data must be set by the application.
Value	Meaning								
JXFS_PTR_FRM_CLASS_STATIC	Field data cannot be set by the application.								
JXFS_PTR_FRM_CLASS_OPTIONAL	Field data can be set by the application.								
JXFS_PTR_FRM_CLASS_REQUIRED	Field data must be set by the application.								

fieldName (R)

Type	<i>String</i>
Initial Value	empty String
Description	Indicates the name of the field.

format (R)

Type	<i>String</i>
Initial Value	empty String
Description	Indicates the format as defined in the form for this field.

indexCount (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the number of entries for an index field. A value of zero indicates that this field is not an index field. Index fields are typically used to present information in a tabular fashion.

initialValue (R)

Type	<i>String</i>
Initial Value	empty String
Description	Indicates the initial value of the field. When the form is printed, this value will be used if another value is not provided.

overflow (R)

Type	<i>int</i>
Initial Value	JXFS_PTR_FRM_OVF_TRUNCATE
Description	Indicates how an overflow of the field data should be handled and can be one of the following values:
Value	Meaning
JXFS_PTR_FRM_OVF_TERMINATE	Return an error and terminate printing the form.
JXFS_PTR_FRM_OVF_TRUNCATE	Truncate field data to fit in the field.
JXFS_PTR_FRM_OVF_BEST_FIT	Fit text in the field.
JXFS_PTR_FRM_OVF_OVERWRITE	Print field data beyond the extents of the field boundary.
JXFS_PTR_FRM_OVF_WORDWRAP	If field can hold more than one line the text is wrapped around.

type (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the type of the field and can be one of the following values:
Value	Meaning
JXFS_PTR_FRM_FIELD_BARCODE	Barcode field.
JXFS_PTR_FRM_FIELD_GRAPHIC	Graphic field
JXFS_PTR_FRM_FIELD_MICR	Magnetic Ink Character Recognition field.
JXFS_PTR_FRM_FIELD_MSF	Magnetic Stripe Facility field.
JXFS_PTR_FRM_FIELD_OCR	Optical Recognition Character field.
JXFS_PTR_FRM_FIELD_PAGEMARK	Page Mark field.
JXFS_PTR_FRM_FIELD_TEXT	Text field.

5.5 JxfsPtrFieldFailure

This class specifies the control media capabilities of the printer.

5.5.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
formName	String	R	
fieldName	String	R	
fieldFailure	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrFieldFailure	formName	String
	fieldName	String
	fieldFailure	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.5.2 Properties

formName (R)

Type	<i>String</i>
Initial Value	empty String
Description	Specifies the name of the form at which the error occurred

fieldName (R)

Type	<i>String</i>
Initial Value	empty String
Description	Specifies the name of the field at which the error occurred

fieldFailure (R)

Type	<i>String</i>
Initial Value	empty String
Description	Specifies the type of failure and can be one of the following: JXFS_E_PTR_FIELD_GRAPHIC JXFS_E_PTR_FIELD_HW_ERROR JXFS_E_PTR_FIELD_NOT_READ JXFS_E_PTR_FIELD_NOT_WRITE JXFS_E_PTR_FIELD_OVERFLOW JXFS_E_PTR_FIELD_REQUIRED JXFS_E_PTR_FIELD_STATIC_OVWR JXFS_E_PTR_FIELD_TYPE_NOT_SUPPORTED

5.6 JxfsPtrForm

The JxfsPtrForm class contains the properties of a specified form.

5.6.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
alignment	int	R	
base	int	R	
fields	String[]	R	
formName	String	R	
height	int	R	
offsetX	int	R	
offsetY	int	R	
orientation	int	R	
unitX	int	R	
unitY	int	R	
userPrompt	String	R	
versionMajor	int	R	
versionMinor	int	R	
width	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrForm	alignment	int
	base	int
	fields	String
	formName	String
	height	int
	offsetX	int
	offsetY	int
	orientation	int
	unitX	int
	unitY	int
	userPrompt	String
	versionMajor	int
	versionMinor	int
width	int	

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.6.2 Properties

alignment (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the relative alignment of the form on the media as one of the following values :
Value	Meaning
JXFS_PTR_ALN_TOPLEFT	Align the form to top left of media
JXFS_PTR_ALN_TOPRIGHT	Align the form to top right of media
JXFS_PTR_ALN_BOTTOMLEFT	Align the form to bottom left of media
LEFT	Align the form to bottom right of media
JXFS_PTR_ALN_BOTTOMRIGHT	

base (R)

Type	<i>int</i>
Initial Value	JXFS_PTR_FRM_MM
Description	Indicates the base unit of measurement of the form as one of the following values:
Value	Meaning
JXFS_PTR_FRM_INCH	Base unit is inches.
JXFS_PTR_FRM_MM	Base unit is millimeters.
JXFS_PTR_FRM_ROW COLUMN	Base unit is rows and columns.

fields (R)

Type	<i>String[]</i>
Initial Value	empty String[]
Description	Indicates the field names on the form

formName (R)

Type	<i>String</i>
Initial Value	empty String
Description	Indicates the name of the form.

height (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the height of the form in terms of the base vertical resolution.

offsetX (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the horizontal offset of the position of the top-left corner of the form, relative to the left or right edge specified by <i>alignment</i> . This value is specified in terms of the base horizontal resolution and is always positive.

offsetY (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the vertical offset of the position of the top-left corner of the form, relative to the left or right edge specified by <i>alignment</i> . This value is specified in terms of the base vertical resolution and is always positive.

orientation (R)

Type	<i>int</i>	
Initial Value	JXFS_PTR_FRM_PORTRAIT	
Description	Indicates the orientation of the form and can be one of the following values :	
	Value	Meaning
	JXFS_PTR_FRM_LAND	Orientation of the form is landscape.
	SCAPE	
	JXFS_PTR_FRM_PORTRAIT	Orientation of the form is portrait.

unitX (R)

Type	<i>int</i>	
Initial Value	1	
Description	Indicates the horizontal resolution of the base units as a fraction of the <i>base</i> value.	

unitY (R)

Type	<i>int</i>	
Initial Value	1	
Description	Indicates the vertical resolution of the base units as a fraction of the <i>base</i> value.	

userPrompt (R)

Type	<i>String</i>	
Initial Value	empty String	
Description	Indicates the user prompt string.	

versionMajor (R)

Type	<i>int</i>	
Initial Value	0	
Description	Indicates the major version of the form.	

versionMinor (R)

Type	<i>int</i>	
Initial Value	0	
Description	Indicates the minor version of the form.	

width (R)

Type	<i>int</i>	
Initial Value	0	
Description	Indicates the width of the form in terms of the base horizontal resolution.	

5.7 JxfsPtrFormsConfig

This class contains properties and methods to configure the usage of forms.

