

Remote Network Monitoring Management Information Base
Version 2
using SMIV2

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it defines objects for managing remote network monitoring devices.

Table of Contents

1 The Network Management Framework	2
2 Overview	2
2.1 Remote Network Management Goals	3
2.2 Structure of MIB	5
3 Control of Remote Network Monitoring Devices	6
3.1 Resource Sharing Among Multiple Management Sta- tions	7
3.2 Row Addition Among Multiple Management Stations	9
4 Conventions	10
5 RMON 2 Conventions	10
5.1 Usage of the term Application Level	10
5.2 Protocol Directory and Limited Extensibility	11
5.3 Errors in packets	11
6 Definitions	12
7 Security Considerations	122
8 Appendix - TimeFilter Implementation Notes	123
9 Acknowledgments	129
10 References	129
11 Author's Address.....	130

1. The Network Management Framework

The Internet-standard Network Management Framework consists of three components. They are:

RFC 1902 [1] which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management.

RFC 1213, STD 17, [3] which defines MIB-II, the core set of managed objects for the Internet suite of protocols.

RFC 1905 [4] which defines the SNMP, the protocol used for network access to managed objects.

The Framework permits new objects to be defined for the purpose of experimentation and evaluation.

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Within a given MIB module, objects are defined using the SMI's OBJECT-TYPE macro. At a minimum, each object has a name, a syntax, an access-level, and an implementation-status.

The name is an object identifier, an administratively assigned name, which specifies an object type. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the object descriptor, to also refer to the object type.

The syntax of an object type defines the abstract data structure corresponding to that object type. The ASN.1 [6] language is used for this purpose. However, RFC 1902 purposely restricts the ASN.1 constructs which may be used. These restrictions are explicitly made for simplicity.

The access-level of an object type defines whether it makes "protocol sense" to read and/or write the value of an instance of the object type. (This access-level is independent of any administrative authorization policy.)

The implementation-status of an object type indicates whether the object is mandatory, optional, obsolete, or deprecated.

2. Overview

This document continues the architecture created in the RMON MIB [RFC 1757] by providing a major feature upgrade, primarily by providing RMON analysis up to the application layer.

Remote network monitoring devices, often called monitors or probes, are instruments that exist for the purpose of managing a network. Often these remote probes are stand-alone devices and devote significant internal resources for the sole purpose of managing a network. An organization may employ many of these devices, one per network segment, to manage its internet. In addition, these devices may be used for a network management service provider to access a client network, often geographically remote.

The objects defined in this document are intended as an interface between an RMON agent and an RMON management application and are not intended for direct manipulation by humans. While some users may tolerate the direct display of some of these objects, few will tolerate the complexity of manually manipulating objects to accomplish row creation. These functions should be handled by the management application.

2.1. Remote Network Management Goals

o Offline Operation

There are sometimes conditions when a management station will not be in constant contact with its remote monitoring devices. This is sometimes by design in an attempt to lower communications costs (especially when communicating over a WAN or dialup link), or by accident as network failures affect the communications between the management station and the probe.

For this reason, this MIB allows a probe to be configured to perform diagnostics and to collect statistics continuously, even when communication with the management station may not be possible or efficient. The probe may then attempt to notify the management station when an exceptional condition occurs. Thus, even in circumstances where communication between management station and probe is not continuous, fault, performance, and configuration information may be continuously accumulated and communicated to the management station conveniently and efficiently.

- o Proactive Monitoring

Given the resources available on the monitor, it is potentially helpful for it continuously to run diagnostics and to log network performance. The monitor is always available at the onset of any failure. It can notify the management station of the failure and can store historical statistical information about the failure. This historical information can be played back by the management station in an attempt to perform further diagnosis into the cause of the problem.

- o Problem Detection and Reporting

The monitor can be configured to recognize conditions, most notably error conditions, and continuously to check for them. When one of these conditions occurs, the event may be logged, and management stations may be notified in a number of ways.

- o Value Added Data

Because a remote monitoring device represents a network resource dedicated exclusively to network management functions, and because it is located directly on the monitored portion of the network, the remote network monitoring device has the opportunity to add significant value to the data it collects. For instance, by highlighting those hosts on the network that generate the most traffic or errors, the probe can give the management station precisely the information it needs to solve a class of problems.

- o Multiple Managers

An organization may have multiple management stations for different units of the organization, for different functions (e.g. engineering and operations), and in an attempt to provide disaster recovery. Because environments with multiple management stations are common, the remote network monitoring device has to deal with more than one management station, potentially using its resources concurrently.

2.2. Structure of MIB

The objects are arranged into the following groups:

- protocol directory
- protocol distribution
- address mapping
- network layer host
- network layer matrix
- application layer host
- application layer matrix
- user history
- probe configuration

These groups are the basic units of conformance. If a remote monitoring device implements a group, then it must implement all objects in that group. For example, a managed agent that implements the network layer matrix group must implement the nlMatrixSDTable and the nlMatrixDSTable.

Implementations of this MIB must also implement the system and interfaces group of MIB-II [3]. MIB-II may also mandate the implementation of additional groups.

These groups are defined to provide a means of assigning object identifiers, and to provide a method for managed agents to know which objects they must implement.

This document also contains enhancements to tables defined in the RMON MIB [RFC 1757]. These enhancements include:

- 1) Adding the DroppedFrames and LastCreateTime conventions to each table defined in the RMON MIB.
- 2) Augmenting the RMON filter table with a mechanism that allows filtering based on an offset from the beginning of a particular protocol, even if the protocol headers are variable length.

- 3) Augmenting the RMON filter and capture status bits with additional bits for WAN media and generic media. These bits are defined here as:

Bit	Definition
6	For WAN media, this bit is set for packets coming from one direction and cleared for packets coming from the other direction. It is an implementation specific matter as to which bit is assigned to which direction, but it must be consistent for all packets received by the agent, and if the agent knows which end of the link is "local" and which end is "network", the bit should be set for packets from the "local" side and should be cleared for packets from the "network" side.
7	For any media, this bit is set for any packet with a physical layer error. This bit may be set in addition to other media-specific bits that denote the same condition.
8	For any media, this bit is set for any packet that is too short for the media. This bit may be set in addition to other media-specific bits that denote the same condition.
9	For any media, this bit is set for any packet that is too long for the media. This bit may be set in addition to other media-specific bits that denote the same condition.

These enhancements are implemented by RMON-2 probes that also implement RMON and do not add any requirements to probes that are compliant to just RMON.

3. Control of Remote Network Monitoring Devices

Due to the complex nature of the available functions in these devices, the functions often need user configuration. In many cases, the function requires parameters to be set up for a data collection operation. The operation can proceed only after these parameters are fully set up.

Many functional groups in this MIB have one or more tables in which to set up control parameters, and one or more data tables in which to place the results of the operation. The control tables are typically read/write in nature, while the data tables are typically read/only.

Because the parameters in the control table often describe resulting data in the data table, many of the parameters can be modified only when the control entry is not active. Thus, the method for modifying these parameters is to de-activate the entry, perform the SNMP Set operations to modify the entry, and then re-activate the entry. Deleting the control entry causes the deletion of any associated data entries, which also gives a convenient method for reclaiming the resources used by the associated data.

Some objects in this MIB provide a mechanism to execute an action on the remote monitoring device. These objects may execute an action as a result of a change in the state of the object. For those objects in this MIB, a request to set an object to the same value as it currently holds would thus cause no action to occur.

To facilitate control by multiple managers, resources have to be shared among the managers. These resources are typically the memory and computation resources that a function requires.

3.1. Resource Sharing Among Multiple Management Stations

When multiple management stations wish to use functions that compete for a finite amount of resources on a device, a method to facilitate this sharing of resources is required. Potential conflicts include:

- o Two management stations wish to simultaneously use resources that together would exceed the capability of the device.
- o A management station uses a significant amount of resources for a long period of time.
- o A management station uses resources and then crashes, forgetting to free the resources so others may use them.

The OwnerString mechanism is provided for each management station initiated function in this MIB to avoid these conflicts and to help resolve them when they occur. Each function has a label identifying the initiator (owner) of the function. This label is set by the initiator to provide for the following possibilities:

- o A management station may recognize resources it owns and no longer needs.
- o A network operator can find the management station that owns the resource and negotiate for it to be freed.
- o A network operator may decide to unilaterally free resources another network operator has reserved.

- o Upon initialization, a management station may recognize resources it had reserved in the past. With this information it may free the resources if it no longer needs them.

Management stations and probes should support any format of the owner string dictated by the local policy of the organization. It is suggested that this name contain one or more of the following: IP address, management station name, network manager's name, location, or phone number. This information will help users to share the resources more effectively.

There is often default functionality that the device or the administrator of the probe (often the network administrator) wishes to set up. The resources associated with this functionality are then owned by the device itself or by the network administrator, and are intended to be long-lived. In this case, the device or the administrator will set the relevant owner object to a string starting with 'monitor'. Indiscriminate modification of the monitor-owned configuration by network management stations is discouraged. In fact, a network management station should only modify these objects under the direction of the administrator of the probe.

Resources on a probe are scarce and are typically allocated when control rows are created by an application. Since many applications may be using a probe simultaneously, indiscriminate allocation of resources to particular applications is very likely to cause resource shortages in the probe.

When a network management station wishes to utilize a function in a monitor, it is encouraged to first scan the control table of that function to find an instance with similar parameters to share. This is especially true for those instances owned by the monitor, which can be assumed to change infrequently. If a management station decides to share an instance owned by another management station, it should understand that the management station that owns the instance may indiscriminately modify or delete it.

It should be noted that a management application should have the most trust in a monitor-owned row because it should be changed very infrequently. A row owned by the management application is less long-lived because a network administrator is more likely to re-assign resources from a row that is in use by one user than from a monitor-owned row that is potentially in use by many users. A row owned by another application would be even less long-lived because the other application may delete or modify that row completely at its discretion.

3.2. Row Addition Among Multiple Management Stations

The addition of new rows is achieved using the RowStatus method described in RFC 1903 [2]. In this MIB, rows are often added to a table in order to configure a function. This configuration usually involves parameters that control the operation of the function. The agent must check these parameters to make sure they are appropriate given restrictions defined in this MIB as well as any implementation specific restrictions such as lack of resources. The agent implementor may be confused as to when to check these parameters and when to signal to the management station that the parameters are invalid. There are two opportunities:

- o When the management station sets each parameter object.
- o When the management station sets the row status object to active.

If the latter is chosen, it would be unclear to the management station which of the several parameters was invalid and caused the badValue error to be emitted. Thus, wherever possible, the implementor should choose the former as it will provide more information to the management station.

A problem can arise when multiple management stations attempt to set configuration information simultaneously using SNMP. When this involves the addition of a new conceptual row in the same control table, the managers may collide, attempting to create the same entry. To guard against these collisions, each such control entry contains a status object with special semantics that help to arbitrate among the managers. If an attempt is made with the row addition mechanism to create such a status object and that object already exists, an error is returned. When more than one manager simultaneously attempts to create the same conceptual row, only the first will succeed. The others will receive an error.

In the RMON MIB [RFC 1757], the EntryStatus textual convention was introduced to provide this mutual exclusion function. Since then, this function was added to the SNMP framework as the RowStatus textual convention. The RowStatus textual convention is used for the definition of all new tables.

When a manager wishes to create a new control entry, it needs to choose an index for that row. It may choose this index in a variety of ways, hopefully minimizing the chances that the index is in use by another manager. If the index is in use, the mechanism mentioned previously will guard against collisions. Examples of schemes to choose index values include random selection or scanning the control

table looking for the first unused index. Because index values may be any valid value in the range and they are chosen by the manager, the agent must allow a row to be created with any unused index value if it has the resources to create a new row.

Some tables in this MIB reference other tables within this MIB. When creating or deleting entries in these tables, it is generally allowable for dangling references to exist. There is no defined order for creating or deleting entries in these tables.

4. Conventions

The following conventions are used throughout the RMON MIB and its companion documents.

Good Packets

Good packets are error-free packets that have a valid frame length. For example, on Ethernet, good packets are error-free packets that are between 64 octets long and 1518 octets long. They follow the form defined in IEEE 802.3 section 3.2.all.

Bad Packets

Bad packets are packets that have proper framing and are therefore recognized as packets, but contain errors within the packet or have an invalid length. For example, on Ethernet, bad packets have a valid preamble and SFD, but have a bad CRC, or are either shorter than 64 octets or longer than 1518 octets.

5. RMON 2 Conventions

The following practices and conventions are introduced in the RMON 2 MIB.

5.1. Usage of the term Application Level

There are many cases in this MIB where the term Application Level is used to describe a class of protocols or a capability. This does not typically mean a protocol that is an OSI Layer 7 protocol. Rather, it is used to identify a class of protocols that is not limited to MAC-layer and network-layer protocols, but can also include transport, session, presentation, and application-layer protocols.

5.2. Protocol Directory and Limited Extensibility

Every RMON 2 implementation will have the capability to parse certain types of packets and identify their protocol type at multiple levels. The protocol directory presents an inventory of those protocol types the probe is capable of monitoring, and allows the addition, deletion, and configuration of protocol types in this list.

One concept deserves special attention: the "limited extensibility" of the protocol directory table. The RMON 2 model is that protocols are detected by static software that has been written at implementation time. Therefore, as a matter of configuration, an implementation does not have the ability to suddenly learn how to parse new packet types. However, an implementation may be written such that the software knows where the demultiplexing field is for a particular protocol, and can be written in such a way that the decoding of the next layer up is table-driven. This works when the code has been written to accommodate it and can be extended no more than one level higher. This extensibility is called "limited extensibility" to highlight these limitations. However, this can be a very useful tool.

For example, suppose that an implementation has C code that understands how to decode IP packets on any of several ethernet encapsulations, and also knows how to interpret the IP protocol field to recognize UDP packets and how to decode the UDP port number fields. That implementation may be table-driven so that among the many different UDP port numbers possible, it is configured to recognize 161 as SNMP, port 53 as DNS, and port 69 as TFTP. The limited extensibility of the protocol directory table would allow an SNMP operation to create an entry that would create an additional table mapping for UDP that would recognize UDP port 123 as NTP and begin counting such packets.

This limited extensibility is an option that an implementation can choose to allow or disallow for any protocol that has child protocols.

5.3. Errors in packets

Packets with link-level errors are not counted anywhere in this MIB because most variables in this MIB requires the decoding of the contents of the packet, which is meaningless if there is a link-level error.

Packets in which protocol errors are detected are counted for all protocols below the layer in which the error was encountered. The implication of this is that packets in which errors are detected at

the network-layer are not counted anywhere in this MIB, while packets with errors detected at the transport layer may have network-layer statistics counted.

6. Definitions

```

RMON2-MIB DEFINITIONS ::= BEGIN
IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, Counter32, Integer32,
        Gauge32, IpAddress, TimeTicks                FROM SNMPv2-SMI
    TEXTUAL-CONVENTION, RowStatus, DisplayString, TimeStamp
                                                FROM SNMPv2-TC
    MODULE-COMPLIANCE, OBJECT-GROUP FROM SNMPv2-CONF
    mib-2, ifIndex                                FROM RFC1213-MIB
    OwnerString, statistics, history, hosts,
    matrix, filter, etherStatsEntry, historyControlEntry,
    hostControlEntry, matrixControlEntry, filterEntry,
    channelEntry                                FROM RMON-MIB
    tokenRing, tokenRingMLStatsEntry, tokenRingPStatsEntry,
    ringStationControlEntry, sourceRoutingStatsEntry
                                                FROM TOKEN-RING-RMON-MIB;
-- Remote Network Monitoring MIB

rmon MODULE-IDENTITY
    LAST-UPDATED "9605270000Z"
    ORGANIZATION "IETF RMON MIB Working Group"
    CONTACT-INFO
        "Steve Waldbusser    (WG Editor)
        Postal: International Network Services
        650 Castro Street, Suite 260
        Mountain View, CA 94041
        Phone:   +1 415 254 4251
        Email:   waldbusser@ins.com

        Andy Bierman    (WG Chair)
        Phone:   +1 805 648 2028
        Email:   abierman@west.net"
    DESCRIPTION
        "The MIB module for managing remote monitoring
        device implementations. This MIB module
        augments the original RMON MIB as specified in
        RFC 1757."
    ::= { mib-2 16 }

-- { rmon 1 } through { rmon 10 } are defined in RMON and
-- the Token Ring RMON MIB [RFC 1513]

```

```

protocolDir      OBJECT IDENTIFIER ::= { rmon 11 }
protocolDist     OBJECT IDENTIFIER ::= { rmon 12 }
addressMap       OBJECT IDENTIFIER ::= { rmon 13 }
nlHost           OBJECT IDENTIFIER ::= { rmon 14 }
nlMatrix         OBJECT IDENTIFIER ::= { rmon 15 }
alHost           OBJECT IDENTIFIER ::= { rmon 16 }
alMatrix         OBJECT IDENTIFIER ::= { rmon 17 }
usrHistory       OBJECT IDENTIFIER ::= { rmon 18 }
probeConfig      OBJECT IDENTIFIER ::= { rmon 19 }
rmonConformance OBJECT IDENTIFIER ::= { rmon 20 }

```

-- Textual Conventions

ZeroBasedCounter32 ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"This TC describes an object which counts events with the following semantics: objects of this type will be set to zero(0) on creation and will thereafter count appropriate events, wrapping back to zero(0) when the value 2^{32} is reached.

Provided that an application discovers the new object within the minimum time to wrap it can use the initial value as a delta since it last polled the table of which this object is part. It is important for a management station to be aware of this minimum time and the actual time between polls, and to discard data if the actual time is too long or there is no defined minimum time.

Typically this TC is used in tables where the INDEX space is constantly changing and/or the TimeFilter mechanism is in use."

SYNTAX Gauge32

LastCreateTime ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"This TC describes an object that stores the last time its entry was created.

This can be used for polling applications to determine that an entry has been deleted and re-created between polls, causing an otherwise undetectable discontinuity in the data."

SYNTAX TimeStamp

TimeFilter ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"To be used for the index to a table. Allows an application to download only those rows changed since a particular time. A row is considered changed if the value of any object in the row changes or if the row is created or deleted.

When sysUpTime is equal to zero, this table shall be empty.

One entry exists for each past value of sysUpTime, except that the whole table is purged should sysUpTime wrap.

