



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

---

# WORKSHOP AGREEMENT

**CWA 13937-8**

August 2000

---

ICS 35.240.40

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

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

© 2000 CEN

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

**Ref. No CWA 13937-8:2000 E**

## Foreword

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

The CEN/ISSS J/XFS Workshop gathers suppliers (among others the J/XFS Forum members), service providers as well as banks and other financial service companies. A list of companies participating in this Workshop and in support of this CWA is available from the CEN/ISSS Secretariat. The specification was agreed upon by the J/XFS Workshop Meeting of 1999-12-15/16 in Geneva and a subsequent electronic review by the Workshop participants, and the final version was sent to CEN for publication on 2000/06-21.

The specification is continuously reviewed and commented in the CEN/ISSS J/XFS Workshop. It is therefore expected that an update of the specification will be published in due time as a CWA, superseding this one. The information published in this CWA is furnished for informational purposes only. CEN/ISSS makes no warranty expressed or implied, with respect to this document. Updates of the specification will be available from the CEN/ISSS J/XFS Workshop public web pages pending their integration in a new version of the CWA (see: <http://www.cenorm.be/iss/wkshp/j-xfs/cwa-updates>).

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

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

This CWA is composed of the following parts:

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

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

## Contents

<b>1 SCOPE .....</b>	<b>5</b>
<b>2 OVERVIEW .....</b>	<b>6</b>
<b>3 DEVICE BEHAVIOUR.....</b>	<b>7</b>
3.1 DEVICE OPEN().....	7
<b>4 CLASS HIERARCHY .....</b>	<b>8</b>
<b>5 CLASS AND INTERFACE SUMMARY.....</b>	<b>9</b>
5.1 SUPPORT CLASSES.....	10
<b>6 COMPATIBILITY.....</b>	<b>12</b>
<b>7 CLASS AND INTERFACE DETAILS.....</b>	<b>13</b>
7.1 ACCESS TO PROPERTIES.....	13
7.2 EXCEPTIONS.....	13
7.3 JXFSIU.....	14
7.3.1 Introduction.....	14
7.3.2 Properties.....	14
7.3.3 Methods.....	15
<b>8 SUPPORT CLASSES.....</b>	<b>18</b>
8.1 JXFSIUPORTSTATUS.....	18
8.2 JXFSIUSENSORSTATUS.....	19
8.2.1 Properties.....	19
8.3 JXFSIUDOORSTATUS.....	22
8.3.1 Properties.....	22
8.4 JXFSIUINDICATORSTATUS.....	24
8.4.1 Properties.....	24
8.5 JXFSIU AUXILIARYSTATUS.....	26
8.5.1 Properties.....	26
8.6 JXFSIUGUIDLIGHTSTATUS.....	28
8.6.1 Properties.....	28
8.7 JXFSIUSTATUS.....	29
8.7.1 Summary.....	29
8.7.2 Properties.....	30
8.8 JXFSIUSENSORCAPABILITY.....	40
8.8.1 Properties.....	40
8.8.2 Methods.....	41
8.9 JXFSIUDOORCAPABILITY.....	42
8.9.1 Properties.....	43
8.9.2 Methods.....	44
8.10 JXFSIUINDICATORCAPABILITY.....	46
8.10.1 Properties.....	46
8.10.2 Methods.....	46
8.11 JXFSIU AUXILIARYCAPABILITY.....	47
8.11.1 Properties.....	48
8.11.2 Methods.....	49
8.12 JXFSIUGUIDLIGHTCAPABILITY.....	50
8.12.1 Properties.....	50
8.12.2 Methods.....	50
8.13 JXFSIUCAPABILITIES.....	51
8.13.1 Summary.....	51
8.13.2 Properties.....	52
8.14 JXFSIUENABLE.....	61

8.14.1 Properties .....	61
8.15 JXFSIUENABLEEVENTS .....	62
8.15.1 Summary .....	62
8.15.2 Properties .....	63
8.16 JXFSIUDOORPORT .....	64
8.16.1 Properties .....	64
8.17 JXFSIUINDICATORPORT .....	65
8.17.1 Properties .....	65
8.18 JXFSIU AUXILIARYPORT .....	67
8.18.1 Properties .....	68
8.19 JXFSIUGUIDLIGHTPORT .....	69
8.19.1 Properties .....	69
8.20 JXFSIUSETPORTS .....	70
8.20.1 Summary .....	70
8.20.2 Properties .....	70
8.21 JXFSIUSETDOOR .....	75
8.21.1 Summary .....	75
8.21.2 Properties .....	75
8.22 JXFSIUSETINDICATOR .....	76
8.22.1 Summary .....	76
8.22.2 Properties .....	76
8.23 JXFSIUSETAUXILIARY .....	77
8.23.1 Summary .....	77
8.23.2 Properties .....	77
8.24 JXFSIUSETGUIDLIGHT .....	78
8.24.1 Summary .....	78
8.24.2 Properties .....	78
8.25 JXFSIUPORTCHANGESTATUS .....	79
8.25.1 Summary .....	79
8.25.2 Properties .....	79
8.26 JXFSIUPORTERROR .....	82
8.26.1 Summary .....	82
8.26.2 Properties .....	82
<b>9 CODES .....</b>	<b>85</b>
9.1 ERROR CODES .....	85
9.2 STATUS CODES .....	85
9.3 INDEX CODES .....	86
<b>10 DEVICE SERVICE INTERFACE METHODS .....</b>	<b>90</b>
<b>INDEX .....</b>	<b>91</b>

## 1 Scope

This document describes the Sensors and Indicators 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.

## 2 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;**
- **Guidance lights.**

The J/XFS Sensors and Indicators 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.

### 3 Device behavior

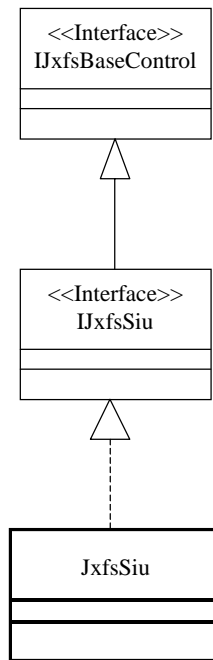
#### 3.1 Device open()

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

JXFS_E_HARDWAREERROR	If the device could not be accessed. This may be that the device is not connected or broken. This error should only be issued, if the device service does not see a reasonable chance to make the device work again. For (maybe temporary) error conditions, the open should succeed but the device status should indicate the error condition.
JXFS_E_OPEN	The open was already done by this Device Control.

## 4 Class Hierarchy

J/XFS  
Sensors and Indicators  
Control Interfaces





## 5 Class and Interface Summary

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

Class or Interface	Name	Description	Extends / Implements
Interface	<b>IJxfsBaseControl</b>	Base interface for all device controls. Contains methods specific to all the device controls.	--
Class	<b>JxfsBaseControl</b>	Base class for all device controls. Implements the methods defined in the <b>IJxfsBaseControl</b> Interface. Contains the properties specific to all device controls.	Implements: <b>IJxfsBaseControl</b>
Interface	<b>IJxfsSiu</b>	Base interface for all sensor and indicator controls.	Extends: <b>IJxfsBaseControl</b>
Class	<b>JxfsSiu</b>	Class for the SIU control	Extends: <b>JxfsBaseControl</b> Implements: <b>IJxfsSiu</b>

## 5.1 Support Classes

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

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

## 6 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 compatibility. 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

## 7 Class and Interface Details

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

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

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

### 7.1 Access to properties

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

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

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

#### **getProperty**

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

#### **setProperty**

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

### 7.2 Exceptions

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

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

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

## 7.3 IJxfsSiu

### 7.3.1 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.

#### Summary

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

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

### 7.3.2 Properties

#### capabilities (R)

Type	<i>JxfsSiuCapabilities</i>
Initial Value	<i>Depends on device</i>
Description	<i>see JxfsSiuCapabilities</i>

