

CEN

WORKSHOP

AGREEMENT

CWA 16926-36

February 2017

35.240.40

English version

**Extensions for Financial Services (XFS) interface specification –
Release 3.30 – Part 36: XFS MIB Device Specific Definitions –
Text Terminal Unit Device Class MIB 3.30**

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

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

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

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

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



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

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

© 2017 CEN All rights of exploitation in any form and by any means reserved worldwide for CEN national Members.
Ref. No. CWA 16926-36:2017 D/E/F

Table of Contents

Foreword	3
1. Introduction	7
2. XFS TTU MIB variables	10
2.1 XFS TTU STATUS TABLE	10
2.1.1 <i>xfsTTUStatusTable: States</i>	10
2.2 XFS TTU SUB DEVICE TABLE	13
2.2.1 <i>xfsTTUSubDeviceTable: States</i>	13
2.3 XFS TTU ERROR TABLE	14
2.4 XFS TTU RESET TABLE	14
2.5 XFS TTU RESET DEVICE TABLE	15
2.6 XFS TTU CAPABILITIES TABLE	15
2.6.1 <i>xfsTTUCapabilitiesTable: Capabilities</i>	16
3. TTU Traps	19
3.1 TTU DETAILED DEVICE STATUS CHANGE TRAP	19
3.1.1 <i>TTU Detailed Device Status Change Trap Format</i>	19
3.1.2 <i>TTU Detailed Device Status Change Trap: an example</i>	22
3.2 TTU SUB-DEVICE STATUS CHANGE TRAP	23
3.3 TTU RESET DEVICE COMPLETE TRAP	24
3.3.1 <i>TTU Reset Device Complete Trap Format</i>	24
3.3.2 <i>TTU Reset Device Complete: an example</i>	26
4. Appendix A - TTU MIB sub-tree	29
4.1 TTU MIB IN SMIV2 AND SMIV1 FORMAT	29
5. Appendix B - C-Header files	40
5.1 XFSMIBTTU.H	40

Foreword

This CWA is revision 3.30 of the XFS interface specification.

This CEN Workshop Agreement has been drafted and approved by a Workshop of representatives of interested parties on March 19th 2015, the constitution of which was supported by CEN following the public call for participation made on 1998-06-24. The specification is continuously reviewed and commented in the CEN/ISSS Workshop on XFS. It is therefore expected that an update of the specification will be published in due time as a CWA, superseding this revision 3.30.

A list of the individuals and organizations which supported the technical consensus represented by the CEN Workshop Agreement is available from the CEN/XFS Secretariat. The CEN XFS Workshop gathered suppliers as well as banks and other financial service companies.

The CWA is published as a multi-part document, consisting of:

Part 1: Application Programming Interface (API) - Service Provider Interface (SPI) - Programmer's Reference

Part 2: Service Classes Definition - Programmer's Reference

Part 3: Printer and Scanning Device Class Interface - Programmer's Reference

Part 4: Identification Card Device Class Interface - Programmer's Reference

Part 5: Cash Dispenser Device Class Interface - Programmer's Reference

Part 6: PIN Keypad Device Class Interface - Programmer's Reference

Part 7: Check Reader/Scanner Device Class Interface - Programmer's Reference

Part 8: Depository Device Class Interface - Programmer's Reference

Part 9: Text Terminal Unit Device Class Interface - Programmer's Reference

Part 10: Sensors and Indicators Unit Device Class Interface - Programmer's Reference

Part 11: Vendor Dependent Mode Device Class Interface - Programmer's Reference

Part 12: Camera Device Class Interface - Programmer's Reference

Part 13: Alarm Device Class Interface - Programmer's Reference

Part 14: Card Embossing Unit Device Class Interface - Programmer's Reference

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

Part 16: Card Dispenser Device Class Interface - Programmer's Reference

Part 17: Barcode Reader Device Class Interface - Programmer's Reference

Part 18: Item Processing Module Device Class Interface - Programmer's Reference

Parts 19 - 28: Reserved for future use.

Parts 29 through 47 constitute an optional addendum to this CWA. They define the integration between the SNMP standard and the set of status and statistical information exported by the Service Providers.

Part 29: XFS MIB Architecture and SNMP Extensions - Programmer's Reference

Part 30: XFS MIB Device Specific Definitions - Printer Device Class

Part 31: XFS MIB Device Specific Definitions - Identification Card Device Class

Part 32: XFS MIB Device Specific Definitions - Cash Dispenser Device Class

Part 33: XFS MIB Device Specific Definitions - PIN Keypad Device Class

Part 34: XFS MIB Device Specific Definitions - Check Reader/Scanner Device Class

Part 35: XFS MIB Device Specific Definitions - Depository Device Class

Part 36: XFS MIB Device Specific Definitions - Text Terminal Unit Device Class

Part 37: XFS MIB Device Specific Definitions - Sensors and Indicators Unit Device Class

Part 38: XFS MIB Device Specific Definitions - Camera Device Class

Part 39: XFS MIB Device Specific Definitions - Alarm Device Class

Part 40: XFS MIB Device Specific Definitions - Card Embossing Unit Class

Part 41: XFS MIB Device Specific Definitions - Cash-In Module Device Class

Part 42: Reserved for future use.

Part 43: XFS MIB Device Specific Definitions - Vendor Dependent Mode Device Class

Part 44: XFS MIB Application Management

Part 45: XFS MIB Device Specific Definitions - Card Dispenser Device Class

Part 46: XFS MIB Device Specific Definitions - Barcode Reader Device Class

Part 47: XFS MIB Device Specific Definitions - Item Processing Module Device Class

Parts 48 - 60 are reserved for future use.

Part 61: Application Programming Interface (API) - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Service Provider Interface (SPI) - Programmer's Reference

Part 62: Printer and Scanning Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 63: Identification Card Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 64: Cash Dispenser Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 65: PIN Keypad Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 66: Check Reader/Scanner Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 67: Depository Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 68: Text Terminal Unit Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 69: Sensors and Indicators Unit Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 70: Vendor Dependent Mode Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 71: Camera Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 72: Alarm Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 73: Card Embossing Unit Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 74: Cash-In Module Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 75: Card Dispenser Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 76: Barcode Reader Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

Part 77: Item Processing Module Device Class Interface - Migration from Version 3.20 (CWA 16374) to Version 3.30 (this CWA) - Programmer's Reference

In addition to these Programmer's Reference specifications, the reader of this CWA is also referred to a complementary document, called Release Notes. The Release Notes contain clarifications and explanations on the CWA specifications, which are not requiring functional changes. The current version of the Release Notes is

available online from <http://www.cen.eu/work/areas/ict/ebusiness/pages/ws-xfs.aspx>.

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

The formal process followed by the Workshop in the development of the CEN Workshop Agreement has been endorsed by the National Members of CEN but neither the National Members of CEN nor the CEN-CENELEC Management Centre can be held accountable for the technical content of the CEN Workshop Agreement or possible conflict with standards or legislation. This CEN Workshop Agreement can in no way be held as being an official standard developed by CEN and its members.

The final review/endorsement round for this CWA was started on 2015-01-16 and was successfully closed on 2015-03-19. The final text of this CWA was submitted to CEN for publication on 2015-06-19. The specification is continuously reviewed and commented in the CEN Workshop on XFS. It is therefore expected that an update of the specification will be published in due time as a CWA, superseding this revision 3.30.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

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

Revision History:

1.0	January 20, 2004	Initial release of XFS MIB specification.
1.10	April 15, 2007	Update of the MIB to add support for a Detailed Status Trap, a Device Reset capability and the support of SMIV2.
3.10	December 14, 2010	Update of the MIB to add support for a Capabilities table and to align the MIB with XFS 3.10.
3.20	March 28, 2014	Update release to align the MIB with XFS 3.20.
3.30	February 6, 2017	Update release to align the MIB with XFS 3.30.

1. Introduction

This document provides the device specific MIB definition (Management Information Base) variables for the xfsTTU sub-tree version one, as foreseen by the *XFS MIB Architecture and SNMP Extensions Programmer's Reference* document. All the attributes in all the MIBs are Mandatory. In the case where a vendor's device does not support an attribute then a request for this unsupported attribute should return NULL.

The xfsTTU version one sub-tree is identified by:

