Standard SNMP MIBs

Doc Type Name: 3/198 18-CRA 110 1031/1 Uen

Rights and Restrictions

All statements, specifications, recommendations, and technical information contained are current or planned as of the date of publication of this document. They are reliable as of the time of this writing and are presented without warranty of any kind, expressed or implied. In an effort to continuously improve the product and add features, Redback Networks Inc. (“Redback”) or Ericsson AB (“Ericsson”) and their affiliate companies reserve the right to change any specifications contained in this document without prior notice of any kind.

Neither Redback or Ericsson nor its parent or affiliate companies shall be liable for technical or editorial errors or omissions which may occur in this document. Neither Redback or Ericsson nor its affiliate companies shall be liable for any indirect, special, incidental or consequential damages resulting from the furnishing, performance, or use of this document.

Disclaimer

No part of this document may be reproduced in any form without the written permission of the copyright owner.

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Redback or Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Table 3 Circuits for IF-MIB (continued)

Circuit Name	ifType Value	ifDescr Value Format
Link group multilink FR	163—frf16MfrBundle	<lg_name> Type:mfr ID: <lg_id>
Link group multilink PPP	108—pppMultilinkBundle	<lg_name> Type:mp ID: <lg_id>
IP interfaces	53—propVirtual	<interface name>
Loopback	24—softwareLoopback	<interface name>

Objects for IF-MIB

The following sections contain information about objects for IF-MIB. Index objects for each table are identified before each table by the word “index” in bold:

- Non-Table Objects for IF-MIB
- ifTable
- ifXTable
- ifStackTable (Interface Stack Group)
- ifRcvAddressTable (Generic Receive Address Table)
- IF-MIB Groups
- IF-MIB Object Detail

Non-Table Objects for IF-MIB

Table 4 describes the objects for IF-MIB. All objects in the table are read-only.

Table 4 Non-Table Objects in IF-MIB

Object and Object Identifier	Type	Value Range	Description
ifNumber interfaces 1	Integer32	N/A	Number of network interfaces (regardless of their current state) present on this system.
ifTableLastChange ifMIBObjects 5	TimeTicks	N/A	Value of the sysUpTime object as of the last creation or deletion of an entry in the ifTable. If the number of entries is unchanged since the last reinitialization of the local network management subsystem, this object contains a value of 0 (zero).
ifStackLastChange	TimeTicks	N/A	Value of the sysUpTime object at the time of the last change of the whole interface stack. A change of the interface stack is any creation, deletion, or change in value of any instance of ifStackStatus. If the interface stack is unchanged since the last reinitialization of the local network management subsystem, then this object contains a value of zero.

ifTable

The ifTable table contains information about the interface of an entity. Each sublayer below the internetwork layer of a network interface is considered an interface.

Index: ifIndex

Table 5 describes the objects in ifTable. All objects in the table are read-only.

Table 5 Objects in ifTable

Object and Object Identifier	Type	Value Range	Description
ifIndex ifEntry 1	InterfaceIndex	N/A	Unique value, greater than zero, for each interface. We recommend that values be assigned contiguously starting from 1. The value for each interface sublayer must remain constant from at least one reinitialization of the network management system for the entity to the next reinitialization.
ifDescr ifEntry 2	DisplayString	SIZE (0—255)	Text string containing information about the interface. This string should include the name of the manufacturer, the product name, and the version of the interface hardware or software. The value of this object is the same as the value of the ifName object, which is the textual name of the interface. If several entries represent a single interface, they should have the same ifName value or a common ifName prefix. The prefix is set equal to the corresponding port, channel, or subchannel of the ifName object, with the addition of a suffix.
ifType ifEntry 3	IANAifType	N/A	Type of interface. The Internet Assigned Numbers Authority (IANA) assigns additional values for the ifType object by updating the syntax of the IANAifType text convention. For a list of possible values, their descriptions, and requirements for this object, see Table 18.
ifMtu ifEntry 4	Integer32	N/A	Size of the largest packet sent or received on the interface, specified in octets. For interfaces that are used for transmitting network datagrams, the value of this object is the size of the largest network datagram that can be sent on the interface.
ifSpeed ifEntry 5	Gauge32	N/A	Estimate of the current bandwidth of the interface in bits per second. For interfaces with no or an undetermined variance in bandwidth, the value of this object is the nominal bandwidth. If the interface bandwidth is greater than the maximum value of this object, then this object reports its maximum value (4,294,967,295) and the value of the ifHighSpeed object is the interace speed. For a sublayer with no bandwidth, this object is zero.
ifPhysAddress ifEntry 6	PhysAddress	N/A	Interface address at the protocol sublayer. For example, for an 802.x interface, this object normally contains a MAC address. The media-specific MIB of the interface defines the bit and byte ordering and the format of the value of this object. For interfaces with no interface address (for example, a serial line), this object should contain an octet string of zero length.
ifAdminStatus ifEntry 7	Integer	1—up, ready to pass packets 2—down 3—testing	Desired state of the interface. The testing (3) state indicates that no operational packets can pass. When a managed system is initialized, all interfaces start with the ifAdminStatus object in the down (2) state. As a result of either explicit management action or from configuration information retained by the managed system, the ifAdminStatus object is then changed to either the up (1) or testing (3) state or remains in the down (2) state.