### 7.3.3 Methods

#### enableEvents

<b>Syntax</b>	<i>identificationID enableEvents(JxfsSiuEnableEvents events) throws JxfsException;</i>										
<b>Description</b>	This command is used to define the events that shall issue a status event in case of a change.										
<b>Parameter</b>	<table border="0"> <thead> <tr> <th style="text-align: left;"><b>Type</b></th> <th style="text-align: left;"><b>Name</b></th> <th style="text-align: left;"><b>Meaning</b></th> </tr> </thead> <tbody> <tr> <td><i>JxfsSiuEnableEvents</i></td> <td>events</td> <td>Specifies the events to be changed.</td> </tr> </tbody> </table>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>	<i>JxfsSiuEnableEvents</i>	events	Specifies the events to be changed.				
<b>Type</b>	<b>Name</b>	<b>Meaning</b>									
<i>JxfsSiuEnableEvents</i>	events	Specifies the events to be changed.									
<b>Exceptions</b>	No additional exceptions generated.										
<b>Events</b>	<p>Additional Events can be generated :</p> <p><b>OperationCompleteEvent</b></p> <p>When the enabling of events is completed an OperationCompleteEvent will be sent by J/XFS SIU Device Control to all registered OperationCompleteListeners with the following data:</p> <table border="0"> <thead> <tr> <th style="text-align: left;"><b>Field</b></th> <th style="text-align: left;"><b>Value</b></th> </tr> </thead> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_SIU_ENABLE_EVENTS</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table>	<b>Field</b>	<b>Value</b>	<i>operationID</i>	JXFS_O_SIU_ENABLE_EVENTS	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR	<i>data</i>	none
<b>Field</b>	<b>Value</b>										
<i>operationID</i>	JXFS_O_SIU_ENABLE_EVENTS										
<i>identificationID</i>	The corresponding ID										
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR										
<i>data</i>	none										

#### setPorts

<b>Syntax</b>	<i>identificationID setPorts(JxfsSiuSetPorts ports) throws JxfsException;</i>										
<b>Description</b>	This method is used to set/change the current condition of an output port.										
<b>Parameter</b>	<table border="0"> <thead> <tr> <th style="text-align: left;"><b>Type</b></th> <th style="text-align: left;"><b>Name</b></th> <th style="text-align: left;"><b>Meaning</b></th> </tr> </thead> <tbody> <tr> <td><i>JxfsSiuSetPorts</i></td> <td>ports</td> <td>Specifies the ports to be changed and the values they shall be changed to.</td> </tr> </tbody> </table>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>	<i>JxfsSiuSetPorts</i>	ports	Specifies the ports to be changed and the values they shall be changed to.				
<b>Type</b>	<b>Name</b>	<b>Meaning</b>									
<i>JxfsSiuSetPorts</i>	ports	Specifies the ports to be changed and the values they shall be changed to.									
<b>Exceptions</b>	No additional exceptions generated.										
<b>Events</b>	<p>Additional Events can be generated :</p> <p><b>OperationCompleteEvent</b></p> <p>When the selected ports have been changed an OperationCompleteEvent will be sent by J/XFS SIU Device Control to all registered OperationCompleteListeners with the following data:</p> <table border="0"> <thead> <tr> <th style="text-align: left;"><b>Field</b></th> <th style="text-align: left;"><b>Value</b></th> </tr> </thead> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_SIU_SET_PORT</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table>	<b>Field</b>	<b>Value</b>	<i>operationID</i>	JXFS_O_SIU_SET_PORT	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR	<i>data</i>	none
<b>Field</b>	<b>Value</b>										
<i>operationID</i>	JXFS_O_SIU_SET_PORT										
<i>identificationID</i>	The corresponding ID										
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR										
<i>data</i>	none										

## setPorts

<b>Syntax</b>	<i>identificationID setPorts(JxfsSiuSetDoor door) throws JxfsException;</i>		
<b>Description</b>	This method is used to set/change the current condition of a specific door port.		
<b>Parameter</b>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>
	<i>JxfsSiuSetDoor</i>	door	Specifies the door to be changed and the value the door shall be changed to.
<b>Exceptions</b>	No additional exceptions generated.		
<b>Events</b>	Additional Events can be generated : <b>OperationCompleteEvent</b> When the selected door has been changed an OperationCompleteEvent will be sent by J/XFS SIU Device Control to all registered OperationCompleteListeners with the following data:		
	<b>Field</b>	<b>Value</b>	
	<i>operationID</i>	JXFS_O_SIU_SET_PORT	
	<i>identificationID</i>	The corresponding ID	
	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR	
	<i>data</i>	none	

## setPorts

<b>Syntax</b>	<i>identificationID setPorts(JxfsSiuSetIndicator indicator) throws JxfsException;</i>		
<b>Description</b>	This method is used to set/change the current condition of a specific indicator port.		
<b>Parameter</b>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>
	<i>JxfsSiuSetIndicator</i>	indicator	Specifies the indicator to be changed and the value the indicator shall be changed to.
<b>Exceptions</b>	No additional exceptions generated.		
<b>Events</b>	Additional Events can be generated : <b>OperationCompleteEvent</b> When the selected indicator has been changed an OperationCompleteEvent will be sent by J/XFS SIU Device Control to all registered OperationCompleteListeners with the following data:		
	<b>Field</b>	<b>Value</b>	
	<i>operationID</i>	JXFS_O_SIU_SET_PORT	
	<i>identificationID</i>	The corresponding ID	
	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR	
	<i>data</i>	none	



## setPorts

<b>Syntax</b>	<i>identificationID setPorts(JxfsSiuSetAuxiliary auxiliary) throws JxfsException;</i>										
<b>Description</b>	This method is used to set/change the current condition of a specific auxiliary port.										
<b>Parameter</b>	<table border="0"> <thead> <tr> <th style="text-align: left;"><b>Type</b></th> <th style="text-align: left;"><b>Name</b></th> <th style="text-align: left;"><b>Meaning</b></th> </tr> </thead> <tbody> <tr> <td><i>JxfsSiuSetAuxiliary</i></td> <td>auxiliary</td> <td>Specifies the auxiliary to be changed and the value the auxiliary shall be changed to.</td> </tr> </tbody> </table>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>	<i>JxfsSiuSetAuxiliary</i>	auxiliary	Specifies the auxiliary to be changed and the value the auxiliary shall be changed to.				
<b>Type</b>	<b>Name</b>	<b>Meaning</b>									
<i>JxfsSiuSetAuxiliary</i>	auxiliary	Specifies the auxiliary to be changed and the value the auxiliary shall be changed to.									
<b>Exceptions</b>	No additional exceptions generated.										
<b>Events</b>	<p>Additional Events can be generated :</p> <p><b>OperationCompleteEvent</b></p> <p>When the selected auxiliary indicator has been changed an OperationCompleteEvent will be sent by J/XFS SIU Device Control to all registered OperationCompleteListeners with the following data:</p> <table border="0"> <thead> <tr> <th style="text-align: left;"><b>Field</b></th> <th style="text-align: left;"><b>Value</b></th> </tr> </thead> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_SIU_SET_PORT</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table>	<b>Field</b>	<b>Value</b>	<i>operationID</i>	JXFS_O_SIU_SET_PORT	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR	<i>data</i>	none
<b>Field</b>	<b>Value</b>										
<i>operationID</i>	JXFS_O_SIU_SET_PORT										
<i>identificationID</i>	The corresponding ID										
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR										
<i>data</i>	none										

## setPorts

<b>Syntax</b>	<i>identificationID setPorts(JxfsSiuSetGuidLight guidLight) throws JxfsException;</i>										
<b>Description</b>	This method is used to set/change the current condition of a specific guidance light.										
<b>Parameter</b>	<table border="0"> <thead> <tr> <th style="text-align: left;"><b>Type</b></th> <th style="text-align: left;"><b>Name</b></th> <th style="text-align: left;"><b>Meaning</b></th> </tr> </thead> <tbody> <tr> <td><i>JxfsSiuSetGuidLight</i></td> <td>guidLight</td> <td>Specifies the guidance light to be changed and the value the guidance light shall be changed to.</td> </tr> </tbody> </table>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>	<i>JxfsSiuSetGuidLight</i>	guidLight	Specifies the guidance light to be changed and the value the guidance light shall be changed to.				
<b>Type</b>	<b>Name</b>	<b>Meaning</b>									
<i>JxfsSiuSetGuidLight</i>	guidLight	Specifies the guidance light to be changed and the value the guidance light shall be changed to.									
<b>Exceptions</b>	No additional exceptions generated.										
<b>Events</b>	<p>Additional Events can be generated :</p> <p><b>OperationCompleteEvent</b></p> <p>When the selected guidance light has been changed an OperationCompleteEvent will be sent by J/XFS SIU Device Control to all registered OperationCompleteListeners with the following data:</p> <table border="0"> <thead> <tr> <th style="text-align: left;"><b>Field</b></th> <th style="text-align: left;"><b>Value</b></th> </tr> </thead> <tbody> <tr> <td><i>operationID</i></td> <td>JXFS_O_SIU_SET_PORT</td> </tr> <tr> <td><i>identificationID</i></td> <td>The corresponding ID</td> </tr> <tr> <td><i>result</i></td> <td>JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR</td> </tr> <tr> <td><i>data</i></td> <td>none</td> </tr> </tbody> </table>	<b>Field</b>	<b>Value</b>	<i>operationID</i>	JXFS_O_SIU_SET_PORT	<i>identificationID</i>	The corresponding ID	<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR	<i>data</i>	none
<b>Field</b>	<b>Value</b>										
<i>operationID</i>	JXFS_O_SIU_SET_PORT										
<i>identificationID</i>	The corresponding ID										
<i>result</i>	JXFS_RC_SUCCESSFUL JXFS_E_SIU_INVALID_PORT JXFS_E_SIU_SYNTAX JXFS_E_SIU_PORT_ERROR										
<i>data</i>	none										