5.7.1 Summary

Implements : --

Extends : *JxfsType*

Property	Type	Access	Initialized after
alignment	int	R/W	
base	int	R/W	
formsDescriptionList	JxfsPtrForm[]	R/W	
mediaDescriptionList	JxfsPtrMedia[]	R/W	
offsetX	int	R/W	
offsetY	int	R/W	
unitX	int	R/W	
unitY	int	R/W	

Constructor	Parameter	Parameter-Type
JxfsPtrFormsConfig	<i>none</i>	
JxfsPtrFormsConfig	alignment	int
	base	int
	offsetX	int
	offsetY	int
	unitX	int
	unitY	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
<i>setProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.7.2 Properties

alignment (R/W)

Type	<i>int</i>	
Initial Value	JXFS_PTR_ALN_USEFORMDEFN	
Description	Indicates the relative alignment of the form on the media, as one of the following values:	
	Value	Meaning
	JXFS_PTR_ALN_BOTTOMLE	Align the form to bottom left of media.
	FT	
	JXFS_PTR_ALN_BOTTOMRI	Align the form to bottom right of media.
	GHT	
	JXFS_PTR_ALN_TOPLEFT	Align the form to top left of media.
	JXFS_PTR_ALN_TOPRIGHT	Align the form to top right of media.
	JXFS_PTR_ALN_USEFORMD	Use alignment specified in the form definition.
	EFN	

base (R/W)

Type	<i>int</i>	
Initial Value	JXFS_PTR_FRM_MM	
Description	Indicates the base unit of measurement of the media and can be one of the following values:	
	Value	Meaning
	JXFS_PTR_FRM_INCH	Base unit is inches.
	JXFS_PTR_FRM_MM	Base unit is millimeters.
	JXFS_PTR_FRM_ROW	Base unit is rows and columns.
	COLUMN	

formsDescriptionList (R/W)

Type	<i>JxfsPtrForm[]</i>	
Initial Value	an empty JxfsPtrForm List	
Description	Set to an uninitialized JxfsPtrForm List. The JxfsPtrForm object will be initialized or updated when the form specified in <i>formName</i> is accessed. If <i>formName</i> has been set and if an operation that requires the form has not been performed and a <i>queryForm()</i> is requested then the form will be accessed, the JxfsPtrForm will be initialized and a DataEvent will be sent to the application. If a <i>queryForm()</i> is requested and the JxfsPtrForm has already been initialized then a DataEvent is sent without re-accessing the form. The JxfsPtrFormList can only be set by the Device Service internally.	
Event	When a <i>queryForm()</i> operation is completed a DataEvent event will be sent by J/XFS Printer Device Control to all registered DataListeners with a status value of:	
	Value	Meaning
	JXFS_S_PTR_FORM_	<i>formsDescriptionList</i> has changed
	DESCR_LIST	

mediaDescriptionList (R/W)

Type	<i>JxfsPtrMedia[]</i>	
Initial Value	an empty JxfsPtrMedia List	
Description	Set to an uninitialized JxfsPtrMedia. The JxfsPtrMedia object will be initialized or updated when the field specified in <i>mediaName</i> is accessed. If <i>mediaName</i> has been set and if an operation that requires the field has not been performed and a <i>queryMedia()</i> is requested then the media description will be accessed, the JxfsPtrMedia will be initialized and a DataEvent will be sent to the application. If a <i>queryMedia()</i> is requested and the JxfsPtrMedia has already been initialized then a DataEvent is sent without re-accessing the media	

description. The `JxfsPtrMediaList` can only be set by the Device Service internally.

Event When a `queryMedia()` operation is completed a `DataEvent` event will be sent by J/XFS Printer Device Control to all registered `DataListeners` with a status value of:

Value	Meaning
<code>JXFS_S_PTR_MEDIA_DESCR_LIST</code>	<code>mediaDescriptionList</code> has changed

offsetX (R/W)

Type
Initial Value
Description

int
`JXFS_PTR_OFFSET_USEFORMDEFN`
Indicates the horizontal offset of the position of the top-left corner of the form, relative to the left or right edge specified by the property *alignment*. This value is specified in terms of the base horizontal resolution and is always positive. A value of `JXFS_PTR_OFFSET_USEFORMDEFN` specifies that the *offsetX* from the form definition should be used.

offsetY (R/W)

Type
Initial Value
Description

int
`JXFS_PTR_OFFSET_USEFORMDEFN`
Indicates the vertical offset of the position of the top-left corner of the form, relative to the left or right edge specified by the property *alignment*. This value is specified in terms of the base vertical resolution and is always positive. A value of `JXFS_PTR_OFFSET_USEFORMDEFN` specifies that the *offsetY* from the form definition should be used.

unitX (R/W)

Type
Initial Value
Description

int
1
Indicates the horizontal resolution of the base units as a fraction of the property *base*.

unitY (R/W)

Type
Initial Value
Description

int
1
Indicates the vertical resolution of the base units as a fraction of the property *base*.

5.8 JxfsPtrImage

This class specifies the data of the image read by the readImage method.

5.8.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
imageData	byte[]	R	
imageType	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrImage	imageData	byte[]
	imageType	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.8.2 Properties

imageData (R)

Type *byte[]*
Initial Value empty byte[]
Description Image data from the current media.

imageType (R)

Type *int*
Initial Value 0
Description Set to the image data format and can be one of the following values:
JXFS_PTR_IMAGE_TIF
Image data is in TIF format.
JXFS_PTR_IMAGE_MTF
Image data is in MTF format.
JXFS_PTR_IMAGE_BMP
Image data is in BMP format.

5.9 JxfsPtrMaxRetractCapability

This class specifies the maximum possible number of retracts the printer can perform.

5.9.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
maxRetractCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrMaxRetractCapability	maxRetractCapability	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.9.2 Properties

maxRetractCapability (R)

Type	<i>int</i>
Initial Value	0
Description	Specifies the maximum number of media items that the retract bin can hold (zero if not available).

5.10 JxfsPtrMaxStackerCapability

This class defines the maximum number of media items that the stacker can hold.

5.10.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
maxStackerCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrMaxStackerCapability	maxStackerCapability	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.10.2 Properties

maxStackerCapability (R)

Type	<i>int</i>
Initial Value	0
Description	Specifies the maximum number of media items that the stacker can hold (zero if not available).
Event	No additional events
Exceptions	Some possible JxfsException <i>value codes</i> . See section on JxfsExceptions for other JxfsException value codes.
Value	Meaning

5.11 JxfsPtrMedia

The JxfsPtrMedia class contains the properties of a specified media.

5.11.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
base	int	R	
foldtype	int	R	
lineCount	int	R	
mediaType	int	R	
pageCount	int	R	
printAreaHeight	int	R	
printAreaWidth	int	R	
printAreaX	int	R	
printAreaY	int	R	
restrictedAreaHeight	int	R	
restrictedAreaWidth	int	R	
restrictedAreaX	int	R	
restrictedAreaY	int	R	
sizeHeight	int	R	
sizeWidth	int	R	
stagger	int	R	
unitX	int	R	
unitY	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrMedia	base	int
	foldtype	int
	lineCount	int
	mediaType	int
	pageCount	int
	printAreaHeight	int
	printAreaWidth	int
	printAreaX	int
	printAreaY	int
	restrictedAreaHeight	int
	restrictedAreaWidth	int
	restrictedAreaX	int
	restrictedAreaY	int
	sizeHeight	int
	sizeWidth	int
	stagger	int
	unitX	int
	unitY	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.11.2 Properties

base (R)

Type *int*

Initial Value	JXFS_PTR_FRM_MM
Description	Indicates the base unit of measurement of the form as one of the following values:
Value	Meaning
JXFS_PTR_FRM_INCH	Base unit is inches.
JXFS_PTR_FRM_MM	Base unit is millimeters.
JXFS_PTR_FRM_ROW	Base unit is rows and columns.
COLUMN	

foldType (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the type of fold for a media of type JXFS_PTR_FRM_MEDIA_PASSBOOK and can be one of the following values :
Value	Meaning
JXFS_PTR_FRM_FOLD_HORIZONTAL	Passbook has horizontal fold.
JXFS_PTR_FRM_FOLD_NONE	Passbook has no fold.
JXFS_PTR_FRM_FOLD_VERTICAL	Passbook has vertical fold.