Table 5 Objects in ifTable (continued)

Object and Object Identifier	Type	Value Range	Description
ifOperStatus ifEntry 8	Integer	1—up 2—down 3—testing 4—unknown 5—dormant 6—notPresent 7—lowerLayerDown	Current operational state of the interface. The testing (3) state indicates that no operational packets can be passed. If the ifAdminStatus object is down (2), then the ifOperStatus object is down (2). If the ifAdminStatus object changes to up (1), then the ifOperStatus object changes to up (1) if the interface is ready to transmit and receive network traffic. The value of this object changes to dormant (5) if the interface waits for external actions (such as a serial line waiting for an incoming connection). The system remains in the down (2) state if an error prevents it from transitioning to the up (1) state. This object remains in the notPresent (6) state if the interface misses components (for example, hardware).
ifLastChange ifEntry 9	TimeTicks	N/A	Value of the sysUpTime object when the interface entered its current operational state. If the current state was entered before the local network management system was last reinitialized, then this object contains a value of 0 (zero).
ifInOctets ifEntry 10	Counter32	N/A	Total number of octets received on the interface, including framing characters. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.
ifInUcastPkts ifEntry 11	Counter32	N/A	Number of packets, delivered by this sublayer to a higher layer or sublayer, not addressed to a multicast or broadcast address at this sublayer. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.
ifInDiscards ifEntry 13	Counter32	N/A	Number of discarded inbound packets with detected errors that could prevent delivery to a higher-layer protocol (for example, to free buffer space). Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.
ifInErrors ifEntry 14	Counter32	N/A	Number of inbound packets for packet-oriented interfaces that contain errors, which prevents them from being delivered to a higher-layer protocol. For character-oriented or fixed-length interfaces, the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.

Table 5 Objects in ifTable (continued)

Object and Object Identifier	Type	Value Range	Description
ifInUnknownProtos ifEntry 15	Counter32	N/A	Number of packets for packet-oriented interfaces that were received through the interface but discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing, this value is the number of transmission units received through the interface that were discarded because of an unknown or unsupported protocol. For any interface that does not support protocol multiplexing, this counter is always 0. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.
ifOutOctets ifEntry 16	Counter32	N/A	Total number of octets transmitted from the interface, including framing characters. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.
ifOutUcastPkts ifEntry 17	Counter32	N/A	Total number of packets that higher-level protocols requested be transmitted and that were not addressed to a multicast or broadcast address at this sublayer, including those that were discarded or not sent. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.
ifOutDiscards ifEntry 19	Counter32	N/A	Number of outbound packets that were chosen to be discarded even though no errors had been detected to prevent their transmission. These packets may be discarded to free buffer space. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.
ifOutErrors ifEntry 20	Counter32	N/A	Number of outbound packets for packet-oriented interfaces that could not be transmitted because of errors. For character-oriented or fixed-length interfaces, the value of this object is the number of outbound transmission units that could not be transmitted because of errors. Inconsistencies in the value of this counter can occur when the management system is reinitialized and as indicated by the value of the ifCounterDiscontinuityTime object.