## 8 Support Classes

### 8.1 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	<i>none</i>	

Event	May occur after
none	

## 8.2 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

### 8.2.1 Properties

#### sensorStatus (R)

<b>Type</b>	<i>int</i>																		
<b>Initial Value</b>	none																		
<b>Description</b>	<p>Specifies the current status of the specific sensor port. The possible values and their meaning depend on the type of sensor port.</p> <p>If any of these sensor ports is not available this is defined as</p> <table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_SIU_NOT_AVAILABLE</td> <td>The port is not available.</td> </tr> </tbody> </table> <p>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 <b>Run</b> mode is used for normal consumer operations/transactions. The <b>Maintenance</b> mode is used when replenish the terminal. The <b>Supervisor</b> 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:</p> <table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_SIU_RUN</td> <td>The switch is in Run Mode.</td> </tr> <tr> <td>JXFS_SIU_MAINTENANCE</td> <td>The switch is in Maintenance Mode.</td> </tr> <tr> <td>JXFS_SIU_SUPERVISOR</td> <td>The switch is in Supervisor mode.</td> </tr> </tbody> </table> <p>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:</p> <table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_SIU_OFF</td> <td>There is no indication of a tampering attempt.</td> </tr> <tr> <td>JXFS_SIU_ON</td> <td>There has been a tampering attempt.</td> </tr> </tbody> </table>	Value	Meaning	JXFS_SIU_NOT_AVAILABLE	The port is not available.	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.	Value	Meaning	JXFS_SIU_OFF	There is no indication of a tampering attempt.	JXFS_SIU_ON	There has been a tampering attempt.
Value	Meaning																		
JXFS_SIU_NOT_AVAILABLE	The port is not available.																		
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.																		
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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
JXFS_SIU_OFF	The switch is set.
JXFS_SIU_ON	The switch is not set.

### 8.3 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

#### 8.3.1 Properties

##### doorStatus (R)

<b>Type</b>	<i>int</i>																								
<b>Initial Value</b>	none																								
<b>Description</b>	<p>Specifies the current status of the specific door. The possible values and their meaning depend on the type of door.</p> <p>If any of these door ports is not available then this is defined as</p> <table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_SIU_NOT_AVAILABLE</td> <td>The status is not available.</td> </tr> </tbody> </table> <p>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:</p> <table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_SIU_OPEN</td> <td>At least one of the Cabinet Doors is open.</td> </tr> <tr> <td>JXFS_SIU_CLOSED</td> <td>The Cabinet Doors are closed.</td> </tr> <tr> <td>JXFS_SIU_LOCKED</td> <td>The Cabinet Doors are closed and locked.</td> </tr> <tr> <td>JXFS_SIU_BOLTED</td> <td>The Cabinet Doors are closed, locked and bolted.</td> </tr> </tbody> </table> <p>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:</p> <table> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>JXFS_SIU_OPEN</td> <td>At least one of the Safe Doors is open.</td> </tr> <tr> <td>JXFS_SIU_CLOSED</td> <td>The Safe Doors are closed.</td> </tr> <tr> <td>JXFS_SIU_LOCKED</td> <td>The Safe Doors are closed and locked.</td> </tr> <tr> <td>JXFS_SIU_BOLTED</td> <td>The Safe Doors are closed, locked and bolted.</td> </tr> </tbody> </table>	Value	Meaning	JXFS_SIU_NOT_AVAILABLE	The status is not available.	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.	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.
Value	Meaning																								
JXFS_SIU_NOT_AVAILABLE	The status is not available.																								
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.																								
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:

<b>Value</b>	<b>Meaning</b>
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.

<b>Value</b>	<b>Meaning</b>
JXFS_SIU_OPEN	The Front Top Door is open.
JXFS_SIU_CLOSED	The Front Top Door is closed.
JXFS_SIU_BOLTED	The Front Top Door is closed and bolted.

## 8.4 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

### 8.4.1 Properties

#### indicatorStatus (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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:
<b>Value</b>	<b>Meaning</b>
JXFS_SIU_NOT_AVAILABLE	The status is not available.
	Specifies the state of the Open/Closed Indicator as one of the following flags:
<b>Value</b>	<b>Meaning</b>
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:
<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>	<b>Type</b>
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 turned on continuously.	C

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

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
JXFS_SIU_OFF	The Fascia Light is turned off.
JXFS_SIU_ON	The Fascia Light is turned on.

## 8.5 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

### 8.5.1 Properties

#### auxiliaryStatus (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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:

<b>Value</b>	<b>Meaning</b>
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:

<b>Value</b>	<b>Meaning</b>
JXFS_SIU_AVAILABLE	A software poweroff is available/possible.

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

<b>Value</b>	<b>Meaning</b>
JXFS_SIU_OFF	The Relay is turned off.
JXFS_SIU_ON	The Relay is turned on.

## 8.6 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

### 8.6.1 Properties

#### guidlightStatus (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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).

## 8.7 JxfsSiuStatus

This class contains properties and methods to query the status 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 extent this status in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

### 8.7.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
sensorStatus	<b>JxfsSiuSensorStatus[]</b>	R	
doorStatus	<b>JxfsSiuDoorStatus[]</b>	R	
indicatorStatus	<b>JxfsSiuIndicatorStatus[]</b>	R	
auxiliaryStatus	<b>JxfsSiuAuxiliaryStatus[]</b>	R	
guidlightStatus	<b>JxfsSiuGuidLightStatus[]</b>	R	

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

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

Event	May occur after
none	

## 8.7.2 Properties

### sensorStatus[JXFS\_SIU\_OPERATORSWITCH]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>				
<b>Description</b>	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 <b>Run</b> mode is used for normal consumer operations/transactions. The <b>Maintenance</b> mode is used when replenish the terminal. The <b>Supervisor</b> mode is used when operating the terminal for service and testing. Supervisor mode has higher priority than maintenance mode.				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

### sensorStatus[JXFS\_SIU\_TAMPER]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>				
<b>Description</b>	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).				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

### sensorStatus[JXFS\_SIU\_INTTAMPER]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	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).
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:

### sensorStatus[JXFS\_SIU\_SEISMIC]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>				
<b>Description</b>	Specifies the state of the Seismic Sensor. This sensor indicates whether the terminal has been shaken (e.g. burglar attempt or seismic activity).				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

### sensorStatus[JXFS\_SIU\_HEAT]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>				
<b>Description</b>	Specifies the state of the Heat Sensor. This sensor is triggered by excessive heat (fire) near the terminal.				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

**sensorStatus[JXFS\_SIU\_PROXIMITY]**

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	Specifies the state of the Proximity Sensor. This sensor is triggered by movements around the terminal.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**sensorStatus[JXFS\_SIU\_AMBLIGHT]**

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	Specifies the state of the Ambient Light Sensor. This sensor indicates the level of ambient light around the terminal.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**sensorStatus[JXFS\_SIU\_INPUT1]**

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	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.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**sensorStatus[JXFS\_SIU\_INPUT2]**

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	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.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**sensorStatus[JXFS\_SIU\_INPUT3]**

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	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.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### sensorStatus[JXFS\_SIU\_INPUT4]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	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.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### sensorStatus[JXFS\_SIU\_VENTILATOR]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	Specifies the state of the Ventilator.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### sensorStatus[JXFS\_SIU\_BOOTSWITCH]

<b>Type</b>	<i>JxfsSiuSensorStatus</i>
<b>Description</b>	Specifies the state of the Switch that indicates a Boot request.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### doorStatus[JXFS\_SIU\_CABINET]

<b>Type</b>	<i>JxfsSiuDoorStatus</i>
<b>Description</b>	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.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### doorStatus[JXFS\_SIU\_SAFE]

<b>Type</b>	<i>JxfsSiuDoorStatus</i>
<b>Description</b>	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.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.