lineCount (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the number of lines on a page for media of type JXFS_PTR_FRM_MEDIA_PASSBOOK .

mediaType (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the type of media as one of the following values :
Value	Meaning
JXFS_PTR_FRM_MEDIA_GENERIC	Generic media, i.e., single sheet.
JXFS_PTR_FRM_MEDIA_MULTIPART	Multipart media.
JXFS_PTR_FRM_MEDIA_PASSBOOK	Passbook media.

pageCount (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the number of pages on a page for media of type JXFS_PTR_FRM_MEDIA_PASSBOOK .

printAreaHeight (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the printable area height of the media in terms of the base vertical resolution.

printAreaWidth (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the printable area width of the media in terms of the base

horizontal resolution.

printAreaX (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the horizontal offset of the printable area relative to the top left corner of the media in terms of the base horizontal resolution.

printAreaY (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the vertical offset of the printable area relative to the top left corner of the media in terms of the base vertical resolution.

restrictedAreaHeight (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the restricted area height of the media in terms of the base vertical resolution.

restrictedAreaWidth (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the restricted area width of the media in terms of the base horizontal resolution.

restrictedAreaX (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the horizontal offset of the restricted area relative to the top left corner of the media in terms of the base horizontal resolution.

restrictedAreaY (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the vertical offset of the restricted area relative to the top left corner of the media in terms of the base vertical resolution.

sizeHeight (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the height of the media in terms of the base vertical resolution.

sizeWidth (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the width of the media in terms of the base horizontal resolution.

stagger (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the staggering from the top in terms of the base vertical resolution for a media of type JXFS_PTR_FRM_MEDIA_PASSBOOK .

unitX (R)

Type	<i>int</i>
Initial Value	1
Description	Indicates the horizontal resolution of the base units as a fraction of the <i>base</i> value.

unitY (R)

Type	<i>int</i>
Initial Value	1
Description	Indicates the vertical resolution of the base units as a fraction of the <i>base</i> value.

5.12 JxfsPtrMediaExtents

This class contains the properties to return a media's extents.

5.12.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
sizeX	int	R	
sizeY	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrMediaExtents	sizeX	int
	sizeY	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

5.12.2 Properties

sizeX (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the width of the media in terms of the base horizontal resolution.

sizeY (R)

Type	<i>int</i>
Initial Value	0
Description	Indicates the height of the media in terms of the base vertical resolution.

5.13 JxfsPtrReadFormCapability

This class specifies the read form capabilities of the printer.

5.13.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
readFormCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrReadFormCapability	readFormCapability	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
isBarcodeReadSupported	boolean	
isImageReadSupported	boolean	
isOCRReadSupported	boolean	
isTextReadSupported	boolean	
isMICRReadSupported	boolean	
isMSFReadSupported	boolean	
isPagemarkReadSupported	boolean	

Event	May occur after
none	

5.13.2 Properties

readFormCapability (R)

Type	<i>int</i>																
Initial Value	JXFS_PTR_READ_TEXT																
Description	Specifies whether the device can read data from the media. Depending on the device capability <i>readFormCapability</i> will be set as a combination of the following values:																
	<table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_PTR_READ_BARCODE</td> <td>Device has Barcode capability.</td> </tr> <tr> <td>JXFS_PTR_READ_IMAGE</td> <td>Device has imaging capability.</td> </tr> <tr> <td>JXFS_PTR_READ_MICR</td> <td>Device has MICR capability.</td> </tr> <tr> <td>JXFS_PTR_READ_MSF</td> <td>Device has MSF capability.</td> </tr> <tr> <td>JXFS_PTR_READ_OCR</td> <td>Device has OCR capability.</td> </tr> <tr> <td>JXFS_PTR_READ_TEXT</td> <td>Device has Text capability.</td> </tr> <tr> <td>JXFS_PTR_READ_PAGE MARK</td> <td>Device has pagemark capability.</td> </tr> </tbody> </table>	Value	Meaning	JXFS_PTR_READ_BARCODE	Device has Barcode capability.	JXFS_PTR_READ_IMAGE	Device has imaging capability.	JXFS_PTR_READ_MICR	Device has MICR capability.	JXFS_PTR_READ_MSF	Device has MSF capability.	JXFS_PTR_READ_OCR	Device has OCR capability.	JXFS_PTR_READ_TEXT	Device has Text capability.	JXFS_PTR_READ_PAGE MARK	Device has pagemark capability.
Value	Meaning																
JXFS_PTR_READ_BARCODE	Device has Barcode capability.																
JXFS_PTR_READ_IMAGE	Device has imaging capability.																
JXFS_PTR_READ_MICR	Device has MICR capability.																
JXFS_PTR_READ_MSF	Device has MSF capability.																
JXFS_PTR_READ_OCR	Device has OCR capability.																
JXFS_PTR_READ_TEXT	Device has Text capability.																
JXFS_PTR_READ_PAGE MARK	Device has pagemark capability.																

5.13.3 Methods

isBarcodeReadSupported

Syntax	<i>boolean isBarcodeReadSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has Barcode capability (the <i>readFormCapability</i> property contains the value JXFS_PTR_READ_BARCODE).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isMICRReadSupported

Syntax	<i>boolean isMICRReadSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has MICR capability (the <i>readFormCapability</i> property contains the value JXFS_PTR_READ_MICR).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isMSFReadSupported

Syntax	<i>boolean isMSFReadSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has MSF capability (the <i>readFormCapability</i> property contains the value JXFS_PTR_READ_MSF).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isOCRReadSupported

Syntax	<i>boolean isOCRReadSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has OCR capability (the <i>readFormCapability</i> property contains the value JXFS_PTR_READ_OCR).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isPagemarkReadSupported

Syntax	<i>boolean isPagemarkReadSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has pagemark capability (the <i>readFormCapability</i> property contains the value JXFS_PTR_READ_PAGEMARK).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isTextReadSupported

Syntax	<i>boolean isTextReadSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has text reading capability (the <i>readFormCapability</i> property contains the value JXFS_PTR_READ_TEXT).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isImageReadSupported

Syntax	<i>boolean isImageReadSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has imaging capability (the <i>readFormCapability</i> property contains the value JXFS_PTR_READ_IMGAGE).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

5.14 JxfsPtrReadImageCapability

This class specifies the read image capabilities of the printer.

5.14.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
readImageCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrReadImageCapability	readImageCapability	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
isImageTIFSupported	boolean	
isImageMTFSupported	boolean	
isImageBMPSupported	boolean	

Event	May occur after
none	

5.14.2 Properties

readImageCapability (R)

Type *int*

Initial Value 0

Description Specifies whether the device can read image data from the media. Depending on the device capability *readImageCapability* will be set as a combination of the following values:

Value	Meaning
JXFS_PTR_IMAGE_TIF	Device has capability to read tif format.
JXFS_PTR_IMAGE_MTF	Device has capability to read mtf format.
JXFS_PTR_IMAGE_BMP	Device has capability to read bmp format.

5.14.3 Methods

isImageTIFFSupported

Syntax	<i>boolean isImageTIFFSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the device has the capability to read tif format (the <i>readImageCapability</i> property contains the value JXFS_PTR_IMAGE_TIF).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isImageMTFSupported

Syntax	<i>boolean isImageMTFSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the device has the capability to read tif format (the <i>readImageCapability</i> property contains the value JXFS_PTR_IMAGE_MTF).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isImageBMPSupported

Syntax	<i>boolean isImageBMPSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the device has the capability to read tif format (the <i>readImageCapability</i> property contains the value JXFS_PTR_IMAGE_BMP).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

5.15 JxfsPtrRetractCount

This class specifies the number of media the printer has retracted.