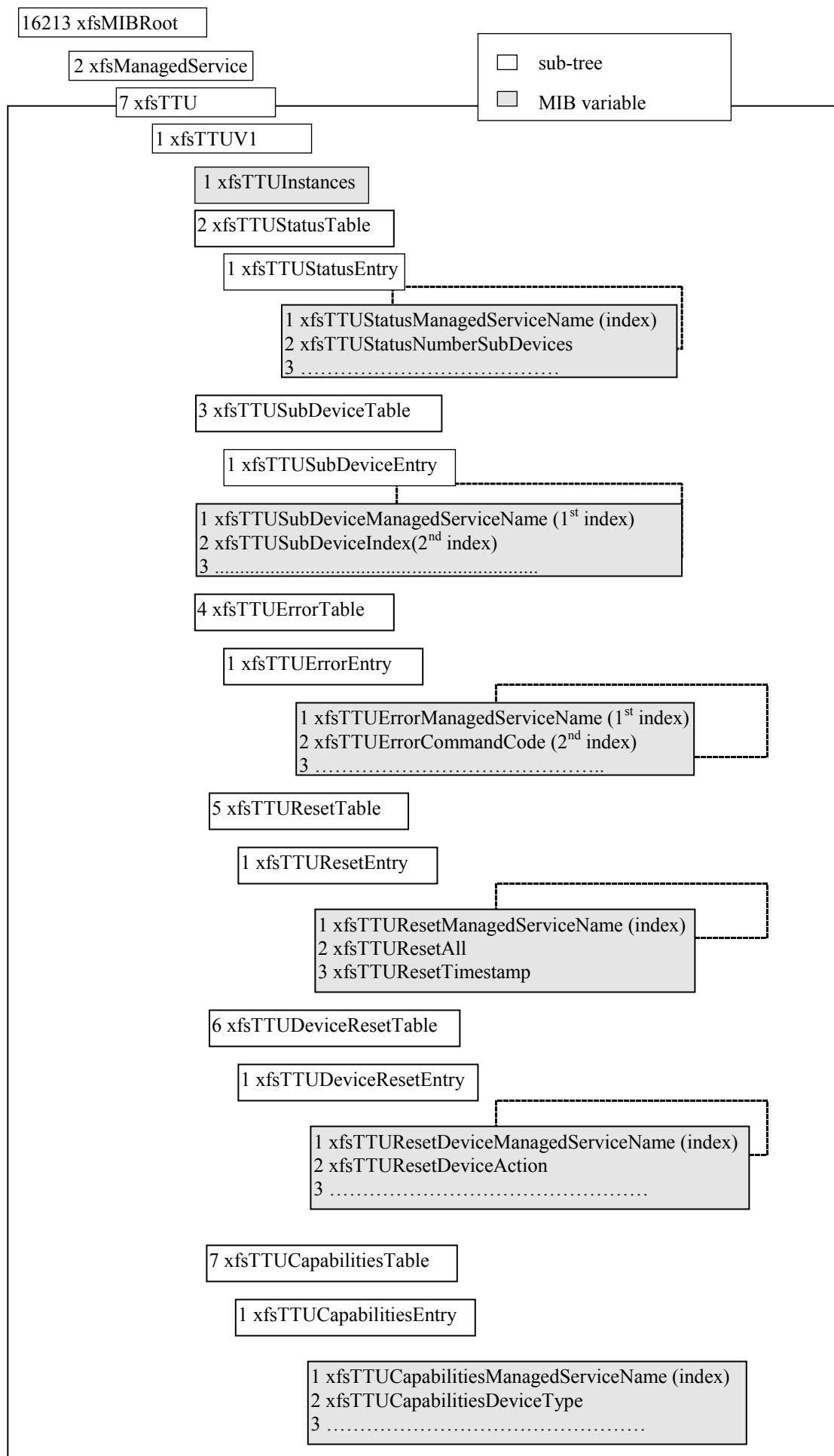
xfsMIBRoot

- xfsManagedService (2)
 - xfsTTU (7)
 - xfsTTUV1 (1)

The xfsTTUV1 sub-tree contains the following variables:

- * *xfsTTUInstances(1)* is the number of physical devices for the TTU class installed on the XFS subsystem. It is a 32 bit numerical field.
- * *xfsTTUStatusTable(2)* identifies the table for the TTU variables.
- * *xfsTTUSubDeviceTable(3)* not applicable to the TTU device.
- * *xfsTTUErrorTable(4)* identifies the table for the TTU error counters.
- * *xfsTTUResetTable(5)* identifies the table for the TTU reset variable.
- * *xfsTTUResetDeviceTable(6)* identifies the table for the TTU reset device variables.
- * *xfsTTUCapabilitiesTable(7)* identifies the table for the TTU capabilities variables.

The *XFS MIB Architecture and SNMP Extensions Programmer's Reference* document provides an overview of the MIB structure. The following picture shows the structure of the *xfsTTUV1* sub-tree.



Section 2 describes how the Status, Sub-Device, Error, Reset, Reset Device and Capabilities tables apply to the TTU device class.

2. XFS TTU MIB variables

This section describes the MIB variables for the tables of the TTU Class. All variables are 32-bit numerical fields. The description of the variables listed below includes, where it is meaningful, a reference to relevant data structures and commands defined inside the *Text Terminal Unit Device Class Interface Programmer's Reference*.

The following are some general notes pertaining to the MIB variables:

- All command response counters maintained by the Service Provider are persistent across re-boots.
- One application command may trigger only one command-related counter to be updated.
- One application command may trigger one or multiple status variables to be updated.
- All command response counters are read-writable unless otherwise specified.
- Each managed service has a Reset table that allows all the response counters to be reset.
- Each managed service has a Reset Device table that allows the WFS_CMD_TTU_RESET command to be executed from the management station.

2.1 XFS TTU Status Table

The *xfstTUStatusTable(2)* groups the variables identifying device status information, statistics and additional variables. It is indexed through a single parameter, *xfstTUStatusManagedServiceName*. All device status variables are read-only.

Additional variables can be used to contain vendor-dependent variables. These variables do not start immediately after the standard variables in order to allow for expansion of the standard variables, the first additional variable can be added at position 1000.

xfstTUStatusManagedServiceName is the instance identifier of the managed service and uniquely identifies one instance of the TTU class.

As an example, the identifier for the device status value of *xfstTUStatusKeyboard(4)* for a device with managed service name equal to "TextTerminal1" is as follows:

Character	T	e	x	t	T	e	r	m	i	n	a	l	l
ASCII Hex	54	65	78	74	54	65	72	6D	69	6E	61	6C	31
ASCII Dec	84	101	120	116	84	101	114	109	105	110	97	108	49

NOTE SNMP OID representation of strings consists of a length field specifying the number of characters in the string followed by the ASCII code in decimal for each character in the string. Therefore the OID of the above example is:

xfstMIBRoot.2.7.1.2.1.4.13.84.101.120.116.84.101.114.109.105.110.97.108.49

2.1.1 xfstTUStatusTable: States

The first three status variables are common across all device classes, the other variables are device class specific.

xfstTUStatusManagedServiceName (1)
Uniquely identifies the managed service.

xfstTUStatusNumberSubDevices (2)
Defines how many sub-devices the service has. This is always 0 (zero) in the TTU device class.

xfstTUStatusDevice (3)

It contains the state of the text terminal unit. It is a numeric type field. Allowed values are:

Value	Meaning
xfsDevOnline(1)	The device is online (i.e., powered on and operable).
xfsDevOffline(2)	The device is offline (e.g., the operator has taken the device offline by turning a switch).
xfsDevPowerOff(3)	The device is powered off or physically not connected.
xfsDevNoDevice(4)	There is no device intended to be there; e.g. this type of self service machine does not contain such a device or it is internally not configured.
xfsDevHWEError(5)	The device is inoperable due to a hardware error.
xfsDevUserError(6)	The device is inoperable because a person is preventing proper device operation.
xfsDevBusy(7)	The device is busy and unable to process an execute command at this time.
xfsDevFraudAttempt(8)	The device is present but is inoperable because it has detected a fraud attempt.
xfsDevPotentialFraud(9)	The device is present but has detected a potential fraud attempt and is capable of remaining in service.

xfsTTUStatusKeyboard (4)

It contains the state of the keyboard in the text terminal unit. It is a numeric type field. Allowed values are as follows:

Value	Meaning
xfsTTUKeyboardNotAvailable(1)	The keyboard is not available..
xfsTTUKeyboardOn(2)	The keyboard is activated.
xfsTTUKeyboardOff(3)	The keyboard is not activated.

xfsTTUStatusKeyLock (5)

It contains the state of the keyboard lock of the text terminal unit. It is a numeric type field. Allowed values are as follows:

Value	Meaning
xfsTTUKeyboardLockNotAvailable(1)	The keyboard lock state is not available.
xfsTTUKeyboardLockOn(2)	The keyboard lock switch is activated.
xfsTTUKeyboardLockOff(3)	The keyboard lock switch is not activated.

xfsTTUStatusLed1 (6)

It contains the state of LED 1. It is a numeric type field. Allowed values are as follows:

Value	Meaning
xfsTTULedNotAvailable(1)	The led state cannot be reported.
xfsTTULedOff(2)	The led is turned off.
xfsTTULedSlowFlash(3)	The led is blinking slowly.
xfsTTULedMediumFlash(5)	The led is blinking with medium frequency.
xfsTTULedQuickFlash(9)	The led is blinking quickly.
xfsTTULedContinuous(129)	The led is turned on continuous (steady).

The following *xfsTTUStatusLedx* fields can also be one of the above values. The maximum number of LEDs is defined by the WFS_TTU_LEDS_MAX value.

xfsTTUStatusLed2 (7)

It contains the state of LED 2. It is a numeric type field. Allowed values are as those for *xfsTTUStatusLed1*.

xfsTTUStatusLed3 (8)

It contains the state of LED 3. It is a numeric type field. Allowed values are as those for *xfsTTUStatusLed1*.

xfsTTUStatusLed4 (9)

It contains the state of LED 4. It is a numeric type field. Allowed values are as those for *xfsTTUStatusLed1*.

xfsTTUStatusLed5 (10)

It contains the state of LED 5. It is a numeric type field. Allowed values are as those for *xfsTTUStatusLed1*.

xfsTTUStatusLed6 (11)

It contains the state of LED 6. It is a numeric type field. Allowed values are as those for *xfsTTUStatusLed1*.

xfsTTUStatusLed7 (12)

It contains the state of LED 7. It is a numeric type field. Allowed values are as those for *xfsTTUStatusLed1*.

xfsTTUStatusLed8 (13)

It contains the state of LED 8. It is a numeric type field. Allowed values are as those for *xfsTTUStatusLed1*.

xfsTTUStatusDisplaySizeX (14)

It contains the horizontal display size (the number of columns that can be displayed). It is a numeric type field.

xfsTTUStatusDisplaySizeY (15)

It contains the vertical display size (the number of rows that can be displayed). It is a numeric type field.

xfsTTUStatusDevicePosition (16)

It contains the device position. It is a numeric type field. Allowed values are as follows:

Value	Meaning
<i>xfsTTUDeviceInPosition(1)</i>	The device is in its normal operating position, or is fixed in place and cannot be moved.
<i>xfsTTUDeviceNotInPosition(2)</i>	The device has been removed from its normal operating position.
<i>xfsTTUDevicePosUnknown(3)</i>	Due to a hardware error or other condition, the position of the device cannot be determined.
<i>xfsTTUDevicePosNotSupported(4)</i>	The physical device does not have the capability of detecting the position.

xfsTTUStatusPowerSaveRecoveryTime (17)

It contains the actual number of seconds required by the device to resume its normal operational state from the current power saving mode. This value is zero if either the power saving mode has not been activated or no power save control is supported. It is a numeric type field

xfsTTUStatusAntiFraudModule (18)

It contains the state of the anti-fraud module. Allowed values are as follows:

Value	Meaning
<i>xfsTTUAFMNotSupported(1)</i>	No anti-fraud module is available.
<i>xfsTTUAFMOK(2)</i>	Anti-fraud module is in a good state and no foreign device is detected.
<i>xfsTTUAFMInop(3)</i>	Anti-fraud module is inoperable.
<i>xfsTTUAFMDeviceDetected(4)</i>	Anti-fraud module detected the presence of a foreign device.
<i>xfsTTUAFMUnknown(5)</i>	The state of the anti-fraud module cannot be determined.

xfsTTUStatusExtraStatus (100)

It contains a list of vendor-specific, or any other extended, information as an OCTET STRING. The information is returned as a series of "key=value" strings. Each string is null-terminated, with the final string terminating with two null characters. An empty list is indicated by two consecutive null characters.