**doorStatus[JXFS\_SIU\_VANDALSHIELD]**

<b>Type</b>	<i>JxfsSiuDoorStatus</i>
<b>Description</b>	Specifies the state of the Vandal Shield. The Vandal Shield is a door that open up for consumer access to the terminal.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**doorStatus[JXFS\_SIU\_FRONT\_TOP]**

<b>Type</b>	<i>JxfsSiuDoorStatus</i>
<b>Description</b>	Specifies the state of the Front Top Door.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**doorStatus[JXFS\_SIU\_REAR\_TOP]**

<b>Type</b>	<i>JxfsSiuDoorStatus</i>
<b>Description</b>	Specifies the state of the Rear Top Door.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**doorStatus[JXFS\_SIU\_FRONT\_BOTTOM]**

<b>Type</b>	<i>JxfsSiuDoorStatus</i>
<b>Description</b>	Specifies the state of the Front Bottom Door.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**doorStatus[JXFS\_SIU\_REAR\_BOTTOM]**

<b>Type</b>	<i>JxfsSiuDoorStatus</i>
<b>Description</b>	Specifies the state of the Rear Bottom Door.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**indicatorStatus[JXFS\_SIU\_OPENCLOSE]**

<b>Type</b>	<i>JxfsSiuIndicatorStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Open/Closed Indicator.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**indicatorStatus[JXFS\_SIU\_FASCIALIGHT]**

<b>Type</b>	<i>JxfsSiuIndicatorStatus</i>
-------------	-------------------------------

<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Fascia Light.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### indicatorStatus[JXFS\_SIU\_LOGOLIGHT]

<b>Type</b>	<i>JxfsSiuIndicatorStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Logo Light.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### indicatorStatus[JXFS\_SIU\_AUDIO]

<b>Type</b>	<i>JxfsSiuIndicatorStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Audio Indicator.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### indicatorStatus[JXFS\_SIU\_HEATING]

<b>Type</b>	<i>JxfsSiuIndicatorStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the internal heating
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### auxiliaryStatus[JXFS\_SIU\_VOLUME]

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	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.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### auxiliaryStatus[JXFS\_SIU\_UPS]

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Uninterruptable Power Supply device.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### auxiliaryStatus[JXFS\_SIU\_MONITOR]

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Monitor.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### auxiliaryStatus[JXFS\_SIU\_POWEROFF]

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the software Poweroff.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### auxiliaryStatus[JXFS\_SIU\_RELAY1]

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the first Relay.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### auxiliaryStatus[JXFS\_SIU\_RELAY2]

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the second Relay.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### **auxiliaryStatus[JXFS\_SIU\_RELAY3]**

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the third Relay.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### **auxiliaryStatus[JXFS\_SIU\_RELAY4]**

<b>Type</b>	<i>JxfsSiuAuxiliaryStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the fourth Relay.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### **guidlightStatus[JXFS\_SIU\_CARDUNIT]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the Card Unit (MSD/CCD).
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### **guidlightStatus[JXFS\_SIU\_PINPAD]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the PIN pad unit.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### **guidlightStatus[JXFS\_SIU\_NOTESDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the note dispenser unit.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

#### **guidlightStatus[JXFS\_SIU\_COINDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>				
<b>Initial Value</b>					
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the coin dispenser unit.				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

#### **guidlightStatus[JXFS\_SIU\_RECEIPTPRINTER]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>				
<b>Initial Value</b>					
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the receipt printer unit.				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

#### **guidlightStatus[JXFS\_SIU\_PASSBOOKPRINTER]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>				
<b>Initial Value</b>					
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the passbook printer unit.				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

#### **guidlightStatus[JXFS\_SIU\_ENVDEPOSITORY]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>				
<b>Initial Value</b>					
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the envelope depository unit.				
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of: <table><tr><td><b>Value</b></td><td><b>Meaning</b></td></tr><tr><td>JXFS_S_SIU_PORT_STATUS</td><td>The value of a port has changed.</td></tr></table>	<b>Value</b>	<b>Meaning</b>	JXFS_S_SIU_PORT_STATUS	The value of a port has changed.
<b>Value</b>	<b>Meaning</b>				
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.				

**guidlightStatus[JXFS\_SIU\_CHEQUEUNIT]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the cheque processing unit.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**guidlightStatus[JXFS\_SIU\_BILLACCEPTOR]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the bill acceptor unit.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

**guidlightStatus[JXFS\_SIU\_ENVDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightStatus</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the state of the Guidance Light Indicator on the envelope dispenser unit.
<b>Event</b>	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 StatusListeners a StatusEvent with a status value of:
<b>Value</b>	<b>Meaning</b>
JXFS_S_SIU_PORT_STATUS	The value of a port has changed.

## 8.8 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
<i>getProperty</i>	<i>Property</i>	
isRunModeSupported	boolean	
isMaintenanceModeSupported	boolean	
isSupervisorModeSupported	boolean	
isAvailable	boolean	

Event	May occur after
none	

### 8.8.1 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

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.



## 8.8.2 Methods

### isRunModeSupported

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

### isMaintenanceModeSupported

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

### isSupervisorModeSupported

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

### isAvailable

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

## 8.9 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
<i>getProperty</i>	<i>Property</i>	
isLockedSupported	boolean	
isBoltedSupported	boolean	
isClosedSupported	boolean	
isOpenSupported	boolean	
isServiceSupported	boolean	
isKeyboardSupported	boolean	
isAjarSupported	boolean	
isJammedSupported	boolean	

Event	May occur after
none	

## 8.9.1 Properties

### doorCapability (R)

**Type**  
**Initial Value**  
**Description**

*int*

none

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 open	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 Door 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 open	B

## 8.9.2 Methods

### isLockedSupported

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

### isBoltedSupported

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

### isClosedSupported

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

### isOpenSupported

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

### isServiceSupported

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

### isKeyboardSupported

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

## 8.10 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
<i>getProperty</i>	<i>Property</i>	
isAvailable	boolean	

Event	May occur after
none	

### 8.10.1 Properties

#### indicatorCapability (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	Specifies the 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.

### 8.10.2 Methods

#### isAvailable

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

## 8.11 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
<i>getProperty</i>	<i>Property</i>	
isAvailable	boolean	
isLowSupported	boolean	
isEngagedSupported	boolean	
isPoweringSupported	boolean	
isRecoveredSupported	boolean	

Event	May occur after
none	

### 8.11.1 Properties

#### auxiliaryCapability (R)

**Type**  
**Initial Value**  
**Description**

*int*

none

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

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



## 8.11.2 Methods

### isAvailable

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

### isLowSupported

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

### isEngagedSupported

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

### isPoweringSupported

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

### isRecoveredSupported

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

## 8.12 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
<i>getProperty</i>	<i>Property</i>	
isAvailable	boolean	

Event	May occur after
none	

### 8.12.1 Properties

#### guidLightCapability (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	Specifies the 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.
JXFS_SIU_AVAILABLE	The indicator is available.