5.15.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
retractCount	int	R/W	

Constructor	Parameter	Parameter-Type
JxfsPtrRetractCount	retractCount	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
<i>setProperty</i>	<i>Property</i>	
resetRetractCount	void	

Event	May occur after
none	

5.15.2 Properties

retractCount (R/W)

Type	<i>int</i>
Initial Value	0
Description	The number of media retracted; applicable only to printers with retract capability. This value is persistent: It is reset to zero by the <i>resetRetractCount</i> method. The retractCount can only be set by the Device Service internally.
Event	If the value of this property changes, the Device Service will send all registered StatusListeners a StatusEvent with a status value of: Value JXFS_S_PTR_RETRACT COUNT Meaning <i>retractCount</i> changed.

5.15.3 Methods

resetRetractCount

Syntax	<i>void resetRetractCount(void) throws JxfsException;</i>
Description	Sets the number of retracts to zero.
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

5.16 JxfsPtrWriteFormCapability

This class specifies the write form capabilities of the printer.

5.16.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
writeFormCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrWriteFormCapability	writeFormCapability	int

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	
isBarcodeWriteSupported	boolean	
isGraphicsWriteSupported	boolean	
isOCRWriteSupported	boolean	
isTextWriteSupported	boolean	
isMICRWriteSupported	boolean	
isMSFWriteSupported	boolean	

Event	May occur after
none	

5.16.2 Properties

writeFormCapability (R)

Type	<i>int</i>														
Initial Value	JXFS_PTR_WRITE_TEXT														
Description	Specifies whether the device can write data to the media. Depending on the device capability <i>writeFormCapability</i> will be set as a combination of the following values:														
	<table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_PTR_WRITE_BARCODE</td> <td>Device has Barcode capability.</td> </tr> <tr> <td>JXFS_PTR_WRITE_GRAPHICS</td> <td>Device has Graphics capability.</td> </tr> <tr> <td>JXFS_PTR_WRITE_MICR</td> <td>Device has MICR capability.</td> </tr> <tr> <td>JXFS_PTR_WRITE_MSF</td> <td>Device has MSF capability.</td> </tr> <tr> <td>JXFS_PTR_WRITE_OCR</td> <td>Device has OCR capability.</td> </tr> <tr> <td>JXFS_PTR_WRITE_TEXT</td> <td>Device has Text capability.</td> </tr> </tbody> </table>	Value	Meaning	JXFS_PTR_WRITE_BARCODE	Device has Barcode capability.	JXFS_PTR_WRITE_GRAPHICS	Device has Graphics capability.	JXFS_PTR_WRITE_MICR	Device has MICR capability.	JXFS_PTR_WRITE_MSF	Device has MSF capability.	JXFS_PTR_WRITE_OCR	Device has OCR capability.	JXFS_PTR_WRITE_TEXT	Device has Text capability.
Value	Meaning														
JXFS_PTR_WRITE_BARCODE	Device has Barcode capability.														
JXFS_PTR_WRITE_GRAPHICS	Device has Graphics capability.														
JXFS_PTR_WRITE_MICR	Device has MICR capability.														
JXFS_PTR_WRITE_MSF	Device has MSF capability.														
JXFS_PTR_WRITE_OCR	Device has OCR capability.														
JXFS_PTR_WRITE_TEXT	Device has Text capability.														

5.16.3 Methods

isBarcodeWriteSupported

Syntax	<i>boolean isBarcodeWriteSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to write barcode to the media (the <i>writeFormCapability</i> property contains the value JXFS_PTR_WRITE_BARCODE).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isGraphicsWriteSupported

Syntax	<i>boolean isGraphicsWriteSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to write graphics to the media (the <i>writeFormCapability</i> property contains the value JXFS_PTR_WRITE_GRAPHICS).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isOCRWriteSupported

Syntax	<i>boolean isOCRWriteSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to write OCR codes to the media (the <i>writeFormCapability</i> property contains the value JXFS_PTR_WRITE_OCR).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isTextWriteSupported

Syntax	<i>boolean isTextWriteSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to write text to the media (the <i>writeFormCapability</i> property contains the value JXFS_PTR_WRITE_TEXT).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isMICRWriteSupported

Syntax	<i>boolean isMICRWriteSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to write MICR to the media (the <i>writeFormCapability</i> property contains the value JXFS_PTR_WRITE_MICR).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isMSFWriteSupported

Syntax	<i>boolean isMSFWriteSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the printer has the capability to write text to the media (the <i>writeFormCapability</i> property contains the value JXFS_PTR_WRITE_MSF).
Parameter	None
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

6 Status Event Classes

If a device status changes one of the following classes is returned via a *StatusEvent*. This *xxxStatus*-Class is passed with the *details* property of the *StatusEvent*.
Each *xxxStatus*-Class provides several methods to query the changed device status.

6.1 JxfsThresholdStatus

This class specifies the threshold status of a container bin in the printer device. This can be the threshold of the paper supply, the toner supply or the retain bin.

6.1.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

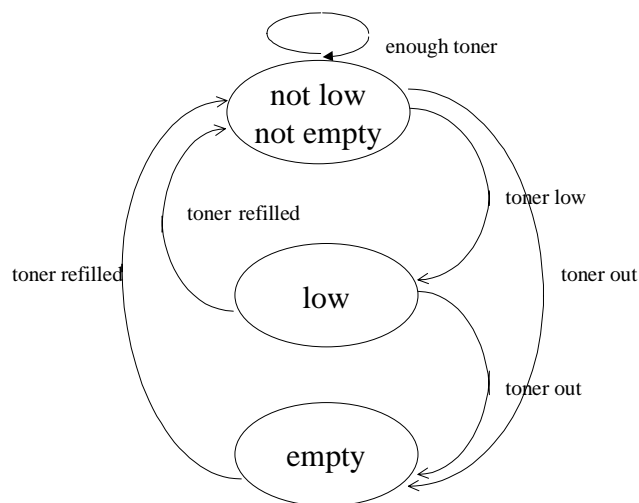
Property	Type	Access	Initialized after
thresholdState	int	R	

Constructor	Parameter	Parameter-Type
JxfsThresholdStatus	thresholdState	int

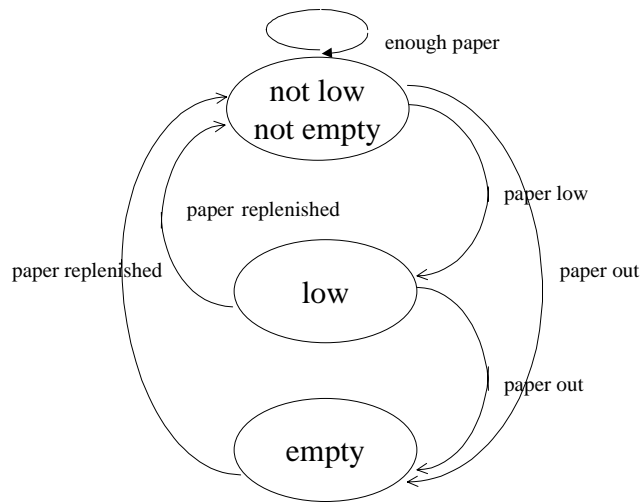
Method	Return	May be used after
isFull	<i>boolean</i>	
isHigh	<i>boolean</i>	
isLow	<i>boolean</i>	
isEmpty	<i>boolean</i>	
isUnknown	<i>boolean</i>	
toString	<i>String</i>	

For a description of the class and its properties and methods see "Base Architecture Guide".