As this basic row is updated new conceptual rows are created (which still share the now updated object values with all other instances). The number of instances which are created is determined by the value of sysUpTime at which the basic row was last updated. One instance will exist for each value of sysUpTime at the last update time for the row. A new timeMark instance is created for each new sysUpTime value. Each new conceptual row will be associated with the timeMark instance which was created at the value of sysUpTime with which the conceptual row is to be associated.

By definition all conceptual rows were updated at or after time zero and so at least one conceptual row (associated with timeMark.0) must exist for each underlying (basic) row.

See the appendix for further discussion of this variable.

Consider the following fooTable:

```
fooTable ...
INDEX { fooTimeMark, fooIndex }

FooEntry {
    fooTimeMark    TimeFilter
    fooIndex       INTEGER,
    fooCounts      Counter
}
```

Should there be two basic rows in this table (fooIndex == 1, fooIndex == 2) and row 1 was updated most recently at time 6, while row 2 was updated most recently at time 8, and both rows had been updated on several earlier occasions such that the current values were 5 and 9 respectively then the following fooCounts instances would exist.

```
fooCounts.0.1  5
fooCounts.0.2  9
fooCounts.1.1  5
```

```

fooCounts.1.2  9
fooCounts.2.1  5
fooCounts.2.2  9
fooCounts.3.1  5
fooCounts.3.2  9
fooCounts.4.1  5
fooCounts.4.2  9
fooCounts.5.1  5
fooCounts.5.2  9
fooCounts.6.1  5
fooCounts.6.2  9
fooCounts.7.2  9    -- note that row 1 doesn't exist for
fooCounts.8.2  9    -- times 7 and 8"
SYNTAX      TimeTicks

```

```
DataSource ::= TEXTUAL-CONVENTION
```

```
STATUS      current
```

```
DESCRIPTION
```

"Identifies the source of the data that the associated function is configured to analyze. This source can be any interface on this device.

In order to identify a particular interface, this object shall identify the instance of the ifIndex object, defined in [3,5], for the desired interface.

For example, if an entry were to receive data from interface #1, this object would be set to ifIndex.1."

```
SYNTAX      OBJECT IDENTIFIER
```

```
--
```

```
-- Protocol Directory Group
```

```
--
```

```
-- Lists the inventory of protocols the probe has the capability of
-- monitoring and allows the addition, deletion, and configuration of
-- entries in this list.
```

```
protocolDirLastChange OBJECT-TYPE
```

```
SYNTAX      TimeStamp
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"The value of sysUpTime at the time the protocol directory was last modified, either through insertions or deletions, or through modifications of either the protocolDirAddressMapConfig, protocolDirHostConfig, or protocolDirMatrixConfig."

```
::= { protocolDir 1 }
```

protocolDirTable OBJECT-TYPE

SYNTAX SEQUENCE OF ProtocolDirEntry
 MAX-ACCESS not-accessible
 STATUS current

DESCRIPTION

"This table lists the protocols that this agent has the capability to decode and count. There is one entry in this table for each such protocol. These protocols represent different network layer, transport layer, and higher-layer protocols. The agent should boot up with this table preconfigured with those protocols that it knows about and wishes to monitor. Implementations are strongly encouraged to support protocols higher than the network layer (at least for the protocol distribution group), even for implementations that don't support the application layer groups."

::= { protocolDir 2 }

protocolDirEntry OBJECT-TYPE

SYNTAX ProtocolDirEntry
 MAX-ACCESS not-accessible
 STATUS current

DESCRIPTION

"A conceptual row in the protocolDirTable.

An example of the indexing of this entry is protocolDirLocalIndex.8.0.0.0.1.0.0.8.0.2.0.0, which is the encoding of a length of 8, followed by 8 subids encoding the protocolDirID of 1.2048, followed by a length of 2 and the 2 subids encoding zero-valued parameters."

INDEX { protocolDirID, protocolDirParameters }
 ::= { protocolDirTable 1 }

```
ProtocolDirEntry ::= SEQUENCE {
  protocolDirID          OCTET STRING,
  protocolDirParameters OCTET STRING,
  protocolDirLocalIndex Integer32,
  protocolDirDescr      DisplayString,
  protocolDirType       BITS,
  protocolDirAddressMapConfig INTEGER,
  protocolDirHostConfig INTEGER,
  protocolDirMatrixConfig INTEGER,
  protocolDirOwner      OwnerString,
  protocolDirStatus     RowStatus
}
```

protocolDirID OBJECT-TYPE

SYNTAX OCTET STRING
 MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A unique identifier for a particular protocol. Standard identifiers will be defined in a manner such that they can often be used as specifications for new protocols - i.e. a tree-structured assignment mechanism that matches the protocol encapsulation 'tree' and which has algorithmic assignment mechanisms for certain subtrees. See RFC XXX for more details.

Despite the algorithmic mechanism, the probe will only place entries in here for those protocols it chooses to collect. In other words, it need not populate this table with all of the possible ethernet protocol types, nor need it create them on the fly when it sees them. Whether or not it does these things is a matter of product definition (cost/benefit, usability), and is up to the designer of the product.

If an entry is written to this table with a protocolDirID that the agent doesn't understand, either directly or algorithmically, the SET request will be rejected with an inconsistentName or badValue (for SNMPv1) error."

::= { protocolDirEntry 1 }

protocolDirParameters OBJECT-TYPE

SYNTAX OCTET STRING
 MAX-ACCESS not-accessible
 STATUS current

DESCRIPTION

"A set of parameters for the associated protocolDirID. See the associated RMON2 Protocol Identifiers document for a description of the possible parameters. There will be one octet in this string for each sub-identifier in the protocolDirID, and the parameters will appear here in the same order as the associated sub-identifiers appear in the protocolDirID.

Every node in the protocolDirID tree has a different, optional set of parameters defined (that is, the definition of parameters for a node is optional). The proper parameter value for each node is included in this string. Note that the inclusion of a parameter value in this string for each node is not optional - what is optional is that a node may have no parameters defined, in which case the parameter field for that node will be zero."

::= { protocolDirEntry 2 }

protocolDirLocalIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The locally arbitrary, but unique identifier associated with this protocolDir entry.

The value for each supported protocol must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization, except that if a protocol is deleted and re-created, it must be re-created with a new value that has not been used since the last re-initialization.

The specific value is meaningful only within a given SNMP entity. A protocolDirLocalIndex must not be re-used until the next agent-restart in the event the protocol directory entry is deleted."

::= { protocolDirEntry 3 }

protocolDirDescr OBJECT-TYPE

SYNTAX DisplayString (SIZE (1..64))

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A textual description of the protocol encapsulation. A probe may choose to describe only a subset of the entire encapsulation (e.g. only the highest layer).

This object is intended for human consumption only.

This object may not be modified if the associated protocolDirStatus object is equal to active(1)."

::= { protocolDirEntry 4 }

protocolDirType OBJECT-TYPE

SYNTAX BITS {
 extensible(0),
 addressRecognitionCapable(1)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object describes 2 attributes of this protocol directory entry.

The presence or absence of the 'extensible' bit describes whether or not this protocol directory entry can be extended

by the user by creating protocol directory entries which are children of this protocol.

An example of an entry that will often allow extensibility is 'ip.udp'. The probe may automatically populate some children of this node such as 'ip.udp.snmp' and 'ip.udp.dns'. A probe administrator or user may also populate additional children via remote SNMP requests that create entries in this table. When a child node is added for a protocol for which the probe has no built in support, extending a parent node (for which the probe does have built in support), that child node is not extendible. This is termed 'limited extensibility'.

When a child node is added through this extensibility mechanism, the values of protocolDirLocalIndex and protocolDirType shall be assigned by the agent.

The other objects in the entry will be assigned by the manager who is creating the new entry.

This object also describes whether or not this agent can recognize addresses for this protocol, should it be a network level protocol. That is, while a probe may be able to recognize packets of a particular network layer protocol and count them, it takes additional logic to be able to recognize the addresses in this protocol and to populate network layer or application layer tables with the addresses in this protocol. If this bit is set, the agent will recognize network layer addresses for this protocol and populate the network and application layer host and matrix tables with these protocols.

Note that when an entry is created, the agent will supply values for the bits that match the capabilities of the agent with respect to this protocol. Note that since row creations usually exercise the limited extensibility feature, these bits will usually be set to zero."

```
::= { protocolDirEntry 5 }
```

```
protocolDirAddressMapConfig OBJECT-TYPE
```

```
SYNTAX      INTEGER {
                notSupported(1),
                supportedOff(2),
                supportedOn(3)
            }
MAX-ACCESS  read-create
STATUS      current
```

DESCRIPTION

"This object describes and configures the probe's support for address mapping for this protocol. When the probe creates entries in this table for all protocols that it understands, it will set the entry to notSupported(1) if it doesn't have the capability to perform address mapping for the protocol or if this protocol is not a network-layer protocol. When an entry is created in this table by a management operation as part of the limited extensibility feature, the probe must set this value to notSupported(1), because limited extensibility of the protocolDirTable does not extend to interpreting addresses of the extended protocols.

If the value of this object is notSupported(1), the probe will not perform address mapping for this protocol and shall not allow this object to be changed to any other value. If the value of this object is supportedOn(3), the probe supports address mapping for this protocol and is configured to perform address mapping for this protocol for all addressMappingControlEntries and all interfaces.

If the value of this object is supportedOff(2), the probe supports address mapping for this protocol but is configured to not perform address mapping for this protocol for any addressMappingControlEntries and all interfaces.

Whenever this value changes from supportedOn(3) to supportedOff(2), the probe shall delete all related entries in the addressMappingTable."

```
::= { protocolDirEntry 6 }
```

protocolDirHostConfig OBJECT-TYPE

```
SYNTAX      INTEGER {
                notSupported(1),
                supportedOff(2),
                supportedOn(3)
            }
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

DESCRIPTION

"This object describes and configures the probe's support for the network layer and application layer host tables for this protocol. When the probe creates entries in this table for all protocols that it understands, it will set the entry to notSupported(1) if it doesn't have the capability to track the nlHostTable for this protocol or if the alHostTable is implemented but doesn't have the capability to track this protocol. Note that if the alHostTable is implemented, the probe may only support a protocol if it is supported in both the nlHostTable and the alHostTable.

If the associated protocolDirType object has the addressRecognitionCapable bit set, then this is a network layer protocol for which the probe recognizes addresses, and thus the probe will populate the nlHostTable and alHostTable with addresses it discovers for this protocol.

If the value of this object is notSupported(1), the probe will not track the nlHostTable or alHostTable for this protocol and shall not allow this object to be changed to any other value. If the value of this object is supportedOn(3), the probe supports tracking of the nlHostTable and alHostTable for this protocol and is configured to track both tables for this protocol for all control entries and all interfaces. If the value of this object is supportedOff(2), the probe supports tracking of the nlHostTable and alHostTable for this protocol but is configured to not track these tables for any control entries or interfaces. Whenever this value changes from supportedOn(3) to supportedOff(2), the probe shall delete all related entries in the nlHostTable and alHostTable.

Note that since each alHostEntry references 2 protocol directory entries, one for the network address and one for the type of the highest protocol recognized, that an entry will only be created in that table if this value is supportedOn(3) for both protocols."

```
::= { protocolDirEntry 7 }
```

protocolDirMatrixConfig OBJECT-TYPE

```
SYNTAX      INTEGER {
                notSupported(1),
                supportedOff(2),
                supportedOn(3)
            }
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

DESCRIPTION

"This object describes and configures the probe's support for the network layer and application layer matrix tables for this protocol. When the probe creates entries in this table for all protocols that it understands, it will set the entry to notSupported(1) if it doesn't have the capability to track the nlMatrixTables for this protocol or if the alMatrixTables are implemented but don't have the capability to track this protocol. Note that if the alMatrix tables are implemented, the probe may only support a protocol if it is supported in the the both of the nlMatrixTables and both of the alMatrixTables.

If the associated protocolDirType object has the addressRecognitionCapable bit set, then this is a network layer protocol for which the probe recognizes addresses, and thus the probe will populate both of the nlMatrixTables and both of the alMatrixTables with addresses it discovers for this protocol.

If the value of this object is notSupported(1), the probe will not track either of the nlMatrixTables or the alMatrixTables for this protocol and shall not allow this object to be changed to any other value. If the value of this object is supportedOn(3), the probe supports tracking of both of the nlMatrixTables and (if implemented) both of the alMatrixTables for this protocol and is configured to track these tables for this protocol for all control entries and all interfaces. If the value of this object is supportedOff(2), the probe supports tracking of both of the nlMatrixTables and (if implemented) both of the alMatrixTables for this protocol but is configured to not track these tables for this protocol for any control entries or interfaces. Whenever this value changes from supportedOn(3) to supportedOff(2), the probe shall delete all related entries in the nlMatrixTables and the alMatrixTables.

Note that since each alMatrixEntry references 2 protocol directory entries, one for the network address and one for the type of the highest protocol recognized, that an entry will only be created in that table if this value is supportedOn(3) for both protocols."

```
::= { protocolDirEntry 8 }
```

```
protocolDirOwner OBJECT-TYPE
```

```
SYNTAX      OwnerString
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The entity that configured this entry and is
  therefore using the resources assigned to it."
```

```
::= { protocolDirEntry 9 }
```

```
protocolDirStatus OBJECT-TYPE
```

```
SYNTAX      RowStatus
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The status of this protocol directory entry.
```

```
An entry may not exist in the active state unless all
```

objects in the entry have an appropriate value.

If this object is not equal to active(1), all associated entries in the nlHostTable, nlMatrixSDTable, nlMatrixDSTable, alHostTable, alMatrixSDTable, and alMatrixDSTable shall be deleted."

```
::= { protocolDirEntry 10 }
```

```
--  
-- Protocol Distribution Group (protocolDist)  
--  
-- Collects the relative amounts of octets and packets for the  
-- different protocols detected on a network segment.  
-- protocolDistControlTable,  
-- protocolDistStatsTable
```

protocolDistControlTable OBJECT-TYPE

SYNTAX SEQUENCE OF ProtocolDistControlEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Controls the setup of protocol type distribution statistics tables.

Implementations are encouraged to add an entry per monitored interface upon initialization so that a default collection of protocol statistics is available.

Rationale:

This table controls collection of very basic statistics for any or all of the protocols detected on a given interface. An NMS can use this table to quickly determine bandwidth allocation utilized by different protocols.

A media-specific statistics collection could also be configured (e.g. etherStats, trPStats) to easily obtain total frame, octet, and droppedEvents for the same interface."

```
::= { protocolDist 1 }
```

protocolDistControlEntry OBJECT-TYPE

SYNTAX ProtocolDistControlEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the protocolDistControlTable.

An example of the indexing of this entry is

```

        protocolDistControlDroppedFrames.7"
INDEX { protocolDistControlIndex }
 ::= { protocolDistControlTable 1 }

```

```

ProtocolDistControlEntry ::= SEQUENCE {
    protocolDistControlIndex          Integer32,
    protocolDistControlDataSource     DataSource,
    protocolDistControlDroppedFrames Counter32,
    protocolDistControlCreateTime     LastCreateTime,
    protocolDistControlOwner          OwnerString,
    protocolDistControlStatus         RowStatus
}

```

```

protocolDistControlIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique index for this protocolDistControlEntry."
    ::= { protocolDistControlEntry 1 }

```

```

protocolDistControlDataSource OBJECT-TYPE
    SYNTAX      DataSource
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The source of data for the this protocol distribution.

        The statistics in this group reflect all packets
        on the local network segment attached to the
        identified interface.

        This object may not be modified if the associated
        protocolDistControlStatus object is equal to active(1)."
    ::= { protocolDistControlEntry 2 }

```

```

protocolDistControlDroppedFrames OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of frames which were received by the probe
        and therefore not accounted for in the *StatsDropEvents, but
        for which the probe chose not to count for this entry for
        whatever reason.  Most often, this event occurs when the probe
        is out of some resources and decides to shed load from this
        collection."

```

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { protocolDistControlEntry 3 }
```

```
protocolDistControlCreateTime OBJECT-TYPE
```

```
SYNTAX      LastCreateTime
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"The value of sysUpTime when this control entry was last activated. This can be used by the management station to ensure that the table has not been deleted and recreated between polls."

```
::= { protocolDistControlEntry 4 }
```

```
protocolDistControlOwner OBJECT-TYPE
```

```
SYNTAX      OwnerString
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"The entity that configured this entry and is therefore using the resources assigned to it."

```
::= { protocolDistControlEntry 5 }
```

```
protocolDistControlStatus OBJECT-TYPE
```

```
SYNTAX      RowStatus
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"The status of this row.

An entry may not exist in the active state unless all objects in the entry have an appropriate value.

If this object is not equal to active(1), all associated entries in the protocolDistStatsTable shall be deleted."

```
::= { protocolDistControlEntry 6 }
```

```
-- per interface protocol distribution statistics table
```

```
protocolDistStatsTable OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF ProtocolDistStatsEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

"An entry is made in this table for every protocol in the

protocolDirTable which has been seen in at least one packet. Counters are updated in this table for every protocol type that is encountered when parsing a packet, but no counters are updated for packets with MAC-layer errors.

Note that if a protocolDirEntry is deleted, all associated entries in this table are removed."

```
::= { protocolDist 2 }
```

protocolDistStatsEntry OBJECT-TYPE

SYNTAX ProtocolDistStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the protocolDistStatsTable.

The index is composed of the protocolDistControlIndex of the associated protocolDistControlEntry followed by the protocolDirLocalIndex of the associated protocol that this entry represents. In other words, the index identifies the protocol distribution an entry is a part of as well as the particular protocol that it represents.

