CEN

WORKSHOP

AGREEMENT

CWA 16008-8

August 2009

ICS 35.240.40

English version

J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Part 8: Sensors and Indicators Unit Device Class Interface - Programmer's Reference

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

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

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

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

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

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

Management Centre: Avenue Marnix 17, B-1000 Brussels

Ref. No.:CWA 16008-8:2009 E

Contents

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 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 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 Secretariat , and at http://www.cen.eu/cenorm/sectors/sectors/isss/activity/jxfs_membership.asp. The specification was agreed upon by the J/XFS Workshop Meeting of 2009-05-6/9 in Brussels, and the final version was sent to CEN for publication on 2009-06-12.

The specification is continuously reviewed and commented in the CEN J/XFS Workshop. The information published in this CWA is furnished for informational purposes only. CEN makes no warranty expressed or implied, with respect to this document. Updates of the specification will be available from the CEN J/XFS Workshop public web pages pending their integration in a new version of the CWA (see http://www.cen.eu/cenorm/sectors/sectors/isss/activity/jxfs_cwas.asp).

The J/XFS specifications are now further developed in the CEN J/XFS Workshop. CEN Workshops are open to all interested parties offering to contribute. Parties interested in participating and parties wanting to submit questions and comments for the J/XFS specifications, please contact the J/XFS Workshop Secretariat hosted in CEN (jxfs-helpdesk@cen.eu).

Questions and comments can also be submitted to the members of the J/XFS Forum through the J/XFS Forum web-site http://www.jxfs.net.

This CWA is composed of the following parts:

- Part 1: J/eXtensions for Financial Services (J/XFS) for the Java Platform – Release 2009 - Base Architecture - Programmer's Reference
- Part 2: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Pin Keypad Device Class Interface - Programmer's Reference
- Part 3: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Magnetic Stripe & Chip Card Device Class Interface - Programmer's Reference
- Part 4: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Text Input/Output Device Class Interface - Programmer's Reference
- Part 5: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Cash Dispenser, Recycler and ATM Device Class Interface - Programmer's Reference
- Part 6: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Printer Device Class Interface - Programmer's Reference
- Part 7: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Alarm Device Class Interface - Programmer's Reference
- Part 8: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Sensors and Indicators Unit Device Class Interface - Programmer's Reference
- Part 9: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Depository Device Class Interface - Programmer's Reference
- Part 10: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Check Reader/Scanner Device Class Interface - Programmer's Reference (deprecated in favour of Part 13)
- Part 11: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Camera Device Class Interface - Programmer's Reference
- Part 12: J/eXtensions for Financial Services (J/XFS) for the Java Platform - Release 2009 - Vendor Dependant Mode Specification - Programmer's Reference
- Part 13: J/eXtensions for Financial Services (J/XFS) for the Java Platform – Scanner Device Class Interface - Programmer's Reference (recommended replacement for Part 10)

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://www.sun.com. All other trademarks are trademarks of their respective owners.

This CEN Workshop Agreement is publicly available as a reference document from the National Members of CEN : AENOR, AFNOR, ASRO, BDS, BSI, CSNI, CYS, DIN, DS, ELOT, EVS, IBN, IPQ, IST, LVS, LST, MSA, MSZT, NEN, NSAI, ON, PKN, SEE, SIS, SIST, SFS, SN, SNV, SUTN and UNI.

Comments or suggestions from the users of the CEN Workshop Agreement are welcome and should be addressed to the CEN Management Centre.

1 History

Main differences to CWA 14923-8:2004 are:

- Updated Foreword
- Replaced OperationCompletEvent by JxfsOperationCompletEvent and also for StatusEvent
- References were added
- Include SIU Open Method Proposal 1.0, 2006-03-13 (276)
- Include Clarification to handle a non supported state of a light 1.0, 2007-05-29 (356)
- In chapter 7.1 the argument of setProperty was falsely void.
- Changed the definition of JXFS_SIU_NOT_AVAILABLE at JxfsSIUGuidLightCapability.

Main differences to CWA 13937-8:2000 are:

- New guidance lights for scanner, coin acceptor and document printer."
- Added missing square bracket "sensorCapabilities[JXFS_SIU_ENHANCEDAUDIO]".
- Included corrections according to NCR comments of 2002-06-04.
- Corrected wrong mark for JXFS_SIU_ENHANCEDAUDIOCONTROL in chapter "9.3 Index Codes"
- Three new guidance lights for scanner, coin acceptor and document printer.
- Comment about additional indicators at front office devices in overview section.
- Rephrased a sentence in chapter 8.20.2.
- New capability to specify the ways how to control the Audio Jack.
- Additional note that switching the Audio Jack Mode will be supported only by devices that support this feature in the capabilities.
- Included definition how to handle null references.
- Changed Audio Jack handling behaviour to be compatible with XFS.

2 Scope

This document describes the Sensors and Indicators Unit Device Class (SIU) 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 Sensors and Indicators Units, the basic Device Control structure is extended with various properties and methods specific to this device which are described on the following pages.

3 Overview

The J/XFS Sensors and Indicators Unit Device Support allows for the operation of the following functionalities of a generic Sensors and Indicators Unit (SIU):

- **Door sensors, such as cabinet, safe or vandal shield doors;**
- **Alarm sensors, such as tamper, seismic or heat sensors;**
- **Generic sensors, such as proximity or ambient light sensors;**
- **Key switch sensors, such as the ATM operator switch;**
- **Lamp/sign indicators, such as fascia light or audio indicators;**
- **Auxiliary indicators;**
- **Audio Jack device, for use by the partially deaf;**
- **Guidance lights.**

Additional indicators on typical front office devices like user indicators in printers or error indicators at card readers are not covered by this specification.

The J/XFS Sensors and Indicators Unit Device Support uses the event driven model. The application obtains a J/XFS SIU Device Control Object from the Device Manager and then calls the defined I/O methods with passing data objects containing the parameters. When an I/O method is called, the J/XFS SIU 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 SIU Device Control Object for the various types of events it wishes to handle.

4 Device behavior

4.1 Audio Jack Behavior

The Audio Jack device is provided to support the requirements of the American Disabilities Act. This device allows audio feedback publicly and / or via the consumers' personal headset (vendor hardware permitting). For privacy, the device allows input to only be directed to the consumers' headset. In 'auto' & 'semi.auto' mode (and where the vendor's hardware allows), public transmission of audio can be automatically inhibited when the consumer's headset is plugged into the audio jack. In 'auto' mode (and where the vendor's hardware allows), public transmission of audio can be automatically re-activated when the consumer's headset is unplugged from the Audio Jack.

The audio jack provides the application with the following information

- If the headset is present
- Whether the audio output is to the speakers or headset
- Privacy\Public mode: ie. Whether insertion of a headset automatically switches off public audio.

The device is managed by a sensor JXFS_SIU_ENHANCEDAUDIO and an auxiliary JXFS_SIU_ENHANCEDAUDIOCONTROL.

The JXFS_SIU_ENHANCEDAUDIO sensor is used to

- Provide information on the presence of the Audio Jack device
- To report whether a headset is currently attached
- Report state change events when a headset is inserted or removed.

The JXFS_SIU_ENHANCEDAUDIOCONTROL auxiliary is used to control the behaviour of the Audio Jack. It allows the application to,

- Set the mode of the Audio Jack – auto-mode, semi-auto mode or manual mode.
- Set the state of the Audio Jack – public or private.

There are no events associated with this auxiliary.

A full description of auto, semi-auto & manual mode, as well as public & private states is contained in the following pages.

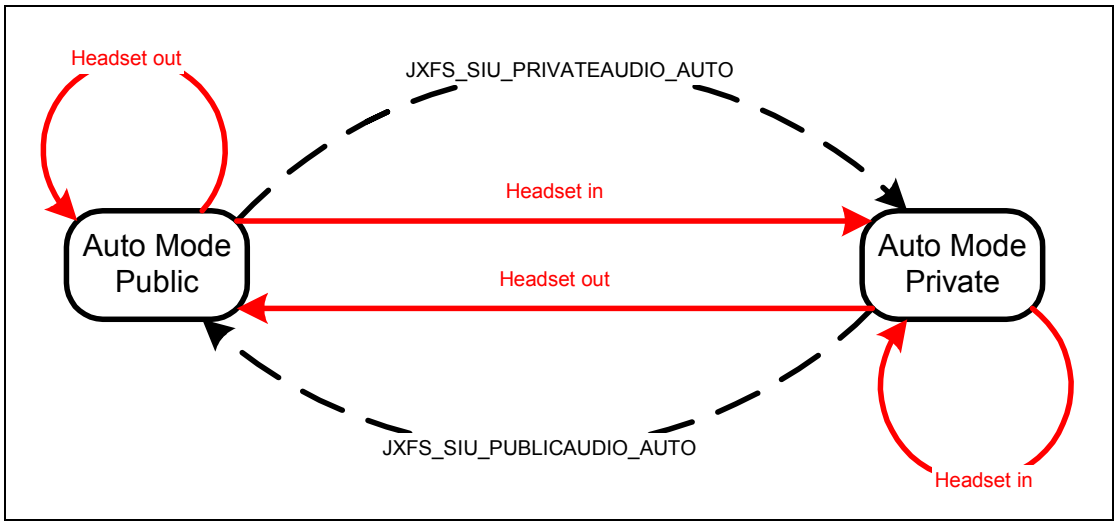
The following describes the device behaviour during auto, semi-auto and manual mode.

Auto Mode

In auto mode, when a consumer headset is plugged into the jack, the audio is automatically directed to the headset and the audio is no longer sent to the speakers. When the headset is removed the audio is redirected to the speakers. The following state diagram completely describes the behaviour of the device in auto mode.

State Description

Auto Mode Public	audio output is played through the public speakers only
Auto Mode Private	audio is played through the consumer headset only



Auto-mode State diagram 1

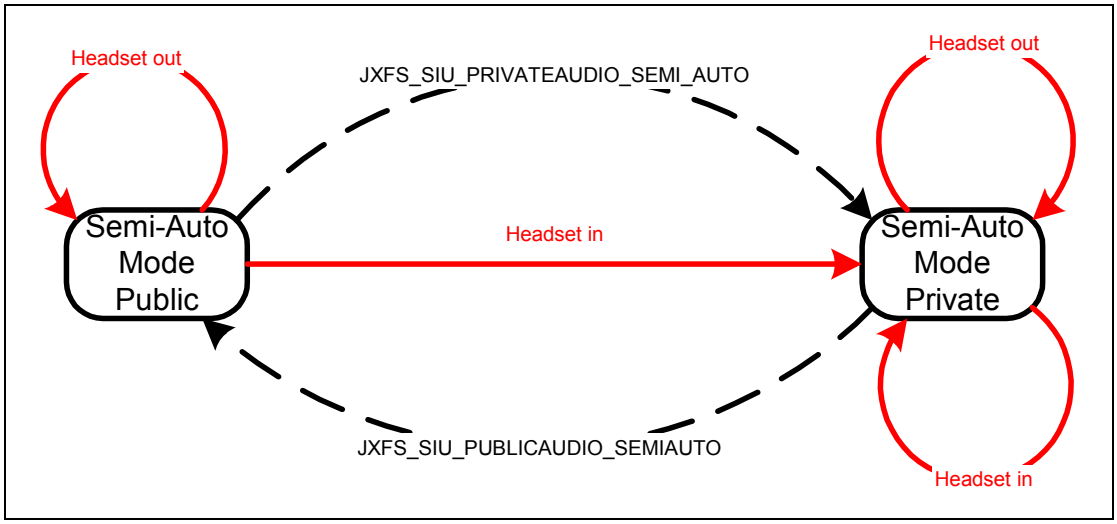
The dashed-line transitions are caused by application calls to setPorts for the JXFS_SIU_ENHANCEDAUDIOCONTROL auxiliary with values of JXFS_SIU_PRIVATEAUDIO_AUTO or JXFS_SIU_PUBLICAUDIO_AUTO

Semi-Auto Mode

This mode is required to ensure that customer sensitive information is not broadcasted via the public speakers when the consumer's headset is deliberately or otherwise unplugged.

In semi-auto mode, when a consumer headset is plugged into the jack, the audio is automatically directed to the headset and the audio is no longer sent to the speakers. When the headset is removed the audio remains via the jack. If required, the application must explicitly return the device to its public state if audio is required via the speakers. The following state diagram completely describes the behaviour of the device in semi-auto mode.

State Description
Semi-Auto Mode Public audio output is played through the public speakers only
Semi-Auto Mode Private audio is played through the consumer headset only



Semi-Auto-mode State diagram 2

The dashed-line transitions are caused by application calls to setPorts for the JXFS_SIU_ENHANCEDAUDIOCONTROL auxiliary with values of JXFS_SIU_PRIVATEAUDIO_SEMI_AUTO or JXFS_SIU_PUBLICAUDIO_SEMI_AUTO

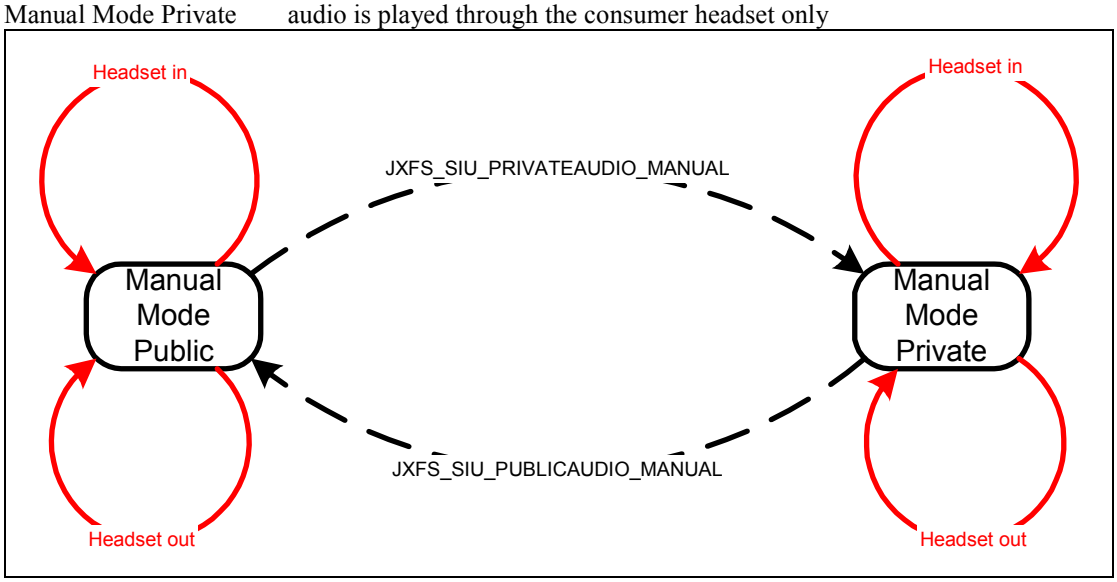
Manual mode

In manual mode, when a consumer headset is plugged into the jack, the audio remains directed at the existing interface (i. e. the speaker). The application must explicitly change to the other mode, if required. Note that the application must explicitly return the device to its public state if audio is required via the speakers. The following state diagram completely describes the behaviour of the device in manual mode

State Description

Manual Mode Public audio output is played through the public speakers

Manual Mode Private audio is played through the consumer headset only



Manual Mode State Diagram 3

The dashed-line transitions are caused by application calls to setPorts for the JXFS_SIU_ENHANCEDAUDIOCONTROL auxiliary with values of JXFS_SIU_PRIVATEAUDIO_MANUAL or JXFS_SIU_PUBLICAUDIO_MANUAL

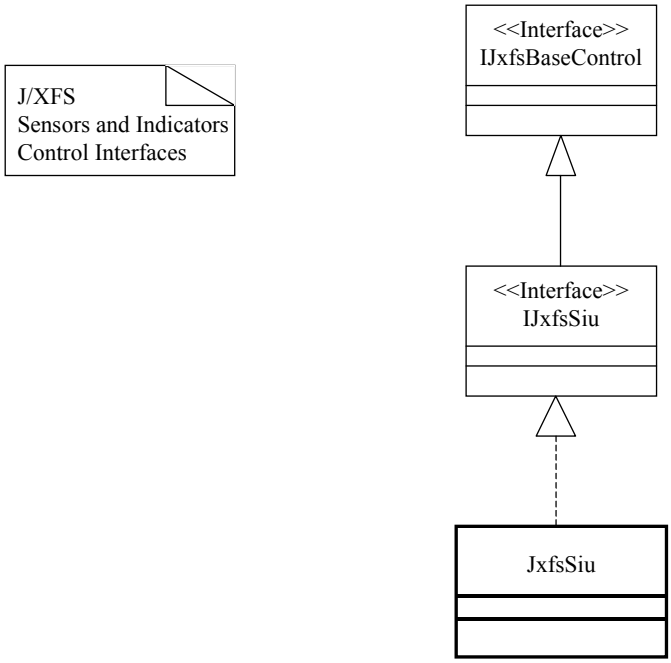
Inter-Mode behaviour

The values described in the previous sections (_AUTO, SEMI_AUTO, and _MANUAL) can also be used to move from one mode to another. This will then change the mode of the device.