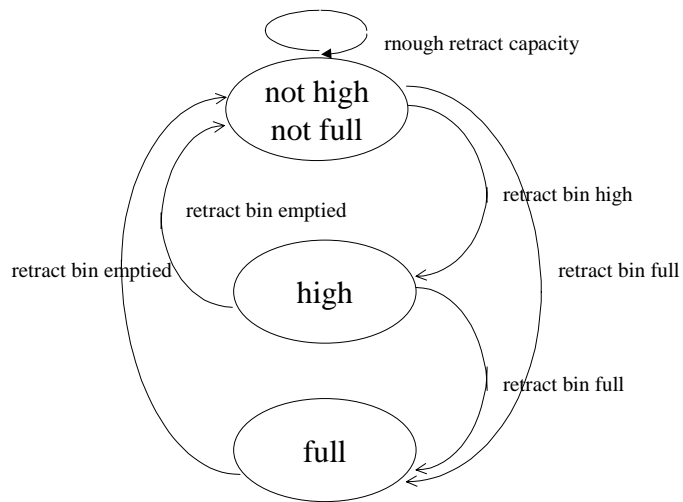
The threshold status of the toner supply may change according to the following state transition diagram :



The threshold status of the paper supply may change according to the following state transition diagram :



The threshold status of the retract bin may change according to the following state transition diagram:



6.2 JxfsPtrLampStatus

This class specifies the status of the scanner's imaging lamp.

6.2.1 Summary

Implements : *Serializable*

Extends : *JxfsType*

Property	Type	Access	Initialized after
lampStatus	int	R	

Constructor	Parameter	Parameter-Type
JxfsPtrLampStatus	lampStatus	int

Method	Return	May be used after
isLampFading	<i>boolean</i>	
isLampInoperable	<i>boolean</i>	
isLampNotSupported	<i>boolean</i>	
isLampOk	<i>boolean</i>	
isLampUnknown	<i>boolean</i>	

6.2.2 Properties

lampStatus (R)

Type	<i>int</i>
Initial Value	see Values below
Description	Specifies the status of the printer imaging lamp. Depending on device capability, <i>lampStatus</i> will be set to one of the following values:
Value	Meaning
JXFS_S_PTR_LAMP_OK	Imaging lamp is ok.
JXFS_S_PTR_LAMP_FADING	Imaging lamp should be changed.
JXFS_S_PTR_LAMP_INOP	Imaging lamp is inoperable.
JXFS_S_PTR_LAMP_NOT SUPP	Capability to report the state of the imaging lamp is not supported by the printer.
JXFS_S_PTR_LAMP_UNKNOWN	State of the imaging lamp cannot be determined with the printer in its current state.
Event	If the value of this property changes, the Device Service will send all registered StatusListeners a StatusEvent with a status value of:
Value	Meaning
JXFS_S_PTR_LAMP	<i>lampStatus</i> changed.

6.2.3 Methods

isLampFading

Syntax	<i>boolean isLampFading(void) throws JxfsException;</i>
Description	Returns TRUE if the imaging lamp should be changed (the value of the <i>lampStatus</i> property is JXFS_S_PTR_LAMP_FADING).
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isLampInoperable

Syntax	<i>boolean isLampInoperable(void) throws JxfsException;</i>
Description	Returns TRUE if the imaging lamp is inoperable (the value of the <i>lampStatus</i> property is JXFS_S_PTR_LAMP_INOP).
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isLampNotSupported

Syntax	<i>boolean isLampNotSupported(void) throws JxfsException;</i>
Description	Returns TRUE if the capability to report the state of the imaging lamp is not supported by the printer (the value of the <i>lampStatus</i> property is JXFS_S_PTR_LAMP_NOTSUPP).
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isLampOk

Syntax	<i>boolean isLampOk(void) throws JxfsException;</i>
Description	Returns TRUE if the imaging lamp is ok (the value of the <i>lampStatus</i> property is JXFS_S_PTR_LAMP_OK).
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isLampUnknown

Syntax	<i>boolean isLampUnknown(void) throws JxfsException;</i>
Description	Returns TRUE if the status of the imaging lamp cannot be determined with the printer in its current state (the value of the <i>lampStatus</i> property is JXFS_S_PTR_LAMP_UNKNOWN).
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

6.3 JxfsMediaStatus

This class specifies the status of the printer media.

Query	Return
isEjected	<i>boolean</i>
isJammed	<i>boolean</i>
isPresent	<i>boolean</i>
isNotSupported	<i>boolean</i>
isUnknown	<i>boolean</i>

For a description of the class and its properties and methods see "Base Architecture Guide".

6.4 JxfsPtrStatus

This class contains properties and methods to query the status of the printer device and its resources.

6.4.1 Summary

Implements : *Serializable*

Extends : *JxfsStatus*

Property	Type	Access	Initialized after
mediaStatus	JxfsMediaStatus	R	
paperStatus	JxfsThresholdStatus	R	
tonerStatus	JxfsThresholdStatus	R	

Constructor	Parameter	Parameter-Type
JxfsPtrStatus	mediaStatus	JxfsMediaStatus
	paperStatus	JxfsThresholdStatus
	tonerStatus	JxfsThresholdStatus

Method	Return	May be used after
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

6.4.2 Properties

mediaStatus (R)

Type	<i>JxfsMediaStatus</i>
Description	Specifies the state of the print media (i.e., the paper, passbook, single sheet, etc.).
Event	If the value of this property changes, the Device Service will send all registered StatusListeners a StatusEvent with a status value of:
Value	Meaning
JXFS_S_PTR_MEDIA	<i>mediaStatus</i> changed.

paperStatus (R)

Type	<i>JxfsThresholdStatus</i>
Description	Specifies the state of the paper supply.
Event	If the value of this property changes, the Device Service will send all registered StatusListeners a StatusEvent with a status value of:
Value	Meaning
JXFS_S_PTR_PAPER	<i>paperStatus</i> changed.

tonerStatus (R)

Type	<i>JxfsThresholdStatus</i>
Description	Specifies the status of the toner supply.
Event	If the value of this property changes, the Device Service will send all registered StatusListeners a StatusEvent with a status value of:
Value	Meaning
JXFS_S_PTR_TONER	<i>tonerStatus</i> changed.

6.4.3 Events

StatusEvent

Interface

jxfs.events.StatusListener

Method

statusOccurred(StatusEvent *e*);

Remarks

Issued to present status data from the printer to the application

Properties

Type

int

Name

status

Meaning

The *status* value can be one of the following Codes:

Codes

Value

JXFS_S_DEVICE

JXFS_S_PTR_MEDIA

JXFS_S_PTR_PAPER

JXFS_S_PTR_TONER

Meaning

Property *deviceStatus* changed.

Property *mediaStatus* changed.

Property *paperStatus* changed.

Property *tonerStatus* changed.