### 8.12.2 Methods

#### isAvailable

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

## 8.13 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 extent the capabilities in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

### 8.13.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsType*

Property	Type	Access	Initialized after
sensorCapabilities	<b>JxfsSiuSensorCapability</b> []	R	
doorCapabilities	<b>JxfsSiuDoorCapability</b> []	R	
indicatorCapabilities	<b>JxfsSiuIndicatorCapability</b> []	R	
auxiliaryCapabilities	<b>JxfsSiuAuxiliaryCapability</b> []	R	
guidLightCapabilities	<b>JxfsSiuGuidLightCapability</b> []	R	

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

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

Event	May occur after
none	

## 8.13.2 Properties

### sensorCapabilities[JXFS\_SIU\_OPERATORSWITCH]

Type	<i>JxfsSiuSensorCapability</i>
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	<i>JxfsSiuSensorCapability</i>
Initial Value	0
Description	Specifies whether the Tamper Sensor is available.
Event	none

### sensorCapabilities[JXFS\_SIU\_INTTAMPER]

Type	<i>JxfsSiuSensorCapability</i>
Initial Value	0
Description	Specifies whether the Tamper Sensor for internal alarm is available.
Event	none

### sensorCapabilities[JXFS\_SIU\_SEISMIC]

Type	<i>JxfsSiuSensorCapability</i>
Initial Value	0
Description	Specifies whether the Seismic Sensor is available.
Event	none

### sensorCapabilities[JXFS\_SIU\_HEAT]

Type	<i>JxfsSiuSensorCapability</i>
Initial Value	0
Description	Specifies whether the Heat Sensor is available.
Event	none

### sensorCapabilities[JXFS\_SIU\_PROXIMITY]

Type	<i>JxfsSiuSensorCapability</i>
Initial Value	0
Description	Specifies whether the Proximity Sensor is available.
Event	none

### sensorCapabilities[JXFS\_SIU\_AMBLIGHT]

Type	<i>JxfsSiuSensorCapability</i>
Initial Value	0
Description	Specifies whether the Ambient Light Sensor is available.
Event	none

### sensorCapabilities[JXFS\_SIU\_INPUT1]

Type	<i>JxfsSiuSensorCapability</i>
Initial Value	0
Description	Specifies whether the first Input Contact is available.
Event	none

**sensorCapabilities[JXFS\_SIU\_INPUT2]**

<b>Type</b>	<i>JxfsSiuSensorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the second Input Contact is available.
<b>Event</b>	none

**sensorCapabilities[JXFS\_SIU\_INPUT3]**

<b>Type</b>	<i>JxfsSiuSensorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the third Input Contact is available.
<b>Event</b>	none

**sensorCapabilities[JXFS\_SIU\_INPUT4]**

<b>Type</b>	<i>JxfsSiuSensorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the fourth Input Contact is available.
<b>Event</b>	none

**sensorCapabilities[JXFS\_SIU\_VENTILATOR]**

<b>Type</b>	<i>JxfsSiuSensorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Ventilator is available.
<b>Event</b>	none

**sensorCapabilities[JXFS\_SIU\_BOOTSWITCH]**

<b>Type</b>	<i>JxfsSiuSensorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Switch that indicates a Boot request is available.
<b>Event</b>	none

**doorCapabilities[JXFS\_SIU\_CABINET]**

<b>Type</b>	<i>JxfsSiuDoorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Cabinet Doors are available, and if so, which states they can take.
<b>Event</b>	none

**doorCapabilities[JXFS\_SIU\_SAFE]**

<b>Type</b>	<i>JxfsSiuDoorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Safe Doors are available, and if so, which states they can take.
<b>Event</b>	none

**doorCapabilities[JXFS\_SIU\_VANDALSHIELD]**

<b>Type</b>	<i>JxfsSiuDoorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Vandal Shield is available, and if so, which states it can take.
<b>Event</b>	none

**doorCapabilities[JXFS\_SIU\_FRONT\_TOP]**

<b>Type</b>	<i>JxfsSiuDoorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Front Top Door is available, and if so, which states it can take.
<b>Event</b>	none

**doorCapabilities[JXFS\_SIU\_REAR\_TOP]**

<b>Type</b>	<i>JxfsSiuDoorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Rear Top Door is available, and if so, which states it can take.
<b>Event</b>	none

**doorCapabilities[JXFS\_SIU\_FRONT\_BOTTOM]**

<b>Type</b>	<i>JxfsSiuDoorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Front Bottom Door is available, and if so, which states it can take.
<b>Event</b>	none

**doorCapabilities[JXFS\_SIU\_REAR\_BOTTOM]**

<b>Type</b>	<i>JxfsSiuDoorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Rear Bottom Door is available, and if so, which states it can take.
<b>Event</b>	none

**indicatorCapabilities[JXFS\_SIU\_OPENCLOSE]**

<b>Type</b>	<i>JxfsSiuIndicatorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Open/Closed Indicator is available.
<b>Event</b>	none

**indicatorCapabilities[JXFS\_SIU\_FASCIALIGHT]**

<b>Type</b>	<i>JxfsSiuIndicatorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Fascia Light is available.
<b>Event</b>	none

**indicatorCapabilities[JXFS\_SIU\_AUDIO]**

<b>Type</b>	<i>JxfsSiuIndicatorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Audio Indicator device is available.
<b>Event</b>	none

**indicatorCapabilities[JXFS\_SIU\_HEATING]**

<b>Type</b>	<i>JxfsSiuIndicatorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the internal Heating device is available.
<b>Event</b>	none

**indicatorCapabilities[JXFS\_SIU\_LOGOLIGHT]**

<b>Type</b>	<i>JxfsSiuIndicatorCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Logo Light is available.
<b>Event</b>	none

**auxiliaryCapabilities[JXFS\_SIU\_VOLUME]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the volume control is available, and if so, the increment/decrement value for the volume control.
<b>Event</b>	none

**auxiliaryCapabilities[JXFS\_SIU\_UPS]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the UPS device is available, and if so, which states it can take.
<b>Event</b>	none

**auxiliaryCapabilities[JXFS\_SIU\_MONITOR]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the possibility to switch the monitor is available.
<b>Event</b>	none

**auxiliaryCapabilities[JXFS\_SIU\_POWEROFF]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the software Poweroff is available.
<b>Event</b>	none

**auxiliaryCapabilities[JXFS\_SIU\_RELAY1]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the first Relay is available.
<b>Event</b>	none

**auxiliaryCapabilities[JXFS\_SIU\_RELAY2]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the second Relay is available.
<b>Event</b>	none

**auxiliaryCapabilities[JXFS\_SIU\_RELAY3]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the third Relay is available.
<b>Event</b>	none



**auxiliaryCapabilities[JXFS\_SIU\_RELAY4]**

<b>Type</b>	<i>JxfsSiuAuxiliaryCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the fourth Relay is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_CARDUNIT]**

<b>Type</b>	<i>JxfsSiuGuidLightCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the Card Unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_PINPAD]**

<b>Type</b>	<i>JxfsSiuGuidLightCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the PIN pad is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_NOTESDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the note dispenser unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_COINDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the coin dispenser unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_RECEIPTPRINTER]**

<b>Type</b>	<i>JxfsSiuGuidLightCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the receipt printer unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_PASSBOOKPRINTER]**

<b>Type</b>	<i>JxfsSiuGuidLightCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the passbook printer unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_ENVDEPOSITORY]**

<b>Type</b>	<i>JxfsSiuGuidLightCapability</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the envelope depository unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_CHEQUEUNIT]**

<b>Type</b>	<i>JxfsSiuGuidLightCapabilitiy</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the cheque processing unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_BILLACCEPTOR]**

<b>Type</b>	<i>JxfsSiuGuidLightCapabilitiy</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the bill acceptor unit is available.
<b>Event</b>	none

**guidLightCapabilities[JXFS\_SIU\_ENVDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightCapabilitiy</i>
<b>Initial Value</b>	0
<b>Description</b>	Specifies whether the Guidance Light Indicator on the envelope dispenser unit is available.
<b>Event</b>	none

## 8.14 JxfsSiuEnable

This class specifies the capabilities of the auxiliary indicators.

### 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

### 8.14.1 Properties

#### enable (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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:
<b>Value</b>	<b>Meaning</b>
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.