Notes

- Note that if a vendor device does not support auto-mode or semi-auto mode then the JXFS_S_SIU_PORT_ERROR event is received on any attempt to call setPorts with the JXFS_SIU_PUBLICAUDIO_AUTO, JXFS_SIU_PRIVATEAUDIO_AUTO, JXFS_SIU_PUBLICAUDIO_SEMI_AUTO, and JXFS_SIU_PRIVATEAUDIO_SEMI_AUTO settings. The same event is generated if calls to change the mode to manual are received when the vendor device does not support manual mode.
- The existing *JXFS_SIU_VOLUME* auxiliary can be used to control the volume setting of any audio delivered to connected headset, as well as the speakers. Independent volume control of the speakers and headset is not supported.
- Any 'beep' tones generated by the PINPAD, etc will be fed to a connected headset (vendor hardware permitting).

5 Class Hierarchy



6 Class and Interface Summary

The following classes and interfaces are used by the J/XFS SIU Device Controls.

Class or Inter-face	Name	Description	Extends / Implements
Inter-face	**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**
Inter-face	**IJxfsSiu**	Base interface for all sensor and indicator controls.	Extends: **IJxfsBaseControl**
Class	**JxfsSiu**	Class for the SIU control	Extends: **JxfsBaseControl** Implements: **IJxfsSiu**

6.1 Support Classes

Class or Inter-face	Name	Description	Extends / Implements
Class	**JxfsSiuPortStatus**	Abstract class to represent a port status.	Extends: **JxfsType**
Class	**JxfsSiuSensorStatus**	Class to represent the current status of a sensor port.	Extends: **JxfsSiuPortStatus**
Class	**JxfsSiuDoorStatus**	Class to represent the current status of a door.	Extends: **JxfsSiuPortStatus**
Class	**JxfsSiuIndicatorStatus**	Class to represent the current status of an indicator port.	Extends: **JxfsSiuPortStatus**
Class	**JxfsSiuAuxiliaryStatus**	Class to represent the current status of an auxiliary indicator port.	Extends: **JxfsSiuPortStatus**
Class	**JxfsSiuGuidLightStatus**	Class to represent the current status of a guidance light.	Extends: **JxfsSiuPortStatus**
Class	**JxfsSiuStatus**	Class containing the whole status describing the status of all available ports.	Extends: **JxfsStatus**
Class	**JxfsSiuSensorCapability**	Class containing the capability information of a sensor port.	Extends: **JxfsType**
Class	**JxfsSiuDoorCapability**	Class containing the capability information of a door.	Extends: **JxfsType**
Class	**JxfsSiuIndicatorCapability**	Class containing the capability information of an indicator port.	Extends: **JxfsType**
Class	**JxfsSiuAuxiliaryCapability**	Class containing the capability information of an auxiliary indicator port.	Extends: **JxfsType**
Class	**JxfsSiuGuidLightCapability**	Class containing the capability information of a guidance light.	Extends: **JxfsType**
Class	**JxfsSiuCapabilities**	Class containing the capabilities of all available ports.	Extends: **JxfsType**
Class	**JxfsSiuEnable**	Class containing the information if changes of the port shall be reported.	Extends: **JxfsType**
Class	**JxfsSiuEnableEvents**	Class containing enable information for all available ports.	Extends: **JxfsType**
Class	**JxfsSiuDoorPort**	Class containing change information for a door port.	Extends: **JxfsType**
Class	**JxfsSiuIndicatorPort**	Class containing change information for an indicator port.	Extends: **JxfsType**
Class	**JxfsSiuAuxiliaryPort**	Class containing change information for an auxiliary port.	Extends: **JxfsType**

Class or Inter-face	Name	Description	Extends / Implements
Class	**JxfsSiuGuidLightPort**	Class containing change information for a guidance light.	Extends: **JxfsType**
Class	**JxfsSiuSetPorts**	Class containing change information for all available ports.	Extends: **JxfsType**
Class	**JxfsSiuSetDoor**	Class containing the change information for a specified door.	Extends: **JxfsType**
Class	**JxfsSiuSetIndicator**	Class containing the change information for a specified indicator port.	Extends: **JxfsType**
Class	**JxfsSiuSetAuxiliary**	Class containing the change information for a specified auxiliary indicator port.	Extends: **JxfsType**
Class	**JxfsSiuSetGuidLight**	Class containing the change information for a specified guidance light.	Extends: **JxfsType**
Class	**JxfsSiuPortChangeStatus**	Class containing information about a changed port status.	Extends: **JxfsStatus**
Class	**JxfsSiuPortError**	Class containing the information about an error on a specific port.	Extends: **JxfsStatus**
Inter-face	**JxfsConst**	Interface containing the JXFS constants that are common to several device categories	--
Inter-face	**JxfsSiuConst**	Interface containing the JXFS constants that are common to the SIU device control.	--
Class	**JxfsEvent**	Abstract class from which all JXFS event classes are extended	Extends: **java.util. EventObject**
Class	**Event**	The Device Service creates event instances of this class and delivers them through the J/XFS SIU Device Control's event callbacks to the application	Extends: **JxfsEvent**
Class	**JxfsException**	Exception class. The J/XFS SIU Device Control creates and throws exceptions on method failure and property access failure.	Extends: **java.lang.Exception**

7 Compatibility

The Sensors and Indicators Unit is one of the devices where it is most likely that it will be extended by other input and output ports in the upcoming versions of J/XFS. Therefore the design of the SIU device class interface takes such extensions into account to allow optimal forward and backward compatibility between device services and applications.

The input and output ports are organized as members of arrays or can be addressed via index values. This allows them to be extended in upcoming versions of J/XFS while remaining backward compatibilty. In this case we have to distinguish between two main cases:

a) New application, old device service

In this case the application should use the capabilities of the device service to investigate the ports that are supported by the current device service. The application should not rely on the existence of ports that have been defined in later versions of J/XFS than the initial version, but should make use of the length property of the arrays to see, if the device service knows this port.

b) Old application, new device service

To allow this case a device service has to accept arrays (when setting ports or enabling events) that are shorter than the number of ports supplied by this device service, but at least as long as the arrays in the initial version of J/XFS.

Another case in the area of compatibility are vendor special extensions. Like in WOSA/XFS it is possible to extend the arrays for the ports if an application and a device service agree on the extended ports. But it should be explicitly mentioned that these extensions are vendor specific and therefore not covered by the standard. Nevertheless it is recommended that these additional ports are not introduced as ports that directly succeed to the J/XFS specified ports, but should have a gap to be prepared for other ports that may be defined in upcoming versions of J/XFS.

In any case if an array is handed over from device service to the application or vice versa it must be ensured that all members of the array are instantiated.

8 Class and Interface Details

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

After processing has taken place, a *JxfsOperationCompleteEvent* 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.

8.1 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.

get*Property*

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

set*Property*

Syntax	**Property *set*Property*(value) throws JxfsException;***
Description	Sets the requested property.
Parameter	**Single parameter of property type.**
Event	No additional events are generated.
Exceptions	Some possible JxfsException *value codes*. See section on JxfsExceptions for other JxfsException value codes. JXFS_E_CLOSED JXFS_E_PARAMETER_INVALID JXFS_E_REMOTE JXFS_E_UNREGISTERED

8.2 IJxfsSiu

8.2.1 Introduction

The J/XFS Siu Device Control Subclass is defined in JxfsSiu and is a subclass of JxfsDeviceControl. Its interface is defined in IJxfsSiu which is a subclass of IJxfsBaseControl. The intent of the J/XFS SIU 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.

Whenever the value or the status of a port changes the appropriate events will be sent. More detailed information about the JXFS_S_SIU_PORT_STATUS and JXFS_S_SIU_PORT_ERROR events is available in the description of the JxfsSiuPortChangeStatus and JxfsSiuPortError support classes.

Summary

Property	Type	Access	Initialized after
capabilities	**JxfsSiuCapabilities**	R	successfull open()

Method	Return	May be used after
get*Property*	*Property*	
enableEvents	identificationID	
setPorts	identificationID	

8.2.2 Properties

capabilities (R)

Type ***JxfsSiuCapabilities***
Initial Value *Depends on device*
Description see *JxfsSiuCapabilities.*

Because a device service may detect the exact capabilities only after communicating with the hardware (plug&play hardware), this information may be available accurately only after a successfull open().

8.2.3 Methods

enableEvents

Syntax ***identificationID enableEvents(*JxfsSiuEnableEvents *events) throws JxfsException;***

Description This command is used to define the events that shall issue a status event in case of a change.

Parameter

Type	Name	Meaning
JxfsSiuEnableEvents	events	Specifies the events to be enabled.

Exceptions No additional exceptions generated.

Events Additional Events can be generated :

JxfsOperationCompleteEvent

When the enabling of events is completed a *JxfsOperationCompleteEvent* will be sent by J/XFS SIU Device Control to all registered *IJxfsOperationCompleteListeners* with the following data:

Field	Value
operationID	JXFS_O_SIU_ENABLE_EVENTS
identificationID	The corresponding ID
result	Common or device dependent error code. (See section on *Error Codes*).
data	none

setPorts

Syntax ***identificationID setPorts(JxfsSiuSetPorts ports) throws JxfsException;***

Description This method is used to set/change the current condition of an output port.

Parameter

Type	Name	Meaning
JxfsSiuSetPorts	ports	Specifies the ports to be changed and the values they shall be changed to.

Exceptions No additional exceptions generated.

Events Additional Events can be generated :

JxfsOperationCompleteEvent

When the selected ports have been changed a *JxfsOperationCompleteEvent* will be sent by J/XFS SIU Device Control to all registered *IJxfsOperationCompleteListeners* with the following data:

Field	Value
operationID	JXFS_O_SIU_SET_PORT
identificationID	The corresponding ID
result	Common or device dependent error code. (See section on *Error Codes*).
data	none

setPorts

Syntax ***identificationID setPorts(JxfsSiuSetDoor door) throws JxfsException;***

Description This method is used to set/change the current condition of a specific door port.

Parameter

Type	Name	Meaning
JxfsSiuSetDoor	door	Specifies the door to be changed and the value the door shall be changed to.

Exceptions No additional exceptions generated.

Events Additional Events can be generated :

JxfsOperationCompleteEvent

When the selected door has been changed a *JxfsOperationCompleteEvent* will be sent by J/XFS SIU Device Control to all registered *IJxfsOperationCompleteListeners* with the following data:

Field	Value
operationID	JXFS_O_SIU_SET_PORT
identificationID	The corresponding ID
result	Common or device dependent error code. (See section on *Error Codes*).
data	none

setPorts

Syntax ***identificationID setPorts(JxfsSiuSetIndicator indicator) throws JxfsException;***

Description This method is used to set/change the current condition of a specific indicator port.

Parameter

Type	Name	Meaning
JxfsSiuSetIndicator	indicator	Specifies the indicator to be changed and the value the indicator shall be changed to.

Exceptions No additional exceptions generated.

Events Additional Events can be generated :

JxfsOperationCompleteEvent

When the selected indicator has been changed a *JxfsOperationCompleteEvent* will be sent by J/XFS SIU Device Control to all registered *IJxfsOperationCompleteListeners* with the following data:

Field	Value
operationID	JXFS_O_SIU_SET_PORT
identificationID	The corresponding ID
result	Common or device dependent error code. (See section on *Error Codes*).
data	none

setPorts

Syntax ***identificationID setPorts(JxfsSiuSetAuxiliary auxiliary) throws JxfsException;***

Description This method is used to set/change the current condition of a specific auxiliary port.

Parameter

Type	Name	Meaning
JxfsSiuSetAuxiliary	auxiliary	Specifies the auxiliary to be changed and the value the auxiliary shall be changed to.

Exceptions No additional exceptions generated.

Events Additional Events can be generated :

JxfsOperationCompleteEvent

When the selected auxiliary indicator has been changed a *JxfsOperationCompleteEvent* will be sent by J/XFS SIU Device Control to all registered *IJxfsOperationCompleteListeners* with the following data:

Field	Value
operationID	JXFS_O_SIU_SET_PORT
identificationID	The corresponding ID
result	Common or device dependent error code. (See section on *Error Codes*).
data	none

setPorts

Syntax ***identificationID setPorts(JxfsSiuSetGuidLight guidLight) throws JxfsException;***

Description This method is used to set/change the current condition of a specific guidance light.

Parameter

Type	Name	Meaning
JxfsSiuSetGuidLight	guidLight	Specifies the guidance light to be changed and the value the guidance light shall be changed to.

Exceptions No additional exceptions generated.

Events Additional Events can be generated :

JxfsOperationCompleteEvent

When the selected guidance light has been changed a *JxfsOperationCompleteEvent* will be sent by J/XFS SIU Device Control to all registered *IJxfsOperationCompleteListeners* with the following data:

Field	Value
operationID	JXFS_O_SIU_SET_PORT
identificationID	The corresponding ID
result	Common or device dependent error code. (See section on *Error Codes*).
data	none

9 Support Classes

9.1 JxfsSiuPortStatus

This abstract class specifies the status of a port.

A port is always defined by the array index associated with the port.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
none	none		

Method	Return	May be used after
none	*none*	

Event	May occur after
none	

9.2 JxfsSiuSensorStatus

This class specifies the status of a sensor port.

Summary

Implements : *Serializable* **Extends :** *JxfsSiuPortStatus*

Property	Type	Access	Initialized after
sensorStatus	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSensorStatus	sensorStatus	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.2.1 Properties

sensorStatus (R)

Type ***int***
Initial Value none
Description Specifies the current status of the specific sensor port. The possible values and their meaning depend on the type of sensor port.

If any of these sensor ports is not available this is defined as

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The port is not available.

Specifies the state of the Operator Switch(es). This switch is used to tell the terminal if an Operator/Supervisor wants to change the state from Run to Operators/Supervisors mode or vice versa. The **Run** mode is used for normal consumer operations/transactions. The **Maintenance** mode is used when replenishing the terminal. The **Supervisor** mode is used when operating the terminal for service and testing. Supervisor mode has higher priority than maintenance mode. The state of an Operator switch is defined as one of the following flags:

Value	Meaning
JXFS_SIU_RUN	The switch is in Run Mode.
JXFS_SIU_MAINTENANCE	The switch is in Maintenance Mode.
JXFS_SIU_SUPERVISOR	The switch is in Supervisor mode.

Specifies the state of the Tamper Sensor for the terminal. This sensor indicates whether the terminal has been tampered with (such as a burglar attempt). The state of the Tamper Sensor is defined as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	There is no indication of a tampering attempt.
JXFS_SIU_ON	There has been a tampering attempt.

Specifies the state of the Tamper Sensor for the internal alarm. This sensor indicates whether the internal alarm has been tampered with (such as a burglar attempt). The state of the Tamper Sensor for the internal alarm is defined as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	There is no indication of a tampering attempt.
JXFS_SIU_ON	There has been a tampering attempt.

Specifies the state of the Seismic Sensor. This sensor indicates whether the terminal has been shaken (e.g. burglar attempt or seismic activity). Specified as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The seismic activity has not yet been high enough to trigger the sensor.
JXFS_SIU_ON	The seismic or other activity has triggered the sensor.