7 Codes

7.1 Error Codes

Value	Meaning
JXFS_E_PTR_EXTENT_NOT_SUPPORTED	Printer cannot report extent(s).
JXFS_E_PTR_FIELD_ERROR	An error occurred while processing a field, causing termination of the print request. Details can be found in the <i>extendedErrorCode</i> .
JXFS_E_PTR_FIELD_INVALID	Specified field is invalid.
JXFS_E_PTR_FIELD_NOT_FOUND	Specified field does not exist.
JXFS_E_PTR_FIELD_SPEC_FAILURE	Syntax of the <i>fieldWriteData</i> is invalid.
JXFS_E_PTR_FLUSH_FAIL	Printer was not able to flush data.
JXFS_E_PTR_FORM_INVALID	Specified form definition is invalid
JXFS_E_PTR_NOFORMS	There are no form descriptions available on the printer.
JXFS_E_PTR_NOMEDIAS	There are no media descriptions available on the printer.
JXFS_E_PTR_FORM_NOT_FOUND	Specified form definition cannot be found.
JXFS_E_PTR_MEDIA_INVALID	Specified media definition is invalid.
JXFS_E_PTR_MEDIA_NOT_FOUND	Specified media definition cannot be found.
JXFS_E_PTR_MEDIA_OVERFLOW	Form overflowed the media.
JXFS_E_PTR_MEDIA_SKEWED	Media skew exceeded the limit in the form definition.
JXFS_E_PTR_MEDIA_TURN_FAIL	Printer was not able to turn the inserted media
JXFS_E_PTR_NO_MEDIA_PRESENT	Media is not present in the printer (after the specified waiting time).
JXFS_E_PTR_RETRACT_BIN_FULL	Retract bin is full. No more media can be retracted. Current media is still in the printer.
<i>extendedErrorCodes :</i>	
JXFS_E_PTR_FIELD_GRAPHIC	Specified graphic image could not be printed.
JXFS_E_PTR_FIELD_HW_ERROR	Specified field uses special hardware and an error occurred.
JXFS_E_PTR_FIELD_NOT_READ	Attempt was made to read an output field.
JXFS_E_PTR_FIELD_NOT_WRITE	Attempt was made to write to an input field.
JXFS_E_PTR_FIELD_OVERFLOW	Value specified for the field is too long
JXFS_E_PTR_FIELD_REQUIRED	Specified field <i>must</i> be supplied by the application.
JXFS_E_PTR_FIELD_STATIC_OVWR	Specified field is <i>static</i> and thus cannot be overwritten by the application.
JXFS_E_PTR_FIELD_TYPE_NOT_SUPPORTED	Form field type is not supported by the printer.

7.2 Exception Codes

Value	Meaning
JXFS_E_CLOSED	Device has not been opened yet.
JXFS_E_PARAMETER_INVALID	An invalid parameter was given to the operation.
JXFS_E_NOT_SUPPORTED	Operation is not supported by device.
JXFS_E_REMOTE	Communication error during remote call.

7.3 Status Codes

General Status Codes

General Status Codes that specify a value change.

Value	Meaning
JXFS_S_DEVICE	The status of the printer device has changed.
JXFS_S_PTR_LAMP	The status of the scanner's imaging lamp has changed.
JXFS_S_PTR_MEDIA	The status of the media has changed.
JXFS_S_PTR_PAPER	The status of the paper has changed.
JXFS_S_PTR_RETRACT_BIN	The status of the retract bin has changed.
JXFS_S_PTR_RETRACTCOUNT	The retract count has changed.
JXFS_S_PTR_TONER	The status of the toner has changed.

Lamp Status Codes

Defines the status the scanner's imaging lamp can report.

Value	Meaning
JXFS_S_PTR_LAMP_FADING	Imaging lamp should be changed.
JXFS_S_PTR_LAMP_INOP	Imaging lamp is inoperable.
JXFS_S_PTR_LAMP_OK	Imaging lamp is ok.
JXFS_S_PTR_LAMP_NOTSUPP	Capability to report the state of the imaging lamp is not supported by the printer.
JXFS_S_PTR_LAMP_UNKNOWN	State of the imaging lamp cannot be determined with the printer in its current state.

Bin Status Codes

Defines the status code the paper supply, the toner supply or the retain bin can report.

Value	Meaning
JXFS_S_BIN_EMPTY	Bin is empty.
JXFS_S_BIN_FULL	Bin is full.
JXFS_S_BIN_HIGH	Bin is high.
JXFS_S_BIN_LOW	Bin is low.
JXFS_S_BIN_NOTSUPPORTED	Capability to report the state of the bin is not supported by the device.
JXFS_S_BIN_OK	Bin is available and neither high nor full.
JXFS_S_BIN_UNKNOWN	State of the bin cannot be determined with the device in its current state.

Media Status Codes

Defines the status codes that can be reported for the media.

Value	Meaning
JXFS_S_MEDIA_EJECTED	Media is in the entry/exit slot of the device.
JXFS_S_MEDIA_JAMMED	Media is jammed in the device.
JXFS_S_MEDIA_NOTSUPPORTED	Capability to report the state of the media is not supported by the device.
JXFS_S_MEDIA_UNKNOWN	State of the media cannot be determined with the device in its current state.

7.4 Constants

Alignment codes

The alignment codes are returned by the `getAlignment` method of the `JxfsPtrFormsConfig` class or are an input parameter for the `setAlignment` method and the constructor of this class. The class is used to configure the usage of forms.

Value	Meaning
JXFS_PTR_ALN_BOTTOMLEFT	Align the form to the bottom left of the media.
JXFS_PTR_ALN_BOTTOMRIGHT	Align the form to the bottom right of the media.
JXFS_PTR_ALN_TOPLEFT	Align the form to the top left of the media.
JXFS_PTR_ALN_TOPRIGHT	Align the form to the top right of the media.

Base unit codes

The base unit codes are returned by the getBase method of the JxfsPtrFormsConfig class or are an input parameter for the setBase method and the constructor of this class. The class is used to configure the usage of forms.

Value	Meaning
JXFS_PTR_FRM_INCH	Base unit is inches.
JXFS_PTR_FRM_MM_BOTTOMRIGHT	Base unit is millimeters.
JXFS_PTR_FRM_ROW	Base unit is rows and columns.

Control Media Codes

The control media codes are returned by the ctrlMediaCapability method or are an input parameter for the ctrlMedia method. The codes can be or'ed.

Value	Meaning
JXFS_PTR_CTRL_ALARM	Device can/shall ring a bell, beep or otherwise sound an audible alarm.
JXFS_PTR_CTRL_STAMP	Device can/shall stamp the media.
JXFS_PTR_CTRL_CUT	Device can/shall cut the media.
JXFS_PTR_CTRL_EJECT	Device can/shall eject the media.
JXFS_PTR_CTRL_FLUSH	Device can be sent data that is buffered internally, and flushed to the printer on request.
JXFS_PTR_CTRL_PARTIALCUT	Device can/shall partially cut the media
JXFS_PTR_CTRL_PERFORATE	Device can/shall perforate the media.
JXFS_PTR_CTRL_RETRACT	Device can/shall retract the media.
JXFS_PTR_CTRL_SKIP	Device can/shall skip to mark.
JXFS_PTR_CTRL_STACK	Device can/shall stack media items before ejecting as a bundle.

Control Turn Media Codes

The control turn media codes are returned by the ctrlTurnCapability method.

JXFS_PTR_CTRL_ATPBACKWARD	Device can/shall turn one page backward.
JXFS_PTR_CTRL_ATP_FORWARD	Device can/shall turn one page forward.
JXFS_PTR_CTRL_TURNMEDIA	Device can/shall turn the media.

Extent Capability Codes

Following extent capability codes can be or'ed.