2.2 XFS TTU Sub Device Table

The *xfsTTUSubDeviceTable(3)* groups the variables identifying information for the extended LEDs. It is indexed through two parameters, *xfsTTUSubDeviceManagedServiceName* and *xfsTTUSubDeviceIndex*. All sub-device status variables are read-only and, if maintained by the SP, persist across re-boots.

xfsTTUSubDeviceManagedServiceName is the instance identifier of the managed service and uniquely identifies one instance of the TTU class. In XFS this information comes from the WFS_INF_TTU_STATUS command.

As an example, the identifier for the sub-device status value of *xfsTTUSubDeviceLED(3)* for a sub-device index 1 with managed service name equal to “TextTerminal1” is as follows:

Character	T	e	x	t	T	e	r	m	i	n	a	l	l
ASCII Hex	54	65	78	74	54	65	72	6D	69	6E	61	6C	31
ASCII Dec	84	101	120	116	84	101	114	109	105	110	97	108	49

NOTE SNMP OID representation of strings consists of a length field specifying the number of characters in the string followed by the ASCII code in decimal for each character in the string followed by the sub-device. Therefore the OID of the above example is:

xfsMIBRoot.2.7.1.3.1.3.8.84.101.120.116.84.101.114.109.105.110.97.108.49.1

2.2.1 xfsTTUSubDeviceTable: States

The first two variables are common across all sub-device classes, the other variables are sub-device class specific.

xfsTTUSubDeviceManagedServiceName (1)

Uniquely identifies the managed service.

xfsTTUSubDeviceIndex (2)

Index into the table of sub-devices supported. This is an index (starting from 1) into the TTU Sub-Device table.

xfsTTUSubDeviceLED (3)

It contains the state of the LED.

Value	Meaning	
Value	XFS Name	Meaning
0x00000000	WFS_TTU_LEDNA	The status is not available.
0x00000001	WFS_TTU_LED OFF	The LED is turned off.
0x00000002	WFS_TTU_LEDSLOWFLASH	The LED is blinking slowly.
0x00000004	WFS_TTU_LED MEDIUMFLASH	The LED is blinking medium frequency.
0x00000008	WFS_TTU_LEDQUICKFLASH	The LED is blinking quickly.
0x00000080	WFS_TTU_LED CONTINUOUS	The LED is turned on continuous (steady).
0x00000100	WFS_TTU_LEDRED	The LED is red.
0x00000200	WFS_TTU_LEDGREEN	The LED is green.
0x00000400	WFS_TTU_LEDYELLOW	The LED is yellow.
0x00000800	WFS_TTU_LEDBLUE	The LED is blue.
0x00001000	WFS_TTU_LEDCYAN	The LED is cyan.
0x00002000	WFS_TTU_LED MAGENTA	The LED is magenta.
0x00004000	WFS_TTU_LEDWHITE	The LED is white.

2.3 XFS TTU Error Table

The *xfsTTUErrorTable(4)* provides access to all command response counters supported by a device class. The error table contains the set of counters for every combination of executable command and associated response that the Service Provider supports. The counters report the number of times that a response has been returned from a particular command since the counts were last reset. Selection of the required counter is made by specifying the managed service name, command code and response code through the following parameters:

xfsTTUErrorManagedServiceName
xfsTTUErrorCommandCode
xfsTTUErrorResponseCode

The *xfsTTUErrorTable* is defined as:

- *xfsTTUErrorManagedServiceName(1)* which provides the primary index to the service in question. It is Display String field. The *xfsTTUErrorManagedServiceName* parameter corresponds to the value of *xfsMIBRoot.xfsGeneral.xfsMIBV1.xfsManagedServiceTable.xfsManagedServiceEntry.xfsManagedServiceName* in the general table, e.g. "TextTerminal1".
- *xfsTTUErrorCommandCode(2)* is an index which identifies the command code that that response code is related to. It is a 32 bit numerical field.
- *xfsTTUErrorResponseCode(3)* is an index which identifies the response code that the count is required for. It is the absolute value of the error code. It is a 32 bit numerical field.
- *xfsTTUErrorCount(4)* is the count of the number of times that a particular response code has been generated while executing a specific command, since they were last reset. It is a 32 bit numerical field.

All counter variables are read-write. Issue of a Set command on a specific counter with value *x* will result in the individual counter being set to value *x*.

As an example, the identifier for the error count value for the WFS_ERR_TTU_INVALIDLED (-707) error returned from the WFS_CMD_TTU_SET_LED (704) command for a device with managed service name equal to "TextTerminal1" is as follows:

xfsMIBRoot.2.7.1.4.1.4.13.84.101.120.116.84.101.114.109.105.110.97.108.49.704.707

2.4 XFS TTU Reset Table

The *xfsTTUResetTable(5)* contains the *xfsTTUResetAll* and *xfsTTUResetTimestamp* variables and is indexed by the single variable, *xfsTTUResetManagedServiceName*. When the *xfsTTUResetAll* variable is set to 0 (zero), all the counters in the error table for the managed service are reset to 0 (zero), all other values are ignored.

The *xfsTTUResetTable* is defined as:

- *xfsTTUResetManagedServiceName(1)* which provides the index to the service in question. It is Display String field. The *xfsTTUResetManagedServiceName* parameter corresponds to the value of *xfsMIBRoot.xfsGeneral.xfsMIBV1.xfsManagedServiceTable.xfsManagedServiceEntry.xfsManagedServiceName* in the general table, e.g. "TextTerminal1".
- *xfsTTUResetAll(2)* is a read-write variable. Issue of a Set command on the *xfsTTUResetAll* variable with value 0 (zero) will result in all counters for the managed service being reset to value 0 (zero). Any other value will be ignored. A query of the *xfsTTUResetAll* variable will return 0 (zero).
- *xfsTTUResetTimestamp(3)* is a read-only variable which represents the UTC date and time when the counters in the error table was reset, it is a Display String field. The data is formatted in the following way: "DD/MM/YYYY HH:MM:SS +ZZZ" where DD/MM/YYYY HH:MM:SS is the local date and time. ZZZ is the bias, which is the difference, in minutes, between Co-ordinated Universal Time (UTC) and local time.

As an example, all the error counts can be reset for a device with managed service name equal to “TextTerminal1” by setting the value zero in the *xfsttUResetAll* variable represented by:

xfsttUResetAll.2.7.1.5.1.2.13.84.101.120.116.84.101.114.109.105.110.97.108.49

2.5 XFS TTU Reset Device Table

The *xfsttUResetDeviceTable*(6) is indexed by the single variable, *xfsttUResetDeviceManagedServiceName*. This table contains variables which monitor and control the execution of the reset request.

The *xfsttUResetDeviceAction* variable is used to initiate a reset. Setting this variable will cause the following to happen:

1. The SNMP agent will determine if a Device Reset is allowed by checking the *RemoteDeviceResetAllowed* configuration flag (see XFS Common Management Configuration section, within the *XFS MIB Architecture and SNMP Extensions Programmer's Reference* document). If it is not allowed then the flow continues with step 5, otherwise the flow continues with step 2.
2. Exclusive access to the device will be obtained.
3. A WFS_CMD_TTU_RESET command will be issued.
4. Exclusive access to the device will be relinquished when the WFS_CMD_TTU_RESET command completes.

NOTE Exclusive access must be relinquished as soon as possible and implemented in such a way that deadlocks are avoided.

5. A *xfsttUResetDeviceCompleteTrap* trap will be generated to report the result of the Device Reset request.

The *xfsttUResetDeviceTable* is defined as:

- *xfsttUResetDeviceManagedServiceName*(1) which provides the index to the service in question. It is a Display String field. The *xfsttUResetDeviceManagedServiceName* parameter corresponds to the value of *xfsttUResetDeviceManagedServiceName* in the general table, e.g. “TextTerminal1”.
- *xfsttUResetDeviceAction*(2) is a read-write variable. Issue of a Set command on the *xfsttUResetDeviceAction* variable with value *executeReset*(1) will result in the device being reset as described above.
- *xfsttUResetDeviceMediaControl*(3) is a read-only variable. As there is no media in the TTU device class this variable can only report the *mediaDefault*(1) value.
- *xfsttUResetDeviceStatus*(4) is a read only variable This variable can be used to check if a reset operation is still in progress. It is set when the reset is initiated and cleared when the reset command completes.

As an example, the device with managed service name equal to “TextTerminal1” is reset by setting the *xfsttUResetDeviceAction* variable represented by:

xfsttUResetDeviceAction.2.7.1.6.1.2.13.84.101.120.116.84.101.114.109.105.110.97.108.49

2.6 XFS TTU Capabilities Table

The *xfsttUCapabilitiesTable*(7) groups the variables identifying device capabilities information and additional variables. It is indexed through a single parameter, *xfsttUCapabilitiesManagedServiceName*. All device capabilities variables are read-only.

Additional variables can be used to contain vendor-dependent variables. These variables do not start immediately after the standard variables in order to allow for expansion of the standard variables, the first additional variable can be added at position 1000.

xfsttUCapabilitiesManagedServiceName is the instance identifier of the managed service and uniquely identifies one instance of the TTU class.