Specifies the state of the Heat Sensor. This sensor is triggered by excessive heat (fire) near the terminal. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The heat has not been high enough to trigger the sensor.
JXFS_SIU_ON	The heat has been high enough to trigger the sensor.

Specifies the state of the Proximity Sensor. This sensor is triggered by movements around the terminal. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_NOT_PRESENT	The sensor can not sense any people around the terminal.
JXFS_SIU_PRESENT	The sensor is showing that there is someone present at the terminal.

Specifies the state of the Ambient Light Sensor. This sensor indicates the level of ambient light around the terminal. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_VERY_LIGHT	The level of light is: very light.
JXFS_SIU_LIGHT	The level of light is: light.
JXFS_SIU_MEDIUM_LIGHT	The level of light is: medium light.
JXFS_SIU_DARK	The level of light is: dark.
JXFS_SIU_VERY_DARK	The level of light is: very dark.

Specifies the state of the first, second, third or fourth Input Contact. An external sensor can be connected to these contacts changing its state when the sensor is triggered. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The sensor was not triggered.
JXFS_SIU_ON	The sensor was triggered.

Specifies the state of the Ventilator. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_HWERROR	Due to a hardware error the ventilator is not running.
JXFS_SIU_ON	The ventilator is up and running.

Specifies the state of the Switch that indicates a Boot request. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_ON	The switch is set.
JXFS_SIU_OFF	The switch is not set.

Specifies the presence or otherwise of consumer headphone connected to the Audio Jack. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_PRESENT	There is a headset connected.
JXFS_SIU_NOT_PRESENT	There is no headset connected

9.3 JxfsSiuDoorStatus

This class specifies the status of a door.

Summary

Implements : *Serializable* **Extends :** *JxfsSiuPortStatus*

Property	Type	Access	Initialized after
doorStatus	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuDoorStatus	doorStatus	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.3.1 Properties

doorStatus (R)

Type ***int***
Initial Value none
Description Specifies the current status of the specific door. The possible values and their meaning depend on the type of door.

If any of these door ports is not available then this is defined as

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The status is not available.

Specifies the state of the Cabinet Doors. Cabinet Doors are doors that open up for consumables, and hardware that does not have to be in a secure place. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_OPEN	At least one of the Cabinet Doors is open.
JXFS_SIU_CLOSED	The Cabinet Doors are closed.
JXFS_SIU_LOCKED	The Cabinet Doors are closed and locked.
JXFS_SIU_BOLTED	The Cabinet Doors are closed, locked and bolted.

Specifies the state of the Safe Doors. Safe Doors are doors that open up for secure hardware, such as the note dispenser, the security device, etc. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_OPEN	At least one of the Safe Doors is open.
JXFS_SIU_CLOSED	The Safe Doors are closed.
JXFS_SIU_LOCKED	The Safe Doors are closed and locked.
JXFS_SIU_BOLTED	The Safe Doors are closed, locked and bolted.

Specifies the state of the Vandal Shield. The Vandal Shield is a door that open up for consumer access to the terminal. Specified as one of the following flags:

Value	Meaning
JXFS_SIU_OPEN	The Vandal Shield is open.
JXFS_SIU_CLOSED	The Vandal Shield is closed.
JXFS_SIU_LOCKED	The Vandal Shield closed and locked.
JXFS_SIU_SERVICE	The Vandal Shield is in service position.
JXFS_SIU_KEYBOARD	The Vandal Shield position permits access to the keyboard
JXFS_SIU_AJAR	The Vandal Shield is ajar.
JXFS_SIU_JAMMED	The Vandal Shield is jammed.

Specifies the state of the Front Top Door, the Rear Top Door, the Front Bottom Door or the Rear Bottom Door. Specified as one of the following flags.

Value	Meaning
JXFS_SIU_OPEN	The door is open.
JXFS_SIU_CLOSED	The door is closed.
JXFS_SIU_BOLTED	The door is closed and bolted.

9.4 JxfsSiuIndicatorStatus

This class specifies the status of an indicator.

Summary

Implements : *Serializable* **Extends :** *JxfsSiuPortStatus*

Property	Type	Access	Initialized after
indicatorStatus	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuIndicatorStatus	indicatorStatus	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.4.1 Properties

indicatorStatus (R)

Type ***int***
Initial Value none
Description Specifies the current status of the specific indicator. The possible values and their meaning depend on the type of indicator.

If any of the indicator ports is not available then this is defined as:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The status is not available.

Specifies the state of the Open/Closed Indicator as one of the following flags:

Value	Meaning
JXFS_SIU_CLOSED	The terminal is closed for a consumer.
JXFS_SIU_OPEN	The terminal is open to be used by a consumer.

Specifies the state of the Fascia Light as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The Fascia Light is turned off.
JXFS_SIU_ON	The Fascia Light is turned on.

Specifies the state of the Audio Indicator as one of the following flags of type A and B, or as JXFS_SIU_CONTINUOUS in combination with one of the flags of type B:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	The status is not available.	A
JXFS_SIU_OFF	The Audio Indicator is turned off.	A
JXFS_SIU_KEYPRESS	The Audio Indicator sounds a key click signal.	B
JXFS_SIU_EXCLAMATION	The Audio Indicator sounds an exclamation signal.	B
JXFS_SIU_WARNING	The Audio Indicator sounds a warning signal.	B
JXFS_SIU_ERROR	The Audio Indicator sounds an error signal.	B
JXFS_SIU_CRITICAL	The Audio Indicator sounds a critical signal	B
JXFS_SIU_CONTINUOUS	The Audio Indicator sound is turnd on continuously.	C

Example:

If there is no audio signal active, the value is:
JXFS_SIU_OFF

If there is a continuous error signal, the value is:
JXFS_SIU_ERROR | JXFS_SIU_CONTINUOUS.

Specifies the state of the internal heating as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The Heating is turned off.
JXFS_SIU_ON	The Heating is turned on.

Specifies the state of the Logo Light as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The Logo Light is turned off.
JXFS_SIU_ON	The Logo Light is turned on.

9.5 JxfsSiuAuxiliaryStatus

This class specifies the status of the auxiliary indicators.

Summary

Implements : *Serializable* **Extends :** *JxfsSiuPortStatus*

Property	Type	Access	Initialized after
auxiliaryStatus	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuAuxiliaryStatus	auxiliaryStatus	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.5.1 Properties

auxiliaryStatus (R)

Type ***int***
Initial Value none
Description Specifies the current status of the specific auxiliary indicator. The possible values and their meaning depend on the type of auxiliary indicator.

If any of the auxiliary indicators is not available this is defined by:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The port is not available.

Specifies the value of the volume control. The value of volume control is defined in an interval from 1 to 1000 where 1 is the lowest volume level and 1000 is the highest volume level. The interval is defined in logarithmic steps, e.g. a volume control on a radio.

Value	Meaning
1, ..., 1000	The volume level.

Specifies the state of the Uninterruptable Power Supply device as WFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no UPS available.	A
JXFS_SIU_AVAILABLE	The UPS is available.	B
JXFS_SIU_LOW	The charge level of the UPS is low.	B
JXFS_SIU_ENGAGED	The UPS is engaged.	B
JXFS_SIU_POWERING	The UPS is powering the system. The main power supply is off.	B
JXFS_SIU_RECOVERED	The UPS was engaged when the main power went off.	B

Specifies the state of the Monitor as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The Monitor is turned off.
JXFS_SIU_ON	The Monitor is turned on.

Specifies the state of the software Poweroff as one of the following flags:

Value	Meaning
JXFS_SIU_AVAILABLE	A software poweroff is available/possible.

Specifies the state of the Relays as one of the following flags:

Value	Meaning
JXFS_SIU_OFF	The Relay is turned off.
JXFS_SIU_ON	The Relay is turned on.

Specifies the state of the Audio Jack controller (JXFS_ENHANCEDAUDIOCONTROL) as one of the following flags:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	There is no Audio Jack Controller available
JXFS_SIU_PUBLICAUDIO_MANUAL	The Audio Jack is in manual-mode and is in the public state (i.e. audio will be played through speakers). Connecting a headset will have no impact, ie Output will remain through the speakers & no audio will be directed to the headset.
JXFS_SIU_PUBLICAUDIO_AUTO	The Audio Jack in in auto-mode and is in the public state (i.e. audio will be played through speakers). When a headset is connected, the device will go to the private state.
JXFS_SIU_PUBLICAUDIO_SEMI_AUTO	The Audio Jack is in semi-auto mode and is in the public state (i.e. audio will be played through speakers). When a headset is connected, the device will go to the private state
JXFS_SIU_PRIVATEAUDIO_MANUAL	The Audio Jack is in manual-mode and is in the private state (i.e. audio will be played only through a connected headset). In private mode, no audio is transmitted through the speakers.
JXFS_SIU_PRIVATEAUDIO_AUTO	The Audio Jack is in auto-mode and is in the private state (i.e. audio will be played only through a connected headset). In private mode, no audio is transmitted through the speakers.

JXFS_SIU_PRIVATEAUDIO_SEMI_AUTO	The Audio Jack is in semi-auto mode and is in the private state (i.e. audio will be played only through a connected headset). In private mode, no audio is transmitted through the speakers. When a headset is disconnected, the device will remain in the private state

9.6 JxfsSiuGuidLightStatus

This class specifies the status of the guidance lights.

Summary

Implements : *Serializable* **Extends :** *JxfsSiuPortStatus*

Property	Type	Access	Initialized after
guidlightStatus	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuGuidLightStatus	guidlightStatus	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.6.1 Properties

guidlightStatus (R)

Type ***int***
Initial Value none
Description Specifies the current status of the specific guidance light.

The status of the guidance lights is one of the following values:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The status is not available.
JXFS_SIU_OFF	The light is turned off.
JXFS_SIU_SLOW_FLASH	The light is blinking slowly.
JXFS_SIU_MEDIUM_FLASH	The light is blinking medium frequency.
JXFS_SIU_QUICK_FLASH	The light is blinking quickly.
JXFS_SIU_CONTINUOUS	The light is turned on continuously (steady).

If the application chooses a value which is not supported by the guidance light the device service - as it knows the capabilities of its Device - should map the value to the next best state and should not return JXFS_RC_UNSUCCESSFUL.

9.7 JxfsSiuStatus

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

The JxfsSiuStatus object is the subclass of JxfsStatus, returned in response to a getStatus method call.

The implementation of the Properties as arrays allows them to be extended by other ports of the same type (sensors, doors, indicators, etc.), if the implementation requires this. This way it is possible to extend this status in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

Null references as properties of the JxfsSiuStatus class are not allowed. All arrays must be present and all elements of the arrays must exist. The length of the arrays must be at least as long as defined in the initial J/XFS CWA. If more than one object of an array is not represented by a physical port, the references may refer to the same object represented as JXFS_SIU_NOT_AVAILABLE.

9.7.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
sensorStatus	**JxfsSiuSensorStatus []**	R	
doorStatus	**JxfsSiuDoorStatus[]**	R	
indicatorStatus	**JxfsSiuIndicatorStatus[]**	R	
auxiliaryStatus	**JxfsSiuAuxiliaryStatus[]**	R	
guidlightStatus	**JxfsSiuGuidLightStatus[]**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuStatus	sensorStatus	JxfsSiuSensorStatus[]
	doorStatus	JxfsSiuDoorStatus[]
	indicatorStatus	JxfsSiuIndicatorStatus[]
	auxiliaryStatus	JxfsSiuAuxiliaryStatus[]
	guidlightStatus	JxfsSiuGuidLightStatus[]

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.7.2 Properties

sensorStatus[JXFS_SIU_OPERATORSWITCH]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the Operator Switch(es). This switch is used to tell the terminal if an Operator/Supervisor wants to change the state from Run to Operators/Supervisors mode or vice versa. The **Run** mode is used for normal consumer operations/transactions. The **Maintenance** mode is used when replenishing the terminal. The **Supervisor** mode is used when operating the terminal for service and testing. Supervisor mode has higher priority than maintenance mode.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_TAMPER]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the Tamper Sensor for the terminal. This sensor indicates whether the terminal has been tampered with (such as a burglar attempt).

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_INTTAMPER]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the Tamper Sensor for the internal alarm. This sensor indicates whether the internal alarm has been tampered with (such as a burglar attempt).

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_SEISMIC]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the Seismic Sensor. This sensor indicates whether the terminal has been shaken (e.g. burglar attempt or seismic activity).

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_HEAT]

Type ***JxfsSiuSensorStatus***

Description Specifies the state of the Heat Sensor. This sensor is triggered by excessive heat (fire) near the terminal.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_PROXIMITY]

Type ***JxfsSiuSensorStatus***

Description Specifies the state of the Proximity Sensor. This sensor is triggered by movements around the terminal.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_AMBLIGHT]

Type ***JxfsSiuSensorStatus***

Description Specifies the state of the Ambient Light Sensor. This sensor indicates the level of ambient light around the terminal.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_INPUT1]

Type ***JxfsSiuSensorStatus***

Description Specifies the state of the first Input Contact. An external sensor can be connected to this contact changing its state when the sensor is triggered.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_INPUT2]

Type ***JxfsSiuSensorStatus***

Description Specifies the state of the second Input Contact. An external sensor can be connected to this contact changing its state when the sensor is triggered.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_INPUT3]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the third Input Contact. An external sensor can be connected to this contact changing its state when the sensor is triggered.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_INPUT4]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the fourth Input Contact. An external sensor can be connected to this contact changing its state when the sensor is triggered.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_VENTILATOR]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the Ventilator.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_BOOTSWITCH]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the Switch that indicates a Boot request.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

sensorStatus[JXFS_SIU_ENHANCEDAUDIO]

Type *JxfsSiuSensorStatus*

Description Specifies the state of the Audio Jack.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

doorStatus[JXFS_SIU_CABINET]

Type ***JxfsSiuDoorStatus***

Description Specifies the state of the Cabinet Doors. Cabinet Doors are doors that open up for consumables, and hardware that does not have to be in a secure place.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

doorStatus[JXFS_SIU_SAFE]

Type ***JxfsSiuDoorStatus***

Description Specifies the state of the Safe Doors. Safe Doors are doors that open up for secure hardware, such as the note dispenser, the security device, etc.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

doorStatus[JXFS_SIU_VANDALSHIELD]

Type ***JxfsSiuDoorStatus***

Description Specifies the state of the Vandal Shield. The Vandal Shield is a door that opens up for consumer access to the terminal.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

doorStatus[JXFS_SIU_FRONT_TOP]

Type ***JxfsSiuDoorStatus***

Description Specifies the state of the Front Top Door.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

doorStatus[JXFS_SIU_REAR_TOP]

Type ***JxfsSiuDoorStatus***

Description Specifies the state of the Rear Top Door.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

doorStatus[JXFS_SIU_FRONT_BOTTOM]

Type ***JxfsSiuDoorStatus***
Description Specifies the state of the Front Bottom Door.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

doorStatus[JXFS_SIU_REAR_BOTTOM]

Type ***JxfsSiuDoorStatus***
Description Specifies the state of the Rear Bottom Door.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

indicatorStatus[JXFS_SIU_OPENCLOSE]

Type ***JxfsSiuIndicatorStatus***
Description Specifies the state of the Open/Closed Indicator.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

indicatorStatus[JXFS_SIU_FASCIALIGHT]

Type ***JxfsSiuIndicatorStatus***
Description Specifies the state of the Fascia Light.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

indicatorStatus[JXFS_SIU_LOGOLIGHT]

Type ***JxfsSiuIndicatorStatus***
Description Specifies the state of the Logo Light.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

indicatorStatus[JXFS_SIU_AUDIO]

Type *JxfsSiuIndicatorStatus*
Description Specifies the state of the Audio Indicator.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

indicatorStatus[JXFS_SIU_HEATING]

Type *JxfsSiuIndicatorStatus*
Description Specifies the state of the internal heating
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_VOLUME]

Type *JxfsSiuAuxiliaryStatus*
Description Specifies the value of the volume control. The value of volume control is defined in an interval from 1 to 1000 where 1 is the lowest volume level and 1000 is the highest volume level. The interval is defined in logarithmic steps, e.g. a volume control on a radio. Note: The volume control field is handled as unsigned short.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_UPS]

Type *JxfsSiuAuxiliaryStatus*
Description Specifies the state of the Uninterruptable Power Supply device.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_MONITOR]

Type *JxfsSiuAuxiliaryStatus*
Description Specifies the state of the Monitor.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_POWEROFF]

Type	***JxfsSiuAuxiliaryStatus***
Description	Specifies the state of the software Poweroff.
Event	If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	**Meaning**
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_RELAY1]

Type	***JxfsSiuAuxiliaryStatus***
Description	Specifies the state of the first Relay.
Event	If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	**Meaning**
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_RELAY2]

Type	***JxfsSiuAuxiliaryStatus***
Description	Specifies the state of the second Relay.
Event	If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	**Meaning**
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_RELAY3]

Type	***JxfsSiuAuxiliaryStatus***
Description	Specifies the state of the third Relay.
Event	If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	**Meaning**
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_RELAY4]

Type	***JxfsSiuAuxiliaryStatus***
Description	Specifies the state of the fourth Relay.
Event	If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	**Meaning**
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

auxiliaryStatus[JXFS_SIU_ENHANCEDAUDIOCONTROL]

Type *JxfsSiuAuxiliaryStatus*
Description Specifies the state of the Audio Jack control.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_CARDUNIT]

Type *JxfsSiuGuidLightStatus*
Description Specifies the state of the Guidance Light Indicator on the Card Unit (MSD/CCD).
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_PINPAD]

Type *JxfsSiuGuidLightStatus*
Description Specifies the state of the Guidance Light Indicator on the PIN pad unit.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_NOTESDISPENSER]

Type *JxfsSiuGuidLightStatus*
Description Specifies the state of the Guidance Light Indicator on the note dispenser unit.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_COINDISPENSER]

Type *JxfsSiuGuidLightStatus*
Description Specifies the state of the Guidance Light Indicator on the coin dispenser unit.
Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_RECEIPTPRINTER]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the receipt printer unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_PASSBOOKPRINTER]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the passbook printer unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_ENVDEPOSITORY]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the envelope depository unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_CHEQUEUNIT]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the cheque processing unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_BILLACCEPTOR]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the bill acceptor unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_ENVDISPENSER]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the envelope

dispenser unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_SCANNER]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the scanner unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_COINACCEPTOR]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the coin acceptor unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

guidlightStatus[JXFS_SIU_DOCUMENTPRINTER]

Type ***JxfsSiuGuidLightStatus***

Description Specifies the state of the Guidance Light Indicator on the document printer unit.