Value	Meaning
JXFS_PTR_EXT_HORIZONTAL	Device has horizontal size detection capability.
JXFS_PTR_EXT_VERTICAL	Device has vertical size detection capability.

Read Capability Codes

Following read form capability codes can be or'ed.

Value	Meaning
JXFS_PTR_READ_BARCODE	Device has Barcode capability.
JXFS_PTR_READ_IMAGE	Device has imaging capability.
JXFS_PTR_READ_MICR	Device has MICR capability.
JXFS_PTR_READ_MSF	Device has MSF capability.
JXFS_PTR_READ_OCR	Device has OCR capability.
JXFS_PTR_READ_TEXT	Device has Text capability.

JXFS_PTR_READ_PAGEMARK	Device has pagemark capability.
------------------------	---------------------------------

Read Image Capability Codes

The read image capability codes are returned by the readImageCapability method of the class JxfsPtrReadImageCapability. The values can be or'ed.

Value	Meaning
JXFS_PTR_IMAGE_TIF	Device has capability to read tif.
JXFS_PTR_IMAGE_MTF	Device has capability to read mtf format.
JXFS_PTR_IMAGE_BMP	Device has capability to read bmp format.

Write Capability Codes

Following write capability codes can be or'ed.

Value	Meaning
JXFS_PTR_WRITE_BARCODE CODE	Device has Barcode capability.
JXFS_PTR_WRITE_GRAPHICS	Device has Graphics capability.
JXFS_PTR_WRITE_MICR	Device has MICR capability.
JXFS_PTR_WRITE_MSF	Device has MSF capability.
JXFS_PTR_WRITE_OCR	Device has OCR capability.
JXFS_PTR_WRITE_TEXT	Device has Text capability.

Description

Set to the image data format and can be one of the following values:

JXFS_PTR_IMAGE_TIF

Image data is in TIF
format.

JXFS_PTR_IMAGE_MTF

Image data is in MTF
format.

JXFS_PTR_IMAGE_BMP

Image data is in BMP
format.

7.5 Operation ID Codes

Following codes specify the operation which generated the OperationCompleteEvent.

Value	Method
JXFS_O_PTR_ATP_BACKWARD	<i>atpBackward</i>
JXFS_O_PTR_ATP_FORWARD	<i>atpForward</i>
JXFS_O_PTR_CTRL_MEDIA	<i>ctrlMedia</i>
JXFS_O_PTR_EJECT_MEDIA	<i>ejectMedia</i>
JXFS_O_PTR_MEDIA_EXTENTS	<i>mediaExtents</i>
JXFS_O_PTR_PREPARE_EJECT	<i>prepareEject</i>
JXFS_O_PTR_RESET_PRINTER	<i>resetPrinter</i>
JXFS_O_PTR_READ_FORM_DATA	<i>readForm</i>
JXFS_O_PTR_READ_IMAGE	<i>readImage</i>
JXFS_O_PTR_RETRACT_MEDIA	<i>retractMedia</i>
JXFS_O_PTR_TURN_MEDIA	<i>turnMedia</i>
JXFS_O_PTR_WRITE_FORM_DATA	<i>printForm</i>
JXFS_O_PTR_WRITE_RAW_DATA	<i>printRawData</i>

8 Device Service Interface Methods

The Device Service interface is common to all device services of this device type. It is used by the Device Controls to access the functionality of the device. This interface has to be implemented by any J/XFS Device Service.

The device type specific Device Service interface is similar to the Device Control interface. All device specific method calls are extended by an additional parameter (int control_id). This is always added as the last parameter in every operation.

8.1 Form, Field and Media Definitions

For the definition of forms, the fields within them, and the media on which they are printed see the WOSA/XFS Specification.

9 Index

access.....	42
alignment.....	46, 49
atpBackward.....	30
atpForward.....	30
base.....	46, 49, 55
classType.....	42
compound.....	13
ctrlMedia.....	14
ctrlMediaCapability.....	36
ctrlTurnMediaCapability.....	39
ejectMedia.....	24
extentCapability.....	41
fieldFailure.....	44
fieldName.....	43, 44
fields.....	46
foldType.....	55
format.....	43
formName.....	44, 46
formsDescriptionList.....	49
getFieldDescription.....	14
getFormDescription.....	15
getFormList.....	15
getMediaDescription.....	15
getMediaList.....	16
getProperty.....	11
height.....	46
IJxfsEject.....	23
IJxfsMediaTurn.....	29
IJxfsPrinterControl.....	12
IJxfsRead.....	31
IJxfsRetract.....	26
imageData.....	51
imageType.....	51
indexCount.....	43
initialValue.....	43
isBarcodeReadSupported.....	59
isBarcodeWriteSupported.....	64
isCtrlAlarmSupported.....	37
isCtrlATPBackwardSupported.....	39
isCtrlATPForwardSupported.....	39
isCtrlCutSupported.....	37
isCtrlEjectSupported.....	37
isCtrlFlushSupported.....	38
isCtrlPartialCutSupported.....	38
isCtrlPerforateSupported.....	38
isCtrlRetractSupported.....	38
isCtrlSkipSupported.....	38
isCtrlStackSupported.....	38
isCtrlStampSupported.....	37
isCtrlTurnMediaSupported.....	40
isEjected.....	70
isExtHorizontalSupported.....	41
isExtVerticalSupported.....	41
isGraphicsWriteSupported.....	65
isImageBMPSupported.....	62
isImageMTFSupported.....	62
isImageReadSupported.....	60
isImageTIFSupported.....	62
isJammed.....	70

isLampFading.....	69
isLampInoperable.....	69
isLampNotSupported.....	69
isLampOk.....	69
isLampUnknown.....	69
isMICRReadSupported.....	60
isMICRWriteSupported.....	65
isMSFReadSupported.....	60
isMSFWriteSupported.....	65
isNotSupported.....	70
isOCRReadSupported.....	60
isOCRWriteSupported.....	65
isPagemarkReadSupported.....	60
isPresent.....	70
isTextReadSupported.....	60
isTextWriteSupported.....	65
isUnknown.....	70
JXFS_E_NOT_SUPPORTED.....	73
JXFS_E_CLOSED.....	73
JXFS_E_PARAMETER_INVALID.....	73
JXFS_E_PTR_ALARM_NOT_SUPPORTED.....	73
JXFS_E_PTR_FIELD_ERROR.....	73
JXFS_E_PTR_FIELD_GRAPHIC.....	73
JXFS_E_PTR_FIELD_HW_ERROR.....	73
JXFS_E_PTR_FIELD_INVALID.....	73
JXFS_E_PTR_FIELD_NOT_FOUND.....	73
JXFS_E_PTR_FIELD_NOT_READ.....	73
JXFS_E_PTR_FIELD_NOT_WRITE.....	73
JXFS_E_PTR_FIELD_OVERFLOW.....	73
JXFS_E_PTR_FIELD_REQUIRED.....	73
JXFS_E_PTR_FIELD_SPEC_FAILURE.....	73
JXFS_E_PTR_FIELD_STATIC_OVWR.....	73
JXFS_E_PTR_FIELD_TYPE_NOT_SUPPORTED.....	73
JXFS_E_PTR_FLUSH_FAIL.....	73
JXFS_E_PTR_FORM_INVALID.....	73
JXFS_E_PTR_FORM_NOT_FOUND.....	73
JXFS_E_PTR_MEDIA_INVALID.....	73
JXFS_E_PTR_MEDIA_NOT_FOUND.....	73
JXFS_E_PTR_MEDIA_OVERFLOW.....	73
JXFS_E_PTR_MEDIA_SKEWED.....	73
JXFS_E_PTR_MEDIA_TURN_FAIL.....	73
JXFS_E_PTR_NO_MEDIA_PRESENT.....	73
JXFS_E_PTR_NOFORMS.....	73
JXFS_E_PTR_NOMEDIAS.....	73
JXFS_E_PTR_RETRACT_BIN_FULL.....	73
JXFS_E_REMOTE.....	73
JXFS_FRM_MM_BOTTOMRIGHT.....	76
JXFS_O_PTR_TURN_MEDIA.....	30
JXFS_O_PTR_ATP_BACKWARD.....	30
JXFS_O_PTR_ATP_FORWARD.....	30
JXFS_O_PTR_CTRL_MEDIA.....	78
JXFS_O_PTR_CTRL_MEDIA.....	14
JXFS_O_PTR_EJECT_MEDIA.....	24
JXFS_O_PTR_MEDIA_EXTENTS.....	16
JXFS_O_PTR_PREPARE_EJECT.....	25
JXFS_O_PTR_READ_FORM_DATA.....	32, 33
JXFS_O_PTR_READ_IMAGE.....	33
JXFS_O_PTR_RESET_PRINTER.....	78
JXFS_O_PTR_RETRACT_MEDIA.....	27
JXFS_O_PTR_WRITE_FORM_DATA.....	78
JXFS_O_PTR_WRITE_FORM_DATA.....	17
JXFS_O_PTR_WRITE_RAW_DATA.....	78