An example of the indexing of this entry is

protocolDistStatsPkts.1.18"

```
INDEX { protocolDistControlIndex, protocolDirLocalIndex }
```

```
::= { protocolDistStatsTable 1 }
```

```
ProtocolDistStatsEntry ::= SEQUENCE {
```

```
    protocolDistStatsPkts
```

```
                                ZeroBasedCounter32,
```

```
    protocolDistStatsOctets
```

```
                                ZeroBasedCounter32
```

```
}
```

protocolDistStatsPkts OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets without errors received of this protocol type. Note that this is the number of link-layer packets, so if a single network-layer packet is fragmented into several link-layer frames, this counter is incremented several times."

```
::= { protocolDistStatsEntry 1 }
```

protocolDistStatsOctets OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

```
STATUS      current
DESCRIPTION
    "The number of octets in packets received of this protocol
    type since it was added to the protocolDistStatsTable
    (excluding framing bits but including FCS octets), except for
    those octets in packets that contained errors.

    Note this doesn't count just those octets in the particular
    protocol frames, but includes the entire packet that contained
    the protocol."
 ::= { protocolDistStatsEntry 2 }

--
-- Address Map Group      (addressMap)
--
-- Lists MAC address to network address bindings discovered by the
-- probe and what interface they were last seen on.
--   addressMapControlTable
--   addressMapTable

addressMapInserts OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The number of times an address mapping entry has been
    inserted into the addressMapTable.  If an entry is inserted,
    then deleted, and then inserted, this counter will be
    incremented by 2.

    Note that the table size can be determined by subtracting
    addressMapDeletes from addressMapInserts."
 ::= { addressMap 1 }

addressMapDeletes OBJECT-TYPE
SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The number of times an address mapping entry has been
    deleted from the addressMapTable (for any reason).  If
    an entry is deleted, then inserted, and then deleted, this
    counter will be incremented by 2.

    Note that the table size can be determined by subtracting
    addressMapDeletes from addressMapInserts."
 ::= { addressMap 2 }
```

addressMapMaxDesiredEntries OBJECT-TYPE

SYNTAX Integer32 (-1..2147483647)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The maximum number of entries that are desired in the addressMapTable. The probe will not create more than this number of entries in the table, but may choose to create fewer entries in this table for any reason including the lack of resources.

If this object is set to a value less than the current number of entries, enough entries are chosen in an implementation-dependent manner and deleted so that the number of entries in the table equals the value of this object.

If this value is set to -1, the probe may create any number of entries in this table.

This object may be used to control how resources are allocated on the probe for the various RMON functions."

::= { addressMap 3 }

addressMapControlTable OBJECT-TYPE

SYNTAX SEQUENCE OF AddressMapControlEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table to control the collection of network layer address to physical address to interface mappings.

Note that this is not like the typical RMON controlTable and dataTable in which each entry creates its own data table. Each entry in this table enables the discovery of addresses on a new interface and the placement of address mappings into the central addressMapTable.

Implementations are encouraged to add an entry per monitored interface upon initialization so that a default collection of address mappings is available."

::= { addressMap 4 }

addressMapControlEntry OBJECT-TYPE

SYNTAX AddressMapControlEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the addressMapControlTable.

An example of the indexing of this entry is
addressMapControlDroppedFrames.1"

```
INDEX { addressMapControlIndex }
 ::= { addressMapControlTable 1 }
```

```
AddressMapControlEntry ::= SEQUENCE {
    addressMapControlIndex          Integer32,
    addressMapControlDataSource     DataSource,
    addressMapControlDroppedFrames Counter32,
    addressMapControlOwner         OwnerString,
    addressMapControlStatus        RowStatus
}
```

```
addressMapControlIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique index for this entry in the addressMapControlTable."
    ::= { addressMapControlEntry 1 }
```

```
addressMapControlDataSource OBJECT-TYPE
    SYNTAX      DataSource
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The source of data for this addressMapControlEntry."
    ::= { addressMapControlEntry 2 }
```

```
addressMapControlDroppedFrames OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of frames which were received by the probe
        and therefore not accounted for in the *StatsDropEvents, but
        for which the probe chose not to count for this entry for
        whatever reason. Most often, this event occurs when the probe
        is out of some resources and decides to shed load from this
        collection.
```

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { addressMapControlEntry 3 }
```

addressMapControlOwner OBJECT-TYPE

SYNTAX OwnerString

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The entity that configured this entry and is therefore using the resources assigned to it."

::= { addressMapControlEntry 4 }

addressMapControlStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The status of this addressMap control entry.

An entry may not exist in the active state unless all objects in the entry have an appropriate value.

If this object is not equal to active(1), all associated entries in the addressMapTable shall be deleted."

::= { addressMapControlEntry 5 }

addressMapTable OBJECT-TYPE

SYNTAX SEQUENCE OF AddressMapEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table of network layer address to physical address to interface mappings.

The probe will add entries to this table based on the source MAC and network addresses seen in packets without MAC-level errors. The probe will populate this table for all protocols in the protocol directory table whose value of protocolDirAddressMapConfig is equal to supportedOn(3), and will delete any entries whose protocolDirEntry is deleted or has a protocolDirAddressMapConfig value of supportedOff(2)."

::= { addressMap 5 }

addressMapEntry OBJECT-TYPE

SYNTAX AddressMapEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the addressMapTable.

The protocolDirLocalIndex in the index identifies the network layer protocol of the addressMapNetworkAddress.

An example of the indexing of this entry is
 addressMapSource.783495.18.4.128.2.6.6.11.1.3.6.1.2.1.2.2.1.1.1"
 INDEX { addressMapTimeMark, protocolDirLocalIndex,
 addressMapNetworkAddress, addressMapSource }
 ::= { addressMapTable 1 }

```
AddressMapEntry ::= SEQUENCE {
    addressMapTimeMark          TimeFilter,
    addressMapNetworkAddress    OCTET STRING,
    addressMapSource            OBJECT IDENTIFIER,
    addressMapPhysicalAddress   OCTET STRING,
    addressMapLastChange        TimeStamp
}
```

```
addressMapTimeMark OBJECT-TYPE
    SYNTAX          TimeFilter
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "A TimeFilter for this entry.  See the TimeFilter textual
        convention to see how this works."
    ::= { addressMapEntry 1 }
```

```
addressMapNetworkAddress OBJECT-TYPE
    SYNTAX          OCTET STRING
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "The network address for this relation.
```

This is represented as an octet string with specific semantics and length as identified by the protocolDirLocalIndex component of the index.

For example, if the protocolDirLocalIndex indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

```
::= { addressMapEntry 2 }
```

```
addressMapSource OBJECT-TYPE
    SYNTAX          OBJECT IDENTIFIER
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "The interface or port on which the associated network
        address was most recently seen.
```

If this address mapping was discovered on an interface, this object shall identify the instance of the ifIndex object, defined in [3,5], for the desired interface. For example, if an entry were to receive data from interface #1, this object would be set to ifIndex.1.

If this address mapping was discovered on a port, this object shall identify the instance of the rptrGroupPortIndex object, defined in [RFC1516], for the desired port. For example, if an entry were to receive data from group #1, port #1, this object would be set to rptrGroupPortIndex.1.1.

Note that while the dataSource associated with this entry may only point to index objects, this object may at times point to repeater port objects. This situation occurs when the dataSource points to an interface which is a locally attached repeater and the agent has additional information about the source port of traffic seen on that repeater."

```
::= { addressMapEntry 3 }
```

addressMapPhysicalAddress OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The last source physical address on which the associated network address was seen. If the protocol of the associated network address was encapsulated inside of a network-level or higher protocol, this will be the address of the next-lower protocol with the addressRecognitionCapable bit enabled and will be formatted as specified for that protocol."

```
::= { addressMapEntry 4 }
```

addressMapLastChange OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at the time this entry was last created or the values of the physical address changed.

This can be used to help detect duplicate address problems, in which case this object will be updated frequently."

```
::= { addressMapEntry 5 }
```

--

-- Network Layer Host Group

```
--
-- Counts the amount of traffic sent from and to each network address
-- discovered by the probe.
-- Note that while the hlHostControlTable also has objects that
-- control an optional alHostTable, implementation of the alHostTable is
-- not required to fully implement this group.
```

hlHostControlTable OBJECT-TYPE

```
SYNTAX      SEQUENCE OF HlHostControlEntry
MAX-ACCESS  not-accessible
STATUS      current
```

DESCRIPTION

"A list of higher layer (i.e. non-MAC) host table control entries.

These entries will enable the collection of the network and application level host tables indexed by network addresses. Both the network and application level host tables are controlled by this table is so that they will both be created and deleted at the same time, further increasing the ease with which they can be implemented as a single datastore (note that if an implementation stores application layer host records in memory, it can derive network layer host records from them).

Entries in the nlHostTable will be created on behalf of each entry in this table. Additionally, if this probe implements the alHostTable, entries in the alHostTable will be created on behalf of each entry in this table.

Implementations are encouraged to add an entry per monitored interface upon initialization so that a default collection of host statistics is available."

```
::= { nlHost 1 }
```

hlHostControlEntry OBJECT-TYPE

```
SYNTAX      HlHostControlEntry
MAX-ACCESS  not-accessible
STATUS      current
```

DESCRIPTION

"A conceptual row in the hlHostControlTable.

An example of the indexing of this entry is
hlHostControlNlDroppedFrames.1"

```
INDEX { hlHostControlIndex }
::= { hlHostControlTable 1 }
```

```
HlHostControlEntry ::= SEQUENCE {
    hlHostControlIndex      Integer32,
    hlHostControlDataSource DataSource,
```

```

hlHostControlNlDroppedFrames      Counter32,
hlHostControlNlInserts            Counter32,
hlHostControlNlDeletes            Counter32,
hlHostControlNlMaxDesiredEntries  Integer32,
hlHostControlAlDroppedFrames      Counter32,
hlHostControlAlInserts            Counter32,
hlHostControlAlDeletes            Counter32,
hlHostControlAlMaxDesiredEntries  Integer32,
hlHostControlOwner                OwnerString,
hlHostControlStatus                RowStatus
}

```

hlHostControlIndex OBJECT-TYPE

```

SYNTAX      Integer32 (1..65535)
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION

```

"An index that uniquely identifies an entry in the hlHostControlTable. Each such entry defines a function that discovers hosts on a particular interface and places statistics about them in the nlHostTable, and optionally in the alHostTable, on behalf of this hlHostControlEntry."

```
 ::= { hlHostControlEntry 1 }
```

hlHostControlDataSource OBJECT-TYPE

```

SYNTAX      DataSource
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION

```

"The source of data for the associated host tables.

The statistics in this group reflect all packets on the local network segment attached to the identified interface.

This object may not be modified if the associated hlHostControlStatus object is equal to active(1)."

```
 ::= { hlHostControlEntry 2 }
```

hlHostControlNlDroppedFrames OBJECT-TYPE

```

SYNTAX      Counter32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION

```

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for the associated

nlHost entries for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that if the nlHostTable is inactive because no protocols are enabled in the protocol directory, this value should be 0.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { hlHostControlEntry 3 }
```

hlHostControlNlInserts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an nlHost entry has been inserted into the nlHost table. If an entry is inserted, then deleted, and then inserted, this counter will be incremented by 2.

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal data structures for those short periods of time.

Note that the table size can be determined by subtracting hlHostControlNlDeletes from hlHostControlNlInserts."

```
::= { hlHostControlEntry 4 }
```

hlHostControlNlDeletes OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an nlHost entry has been deleted from the nlHost table (for any reason). If an entry is deleted, then inserted, and then deleted, this counter will be incremented by 2.

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal

data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal data structures for those short periods of time.

Note that the table size can be determined by subtracting hlHostControlNlDeletes from hlHostControlNlInserts."

```
::= { hlHostControlEntry 5 }
```

hlHostControlNlMaxDesiredEntries OBJECT-TYPE

SYNTAX Integer32 (-1..2147483647)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The maximum number of entries that are desired in the nlHostTable on behalf of this control entry. The probe will not create more than this number of associated entries in the table, but may choose to create fewer entries in this table for any reason including the lack of resources.

If this object is set to a value less than the current number of entries, enough entries are chosen in an implementation-dependent manner and deleted so that the number of entries in the table equals the value of this object.

If this value is set to -1, the probe may create any number of entries in this table. If the associated hlHostControlStatus object is equal to 'active', this object may not be modified.

This object may be used to control how resources are allocated on the probe for the various RMON functions."

```
::= { hlHostControlEntry 6 }
```

hlHostControlAlDroppedFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for the associated alHost entries for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that if the alHostTable is not implemented or is inactive because no protocols are enabled in the protocol directory, this value should be 0.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { hlHostControlEntry 7 }
```

hlHostControlAlInserts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an alHost entry has been inserted into the alHost table. If an entry is inserted, then deleted, and then inserted, this counter will be incremented by 2.

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal data structures for those short periods of time.

Note that the table size can be determined by subtracting hlHostControlAlDeletes from hlHostControlAlInserts."

```
::= { hlHostControlEntry 8 }
```

hlHostControlAlDeletes OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an alHost entry has been deleted from the alHost table (for any reason). If an entry is deleted, then inserted, and then deleted, this counter will be incremented by 2.

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal data structures for those short periods of time.

Note that the table size can be determined by subtracting hlHostControlAlDeletes from hlHostControlAlInserts."

```
::= { hlHostControlEntry 9 }
```

```
hlHostControlAlMaxDesiredEntries OBJECT-TYPE
```

```
SYNTAX      Integer32 (-1..2147483647)
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"The maximum number of entries that are desired in the alHost table on behalf of this control entry. The probe will not create more than this number of associated entries in the table, but may choose to create fewer entries in this table for any reason including the lack of resources.

If this object is set to a value less than the current number of entries, enough entries are chosen in an implementation-dependent manner and deleted so that the number of entries in the table equals the value of this object.

If this value is set to -1, the probe may create any number of entries in this table. If the associated hlHostControlStatus object is equal to 'active', this object may not be modified.

This object may be used to control how resources are allocated on the probe for the various RMON functions."

```
::= { hlHostControlEntry 10 }
```

```
hlHostControlOwner OBJECT-TYPE
```

```
SYNTAX      OwnerString
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"The entity that configured this entry and is therefore using the resources assigned to it."

```
::= { hlHostControlEntry 11 }
```

```
hlHostControlStatus OBJECT-TYPE
```

```
SYNTAX      RowStatus
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"The status of this hlHostControlEntry.

An entry may not exist in the active state unless all objects in the entry have an appropriate value.

If this object is not equal to active(1), all associated entries in the nlHostTable and alHostTable shall be deleted."

```
::= { hlHostControlEntry 12 }
```

```
nlHostTable OBJECT-TYPE
```

```
SYNTAX SEQUENCE OF NlHostEntry
```

```
MAX-ACCESS not-accessible
```

```
STATUS current
```

```
DESCRIPTION
```

"A collection of statistics for a particular network layer address that has been discovered on an interface of this device.

The probe will populate this table for all network layer protocols in the protocol directory table whose value of protocolDirHostConfig is equal to supportedOn(3), and will delete any entries whose protocolDirEntry is deleted or has a protocolDirHostConfig value of supportedOff(2).

The probe will add to this table all addresses seen as the source or destination address in all packets with no MAC errors, and will increment octet and packet counts in the table for all packets with no MAC errors."

```
::= { nlHost 2 }
```

```
nlHostEntry OBJECT-TYPE
```

```
SYNTAX NlHostEntry
```

```
MAX-ACCESS not-accessible
```

```
STATUS current
```

```
DESCRIPTION
```

"A conceptual row in the nlHostTable.

The hlHostControlIndex value in the index identifies the hlHostControlEntry on whose behalf this entry was created. The protocolDirLocalIndex value in the index identifies the network layer protocol of the nlHostAddress.

An example of the indexing of this entry is nlHostOutPkts.1.783495.18.4.128.2.6.6."

```
INDEX { hlHostControlIndex, nlHostTimeMark,
        protocolDirLocalIndex, nlHostAddress }
```

```
::= { nlHostTable 1 }
```

```
NlHostEntry ::= SEQUENCE {
```

```
  nlHostTimeMark          TimeFilter,
  nlHostAddress           OCTET STRING,
  nlHostInPkts            ZeroBasedCounter32,
  nlHostOutPkts           ZeroBasedCounter32,
  nlHostInOctets          ZeroBasedCounter32,
  nlHostOutOctets         ZeroBasedCounter32,
```

```
nlHostOutMacNonUnicastPkts  ZeroBasedCounter32,
nlHostCreateTime            LastCreateTime
}

nlHostTimeMark OBJECT-TYPE
    SYNTAX      TimeFilter
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A TimeFilter for this entry.  See the TimeFilter textual
        convention to see how this works."
    ::= { nlHostEntry 1 }

nlHostAddress OBJECT-TYPE
    SYNTAX      OCTET STRING
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The network address for this nlHostEntry.

        This is represented as an octet string with
        specific semantics and length as identified
        by the protocolDirLocalIndex component of the index.

        For example, if the protocolDirLocalIndex indicates an
        encapsulation of ip, this object is encoded as a length
        octet of 4, followed by the 4 octets of the ip address,
        in network byte order."
    ::= { nlHostEntry 2 }

nlHostInPkts OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of packets without errors transmitted to
        this address since it was added to the nlHostTable.  Note that
        this is the number of link-layer packets, so if a single
        network-layer packet is fragmented into several link-layer
        frames, this counter is incremented several times."
    ::= { nlHostEntry 3 }

nlHostOutPkts OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of packets without errors transmitted by
```

this address since it was added to the nlHostTable. Note that this is the number of link-layer packets, so if a single network-layer packet is fragmented into several link-layer frames, this counter is incremented several times."

::= { nlHostEntry 4 }

nlHostInOctets OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets transmitted to this address since it was added to the nlHostTable (excluding framing bits but including FCS octets), excluding those octets in packets that contained errors.

Note this doesn't count just those octets in the particular protocol frames, but includes the entire packet that contained the protocol."

::= { nlHostEntry 5 }

nlHostOutOctets OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets transmitted by this address since it was added to the nlHostTable (excluding framing bits but including FCS octets), excluding those octets in packets that contained errors.

Note this doesn't count just those octets in the particular protocol frames, but includes the entire packet that contained the protocol."