As an example, the identifier for the device capabilities value of *xfsTTUCapabilitiesNumberOfLEDs(4)* for a device with managed service name equal to "TextTerminal1" is as follows:

Character	T	e	x	t	T	e	r	m	i	n	a	l	l
ASCII Hex	54	65	78	74	54	65	72	6D	69	6E	61	6C	31
ASCII Dec	84	101	120	116	84	101	114	109	105	110	97	108	49

NOTE SNMP OID representation of strings consists of a length field specifying the number of characters in the string followed by the ASCII code in decimal for each character in the string. Therefore the OID of the above example is:

xfsMIBRoot.2.7.1.7.1.4.13.84.101.120.116.84.101.114.109.105.110.97.108.49

2.6.1 xfsTTUCapabilitiesTable: Capabilities

The first capabilities variable is common across all device classes, the other variables are device class specific.

xfsTTUCapabilitiesManagedServiceName (1)

Uniquely identifies the managed service.

xfsTTUCapabilitiesDeviceType (2)

This specifies the type of the TTU device. Allowed values are as follows:

Value	Meaning
<i>xfsTTUFixed</i> (2)	The TTU is a fixed device.
<i>xfsTTURemovable</i> (3)	The TTU is removable.

xfsTTUCapabilitiesResolution (3)

This holds x,y coordinates specifying the resolution in OCTET STRING. Each x,y coordinate is separated by a NULL terminator while the field is terminated by a double NULL terminator. For example, for a TTU device with *n* number of resolutions, this value will be as follows where <null> is a null terminator:

SizeX₁,SizeY₁<null>SizeX₂,SizeY₂<null>SizeX₃,SizeY₃<null>.....SizeX_n,SizeY_n<null><null>

If *n* is equal to 0, the value is as follows where <null> is a null terminator:

<null><null>

A specific example for a TTU device with three resolutions would be as follows, where <null> is a null terminator:

32,16<null>40,24<null>80,48<null><null>

The resolution indicated in the first position is the default resolution and the device will be placed in this resolution when the Service Provider is initialized or reset through the WFS_CMD_TTU_RESET command.

xfsTTUCapabilitiesNumberOfLEDs (4)

It contains the number of LEDs available in this text terminal unit. This is an integer field.

xfsTTUCapabilitiesKeyLock (5)

It contains whether the text terminal unit has a key lock switch or not. Allowed values are as follows:

Value	Meaning
True (1)	Key lock switch is available.
False (2)	Key lock switch is not available.

xfsTTUCapabilitiesDisplayLight (6)

It contains whether the text terminal unit has a display light that can be switched ON and OFF with the WFS_CMD_TTU_DISPLIGHT command.. Allowed values are as follows:

Value	Meaning
True (1)	Display light is available.
False (2)	Display light is not available.

xfsTTUCapabilitiesCursor (7)

It contains whether the text terminal unit supports cursor or not. Allowed values are as follows:

Value	Meaning
True (1)	Cursor is available.
False (2)	Cursor is not available.

xfsTTUCapabilitiesForms (8)

It contains whether the text terminal unit supports forms oriented input or output or not. Allowed values are as follows:

Value	Meaning
True (1)	Forms oriented input and output is supported.
False (2)	Forms oriented input and output is not supported.

xfsTTUCapabilitiesCharactersSupported (9)

It contains one or more flags specifying the character sets in addition to single byte ASCII, that is supported by the Service Provider. It is a numeric type field and will be set to a combination of hex values according to the values in the following table (zero if none of the choices is supported):

Value	XFS Name	Meaning
0x00000001	WFS_TTU_ASCII	ASCII is supported for XFS forms.
0x00000002	WFS_TTU_UNICODE	UNICODE is supported for XFS forms.

xfsTTUCapabilitiesPowerSaveControl (10)

It contains the capability of the power saving control. It is a TruthValue type field. Allowed values are:

Value	Meaning
True (1)	Power saving is supported.
False (2)	Power saving output is not supported.

xfsTTUCapabilitiesNumberOfLEDsEx (11)

It contains the number of LEDs available in this text terminal unit in the extended structure. This is an integer field.

xfsTTUCapabilitiesAntiFraudModule (12)

It contains the capability of the anti-fraud module. It is a TruthValue type field. Allowed values are as follows:

Value	Meaning
True (1)	The device has an anti-fraud module.
False (2)	The device does not have an anti-fraud module.

xfsTTUCapabilitiesSynchronizableCommands (13)

This holds the list of execute command IDs that can be synchronized. It is an OCTET STRING with each command ID separated by a NULL terminator, while the field itself is terminated by a double NULL terminator. For example, for a device that can synchronize more than one command ID, this value will be as follows where <null> is null terminator:

commandID₁<null>commandID₂<null>.....commandID_n<null><null>

If there are no command IDs that can be synchronized then the value is as follows where <null> is null terminator:

<null><null>

Possible values for each command ID are:

Value	XFS Name	Meaning
701	WFS_CMD_TTU_BEEP	The WFS_CMD_TTU_BEEP command can be synchronized.
702	WFS_CMD_TTU_CLEARSCREEN	The WFS_CMD_TTU_CLEARSCREEN command can be synchronized.
703	WFS_CMD_TTU_DISPLIGHT	The WFS_CMD_TTU_DISPLIGHT command can be synchronized.
704	WFS_CMD_TTU_SET_LED	The WFS_CMD_TTU_SET_LED command can be synchronized.
705	WFS_CMD_TTU_SET_RESOLUTION	The WFS_CMD_TTU_SET_RESOLUTION command can be synchronized.
706	WFS_CMD_TTU_WRITE_FORM	The WFS_CMD_TTU_WRITE_FORM command can be synchronized.
707	WFS_CMD_TTU_READ_FORM	The WFS_CMD_TTU_READ_FORM command can be synchronized.
708	WFS_CMD_TTU_WRITE	The WFS_CMD_TTU_WRITE command can be synchronized.
709	WFS_CMD_TTU_READ	The WFS_CMD_TTU_READ command can be synchronized.
710	WFS_CMD_TTU_RESET	The WFS_CMD_TTU_RESET command can be synchronized.
711	WFS_CMD_TTU_DEFINE_KEYS	The WFS_CMD_TTU_DEFINE_KEYS command can be synchronized.
712	WFS_CMD_TTU_POWER_SAVE_CONTROL	The WFS_CMD_TTU_POWER_SAVE_CONTROL command can be synchronized.
713	WFS_CMD_TTU_SET_LED_EX	The WFS_CMD_TTU_SET_LED_EX command can be synchronized.

xfstTUCapabilitiesExtraCapability (100)

It contains the vendor dependent additional device capability information as an OCTET STRING. The information is returned as a series of "*key=value*" strings. Each string is null-terminated, with the final string terminating with two null characters.

3. TTU Traps

The following sections define XFS Traps that are specific to the TTU device class.

3.1 TTU Detailed Device Status Change Trap

Status changes within managed services are reported as system events to the XFS Agent. The following section explicitly defines the format of the TTU Detailed Device Status Change trap. However, the format is split into two sections; the fields that are common to all device specific traps and the fields that are specific to each device class. The common fields are defined in the *XFS MIB Architecture and SNMP Extensions Programmer's Reference* document. The fields that are specific to the TTU reflect the TTU Status Table as defined in section 2.1.

The detailed device status change event is only generated when the top level status changes within a managed service, i.e. the trap is generated when the *fwDevice* value in the *WFS_INF_TTU_STATUS* response has changed. In addition, this trap is only generated on version 1.1 of the MIB and higher and is sent in addition to the summary device status change trap.

The SNMP Specific trap value 107 defines the trap as a TTU Detailed Device Status Change trap. In the following section, the numbers in parenthesis at the end of each binding just indicate the sequence of the variable bindings within the trap, they do not represent an OID value.

3.1.1 TTU Detailed Device Status Change Trap Format

The following defines the variable bindings included in the TTU Detailed Device Status Change Trap.

xfsmibRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapSysName (1)

This variable binding contains the system generating the alarm, it is a Display String field. It corresponds to *lpszWorkstationName* in the device status change event data from the Service Provider.

xfsmibRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceName (2)

This variable binding represents the managed service name generating the alarm, it is a Display String field. The agent derives this field from the device status change event.

xfsmibRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClass (3)

This variable binding represents the XFS service class identifier generating the alarm, it is a 32-bit integer (INT32). It corresponds to the class identifier for the class name. The class name is identified from the registry value `HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\class`. This ID matches the class OID branch number i.e. PTR=1, IDC=2, CDM=3, etc. See the *XFS MIB Architecture and SNMP Extensions Programmer's Reference* document for a complete list of these values.

xfsmibRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClassName (4)

This variable binding represents the XFS service class name generating the alarm, it is a Display String field. It corresponds to the three character representation of the XFS device class name, and it is useful for human interpretation of a trap. The class name is identified from the registry value `HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\class`.

xfsmibRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceType (5)

This variable binding represents the XFS type identifier generating the alarm, it is a 32-bit integer (INT32). It corresponds to the type identifier as defined in the *WFS_INF_TTU_CAPABILITIES.fwType* field.

xfsmibRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceOid (6)

This variable binding represents the OID of the sub-tree within *xfsmManagedService* defining the management information for this class of managed service. This variable, along with the managed service name as an index, prevents the need for additional querying to find the service specific MIB branch. The TTU MIB class is represented by .1.3.6.1.4.1.16213.2.7

xfsmibRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapPhysicalDeviceName (7)

This variable binding represents the physical device name or names associated with the managed service generating the alarm, it is a Display String field. It corresponds to the physical device name or names identified by the managed service. The managed service name is used to identify the physical device name or

names, from registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\PhysicalDeviceName. Multiple physical device names are comma separated.

xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDeviceVendor (8)

This variable binding represents the XFS device vendor name of the device generating the alarm, it is a Display String field. It corresponds to the vendor name for the Service Provider. The Service Provider is identified from the managed service name and the registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\ServiceProvider.

The Service Provider name is then used to identify the vendor, from the registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\SERVICE_PROVIDERS\<ServiceProviderName>\vendor_name.

xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapMIBVersion (9)

This variable binding represents the XFS MIB version of the device generating the alarm, it is a Display String field. It corresponds to the XFS MIB version for the managed service. The managed service name is used to identify the XFS MIB version, from registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\MibVersion.

xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapEvent (10)

In case of XFS this variable binding represents the XFS event generating the alarm, it is a 32-bit integer (INT32). It corresponds to u.dwEventID in the event data from the Service Provider. See the Application Programming Interface (API) - Service Provider Interface (SPI); Programmer's Reference for a complete description of the event structure.

xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDate (11)

This variable represents the UTC and bias for local translation of the date and time when the event was generated. It is a Display String field. The data is formatted in the following way: "DD/MM/YYYY HH:MM:SS +ZZZ" where DD/MM/YYYY HH:MM:SS is the local date and time. ZZZ is the bias, which is the difference, in minutes, between Co-ordinated Universal Time (UTC) and local time.

xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapSPVersion (12)

This variable represents the vendor-defined version of the Service Provider generating the alarm, it is a Display String field. The Service Provider is identified from the managed service name and the registry value HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\ServiceProvider.