Event If the value of this property changes and the Device Control has registered for the change of this property, the Device Service will send all registered *IJxfsStatusListeners* a JxfsStatusEvent with a status value of:

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

9.8 JxfsSiuSensorCapability

This class specifies the capabilities of a sensor port.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
sensorCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSensorCapability	sensorCapability	int

Method	Return	May be used after
get*Property*	*Property*	
isRunModeSupported	boolean	
isMaintenanceModeSupported	boolean	
isSupervisorModeSupported	boolean	
isAvailable	boolean	
isManualModeSupported	boolean	
isAutoModeSupported	boolean	
isSemiAutoModeSupported	boolean	

Event	May occur after
none	

9.8.1 Properties

sensorCapability (R)

Type ***int***
Initial Value none
Description Specifies the capability of the specific sensor port. The possible values and their meaning depend on the type of sensor port.

Specified as JXFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B for the operator switch:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no Operator Switch available.	A
JXFS_SIU_RUN	The switch can be set in Run mode.	B
JXFS_SIU_MAINTENANCE	The switch can be set in maintenance mode.	B
JXFS_SIU_SUPERVISOR	The switch can be set in Supervisors mode.	B

Example:

If you have an operator switch that has two positions, one for the normal mode and one for a maintenance mode, the value would be:

JXFS_SIU_RUN | JXFS_SIU_MAINTENANCE

Specifies whether the Audio Jack is present, and if so, which modes it supports. Specified as JXFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B.

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no Audio Jack available.	A
JXFS_SIU_MANUAL	The Audio Jack is available and supports manual-mode.	B
JXFS_SIU_AUTO	The Audio Jack is available and supports auto-mode.	B
JXFS_SIU_SEMI_AUTO	The Audio Jack is available and supports semi-auto-mode.	B

For all other sensor ports the possible values are one of the following flags:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The specified sensor port is not available.
JXFS_SIU_AVAILABLE	The specified sensor port is available.

9.8.2 Methods

isRunModeSupported

Syntax	***boolean isRunModeSupported(void);***
Description	Returns TRUE if the sensor is an Operator switch and the Run mode is supported by this kind of switch (the *sensorCapability* property contains the value JXFS_SIU_RUN).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isMaintenanceModeSupported

Syntax	***boolean isMaintenanceModeSupported(void);***
Description	Returns TRUE if the sensor is an Operator switch and the Maintenance mode is supported by this kind of switch (the *sensorCapability* property contains the value JXFS_SIU_MAINTENANCE).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isSupervisorModeSupported

Syntax	***boolean isSupervisorModeSupported(void);***
Description	Returns TRUE if the sensor is an Operator switch and the Supervisor mode is supported by this kind of switch (the *sensorCapability* property contains the value JXFS_SIU_SUPERVISOR).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isAvailable

Syntax	***boolean isAvailable(void);***
Description	Returns TRUE if the sensor is not an Operator switch and the sensor port is supported (the *sensorCapability* property contains the value JXFS_SIU_AVAILABLE).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isManualModeSupported

Syntax	***boolean isManualModeSupported(void);***
Description	Returns TRUE if the sensor is an Enhanced Audio sensor and if the Audio Jack is present and if the Audio Jack supports the manual mode.
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isAutoModeSupported

Syntax	***boolean isAutoModeSupported(void);***
Description	Returns TRUE if the sensor is an Enhanced Audio sensor and if the Audio Jack is present and if the Audio Jack supports the auto mode.
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isSemiAutoModeSupported

Syntax	***boolean isSemiAutoModeSupported(void);***
Description	Returns TRUE if the sensor is an Enhanced Audio sensor and if the Audio Jack is present and if the Audio Jack supports the semi-auto mode.
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

9.9 JxfsSiuDoorCapability

This class specifies the capability of a door.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
doorCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuDoorCapability	doorCapability	int

Method	Return	May be used after
get*Property*	*Property*	
isLockedSupported	boolean	
isBoltedSupported	boolean	
isClosedSupported	boolean	
isOpenSupported	boolean	
isServiceSupported	boolean	
isKeyboardSupported	boolean	
isAjarSupported	boolean	
isJammedSupported	boolean	

Event	May occur after
none	

9.9.1 Properties

doorCapability (R)

Type *int*
Initial Value none
Description Specifies the capabilities of the specific door. The possible values and their meaning depend on the type of door.

Specifies the capabilities of the Cabinet Doors or the Safe Doors and the states they can take if present. Specified as JXFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no Cabinet/Safe Door available.	A
JXFS_SIU_LOCKED	The Cabinet/Safe Doors can be locked.	B
JXFS_SIU_BOLTED	The Cabinet/Safe Doors can be bolted.	B
JXFS_SIU_CLOSED	The Cabinet/Safe Doors can be closed.	B
JXFS_SIU_OPEN	The Cabinet/Safe Doors can be opened.	B

Specifies the capabilities of the Vandal Shield Doors and the states they can take if present. Specified as JXFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no Vandal Shield available.	A
JXFS_SIU_LOCKED	The Vandal Shield can be locked.	B
JXFS_SIU_SERVICE	The Vandal Shield can be in service position.	B
JXFS_SIU_CLOSED	The Vandal Shield can be closed.	B
JXFS_SIU_OPEN	The Vandal Shield can be open.	B
JXFS_SIU_KEYBOARD	The Vandal Shield can be in position that permits access to the keyboard.	B
JXFS_SIU_AJAR	The Vandal Shield can be ajar.	B
JXFS_SIU_JAMMED	The Vandal Shield can be jammed.	B

Specifies the capabilities of Front Top/Rear Top/Front Bottom/Rear Bottom Doors and the states they can take if present. Specified as JXFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no Door available.	A
JXFS_SIU_BOLTED	The Door can be bolted.	B
JXFS_SIU_CLOSED	The Door can be closed.	B
JXFS_SIU_OPEN	The Door can be opened.	B

9.9.2 Methods

isLockedSupported

Syntax	***boolean isLockedSupported(void);***
Description	Returns TRUE if the door can be locked (the *doorCapability* property contains the value JXFS_SIU_LOCKED).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isBoltedSupported

Syntax	***boolean isBoltedSupported(void);***
Description	Returns TRUE if the door can be bolted (the *doorCapability* property contains the value JXFS_SIU_BOLTED).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isClosedSupported

Syntax	***boolean isClosedSupported(void);***
Description	Returns TRUE if the door can be closed (the *doorCapability* property contains the value JXFS_SIU_CLOSED).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isOpenSupported

Syntax	***boolean isOpenSupported(void);***
Description	Returns TRUE if the door can be open (the *doorCapability* property contains the value JXFS_SIU_OPEN).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isServiceSupported

Syntax	***boolean isServiceSupported(void);***
Description	Returns TRUE if the door can be in Service position (the *doorCapability* property contains the value JXFS_SIU_SERVICE).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isKeyboardSupported

Syntax	***boolean isKeyboardSupported(void);***
Description	Returns TRUE if the door can be put in a position that allows access to the keyboard (the *doorCapability* property contains the value JXFS_SIU_KEYBOARD).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isAjarSupported

Syntax	***boolean isAjarSupported(void);***
Description	Returns TRUE if the door can be ajar (the *doorCapability* property contains the value JXFS_SIU_AJAR).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isJammedSupported

Syntax	***boolean isJammedSupported(void);***
Description	Returns TRUE if the door can be jammed (the *doorCapability* property contains the value JXFS_SIU_JAMMED).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

9.10 JxfsSiuIndicatorCapability

This class specifies the capability of an indicator.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
indicatorCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuIndicatorCapability	indicatorCapability	int

Method	Return	May be used after
get*Property*	*Property*	
isAvailable	boolean	

Event	May occur after
none	

9.10.1 Properties

indicatorCapability (R)

Type ***int***
Initial Value none
Description Specifies which indicators are available and which states they can take.

Specifies the capabilities of an indicator as one of the following values:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The indicator is not available.
JXFS_SIU_AVAILABLE	The indicator is available.

9.10.2 Methods

isAvailable

Syntax ***boolean isAvailable(void);***
Description Returns TRUE if the indicator is available (the *indicatorCapability* property contains the value JXFS_SIU_AVAILABLE).
Parameter **None**
Exceptions No additional exceptions are generated.
Event No additional events are generated.

9.11 JxfsSiuAuxiliaryCapability

This class specifies the capabilities of the auxiliary indicators.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
auxiliaryCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuAuxiliaryCapability	auxiliaryCapability	int

Method	Return	May be used after
get*Property*	*Property*	
isAvailable	boolean	
isLowSupported	boolean	
isEngagedSupported	boolean	
isPoweringSupported	boolean	
isRecoveredSupported	boolean	

Event	May occur after
none	

9.11.1 Properties

auxiliaryCapability (R)

Type	***int***
Initial Value	none
Description	Specifies which auxiliary indicators are available and which states they can take. The possible values depend on the type of auxiliary indicator. Specifies the capabilities of the volume control as one of the following values:

Value	**Meaning**
JXFS_SIU_NOT_AVAILABLE	There is no volume control available.
1, ..., 1000	The recommended increment / decrement value for the volume control.

Specifies if the UPS is available, and if so, which states it can take. Specified as JXFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no UPS available.	A
JXFS_SIU_AVAILABLE	The UPS is available.	B
JXFS_SIU_LOW	The UPS can indicate that its charge level is low.	B
JXFS_SIU_ENGAGED	The UPS can be engaged and disengaged by the application.	B
JXFS_SIU_POWERING	The UPS can indicate that it is powering the system while the main power supply is off.	B
JXFS_SIU_RECOVERED	The UPS can indicate that it was engaged when the main power went off.	B

Specifies whether the Audio Jack Controller (index JXFS_SIU_ENHANCEDAUDIOCONTROL) is available, and if so, which modes it supports. Specified as JXFS_SIU_NOT_AVAILABLE or as a combination of the following flags of type B:

Value	Meaning	Type
JXFS_SIU_NOT_AVAILABLE	There is no Audio Jack available.	A
JXFS_SIU_HEADSET_DETECTION	The Audio Jack is available and supports headset insertion & removal. The device is able to report events to indicate headset insertion & removal.	B
JXFS_SIU_MODE_CONTROLLABLE	The Audio Jack is available and supports application control of the Audio Jack mode via the setPorts() method.	B

Specifies the capabilities of auxiliary indicators other than volume control and UPS service and JXFS_SIU_ENHANCEDAUDIOCONTROL as one of the following values:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The indicator is not available.
JXFS_SIU_AVAILABLE	The indicator is available.

9.11.2 Methods

isAvailable

Syntax	***boolean isAvailable(void);***
Description	Returns TRUE if the auxiliary indicator is available (the *auxiliaryCapability* property contains the value JXFS_SIU_AVAILABLE).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isLowSupported

Syntax	***boolean isLowSupported(void);***
Description	Returns TRUE if the UPS can indicate that its charge level is low (the *auxiliaryCapability* property contains the value JXFS_SIU_LOW).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isEngagedSupported

Syntax	***boolean isEngagedSupported(void);***
Description	Returns TRUE if the UPS can be engaged and disengaged by the application (the *auxiliaryCapability* property contains the value JXFS_SIU_LOW).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isPoweringSupported

Syntax	***boolean isPoweringSupported(void);***
Description	Returns TRUE if the UPS can indicate that it is powering the system while the main power supply is off (the *auxiliaryCapability* property contains the value JXFS_SIU_POWERING).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

isRecoveredSupported

Syntax	***boolean isRecoveredSupported(void);***
Description	Returns TRUE if the UPS can indicate that it was engaged when the main power went off (the *auxiliaryCapability* property contains the value JXFS_SIU_RECOVERED).
Parameter	**None**
Exceptions	No additional exceptions are generated.
Event	No additional events are generated.

9.12 JxfsSiuGuidLightCapability

This class specifies the capability of a guidance light

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
guidLightCapability	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuGuidLightCapability	guidLightCapability	int

Method	Return	May be used after
get*Property*	*Property*	
isAvailable	boolean	

Event	May occur after
none	

9.12.1 Properties

guidLightCapability (R)

Type ***int***
Initial Value none
Description Specifies which guidance lights are available and which states they can take.

Specifies the capabilities of a guidance light as one of the following values:

Value	Meaning
JXFS_SIU_NOT_AVAILABLE	The indicator is not available at this position or the device service controls the indicator directly with no application control possible.
JXFS_SIU_AVAILABLE	The indicator is available.

9.12.2 Methods

isAvailable

Syntax ***boolean isAvailable(void);***
Description Returns TRUE if the guidance light is available (the *guidLightCapability* property contains the value JXFS_SIU_AVAILABLE).
Parameter **None**
Exceptions No additional exceptions are generated.
Event No additional events are generated.

9.13 JxfsSiuCapabilities

This class contains properties and methods to query the capabilities and functionalities of the SIU device and its resources.

The implementation of the Properties as arrays allows them to be extended by other ports of the same type (sensors, doors, indicators, etc.), if the implementation requires this. This way it is possible to extend the capabilities in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

Null references as properties of the JxfsSiuCapabilities class are not allowed. All arrays must be present and all elements of the arrays must exist. The length of the arrays must be at least as long as defined in the initial J/XFS CWA. If more than one object of an array is not represented by a physical port, the references may refer to the same object represented as JXFS_SIU_NOT_AVAILABLE.