::= { nlHostEntry 6 }

nlHostOutMacNonUnicastPkts OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets without errors transmitted by this address that were directed to any MAC broadcast addresses or to any MAC multicast addresses since this host was added to the nlHostTable. Note that this is the number of link-layer packets, so if a single network-layer packet is fragmented into several link-layer frames, this counter is incremented several times."

```
::= { nlHostEntry 7 }
```

```
nlHostCreateTime OBJECT-TYPE
```

```
SYNTAX      LastCreateTime
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The value of sysUpTime when this entry was last activated.
This can be used by the management station to ensure that the
entry has not been deleted and recreated between polls."
```

```
::= { nlHostEntry 8 }
```

```
--
```

```
-- Network Layer Matrix Group
```

```
--
```

```
-- Counts the amount of traffic sent between each pair of network
-- addresses discovered by the probe.
```

```
-- Note that while the hlMatrixControlTable also has objects that
-- control optional alMatrixTables, implementation of the
-- alMatrixTables is not required to fully implement this group.
```

```
hlMatrixControlTable OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF HlMatrixControlEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"A list of higher layer (i.e. non-MAC) matrix control entries.
```

```
These entries will enable the collection of the network and
application level matrix tables containing conversation
statistics indexed by pairs of network addresses.
Both the network and application level matrix tables are
controlled by this table is so that they will both be created
and deleted at the same time, further increasing the ease with
which they can be implemented as a single datastore (note that
if an implementation stores application layer matrix records
in memory, it can derive network layer matrix records from
them).
```

```
Entries in the nlMatrixSDTable and nlMatrixDSTable will be
created on behalf of each entry in this table.  Additionally,
if this probe implements the alMatrix tables, entries in the
alMatrix tables will be created on behalf of each entry in
this table."
```

```
::= { nlMatrix 1 }
```

```
hlMatrixControlEntry OBJECT-TYPE
```

```
SYNTAX      HlMatrixControlEntry
```

```

MAX-ACCESS    not-accessible
STATUS        current
DESCRIPTION
    "A conceptual row in the hlMatrixControlTable.

```

```

    An example of indexing of this entry is
    hlMatrixControlNlDroppedFrames.1"

```

```

INDEX { hlMatrixControlIndex }
 ::= { hlMatrixControlTable 1 }

```

```

HlMatrixControlEntry ::= SEQUENCE {
    hlMatrixControlIndex          Integer32,
    hlMatrixControlDataSource     DataSource,
    hlMatrixControlNlDroppedFrames Counter32,
    hlMatrixControlNlInserts      Counter32,
    hlMatrixControlNlDeletes      Counter32,
    hlMatrixControlNlMaxDesiredEntries Integer32,
    hlMatrixControlAlDroppedFrames Counter32,
    hlMatrixControlAlInserts      Counter32,
    hlMatrixControlAlDeletes      Counter32,
    hlMatrixControlAlMaxDesiredEntries Integer32,
    hlMatrixControlOwner         OwnerString,
    hlMatrixControlStatus        RowStatus
}

```

```

hlMatrixControlIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An index that uniquely identifies an entry in the
        hlMatrixControlTable. Each such entry defines
        a function that discovers conversations on a particular
        interface and places statistics about them in the
        nlMatrixSDTable and the nlMatrixDSTable, and optionally the
        alMatrixSDTable and alMatrixDSTable, on behalf of this
        hlMatrixControlEntry."
    ::= { hlMatrixControlEntry 1 }

```

```

hlMatrixControlDataSource OBJECT-TYPE
    SYNTAX      DataSource
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The source of the data for the associated matrix tables.

        The statistics in this group reflect all packets
        on the local network segment attached to the

```

identified interface.

This object may not be modified if the associated
hlMatrixControlStatus object is equal to active(1)."

```
::= { hlMatrixControlEntry 2 }
```

hlMatrixControlNlDroppedFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for this entry for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that if the nlMatrixTables are inactive because no protocols are enabled in the protocol directory, this value should be 0.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { hlMatrixControlEntry 3 }
```

hlMatrixControlNlInserts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an nlMatrix entry has been inserted into the nlMatrix tables. If an entry is inserted, then deleted, and then inserted, this counter will be incremented by 2. The addition of a conversation into both the nlMatrixSDTable and nlMatrixDSTable shall be counted as two insertions (even though every addition into one table must be accompanied by an insertion into the other).

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal data structures for those short periods of time.

Note that the sum of then nlMatrixSDTable and nlMatrixDSTable sizes can be determined by subtracting hlMatrixControlNlDeletes from hlMatrixControlNlInserts."

::= { hlMatrixControlEntry 4 }

hlMatrixControlNlDeletes OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an nlMatrix entry has been deleted from the nlMatrix tables (for any reason). If an entry is deleted, then inserted, and then deleted, this counter will be incremented by 2. The deletion of a conversation from both the nlMatrixSDTable and nlMatrixDSTable shall be counted as two deletions (even though every deletion from one table must be accompanied by a deletion from the other).

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal data structures for those short periods of time.

Note that the table size can be determined by subtracting hlMatrixControlNlDeletes from hlMatrixControlNlInserts."

::= { hlMatrixControlEntry 5 }

hlMatrixControlNlMaxDesiredEntries OBJECT-TYPE

SYNTAX Integer32 (-1..2147483647)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The maximum number of entries that are desired in the nlMatrix tables on behalf of this control entry. The probe will not create more than this number of associated entries in the table, but may choose to create fewer entries in this table for any reason including the lack of resources.

If this object is set to a value less than the current number of entries, enough entries are chosen in an implementation-dependent manner and deleted so that the number of entries in the table equals the value of this object.

If this value is set to -1, the probe may create any number of entries in this table. If the associated

hlMatrixControlStatus object is equal to 'active', this object may not be modified.

This object may be used to control how resources are allocated on the probe for the various RMON functions."

```
::= { hlMatrixControlEntry 6 }
```

hlMatrixControlAlDroppedFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for this entry for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that if the alMatrixTables are not implemented or are inactive because no protocols are enabled in the protocol directory, this value should be 0.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { hlMatrixControlEntry 7 }
```

hlMatrixControlAlInserts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an alMatrix entry has been inserted into the alMatrix tables. If an entry is inserted, then deleted, and then inserted, this counter will be incremented by 2. The addition of a conversation into both the alMatrixSDTable and alMatrixDSTable shall be counted as two insertions (even though every addition into one table must be accompanied by an insertion into the other).

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal

data structures for those short periods of time.

Note that the table size can be determined by subtracting hlMatrixControlAlDeletes from hlMatrixControlAlInserts."

```
::= { hlMatrixControlEntry 8 }
```

hlMatrixControlAlDeletes OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times an alMatrix entry has been deleted from the alMatrix tables. If an entry is deleted, then inserted, and then deleted, this counter will be incremented by 2. The deletion of a conversation from both the alMatrixSDTable and alMatrixDSTable shall be counted as two deletions (even though every deletion from one table must be accompanied by a deletion from the other).

To allow for efficient implementation strategies, agents may delay updating this object for short periods of time. For example, an implementation strategy may allow internal data structures to differ from those visible via SNMP for short periods of time. This counter may reflect the internal data structures for those short periods of time.

Note that the table size can be determined by subtracting hlMatrixControlAlDeletes from hlMatrixControlAlInserts."

```
::= { hlMatrixControlEntry 9 }
```

hlMatrixControlAlMaxDesiredEntries OBJECT-TYPE

SYNTAX Integer32 (-1..2147483647)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The maximum number of entries that are desired in the alMatrix tables on behalf of this control entry. The probe will not create more than this number of associated entries in the table, but may choose to create fewer entries in this table for any reason including the lack of resources.

If this object is set to a value less than the current number of entries, enough entries are chosen in an implementation-dependent manner and deleted so that the number of entries in the table equals the value of this object.

If this value is set to -1, the probe may create any number of entries in this table. If the associated

hlMatrixControlStatus object is equal to 'active', this object may not be modified.

This object may be used to control how resources are allocated on the probe for the various RMON functions."

```
::= { hlMatrixControlEntry 10 }
```

hlMatrixControlOwner OBJECT-TYPE

SYNTAX OwnerString

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The entity that configured this entry and is therefore using the resources assigned to it."

```
::= { hlMatrixControlEntry 11 }
```

hlMatrixControlStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The status of this hlMatrixControlEntry.

An entry may not exist in the active state unless all objects in the entry have an appropriate value.

If this object is not equal to active(1), all associated entries in the nlMatrixSDTable, nlMatrixDSTable, alMatrixSDTable, and the alMatrixDSTable shall be deleted by the agent."

```
::= { hlMatrixControlEntry 12 }
```

nlMatrixSDTable OBJECT-TYPE

SYNTAX SEQUENCE OF NlMatrixSDEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A list of traffic matrix entries which collect statistics for conversations between two network-level addresses. This table is indexed first by the source address and then by the destination address to make it convenient to collect all conversations from a particular address.

The probe will populate this table for all network layer protocols in the protocol directory table whose value of protocolDirMatrixConfig is equal to supportedOn(3), and will delete any entries whose protocolDirEntry is deleted or has a protocolDirMatrixConfig value of supportedOff(2).

The probe will add to this table all pairs of addresses seen in all packets with no MAC errors, and will increment octet and packet counts in the table for all packets with no MAC errors.

Further, this table will only contain entries that have a corresponding entry in the nlMatrixDSTable with the same source address and destination address."

```
::= { nlMatrix 2 }
```

```
nlMatrixSDEntry OBJECT-TYPE
    SYNTAX      NlMatrixSDEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
```

```
    "A conceptual row in the nlMatrixSDTable.
```

```
    The hlMatrixControlIndex value in the index identifies the
    hlMatrixControlEntry on whose behalf this entry was created.
    The protocolDirLocalIndex value in the index identifies the
    network layer protocol of the nlMatrixSDSourceAddress and
    nlMatrixSDDestAddress.
```

```
    An example of the indexing of this table is
    nlMatrixSDPkts.1.783495.18.4.128.2.6.6.4.128.2.6.7"
```

```
INDEX { hlMatrixControlIndex, nlMatrixSDTimeMark,
        protocolDirLocalIndex,
        nlMatrixSDSourceAddress, nlMatrixSDDestAddress }
 ::= { nlMatrixSDTable 1 }
```

```
NlMatrixSDEntry ::= SEQUENCE {
    nlMatrixSDTimeMark          TimeFilter,
    nlMatrixSDSourceAddress     OCTET STRING,
    nlMatrixSDDestAddress       OCTET STRING,
    nlMatrixSDPkts              ZeroBasedCounter32,
    nlMatrixSDOctets            ZeroBasedCounter32,
    nlMatrixSDCreateTime       LastCreateTime
}
```

```
nlMatrixSDTimeMark OBJECT-TYPE
    SYNTAX      TimeFilter
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
```

```
    "A TimeFilter for this entry. See the TimeFilter textual
    convention to see how this works."
```

```
::= { nlMatrixSDEntry 1 }
```

nlMatrixSDSourceAddress OBJECT-TYPE

SYNTAX OCTET STRING
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION

"The network source address for this nlMatrixSDEntry.

This is represented as an octet string with specific semantics and length as identified by the protocolDirLocalIndex component of the index.

For example, if the protocolDirLocalIndex indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

::= { nlMatrixSDEntry 2 }

nlMatrixSDDestAddress OBJECT-TYPE

SYNTAX OCTET STRING
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION

"The network destination address for this nlMatrixSDEntry.

This is represented as an octet string with specific semantics and length as identified by the protocolDirLocalIndex component of the index.

For example, if the protocolDirLocalIndex indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

::= { nlMatrixSDEntry 3 }

nlMatrixSDPkts OBJECT-TYPE

SYNTAX ZeroBasedCounter32
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"The number of packets without errors transmitted from the source address to the destination address since this entry was added to the nlMatrixSDTable. Note that this is the number of link-layer packets, so if a single network-layer packet is fragmented into several link-layer frames, this counter is incremented several times."

::= { nlMatrixSDEntry 4 }

nlMatrixSDOctets OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets transmitted from the source address to the destination address since this entry was added to the nlMatrixSDTable (excluding framing bits but including FCS octets), excluding those octets in packets that contained errors.

Note this doesn't count just those octets in the particular protocol frames, but includes the entire packet that contained the protocol."

::= { nlMatrixSDEntry 5 }

nlMatrixSDCreateTime OBJECT-TYPE

SYNTAX LastCreateTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime when this entry was last activated.

This can be used by the management station to ensure that the entry has not been deleted and recreated between polls."

::= { nlMatrixSDEntry 6 }

-- Traffic matrix tables from destination to source

nlMatrixDSTable OBJECT-TYPE

SYNTAX SEQUENCE OF NlMatrixDSEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A list of traffic matrix entries which collect statistics for conversations between two network-level addresses. This table is indexed first by the destination address and then by the source address to make it convenient to collect all conversations to a particular address.

The probe will populate this table for all network layer protocols in the protocol directory table whose value of protocolDirMatrixConfig is equal to supportedOn(3), and will delete any entries whose protocolDirEntry is deleted or has a protocolDirMatrixConfig value of supportedOff(2).

The probe will add to this table all pairs of addresses seen in all packets with no MAC errors, and will increment

octet and packet counts in the table for all packets with no MAC errors.

Further, this table will only contain entries that have a corresponding entry in the nlMatrixSDTable with the same source address and destination address."

```
::= { nlMatrix 3 }
```

```
nlMatrixDSEntry OBJECT-TYPE
    SYNTAX      NlMatrixDSEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
```

"A conceptual row in the nlMatrixDSTable.

The hlMatrixControlIndex value in the index identifies the hlMatrixControlEntry on whose behalf this entry was created. The protocolDirLocalIndex value in the index identifies the network layer protocol of the nlMatrixDSSourceAddress and nlMatrixDSDestAddress.

An example of the indexing of this table is
nlMatrixDSPkts.1.783495.18.4.128.2.6.7.4.128.2.6.6"

```
INDEX { hlMatrixControlIndex, nlMatrixDSTimeMark,
        protocolDirLocalIndex,
        nlMatrixDSDestAddress, nlMatrixDSSourceAddress }
 ::= { nlMatrixDSTable 1 }
```

```
NlMatrixDSEntry ::= SEQUENCE {
    nlMatrixDSTimeMark          TimeFilter,
    nlMatrixDSSourceAddress    OCTET STRING,
    nlMatrixDSDestAddress      OCTET STRING,
    nlMatrixDSPkts             ZeroBasedCounter32,
    nlMatrixDSOctets           ZeroBasedCounter32,
    nlMatrixDSCreateTime       LastCreateTime
}
```

```
nlMatrixDSTimeMark OBJECT-TYPE
    SYNTAX      TimeFilter
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A TimeFilter for this entry. See the TimeFilter textual
        convention to see how this works."
    ::= { nlMatrixDSEntry 1 }
```

```
nlMatrixDSSourceAddress OBJECT-TYPE
    SYNTAX      OCTET STRING
```

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The network source address for this nlMatrixDSEntry.

This is represented as an octet string with specific semantics and length as identified by the protocolDirLocalIndex component of the index.

For example, if the protocolDirLocalIndex indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

::= { nlMatrixDSEntry 2 }

nlMatrixDSDestAddress OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The network destination address for this nlMatrixDSEntry.

This is represented as an octet string with specific semantics and length as identified by the protocolDirLocalIndex component of the index.

For example, if the protocolDirLocalIndex indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

::= { nlMatrixDSEntry 3 }

nlMatrixDSPkts OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets without errors transmitted from the source address to the destination address since this entry was added to the nlMatrixDSTable. Note that this is the number of link-layer packets, so if a single network-layer packet is fragmented into several link-layer frames, this counter is incremented several times."

::= { nlMatrixDSEntry 4 }

nlMatrixDSOctets OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only
 STATUS current
 DESCRIPTION

"The number of octets transmitted from the source address to the destination address since this entry was added to the nlMatrixDSTable (excluding framing bits but including FCS octets), excluding those octets in packets that contained errors.

Note this doesn't count just those octets in the particular protocol frames, but includes the entire packet that contained the protocol."

::= { nlMatrixDSEntry 5 }

nlMatrixDSCreateTime OBJECT-TYPE

SYNTAX LastCreateTime
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION

"The value of sysUpTime when this entry was last activated. This can be used by the management station to ensure that the entry has not been deleted and recreated between polls."

::= { nlMatrixDSEntry 6 }

nlMatrixTopNControlTable OBJECT-TYPE

SYNTAX SEQUENCE OF NlMatrixTopNControlEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION

"A set of parameters that control the creation of a report of the top N matrix entries according to a selected metric."

::= { nlMatrix 4 }

nlMatrixTopNControlEntry OBJECT-TYPE

SYNTAX NlMatrixTopNControlEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION

"A conceptual row in the nlMatrixTopNControlTable.

An example of the indexing of this table is
 nlMatrixTopNControlDuration.3"

INDEX { nlMatrixTopNControlIndex }
 ::= { nlMatrixTopNControlTable 1 }

NlMatrixTopNControlEntry ::= SEQUENCE {
 nlMatrixTopNControlIndex Integer32,

```

nlMatrixTopNControlMatrixIndex      Integer32,
nlMatrixTopNControlRateBase         INTEGER,
nlMatrixTopNControlTimeRemaining    Integer32,
nlMatrixTopNControlGeneratedReports Counter32,
nlMatrixTopNControlDuration         Integer32,
nlMatrixTopNControlRequestedSize    Integer32,
nlMatrixTopNControlGrantedSize      Integer32,
nlMatrixTopNControlStartTime        TimeStamp,
nlMatrixTopNControlOwner            OwnerString,
nlMatrixTopNControlStatus           RowStatus
}

```

nlMatrixTopNControlIndex OBJECT-TYPE

SYNTAX Integer32 (1..65535)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An index that uniquely identifies an entry in the nlMatrixTopNControlTable. Each such entry defines one top N report prepared for one interface."

::= { nlMatrixTopNControlEntry 1 }

nlMatrixTopNControlMatrixIndex OBJECT-TYPE

SYNTAX Integer32 (1..65535)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The nlMatrix[SD/DS] table for which a top N report will be prepared on behalf of this entry. The nlMatrix[SD/DS] table is identified by the value of the hlMatrixControlIndex for that table - that value is used here to identify the particular table.

This object may not be modified if the associated nlMatrixTopNControlStatus object is equal to active(1)."

::= { nlMatrixTopNControlEntry 2 }

nlMatrixTopNControlRateBase OBJECT-TYPE

SYNTAX INTEGER {
 nlMatrixTopNPkts(1),
 nlMatrixTopNOctets(2)
 }

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The variable for each nlMatrix[SD/DS] entry that the nlMatrixTopNEntries are sorted by.

This object may not be modified if the associated
nlMatrixTopNControlStatus object is equal to active(1)."
 ::= { nlMatrixTopNControlEntry 3 }

nlMatrixTopNControlTimeRemaining OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The number of seconds left in the report currently being collected. When this object is modified by the management station, a new collection is started, possibly aborting a currently running report. The new value is used as the requested duration of this report, and is immediately loaded into the associated nlMatrixTopNControlDuration object. When the report finishes, the probe will automatically start another collection with the same initial value of nlMatrixTopNControlTimeRemaining. Thus the management station may simply read the resulting reports repeatedly, checking the startTime and duration each time to ensure that a report was not missed or that the report parameters were not changed.