The Service Provider name is then used to identify the version, from the registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\SERVICE_PROVIDERS\<ServiceProviderName>\version.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.**xfsTTUStatusDevice.xfsTTUStatusManagedServiceName** (13)

It contains the state of the physical device. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.**xfsTTUStatusNumberSubDevices.xfsTTUStatusManagedServiceName** (14)

Defines how many sub-devices the service has. This is the number of LEDs the device supports. xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.**xfsTTUStatusKeyboard.xfsTTUStatusManagedServiceName** (15)

It contains the keyboard state. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.**xfsTTUStatusKeyboardLock.xfsTTUStatusManagedServiceName** (16)

It contains the state of the keyboard lock. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.**xfsTTUStatusLed1.xfsTTUStatusManagedServiceName** (17)

It contains the state of Led 1. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.**xfsTTUStatusLed2.xfsTTUStatusManagedServiceName** (18)

It contains the state of Led 2. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed3.xfsTTUStatusManagedServiceName (19)

It contains the state of Led 3. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed4.xfsTTUStatusManagedServiceName (20)

It contains the state of Led 4. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed5.xfsTTUStatusManagedServiceName (21)

It contains the state of Led 5. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed6.xfsTTUStatusManagedServiceName (22)

It contains the state of Led 6. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed7.xfsTTUStatusManagedServiceName (23)

It contains the state of Led 7. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed8.xfsTTUStatusManagedServiceName (24)

It contains the state of Led 8. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDisplaySizeX.xfsTTUStatusManagedServiceName (25)

It contains the state of the horizontal size of display. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDisplaySizeY.xfsTTUStatusManagedServiceName (26)

It contains the state of the vertical size of display. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusExtraStatus.xfsTTUStatusManagedServiceName (27)

It contains the vendor dependent additional device status information as an OCTET STRING. The information is returned as a series of "key=value" strings. Each string is null-terminated, with the final string terminating with two null characters.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDevicePosition.xfsTTUStatusManagedServiceName (28)

It contains the device position. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusPowerSaveRecoveryTime.xfsTTUStatusManagedServiceName (29)

It contains the actual number of seconds required by the device to resume its normal operational state from the current power saving mode. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusAntiFraudModule.xfsTTUStatusManagedServiceName (30)

It contains the state of the anti-fraud module. It is a numeric type field.

3.1.2 TTU Detailed Device Status Change Trap: an example

As an example, the following variable binding list represents a detailed device status change trap (6, 107) that is generated for a TTU with a managed service name of “TextTerminal1”. It reports that the device is in HARDWARE ERROR because the device just had a hardware error.

xfsMIBRoot.3.1.3.1	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapSysName)
	“SST System 1”
xfsMIBRoot.3.1.3.2	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceName)
	“TextTerminal1”
xfsMIBRoot.3.1.3.3	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClass)
	7 (WFS_SERVICE_CLASS_TTU)
xfsMIBRoot.3.1.3.4	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClassName)
	“TTU”
xfsMIBRoot.3.1.3.5	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceType)
	1 (WFS_TTU_FIXED)
xfsMIBRoot.3.1.3.6	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceOid)
	“.1.3.6.1.4.1.16213.2.7”
xfsMIBRoot.3.1.3.7	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapPhysicalDeviceName)
	“KB Soft Text Terminal”
xfsMIBRoot.3.1.3.8	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDeviceVendor)
	“KB Text Terminals Incorporated”
xfsMIBRoot.3.1.3.9	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapMIBVersion)
	“1.10”
xfsMIBRoot.3.1.3.10	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapEvent)
	4 (WFS_SYSE_DEVICE_STATUS)
xfsMIBRoot.3.1.3.11	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDate)
	“20/03/2003 15:40:53 -300”
xfsMIBRoot.3.1.3.12	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapSPVersion)
	“1.23”
xfsMIBRoot.2.7.1.2.1.3.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDevice.xfsTTUStatusManagedServiceName)
	5 (WFS_STAT_HWERROR)
xfsMIBRoot.2.7.1.2.1.2.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusNumberSubDevices.xfsTTUStatusManagedServiceName)
	1 (One sub device)
xfsMIBRoot.2.7.1.2.1.4.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusKeyboard.xfsTTUStatusManagedServiceName)
	3 (xfsTTUKeyboardOff)
xfsMIBRoot.2.7.1.2.1.5.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusKeylock.xfsTTUStatusManagedServiceName)
	2 (xfsTTUKeyboardLockOn)
xfsMIBRoot.2.7.1.2.1.	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry

6.Index	.xfsTTUStatusLed1.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 7.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed2.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 8.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed3.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 9.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed4.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 10.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed5.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 11.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed6.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 12.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed7.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 13.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed8.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 14.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusDisplaySizeX.xfsTTUStatusManagedServiceName)
	0
xfsMIBRoot.2.7.1.2.1. 15.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusDisplaySizeY.xfsTTUStatusManagedServiceName)
	1
xfsMIBRoot.2.7.1.2.1. 100.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusExtraStatus.xfsTTUStatusManagedServiceName)
	"0"0' (No extra data)
xfsMIBRoot.2.7.1.2.1. 16.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusDevicePosition.xfsTTUStatusManagedServiceName)
	1 (xfsTTUDeviceInPosition)
xfsMIBRoot.2.7.1.2.1. 17.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusPowerSaveRecoveryTime.xfsTTUStatusManagedServiceName)
	0
xfsMIBRoot.2.7.1.2.1. 18.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusAntiFraudModule.xfsTTUStatusManagedServiceName)
	2 (xfsTTUAFMOK)

3.2 TTU Sub-Device Status Change Trap

The TTU does support sub-devices, but there are no associated events, so the TTU Sub-Device Status Change Trap is not currently defined. The SNMP Specific trap value 207 is reserved in case a sub-device trap is ever added to the TTU device class.

3.3 TTU Reset Device Complete Trap

On the TTU device class this trap reports the completion of the reset device request and includes the status of the device at that point. If the reset has changed the status of the device then the Device Status Change and a Detail Device Status traps will also be generated.

The SNMP Specific trap value 307 defines the trap as a TTU Reset Device Complete trap.

3.3.1 TTU Reset Device Complete Trap Format

The following defines the variable bindings included in the TTU Reset Device Complete Trap. In the following section, the numbers in parenthesis at the end of each binding just indicate the sequence of the variable bindings within the trap, they do not represent an OID value.

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapResetDeviceResult (1)`

This variable binding contains a value indicating if the reset was executed, and if not provides a reason. It does not report the status of the device (i.e. the result of the reset), the current status of the device is reported within the **xfsxfsTTUStatusDevice** binding (var bind 12 below).

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceName (2)`

This variable binding represents the managed service name generating the alarm, it is a Display String field. The agent derives this field from the device status change event.

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClass (3)`

This variable binding represents the XFS service class identifier generating the alarm, it is a 32-bit integer (INT32). It corresponds to the class identifier for the class name. The class name is identified from the registry value

`HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\class`. This ID matches the class OID branch number i.e. PTR=1, IDC=2, CDM=3, etc. See the *XFS MIB Architecture and SNMP Extensions Programmer's Reference* document for a complete list of these values.

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClassName (4)`

This variable binding represents the XFS service class name generating the alarm, it is a Display String field. It corresponds to the three character representation of the XFS device class name, and it is useful for human interpretation of a trap. The class name is identified from the registry value

`HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\class`.

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceType (5)`

This variable binding represents the XFS type identifier generating the alarm, it is a 32-bit integer (INT32). It corresponds to the type identifier as defined in the `WFS_INF_TTU_CAPABILITIES.fwType` field.

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceOid (6)`

This variable binding represents the OID of the sub-tree within *xfsManagedService* defining the management information for this class of managed service. This variable, along with the managed service name as an index, prevents the need for additional querying to find the service specific MIB branch. The TTU MIB class is represented by .1.3.6.1.4.1.16213.2.7

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapPhysicalDeviceName (7)`

This variable binding represents the physical device name or names associated with the managed service generating the alarm, it is a Display String field. It corresponds to the physical device name or names identified by the managed service. The managed service name is used to identify the physical device name or names, from registry value

`HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\PhysicalDeviceName`. Multiple physical device names are comma separated.

`xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDeviceVendor (8)`

This variable binding represents the XFS device vendor name of the device generating the alarm, it is a Display String field. It corresponds to the vendor name for the Service Provider. The Service Provider is identified from the managed service name and the registry value

`HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\ServiceProvider`.

The Service Provider name is then used to identify the vendor, from the registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\SERVICE_PROVIDERS\<ServiceProviderName>\vendor_name.

xfMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapMIBVersion (9)

This variable binding represents the XFS MIB version of the device generating the alarm, it is a Display String field. It corresponds to the XFS MIB version for the managed service. The managed service name is used to identify the XFS MIB version, from registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\MibVersion.

xfMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDate (10)

This variable represents the UTC and bias for local translation of the date and time when the event was generated. It is a Display String field. The data is formatted in the following way: "DD/MM/YYYY HH:MM:SS+ZZZ" where DD/MM/YYYY HH:MM:SS is the local date and time. ZZZ is the bias, which is the difference, in minutes, between Co-ordinated Universal Time (UTC) and local time.

xfMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapSPVersion (11)

This variable represents the vendor-defined version of the Service Provider generating the alarm, it is a Display String field. The Service Provider is identified from the managed service name and the registry value HKEY_LOCAL_MACHINE\SOFTWARE\XFS\MANAGEMENT_PROVIDERS\<ManagedServiceName>\ServiceProvider.

The Service Provider name is then used to identify the version, from the registry value

HKEY_LOCAL_MACHINE\SOFTWARE\XFS\SERVICE_PROVIDERS\<ServiceProviderName>\version.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDevice.xfsTTUStatusManagedServiceName (12)

This variable binding represents the current state of the physical device managed by the service. It is a 32 bit integer (INT32).

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusNumberSubDevices.xfsTTUStatusManagedServiceName (13)

Defines how many sub-devices the service has. This is the number of LEDs the device supports.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusKeyboard.xfsTTUStatusManagedServiceName (14)

It contains the keyboard state. It is a numeric type field.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusKeyboardLock.xfsTTUStatusManagedServiceName (15)

It contains the state of the keyboard lock. It is a numeric type field.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed1.xfsTTUStatusManagedServiceName (16)

It contains the state of Led 1. It is a numeric type field.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed2.xfsTTUStatusManagedServiceName (17)

It contains the state of Led 2. It is a numeric type field.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed3.xfsTTUStatusManagedServiceName (18)

It contains the state of Led 3. It is a numeric type field.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed4.xfsTTUStatusManagedServiceName (19)

It contains the state of Led 4. It is a numeric type field.

xfMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed5.xfsTTUStatusManagedServiceName (20)

It contains the state of Led 5. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed6.xfsTTUStatusManagedServiceName (21)

It contains the state of Led 6. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed7.xfsTTUStatusManagedServiceName (22)

It contains the state of Led 7. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed8.xfsTTUStatusManagedServiceName (23)

It contains the state of Led 8. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDisplaySizeX.xfsTTUStatusManagedServiceName (24)

It contains the state of the horizontal size of display. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDisplaySizeY.xfsTTUStatusManagedServiceName (25)

It contains the state of the vertical size of display. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusExtraStatus.xfsTTUStatusManagedServiceName (26)

It contains the vendor dependent additional device status information as an OCTET STRING. The information is returned as a series of "key=value" strings. Each string is null-terminated, with the final string terminating with two null characters.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDevicePosition.xfsTTUStatusManagedServiceName (27)

It contains the device position. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusPowerSaveRecoveryTime.xfsTTUStatusManagedServiceName (28)

It contains the actual number of seconds required by the device to resume its normal operational state from the current power saving mode. It is a numeric type field.

xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusAntiFraudModule.xfsTTUStatusManagedServiceName (29)

It contains the state of the anti-fraud module. It is a numeric type field.