The following sections provide examples of ifTable:

- Example: MIB Walk to Collect Port Statistics
- Example: Statistics Collection for ifDescr

Example: MIB Walk to Collect Port Statistics

Use the ifTable object to collect port statistics that the management system needs. The IfTable object is indexed by the ifIndex object, and the ifIndex values, unique values greater than zero, are assigned for each interface. The MIB walk and **snmpget** command limit the operation to the maximum value of the ifIndex object. The following example shows sample output from a MIB walk:

```
sh-3.00# snmpbulkwalk -v2c –c public –m IF-MIB smartedge-lab-1 ifTable
IF-MIB::ifIndex.1 = INTEGER: 1
IF-MIB::ifIndex.2 = INTEGER: 2
IF-MIB::ifIndex.3 = INTEGER: 3
IF-MIB::ifIndex.4 = INTEGER: 4
IF-MIB::ifIndex.5 = INTEGER: 5
IF-MIB::ifIndex.6 = INTEGER: 6
IF-MIB::ifIndex.7 = INTEGER: 7
IF-MIB::ifIndex.8 = INTEGER: 8
IF-MIB::ifIndex.9 = INTEGER: 9
IF-MIB::ifIndex.10 = INTEGER: 10
IF-MIB::ifIndex.11 = INTEGER: 11
IF-MIB::ifIndex.12 = INTEGER: 12
IF-MIB::ifIndex.13 = INTEGER: 13
IF-MIB::ifIndex.14 = INTEGER: 14
IF-MIB::ifIndex.15 = INTEGER: 15
IF-MIB::ifIndex.16 = INTEGER: 16
IF-MIB::ifIndex.17 = INTEGER: 17
IF-MIB::ifIndex.29 = INTEGER: 29
IF-MIB::ifIndex.30 = INTEGER: 30
IF-MIB::ifIndex.56 = INTEGER: 56
IF-MIB::ifDescr.1 = STRING: port ethernet 7/1
IF-MIB::ifDescr.2 = STRING: port ethernet 10/1
IF-MIB::ifDescr.3 = STRING: test_mp Type:mp ID:273
IF-MIB::ifDescr.4 = STRING: goo Type:ether ID:278
IF-MIB::ifDescr.5 = STRING: mmpp Type:mp ID:274
IF-MIB::ifDescr.6 = STRING: foo Type:hdlc ID:275
IF-MIB::ifDescr.7 = STRING: moo Type:hdlc ID:276
IF-MIB::ifDescr.8 = STRING: boo Type:hdlc ID:277
IF-MIB::ifDescr.9 = STRING: port ethernet 1/1
IF-MIB::ifDescr.10 = STRING: port ethernet 1/2
IF-MIB::ifDescr.11 = STRING: port ethernet 1/4
IF-MIB::ifDescr.12 = STRING: port ethernet 1/5
IF-MIB::ifDescr.13 = STRING: port ethernet 1/6
IF-MIB::ifDescr.14 = STRING: port ethernet 1/8
IF-MIB::ifDescr.15 = STRING: port ethernet 1/10
IF-MIB::ifDescr.16 = STRING: port ch-oc12 2/1
IF-MIB::ifDescr.17 = STRING: port ch-oc12 2/1:1
IF-MIB::ifDescr.29 = STRING: port ch-oc12 2/1:1
IF-MIB::ifDescr.30 = STRING: port ch-oc12 2/1:1:1
IF-MIB::ifDescr.56 = STRING: port ch-oc12 2/1:1:2
IF-MIB::ifSpeed.1 = Gauge32: 100000000
IF-MIB::ifSpeed.2 = Gauge32: 4294967295
IF-MIB::ifSpeed.3 = Gauge32: 1536000
IF-MIB::ifSpeed.4 = Gauge32: 10000000
```

```
IF-MIB::ifSpeed.5 = Gauge32: 3072000
IF-MIB::ifSpeed.6 = Gauge32: 451008000
IF-MIB::ifSpeed.7 = Gauge32: 0
IF-MIB::ifSpeed.8 = Gauge32: 0
IF-MIB::ifSpeed.9 = Gauge32: 10000000
IF-MIB::ifSpeed.10 = Gauge32: 10000000
IF-MIB::ifSpeed.11 = Gauge32: 10000000
IF-MIB::ifSpeed.12 = Gauge32: 10000000
IF-MIB::ifSpeed.13 = Gauge32: 10000000
IF-MIB::ifSpeed.14 = Gauge32: 10000000