9.13.1 Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
sensorCapabilities	**JxfsSiuSensorCapability[]**	R	
doorCapabilities	**JxfsSiuDoorCapability[]**	R	
indicatorCapabilities	**JxfsSiuIndicatorCapability[]**	R	
auxiliaryCapabilities	**JxfsSiuAuxiliaryCapability []**	R	
guidLightCapabilities	**JxfsSiuGuidLightCapability[]**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuCapabilities	sensorCapabilities	JxfsSiuSensorCapability[]
	doorCapabilities	JxfsSiuDoorCapability[]
	indicatorCapabilities	JxfsSiuIndicatorCapability[]
	auxiliaryCapabilities	JxfsSiuAuxiliaryCapability[]
	guidLightCapabilities	JxfsSiuGuidLightCapability[]

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.13.2 Properties

sensorCapabilities[JXFS_SIU_OPERATORSWITCH]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Operator switch is available, and if so, which states it can take.
Event	none

sensorCapabilities[JXFS_SIU_TAMPER]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Tamper Sensor is available.
Event	none

sensorCapabilities[JXFS_SIU_INTTAMPER]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Tamper Sensor for internal alarm is available.
Event	none

sensorCapabilities[JXFS_SIU_SEISMIC]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Seismic Sensor is available.
Event	none

sensorCapabilities[JXFS_SIU_HEAT]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Heat Sensor is available.
Event	none

sensorCapabilities[JXFS_SIU_PROXIMITY]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Proximity Sensor is available.
Event	none

sensorCapabilities[JXFS_SIU_AMBLIGHT]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Ambient Light Sensor is available.
Event	none

sensorCapabilities[JXFS_SIU_INPUT1]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the first Input Contact is available.
Event	none

sensorCapabilities[JXFS_SIU_INPUT2]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the second Input Contact is available.
Event	none

sensorCapabilities[JXFS_SIU_INPUT3]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the third Input Contact is available.
Event	none

sensorCapabilities[JXFS_SIU_INPUT4]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the fourth Input Contact is available.
Event	none

sensorCapabilities[JXFS_SIU_VENTILATOR]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Ventilator is available.
Event	none

sensorCapabilities[JXFS_SIU_BOOTSWITCH]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Switch that indicates a Boot request is available.
Event	none

sensorCapabilities[JXFS_SIU_ENHANCEDAUDIO]

Type	***JxfsSiuSensorCapabilitiy***
Initial Value	0
Description	Specifies whether the Audio Jack sensor is available.
Event	none

doorCapabilities[JXFS_SIU_CABINET]

Type	***JxfsSiuDoorCapabilitiy***
Initial Value	0
Description	Specifies whether the Cabinet Doors are available, and if so, which states they can take.
Event	none

doorCapabilities[JXFS_SIU_SAFE]

Type	***JxfsSiuDoorCapabilitiy***
Initial Value	0
Description	Specifies whether the Safe Doors are available, and if so, which states they can take.
Event	none

doorCapabilities[JXFS_SIU_VANDALSHIELD]

Type	***JxfsSiuDoorCapabilitiy***
Initial Value	0
Description	Specifies whether the Vandal Shield is available, and if so, which states it can take.
Event	none

doorCapabilities[JXFS_SIU_FRONT_TOP]

Type	***JxfsSiuDoorCapabilitiy***
Initial Value	0
Description	Specifies whether the Front Top Door is available, and if so, which states it can take.
Event	none

doorCapabilities[JXFS_SIU_REAR_TOP]

Type	***JxfsSiuDoorCapabilitiy***
Initial Value	0
Description	Specifies whether the Rear Top Door is available, and if so, which states it can take.
Event	none

doorCapabilities[JXFS_SIU_FRONT_BOTTOM]

Type	***JxfsSiuDoorCapabilitiy***
Initial Value	0
Description	Specifies whether the Front Bottom Door is available, and if so, which states it can take.
Event	none

doorCapabilities[JXFS_SIU_REAR_BOTTOM]

Type	***JxfsSiuDoorCapabilitiy***
Initial Value	0
Description	Specifies whether the Rear Bottom Door is available, and if so, which states it can take.
Event	none

indicatorCapabilities[JXFS_SIU_OPENCLOSE]

Type	***JxfsSiuIndicatorCapabilitiy***
Initial Value	0
Description	Specifies whether the Open/Closed Indicator is available.
Event	none

indicatorCapabilities[JXFS_SIU_FASCIALIGHT]

Type	***JxfsSiuIndicatorCapabilitiy***
Initial Value	0
Description	Specifies whether the Fascia Light is available.
Event	none

indicatorCapabilities[JXFS_SIU_AUDIO]

Type	***JxfsSiuIndicatorCapabilitiy***
Initial Value	0
Description	Specifies whether the Audio Indicator device is available.
Event	none

indicatorCapabilities[JXFS_SIU_HEATING]

Type	***JxfsSiuIndicatorCapabilitiy***
Initial Value	0
Description	Specifies whether the internal Heating device is available.
Event	none

indicatorCapabilities[JXFS_SIU_LOGOLIGHT]

Type	***JxfsSiuIndicatorCapabilitiy***
Initial Value	0
Description	Specifies whether the Logo Light is available.
Event	none

auxiliaryCapabilities[JXFS_SIU_VOLUME]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the volume control is available, and if so, the increment/decrement value for the volume control.
Event	none

auxiliaryCapabilities[JXFS_SIU_UPS]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the UPS device is available, and if so, which states it can take.
Event	none

auxiliaryCapabilities[JXFS_SIU_MONITOR]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the possibility to switch the monitor is available.
Event	none

auxiliaryCapabilities[JXFS_SIU_POWEROFF]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the software Poweroff is available.
Event	none

auxiliaryCapabilities[JXFS_SIU_RELAY1]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the first Relay is available.
Event	none

auxiliaryCapabilities[JXFS_SIU_RELAY2]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the second Relay is available.
Event	none

auxiliaryCapabilities[JXFS_SIU_RELAY3]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the third Relay is available.
Event	none

auxiliaryCapabilities[JXFS_SIU_RELAY4]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the fourth Relay is available.
Event	none

auxiliaryCapabilities[JXFS_SIU_ENHANCEDAUDIOCONTROL]

Type	***JxfsSiuAuxiliaryCapabilitiy***
Initial Value	0
Description	Specifies whether the Audio Jack control is available.
Event	none

guidLightCapabilities[JXFS_SIU_CARDUNIT]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the Card Unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_PINPAD]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the PIN pad is available.
Event	none

guidLightCapabilities[JXFS_SIU_NOTESDISPENSER]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the note dispenser unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_COINDISPENSER]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the coin dispenser unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_RECEIPTPRINTER]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the receipt printer unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_PASSBOOKPRINTER]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the passbook printer unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_ENVDEPOSITORY]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the envelope depository unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_CHEQUEUNIT]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the cheque processing unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_BILLACCEPTOR]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the bill acceptor unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_ENVDISPENSER]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the envelope dispenser unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_SCANNER]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the scanner device is available.
Event	none

guidLightCapabilities[JXFS_SIU_COINACCEPTOR]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the coin acceptor unit is available.
Event	none

guidLightCapabilities[JXFS_SIU_DOCUMENTPRINTER]

Type	***JxfsSiuGuidLightCapabilitiy***
Initial Value	0
Description	Specifies whether the Guidance Light Indicator on the document printer is available.
Event	none

9.14 JxfsSiuEnable

This class is used to specify if a port shall be enabled to send events or not.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
enable	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuEnable	enable	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.14.1 Properties

enable (R)

Type ***int***
Initial Value none
Description Specifies if the SIU device shall report a change of the appropriate setting or not or if the current setting shall not be changed.

Specifies the possible values:

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current setting.
JXFS_SIU_ENABLE_EVENT	Report changes of the state.
JXFS_SIU_DISABLE_EVENT	Do not send events if the state changes.

9.15 JxfsSiuEnableEvents

This class contains properties and methods to specify the events to be sent in case of changes of the current conditions of a port.

The implementation of the Properties as arrays allows them to be extended by other ports of the same type (sensors, doors, indicators, etc.), if the implementation requires this. This way it is possible to extend the capabilities in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

Null references as properties of the JxfsSiuEnableEvents class are not allowed. All arrays must be present and all elements of the arrays must exist. The length of the arrays must be at least as long as defined in the initial J/XFS CWA. If more than one object of an array is not represented by a physical port, the references may refer to the same object represented as JXFS_SIU_NO_CHANGE.

9.15.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
sensorEnable[] (R/W)	**JxfsSiuEnable[]**	R/W	
doorEnable[] (R/W)	**JxfsSiuEnable[]**	R/W	
indicatorEnable[] (R/W)	**JxfsSiuEnable[]**	R/W	
auxiliaryEnable[] (R/W)	**JxfsSiuEnable[]**	R/W	
guidlightEnable[] (R/W)	**JxfsSiuEnable[]**	R/W	

Constructor	Parameter	Parameter-Type
JxfsSiuEnableEvents	sensorEnable[] (R/W)	JxfsSiuEnable[]
	doorEnable[] (R/W)	JxfsSiuEnable[]
	indicatorEnable[] (R/W)	JxfsSiuEnable[]
	auxiliaryEnable[] (R/W)	JxfsSiuEnable[]
	guidlightEnable[] (R/W)	JxfsSiuEnable[]

Method	Return	May be used after
set*Property*	*Property*	
get*Property*	*Property*	

Event	May occur after
none	

9.15.2 Properties

sensorEnable[] (R/W)

Type	***JxfsSiuEnable[]***
Initial Value	none
Description	Specifies whether change events of the corresponding sensor ports of the JxfsSiuStatus shall be reported to the application.
Event	none

doorEnable[] (R/W)

Type	***JxfsSiuEnable[]***
Initial Value	none
Description	Specifies whether change events of the corresponding doors of the JxfsSiuStatus shall be reported to the application.
Event	none

indicatorEnable[] (R/W)

Type	***JxfsSiuEnable[]***
Initial Value	none
Description	Specifies whether change events of the corresponding indicator ports of the JxfsSiuStatus shall be reported to the application.
Event	none

auxiliaryEnable[] (R/W)

Type	***JxfsSiuEnable[]***
Initial Value	none
Description	Specifies whether change events of the corresponding auxiliary ports of the JxfsSiuStatus shall be reported to the application.
Event	none

guidlightEnable[] (R/W)

Type	***JxfsSiuEnable[]***
Initial Value	none
Description	Specifies whether change events of the corresponding guidance light ports of the JxfsSiuStatus shall be reported to the application.
Event	none

9.16 JxfsSiuDoorPort

This class specifies if the appropriate port shall be changed.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
state	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuDoorPort	state	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.16.1 Properties

state (R)

Type ***int***
Initial Value none
Description Specifies if the SIU device shall change the state of the specified port. The possible values depend on the type of port.

Specifies the possible values for the Cabinet Doors (JXFS_SIU_CABINET), the Safe Doors (JXFS_SIU_SAFE), the Front Top Door (JXFS_SIU_FRONT_TOP), the Rear Top Door (JXFS_SIU_REAR_TOP), the Front Bottom Door (JXFS_SIU_FRONT_BOTTOM) and the Rear Bottom Door (JXFS_SIU_REAR_BOTTOM):

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_BOLT	Bolt the door(s).
JXFS_SIU_UNBOLT	Unbolt the door(s).

Specifies the possible values for the Vandal Shield (JXFS_SIU_VANDALSHIELD):

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_OPEN	Open the Vandal Shield.
JXFS_SIU_SERVICE	Move the Vandal Shield into service position.
JXFS_SIU_KEYBOARD	Set the Vandal Shield into a position that permits access to the keyboard.
JXFS_SIU_CLOSED	Close the Vandal Shield.

9.17 JxfsSiuIndicatorPort

This class specifies if the appropriate port shall be changed.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
state	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuIndicatorPort	state	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.17.1 Properties

state (R)

Type ***int***
Initial Value none
Description Specifies if the SIU device shall change the state of the specified port. The possible values depend on the type of port.

Specifies the possible values for the Open/Close Indicator (JXFS_SIU_OPENCLOSE):

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_CLOSED	The indicator is changed to show that the terminal is closed for a consumer.
JXFS_SIU_OPEN	The indicator is changed to show that the terminal is open to be used by a consumer.

Specifies the possible values for the Fascia Light (JXFS_SIU_FASCIALIGHT) and the Logo Light (JXFS_SIU_LOGOLIGHT):

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_OFF	The light is turned off.
JXFS_SIU_ON	The light is turned on.

Specifies whether the Audio Indicator shall be turned on or off as one of the following flags of type A and B, or as JXFS_SIU_CONTINUOUS in combination with one of the flags of type B:

Value	Meaning	Type
JXFS_SIU_NO_CHANGE	Do not change the current status of the beeper.	A
JXFS_SIU_OFF	The Audio Indicator is turned off.	A
JXFS_SIU_KEYPRESS	The Audio Indicator sounds a key click signal.	B
JXFS_SIU_EXCLAMATION	The Audio Indicator sounds an exclamation signal.	B
JXFS_SIU_WARNING	The Audio Indicator sounds a warning signal.	B
JXFS_SIU_ERROR	The Audio Indicator sounds an error signal.	B
JXFS_SIU_CRITICAL	The Audio Indicator sounds a critical signal.	B
JXFS_SIU_CONTINUOUS	The Audio Indicator sound is turnd on continuously.	C

Specifies if the internal Heating (JXFS_SIU_HEATING) shall be turned on or off as one of the following values:

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_OFF	The Heating is turned off.
JXFS_SIU_ON	The Heating is turned on.

9.18 JxfsSiuAuxiliaryPort

This class specifies if the appropriate port shall be changed.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
state	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuAuxiliaryPort	state	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.18.1 Properties

state (R)

Type *int*
Initial Value none
Description Specifies if the SIU device shall change the state of the specified auxiliary port. The possible values depend on the type of port.

Specifies the possible values for the volume control (JXFS_SIU_VOLUME):

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current volume.
1, ..., 1000	The volume level. If a value is greater than 1000 is used, the provider will map the value to 1000.

Specifies whether the UPS device (JXFS_SIU_UPS) shall be engaged or disengaged. The UPS should not be engaged when the charge level is low.

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_ENGAGE	Engage the UPS.
JXFS_SIU_DISENGAGE	Disengage the UPS.

Specifies whether the Monitor (JXFS_SIU_MONITOR) shall be switched on or off. Specified as one of the following values:

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_OFF	Switch the Monitor off.
JXFS_SIU_ON	Switch the Monitor on.

Specifies whether the software Poweroff shall be activated. Specified as one of the following values:

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_OFF	Switch the power off.
JXFS_SIU_RESTART	Restart (cold start) the machine. If a cold start is (currently) not available the device service should try to perform at least a reset or warm boot of the machine if possible.

Specifies whether the appropriate relay shall be switched on or off. This applies to the four available relays (JXFS_SIU_RELAY1, JXFS_SIU_RELAY2, JXFS_SIU_RELAY3 and JXFS_SIU_RELAY4). Specified as one of the following values:

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_OFF	Switch the relay off.
JXFS_SIU_ON	Switch the relay on.

Specifies whether the state of the Audio Jack (JXFS_SIU_ENHANCEDAUDIOCONTROL) should be changed or not. Note that this will only be acted upon for hardware environments that return JXFS_SIU_MODE_CONTROLLABLE for the JXFS_SIU_ENHANCEDAUDIOCONTROL capabilities. Specified as one of the following values:

Value	**Meaning**
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_PUBLICAUDIO_ MANUAL	The Audio Jack will be in manual-mode and in the public state (i.e. audio will be played through speakers). Connecting a headset will have no impact, i.e. Output will remain through the speakers & no audio will be directed to the headset.
JXFS_SIU_PUBLICAUDIO_ AUTO	Set the Audio Jack to auto-mode, public state (i.e. audio will be played through speakers). When a headset is connected, the device will go to the private state.
JXFS_SIU_PUBLICAUDIO_ SEMI_AUTO	Set the Audio Jack to semi-auto mode, public state (i.e. audio will be played through speakers). When a headset is connected, the device will go to the private state.
JXFS_SIU_PRIVATEAUDIO_ MANUAL	Set the Audio Jack to manual-mode, private state (i.e. audio will be played only through a connected headset). In private mode, no audio is transmitted through the speakers.
JXFS_SIU_PRIVATEAUDIO_ AUTO	Set the Audio Jack to auto-mode, private state (i.e. audio will be played only through a connected headset). In private mode, no audio is transmitted through the speakers. When a headset is disconnected, the device will go to the public state.
JXFS_SIU_PRIVATEAUDIO_ SEMI_AUTO	Set the Audio Jack to semi-auto mode, private state (i.e. audio will be played only through a connected headset). In private mode, no audio is transmitted through the speakers. When a headset is disconnected, the device will remain in the private state.