3.3.2 TTU Reset Device Complete: an example

As an example, the following variable binding list represents a Reset Device Complete trap (6, 307) generated as the result of a request to reset the device from the remote management station. The device in question has a managed service name "TextTerminal1".

xfsMIBRoot.3.1.3.13	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapResetDeviceResult)
	0 (resetExecuted)
xfsMIBRoot.3.1.3.2	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceName)
	"TextTerminal1"
xfsMIBRoot.3.1.3.3	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClasses)
	7 (WFS_SERVICE_CLASS_TTU)

xfsMIBRoot.3.1.3.4	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceClassName)
	“TTU”
xfsMIBRoot.3.1.3.5	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceType)
	1 (WFS TTU FIXED)
xfsMIBRoot.3.1.3.6	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapManagedServiceOid)
	“.1.3.6.1.4.1.16213.2.7”
xfsMIBRoot.3.1.3.7	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapPhysicalDeviceName)
	“KB Soft Text Terminal”
xfsMIBRoot.3.1.3.8	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDeviceVendor)
	“KB Text Terminals Incorporated”
xfsMIBRoot.3.1.3.9	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapMIBVersion)
	“1.10”
xfsMIBRoot.3.1.3.11	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapDate)
	“20/03/2003 15:40:53 -300”
xfsMIBRoot.3.1.3.12	(xfsMIBRoot.xfsTrap.xfsTrapV1.xfsCommonTrapVars.xfsCommonTrapSPVersion)
	“1.23”
xfsMIBRoot.2.7.1.2.1.3.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusDevice.xfsTTUStatusManagedServiceName)
	1 (WFS STAT ONLINE)
xfsMIBRoot.2.7.1.2.1.2.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusNumberSubDevices.xfsTTUStatusManagedServiceName)
	1 (One sub device)
xfsMIBRoot.2.7.1.2.1.4.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusKeyboard.xfsTTUStatusManagedServiceName)
	3 (xfsTTUKeyboardOff)
xfsMIBRoot.2.7.1.2.1.5.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusKeylock.xfsTTUStatusManagedServiceName)
	2 (xfsTTUKeyboardLockOn)
xfsMIBRoot.2.7.1.2.1.6.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed1.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1.7.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed2.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1.8.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed3.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1.9.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed4.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1.10.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry.xfsTTUStatusLed5.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)

xfsMIBRoot.2.7.1.2.1. 11.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed6.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 12.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed7.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 13.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusLed8.xfsTTUStatusManagedServiceName)
	3 (xfsTTULedSlowFlash)
xfsMIBRoot.2.7.1.2.1. 14.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusDisplaySizeX.xfsTTUStatusManagedServiceName)
	8
xfsMIBRoot.2.7.1.2.1. 15.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusDisplaySizeY.xfsTTUStatusManagedServiceName)
	1
xfsMIBRoot.2.7.1.2.1. 100.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusExtraStatus.xfsTTUStatusManagedServiceName)
	"0"0' (No extra data)
xfsMIBRoot.2.7.1.2.1. 16.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusDevicePosition.xfsTTUStatusManagedServiceName)
	1(xfsTTUDeviceInPosition)
xfsMIBRoot.2.7.1.2.1. 17.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusPowerSaveRecoveryTime.xfsTTUStatusManagedServiceName)
	0
xfsMIBRoot.2.7.1.2.1. 18.Index	(xfsMIBRoot.xfsManagedService.xfsTTU.xfsTTUV1.xfsTTUStatusTable.xfsTTUStatusEntry .xfsTTUStatusAntiFraudModule.xfsTTUStatusManagedServiceName)
	2 (xfsTTUAFMOK)

4. Appendix A - TTU MIB sub-tree

The following paragraph contains the definition of the XFS TTU MIB sub-tree in ASN-1 format.

4.1 TTU MIB in SMIv2 and SMIv1 format



SMIv2_xfsTTU.mib



SMIv1_xfsTTU.mib

The following text is the content of xfsTTU.mib in SMIv2 format.

```
-- *****
-- XFS 3.30 MIB for TTU
-- Management Information Base for XFS TTU Device
--
-- The TTU Number is 7
-- The ASN.1 prefix to, and including the TTU is: 1.3.6.1.4.1.16213.2.7
--
--*****

XFS-TTU-MIB DEFINITIONS ::= BEGIN

    IMPORTS
        Integer32, OBJECT-TYPE, OBJECT-IDENTITY, NOTIFICATION-TYPE
            FROM SNMPv2-SMI
        DisplayString, TruthValue
            FROM SNMPv2-TC
        xfsTTU, xfsTrap, IxfsMIBDeviceStatus
            FROM XFSMIB;

--
-- Type definitions
--

--*****
-- TTU #defines
--*****

IxfsTTUKeyboardStatus ::= INTEGER
    {xfsTTUKeyboardNotAvailable(1),
     xfsTTUKeyboardOn(2),
     xfsTTUKeyboardOff(3)}

IxfsTTUKeyLockStatus ::= INTEGER
    {xfsTTUKeyboardLockNotAvailable(1),
     xfsTTUKeyboardLockOn(2),
     xfsTTUKeyboardLockOff(3)}

IxfsTTULedsStatus ::= INTEGER
    {xfsTTULedNotAvailable(1),
     xfsTTULedOff(2),
     xfsTTULedSlowFlash(3),
     xfsTTULedMediumFlash(5),
     xfsTTULedQuickFlash(9),
     xfsTTULedContinuous(129)}

IxfsTTUDevicePositionStatus ::= INTEGER
    {xfsTTUDeviceInPosition(1),
     xfsTTUDeviceNotInPosition(2),
     xfsTTUDevicePosUnknown(3),
     xfsTTUDevicePosNotSupported(4)}

IxfsTTUAntiFraudModuleStatus ::= INTEGER
    {
        xfsTTUAFMNotSupported(1),
        xfsTTUAFMOK(2),
```

```

    xfsTTUAFMInop(3),
    xfsTTUAFMDeviceDetected(4),
    xfsTTUAFMUnknown(5)
}

IxfsTTUDeviceTypeCapability ::= INTEGER
{
    xfsTTUFixed(2),
    xfsTTURemovable(3)
}

--
-- Node definitions
--

--*****
-- Version 1 of TTU MIB
--
-- The ASN.1 prefix to, and including the Version 1 of TTU is:
1.3.6.1.4.1.16213.2.7.1
--
--*****
    xfsTTUV1 OBJECT IDENTIFIER          ::= { xfsTTU 1}

    xfsTTUInstances OBJECT-TYPE
        SYNTAX      Integer32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION  "Number that represents the number of TTU managed services."
        ::= {xfsTTUV1 1}

--*****
-- TTU Device Status Table
--*****

xfsTTUStatusTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF XfsTTUStatusEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION  "Define the set of MIB Variables for the TTU status table."
    ::= {xfsTTUV1 2}

xfsTTUStatusEntry OBJECT-TYPE
    SYNTAX      XfsTTUStatusEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION  "TTU Device Status Table Entry."
    INDEX {xfsTTUStatusManagedServiceName}
    ::= {xfsTTUStatusTable 1}

XfsTTUStatusEntry ::= SEQUENCE {
    xfsTTUStatusManagedServiceName      DisplayString,
    xfsTTUStatusNumberSubDevices          Integer32,
    xfsTTUStatusDevice                    IxfsMIBDeviceStatus,
    xfsTTUStatusKeyboard                  IxfsTTUKeyboardStatus,
    xfsTTUStatusKeyLock                   IxfsTTUKeyLockStatus,
    xfsTTUStatusLed1                      IxfsTTULedsStatus,
    xfsTTUStatusLed2                      IxfsTTULedsStatus,
    xfsTTUStatusLed3                      IxfsTTULedsStatus,
    xfsTTUStatusLed4                      IxfsTTULedsStatus,
    xfsTTUStatusLed5                      IxfsTTULedsStatus,
    xfsTTUStatusLed6                      IxfsTTULedsStatus,
    xfsTTUStatusLed7                      IxfsTTULedsStatus,
    xfsTTUStatusLed8                      IxfsTTULedsStatus,
    xfsTTUStatusDisplaySizeX              Integer32,
    xfsTTUStatusDisplaySizeY              Integer32,
    xfsTTUStatusDevicePosition             IxfsTTUDevicePositionStatus,
    xfsTTUStatusPowerSaveRecoveryTime     Integer32,
    xfsTTUStatusAntiFraudModuleStatus     IxfsTTUAntiFraudModuleStatus,
    xfsTTUStatusExtraStatus                OCTET STRING
}

```

```

xfsTTUStatusManagedServiceName OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Instance identifier of the managed service."
    ::= { xfsTTUStatusEntry 1}

xfsTTUStatusNumberSubDevices OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Number of sub devices supported by the TTU device."
    ::= { xfsTTUStatusEntry 2}

xfsTTUStatusDevice OBJECT-TYPE
    SYNTAX      IxfsMIBDeviceStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Device status."
    ::= { xfsTTUStatusEntry 3}

xfsTTUStatusKeyboard OBJECT-TYPE
    SYNTAX      IxfsTTUKeyboardStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Keyboard status.
                  xfsTTUKeyboardNotAvailable(1),
                  xfsTTUKeyboardOn(2),
                  xfsTTUKeyboardOff(3)"
    ::= { xfsTTUStatusEntry 4}

xfsTTUStatusKeyLock OBJECT-TYPE
    SYNTAX      IxfsTTUKeyLockStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Keyboard Lock status.
                  xfsTTUKeyboardLockNotAvailable(1),
                  xfsTTUKeyboardLockOn(2),
                  xfsTTUKeyboardLockOff(3)"
    ::= { xfsTTUStatusEntry 5}

xfsTTUStatusLed1 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Led 1 status.
                  xfsTTULedNotAvailable(1),
                  xfsTTULedOff(2),
                  xfsTTULedSlowFlash(3),
                  xfsTTULedMediumFlash(5),
                  xfsTTULedQuickFlash(9),
                  xfsTTULedContinuous(129)"

    ::= { xfsTTUStatusEntry 6}

xfsTTUStatusLed2 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Led 2 status. Possible values are the same as Led 1."
    ::= { xfsTTUStatusEntry 7}

xfsTTUStatusLed3 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Led 3 status. Possible values are the same as Led 1."
    ::= { xfsTTUStatusEntry 8}

xfsTTUStatusLed4 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION  "Led 4 status. Possible values are the same as Led 1."