IF-MIB::ifSpeed.15 = Gauge32: 10000000
IF-MIB::ifSpeed.16 = Gauge32: 622080000
IF-MIB::ifSpeed.17 = Gauge32: 50112000
IF-MIB::ifSpeed.29 = Gauge32: 44736000
IF-MIB::ifSpeed.30 = Gauge32: 1544000
IF-MIB::ifSpeed.56 = Gauge32: 1544000
IF-MIB::ifPhysAddress.1 = STRING: 0:30:88:0:1c:1
IF-MIB::ifPhysAddress.2 = STRING: 0:0:0:0:0:0
IF-MIB::ifPhysAddress.3 = STRING:
IF-MIB::ifPhysAddress.4 = STRING: 0:30:88:0:32:99
IF-MIB::ifPhysAddress.5 = STRING:
IF-MIB::ifPhysAddress.6 = STRING:
IF-MIB::ifPhysAddress.7 = STRING:
IF-MIB::ifPhysAddress.8 = STRING:
IF-MIB::ifPhysAddress.9 = STRING: 0:30:88:0:32:99
IF-MIB::ifPhysAddress.10 = STRING: 0:30:88:0:32:9a
IF-MIB::ifPhysAddress.11 = STRING: 0:30:88:0:32:9c
IF-MIB::ifPhysAddress.12 = STRING: 0:30:88:0:32:9d
IF-MIB::ifPhysAddress.13 = STRING: 0:30:88:0:32:9e
IF-MIB::ifPhysAddress.14 = STRING: 0:30:88:0:32:a0
IF-MIB::ifPhysAddress.15 = STRING: 0:30:88:0:32:a2
IF-MIB::ifPhysAddress.16 = STRING:
IF-MIB::ifPhysAddress.17 = STRING:
IF-MIB::ifPhysAddress.29 = STRING:
IF-MIB::ifPhysAddress.30 = STRING:
IF-MIB::ifPhysAddress.56 = STRING:
IF-MIB::ifAdminStatus.1 = INTEGER: up(1)
IF-MIB::ifAdminStatus.2 = INTEGER: down(2)
IF-MIB::ifAdminStatus.3 = INTEGER: up(1)
IF-MIB::ifAdminStatus.4 = INTEGER: up(1)
IF-MIB::ifAdminStatus.5 = INTEGER: up(1)
IF-MIB::ifAdminStatus.6 = INTEGER: up(1)
IF-MIB::ifAdminStatus.7 = INTEGER: up(1)
IF-MIB::ifAdminStatus.8 = INTEGER: up(1)
IF-MIB::ifAdminStatus.9 = INTEGER: up(1)
IF-MIB::ifAdminStatus.10 = INTEGER: down(2)
IF-MIB::ifAdminStatus.11 = INTEGER: down(2)
IF-MIB::ifAdminStatus.12 = INTEGER: up(1)
IF-MIB::ifAdminStatus.13 = INTEGER: up(1)
IF-MIB::ifAdminStatus.14 = INTEGER: down(2)
IF-MIB::ifAdminStatus.15 = INTEGER: down(2)
```

```
IF-MIB::ifAdminStatus.16 = INTEGER: down(2)
IF-MIB::ifAdminStatus.17 = INTEGER: down(2)
IF-MIB::ifAdminStatus.29 = INTEGER: down(2)
IF-MIB::ifAdminStatus.30 = INTEGER: down(2)
IF-MIB::ifAdminStatus.56 = INTEGER: down(2)
IF-MIB::ifOperStatus.1 = INTEGER: up(1)
IF-MIB::ifOperStatus.2 = INTEGER: notPresent(6)
IF-MIB::ifOperStatus.3 = INTEGER: down(2)
IF-MIB::ifOperStatus.4 = INTEGER: down(2)
IF-MIB::ifOperStatus.5 = INTEGER: down(2)
IF-MIB::ifOperStatus.6 = INTEGER: down(2)
IF-MIB::ifOperStatus.7 = INTEGER: down(2)
IF-MIB::ifOperStatus.8 = INTEGER: down(2)
IF-MIB::ifOperStatus.9 = INTEGER: down(2)
IF-MIB::ifOperStatus.10 = INTEGER: down(2)
IF-MIB::ifOperStatus.11 = INTEGER: down(2)
IF-MIB::ifOperStatus.12 = INTEGER: down(2)
IF-MIB::ifOperStatus.13 = INTEGER: down(2)
IF-MIB::ifOperStatus.14 = INTEGER: down(2)
IF-MIB::ifOperStatus.15 = INTEGER: down(2)
IF-MIB::ifOperStatus.16 = INTEGER: notPresent(6)
IF-MIB::ifOperStatus.17 = INTEGER: lowerLayerDown(7)
IF-MIB::ifOperStatus.29 = INTEGER: lowerLayerDown(7)
IF-MIB::ifOperStatus.30 = INTEGER: lowerLayerDown(7)
IF-MIB::ifOperStatus.56 = INTEGER: lowerLayerDown(7)