## 8.15 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 extent the capabilities in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

### 8.15.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
sensorEnable	<b>JxfsSiuEnable[]</b>	R/W	
doorEnable	<b>JxfsSiuEnable[]</b>	R/W	
indicatorEnable	<b>JxfsSiuEnable[]</b>	R/W	
auxiliaryEnable	<b>JxfsSiuEnable[]</b>	R/W	
guidlightEnable	<b>JxfsSiuEnable[]</b>	R/W	

Constructor	Parameter	Parameter-Type
JxfsSiuEnableEvents	sensorEnable	JxfsSiuEnable[]
	doorEnable	JxfsSiuEnable[]
	indicatorEnable	JxfsSiuEnable[]
	auxiliaryEnable	JxfsSiuEnable[]
	guidlightEnable	JxfsSiuEnable[]

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

Event	May occur after
none	

## 8.15.2 Properties

### sensorEnable[] (R/W)

Type	<i>JxfsSiuEnable[]</i>
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	<i>JxfsSiuEnable[]</i>
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	<i>JxfsSiuEnable[]</i>
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	<i>JxfsSiuEnable[]</i>
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	<i>JxfsSiuEnable[]</i>
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

## 8.16 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

### 8.16.1 Properties

#### state (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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.

## 8.17 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

### 8.17.1 Properties

#### state (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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 Fascia Light is turned off.
JXFS_SIU_ON	The Fascia 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:

<b>Value</b>	<b>Meaning</b>	<b>Type</b>
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 turned on continuously.	C

Specifies if the internal Heating (JXFS\_SIU\_HEATING) shall be turned on or off as one of the following values:

<b>Value</b>	<b>Meaning</b>
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.



## 8.18 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
none	

## 8.18.1 Properties

### state (R)

**Type**  
**Initial Value**  
**Description**

*int*

none

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**

JXFS\_SIU\_NO\_CHANGE

1, ..., 1000

**Meaning**

Do not change the current volume.

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**

JXFS\_SIU\_NO\_CHANGE

JXFS\_SIU\_ENGAGE

JXFS\_SIU\_DISENGAGE

**Meaning**

Do not change the current state.

Engage the UPS

Disengage the UPS

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

**Value**

JXFS\_SIU\_NO\_CHANGE

JXFS\_SIU\_OFF

JXFS\_SIU\_ON

**Meaning**

Do not change the current state.

Switch the Monitor off.

Switch the Monitor on.

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

**Value**

JXFS\_SIU\_NO\_CHANGE

JXFS\_SIU\_OFF

JXFS\_SIU\_RESTART

**Meaning**

Do not change the current state.

Switch the power off.

Restart (cold start) the machine.

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**

JXFS\_SIU\_NO\_CHANGE

JXFS\_SIU\_OFF

JXFS\_SIU\_ON

**Meaning**

Do not change the current state.

Switch the relay off.

Switch the relay on.

## 8.19 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
<i>getProperty</i>	<i>Property</i>	

Event	May occur after
None	

### 8.19.1 Properties

#### state (R)

<b>Type</b>	<i>Int</i>
<b>Initial Value</b>	none
<b>Description</b>	Specifies if the SIU device shall change the state of the specified guidance light.
	Specifies the possible values for the guidance lights:
<b>Value</b>	<b>Meaning</b>
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).

## 8.20 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 extent this status in upcoming versions with new kinds of sensors and indicators while maintaining backward compatibility.

### 8.20.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
doorPorts	<b>JxfsSiuDoorPort[]</b>	R/W	
indicatorPorts	<b>JxfsSiuIndicatorPort[]</b>	R/W	
auxiliaryPorts	<b>JxfsSiuAuxiliaryPort[]</b>	R/W	
guidlightPorts	<b>JxfsSiuGuidLightPort[]</b>	R/W	

Constructor	Parameter	Parameter-Type
JxfsSiuSetPorts	doorPorts	JxfsSiuDoorPort[]
	indicatorPorts	JxfsSiuIndicatorPort[]
	auxiliaryPorts	JxfsSiuAuxiliaryPort[]
	guidlightPorts	JxfsSiuGuidLightPort[]

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

Event	May occur after
none	

### 8.20.2 Properties

#### doorPorts[JXFS\_SIU\_CABINET]

<b>Type</b>	<i>JxfsSiuDoorPorts</i>
<b>Description</b>	Specifies whether the Cabinet Doors shall be bolted or unbolted.
<b>Event</b>	none

#### doorPorts[JXFS\_SIU\_SAFE]

<b>Type</b>	<i>JxfsSiuDoorPorts</i>
<b>Description</b>	Specifies whether the Safe Doors shall be bolted or unbolted.
<b>Event</b>	none

#### doorPorts [JXFS\_SIU\_VANDALSHIELD]

<b>Type</b>	<i>JxfsSiuDoorPorts</i>
<b>Description</b>	Specifies whether the Vandal Shield shall change ist position.
<b>Event</b>	none

#### doorPorts [JXFS\_SIU\_FRONT\_TOP]

<b>Type</b>	<i>JxfsSiuDoorPorts</i>
<b>Description</b>	Specifies whether the Front Top Door shall be bolted or unbolted.
<b>Event</b>	none

**doorPorts[JXFS\_SIU\_REAR\_TOP]**

Type	<i>JxfsSiuDoorPorts</i>
Description	Specifies whether the Rear Top Door shall be bolted or unbolted.
Event	none

**doorPorts[JXFS\_SIU\_FRONT\_BOTTOM]**

Type	<i>JxfsSiuDoorPorts</i>
Description	Specifies whether the Front Bottom Door shall be bolted or unbolted.
Event	none

**doorPorts[JXFS\_SIU\_REAR\_BOTTOM]**

Type	<i>JxfsSiuDoorPorts</i>
Description	Specifies whether the Rear Bottom Door shall be bolted or unbolted.
Event	none

**indicatorPorts[JXFS\_SIU\_OPENCLOSE]**

Type	<i>JxfsSiuIndicatorPorts</i>
Initial Value	
Description	Specifies whether the Open/Closed Indicator shall show Open or Close to a consumer.
Event	none

**indicatorPorts[JXFS\_SIU\_FASCIALIGHT]**

Type	<i>JxfsSiuIndicatorPorts</i>
Initial Value	
Description	Specifies whether the Fascia Light shall be turned on or off.
Event	none

**indicatorPorts[JXFS\_SIU\_AUDIO]**

Type	<i>JxfsSiuIndicatorPorts</i>
Initial Value	
Description	Specifies whether the Audio Indicator shall be turned on or off.
Event	none

**indicatorPorts[JXFS\_SIU\_HEATING]**

Type	<i>JxfsSiuIndicatorPorts</i>
Initial Value	
Description	Specifies whether the internal heating shall be turned on or off.
Event	none

**indicatorPorts[JXFS\_SIU\_LOGOLIGHT]**

Type	<i>JxfsSiuIndicatorPorts</i>
Initial Value	
Description	Specifies whether the Logo Light shall be turned on or off.
Event	none

**auxiliaryPorts[JXFS\_SIU\_VOLUME]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the value of the volume control shall be changed or not and if it shall be changed then to which level.
<b>Event</b>	none

**auxiliaryPorts[JXFS\_SIU\_UPS]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Uninterruptable Power Supply device shall be engaged or disengaged.
<b>Event</b>	none

**auxiliaryPorts[JXFS\_SIU\_MONITOR]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Monitor shall be switched on or off.
<b>Event</b>	none

**auxiliaryPorts[JXFS\_SIU\_POWEROFF]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the software Poweroff shall be activated or not.
<b>Event</b>	None

**auxiliaryPorts[JXFS\_SIU\_RELAY1]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the first Relay shall be switched on or off.
<b>Event</b>	None

**auxiliaryPorts[JXFS\_SIU\_RELAY2]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the second Relay shall be switched on or off.
<b>Event</b>	None

**auxiliaryPorts[JXFS\_SIU\_RELAY3]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the third Relay shall be switched on or off.
<b>Event</b>	None

**auxiliaryPorts[JXFS\_SIU\_RELAY4]**

<b>Type</b>	<i>JxfsSiuAuxiliaryPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the fourth Relay shall be switched on or off.
<b>Event</b>	None

**guidlightPorts[JXFS\_SIU\_CARDUNIT]**

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

**guidlightPorts[JXFS\_SIU\_PINPAD]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the PIN pad unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_NOTESDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the note dispenser unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_COINDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the coin dispenser unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_RECEIPTPRINTER]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the receipt printer unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_PASSBOOKPRINTER]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the passbook printer unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_ENVDEPOSITORY]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the envelope depository unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_CHEQUEUNIT]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the cheque processing unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_BILLACCEPTOR]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the bill acceptor unit shall be turned on or off or if it shall flash.
<b>Event</b>	none

**guidlightPorts[JXFS\_SIU\_ENVDISPENSER]**

<b>Type</b>	<i>JxfsSiuGuidLightPorts</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies whether the Guidance Light Indicator on the envelope dispenser unit shall be turned on or off or if it shall flash.
<b>Event</b>	none