9.19 JxfsSiuGuidLightPort

This class specifies if the appropriate port shall be changed.

Summary

Implements : *Serializable* **Extends :** *JxfsType*

Property	Type	Access	Initialized after
state	int	R	

Constructor	Parameter	Parameter-Type
JxfsSiuGuidLightPort	state	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
None	

9.19.1 Properties

state (R)

Type ***int***
Initial Value none
Description Specifies if the SIU device shall change the state of the specified guidance light.

Specifies the possible values for the guidance lights:

Value	Meaning
JXFS_SIU_NO_CHANGE	Do not change the current state.
JXFS_SIU_OFF	The light indicator is turned off.
JXFS_SIU_SLOW_FLASH	The light indicator is set to flash slowly.
JXFS_SIU_MEDIUM_FLASH	The light indicator is blinking with medium frequency.
JXFS_SIU_QUICK_FLASH	The light indicator is set to flash quickly.
JXFS_SIU_CONTINUOUS	The light indicator is turned on continuously (steady).

If the application chooses a value which is not supported by the guidance light, the device service - as it knows the capabilities of its Device - should map the value to the next best state and should not return JXFS_RC_UNSUCCESSFUL.

9.20 JxfsSiuSetPorts

This class contains the functionality to specify which ports have to be changed.

The implementation of the Properties as arrays allows them to be extended by other ports of the same type (sensors, doors, indicators, etc.), if the implementation requires this. This way it is possible to extend this status in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

Null references as properties of the JxfsSiuSetPorts class are not allowed. All arrays must be present and all elements of the arrays must exist. The length of the arrays must be at least as long as defined in the initial J/XFS CWA. If more than one object of an array is not represented by a physical port, the references may refer to the same object represented as JXFS_SIU_NO_CHANGE.

9.20.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
doorPorts[JXFS_SIU_CABINET]	**JxfsSiuDoorPort[]**	R/W	
indicatorPorts[JXFS_SIU_OPENCLOSE]	**JxfsSiuIndicatorPort[]**	R/W	
auxiliaryPorts[JXFS_SIU_VOLUME]	**JxfsSiuAuxiliaryPort[]**	R/W	
guidlightPorts[JXFS_SIU_CARDUNIT]	**JxfsSiuGuidLightPort[]**	R/W	

Constructor	Parameter	Parameter-Type
JxfsSiuSetPorts	doorPorts[JXFS_SIU_CABINET]	JxfsSiuDoorPort[]
	indicatorPorts[JXFS_SIU_OPENCLOSE]	JxfsSiuIndicatorPort[]
	auxiliaryPorts[JXFS_SIU_VOLUME]	JxfsSiuAuxiliaryPort[]
	guidlightPorts[JXFS_SIU_CARDUNIT]	JxfsSiuGuidLightPort[]

Method	Return	May be used after
set*Property*	*Property*	
get*Property*	*Property*	

Event	May occur after
none	

9.20.2 Properties

doorPorts[JXFS_SIU_CABINET]

Type ***JxfsSiuDoorPorts***
Description Specifies whether the Cabinet Doors shall be bolted or unbolted.
Event none

doorPorts[JXFS_SIU_SAFE]

Type ***JxfsSiuDoorPorts***

Description Specifies whether the Safe Doors shall be bolted or unbolted.
Event none

doorPorts [JXFS_SIU_VANDALSHIELD]

Type ***JxfsSiuDoorPorts***
Description Specifies whether the Vandal Shield shall change its position.
Event none

doorPorts [JXFS_SIU_FRONT_TOP]

Type ***JxfsSiuDoorPorts***
Description Specifies whether the Front Top Door shall be bolted or unbolted.
Event none

doorPorts[JXFS_SIU_REAR_TOP]

Type ***JxfsSiuDoorPorts***
Description Specifies whether the Rear Top Door shall be bolted or unbolted.
Event none

doorPorts[JXFS_SIU_FRONT_BOTTOM]

Type ***JxfsSiuDoorPorts***
Description Specifies whether the Front Bottom Door shall be bolted or unbolted.
Event none

doorPorts[JXFS_SIU_REAR_BOTTOM]

Type ***JxfsSiuDoorPorts***
Description Specifies whether the Rear Bottom Door shall be bolted or unbolted.
Event none

indicatorPorts[JXFS_SIU_OPENCLOSE]

Type ***JxfsSiuIndicatorPorts***
Description Specifies whether the Open/Closed Indicator shall show Open or Close to a consumer.
Event none

indicatorPorts[JXFS_SIU_FASCIALIGHT]

Type ***JxfsSiuIndicatorPorts***
Description Specifies whether the Fascia Light shall be turned on or off.
Event none

indicatorPorts[JXFS_SIU_AUDIO]

Type ***JxfsSiuIndicatorPorts***
Description Specifies whether the Audio Indicator shall be turned on or off.
Event none

indicatorPorts[JXFS_SIU_HEATING]

Type ***JxfsSiuIndicatorPorts***
Description Specifies whether the internal heating shall be turned on or off.
Event none

indicatorPorts[JXFS_SIU_LOGOLIGHT]

Type ***JxfsSiuIndicatorPorts***
Description Specifies whether the Logo Light shall be turned on or off.
Event none

auxiliaryPorts[JXFS_SIU_VOLUME]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the value of the volume control shall be changed or not and if it shall be changed then to which level.
Event	none

auxiliaryPorts[JXFS_SIU_UPS]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the Uninterruptable Power Supply device shall be engaged or disengaged.
Event	none

auxiliaryPorts[JXFS_SIU_MONITOR]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the Monitor shall be switched on or off.
Event	none

auxiliaryPorts[JXFS_SIU_POWEROFF]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the software Poweroff shall be activated or not.
Event	None

auxiliaryPorts[JXFS_SIU_RELAY1]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the first Relay shall be switched on or off.
Event	None

auxiliaryPorts[JXFS_SIU_RELAY2]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the second Relay shall be switched on or off.
Event	None

auxiliaryPorts[JXFS_SIU_RELAY3]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the third Relay shall be switched on or off.
Event	None

auxiliaryPorts[JXFS_SIU_RELAY4]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies whether the fourth Relay shall be switched on or off.
Event	None

auxiliaryPorts[JXFS_SIU_ENHANCEDAUDIOCONTROL]

Type	***JxfsSiuAuxiliaryPorts***
Description	Specifies the intended state of the Audio Jack controller.
Event	None

guidlightPorts[JXFS_SIU_CARDUNIT]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the Card Unit (MSD/CCD) shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_PINPAD]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the PIN pad unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_NOTESDISPENSER]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the note dispenser unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_COINDISPENSER]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the coin dispenser unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_RECEIPTPRINTER]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the receipt printer unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_PASSBOOKPRINTER]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the passbook printer unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_ENVDEPOSITORY]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the envelope depository unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_CHEQUEUNIT]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the cheque processing unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_BILLACCEPTOR]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the bill acceptor unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_ENVDISPENSER]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the envelope dispenser unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_SCANNER]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the scanner device shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_COINACCEPTOR]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the coin acceptor unit shall be turned on or off or if it shall flash.
Event	none

guidlightPorts[JXFS_SIU_DOCUMENTPRINTER]

Type	***JxfsSiuGuidLightPorts***
Description	Specifies whether the Guidance Light Indicator on the document printer shall be turned on or off or if it shall flash.
Event	none

9.21 JxfsSiuSetDoor

This class is used to set the status of one of the doors.

9.21.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
doorPort	**JxfsSiuDoorPort**	R	
doorIndex	**int**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetDoor	doorPort	JxfsSiuDoorPort
	doorIndex	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.21.2 Properties

doorPort (R)

Type ***JxfsSiuDoorPort***
Description Specifies the state the door shall be changed to.
Event none

doorIndex (R)

Type ***int***
Initial Value none
Description Specifies the door to be changed.

The following values are examples as the door port array may be extended. Dependant on the value of this property there are different possible values for the doorPort property as the doors have different functionality.

Value	Meaning
JXFS_SIU_CABINET	Bolt/unbolt the Cabinet doors
JXFS_SIU_SAFE	Bolt/unbolt the Safe doors.
JXFS_SIU_VANDALSHIELD	Set position of the Vandal Shield.
JXFS_SIU_FRONT_TOP	Bolt/unbolt the Front Top door.
JXFS_SIU_REAR_TOP	Bolt/unbolt the Rear Top door
JXFS_SIU_FRONT_BOTTOM	Bolt/unbolt the Front Bottom door.
JXFS_SIU_REAR_BOTTOM	Bolt/unbolt the Rear Bottom door.

9.22 JxfsSiuSetIndicator

This class is used to set the status of one of the indicators.

9.22.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
indicatorPort	**JxfsSiuIndicatorPort**	R	
indicatorIndex	**int**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetIndicator	indicatorPort	JxfsSiuIndicatorPort
	indicatorIndex	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
None	

9.22.2 Properties

indicatorPort (R)

Type ***JxfsSiuIndicatorPort***
Description Specifies the state the indicator shall be changed to.
Event None

indicatorIndex (R)

Type ***int***
Initial Value None
Description Specifies the door to be changed.

Value	Meaning
JXFS_SIU_OPENCLOSE	Open/Close indicator.
JXFS_SIU_FASCIALIGHT	Fascia light.
JXFS_SIU_AUDIO	Audio Indicator.
JXFS_SIU_HEATING	Heating device.
JXFS_SIU_LOGOLIGHT	Logo device.

9.23 JxfsSiuSetAuxiliary

This class is used to set the status of one of the auxiliary indicators.

9.23.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
AuxiliaryPort	**JxfsSiuAuxiliaryPort**	R	
auxiliaryIndex	**int**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetAuxiliary	auxiliaryPort	JxfsSiuAuxiliaryPort
	auxiliaryIndex	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
None	

9.23.2 Properties

auxiliaryPort (R)

Type ***JxfsSiuAuxiliaryPort***
Description Specifies the state the auxiliary indicator shall be changed to.
Event None

auxiliaryIndex (R)

Type ***int***
Initial Value None
Description Specifies the auxiliary indicator to be changed.

Value	Meaning
JXFS_SIU_VOLUME	Set the value of the volume control.
JXFS_SIU_UPS	Set the value of the UPS.
JXFS_SIU_MONITOR	Set the value of the Monitor.
JXFS_SIU_POWEROFF	Set the value of the software poweroff.
JXFS_SIU_RELAY1	Set the value of the first relay.
JXFS_SIU_RELAY2	Set the value of the second relay.
JXFS_SIU_RELAY3	Set the value of the third relay.
JXFS_SIU_RELAY4	Set the value of the fourth relay.
JXFS_ENHANCEDAUDIOCONTROL	Set the mode of the Audio Jack control.

9.24 JxfsSiuSetGuidLight

This class is used to set the status of one of the guidance lights.

9.24.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
guidLightPort	**JxfsSiuGuidLightPort**	R	
guidLightIndex	**int**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetGuidLight	guidLightPort	JxfsSiuGuidLightPort
	guidLightIndex	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.24.2 Properties

guidLightPort (R)

Type ***JxfsSiuGuidLightPort***
Description Specifies the state the guidance light shall be changed to.
Event none

guidLightIndex (R)

Type ***int***
Initial Value none
Description Specifies the guidance light to be changed.

Value	Meaning
JXFS_SIU_CARDUNIT	Set the state of the Guidance Light Indicator on the Card Unit.
JXFS_SIU_PINPAD	Set the state of the Guidance Light Indicator on the PINpad unit.
JXFS_SIU_NOTESDISPENSER	Set the state of the Guidance Light Indicator on the note dispenser unit.
JXFS_SIU_COINDISPENSER	Set the state of the Guidance Light Indicator on the coin dispenser unit.
JXFS_SIU_RECEIPTPRINTER	Set the state of the Guidance Light Indicator on the receipt printer unit.
JXFS_SIU_ PASSBOOKPRINTER	Set the state of the Guidance Light Indicator on the passbook printer unit.
JXFS_SIU_ENVDEPOSITORY	Set the state of the Guidance Light Indicator on the envelope depository unit.
JXFS_SIU_CHEQUEUNIT	Set the state of the Guidance Light Indicator on the cheque processing unit.
JXFS_SIU_BILLACCEPTOR	Set the state of the Guidance Light Indicator on the bill acceptor unit.
JXFS_SIU_ENVDISPENSER	Set the state of the Guidance Light Indicator on the envelope dispenser unit.

JXFS_SIU_SCANNER	Set the state of the Guidance Light Indicator on the scanner device.
JXFS_SIU_COINACCEPTOR	Set the state of the Guidance Light Indicator on the coin acceptor unit.
JXFS_SIU_ DOCUMENTPRINTER	Set the state of the Guidance Light Indicator on the document printer.

9.25 JxfsSiuPortChangeStatus

This class is used to identify the port that has changed and the value the port has changed to. The kind of port (sensors, doors, indicator, auxiliaries, guidance lights) can be identified by the type of port.

A JxfsSiuPortChangeStatus object will be referenced by the details property of a status changed event with the status code JXFS_S_SIU_PORT_STATUS. This event will be received only by device controls that enabled the specific port for supervision.

The application may identify the specific port by first evaluating the sub class of the port property and then analysing the appropriate index value.

If a port is changed by calling a method, this will also generate the appropriate status events to all registrated listeners about the changed port.

9.25.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
port	**JxfsSiuPortStatus**	R	
index	**int**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuPortChangeStatus	port	**JxfsSiuPortStatus**
	index	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.25.2 Properties

port (R)

Type ***JxfsSiuPortStatus***
Description Specifies the state the port has changed to.
Event none

index (R)

Type ***int***
Description Specifies the index of the port that changed its state.

Value	Meaning
JXFS_SIU_OPERATORSWITCH	The Operator Switch has changed its state.
JXFS_SIU_TAMPER	The Tamper Sensor has changed its state.
JXFS_SIU_INTTAMPER	The internal Tamper Sensor has changed its state.
JXFS_SIU_SEISMIC	The Seismic Sensor has changed its state.

JXFS_SIU_HEAT	The Heat Sensor has changed its state.
JXFS_SIU_PROXIMITY	The proximity Sensor has changed its state.
JXFS_SIU_AMBLIGHT	The Ambient Light Sensor has changed its state.
JXFS_SIU_INPUT1	The first input contact has changed its state.
JXFS_SIU_INPUT2	The second input contact has changed its state.
JXFS_SIU_INPUT3	The third input contact has changed its state.
JXFS_SIU_INPUT4	The fourth input contact has changed its state.
JXFS_SIU_VENTILATOR	The ventilator has changed its state.
JXFS_SIU_BOOTSWITCH	The Boot Switch has changed its state.
JXFS_SIU_ ENHANCEDAUDIO	The Audio Jack has changed its state – a headset has been plugged-in or removed.
JXFS_SIU_CABINET	The Cabinet doors have changed their state.
JXFS_SIU_SAFE	The Safe doors have changed their state.
JXFS_SIU_VANDALSHIELD	The Vandal Shield has changed its position.
JXFS_SIU_FRONT_TOP	The Front Top door has changed its state.
JXFS_SIU_REAR_TOP	The Rear Top door has changed its state.
JXFS_SIU_FRONT_BOTTOM	The Front Bottom door has changed its state.
JXFS_SIU_REAR_BOTTOM	The Rear Bottom door has changed its state.
JXFS_SIU_OPENCLOSE	The Open/Close indicator has changed its state.
JXFS_SIU_FASCIALIGHT	The Fascia light has changed its state.
JXFS_SIU_AUDIO	The Audio Indicator has changed its state.
JXFS_SIU_HEATING	The Heating device has changed its state.
JXFS_SIU_LOGOLIGHT	The Logo light has changed its state.
JXFS_SIU_VOLUME	The volume device control has changed its state.
JXFS_SIU_UPS	The UPS device state has changed.
JXFS_SIU_MONITOR	The Monitor state has changed.
JXFS_SIU_POWEROFF	The software poweroff state has changed.
JXFS_SIU_RELAY1	The state of the first relay has changed.
JXFS_SIU_RELAY2	The state of the second relay has changed.
JXFS_SIU_RELAY3	The state of the third relay has changed.
JXFS_SIU_RELAY4	The state of the fourth relay has changed.