IF-MIB::ifLastChange.1 = Timeticks: (1932) 0:00:19.32
IF-MIB::ifLastChange.2 = Timeticks: (1938) 0:00:19.38
IF-MIB::ifLastChange.3 = Timeticks: (2654275) 7:22:22.75
IF-MIB::ifLastChange.4 = Timeticks: (2654276) 7:22:22.76
IF-MIB::ifLastChange.5 = Timeticks: (2654276) 7:22:22.76
IF-MIB::ifLastChange.6 = Timeticks: (2654276) 7:22:22.76
IF-MIB::ifLastChange.7 = Timeticks: (2654276) 7:22:22.76
IF-MIB::ifLastChange.8 = Timeticks: (2654276) 7:22:22.76
IF-MIB::ifLastChange.9 = Timeticks: (2654819) 7:22:28.19
IF-MIB::ifLastChange.10 = Timeticks: (2654838) 7:22:28.38
IF-MIB::ifLastChange.11 = Timeticks: (2654888) 7:22:28.88
IF-MIB::ifLastChange.12 = Timeticks: (2654893) 7:22:28.93
IF-MIB::ifLastChange.13 = Timeticks: (2655096) 7:22:30.96
IF-MIB::ifLastChange.14 = Timeticks: (2655104) 7:22:31.04
IF-MIB::ifLastChange.15 = Timeticks: (2655135) 7:22:31.35
IF-MIB::ifLastChange.16 = Timeticks: (2655241) 7:22:32.41
IF-MIB::ifLastChange.17 = Timeticks: (2655241) 7:22:32.41
IF-MIB::ifLastChange.29 = Timeticks: (2655241) 7:22:32.41
IF-MIB::ifLastChange.30 = Timeticks: (2655241) 7:22:32.41
IF-MIB::ifLastChange.56 = Timeticks: (2655241) 7:22:32.41
IF-MIB::ifInOctets.2 = Counter32: 0
IF-MIB::ifInOctets.3 = Counter32: 0
IF-MIB::ifInOctets.4 = Counter32: 0
IF-MIB::ifInOctets.5 = Counter32: 0
IF-MIB::ifInOctets.6 = Counter32: 0
IF-MIB::ifInOctets.7 = Counter32: 0
```

```
IF-MIB::ifInOctets.8 = Counter32: 0
IF-MIB::ifInOctets.9 = Counter32: 0
IF-MIB::ifInOctets.10 = Counter32: 0
IF-MIB::ifInOctets.11 = Counter32: 0
IF-MIB::ifInOctets.12 = Counter32: 0
IF-MIB::ifInOctets.13 = Counter32: 0
IF-MIB::ifInOctets.14 = Counter32: 0
IF-MIB::ifInOctets.15 = Counter32: 0
IF-MIB::ifInUcastPkts.2 = Counter32: 0
IF-MIB::ifInUcastPkts.3 = Counter32: 0
IF-MIB::ifInUcastPkts.4 = Counter32: 0
IF-MIB::ifInUcastPkts.5 = Counter32: 0
IF-MIB::ifInUcastPkts.6 = Counter32: 0
IF-MIB::ifInUcastPkts.7 = Counter32: 0
IF-MIB::ifInUcastPkts.8 = Counter32: 0
IF-MIB::ifInUcastPkts.9 = Counter32: 0
IF-MIB::ifInUcastPkts.10 = Counter32: 0
IF-MIB::ifInUcastPkts.11 = Counter32: 0
IF-MIB::ifInUcastPkts.12 = Counter32: 0
IF-MIB::ifInUcastPkts.13 = Counter32: 0
IF-MIB::ifInUcastPkts.14 = Counter32: 0
IF-MIB::ifInUcastPkts.15 = Counter32: 0
IF-MIB::ifInNUcastPkts.2 = Counter32: 0
IF-MIB::ifInNUcastPkts.3 = Counter32: 0
IF-MIB::ifInNUcastPkts.4 = Counter32: 0
IF-MIB::ifInNUcastPkts.5 = Counter32: 0
IF-MIB::ifInNUcastPkts.6 = Counter32: 0
IF-MIB::ifInNUcastPkts.7 = Counter32: 0
IF-MIB::ifInNUcastPkts.8 = Counter32: 0
IF-MIB::ifInNUcastPkts.9 = Counter32: 0
IF-MIB::ifInNUcastPkts.10 = Counter32: 0
IF-MIB::ifInNUcastPkts.11 = Counter32: 0
IF-MIB::ifInNUcastPkts.12 = Counter32: 0
IF-MIB::ifInNUcastPkts.13 = Counter32: 0
IF-MIB::ifInNUcastPkts.14 = Counter32: 0
IF-MIB::ifInNUcastPkts.15 = Counter32: 0
```