```

```

::= { xfsTTUStatusEntry 9}

xfsTTUStatusLed5 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Led 5 status. Possible values are the same as Led 1."
    ::= { xfsTTUStatusEntry 10}

xfsTTUStatusLed6 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Led 6 status. Possible values are the same as Led 1."
    ::= { xfsTTUStatusEntry 11}

xfsTTUStatusLed7 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Led 7 status. Possible values are the same as Led 1."
    ::= { xfsTTUStatusEntry 12}

xfsTTUStatusLed8 OBJECT-TYPE
    SYNTAX      IxfsTTULedsStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Led 8 status. Possible values are the same as Led 1."
    ::= { xfsTTUStatusEntry 13}

xfsTTUStatusDisplaySizeX OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "The size of the horizontal display (number of columns)."
    ::= {xfsTTUStatusEntry 14}

xfsTTUStatusDisplaySizeY OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "The size of the vertical display (number of rows)."
    ::= {xfsTTUStatusEntry 15}

xfsTTUStatusDevicePosition OBJECT-TYPE
    SYNTAX IxfsTTUDevicePositionStatus
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "It contains the device position.
        xfsTTUDeviceInPosition(1),
        xfsTTUDeviceNotInPosition(2),
        xfsTTUDevicePosUnknown(3),
        xfsTTUDevicePosNotSupported(4)"
    ::= { xfsTTUStatusEntry 16 }

xfsTTUStatusPowerSaveRecoveryTime OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "It contains the actual number of seconds required by the device to
resume its normal operational state from the current power saving mode. This value
is zero if either the power saving mode has not been activated or no power save
control is supported."
    ::= { xfsTTUStatusEntry 17 }

xfsTTUStatusAntiFraudModule OBJECT-TYPE
    SYNTAX IxfsTTUAntiFraudModuleStatus
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION "It contains the anti-fraud module status.
        xfsTTUAFMNotSupported(1),
        xfsTTUAFMOK(2),
        xfsTTUAFMInop(3),
        xfsTTUAFMDeviceDetected(4),
        xfsTTUAFMUnknown(5)."

```

```

 ::= { xfsTTUStatusEntry 18 }

xfsTTUStatusExtraStatus OBJECT-TYPE
    SYNTAX      OCTET STRING
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "Vendor dependent additional device status information."
    ::= {xfsTTUStatusEntry 100}

--*****
-- TTU Sub Device Status Table
--
-- Note that the TTU device does not currently have sub-devices. The
-- sub-device table is not required for this device and is shown as an
-- example for those devices that do support sub-devices.
--
-- Note, to ensure consistency across all MIB extensions OID 16213.2.7.1.3
-- must be reserved for the sub-device table.
--*****

xfsTTUSubDeviceTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF XfsTTUSubDeviceEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION  "Define the set of MIB Variables for the TTU Sub-Device Status
Table."
    ::= {xfsTTUV1 3}

xfsTTUSubDeviceEntry OBJECT-TYPE
    SYNTAX      XfsTTUSubDeviceEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION  "TTU Sub-Device Status Table Entry."
    INDEX {xfsTTUSubDeviceManagedServiceName,
          xfsTTUSubDeviceIndex}
    ::= {xfsTTUSubDeviceTable 1}

XfsTTUSubDeviceEntry ::= SEQUENCE {
    xfsTTUSubDeviceManagedServiceName  DisplayString,
    xfsTTUSubDeviceIndex                INTEGER,
    xfsTTUSubDeviceLED                  Integer32 }

xfsTTUSubDeviceManagedServiceName OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "Instance identifier of the managed service."
    ::= {xfsTTUSubDeviceEntry 1}

xfsTTUSubDeviceIndex OBJECT-TYPE
    SYNTAX      INTEGER (1..65535)
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "Index into the array of sub devices supported."
    ::= {xfsTTUSubDeviceEntry 2}

xfsTTUSubDeviceLED OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "It contains the state of the LED."
    ::= {xfsTTUSubDeviceEntry 3}

--*****
-- TTU Error Table
--*****

xfsTTUErrorTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF XfsTTUErrorEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION  "Define the set of MIB Variables for the TTU Error Table."

```

```

 ::= {xfsTTUV1 4}

xfsTTUErrorEntry OBJECT-TYPE
    SYNTAX      XfsTTUErrorEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION "TTU Error Table Entry."
    INDEX {xfsTTUErrorManagedServiceName,
           xfsTTUErrorCommandCode,
           xfsTTUErrorResponseCode}
    ::= {xfsTTUErrorTable 1}

XfsTTUErrorEntry ::= SEQUENCE {
    xfsTTUErrorManagedServiceName  DisplayString,
    xfsTTUErrorCommandCode          INTEGER,
    xfsTTUErrorResponseCode         INTEGER,
    xfsTTUErrorCount                Integer32 }

xfsTTUErrorManagedServiceName OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "Instance identifier of the managed service."
    ::= {xfsTTUErrorEntry 1}

xfsTTUErrorCommandCode OBJECT-TYPE
    SYNTAX      INTEGER (701..800)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "The executable command code supported by the Service
                Provider associated with the error count of interest."
    ::= {xfsTTUErrorEntry 2}

xfsTTUErrorResponseCode OBJECT-TYPE
    SYNTAX      INTEGER (0..99 | 700..799)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION "The response code supported by Service Provider for the
                corresponding command code associated with the error count
                of interest."
    ::= {xfsTTUErrorEntry 3}

xfsTTUErrorCount OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION "The counter value corresponding to the managed service,
                command code and response code."
    ::= {xfsTTUErrorEntry 4}

--*****
-- TTU Reset Table
--*****

xfsTTUResetTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF XfsTTUResetEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION "Defines the set of MIB Variables for the TTU Reset Table."
    ::= {xfsTTUV1 5}

xfsTTUResetEntry OBJECT-TYPE
    SYNTAX      XfsTTUResetEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION "TTU Reset Table Entry."
    INDEX {xfsTTUResetManagedServiceName}
    ::= {xfsTTUResetTable 1}

XfsTTUResetEntry ::= SEQUENCE {
    xfsTTUResetManagedServiceName  DisplayString,
    xfsTTUResetAll                  Integer32,
    xfsTTUResetTimestamp            DisplayString}

```

```

xfsTTUResetManagedServiceName OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION   "Instance identifier of the managed service."
    ::= { xfsTTUResetEntry 1 }

xfsTTUResetAll OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION   "Returns all counter values for this managed service to
                  zero when set to zero and returns zero when read."
    ::= { xfsTTUResetEntry 2 }

xfsTTUResetTimestamp OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION   "Date and time the last reset of the counters was
                  performed."
    ::= { xfsTTUResetEntry 3 }

-- *****
-- TTU Reset Device Table
-- *****
xfsTTUResetDeviceTable OBJECT-TYPE
    SYNTAX SEQUENCE OF XfsTTUResetDeviceEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "Define the set of MIB Variables for the TTU Reset Device Table."
    ::= { xfsTTUV1 6 }

xfsTTUResetDeviceEntry OBJECT-TYPE
    SYNTAX XfsTTUResetDeviceEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "TTU Reset Device Table Entry."
    INDEX { xfsTTUResetDeviceManagedServiceName }
    ::= { xfsTTUResetDeviceTable 1 }

XfsTTUResetDeviceEntry ::=
    SEQUENCE {
        xfsTTUResetDeviceManagedServiceName
            DisplayString,
        xfsTTUResetDeviceAction
            INTEGER,
        xfsTTUResetDeviceMediaControl
            INTEGER,
        xfsTTUResetDeviceStatus
            INTEGER
    }

xfsTTUResetDeviceManagedServiceName OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Instance identifier of the managed service."
    ::= { xfsTTUResetDeviceEntry 1 }

xfsTTUResetDeviceAction OBJECT-TYPE
    SYNTAX INTEGER { executeReset(1) }
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION "Variable that initiates the device reset"
    ::= { xfsTTUResetDeviceEntry 2 }

xfsTTUResetDeviceMediaControl OBJECT-TYPE
    SYNTAX INTEGER

```

```

        {
            mediaDefault(1)
        }
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Variable that reports the media handling during the device reset"
::= { xfsTTUResetDeviceEntry 3 }

xfsTTUResetDeviceStatus OBJECT-TYPE
SYNTAX INTEGER
    {
        resetIdle(1),
        resetInProgress(2)
    }
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Variable that reports the progress of the device reset"
::= { xfsTTUResetDeviceEntry 4 }

-- *****
-- TTU Device Capabilities Table
-- *****

xfsTTUCapabilitiesTable OBJECT-TYPE
SYNTAX SEQUENCE OF XfsTTUCapabilitiesEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "Define the set of MIB Variables for the TTU capabilities table."
::= { xfsTTUV1 7 }

xfsTTUCapabilitiesEntry OBJECT-TYPE
SYNTAX XfsTTUCapabilitiesEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "TTU Device Capabilities Table Entry."
INDEX { xfsTTUCapabilitiesManagedServiceName }
::= { xfsTTUCapabilitiesTable 1 }

XfsTTUCapabilitiesEntry ::=
SEQUENCE {
    xfsTTUCapabilitiesManagedServiceName
        DisplayString,
    xfsTTUCapabilitiesDeviceType
        INTEGER,
    xfsTTUCapabilitiesResolution
        OCTET STRING,
    xfsTTUCapabilitiesNumberOfLEDs
        Integer32,
    xfsTTUCapabilitiesKeyLock
        TruthValue,
    xfsTTUCapabilitiesDisplayLight
        TruthValue,
    xfsTTUCapabilitiesCursor
        TruthValue,
    xfsTTUCapabilitiesForms
        TruthValue,
    xfsTTUCapabilitiesCharactersSupported
        Integer32,
    xfsTTUCapabilitiesPowerSaveControl
        TruthValue,
    xfsTTUCapabilitiesNumberOfLEDsEx
        Integer32,
    xfsTTUCapabilitiesAntiFraudModule
        TruthValue,
    xfsTTUCapabilitiesSynchronizableCommands
        OCTET STRING,
    xfsTTUCapabilitiesExtraCapability
        OCTET STRING
}