While the value of this object is non-zero, it decrements by one per second until it reaches zero. At the time that this object decrements to zero, the report is made accessible in the nlMatrixTopNTable, overwriting any report that may be there.

When this object is modified by the management station, any associated entries in the nlMatrixTopNTable shall be deleted.

(Note that this is a different algorithm than the one used in the hostTopNTable)."

DEFVAL { 1800 }

::= { nlMatrixTopNControlEntry 4 }

nlMatrixTopNControlGeneratedReports OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of reports that have been generated by this entry."
 ::= { nlMatrixTopNControlEntry 5 }

nlMatrixTopNControlDuration OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only
STATUS current
DESCRIPTION

"The number of seconds that this report has collected during the last sampling interval.

When the associated nlMatrixTopNControlTimeRemaining object is set, this object shall be set by the probe to the same value and shall not be modified until the next time the nlMatrixTopNControlTimeRemaining is set. This value shall be zero if no reports have been requested for this nlMatrixTopNControlEntry."

::= { nlMatrixTopNControlEntry 6 }

nlMatrixTopNControlRequestedSize OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-create
STATUS current
DESCRIPTION

"The maximum number of matrix entries requested for this report.

When this object is created or modified, the probe should set nlMatrixTopNControlGrantedSize as closely to this object as is possible for the particular probe implementation and available resources."

DEFVAL { 150 }

::= { nlMatrixTopNControlEntry 7 }

nlMatrixTopNControlGrantedSize OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"The maximum number of matrix entries in this report.

When the associated nlMatrixTopNControlRequestedSize object is created or modified, the probe should set this object as closely to the requested value as is possible for the particular implementation and available resources. The probe must not lower this value except as a result of a set to the associated nlMatrixTopNControlRequestedSize object.

If the value of nlMatrixTopNControlRateBase is equal to nlMatrixTopNPkts, when the next topN report is generated, matrix entries with the highest value of nlMatrixTopNPktRate shall be placed in this table in decreasing order of this rate until there is no more room or until there are no more

matrix entries.

If the value of `nlMatrixTopNControlRateBase` is equal to `nlMatrixTopNOctets`, when the next topN report is generated, matrix entries with the highest value of `nlMatrixTopNOctetRate` shall be placed in this table in decreasing order of this rate until there is no more room or until there are no more matrix entries.

It is an implementation-specific matter how entries with the same value of `nlMatrixTopNPktRate` or `nlMatrixTopNOctetRate` are sorted. It is also an implementation-specific matter as to whether or not zero-valued entries are available."

```
::= { nlMatrixTopNControlEntry 8 }
```

`nlMatrixTopNControlStartTime` OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of `sysUpTime` when this top N report was last started. In other words, this is the time that the associated `nlMatrixTopNControlTimeRemaining` object was modified to start the requested report or the time the report was last automatically (re)started.

This object may be used by the management station to determine if a report was missed or not."

```
::= { nlMatrixTopNControlEntry 9 }
```

`nlMatrixTopNControlOwner` OBJECT-TYPE

SYNTAX OwnerString

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The entity that configured this entry and is therefore using the resources assigned to it."

```
::= { nlMatrixTopNControlEntry 10 }
```

`nlMatrixTopNControlStatus` OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The status of this `nlMatrixTopNControlEntry`.

An entry may not exist in the active state unless all objects in the entry have an appropriate value.

If this object is not equal to active(1), all associated entries in the nlMatrixTopNTable shall be deleted by the agent."

```
::= { nlMatrixTopNControlEntry 11 }
```

nlMatrixTopNTable OBJECT-TYPE

SYNTAX SEQUENCE OF NlMatrixTopNEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A set of statistics for those network layer matrix entries that have counted the highest number of octets or packets."

```
::= { nlMatrix 5 }
```

nlMatrixTopNEntry OBJECT-TYPE

SYNTAX NlMatrixTopNEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the nlMatrixTopNTable.

The nlMatrixTopNControlIndex value in the index identifies the nlMatrixTopNControlEntry on whose behalf this entry was created.

An example of the indexing of this table is
nlMatrixTopNPktRate.3.10"

```
INDEX { nlMatrixTopNControlIndex, nlMatrixTopNIndex }
```

```
::= { nlMatrixTopNTable 1 }
```

```
NlMatrixTopNEntry ::= SEQUENCE {
```

```
  nlMatrixTopNIndex                Integer32,
  nlMatrixTopNProtocolDirLocalIndex Integer32,
  nlMatrixTopNSourceAddress         OCTET STRING,
  nlMatrixTopNDestAddress           OCTET STRING,
  nlMatrixTopNPktRate               Gauge32,
  nlMatrixTopNReversePktRate        Gauge32,
  nlMatrixTopNOctetRate             Gauge32,
  nlMatrixTopNReverseOctetRate      Gauge32
```

```
}
```

nlMatrixTopNIndex OBJECT-TYPE

SYNTAX Integer32 (1..65535)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An index that uniquely identifies an entry in the nlMatrixTopNTable among those in the same report.

This index is between 1 and N, where N is the number of entries in this report.

If the value of `nlMatrixTopNControlRateBase` is equal to `nlMatrixTopNPkts`, increasing values of `nlMatrixTopNIndex` shall be assigned to entries with decreasing values of `nlMatrixTopNPktRate` until index N is assigned or there are no more `nlMatrixTopNEntries`.

If the value of `nlMatrixTopNControlRateBase` is equal to `nlMatrixTopNOctets`, increasing values of `nlMatrixTopNIndex` shall be assigned to entries with decreasing values of `nlMatrixTopNOctetRate` until index N is assigned or there are no more `nlMatrixTopNEntries`."

```
::= { nlMatrixTopNEntry 1 }
```

`nlMatrixTopNProtocolDirLocalIndex` OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The `protocolDirLocalIndex` of the network layer protocol of this entry's network address."

```
::= { nlMatrixTopNEntry 2 }
```

`nlMatrixTopNSourceAddress` OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The network layer address of the source host in this conversation.

This is represented as an octet string with specific semantics and length as identified by the associated `nlMatrixTopNProtocolDirLocalIndex`.

For example, if the `protocolDirLocalIndex` indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

```
::= { nlMatrixTopNEntry 3 }
```

`nlMatrixTopNDestAddress` OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The network layer address of the destination host in this conversation.

This is represented as an octet string with specific semantics and length as identified by the associated `nlMatrixTopNProtocolDirLocalIndex`.

For example, if the `nlMatrixTopNProtocolDirLocalIndex` indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

```
::= { nlMatrixTopNEntry 4 }
```

`nlMatrixTopNPktRate` OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets seen from the source host to the destination host during this sampling interval, counted using the rules for counting the `nlMatrixSDPkts` object. If the value of `nlMatrixTopNControlRateBase` is `nlMatrixTopNPkts`, this variable will be used to sort this report."

```
::= { nlMatrixTopNEntry 5 }
```

`nlMatrixTopNReversePktRate` OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets seen from the destination host to the source host during this sampling interval, counted using the rules for counting the `nlMatrixSDPkts` object (note that the corresponding `nlMatrixSDPkts` object selected is the one whose source address is equal to `nlMatrixTopNDestAddress` and whose destination address is equal to `nlMatrixTopNSourceAddress`.)

Note that if the value of `nlMatrixTopNControlRateBase` is equal to `nlMatrixTopNPkts`, the sort of topN entries is based entirely on `nlMatrixTopNPktRate`, and not on the value of this object."

```
::= { nlMatrixTopNEntry 6 }
```

`nlMatrixTopNOctetRate` OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets seen from the source host to the destination host during this sampling interval, counted using the rules for counting the nlMatrixSDOctets object. If the value of nlMatrixTopNControlRateBase is nlMatrixTopNOctets, this variable will be used to sort this report."

::= { nlMatrixTopNEntry 7 }

nlMatrixTopNReverseOctetRate OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets seen from the destination host to the source host during this sampling interval, counted using the rules for counting the nlMatrixDSOctets object (note that the corresponding nlMatrixSDOctets object selected is the one whose source address is equal to nlMatrixTopNDestAddress and whose destination address is equal to nlMatrixTopNSourceAddress.)"

Note that if the value of nlMatrixTopNControlRateBase is equal to nlMatrixTopNOctets, the sort of topN entries is based entirely on nlMatrixTopNOctetRate, and not on the value of this object."

::= { nlMatrixTopNEntry 8 }

-- Application Layer Functions

--

-- The application layer host, matrix, and matrixTopN functions report
 -- on protocol usage at the network layer or higher. Note that the
 -- use of the term application layer does not imply that only
 -- application-layer protocols are counted, rather it means that
 -- protocols up to and including the application layer are supported.

--

-- Application Layer Host Group

--

-- Counts the amount of traffic, by protocol, sent from and to each
 -- network address discovered by the probe.
 -- Implementation of this group requires implementation of the Network
 -- Layer Host Group.

alHostTable OBJECT-TYPE

SYNTAX SEQUENCE OF AlHostEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A collection of statistics for a particular protocol from a particular network address that has been discovered on an interface of this device.

The probe will populate this table for all protocols in the protocol directory table whose value of protocolDirHostConfig is equal to supportedOn(3), and will delete any entries whose protocolDirEntry is deleted or has a protocolDirHostConfig value of supportedOff(2).

The probe will add to this table all addresses seen as the source or destination address in all packets with no MAC errors, and will increment octet and packet counts in the table for all packets with no MAC errors. Further, entries will only be added to this table if their address exists in the nlHostTable and will be deleted from this table if their address is deleted from the nlHostTable."

::= { alHost 1 }

alHostEntry OBJECT-TYPE

SYNTAX AlHostEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the alHostTable.

The hlHostControlIndex value in the index identifies the hlHostControlEntry on whose behalf this entry was created. The first protocolDirLocalIndex value in the index identifies the network layer protocol of the address. The nlHostAddress value in the index identifies the network layer address of this entry. The second protocolDirLocalIndex value in the index identifies the protocol that is counted by this entry.

An example of the indexing in this entry is
alHostOutPkts.1.783495.18.4.128.2.6.6.34"

INDEX { hlHostControlIndex, alHostTimeMark,
protocolDirLocalIndex, nlHostAddress,
protocolDirLocalIndex }

::= { alHostTable 1 }

AlHostEntry ::= SEQUENCE {

alHostTimeMark	TimeFilter,
alHostInPkts	ZeroBasedCounter32,
alHostOutPkts	ZeroBasedCounter32,

```
    alHostInOctets          ZeroBasedCounter32,
    alHostOutOctets        ZeroBasedCounter32,
    alHostCreateTime       LastCreateTime
}

alHostTimeMark OBJECT-TYPE
    SYNTAX      TimeFilter
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A TimeFilter for this entry.  See the TimeFilter textual
        convention to see how this works."
    ::= { alHostEntry 1 }

alHostInPkts OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of packets of this protocol type without errors
        transmitted to this address since it was added to the
        alHostTable.  Note that this is the number of link-layer
        packets, so if a single network-layer packet is fragmented
        into several link-layer frames, this counter is incremented
        several times."
    ::= { alHostEntry 2 }

alHostOutPkts OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of packets of this protocol type without errors
        transmitted by this address since it was added to the
        alHostTable.  Note that this is the number of link-layer
        packets, so if a single network-layer packet is fragmented
        into several link-layer frames, this counter is incremented
        several times."
    ::= { alHostEntry 3 }

alHostInOctets OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of octets transmitted to this address
        of this protocol type since it was added to the
        alHostTable (excluding framing bits but including
```

FCS octets), excluding those octets in packets that contained errors.

Note this doesn't count just those octets in the particular protocol frames, but includes the entire packet that contained the protocol."

```
::= { alHostEntry 4 }
```

```
alHostOutOctets OBJECT-TYPE
```

```
SYNTAX      ZeroBasedCounter32
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"The number of octets transmitted by this address of this protocol type since it was added to the alHostTable (excluding framing bits but including FCS octets), excluding those octets in packets that contained errors.

Note this doesn't count just those octets in the particular protocol frames, but includes the entire packet that contained the protocol."

```
::= { alHostEntry 5 }
```

```
alHostCreateTime OBJECT-TYPE
```

```
SYNTAX      LastCreateTime
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"The value of sysUpTime when this entry was last activated. This can be used by the management station to ensure that the entry has not been deleted and recreated between polls."

```
::= { alHostEntry 6 }
```

```
--
```

```
-- Application Layer Matrix Group
```

```
--
```

```
-- Counts the amount of traffic, by protocol, sent between each pair  
-- of network addresses discovered by the probe.
```

```
-- Implementation of this group requires implementation of the Network  
-- Layer Matrix Group.
```

```
alMatrixSDTable OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF AlMatrixSDEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

"A list of application traffic matrix entries which collect

statistics for conversations of a particular protocol between two network-level addresses. This table is indexed first by the source address and then by the destination address to make it convenient to collect all statistics from a particular address.

The probe will populate this table for all protocols in the protocol directory table whose value of protocolDirMatrixConfig is equal to supportedOn(3), and will delete any entries whose protocolDirEntry is deleted or has a protocolDirMatrixConfig value of supportedOff(2).

The probe will add to this table all pairs of addresses for all protocols seen in all packets with no MAC errors, and will increment octet and packet counts in the table for all packets with no MAC errors. Further, entries will only be added to this table if their address pair exists in the nlMatrixSDTable and will be deleted from this table if the address pair is deleted from the nlMatrixSDTable."

```
::= { alMatrix 1 }
```

```
alMatrixSDEntry OBJECT-TYPE
    SYNTAX      AlMatrixSDEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
```

```
    "A conceptual row in the alMatrixSDTable.
```

```
The hlMatrixControlIndex value in the index identifies the
hlMatrixControlEntry on whose behalf this entry was created.
The first protocolDirLocalIndex value in the index identifies
the network layer protocol of the nlMatrixSDSourceAddress and
nlMatrixSDDestAddress.
```

```
The nlMatrixSDSourceAddress value in the index identifies the
network layer address of the source host in this conversation.
The nlMatrixSDDestAddress value in the index identifies the
network layer address of the destination host in this
conversation.
```

```
The second protocolDirLocalIndex value in the index identifies
the protocol that is counted by this entry.
```

```
An example of the indexing of this entry is
alMatrixSDPkts.1.783495.18.4.128.2.6.6.4.128.2.6.7.34"
```

```
INDEX { hlMatrixControlIndex, alMatrixSDTimeMark,
        protocolDirLocalIndex,
        nlMatrixSDSourceAddress, nlMatrixSDDestAddress,
        protocolDirLocalIndex }
 ::= { alMatrixSDTable 1 }
```

```
alMatrixSDEntry ::= SEQUENCE {
    alMatrixSDTimeMark      TimeFilter,
    alMatrixSDPkts          ZeroBasedCounter32,
    alMatrixSDOctets        ZeroBasedCounter32,
    alMatrixSDCreateTime    LastCreateTime
}

alMatrixSDTimeMark OBJECT-TYPE
    SYNTAX      TimeFilter
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A TimeFilter for this entry.  See the TimeFilter textual
        convention to see how this works."
    ::= { alMatrixSDEntry 1 }

alMatrixSDPkts OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of packets of this protocol type without errors
        transmitted from the source address to the destination address
        since this entry was added to the alMatrixSDTable.  Note that
        this is the number of link-layer packets, so if a single
        network-layer packet is fragmented into several link-layer
        frames, this counter is incremented several times."
    ::= { alMatrixSDEntry 2 }

alMatrixSDOctets OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of octets in packets of this protocol type
        transmitted from the source address to the destination address
        since this entry was added to the alMatrixSDTable (excluding
        framing bits but including FCS octets), excluding those octets
        in packets that contained errors.