Example: Statistics Collection for ifDescr

Use the ifTable object to collect statistics for port-facing Ethernet port 10/1 that the management system needs. Before collecting statistics for any port, you must retrieve its ifIndex value. A MIB walk of the ifDescr object retrieves the value of the ifIndex object and the corresponding port information as shown in the following example of MIB walk output:

```
IF-MIB::ifDescr.2 = STRING: port ethernet 10/1
```

The system compares a string returned for each ifDescr value in a MIB walk with the related port. After the MIB walk returns the desired value, the ifIndex value is obtained from the object index. In this case, the ifIndex value for the Ethernet port 10/1 is 2.

Enter the **get** command for the ifIndex object to retrieve required statistics. The following examples show how to retrieve port statistics in the CLI by using different **show** commands:

```
[local]victoria#show port 7/1 detail
```

```
ethernet 7/1 state is Up
Description              :
Line state               : Up
Admin state              : Up
Encapsulation            : ethernet
MTU size                 : 1500 Bytes
MAC address              : 00:30:88:00:03:47
Media type               : 100Base-Tx
Speed                    : 100 Mbps
Duplex mode              : full
Active Alarms            : NONE


[local]victoria#show port counters 7/1 detail
Counters for port ethernet 7/1
General Counters
packets sent       : 2037126          packets recvd      : 1191801
bytes sent         : 2316558829       bytes recvd        : 288753306
mcast pkts sent    : 3                mcast pkts recvd   : 512541
bcast pkts sent    : 0                bcast pkts recvd   : 0
dropped pkts out   : 0                dropped pkts in    : 0
pending pkts out   : 0                pending pkts in    : 0
port drops out     : 0                port drops in      : 0
invalid ctx out    : 0                invalid ctx in     : 0
[local]victoria#
```

ifXTable

The ifXTable table contains a list of interface entries. The number of entries is given by the value of the ifNumber object. This table contains additional objects for the interface table.

Index: ifIndex

Table 6 describes the attributes in ifXTable. All objects in the table are read-only.

Table 6 Objects in ifXTable

Object and Object Identifier	Type	Value Range	Description
ifName ifXEntry 1	DisplayString	N/A	Name of the interface in text. The value of this object is the name of the interface as assigned by the local device and is suitable for use in commands entered at the device console. This value might be a text name, such as “le0,” or a simple port number, such as “1,” depending on the interface-naming syntax of the device. If several entries in the ifTable together represent a single interface named by the device, then each entry has the same value as the ifName object. Note that for an agent that responds to SNMP queries about an interface on some other (proxied) device, the value of the ifName object for that interface is the local name of the proxied device. If no local name exists or this object is otherwise not applicable for an instance, then this object contains a zero-length string.