JXFS_O_PTR_WRITE_RAW_DATA.....	19
JXFS_PTR_FRM_INCH.....	76
JXFS_PTR_ALN_BOTTOMLEFT.....	75
JXFS_PTR_ALN_BOTTOMRIGHT.....	75
JXFS_PTR_ALN_TOPLEFT.....	75
JXFS_PTR_ALN_TOPRIGHT.....	75
JXFS_PTR_CTRL_ALARM.....	76
JXFS_PTR_CTRL_ATP_BACKWARD.....	76
JXFS_PTR_CTRL_ATP_FORWARD.....	76
JXFS_PTR_CTRL_CUT.....	76
JXFS_PTR_CTRL_EJECT.....	76
JXFS_PTR_CTRL_FLUSH.....	76
JXFS_PTR_CTRL_PARTIALCUT.....	76
JXFS_PTR_CTRL_PERFORATE.....	76
JXFS_PTR_CTRL_RETRACT.....	76
JXFS_PTR_CTRL_SKIP.....	76
JXFS_PTR_CTRL_STACK.....	76
JXFS_PTR_CTRL_STAMP.....	76
JXFS_PTR_CTRL_TURNMEDIA.....	76
JXFS_PTR_EXT_HORIZONTAL.....	76
JXFS_PTR_EXT_VERTICAL.....	76
JXFS_PTR_FRM_ROW.....	76
JXFS_PTR_IMAGE_BMP.....	77
JXFS_PTR_IMAGE_MTF.....	77
JXFS_PTR_IMAGE_TIF.....	77
JXFS_PTR_READ_BARCODE.....	76
JXFS_PTR_READ_IMAGE.....	76
JXFS_PTR_READ_MICR.....	76
JXFS_PTR_READ_MSF.....	76
JXFS_PTR_READ_OCR.....	76
JXFS_PTR_READ_PAGEMARK.....	77
JXFS_PTR_READ_TEXT.....	77
JXFS_PTR_WRITE_BARCODE.....	77
JXFS_PTR_WRITE_GRAPHICS.....	77
JXFS_PTR_WRITE_MICR.....	77
JXFS_PTR_WRITE_MSF.....	77
JXFS_PTR_WRITE_OCR.....	77
JXFS_PTR_WRITE_TEXT.....	77
JXFS_S_BIN_EMPTY.....	74
JXFS_S_BIN_FULL.....	74
JXFS_S_BIN_HIGH.....	74
JXFS_S_BIN_LOW.....	74
JXFS_S_BIN_NOTSUPPORTED.....	74
JXFS_S_BIN_OK.....	74
JXFS_S_BIN_UNKNOWN.....	74
JXFS_S_DEVICE.....	74
JXFS_S_MEDIA_EJECTED.....	74
JXFS_S_MEDIA_JAMMED.....	74
JXFS_S_MEDIA_NOTSUPPORTED.....	74
JXFS_S_MEDIA_UNKNOWN.....	74
JXFS_S_PTR_INK.....	25, 28
JXFS_S_PTR_LAMP.....	74
JXFS_S_PTR_LAMP.....	34
JXFS_S_PTR_LAMP_FADING.....	74
JXFS_S_PTR_LAMP_INOP.....	74
JXFS_S_PTR_LAMP_NOTSUPP.....	74
JXFS_S_PTR_LAMP_OK.....	74
JXFS_S_PTR_LAMP_UNKNOWN.....	74
JXFS_S_PTR_MEDIA.....	74
JXFS_S_PTR_MEDIA.....	14, 17, 18, 20, 24, 25, 27, 33, 34
JXFS_S_PTR_PAPER.....	74
JXFS_S_PTR_PAPER.....	18, 19

JXFS_S_PTR_RETRACT_BIN	74
JXFS_S_PTR_RETRACTCOUNT	74
JXFS_S_PTR_TONER	74
JXFS_S_PTR_TONER	18, 19
JxfsPtrCtrlMediaCapability	36
JxfsPtrCtrlTurnCapability	39
JxfsPtrExtentCapability	41
JxfsPtrField	42
JxfsPtrForm	45
JxfsPtrFormsConfig	48
JxfsPtrImage	51
JxfsPtrLampStatus	68
JxfsPtrMaxRetractCapability	52
JxfsPtrMaxStackerCapability	53
JxfsPtrMediaExtents	58
JxfsPtrReadFormCapability	59
JxfsPtrReadImageCapability	61
JxfsPtrRetractCount	63
JxfsPtrStatus	71
JxfsPtrWriteFormCapability	64
JxfsThresholdStatus	66
lampStatus	68
lineCount	55
maxRetractCapability	52
maxStackerCapability	53
mediaDescriptionList	49
mediaExtents	16
mediaStatus	71
mediaType	55
OCPtrFieldInfoEvent	21
OCPtrFormInfoEvent	21
OCPtrFormListEvent	21
OCPtrMediaExtentEvent	21
OCPtrMediaInfoEvent	22
OCPtrMediaListEvent	22
OCPtrRawDataEvent	22
OCPtrReadDataEvent	34
OCPtrReadImageEvent	34
offsetX	46, 50
offsetY	46, 50
orientation	47
overflow	43
pageCount	55
paperStatus	71
prepareEject	25
printAreaHeight	55
printAreaWidth	56
printAreaX	56
printAreaY	56
printForm	17
printRawData	18
readForm	32
readFormCapability	59
readImage	33
readImageCapability	61
resetPrinter	20
resetRetractCount	63
restrictedAreaHeight	56
restrictedAreaWidth	56
restrictedAreaX	56
restrictedAreaY	56
retractCount	63

retractMedia	27
setProperty	11
sizeHeight	56
sizeWidth	56
sizeX	58
sizeY	58
stagger	57
status	13
Status Event Classes	66
StatusEvent	72
tonerStatus	71
turnMedia	30
type	43
unitX	47, 50, 57
unitY	47, 50, 57
userPrompt	47
versionMajor	47
versionMinor	47
width	47
writeFormCapability	64