        Note this doesn't count just those octets in the particular
        protocol frames, but includes the entire packet that contained
        the protocol."
    ::= { alMatrixSDEntry 3 }

alMatrixSDCreateTime OBJECT-TYPE
    SYNTAX      LastCreateTime
    MAX-ACCESS  read-only
```

STATUS current

DESCRIPTION

"The value of sysUpTime when this entry was last activated.

This can be used by the management station to ensure that the entry has not been deleted and recreated between polls."

::= { alMatrixSDEntry 4 }

-- Traffic matrix tables from destination to source

alMatrixDSTable OBJECT-TYPE

SYNTAX SEQUENCE OF AlMatrixDSEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A list of application traffic matrix entries which collect statistics for conversations of a particular protocol between two network-level addresses. This table is indexed first by the destination address and then by the source address to make it convenient to collect all statistics to a particular address.

The probe will populate this table for all protocols in the protocol directory table whose value of protocolDirMatrixConfig is equal to supportedOn(3), and will delete any entries whose protocolDirEntry is deleted or has a protocolDirMatrixConfig value of supportedOff(2).

The probe will add to this table all pairs of addresses for all protocols seen in all packets with no MAC errors, and will increment octet and packet counts in the table for all packets with no MAC errors. Further, entries will only be added to this table if their address pair exists in the nlMatrixDSTable and will be deleted from this table if the address pair is deleted from the nlMatrixDSTable."

::= { alMatrix 2 }

alMatrixDSEntry OBJECT-TYPE

SYNTAX AlMatrixDSEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the alMatrixDSTable.

The hlMatrixControlIndex value in the index identifies the hlMatrixControlEntry on whose behalf this entry was created. The first protocolDirLocalIndex value in the index identifies the network layer protocol of the alMatrixDSSourceAddress and alMatrixDSDestAddress.

The nlMatrixDSDestAddress value in the index identifies the network layer address of the destination host in this conversation.

The nlMatrixDSSourceAddress value in the index identifies the network layer address of the source host in this conversation. The second protocolDirLocalIndex value in the index identifies the protocol that is counted by this entry.

An example of the indexing of this entry is

alMatrixDSPkts.1.783495.18.4.128.2.6.7.4.128.2.6.6.34"

```
INDEX { hlMatrixControlIndex, alMatrixDSTimeMark,
        protocolDirLocalIndex,
        nlMatrixDSDestAddress, nlMatrixDSSourceAddress,
        protocolDirLocalIndex }
 ::= { alMatrixDSTable 1 }
```

```
alMatrixDSEntry ::= SEQUENCE {
    alMatrixDSTimeMark          TimeFilter,
    alMatrixDSPkts              ZeroBasedCounter32,
    alMatrixDSOctets            ZeroBasedCounter32,
    alMatrixDSCreateTime        LastCreateTime
}
```

```
alMatrixDSTimeMark OBJECT-TYPE
    SYNTAX          TimeFilter
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "A TimeFilter for this entry. See the TimeFilter textual
        convention to see how this works."
    ::= { alMatrixDSEntry 1 }
```

```
alMatrixDSPkts OBJECT-TYPE
    SYNTAX          ZeroBasedCounter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The number of packets of this protocol type without errors
        transmitted from the source address to the destination address
        since this entry was added to the alMatrixDSTable. Note that
        this is the number of link-layer packets, so if a single
        network-layer packet is fragmented into several link-layer
        frames, this counter is incremented several times."
    ::= { alMatrixDSEntry 2 }
```

```
alMatrixDSOctets OBJECT-TYPE
    SYNTAX          ZeroBasedCounter32
    MAX-ACCESS      read-only
```

STATUS current

DESCRIPTION

"The number of octets in packets of this protocol type transmitted from the source address to the destination address since this entry was added to the alMatrixDSTable (excluding framing bits but including FCS octets), excluding those octets in packets that contained errors.

Note this doesn't count just those octets in the particular protocol frames, but includes the entire packet that contained the protocol."

::= { alMatrixDSEntry 3 }

alMatrixDSCreateTime OBJECT-TYPE

SYNTAX LastCreateTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime when this entry was last activated.

This can be used by the management station to ensure that the entry has not been deleted and recreated between polls."

::= { alMatrixDSEntry 4 }

alMatrixTopNControlTable OBJECT-TYPE

SYNTAX SEQUENCE OF AlMatrixTopNControlEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A set of parameters that control the creation of a report of the top N matrix entries according to a selected metric."

::= { alMatrix 3 }

alMatrixTopNControlEntry OBJECT-TYPE

SYNTAX AlMatrixTopNControlEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row in the alMatrixTopNControlTable.

An example of the indexing of this table is

alMatrixTopNControlDuration.3"

INDEX { alMatrixTopNControlIndex }

::= { alMatrixTopNControlTable 1 }

AlMatrixTopNControlEntry ::= SEQUENCE {

alMatrixTopNControlIndex Integer32,

alMatrixTopNControlMatrixIndex Integer32,

```

    alMatrixTopNControlRateBase          INTEGER,
    alMatrixTopNControlTimeRemaining     Integer32,
    alMatrixTopNControlGeneratedReports  Counter32,
    alMatrixTopNControlDuration          Integer32,
    alMatrixTopNControlRequestedSize     Integer32,
    alMatrixTopNControlGrantedSize       Integer32,
    alMatrixTopNControlStartTime         TimeStamp,
    alMatrixTopNControlOwner             OwnerString,
    alMatrixTopNControlStatus            RowStatus
}

alMatrixTopNControlIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An index that uniquely identifies an entry
         in the alMatrixTopNControlTable.  Each such
         entry defines one top N report prepared for
         one interface."
    ::= { alMatrixTopNControlEntry 1 }

alMatrixTopNControlMatrixIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The alMatrix[SD/DS] table for which a top N report will be
         prepared on behalf of this entry.  The alMatrix[SD/DS] table
         is identified by the value of the hlMatrixControlIndex
         for that table - that value is used here to identify the
         particular table.

         This object may not be modified if the associated
         alMatrixTopNControlStatus object is equal to active(1)."
```

```

    ::= { alMatrixTopNControlEntry 2 }

alMatrixTopNControlRateBase OBJECT-TYPE
    SYNTAX      INTEGER {
        alMatrixTopNTerminalsPkts(1),
        alMatrixTopNTerminalsOctets(2),
        alMatrixTopNAllPkts(3),
        alMatrixTopNAllOctets(4)
    }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The variable for each alMatrix[SD/DS] entry that the
```

alMatrixTopNEntries are sorted by, as well as the selector of the view of the matrix table that will be used.

The values alMatrixTopNTerminalsPkts and alMatrixTopNTerminalsOctets cause collection only from protocols that have no child protocols that are counted. The values alMatrixTopNAllPkts and alMatrixTopNAllOctets cause collection from all alMatrix entries.

This object may not be modified if the associated alMatrixTopNControlStatus object is equal to active(1)."
 ::= { alMatrixTopNControlEntry 3 }

alMatrixTopNControlTimeRemaining OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The number of seconds left in the report currently being collected. When this object is modified by the management station, a new collection is started, possibly aborting a currently running report. The new value is used as the requested duration of this report, and is immediately loaded into the associated alMatrixTopNControlDuration object. When the report finishes, the probe will automatically start another collection with the same initial value of alMatrixTopNControlTimeRemaining. Thus the management station may simply read the resulting reports repeatedly, checking the startTime and duration each time to ensure that a report was not missed or that the report parameters were not changed.

While the value of this object is non-zero, it decrements by one per second until it reaches zero. At the time that this object decrements to zero, the report is made accessible in the alMatrixTopNTable, overwriting any report that may be there.

When this object is modified by the management station, any associated entries in the alMatrixTopNTable shall be deleted.

(Note that this is a different algorithm than the one used in the hostTopNTable)."

DEFVAL { 1800 }

::= { alMatrixTopNControlEntry 4 }

alMatrixTopNControlGeneratedReports OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of reports that have been generated by this entry."

::= { alMatrixTopNControlEntry 5 }

alMatrixTopNControlDuration OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of seconds that this report has collected during the last sampling interval."

When the associated alMatrixTopNControlTimeRemaining object is set, this object shall be set by the probe to the same value and shall not be modified until the next time the alMatrixTopNControlTimeRemaining is set.

This value shall be zero if no reports have been requested for this alMatrixTopNControlEntry."

::= { alMatrixTopNControlEntry 6 }

alMatrixTopNControlRequestedSize OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The maximum number of matrix entries requested for this report."

When this object is created or modified, the probe should set alMatrixTopNControlGrantedSize as closely to this object as is possible for the particular probe implementation and available resources."

DEFVAL { 150 }

::= { alMatrixTopNControlEntry 7 }

alMatrixTopNControlGrantedSize OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum number of matrix entries in this report."

When the associated alMatrixTopNControlRequestedSize object is created or modified, the probe should set this

object as closely to the requested value as is possible for the particular implementation and available resources. The probe must not lower this value except as a result of a set to the associated `alMatrixTopNControlRequestedSize` object.

If the value of `alMatrixTopNControlRateBase` is equal to `alMatrixTopNTerminalsPkts` or `alMatrixTopNAllPkts`, when the next topN report is generated, matrix entries with the highest value of `alMatrixTopNPktRate` shall be placed in this table in decreasing order of this rate until there is no more room or until there are no more matrix entries.

If the value of `alMatrixTopNControlRateBase` is equal to `alMatrixTopNTerminalsOctets` or `alMatrixTopNAllOctets`, when the next topN report is generated, matrix entries with the highest value of `alMatrixTopNOctetRate` shall be placed in this table in decreasing order of this rate until there is no more room or until there are no more matrix entries.

It is an implementation-specific matter how entries with the same value of `alMatrixTopNPktRate` or `alMatrixTopNOctetRate` are sorted. It is also an implementation-specific matter as to whether or not zero-valued entries are available."

```
::= { alMatrixTopNControlEntry 8 }
```

`alMatrixTopNControlStartTime` OBJECT-TYPE

```
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
```

"The value of `sysUpTime` when this top N report was last started. In other words, this is the time that the associated `alMatrixTopNControlTimeRemaining` object was modified to start the requested report or the time the report was last automatically (re)started.

This object may be used by the management station to determine if a report was missed or not."

```
::= { alMatrixTopNControlEntry 9 }
```

`alMatrixTopNControlOwner` OBJECT-TYPE

```
SYNTAX      OwnerString
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
```

"The entity that configured this entry and is therefore using the resources assigned to it."

```
::= { alMatrixTopNControlEntry 10 }
```

alMatrixTopNControlStatus OBJECT-TYPE

```
SYNTAX      RowStatus
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
```

"The status of this alMatrixTopNControlEntry.

An entry may not exist in the active state unless all objects in the entry have an appropriate value.

If this object is not equal to active(1), all associated entries in the alMatrixTopNTable shall be deleted by the agent."

```
::= { alMatrixTopNControlEntry 11 }
```

alMatrixTopNTable OBJECT-TYPE

```
SYNTAX      SEQUENCE OF AlMatrixTopNEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
```

"A set of statistics for those application layer matrix entries that have counted the highest number of octets or packets."

```
::= { alMatrix 4 }
```

alMatrixTopNEntry OBJECT-TYPE

```
SYNTAX      AlMatrixTopNEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
```

"A conceptual row in the alMatrixTopNTable.

The alMatrixTopNControlIndex value in the index identifies the alMatrixTopNControlEntry on whose behalf this entry was created.

An example of the indexing of this table is
alMatrixTopNPktRate.3.10"

```
INDEX { alMatrixTopNControlIndex, alMatrixTopNIndex }
```

```
::= { alMatrixTopNTable 1 }
```

AlMatrixTopNEntry ::= SEQUENCE {

```
  alMatrixTopNIndex                Integer32,
  alMatrixTopNProtocolDirLocalIndex Integer32,
  alMatrixTopNSourceAddress         OCTET STRING,
  alMatrixTopNDestAddress           OCTET STRING,
```

```

alMatrixTopNAppProtocolDirLocalIndex   Integer32,
alMatrixTopNPktRate                     Gauge32,
alMatrixTopNReversePktRate             Gauge32,
alMatrixTopNOctetRate                   Gauge32,
alMatrixTopNReverseOctetRate           Gauge32
}

```

alMatrixTopNIndex OBJECT-TYPE

SYNTAX Integer32 (1..65535)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An index that uniquely identifies an entry in the alMatrixTopNTable among those in the same report. This index is between 1 and N, where N is the number of entries in this report.

If the value of alMatrixTopNControlRateBase is equal to alMatrixTopNTerminalsPkts or alMatrixTopNAllPkts, increasing values of alMatrixTopNIndex shall be assigned to entries with decreasing values of alMatrixTopNPktRate until index N is assigned or there are no more alMatrixTopNEntries.

If the value of alMatrixTopNControlRateBase is equal to alMatrixTopNTerminalsOctets or alMatrixTopNAllOctets, increasing values of alMatrixTopNIndex shall be assigned to entries with decreasing values of alMatrixTopNOctetRate until index N is assigned or there are no more alMatrixTopNEntries."

::= { alMatrixTopNEntry 1 }

alMatrixTopNProtocolDirLocalIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The protocolDirLocalIndex of the network layer protocol of this entry's network address."

::= { alMatrixTopNEntry 2 }

alMatrixTopNSourceAddress OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The network layer address of the source host in this conversation.

This is represented as an octet string with specific semantics and length as identified

by the associated alMatrixTopNProtocolDirLocalIndex.

For example, if the alMatrixTopNProtocolDirLocalIndex indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

```
::= { alMatrixTopNEntry 3 }
```

alMatrixTopNDestAddress OBJECT-TYPE

SYNTAX OCTET STRING

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The network layer address of the destination host in this conversation.

This is represented as an octet string with specific semantics and length as identified by the associated alMatrixTopNProtocolDirLocalIndex.

For example, if the alMatrixTopNProtocolDirLocalIndex indicates an encapsulation of ip, this object is encoded as a length octet of 4, followed by the 4 octets of the ip address, in network byte order."

```
::= { alMatrixTopNEntry 4 }
```

alMatrixTopNAppProtocolDirLocalIndex OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The type of the protocol counted by this matrix entry."

```
::= { alMatrixTopNEntry 5 }
```

alMatrixTopNPktRate OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets seen of this protocol from the source host to the destination host during this sampling interval, counted using the rules for counting the alMatrixSDPkts object.

If the value of alMatrixTopNControlRateBase is alMatrixTopNTerminalsPkts or alMatrixTopNAllPkts, this variable will be used to sort this report."

```
::= { alMatrixTopNEntry 6 }
```

`alMatrixTopNReversePktRate OBJECT-TYPE``SYNTAX Gauge32``MAX-ACCESS read-only``STATUS current``DESCRIPTION`

"The number of packets seen of this protocol from the destination host to the source host during this sampling interval, counted using the rules for counting the `alMatrixDSPkts` object (note that the corresponding `alMatrixSDPkts` object selected is the one whose source address is equal to `alMatrixTopNDestAddress` and whose destination address is equal to `alMatrixTopNSourceAddress`.)

Note that if the value of `alMatrixTopNControlRateBase` is equal to `alMatrixTopNTerminalsPkts` or `alMatrixTopNAllPkts`, the sort of topN entries is based entirely on `alMatrixTopNPktRate`, and not on the value of this object."

`::= { alMatrixTopNEntry 7 }``alMatrixTopNOctetRate OBJECT-TYPE``SYNTAX Gauge32``MAX-ACCESS read-only``STATUS current``DESCRIPTION`

"The number of octets seen of this protocol from the source host to the destination host during this sampling interval, counted using the rules for counting the `alMatrixSDOctets` object.

If the value of `alMatrixTopNControlRateBase` is `alMatrixTopNTerminalsOctets` or `alMatrixTopNAllOctets`, this variable will be used to sort this report."

`::= { alMatrixTopNEntry 8 }``alMatrixTopNReverseOctetRate OBJECT-TYPE``SYNTAX Gauge32``MAX-ACCESS read-only``STATUS current``DESCRIPTION`

"The number of octets seen of this protocol from the destination host to the source host during this sampling interval, counted using the rules for counting the `alMatrixDSOctets` object (note that the corresponding `alMatrixSDOctets` object selected is the one whose source address is equal to `alMatrixTopNDestAddress` and whose destination address is equal to `alMatrixTopNSourceAddress`.)

Note that if the value of `alMatrixTopNControlRateBase` is equal

```
    to alMatrixTopNTerminalsOctets or alMatrixTopNAllOctets, the
    sort of topN entries is based entirely on
    alMatrixTopNOctetRate, and not on the value of this object."
 ::= { alMatrixTopNEntry 9 }
```

```
--
-- User History Collection Group (usrHistory)
--
-- The usrHistory group combines mechanisms seen in the alarm and
-- history groups to provide user-specified history collection,
-- utilizing two additional control tables and one additional data
-- table. This function has traditionally been done by NMS
-- applications, via periodic polling. The usrHistory group allows
-- this task to be offloaded to an RMON probe.
--
-- Data (an ASN.1 INTEGER based object) is collected in the same
-- manner as any history data table (e.g. etherHistoryTable) except
-- that the user specifies the MIB instances to be collected. Objects
-- are collected in bucket-groups, with the intent that all MIB
-- instances in the same bucket-group are collected as atomically as
-- possible by the RMON probe.
--
-- The usrHistoryControlTable is a one-dimensional read-create table.
-- Each row configures a collection of user history buckets, much
-- the same as a historyControlEntry, except that the creation of a
-- row in this table will cause one or more associated instances in
-- the usrHistoryObjectTable to be created. The user specifies the
-- number of bucket elements (rows in the usrHistoryObjectTable)
-- requested, as well as the number of buckets requested.
--
-- The usrHistoryObjectTable is a 2-d read-write table.
-- Each row configures a single MIB instance to be collected.
-- All rows with the same major index constitute a bucket-group.
--
-- The usrHistoryTable is a 3-d read-only table containing
-- the data of associated usrHistoryControlEntries. Each
-- entry represents the value of a single MIB instance
-- during a specific sampling interval (or the rate of
-- change during the interval).
--
-- A sample value is stored in two objects - an absolute value and
-- a status object. This allows numbers from -(2G-1) to +4G to be
-- stored. The status object also indicates whether a sample is
-- valid. This allows data collection to continue if periodic
-- retrieval of a particular instance fails for any reason.
--
-- Row Creation Order Relationships
--
```

```
-- The static nature of the usrHistoryObjectTable creates
-- some row creation/modification issues. The rows in this
-- table need to be set before the associated
-- usrHistoryControlEntry can be activated.
--
-- Note that the usrHistoryObject entries associated with a
-- particular usrHistoryControlEntry are not required to
-- be active before the control entry is activated. However,
-- the usrHistory data entries associated with an inactive
-- usrHistoryObject entry will be inactive (i.e.
-- usrHistoryValStatus == valueNotAvailable).
--
```

```
usrHistoryControlTable OBJECT-TYPE
    SYNTAX SEQUENCE OF UsrHistoryControlEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "A list of data-collection configuration entries."
    ::= { usrHistory 1 }
```

```
usrHistoryControlEntry OBJECT-TYPE
    SYNTAX UsrHistoryControlEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "A list of parameters that set up a group of user-defined
        MIB objects to be sampled periodically (called a
        bucket-group).