JXFS_SIU_ ENHANCEDAUDIOCONTROL	The mode of the Audio Jack Control has changed.
JXFS_SIU_CARDUNIT	The state of the Guidance Light Indicator on the Card Unit has changed..
JXFS_SIU_PINPAD	The state of the Guidance Light Indicator on the PINpad unit has changed.
JXFS_SIU_NOTESDISPENSER	The state of the Guidance Light Indicator on the note dispenser unit has changed.
JXFS_SIU_COINDISPENSER	The state of the Guidance Light Indicator on the coin dispenser unit has changed.
JXFS_SIU_RECEIPTPRINTER	The state of the Guidance Light Indicator on the receipt printer unit has changed.
JXFS_SIU_ PASSBOOKPRINTER	The state of the Guidance Light Indicator on the passbook printer unit has changed.
JXFS_SIU_ENVDEPOSITORY	The state of the Guidance Light Indicator on the envelope depository unit has changed.
JXFS_SIU_CHEQUEUNIT	The state of the Guidance Light Indicator on the cheque processing unit has changed.
JXFS_SIU_BILLACCEPTOR	The state of the Guidance Light Indicator on the bill acceptor unit has changed.
JXFS_SIU_ENVDISPENSER	The state of the Guidance Light Indicator on the envelope dispenser unit has changed.
JXFS_SIU_SCANNERR	The state of the Guidance Light Indicator on the scanner device has changed.
JXFS_SIU_COINACCEPTOR	The state of the Guidance Light Indicator on the coin acceptor unit has changed.
JXFS_SIU_ DOCUMENTPRINTER	The state of the Guidance Light Indicator on the document printer has changed.

9.26 JxfsSiuPortError

This class is used to identify the origin of an error when working with the ports. The kind of port (sensors, doors, indicator, auxiliaries, guidance lights) can be identified by the type of the port property.

As a SIU device must not consist of only one hardware device, but may be build of several hardware devices like electronics connected over RS232, relay cards, etc, a problem with one of those subdevices must not lead to a hardware error state of the whole SIU device service. In the case that only one or more ports are malfunctioning, the SIU device status does not change to hardware error, but for every malfunctioning port a port error event will be sent to all registrated listeners that enabled the receiving of events for that port.

A port error event will be sent by the device service only once to each listener as long as the state of the port does not change or an application registeres/enables (again) events for this port. Another event will be sent, if the port is working again properly.

So, if an application wants to alter a port and this does not work, it receives two events, the OC event indicating that the operation failed and a status event with the value JXFS_S_SIU_PORT_ERROR and a JxfsSiuPortError object as details.

9.26.1 Summary

Implements : *Serializable* **Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
port	**JxfsSiuPortStatus**	R	
index	**int**	R	
portError	**int**	R	

Constructor	Parameter	Parameter-Type
JxfsSiuPortError	port	JxfsSiuPortStatus
	index	int
	portError	int

Method	Return	May be used after
get*Property*	*Property*	

Event	May occur after
none	

9.26.2 Properties

port (R)

Type ***JxfsSiuPortStatus***
Description Specifies the new state of the port.
Event none

index (R)

Type ***int***
Initial Value none
Description Specifies the index of the port that has changed its state.

Value	Meaning
JXFS_SIU_OPERATORSWITCH	The Operator Switch has changed its state.

JXFS_SIU_TAMPER	The Tamper Sensor has changed its state.
JXFS_SIU_INTTAMPER	The internal Tamper Sensor has changed its state.
JXFS_SIU_SEISMIC	The Seismic Sensor has changed its state.
JXFS_SIU_HEAT	The Heat Sensor has changed its state.
JXFS_SIU_PROXIMITY	The Proximity Sensor has changed its state.
JXFS_SIU_AMBLIGHT	The Ambient Light Sensor has changed its state.
JXFS_SIU_INPUT1	The first input contact has changed its state.
JXFS_SIU_INPUT2	The second input contact has changed its state.
JXFS_SIU_INPUT3	The third input contact has changed its state.
JXFS_SIU_INPUT4	The fourth input contact has changed its state.
JXFS_SIU_VENTILATOR	The ventilator has changed its state.
JXFS_SIU_BOOTSWITCH	The Boot Switch has changed its state.
JXFS_SIU_ ENHANCEDAUDIO	The Audio Jack has detected an error
JXFS_SIU_CABINET	The Cabinet doors have changed their state.
JXFS_SIU_SAFE	The Safe doors have changed their state.
JXFS_SIU_VANDALSHIELD	The Vandal Shield has changed its position.
JXFS_SIU_FRONT_TOP	The Front Top door has changed its state.
JXFS_SIU_REAR_TOP	The Rear Top door has changed its state.
JXFS_SIU_FRONT_BOTTOM	The Front Bottom door has changed its state.
JXFS_SIU_REAR_BOTTOM	The Rear Bottom door has changed its state.
JXFS_SIU_OPENCLOSE	The Open/Close indicator has changed its state.
JXFS_SIU_FASCIALIGHT	The Fascia Light has changed its state.
JXFS_SIU_AUDIO	The Audio Indicator has changed its state.
JXFS_SIU_HEATING	The Heating device has changed its state.
JXFS_SIU_LOGOLIGHT	The Logo light has changed its state.
JXFS_SIU_VOLUME	The volume device control has changed its state.
JXFS_SIU_UPS	The UPS device state has changed.
JXFS_SIU_MONITOR	The Monitor state has changed.
JXFS_SIU_POWEROFF	The software poweroff state has changed.
JXFS_SIU_RELAY1	The state of the first relay has changed.

JXFS_SIU_RELAY2	The state of the second relay has changed.
JXFS_SIU_RELAY3	The state of the third relay has changed.
JXFS_SIU_RELAY4	The state of the fourth relay has changed.
JXFS_ENHANCEDAUDIOCONTROL	The Audio Jack Control has detected an error.
JXFS_SIU_CARDUNIT	The state of the Guidance Light Indicator on the Card Unit has changed.
JXFS_SIU_PINPAD	The state of the Guidance Light Indicator on the PINpad unit has changed.
JXFS_SIU_NOTESDISPENSER	The state of the Guidance Light Indicator on the note dispenser unit has changed..
JXFS_SIU_COINDISPENSER	The state of the Guidance Light Indicator on the coin dispenser unit has changed.
JXFS_SIU_RECEIPTPRINTER	The state of the Guidance Light Indicator on the receipt printer unit has changed.
JXFS_SIU_PASSBOOKPRINTER	The state of the Guidance Light Indicator on the passbook printer unit has changed.
JXFS_SIU_ENVDEPOSITORY	The state of the Guidance Light Indicator on the envelope depository unit has changed.
JXFS_SIU_CHEQUEUNIT	The state of the Guidance Light Indicator on the cheque processing unit has changed.
JXFS_SIU_BILLACCEPTOR	The state of the Guidance Light Indicator on the bill acceptor unit has changed.
JXFS_SIU_ENVDISPENSER	The state of the Guidance Light Indicator on the envelope dispenser unit has changed.
JXFS_SIU_SCANNER	The state of the Guidance Light Indicator on the scanner unit has changed.
JXFS_SIU_COINACCEPTOR	The state of the Guidance Light Indicator on the coin acceptor unit has changed.
JXFS_SIU_ DOCUMENTPRINTER	The state of the Guidance Light Indicator on the document printer has changed.

portError (R)

Type *int*

Description Specifies the error of the port indicated by port and index by one of the following flags:

JXFS_E_SIU_INVALID_PORT	An attempt to enable or disable events to a port was invalid because the port does not exist.
JXFS_E_SIU_SYNTAX	Syntax error in the input parameters. Eg.g. an attempt to both enable and disable events to the same port was made.

	JXFS_E_SIU_PORT_ERROR	A hardware error occurred while executing a command.
	JXFS_E_SIU_PORT_OK	The specific port is working again after it had been in an errorneous state (JXFS_E_SIU_PORT_ERROR).
Event	none	

10 Enum Classes

10.1 JxfsSIUStatusSelectorEnum

This enumeration class is used for the base getStatus(java.util.List) method.

Extends	Implements
JxfsStatusSelectorEnum	

Field	Returned Type	Description
siuStatus	*JxfsSiuStatus*	Status of the sensors and indicators unit device.

11 Codes

11.1 Error Codes

Value	Meaning
JXFS_E_SIU_INVALID_PORT	An attempt was made to use a port that does not exist.
JXFS_E_SIU_SYNTAX	The command was invoked with incorrect input data. E. g. an attempt was made to both enable and disable events to the same port.
JXFS_E_SIU_PORT_ERROR	An error occurred when accessing a port.

11.2 Status Codes

Value	Meaning
JXFS_S_SIU_PORT_STATUS	The state of the specified port has changed. The port that changed and the new state of the port are delivered as a *JxfsSiuPortChangeStatus* object. If several ports change, each will be reported as an own status event.
JXFS_S_SIU_PORT_ERROR	There was an error when accessing a port. Specific information about this error is contained in a JxfsSiuPortError object. A status event with this status code is also sent if an errorneous port is working again.

11.3 Index Codes

The specific ports that are defined by their index values support the following capability values:

Index Value	JXFS_SIU_NOT_AVAILABLE	JXFS_SIU_RUN	JXFS_SIU_MAINTENANCE	JXFS_SIU_SUPERVISOR	JXFS_SIU_AVAILABLE	JXFS_SIU_LOCKED	JXFS_SIU_BOLTED	JXFS_SIU_CLOSED	JXFS_:SIU_OPEN	JXFS_SIU_SERVICE	JXFS_SIU_KEYBOARD	JXFS_SIU_AJAR	JXFS_SIU_JAMMED	numeric value (1-1000)	JXFS_SIU_LOW	JXFS_SIU_ENGAGED	JXFS_SIU_POWERING	JXFS_SIU_RECOVERED
JXFS_SIU_OPERATORSWITCH	X	X	X	X														
JXFS_SIU_TAMPER	X				X													
JXFS_SIU_INTTAMPER	X				X													
JXFS_SIU_SEISMIC	X				X													
JXFS_SIU_HEAT	X				X													
JXFS_SIU_PROXIMITY	X				X													
JXFS_SIU_AMBLIGHT	X				X													
JXFS_SIU_INPUT1	X				X													
JXFS_SIU_INPUT2	X				X													
JXFS_SIU_INPUT3	X				X													
JXFS_SIU_INPUT4	X				X													
JXFS_SIU_VENTILATOR	X				X													
JXFS_SIU_BOOTSWITCH	X				X													
JXFS_SIU_ENHANCEDAUDIO	X																	
JXFS_SIU_CABINET	X					X	X	X	X									
JXFS_SIU_SAFE	X					X	X	X	X									
JXFS_SIU_VANDALSHIELD	X					X		X	X	X	X	X	X					
JXFS_SIU_FRONT_TOP	X						X	X	X									
JXFS_SIU_REAR_TOP	X						X	X	X									
JXFS_SIU_FRONT_BOTTOM	X						X	X	X									
JXFS_SIU_REAR_BOTTOM	X						X	X	X									
JXFS_SIU_OPENCLOSE	X				X													
JXFS_SIU_FASCIALIGHT	X				X													
JXFS_SIU_AUDIO	X				X													
JXFS_SIU_HEATING	X				X													
JXFS_SIU_LOGOLIGHT	X				X													
JXFS_SIU_VOLUME	X													X				
JXFS_SIU_UPS	X				X										X	X	X	X
JXFS_SIU_MONITOR	X				X													
JXFS_SIU_POWEROFF	X				X													
JXFS_SIU_RELAY1	X				X													
JXFS_SIU_RELAY2	X				X													
JXFS_SIU_RELAY3	X				X													
JXFS_SIU_RELAY4	X				X													
JXFS_SIU_ENHANCEDAUDIOCONTROL	X																	
JXFS_SIU_CARDUNIT	X				X													
JXFS_SIU_PINPAD	X				X													
JXFS_SIU_NOTESDISPENSER	X				X													
JXFS_SIU_COINDISPENSER	X				X													
JXFS_SIU_RECEIPTPRINTER	X				X													
JXFS_SIU_PASSBOOKPRINTER	X				X													
JXFS_SIU_ENVDEPOSITORY	X				X													
JXFS_SIU_CHEQUEUNIT	X				X													
JXFS_SIU_BILLACCEPTOR	X				X													

Index Value	JXFS_SIU_NOT_AVAILABLE	JXFS_SIU_RUN	JXFS_SIU_MAINTENANCE	JXFS_SIU_SUPERVISOR	JXFS_SIU_AVAILABLE	JXFS_SIU_LOCKED	JXFS_SIU_BOLTED	JXFS_SIU_CLOSED	JXFS_:SIU_OPEN	JXFS_SIU_SERVICE	JXFS_SIU_KEYBOARD	JXFS_SIU_AJAR	JXFS_SIU_JAMMED	numeric value (1-1000)	JXFS_SIU_LOW	JXFS_SIU_ENGAGED	JXFS_SIU_POWERING	JXFS_SIU_RECOVERED
JXFS_SIU_ENVDISPENSER	X				X													
JXFS_SIU_SCANNER	X				X													
JXFS_SIU_COINACCEPTOR	X				X													
JXFS_SIU_DOCUMENTPRINTER	X				X													

Index Value	JXFS_SIU_MANUAL	JXFS_SIU_AUTO	JXFS_SIU_SEMI_AUTO	JXFS_SIU_HEADSET_DETECTION	JXFS_SIU_MODE_CONTROLLABLE
JXFS_SIU_OPERATORSWITCH					
JXFS_SIU_TAMPER					
JXFS_SIU_INTTAMPER					
JXFS_SIU_SEISMIC					
JXFS_SIU_HEAT					
JXFS_SIU_PROXIMITY					
JXFS_SIU_AMBLIGHT					
JXFS_SIU_INPUT1					
JXFS_SIU_INPUT2					
JXFS_SIU_INPUT3					
JXFS_SIU_INPUT4					
JXFS_SIU_VENTILATOR					
JXFS_SIU_BOOTSWITCH					
JXFS_SIU_ENHANCEDAUDIO	X	X	X		
JXFS_SIU_CABINET					
JXFS_SIU_SAFE					
JXFS_SIU_VANDALSHIELD					
JXFS_SIU_FRONT_TOP					
JXFS_SIU_REAR_TOP					
JXFS_SIU_FRONT_BOTTOM					
JXFS_SIU_REAR_BOTTOM					
JXFS_SIU_OPENCLOSE					
JXFS_SIU_FASCIALIGHT					
JXFS_SIU_AUDIO					
JXFS_SIU_HEATING					
JXFS_SIU_LOGOLIGHT					
JXFS_SIU_VOLUME					
JXFS_SIU_UPS					
JXFS_SIU_MONITOR					
JXFS_SIU_POWEROFF					
JXFS_SIU_RELAY1					
JXFS_SIU_RELAY2					
JXFS_SIU_RELAY3					
JXFS_SIU_RELAY4					
JXFS_SIU_ENHANCEDAUDIOCONTROL				X	X
JXFS_SIU_CARDUNIT					
JXFS_SIU_PINPAD					
JXFS_SIU_NOTESDISPENSER					
JXFS_SIU_COINDISPENSER					
JXFS_SIU_RECEIPTPRINTER					
JXFS_SIU_PASSBOOKPRINTER					
JXFS_SIU_ENVDEPOSITORY					
JXFS_SIU_CHEQUEUNIT					
JXFS_SIU_BILLACCEPTOR					
JXFS_SIU_ENVDISPENSER					
JXFS_SIU_SCANNER					
JXFS_SIU_COINACCEPTOR					
JXFS_SIU_DOCUMENTPRINTER					

The specific ports that are defined by their index values support the following status values that are provided to the application from the DC. This list does not define all possible values for the ports that may be set:

Index Value	JXFS_SIU_NOT_AVAILABLE	JXFS_SIU_RUN	JXFS_SIU_MAINTENANCE	JXFS_SIU_SUPERVISOR	JXFS_SIU_OFF	JXFS_SIU_ON	JXFS_SIU_NOT_PRESENT	JXFS_SIU_PRESENT	JXFS_SIU_VERY_LIGHT	JXFS_SIU_LIGHT	JXFS_SIU_MEDIUM_LIGHT	JXFS_SIU_DARK	JXFS_SIU_VERY_DARK	JXFS_SIU_HWERROR
JXFS_SIU_OPERATORSWITCH	X	X	X	X										
JXFS_SIU_TAMPER	X				X	X								
JXFS_SIU_INTTAMPER	X				X	X								
JXFS_SIU_SEISMIC	X				X	X								
JXFS_SIU_HEAT	X				X	X								
JXFS_SIU_PROXIMITY	X						X	X						
JXFS_SIU_AMBLIGHT	X								X	X	X	X	X	
JXFS_SIU_INPUT1	X				X	X								
JXFS_SIU_INPUT2	X				X	X								
JXFS_SIU_INPUT3	X				X	X								
JXFS_SIU_INPUT4	X				X	X								
JXFS_SIU_VENTILATOR	X					X								X
JXFS_SIU_BOOTSWITCH	X				X	X								
JXFS_SIU_ENHANCEDAUDIO	X						X	X						
JXFS_SIU_CABINET	X													
JXFS_SIU_SAFE	X													
JXFS_SIU_VANDALSHIELD	X													
JXFS_SIU_FRONT_TOP	X													
JXFS_SIU_REAR_TOP	X													
JXFS_SIU_FRONT_BOTTOM	X													
JXFS_SIU_REAR_BOTTOM	X													
JXFS_SIU_OPENCLOSE	X													
JXFS_SIU_FASCIALIGHT	X				X	X								
JXFS_SIU_AUDIO	X													
JXFS_SIU_HEATING	X				X	X								
JXFS_SIU_LOGOLIGHT	X				X	X								
JXFS_SIU_VOLUME	X													
JXFS_SIU_UPS	X													
JXFS_SIU_MONITOR	X				X	X								
JXFS_SIU_POWEROFF	X													
JXFS_SIU_RELAY1	X				X	X								
JXFS_SIU_RELAY2	X				X	X								
JXFS_SIU_RELAY3	X				X	X								
JXFS_SIU_RELAY4	X				X	X								
JXFS_SIU_ENHANCEDAUDIOCONTROL	X													
JXFS_SIU_CARDUNIT	X				X									
JXFS_SIU_PINPAD	X				X									
JXFS_SIU_NOTESDISPENSER	X				X									
JXFS_SIU_COINDISPENSER	X				X									
JXFS_SIU_RECEIPTPRINTER	X				X									
JXFS_SIU_PASSBOOKPRINTER	X				X									
JXFS_SIU_ENVDEPOSITORY	X				X									
JXFS_SIU_CHEQUEUNIT	X				X									
JXFS_SIU_BILLACCEPTOR	X				X									
JXFS_SIU_ENVDISPENSER	X				X									
JXFS_SIU_SCANNER	X				X									
JXFS_SIU_COINACCEPTOR	X				X									
JXFS_SIU_DOCUMENTPRINTER	X				X									

Index Value	JXFS_SIU_LOCKED	JXFS_SIU_BOLTED	JXFS_SIU_CLOSED	JXFS_SIU_OPEN	JXFS_SIU_SERVICE	JXFS_SIU_KEYBOARD	JXFS_SIU_AJAR	JXFS_SIU_JAMMED	JXFS_SIU_KEYPRESS	JXFS_SIU_EXCLAMATION	JXFS_SIU_WARNING	JXFS_SIU_ERROR	JXFS_SIU_CRITICAL	JXFS_SIU_CONTINUOUS	numeric value (1-1000)
JXFS_SIU_OPERATORSWITCH															
JXFS_SIU_TAMPER															
JXFS_SIU_INTTAMPER															
JXFS_SIU_SEISMIC															
JXFS_SIU_HEAT															
JXFS_SIU_PROXIMITY															
JXFS_SIU_AMBLIGHT															
JXFS_SIU_INPUT1															
JXFS_SIU_INPUT2															
JXFS_SIU_INPUT3															
JXFS_SIU_INPUT4															
JXFS_SIU_VENTILATOR															
JXFS_SIU_BOOTSWITCH															
JXFS_SIU_ENHANCEDAUDIO															
JXFS_SIU_CABINET	X	X	X	X											
JXFS_SIU_SAFE	X	X	X	X											
JXFS_SIU_VANDALSHIELD	X		X	X	X	X	X	X							
JXFS_SIU_FRONT_TOP		X	X	X											
JXFS_SIU_REAR_TOP		X	X	X											
JXFS_SIU_FRONT_BOTTOM		X	X	X											
JXFS_SIU_REAR_BOTTOM		X	X	X											
JXFS_SIU_OPENCLOSE			X	X											
JXFS_SIU_FASCIALIGHT															
JXFS_SIU_AUDIO									X	X	X	X	X	X	
JXFS_SIU_HEATING															
JXFS_SIU_LOGOLIGHT															
JXFS_SIU_VOLUME															X
JXFS_SIU_UPS															
JXFS_SIU_MONITOR															
JXFS_SIU_POWEROFF															
JXFS_SIU_RELAY1															
JXFS_SIU_RELAY2															
JXFS_SIU_RELAY3															
JXFS_SIU_RELAY4															
JXFS_SIU_ENHANCEDAUDIOCONTROL															
JXFS_SIU_CARDUNIT															
JXFS_SIU_PINPAD															
JXFS_SIU_NOTESDISPENSER															
JXFS_SIU_COINDISPENSER															
JXFS_SIU_RECEIPTPRINTER															
JXFS_SIU_PASSBOOKPRINTER															
JXFS_SIU_ENVDEPOSITORY															
JXFS_SIU_CHEQUEUNIT															
JXFS_SIU_BILLACCEPTOR															
JXFS_SIU_ENVDISPENSER															
JXFS_SIU_SCANNER															
JXFS_SIU_COINACCEPTOR															
JXFS_SIU_DOCUMENTPRINTER															

Index Value	JXFS_SIU_AVAILABLE	JXFS_SIU_LOW	JXFS_SIU_ENGAGED	JXFS_SIU_POWERING	JXFS_SIU_RECOVERED	JXFS_SIU_SLOW_FLASH	JXFS_SIU_MEDIUM_FLASH	JXFS_SIU_QUICK_FLASH	JXFS_SIU_CONTINUOUS	JXFS_SIU_PUBLICAUDIO_MANUAL	JXFS_SIU_PUBLICAUDIO_AUTO	JXFS_SIU_PUBLICAUDIO_SEMI_AUTO	JXFS_SIU_PRIVATEAUDIO_MANUAL	JXFS_SIU_PRIVATEAUDIO_AUTO	JXFS_SIU_PRIVATEAUDIO_SEMI_AUTO
JXFS_SIU_OPERATORSWITCH															
JXFS_SIU_TAMPER															
JXFS_SIU_INTTAMPER															
JXFS_SIU_SEISMIC															
JXFS_SIU_HEAT															
JXFS_SIU_PROXIMITY															
JXFS_SIU_AMBLIGHT															
JXFS_SIU_INPUT1															
JXFS_SIU_INPUT2															
JXFS_SIU_INPUT3															
JXFS_SIU_INPUT4															
JXFS_SIU_VENTILATOR															
JXFS_SIU_BOOTSWITCH															
JXFS_SIU_ENHANCEDAUDIO															
JXFS_SIU_CABINET															
JXFS_SIU_SAFE															
JXFS_SIU_VANDALSHIELD															
JXFS_SIU_FRONT_TOP															
JXFS_SIU_REAR_TOP															
JXFS_SIU_FRONT_BOTTOM															
JXFS_SIU_REAR_BOTTOM															
JXFS_SIU_OPENCLOSE															
JXFS_SIU_FASCIALIGHT															
JXFS_SIU_AUDIO															
JXFS_SIU_HEATING															
JXFS_SIU_LOGOLIGHT															
JXFS_SIU_VOLUME															
JXFS_SIU_UPS	X	X	X	X	X										
JXFS_SIU_MONITOR															
JXFS_SIU_POWEROFF	X														
JXFS_SIU_RELAY1															
JXFS_SIU_RELAY2															
JXFS_SIU_RELAY3															
JXFS_SIU_RELAY4															
JXFS_SIU_ENHANCEDAUDIOCONTROL										X	X	X	X	X	X
JXFS_SIU_CARDUNIT						X	X	X	X						
JXFS_SIU_PINPAD						X	X	X	X						
JXFS_SIU_NOTESDISPENSER						X	X	X	X						
JXFS_SIU_COINDISPENSER						X	X	X	X						
JXFS_SIU_RECEIPTPRINTER						X	X	X	X						
JXFS_SIU_PASSBOOKPRINTER						X	X	X	X						
JXFS_SIU_ENVDEPOSITORY						X	X	X	X						
JXFS_SIU_CHEQUEUNIT						X	X	X	X						
JXFS_SIU_BILLACCEPTOR						X	X	X	X						
JXFS_SIU_ENVDISPENSER						X	X	X	X						
JXFS_SIU_SCANNER						X	X	X	X						
JXFS_SIU_COINACCEPTOR						X	X	X	X						
JXFS_SIU_DOCUMENTPRINTER						X	X	X	X						

11.4 Code Values

Code	Value
JXFS_SIU_OFFSET	10000
JXFS_E_SIU_INVALID_PORT	JXFS_SIU_OFFSET + 1
JXFS_E_SIU_SYNTAX	JXFS_SIU_OFFSET + 2
JXFS_E_SIU_PORT_ERROR	JXFS_SIU_OFFSET + 3
JXFS_E_SIU_PORT_OK	JXFS_SIU_OFFSET + 4
JXFS_SIU_NO_CHANGE	JXFS_SIU_OFFSET + 5
JXFS_SIU_ENABLE_EVENT	JXFS_SIU_OFFSET + 6
JXFS_SIU_DISABLE_EVENT	JXFS_SIU_OFFSET + 7
JXFS_SIU_BOLT	JXFS_SIU_OFFSET + 8
JXFS_SIU_UNBOLT	JXFS_SIU_OFFSET + 9
JXFS_SIU_ENGAGE	JXFS_SIU_OFFSET + 10
JXFS_SIU_DISENGAGE	JXFS_SIU_OFFSET + 11
JXFS_SIU_RESTART	JXFS_SIU_OFFSET + 12
JXFS_S_SIU_PORT_STATUS	JXFS_SIU_OFFSET + 5
JXFS_S_SIU_PORT_ERROR	JXFS_SIU_OFFSET + 6
JXFS_O_SIU_ENABLE_EVENTS	JXFS_SIU_OFFSET + 7
JXFS_O_SIU_SET_PORT	JXFS_SIU_OFFSET + 8
JXFS_SIU_NOT_AVAILABLE	0x80000000
JXFS_SIU_RUN	0x00000002
JXFS_SIU_MAINTENANCE	0x00000004
JXFS_SIU_SUPERVISOR	0x00000008
JXFS_SIU_AVAILABLE	0x00000002
JXFS_SIU_LOCKED	0x00000002
JXFS_SIU_BOLTED	0x00000004
JXFS_SIU_CLOSED	0x00000008
JXFS_SIU_OPEN	0x00000010
JXFS_SIU_SERVICE	0x00000020
JXFS_SIU_KEYBOARD	0x00000040
JXFS_SIU_AJAR	0x00000080
JXFS_SIU_JAMMED	0x00000100
JXFS_SIU_LOW	0x00000004
JXFS_SIU_ENGAGED	0x00000008
JXFS_SIU_POWERING	0x00000010
JXFS_SIU_RECOVERED	0x00000020
JXFS_SIU_OFF	0x00000040
JXFS_SIU_ON	0x00000080
JXFS_SIU_NOT_PRESENT	0x00000100
JXFS_SIU_PRESENT	0x00000200
JXFS_SIU_VERY_LIGHT	0x00000400
JXFS_SIU_LIGHT	0x00000800
JXFS_SIU_MEDIUM_LIGHT	0x00001000
JXFS_SIU_DARK	0x00002000
JXFS_SIU_VERY_DARK	0x00004000
JXFS_SIU_HWERROR	0x00008000
JXFS_SIU_KEYPRESS	0x00010000
JXFS_SIU_EXCLAMATION	0x00020000
JXFS_SIU_WARNING	0x00040000
JXFS_SIU_ERROR	0x00080000
JXFS_SIU_CRITICAL	0x00100000
JXFS_SIU_CONTINUOUS	0x00200000
JXFS_SIU_SLOW_FLASH	0x00400000
JXFS_SIU_MEDIUM_FLASH	0x00800000
JXFS_SIU_QUICK_FLASH	0x01000000

Code	Value
JXFS_SIU_PUBLICAUDIO_MANUAL	0x00000001
JXFS_SIU_PUBLICAUDIO_AUTO	0x00000002
JXFS_SIU_PUBLICAUDIO_SEMI_AUTO	0x00000004
JXFS_SIU_PRIVATEAUDIO_MANUAL	0x00000008
JXFS_SIU_PRIVATEAUDIO_AUTO	0x00000010
JXFS_SIU_PRIVATEAUDIO_SEMI_AUTO	0x00000020
JXFS_SIU_MANUAL	0x00000040
JXFS_SIU_AUTO	0x00000080
JXFS_SIU_SEMI_AUTO	0x00000100
JXFS_SIU_OPERATORSWITCH	0x00000000
JXFS_SIU_TAMPER	0x00000001
JXFS_SIU_INTTAMPER	0x00000002
JXFS_SIU_SEISMIC	0x00000003
JXFS_SIU_HEAT	0x00000004
JXFS_SIU_PROXIMITY	0x00000005
JXFS_SIU_AMBLIGHT	0x00000006
JXFS_SIU_INPUT1	0x00000007
JXFS_SIU_INPUT2	0x00000008
JXFS_SIU_INPUT3	0x00000009
JXFS_SIU_INPUT4	0x0000000A
JXFS_SIU_VENTILATOR	0x0000000B
JXFS_SIU_BOOTSWITCH	0x0000000C
JXFS_SIU_ENHANCEDAUDIO	0x0000000D
JXFS_SIU_CABINET	0x00000000
JXFS_SIU_SAFE	0x00000001
JXFS_SIU_VANDALSHIELD	0x00000002
JXFS_SIU_FRONT_TOP	0x00000003
JXFS_SIU_REAR_TOP	0x00000004
JXFS_SIU_FRONT_BOTTOM	0x00000005
JXFS_SIU_REAR_BOTTOM	0x00000006
JXFS_SIU_OPENCLOSE	0x00000000
JXFS_SIU_FASCIALIGHT	0x00000001
JXFS_SIU_AUDIO	0x00000002
JXFS_SIU_HEATING	0x00000003
JXFS_SIU_LOGOLIGHT	0x00000004
JXFS_SIU_VOLUME	0x00000005
JXFS_SIU_UPS	0x00000006
JXFS_SIU_MONITOR	0x00000007
JXFS_SIU_POWEROFF	0x00000008
JXFS_SIU_RELAY1	0x00000009
JXFS_SIU_RELAY2	0x0000000A
JXFS_SIU_RELAY3	0x0000000B
JXFS_SIU_RELAY4	0x0000000C
JXFS_SIU_ENHANCEDAUDIOCONTROL	0x0000000D
JXFS_SIU_CARDUNIT	0x00000000
JXFS_SIU_PINPAD	0x00000001
JXFS_SIU_NOTESDISPENSER	0x00000002
JXFS_SIU_COINDISPENSER	0x00000003
JXFS_SIU_RECEIPTPRINTER	0x00000004
JXFS_SIU_PASSBOOKPRINTER	0x00000005
JXFS_SIU_ENVDEPOSITORY	0x00000006
JXFS_SIU_CHEQUEUNIT	0x00000007
JXFS_SIU_BILLACCEPTOR	0x00000008
JXFS_SIU_ENVDISPENSER	0x00000009

Code	Value
JXFS_SIU_SCANNER	0x0000000A
JXFS_SIU_COINACCEPTOR	0x0000000B
JXFS_SIU_DOCUMENTPRINTER	0x0000000C

12 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.

The name of the device service interface for SIU is IJxfsSiuService.