## 8.21 JxfsSiuSetDoor

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

### 8.21.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
doorPort	<b>JxfsSiuDoorPort</b>	R	
doorIndex	<b>int</b>	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetDoor	doorPort	JxfsSiuDoorPort
	doorIndex	int

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

Event	May occur after
none	

### 8.21.2 Properties

#### doorPort (R)

<b>Type</b>	<i>JxfsSiuDoorPort</i>
<b>Description</b>	Specifies the state the door shall be changed to.
<b>Event</b>	none

#### doorIndex (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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

## 8.22 JxfsSiuSetIndicator

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

### 8.22.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
indicatorPort	<b>JxfsSiuDoorPort</b>	R	
indicatorIndex	<b>Int</b>	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetIndicator	indicatorPort	JxfsSiuIndicatorPort
	indicatorIndex	int

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

Event	May occur after
None	

### 8.22.2 Properties

#### indicatorPort (R)

<b>Type</b>	<i>JxfsSiuIndicatorPort</i>
<b>Description</b>	Specifies the state the indicator shall be changed to.
<b>Event</b>	None

#### indicatorIndex (R)

<b>Type</b>	<i>Int</i>
<b>Initial Value</b>	None
<b>Description</b>	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.

## 8.23 JxfsSiuSetAuxiliary

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

### 8.23.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
AuxiliaryPort	<b>JxfsSiuAuxiliaryPort</b>	R	
auxiliaryIndex	<b>Int</b>	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetAuxiliary	auxiliaryPort	JxfsSiuAuxiliaryPort
	auxiliaryIndex	int

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

Event	May occur after
None	

### 8.23.2 Properties

#### auxiliaryPort (R)

<b>Type</b>	<i>JxfsSiuAuxiliaryPort</i>
<b>Description</b>	Specifies the state the auxiliary indicator shall be changed to.
<b>Event</b>	None

#### auxiliaryIndex (R)

<b>Type</b>	<i>Int</i>
<b>Initial Value</b>	None
<b>Description</b>	Specifies the door 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.

## 8.24 JxfsSiuSetGuidLight

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

### 8.24.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
guidLightPort	<b>JxfsSiuGuidLightPort</b>	R	
guidLightIndex	<b>Int</b>	R	

Constructor	Parameter	Parameter-Type
JxfsSiuSetGuidLight	guidLightPort	JxfsSiuGuidLightPort
	guidLightIndex	int

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

Event	May occur after
none	

### 8.24.2 Properties

#### guidLightPort (R)

<b>Type</b>	<i>JxfsSiuGuidLightPort</i>
<b>Description</b>	Specifies the state the guidance light shall be changed to.
<b>Event</b>	none

#### guidLightIndex (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	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.

## 8.25 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.

### 8.25.1 Summary

Implements : *Serializable*

Extends : *JxfsStatus*

Property	Type	Access	Initialized after
port	<b>JxfsSiuPortStatus</b>	R	
index	<b>int</b>	R	

Constructor	Parameter	Parameter-Type
JxfsSiuPortChangeStatus	port	JxfsSiuPortStatus
	index	int

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

Event	May occur after
none	

### 8.25.2 Properties

#### port (W)

<b>Type</b>	<i>JxfsSiuPortStatus</i>
<b>Description</b>	Specifies the state the port has changed to.
<b>Event</b>	none

#### index (W)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	
<b>Description</b>	Specifies the index of the port that changed its state.

Value	Meaning
JXFS_SIU_OPERATORSWITCH	The Operator Switch has changed its state.
H	
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_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_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.

## 8.26 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 port.

### 8.26.1 Summary

**Implements :** *Serializable*

**Extends :** *JxfsStatus*

Property	Type	Access	Initialized after
port	<b>JxfsSiuPortStatus</b>	R	
index	<b>int</b>	R	
portError	<b>int</b>	R	

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

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

Event	May occur after
none	

### 8.26.2 Properties

#### port (R)

<b>Type</b>	<i>JxfsSiuPortStatus</i>
<b>Description</b>	Specifies the state the new state of the port.
<b>Event</b>	none

#### index (R)

<b>Type</b>	<i>int</i>
<b>Initial Value</b>	none
<b>Description</b>	Specifies the index of the port that has changed its state.

Value	Meaning
JXFS_SIU_OPERATORSWITCH	The Operator Switch has changed its state.
H	
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_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_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.

**portError (R)**

**Type**  
**Description**

*int*  
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.

**Event**

none

## 9 Codes

### 9.1 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.

### 9.2 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 JxfsSiuPortChangeStatus objects
JXFS_S_SIU_PORT_ERROR	There was an error when accessing a port. Specific information about this error is contained in a JxfsSiuPortError object.







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_CONTINUOUS
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_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_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

## 10 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.



## Index

auxiliaryCapabilities[JXFS_SIU_MONITOR] .....	54	guidLightCapabilities[JXFS_SIU_PASSBOOKPRINTER] .....	55
auxiliaryCapabilities[JXFS_SIU_POWEROFF] .....	54	guidLightCapabilities[JXFS_SIU_PINPAD] .....	55
auxiliaryCapabilities[JXFS_SIU_RELAY1] .....	54	guidLightCapabilities[JXFS_SIU_RECEIPTPRINTER] .....	55
auxiliaryCapabilities[JXFS_SIU_RELAY2] .....	54	guidLightCapability .....	49
auxiliaryCapabilities[JXFS_SIU_RELAY3] .....	54	guidlightEnable[] .....	59
auxiliaryCapabilities[JXFS_SIU_RELAY4] .....	55	guidLightIndex .....	74
auxiliaryCapabilities[JXFS_SIU_UPS] .....	54	guidLightPort .....	74
auxiliaryCapabilities[JXFS_SIU_VOLUME] .....	54	guidlightPorts[JXFS_SIU_CHEQUEUNIT] .....	70
auxiliaryCapability .....	47	guidlightPorts[JXFS_SIU_ENVDEPOSITORY] .....	69
auxiliaryEnable[] .....	59	guidlightPorts[JXFS_SIU_ENVDISPENSER] .....	70
auxiliaryIndex .....	73	guidlightPorts[JXFS_SIU_PASSBOOKPRINTER] .....	69
auxiliaryPort .....	73	guidlightPorts[JXFS_SIU_RECEIPTPRINTER] .....	69
auxiliaryStatus .....	26	guidlightPorts[JXFS_SIU_BILLACCEPTOR] .....	70
capabilities .....	14	guidlightPorts[JXFS_SIU_CARDUNIT] .....	69
doorCapabilities[JXFS_SIU_CABINET] .....	52	guidlightPorts[JXFS_SIU_COINDISPENSER] .....	69
doorCapabilities[JXFS_SIU_FRONT_BOTTOM] .....	53	guidlightPorts[JXFS_SIU_NOTESDISPENSER] .....	69
doorCapabilities[JXFS_SIU_FRONT_TOP] .....	53	guidlightPorts[JXFS_SIU_PINPAD] .....	69
doorCapabilities[JXFS_SIU_REAR_BOTTOM] .....	53	guidlightStatus .....	28
doorCapabilities[JXFS_SIU_REAR_TOP] .....	53	guidlightStatus[JXFS_SIU_CHEQUEUNIT] .....	37
doorCapabilities[JXFS_SIU_SAFE] .....	52	guidlightStatus[JXFS_SIU_ENVDEPOSITORY] .....	37
doorCapabilities[JXFS_SIU_VANDALSHIELD] .....	53	guidlightStatus[JXFS_SIU_ENVDISPENSER] .....	38
doorCapability .....	42	guidlightStatus[JXFS_SIU_NOTESDISPENSER] .....	36
doorEnable[] .....	59	guidlightStatus[JXFS_SIU_PASSBOOKPRINTER] .....	37
doorIndex .....	71	guidlightStatus[JXFS_SIU_RECEIPTPRINTER] .....	37
doorPort .....	71	guidlightStatus[JXFS_SIU_BILLACCEPTOR] .....	38
doorPorts [JXFS_SIU_FRONT_TOP] .....	66	guidlightStatus[JXFS_SIU_CARDUNIT] .....	36
doorPorts [JXFS_SIU_SAFE] .....	66	guidlightStatus[JXFS_SIU_COINDISPENSER] .....	37
doorPorts [JXFS_SIU_VANDALSHIELD] .....	66	guidlightStatus[JXFS_SIU_PINPAD] .....	36
doorPorts[JXFS_SIU_CABINET] .....	66	IJxfsSiu .....	14
doorPorts[JXFS_SIU_FRONT_BOTTOM] .....	67	index .....	75, 78
doorPorts[JXFS_SIU_REAR_BOTTOM] .....	67	indicatorCapabilities[JXFS_SIU_AUDIO] .....	53
doorPorts[JXFS_SIU_REAR_TOP] .....	67	indicatorCapabilities[JXFS_SIU_FASCIALIGHT] .....	53
doorStatus .....	22	indicatorCapabilities[JXFS_SIU_HEATING] .....	54
doorStatus[JXFS_SIU_CABINET] .....	32	indicatorCapabilities[JXFS_SIU_LOGOLIGHT] .....	54
doorStatus[JXFS_SIU_FRONT_BOTTOM] .....	33	indicatorCapabilities[JXFS_SIU_OPENCLOSE] .....	53
doorStatus[JXFS_SIU_FRONT_TOP] .....	33	indicatorCapability .....	45
doorStatus[JXFS_SIU_REAR_BOTTOM] .....	33	indicatorEnable[] .....	59
doorStatus[JXFS_SIU_REAR_TOP] .....	33	indicatorIndex .....	72
doorStatus[JXFS_SIU_SAFE] .....	32	indicatorPort .....	72
doorStatus[JXFS_SIU_VANDALSHIELD] .....	33	indicatorPorts[JXFS_SIU_AUDIO] .....	67
enable .....	57	indicatorPorts[JXFS_SIU_FASCIALIGHT] .....	67
enableEvents .....	15	indicatorPorts[JXFS_SIU_HEATING] .....	67
getProperty .....	13	indicatorPorts[JXFS_SIU_LOGOLIGHT] .....	67
guidLightCapabilities[JXFS_SIU_BILLACCEPTOR] .....	56	indicatorPorts[JXFS_SIU_MONITOR] .....	68
guidLightCapabilities[JXFS_SIU_CARDUNIT] .....	55	indicatorPorts[JXFS_SIU_OPENCLOSE] .....	67
guidLightCapabilities[JXFS_SIU_CHEQUEUNIT] .....	56	indicatorPorts[JXFS_SIU_POWEROFF] .....	68
guidLightCapabilities[JXFS_SIU_COINDISPENSER] .....	55	indicatorPorts[JXFS_SIU_RELAY1] .....	68
guidLightCapabilities[JXFS_SIU_ENVDEPOSITORY] .....	55	indicatorPorts[JXFS_SIU_RELAY2] .....	68
guidLightCapabilities[JXFS_SIU_ENVDISPENSER] .....	56	indicatorPorts[JXFS_SIU_RELAY3] .....	68
guidLightCapabilities[JXFS_SIU_NOTESDISPENSER] .....	55	indicatorPorts[JXFS_SIU_RELAY4] .....	68