        For example, an instance of usrHistoryControlInterval
        might be named usrHistoryControlInterval.1"
    INDEX { usrHistoryControlIndex }
    ::= { usrHistoryControlTable 1 }
```

```
UsrHistoryControlEntry ::= SEQUENCE {
    usrHistoryControlIndex      Integer32,
    usrHistoryControlObjects    Integer32,
    usrHistoryControlBucketsRequested Integer32,
    usrHistoryControlBucketsGranted Integer32,
    usrHistoryControlInterval   Integer32,
    usrHistoryControlOwner      OwnerString,
    usrHistoryControlStatus     RowStatus
}
```

```
usrHistoryControlIndex OBJECT-TYPE
    SYNTAX Integer32 (1..65535)
    MAX-ACCESS not-accessible
```

STATUS current
DESCRIPTION
"An index that uniquely identifies an entry in the
usrHistoryControlTable. Each such entry defines a
set of samples at a particular interval for a specified
set of MIB instances available from the managed system."
 ::= { usrHistoryControlEntry 1 }

usrHistoryControlObjects OBJECT-TYPE
SYNTAX Integer32 (1..65535)
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The number of MIB objects to be collected
in the portion of usrHistoryTable associated with this
usrHistoryControlEntry.

This object may not be modified if the associated instance
of usrHistoryControlStatus is equal to active(1)."
 ::= { usrHistoryControlEntry 2 }

usrHistoryControlBucketsRequested OBJECT-TYPE
SYNTAX Integer32 (1..65535)
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The requested number of discrete time intervals
over which data is to be saved in the part of the
usrHistoryTable associated with this usrHistoryControlEntry.

When this object is created or modified, the probe
should set usrHistoryControlBucketsGranted as closely to
this object as is possible for the particular probe
implementation and available resources."
DEFVAL { 50 }
 ::= { usrHistoryControlEntry 3 }

usrHistoryControlBucketsGranted OBJECT-TYPE
SYNTAX Integer32 (1..65535)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of discrete sampling intervals
over which data shall be saved in the part of
the usrHistoryTable associated with this
usrHistoryControlEntry.

When the associated usrHistoryControlBucketsRequested

object is created or modified, the probe should set this object as closely to the requested value as is possible for the particular probe implementation and available resources. The probe must not lower this value except as a result of a modification to the associated `usrHistoryControlBucketsRequested` object.

The associated `usrHistoryControlBucketsRequested` object should be set before or at the same time as this object to allow the probe to accurately estimate the resources required for this `usrHistoryControlEntry`.

There will be times when the actual number of buckets associated with this entry is less than the value of this object. In this case, at the end of each sampling interval, a new bucket will be added to the `usrHistoryTable`.

When the number of buckets reaches the value of this object and a new bucket is to be added to the `usrHistoryTable`, the oldest bucket associated with this `usrHistoryControlEntry` shall be deleted by the agent so that the new bucket can be added.

When the value of this object changes to a value less than the current value, entries are deleted from the `usrHistoryTable` associated with this `usrHistoryControlEntry`. Enough of the oldest of these entries shall be deleted by the agent so that their number remains less than or equal to the new value of this object.

When the value of this object changes to a value greater than the current value, the number of associated `usrHistory` entries may be allowed to grow."

```
::= { usrHistoryControlEntry 4 }
```

`usrHistoryControlInterval` OBJECT-TYPE

SYNTAX Integer32 (1..2147483647)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The interval in seconds over which the data is sampled for each bucket in the part of the `usrHistory` table associated with this `usrHistoryControlEntry`.

Because the counters in a bucket may overflow at their maximum value with no indication, a prudent manager will take into account the possibility of overflow in any of

the associated counters. It is important to consider the minimum time in which any counter could overflow on a particular media type and set the `usrHistoryControlInterval` object to a value less than this interval.

This object may not be modified if the associated `usrHistoryControlStatus` object is equal to `active(1)`."

```
DEFVAL { 1800 }
 ::= { usrHistoryControlEntry 5 }
```

`usrHistoryControlOwner` OBJECT-TYPE

SYNTAX OwnerString

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The entity that configured this entry and is therefore using the resources assigned to it."

```
 ::= { usrHistoryControlEntry 6 }
```

`usrHistoryControlStatus` OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The status of this variable history control entry.

An entry may not exist in the active state unless all objects in the entry have an appropriate value.

If this object is not equal to `active(1)`, all associated entries in the `usrHistoryTable` shall be deleted."

```
 ::= { usrHistoryControlEntry 7 }
```

-- Object table

`usrHistoryObjectTable` OBJECT-TYPE

SYNTAX SEQUENCE OF `UsrHistoryObjectEntry`

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A list of data-collection configuration entries."

```
 ::= { usrHistory 2 }
```

`usrHistoryObjectEntry` OBJECT-TYPE

SYNTAX `UsrHistoryObjectEntry`

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A list of MIB instances to be sampled periodically.

Entries in this table are created when an associated `usrHistoryControlObjects` object is created.

The `usrHistoryControlIndex` value in the index is that of the associated `usrHistoryControlEntry`.

For example, an instance of `usrHistoryObjectVariable` might be `usrHistoryObjectVariable.1.3`"

```
INDEX { usrHistoryControlIndex, usrHistoryObjectIndex }
 ::= { usrHistoryObjectTable 1 }
```

```
UsrHistoryObjectEntry ::= SEQUENCE {
  usrHistoryObjectIndex      Integer32,
  usrHistoryObjectVariable   OBJECT IDENTIFIER,
  usrHistoryObjectSampleType INTEGER
}
```

```
usrHistoryObjectIndex OBJECT-TYPE
  SYNTAX Integer32 (1..65535)
  MAX-ACCESS not-accessible
  STATUS current
  DESCRIPTION
    "An index used to uniquely identify an entry in the
     usrHistoryObject table. Each such entry defines a
     MIB instance to be collected periodically."
  ::= { usrHistoryObjectEntry 1 }
```

```
usrHistoryObjectVariable OBJECT-TYPE
  SYNTAX OBJECT IDENTIFIER
  MAX-ACCESS read-create
  STATUS current
  DESCRIPTION
    "The object identifier of the particular variable to be
     sampled.
```

Only variables that resolve to an ASN.1 primitive type of `Integer32` (`Integer32`, `Counter`, `Gauge`, or `TimeTicks`) may be sampled.

Because SNMP access control is articulated entirely in terms of the contents of MIB views, no access control mechanism exists that can restrict the value of this object to identify only those objects that exist in a particular MIB view. Because there is thus no acceptable means of restricting the read access that could be obtained through the user history

mechanism, the probe must only grant write access to this object in those views that have read access to all objects on the probe.

During a set operation, if the supplied variable name is not available in the selected MIB view, a badValue error must be returned.

This object may not be modified if the associated usrHistoryControlStatus object is equal to active(1)."

```
::= { usrHistoryObjectEntry 2 }
```

```
usrHistoryObjectSampleType OBJECT-TYPE
```

```
SYNTAX INTEGER {
    absoluteValue(1),
    deltaValue(2)
}
```

```
MAX-ACCESS read-create
```

```
STATUS current
```

```
DESCRIPTION
```

"The method of sampling the selected variable for storage in the usrHistoryTable.

If the value of this object is absoluteValue(1), the value of the selected variable will be copied directly into the history bucket.

If the value of this object is deltaValue(2), the value of the selected variable at the last sample will be subtracted from the current value, and the difference will be stored in the history bucket. If the associated usrHistoryObjectVariable instance could not be obtained at the previous sample interval, then a delta sample is not possible, and the value of the associated usrHistoryValStatus object for this interval will be valueNotAvailable(1).

This object may not be modified if the associated usrHistoryControlStatus object is equal to active(1)."

```
::= { usrHistoryObjectEntry 3 }
```

```
-- data table
```

```
usrHistoryTable OBJECT-TYPE
```

```
SYNTAX SEQUENCE OF UsrHistoryEntry
```

```
MAX-ACCESS not-accessible
```

```
STATUS current
```

```
DESCRIPTION
```

"A list of user defined history entries."

```
::= { usrHistory 3 }
```

usrHistoryEntry OBJECT-TYPE

```
SYNTAX UsrHistoryEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
```

"A historical sample of user-defined variables. This sample is associated with the usrHistoryControlEntry which set up the parameters for a regular collection of these samples.

The usrHistoryControlIndex value in the index identifies the usrHistoryControlEntry on whose behalf this entry was created.

The usrHistoryObjectIndex value in the index identifies the usrHistoryObjectEntry on whose behalf this entry was created.

For example, an instance of usrHistoryAbsValue, which represents the 14th sample of a variable collected as specified by usrHistoryControlEntry.1 and usrHistoryObjectEntry.1.5, would be named usrHistoryAbsValue.1.14.5"

```
INDEX { usrHistoryControlIndex, usrHistorySampleIndex,
        usrHistoryObjectIndex }
 ::= { usrHistoryTable 1 }
```

```
UsrHistoryEntry ::= SEQUENCE {
    usrHistorySampleIndex Integer32,
    usrHistoryIntervalStart TimeStamp,
    usrHistoryIntervalEnd TimeStamp,
    usrHistoryAbsValue Gauge32,
    usrHistoryValStatus INTEGER
}
```

usrHistorySampleIndex OBJECT-TYPE

```
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
```

"An index that uniquely identifies the particular sample this entry represents among all samples associated with the same usrHistoryControlEntry. This index starts at 1 and increases by one as each new sample is taken."

```
::= { usrHistoryEntry 1 }
```

usrHistoryIntervalStart OBJECT-TYPE

```
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
```

DESCRIPTION

"The value of sysUpTime at the start of the interval over which this sample was measured. If the probe keeps track of the time of day, it should start the first sample of the history at a time such that when the next hour of the day begins, a sample is started at that instant.

Note that following this rule may require the probe to delay collecting the first sample of the history, as each sample must be of the same interval. Also note that the sample which is currently being collected is not accessible in this table until the end of its interval."

```
::= { usrHistoryEntry 2 }
```

usrHistoryIntervalEnd OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at the end of the interval over which this sample was measured."

```
::= { usrHistoryEntry 3 }
```

usrHistoryAbsValue OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The absolute value (i.e. unsigned value) of the user-specified statistic during the last sampling period. The value during the current sampling period is not made available until the period is completed.

To obtain the true value for this sampling interval, the associated instance of usrHistoryValStatus must be checked, and usrHistoryAbsValue adjusted as necessary.

If the MIB instance could not be accessed during the sampling interval, then this object will have a value of zero and the associated instance of usrHistoryValStatus will be set to 'valueNotAvailable(1)'."

```
::= { usrHistoryEntry 4 }
```

usrHistoryValStatus OBJECT-TYPE

SYNTAX INTEGER {

valueNotAvailable(1),

valuePositive(2),

```

    valueNegative(3)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "This object indicates the validity and sign of the data in
    the associated instance of usrHistoryAbsValue.

    If the MIB instance could not be accessed during the sampling
    interval, then 'valueNotAvailable(1)' will be returned.

    If the sample is valid and actual value of the sample is
    greater than or equal to zero then 'valuePositive(2)' is
    returned.

    If the sample is valid and the actual value of the sample is
    less than zero, 'valueNegative(3)' will be returned. The
    associated instance of usrHistoryAbsValue should be multiplied
    by -1 to obtain the true sample value."
 ::= { usrHistoryEntry 5 }

```

```

-- The Probe Configuration Group
--
-- This group controls the configuration of various operating
-- parameters of the probe.

```

```
ControlString ::= TEXTUAL-CONVENTION
```

```

STATUS current
DESCRIPTION
    "This data type is used to communicate with a modem or a
    serial data switch.  A ControlString contains embedded
    commands to control how the device will interact with the
    remote device through the serial interface.  Commands are
    represented as two character sequences beginning with
    the '^' character.

    The following commands are recognized by the device (note
    that command characters are case sensitive):

    ^s Send string that follows which is terminated by the
    next command or the end of string.
    ^c Delay for the number of seconds that follows.  Toss
    out any data received rather than storing it in a
    buffer for parsing.
    ^t Set timeout to the value represented by the decimal
    digits that follow.  The default timeout is 20
    seconds. Note that this timeout may be overridden
    by a smaller serialTimeout configured for the

```

- associated serial interface (see serialConfigTable).
- ^w Wait for the reply string that follows which is terminated by the next command or the end of string. Partial and case insensitive matching is applied, ie. if the reply string (any case combination) is found anywhere in the received string, then the a match is found. If the current timeout elapses without a match, then the remaining control string is ignored.
 - ^! The ^ character.
 - ^d Delay the number of seconds specified by the decimal digits that follow.
 - ^b Send break for the number of milliseconds specified by the decimal digits that follow. If no digits follow, break will be enforced for 250 milliseconds by default.

The following ASCII control characters may be inserted into the '^s' send string or the '^w' reply string:

```

^@      0x00
^A      0x01
. .
^M      0x0D
. .
^Z      0x1A
^[      0x1B
^       0x1C
^]      0x1D
^^      0x1E
^_      0x1F

```

Binary data may also be inserted into the data stream. The control sequence for each byte of binary data is ^0x##, where ## is the hexadecimal representation of the data byte. Two ASCII characters (0-9, a-f, A-F) must follow the '^0x' control prefix. For example, '^0x0D^0x0A' is interpreted as a carriage return followed by a line feed."

SYNTAX DisplayString

probeCapabilities OBJECT-TYPE

```

SYNTAX BITS {
    etherStats(0),
    historyControl(1),
    etherHistory(2),
    alarm(3),
    hosts(4),
    hostTopN(5),
    matrix(6),
    filter(7),

```

```

capture(8),
event(9),
tokenRingMLStats(10),
tokenRingPStats(11),
tokenRingMLHistory(12),
tokenRingPHistory(13),
ringStation(14),
ringStationOrder(15),
ringStationConfig(16),
sourceRouting(17),
protocolDirectory(18),
protocolDistribution(19),
addressMapping(20),
nlHost(21),
nlMatrix(22),
alHost(23),
alMatrix(24),
usrHistory(25),
probeConfig(26)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "An indication of the RMON MIB groups supported
    on at least one interface by this probe."
 ::= { probeConfig 1 }

```

```

probeSoftwareRev OBJECT-TYPE
SYNTAX      DisplayString (SIZE(0..15))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The software revision of this device.  This string will have
    a zero length if the revision is unknown."
 ::= { probeConfig 2 }

```

```

probeHardwareRev OBJECT-TYPE
SYNTAX      DisplayString (SIZE(0..31))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The hardware revision of this device.  This string will have
    a zero length if the revision is unknown."
 ::= { probeConfig 3 }

```

```

probeDateTime OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE (0 | 8 | 11))
MAX-ACCESS  read-write

```

STATUS current

DESCRIPTION

"Probe's current date and time.

field	octets	contents	range
----	-----	-----	-----
1	1-2	year	0..65536
2	3	month	1..12
3	4	day	1..31
4	5	hour	0..23
5	6	minutes	0..59
6	7	seconds	0..60
		(use 60 for leap-second)	
7	8	deci-seconds	0..9
8	9	direction from UTC	'+' / '-'
9	10	hours from UTC	0..11
10	11	minutes from UTC	0..59

For example, Tuesday May 26, 1992 at 1:30:15 PM EDT would be displayed as:

1992-5-26,13:30:15.0,-4:0

Note that if only local time is known, then timezone information (fields 8-10) is not present, and if no time information is known, the null string is returned."

::= { probeConfig 4 }

probeResetControl OBJECT-TYPE

SYNTAX INTEGER {
 running(1),
 warmBoot(2),
 coldBoot(3)
 }

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Setting this object to warmBoot(2) causes the device to restart the application software with current configuration parameters saved in non-volatile memory. Setting this object to coldBoot(3) causes the device to reinitialize configuration parameters in non-volatile memory to default values and restart the application software. When the device is running normally, this variable has a value of running(1)."

::= { probeConfig 5 }

```
-- The following download objects do not restrict an implementation
-- from implementing additional download mechanisms (controlled in an
-- implementation-specific manner). Further, in the case where the RMON
-- agent shares a processor with other types of systems, the
-- implementation is not required to download those non-RMON functions
-- with this mechanism.
```

```
probeDownloadFile OBJECT-TYPE
```

```
SYNTAX      DisplayString (SIZE(0..127))
```

```
MAX-ACCESS  read-write
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The file name to be downloaded from the TFTP server when a
download is next requested via this MIB. This value is set to
the zero length string when no file name has been specified."
```

```
::= { probeConfig 6 }
```

```
probeDownloadTFTPServer OBJECT-TYPE
```

```
SYNTAX      IpAddress
```

```
MAX-ACCESS  read-write
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The IP address of the TFTP server that contains the boot
image to load when a download is next requested via this MIB.
This value is set to '0.0.0.0' when no IP address has been
specified."
```

```
::= { probeConfig 7 }
```

```
probeDownloadAction OBJECT-TYPE
```

```
SYNTAX      INTEGER {
                notDownloading(1),
                downloadToPROM(2),
                downloadToRAM(3)
            }
```

```
MAX-ACCESS  read-write
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"When this object is set to downloadToRAM(2) or
downloadToPROM(3), the device will discontinue its
normal operation and begin download of the image specified
by probeDownloadFile from the server specified by
probeDownloadTFTPServer using the TFTP protocol. If
downloadToRAM(2) is specified, the new image is copied
to RAM only (the old image remains unaltered in the flash
EPROM). If downloadToPROM(3) is specified
the new image is written to the flash EPROM
memory after its checksum has been verified to be correct.
When the download process is completed, the device will
```

warm boot to restart the newly loaded application.
When the device is not downloading, this object will have a value of notDownloading(1)."

::= { probeConfig 8 }

probeDownloadStatus OBJECT-TYPE

SYNTAX INTEGER {
downloadSuccess(1),
downloadStatusUnknown(2),
downloadGeneralError(3),
downloadNoResponseFromServer(4),
downloadChecksumError(5),
downloadIncompatibleImage(6),
downloadTftpFileNotFound(7),
downloadTftpAccessViolation(8)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The status of the last download procedure, if any. This object will have a value of downloadStatusUnknown(2) if no download process has been performed."

::= { probeConfig 9 }

serialConfigTable OBJECT-TYPE

SYNTAX SEQUENCE OF SerialConfigEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table of serial interface configuration entries. This data will be stored in non-volatile memory and preserved across probe resets or power loss."

::= { probeConfig 10 }

serialConfigEntry OBJECT-TYPE

SYNTAX SerialConfigEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A set of configuration parameters for a particular serial interface on this device. If the device has no serial interfaces, this table is empty.

The index is composed of the ifIndex assigned to this serial line interface."

INDEX { ifIndex }

::= { serialConfigTable 1 }

```

SerialConfigEntry ::= SEQUENCE {
    serialMode          INTEGER,
    serialProtocol      INTEGER,
    serialTimeout       Integer32 (1..65535),
    serialModemInitString ControlString (SIZE (0..255)),
    serialModemHangUpString ControlString (SIZE (0..255)),
    serialModemConnectResp DisplayString (SIZE (0..255)),
    serialModemNoConnectResp DisplayString (SIZE (0..255)),
    serialDialoutTimeout Integer32 (1..65535),
    serialStatus        RowStatus
}

```

```

serialMode OBJECT-TYPE
    SYNTAX      INTEGER {
                direct(1),
                modem(2)
            }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The type of incoming connection to expect on this serial
        interface."
    DEFVAL { direct }
    ::= { serialConfigEntry 1 }

```

```

serialProtocol OBJECT-TYPE
    SYNTAX      INTEGER {
                other(1),
                slip(2),
                ppp(3)
            }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The type of data link encapsulation to be used on this
        serial interface."
    DEFVAL { slip }
    ::= { serialConfigEntry 2 }

```

```

serialTimeout OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This timeout value is used when the Management Station has
        initiated the conversation over the serial link. This variable
        represents the number of seconds of inactivity allowed before
        terminating the connection on this serial interface. Use the

```

serialDialoutTimeout in the case where the probe has initiated the connection for the purpose of sending a trap."