```

```

xfsTTUCapabilitiesManagedServiceName OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Managed service name"
    ::= { xfsTTUCapabilitiesEntry 1 }

xfsTTUCapabilitiesDeviceType OBJECT-TYPE
    SYNTAX IxfsTTUDeviceTypeCapability
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Managed service device type."
    ::= { xfsTTUCapabilitiesEntry 2 }

xfsTTUCapabilitiesResolution OBJECT-TYPE
    SYNTAX OCTET STRING
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Resolution of the TTU device."
    ::= { xfsTTUCapabilitiesEntry 3 }

xfsTTUCapabilitiesNumberOfLEDs OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Managed service device's number of LEDs."
    ::= { xfsTTUCapabilitiesEntry 4 }

xfsTTUCapabilitiesKeyLock OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Key Lock presence
        TRUE - Key Lock switch is available,
        FALSE - Key Lock switch is unavailable."
    ::= { xfsTTUCapabilitiesEntry 5 }

xfsTTUCapabilitiesDisplayLight OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Key Lock presence
        TRUE - Display light is available,
        FALSE - Display light is unavailable."
    ::= { xfsTTUCapabilitiesEntry 6 }

xfsTTUCapabilitiesCursor OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Key Lock presence
        TRUE - Cursor is available,
        FALSE - Cursor is unavailable."
    ::= { xfsTTUCapabilitiesEntry 7 }

xfsTTUCapabilitiesForms OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Key Lock presence
        TRUE - Key Lock switch is available,

```

```

        FALSE - Key Lock switch is unavailable."
        ::= { xfsTTUCapabilitiesEntry 8 }

xfsTTUCapabilitiesCharactersSupported OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Managed service device supported character set."
    ::= { xfsTTUCapabilitiesEntry 9 }

xfsTTUCapabilitiesPowerSaveControl OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "It specifies whether or not power saving control is available. Allowed values
are:
        True(1),
        False(2)."
```

```

        ::= { xfsTTUCapabilitiesEntry 10 }

xfsTTUCapabilitiesNumberOfLEDsEx OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "It specifies the number of LEDs available in the extended structure (0 if not
supported)."
```

```

        ::= { xfsTTUCapabilitiesEntry 11 }

xfsTTUCapabilitiesAntiFraudModule OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "This TruthValue variable specifies whether or not an anti-fraud module is
available."
```

```

        ::= { xfsTTUCapabilitiesEntry 12 }

xfsTTUCapabilitiesSynchronizableCommands OBJECT-TYPE
    SYNTAX OCTET STRING
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Specifies which Command IDs can be synchronized."
```

```

        ::= { xfsTTUCapabilitiesEntry 13 }

xfsTTUCapabilitiesExtraCapability OBJECT-TYPE
    SYNTAX OCTET STRING
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "Vendor dependent additional device capabilities information."
```

```

        ::= { xfsTTUCapabilitiesEntry 100 }

xfsTrapV2 OBJECT-IDENTITY
    STATUS current
    DESCRIPTION
        "Root node for the converted TRAP-TYPES."
    ::= { xfsTrap 0 }

-- *****
-- Trap definitions
-- *****
xfsTTUDetailedDSCTrap NOTIFICATION-TYPE
    OBJECTS { xfsCommonTrapSysName, xfsCommonTrapManagedServiceName,
        xfsCommonTrapManagedServiceClass,
        xfsCommonTrapManagedServiceClassName,
        xfsCommonTrapManagedServiceType,
        xfsCommonTrapManagedServiceOid, xfsCommonTrapPhysicalDeviceName,
        xfsCommonTrapDeviceVendor, xfsCommonTrapMIBVersion,
```

```

    xfsCommonTrapEvent, xfsCommonTrapDate, xfsCommonTrapSPVersion,
    xfsTTUStatusDevice, xfsTTUStatusNumberSubDevices,
    xfsTTUStatusKeyboard, xfsTTUStatusKeyLock,
    xfsTTUStatusLed1, xfsTTUStatusLed2, xfsTTUStatusLed3,
    xfsTTUStatusLed4, xfsTTUStatusLed5, xfsTTUStatusLed6,
    xfsTTUStatusLed7, xfsTTUStatusLed8,
    xfsTTUStatusDisplaySizeX, xfsTTUStatusDisplaySizeY,
    xfsTTUStatusExtraStatus,
    xfsTTUStatusDevicePosition,
    xfsTTUStatusPowerSaveRecoveryTime,
    xfsTTUStatusAntiFraudModule
}
STATUS current
DESCRIPTION
    "This trap indicates a change in the status of a managed
    service."
::= { xfsTrapV2 107 }

xfsTTUResetDeviceCompleteTrap NOTIFICATION-TYPE
OBJECTS { xfsCommonTrapResetDeviceResult, xfsCommonTrapManagedServiceName,
    xfsCommonTrapManagedServiceClass, xfsCommonTrapManagedServiceClassName,
    xfsCommonTrapManagedServiceType, xfsCommonTrapManagedServiceOid,
    xfsCommonTrapPhysicalDeviceName, xfsCommonTrapDeviceVendor,
    xfsCommonTrapMIBVersion, xfsCommonTrapDate,
    xfsCommonTrapSPVersion, xfsTTUStatusDevice, xfsTTUStatusNumberSubDevices,
    xfsTTUStatusKeyboard, xfsTTUStatusKeyLock, xfsTTUStatusLed1,
    xfsTTUStatusLed2, xfsTTUStatusLed3, xfsTTUStatusLed4, xfsTTUStatusLed5,
    xfsTTUStatusLed6, xfsTTUStatusLed7, xfsTTUStatusLed8,
    xfsTTUStatusDisplaySizeX, xfsTTUStatusDisplaySizeY,
    xfsTTUStatusExtraStatus,
    xfsTTUStatusDevicePosition,
    xfsTTUStatusPowerSaveRecoveryTime,
    xfsTTUStatusAntiFraudModule
}
STATUS current
DESCRIPTION
    "This trap indicates the Reset action has complete and reports the
    state of the device after the reset."
::= { xfsTrapV2 307 }

END

--
-- SMIV2_xfsTTU.mib
--

```

5. Appendix B - C-Header files

5.1 XFSMIBTTU.H

```

/*****
*
* xfsmibttn.h          CEN/XFS - MIB TTU
*
*                      Version 3.30  --  Feb 6, 2017
*
*****/

#ifndef __inc_xfsmibttn_h
#define __inc_xfsmibttn_h

#ifdef __cplusplus
extern "C" {
#endif

enum IxfsTTUKeyboardStatus
{
    xfsTTUKeyboardNotAvailable    = 1,
    xfsTTUKeyboardOn,
    xfsTTUKeyboardOff

} xfsTTUKeyboardStatus;

enum IxfsTTUKeyLockStatus
{
    xfsTTUKeyboardLockNotAvailable = 1,
    xfsTTUKeyboardLockOn,
    xfsTTUKeyboardLockOff

} xfsTTUKeyLockStatus;

enum IxfsTTULedsStatus
{
    xfsTTULedNotAvailable    = 1,
    xfsTTULedOff,
    xfsTTULedSlowFlash,
    xfsTTULedMediumFlash    = 5,
    xfsTTULedQuickFlash     = 9,
    xfsTTULedContinuous     = 129

} xfsTTULedsStatus;

enum IxfsTTUAntiFraudModuleStatus
{
    xfsTTUAFMNotSupported    = 1,
    xfsTTUAFMOK,
    xfsTTUAFMInop,
    xfsTTUAFMDeviceDetected,
    xfsTTUAFMUnknown

} xfsTTUAntiFraudModuleStatus;

/*****
*
* MIB Variables for the Status Table
*
*****/
#define xfsTTUStatusManagedServiceName    (1)
#define xfsTTUStatusNumberSubDevices      (2)
#define xfsTTUStatusDevice                 (3)
#define xfsTTUStatusKeyboard               (4)
#define xfsTTUStatusKeyLock                (5)
#define xfsTTUStatusLed1                   (6)
#define xfsTTUStatusLed2                   (7)
#define xfsTTUStatusLed3                   (8)

```

```

#define xfsTTUStatusLed4          (9)
#define xfsTTUStatusLed5          (10)
#define xfsTTUStatusLed6          (11)
#define xfsTTUStatusLed7          (12)
#define xfsTTUStatusLed8          (13)
#define xfsTTUStatusDisplaySizeX  (14)
#define xfsTTUStatusDisplaySizeY  (15)
#define xfsTTUStatusDevicePosition (16)
#define xfsTTUStatusPowerSaveRecoveryTime (17)
#define xfsTTUStatusAntiFraudModule (18)
#define xfsTTUStatusExtraStatus   (100)

/*****
*
*      MIB Variables for the SubDevice Table
*
*****/
#define xfsTTUSubDeviceManagedServiceName (1)
#define xfsTTUSubDeviceIndex              (2)
#define xfsTTUSubDeviceLED                (3)

/*****
*
*      MIB Variables for the Error Table
*
*****/
//Command codes and error codes correspond to the Service Provider definitions.

/*****
*
*      MIB Variables for the Capabilities Table
*
*****/
#define xfsTTCapabilitiesManagedServiceName (1)
#define xfsTTCapabilitiesDeviceType          (2)
#define xfsTTCapabilitiesResolution          (3)
#define xfsTTCapabilitiesNumberOfLEDs        (4)
#define xfsTTCapabilitiesKeyLock              (5)
#define xfsTTCapabilitiesDisplayLight        (6)
#define xfsTTCapabilitiesCursor              (7)
#define xfsTTCapabilitiesForms                (8)
#define xfsTTCapabilitiesCharactersSupported (9)
#define xfsTTCapabilitiesPowerSaveControl     (10)
#define xfsTTCapabilitiesNumberOfLEDsEx      (11)
#define xfsTTCapabilitiesAntiFraudModule      (12)
#define xfsTTCapabilitiesSynchronizableCommands (13)
#define xfsTTCapabilitiesExtraCapability      (100)

#ifdef __cplusplus
} /* extern "C" */
#endif

#endif /* __inc_xfsmibttu_h */

```