indicatorPorts[JXFS_SIU_UPS] .....	68	JxfsSiuIndicatorCapability.....	45	
indicatorPorts[JXFS_SIU_VOLUME].....	68	JxfsSiuIndicatorPort .....	61	
indicatorStatus .....	24	JxfsSiuIndicatorStatus .....	24	
indicatorStatus[JXFS_SIU_AUDIO] .....	34	JxfsSiuPortChangeStatus.....	75	
indicatorStatus[JXFS_SIU_FASCIALIGHT] .....	34	JxfsSiuPortError .....	78	
indicatorStatus[JXFS_SIU_HEATING] .....	34	JxfsSiuPortStatus .....	18	
indicatorStatus[JXFS_SIU_LOGOLIGHT] .....	34	JxfsSiuSensorCapability .....	39	
indicatorStatus[JXFS_SIU_MONITOR].....	35	JxfsSiuSensorStatus .....	19	
indicatorStatus[JXFS_SIU_OPENCLOSE] .....	33	JxfsSiuSetAuxiliary .....	73	
indicatorStatus[JXFS_SIU_POWEROFF].....	35	JxfsSiuSetDoor .....	71	
indicatorStatus[JXFS_SIU_RELAY1].....	35	JxfsSiuSetGuidLight.....	74	
indicatorStatus[JXFS_SIU_RELAY2].....	35	JxfsSiuSetIndicator .....	72	
indicatorStatus[JXFS_SIU_RELAY3].....	36	JxfsSiuSetPorts .....	66	
indicatorStatus[JXFS_SIU_RELAY4].....	36	JxfsSiuStatus.....	29	
indicatorStatus[JXFS_SIU_UPS].....	35	port.....	75, 78	
indicatorStatus[JXFS_SIU_VOLUME] .....	34	portError .....	80	
isAjarSupported.....	44	sensorCapabilities[JXFS_SIU_AMBLIGHT] .....	51	
isAvailable.....	40, 45, 48, 49	sensorCapabilities[JXFS_SIU_BOOTSWITCH] .....	52	
isBoltedSupported .....	43	sensorCapabilities[JXFS_SIU_HEAT] .....	51	
isClosedSupported.....	43	sensorCapabilities[JXFS_SIU_INPUT1] .....	51	
isEngagedSupported.....	48	sensorCapabilities[JXFS_SIU_INPUT2] .....	52	
isJammedSupported .....	44	sensorCapabilities[JXFS_SIU_INPUT3] .....	52	
isKeyboardSupported .....	43	sensorCapabilities[JXFS_SIU_INPUT4] .....	52	
isLockedSupported.....	43	sensorCapabilities[JXFS_SIU_INTTAMPER] ...	51	
isLowSupported .....	48	sensorCapabilities[JXFS_SIU_OPERATORSWIT	CH] .....	51
isMaintenanceModeSupported .....	40	sensorCapabilities[JXFS_SIU_PROXIMITY]....	51	
isOpenSupported .....	43	sensorCapabilities[JXFS_SIU_SEISMIC] .....	51	
isPoweringSupported .....	48	sensorCapabilities[JXFS_SIU_TAMPER].....	51	
isRecoveredSupported.....	48	sensorCapabilities[JXFS_SIU_VENTILATOR] .	52	
isRunModeSupported.....	40	sensorCapability .....	39	
isServiceSupported.....	43	sensorEnable[] .....	59	
isSupervisorModeSupported .....	40	sensorStatus .....	19	
JXFS_E_SIU_INVALID_PORT .....	81	sensorStatus[JXFS_SIU_AMBLIGHT] .....	31	
JXFS_E_SIU_PORT_ERROR.....	81	sensorStatus[JXFS_SIU_BOOTSWITCH] .....	32	
JXFS_E_SIU_SYNTAX.....	81	sensorStatus[JXFS_SIU_HEAT].....	30	
JXFS_S_SIU_PORT_ERROR.....	81	sensorStatus[JXFS_SIU_INPUT1].....	31	
JXFS_S_SIU_PORT_STATUS .....	81	sensorStatus[JXFS_SIU_INPUT2].....	31	
JxfsSiuAuxiliaryCapability .....	46	sensorStatus[JXFS_SIU_INPUT3].....	31	
JxfsSiuAuxiliaryPort .....	63	sensorStatus[JXFS_SIU_INPUT4].....	32	
JxfsSiuAuxiliaryStatus .....	26	sensorStatus[JXFS_SIU_INTTAMPER].....	30	
JxfsSiuCapabilities .....	50	sensorStatus[JXFS_SIU_OPERATORSWITCH]30		
JxfsSiuDoorCapability .....	41	sensorStatus[JXFS_SIU_PROXIMITY] .....	31	
JxfsSiuDoorPort .....	60	sensorStatus[JXFS_SIU_SEISMIC].....	30	
JxfsSiuDoorStatus .....	22	sensorStatus[JXFS_SIU_TAMPER] .....	30	
JxfsSiuEnable .....	57	sensorStatus[JXFS_SIU_VENTILATOR] .....	32	
JxfsSiuEnableEvents .....	58	setPorts .....	15, 16, 17	
JxfsSiuGuidLightCapability .....	49	setProperty.....	13	
JxfsSiuGuidLightPort.....	65	state.....	60, 61, 64, 65	
JxfsSiuGuidLightStatus.....	28			