```
DEFVAL { 300 }
 ::= { serialConfigEntry 3 }
```

serialModemInitString OBJECT-TYPE

SYNTAX ControlString (SIZE (0..255))

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A control string which controls how a modem attached to this serial interface should be initialized. The initialization is performed once during startup and again after each connection is terminated if the associated serialMode has the value of modem(2).

A control string that is appropriate for a wide variety of modems is: '^s^MATE0Q0V1X4 S0=1 S2=43^M'."

```
 ::= { serialConfigEntry 4 }
```

serialModemHangUpString OBJECT-TYPE

SYNTAX ControlString (SIZE (0..255))

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A control string which specifies how to disconnect a modem connection on this serial interface. This object is only meaningful if the associated serialMode has the value of modem(2).

A control string that is appropriate for a wide variety of modems is: '^d2^s+++^d2^sATH0^M^d2'."

```
 ::= { serialConfigEntry 5 }
```

serialModemConnectResp OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"An ASCII string containing substrings that describe the expected modem connection response code and associated bps rate. The substrings are delimited by the first character in the string, for example:

```
/CONNECT/300/CONNECT 1200/1200/CONNECT 2400/2400/
CONNECT 4800/4800/CONNECT 9600/9600
```

will be interpreted as:

response code	bps rate
CONNECT	300
CONNECT 1200	1200

```
CONNECT 2400      2400
CONNECT 4800      4800
CONNECT 9600      9600
```

The agent will use the information in this string to adjust the bps rate of this serial interface once a modem connection is established.

A value that is appropriate for a wide variety of modems is:
 '/CONNECT/300/CONNECT 1200/1200/CONNECT 2400/2400/
 CONNECT 4800/4800/CONNECT 9600/9600/CONNECT 14400/14400/
 CONNECT 19200/19200/CONNECT 38400/38400/'.

```
::= { serialConfigEntry 6 }
```

```
serialModemNoConnectResp  OBJECT-TYPE
SYNTAX      DisplayString (SIZE (0..255))
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
```

"An ASCII string containing response codes that may be generated by a modem to report the reason why a connection attempt has failed. The response codes are delimited by the first character in the string, for example:

```
/NO CARRIER/BUSY/NO DIALTONE/NO ANSWER/ERROR/
```

If one of these response codes is received via this serial interface while attempting to make a modem connection, the agent will issue the hang up command as specified by serialModemHangUpString.

A value that is appropriate for a wide variety of modems is:
 '/NO CARRIER/BUSY/NO DIALTONE/NO ANSWER/ERROR/'.

```
::= { serialConfigEntry 7 }
```

```
serialDialoutTimeout  OBJECT-TYPE
SYNTAX      Integer32 (1..65535)
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
```

"This timeout value is used when the probe initiates the serial connection with the intention of contacting a management station. This variable represents the number of seconds of inactivity allowed before terminating the connection on this serial interface."

```
DEFVAL { 20 }
```

```
::= { serialConfigEntry 8 }
```

```
serialStatus  OBJECT-TYPE
SYNTAX      RowStatus
MAX-ACCESS  read-create
```

```
STATUS      current
DESCRIPTION
    "The status of this serialConfigEntry.
```

```
    An entry may not exist in the active state unless all
    objects in the entry have an appropriate value."
```

```
::= { serialConfigEntry 9 }
```

```
netConfigTable OBJECT-TYPE
SYNTAX      SEQUENCE OF NetConfigEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "A table of netConfigEntries."
 ::= { probeConfig 11 }
```

```
netConfigEntry OBJECT-TYPE
SYNTAX      NetConfigEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "A set of configuration parameters for a particular
    network interface on this device. If the device has no network
    interface, this table is empty.
```

```
    The index is composed of the ifIndex assigned to the
    corresponding interface."
```

```
INDEX { ifIndex }
 ::= { netConfigTable 1 }
```

```
NetConfigEntry ::= SEQUENCE {
    netConfigIPAddress      IPAddress,
    netConfigSubnetMask     IPAddress,
    netConfigStatus         RowStatus
}
```

```
netConfigIPAddress OBJECT-TYPE
SYNTAX      IPAddress
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The IP address of this Net interface. The default value
    for this object is 0.0.0.0. If either the netConfigIPAddress
    or netConfigSubnetMask are 0.0.0.0, then when the device
    boots, it may use BOOTP to try to figure out what these
    values should be. If BOOTP fails, before the device
    can talk on the network, this value must be configured
    (e.g., through a terminal attached to the device). If BOOTP is
```

used, care should be taken to not send BOOTP broadcasts too frequently and to eventually send very infrequently if no replies are received."

::= { netConfigEntry 1 }

netConfigSubnetMask OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The subnet mask of this Net interface. The default value for this object is 0.0.0.0. If either the netConfigIPAddress or netConfigSubnetMask are 0.0.0.0, then when the device boots, it may use BOOTP to try to figure out what these values should be. If BOOTP fails, before the device can talk on the network, this value must be configured (e.g., through a terminal attached to the device). If BOOTP is used, care should be taken to not send BOOTP broadcasts too frequently and to eventually send very infrequently if no replies are received."

::= { netConfigEntry 2 }

netConfigStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The status of this netConfigEntry.

An entry may not exist in the active state unless all objects in the entry have an appropriate value."

::= { netConfigEntry 3 }

netDefaultGateway OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The IP Address of the default gateway. If this value is undefined or unknown, it shall have the value 0.0.0.0."

::= { probeConfig 12 }

-- Trap Destination Table

--

-- This table defines the destination addresses for traps generated
-- from the device. This table maps a community to one or more trap
-- destination entries.

--

```
-- The same trap will be sent to all destinations specified in the
-- entries that have the same trapDestCommunity as the eventCommunity
-- (as defined by RMON MIB). Information in this table will be stored
-- in non-volatile memory. If the device has gone through a hard
-- restart, this information will be reset to its default state.
```

```
trapDestTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF TrapDestEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A list of trap destination entries."
    ::= { probeConfig 13 }
```

```
trapDestEntry OBJECT-TYPE
    SYNTAX      TrapDestEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This entry includes a destination IP address to which to send
        traps for this community."
    INDEX { trapDestIndex }
    ::= { trapDestTable 1 }
```

```
TrapDestEntry ::= SEQUENCE {
    trapDestIndex          Integer32,
    trapDestCommunity     OCTET STRING,
    trapDestProtocol       INTEGER,
    trapDestAddress        OCTET STRING,
    trapDestOwner          OwnerString,
    trapDestStatus        RowStatus
}
```

```
trapDestIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A value that uniquely identifies this trapDestEntry."
    ::= { trapDestEntry 1 }
```

```
trapDestCommunity OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(0..127))
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "A community to which this destination address belongs.
        This entry is associated with any eventEntries in the RMON
```

MIB whose value of eventCommunity is equal to the value of this object. Every time an associated event entry sends a trap due to an event, that trap will be sent to each address in the trapDestTable with a trapDestCommunity equal to eventCommunity.

This object may not be modified if the associated trapDestStatus object is equal to active(1)."

```
::= { trapDestEntry 2 }
```

```
trapDestProtocol OBJECT-TYPE
```

```
SYNTAX      INTEGER {
                ip(1),
                ipx(2)
            }
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"The protocol with which to send this trap."

```
::= { trapDestEntry 3 }
```

```
trapDestAddress OBJECT-TYPE
```

```
SYNTAX      OCTET STRING
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"The address to send traps on behalf of this entry.

If the associated trapDestProtocol object is equal to ip(1), the encoding of this object is the same as the snmpUDPAddress textual convention in [RFC1906]:

```
-- for a SnmpUDPAddress of length 6:
--
-- octets    contents          encoding
-- 1-4       IP-address        network-byte order
-- 5-6       UDP-port          network-byte order
```

If the associated trapDestProtocol object is equal to ipx(2), the encoding of this object is the same as the snmpIPXAddress textual convention in [RFC1906]:

```
-- for a SnmpIPXAddress of length 12:
--
-- octets    contents          encoding
-- 1-4       network-number    network-byte order
-- 5-10      physical-address   network-byte order
-- 11-12     socket-number      network-byte order
```

This object may not be modified if the associated

```
        trapDestStatus object is equal to active(1)."  
 ::= { trapDestEntry 4 }  
  
trapDestOwner OBJECT-TYPE  
    SYNTAX      OwnerString  
    MAX-ACCESS  read-create  
    STATUS      current  
    DESCRIPTION  
        "The entity that configured this entry and is  
        therefore using the resources assigned to it."  
 ::= { trapDestEntry 5 }  
  
trapDestStatus OBJECT-TYPE  
    SYNTAX      RowStatus  
    MAX-ACCESS  read-create  
    STATUS      current  
    DESCRIPTION  
        "The status of this trap destination entry.  
  
        An entry may not exist in the active state unless all  
        objects in the entry have an appropriate value."  
 ::= { trapDestEntry 6 }  
  
-- Serial Connection Table  
--  
-- The device may communicate with a management station using  
-- SLIP.  In order for the device to send traps via SLIP, it must  
-- be able to initiate a connection over the serial interface.  The  
-- serialConnectionTable stores the parameters for such connection  
-- initiation.  
  
serialConnectionTable OBJECT-TYPE  
    SYNTAX      SEQUENCE OF SerialConnectionEntry  
    MAX-ACCESS  not-accessible  
    STATUS      current  
    DESCRIPTION  
        "A list of serialConnectionEntries."  
 ::= { probeConfig 14 }  
  
serialConnectionEntry OBJECT-TYPE  
    SYNTAX      SerialConnectionEntry  
    MAX-ACCESS  not-accessible  
    STATUS      current  
    DESCRIPTION  
        "Configuration for a SLIP link over a serial line."  
    INDEX { serialConnectIndex }  
 ::= { serialConnectionTable 1 }
```

```

SerialConnectionEntry ::= SEQUENCE {
    serialConnectIndex          Integer32,
    serialConnectDestIpAddress  IPAddress,
    serialConnectType           INTEGER,
    serialConnectDialString     ControlString,
    serialConnectSwitchConnectSeq ControlString,
    serialConnectSwitchDisconnectSeq ControlString,
    serialConnectSwitchResetSeq ControlString,
    serialConnectOwner          OwnerString,
    serialConnectStatus         RowStatus
}

serialConnectIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A value that uniquely identifies this serialConnection
        entry."
    ::= { serialConnectionEntry 1 }

serialConnectDestIpAddress OBJECT-TYPE
    SYNTAX      IPAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The IP Address that can be reached at the other end of this
        serial connection.
        This object may not be modified if the associated
        serialConnectStatus object is equal to active(1)."
```

```

    ::= { serialConnectionEntry 2 }

serialConnectType OBJECT-TYPE
    SYNTAX      INTEGER {
        direct(1),
        modem(2),
        switch(3),
        modemSwitch(4)
    }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The type of outgoing connection to make.  If this object
        has the value direct(1), then a direct serial connection
        is assumed.  If this object has the value modem(2),
        then serialConnectDialString will be used to make a modem
        connection.  If this object has the value switch(3),
```

then serialConnectSwitchConnectSeq will be used to establish the connection over a serial data switch, and serialConnectSwitchDisconnectSeq will be used to terminate the connection. If this object has the value modem-switch(4), then a modem connection will be made first followed by the switch connection.

This object may not be modified if the associated serialConnectStatus object is equal to active(1)."

```
DEFVAL { direct }
 ::= { serialConnectionEntry 3 }
```

```
serialConnectDialString OBJECT-TYPE
SYNTAX      ControlString (SIZE(0..255))
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "A control string which specifies how to dial the phone
    number in order to establish a modem connection. The
    string should include dialing prefix and suffix. For
    example: ``^s^MATD9,888-1234^M`` will instruct the Probe
    to send a carriage return followed by the dialing prefix
    ``ATD``, the phone number ``9,888-1234``, and a carriage
    return as the dialing suffix.
    This object may not be modified if the associated
    serialConnectStatus object is equal to active(1)."
```

```
 ::= { serialConnectionEntry 4 }
```

```
serialConnectSwitchConnectSeq OBJECT-TYPE
SYNTAX      ControlString (SIZE(0..255))
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "A control string which specifies how to establish a
    data switch connection.
    This object may not be modified if the associated
    serialConnectStatus object is equal to active(1)."
```

```
 ::= { serialConnectionEntry 5 }
```

```
serialConnectSwitchDisconnectSeq OBJECT-TYPE
SYNTAX      ControlString (SIZE(0..255))
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "A control string which specifies how to terminate a
    data switch connection.
    This object may not be modified if the associated
```

```

    serialConnectStatus object is equal to active(1)."
 ::= { serialConnectionEntry 6 }

```

```

serialConnectSwitchResetSeq OBJECT-TYPE
    SYNTAX      ControlString (SIZE(0..255))
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "A control string which specifies how to reset a data
        switch in the event of a timeout.
        This object may not be modified if the associated
        serialConnectStatus object is equal to active(1)."
 ::= { serialConnectionEntry 7 }

```

```

serialConnectOwner OBJECT-TYPE
    SYNTAX      OwnerString
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The entity that configured this entry and is
        therefore using the resources assigned to it."
 ::= { serialConnectionEntry 8 }

```

```

serialConnectStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The status of this serialConnectionEntry.

        If the manager attempts to set this object to active(1) when
        the serialConnectType is set to modem(2) or modem-switch(4)
        and the serialConnectDialString is a zero-length string or
        cannot be correctly parsed as a ConnectString, the set
        request will be rejected with badValue(3).

        If the manager attempts to set this object to active(1) when
        the serialConnectType is set to switch(3) or modem-switch(4)
        and the serialConnectSwitchConnectSeq,
        the serialConnectSwitchDisconnectSeq, or
        the serialConnectSwitchResetSeq are zero-length strings
        or cannot be correctly parsed as ConnectStrings, the set
        request will be rejected with badValue(3).

        An entry may not exist in the active state unless all
        objects in the entry have an appropriate value."
 ::= { serialConnectionEntry 9 }

```

```
--
-- Extensions to the RMON 1 MIB for RMON 2 devices
--
-- These extensions include the standard LastCreateTime Textual
-- Convention for all control tables, as well as an augmentation of
-- the filter entry that provides variable-length offsets into
-- packets.

-- Each of the following, except for filterDroppedFrames, is a
-- read-only object which, if implemented, automatically appears when
-- the RMON1 row it is associated with is created.

etherStats2Table OBJECT-TYPE
    SYNTAX      SEQUENCE OF EtherStats2Entry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Contains the RMON-2 augmentations to RMON-1."
    ::= { statistics 4 }

etherStats2Entry OBJECT-TYPE
    SYNTAX      EtherStats2Entry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Contains the RMON-2 augmentations to RMON-1."
    AUGMENTS { etherStatsEntry }
    ::= { etherStats2Table 1 }

EtherStats2Entry ::= SEQUENCE {
    etherStatsDroppedFrames Counter32,
    etherStatsCreateTime   LastCreateTime
}

etherStatsDroppedFrames OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of frames which were received by the probe
        and therefore not accounted for in the *StatsDropEvents, but
        for which the probe chose not to count for this entry for
        whatever reason.  Most often, this event occurs when the probe
        is out of some resources and decides to shed load from this
        collection.

        This count does not include packets that were not counted
```

because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { etherStats2Entry 1 }
```

etherStatsCreateTime OBJECT-TYPE

SYNTAX LastCreateTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime when this control entry was last activated. This can be used by the management station to ensure that the table has not been deleted and recreated between polls."

```
::= { etherStats2Entry 2 }
```

historyControl2Table OBJECT-TYPE

SYNTAX SEQUENCE OF HistoryControl2Entry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

```
::= { history 5 }
```

historyControl2Entry OBJECT-TYPE

SYNTAX HistoryControl2Entry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

AUGMENTS { historyControlEntry }

```
::= { historyControl2Table 1 }
```

```
HistoryControl2Entry ::= SEQUENCE {
    historyControlDroppedFrames Counter32
}
```

historyControlDroppedFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for this entry for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this

collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { historyControl2Entry 1 }
```

```
hostControl2Table OBJECT-TYPE
  SYNTAX      SEQUENCE OF HostControl2Entry
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    "Contains the RMON-2 augmentations to RMON-1."
  ::= { hosts 4 }
```

```
hostControl2Entry OBJECT-TYPE
  SYNTAX      HostControl2Entry
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    "Contains the RMON-2 augmentations to RMON-1."
  AUGMENTS { hostControlEntry }
  ::= { hostControl2Table 1 }
```

```
HostControl2Entry ::= SEQUENCE {
  hostControlDroppedFrames Counter32,
  hostControlCreateTime   LastCreateTime
}
```

```
hostControlDroppedFrames OBJECT-TYPE
  SYNTAX      Counter32
  MAX-ACCESS  read-only
  STATUS      current
  DESCRIPTION
    "The total number of frames which were received by the probe
    and therefore not accounted for in the *StatsDropEvents, but
    for which the probe chose not to count for this entry for
    whatever reason. Most often, this event occurs when the probe
    is out of some resources and decides to shed load from this
    collection.
```

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { hostControl2Entry 1 }
```

```
hostControlCreateTime OBJECT-TYPE
```

```
SYNTAX      LastCreateTime
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The value of sysUpTime when this control entry was last
activated. This can be used by the management station to
ensure that the table has not been deleted and recreated
between polls."
```

```
::= { hostControl2Entry 2 }
```

```
matrixControl2Table OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF MatrixControl2Entry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Contains the RMON-2 augmentations to RMON-1."
```

```
::= { matrix 4 }
```

```
matrixControl2Entry OBJECT-TYPE
```

```
SYNTAX      MatrixControl2Entry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Contains the RMON-2 augmentations to RMON-1."
```

```
AUGMENTS { matrixControlEntry }
```

```
::= { matrixControl2Table 1 }
```

```
MatrixControl2Entry ::= SEQUENCE {
```

```
matrixControlDroppedFrames Counter32,
```

```
matrixControlCreateTime LastCreateTime
```

```
}
```

```
matrixControlDroppedFrames OBJECT-TYPE
```

```
SYNTAX      Counter32
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The total number of frames which were received by the probe
and therefore not accounted for in the *StatsDropEvents, but
for which the probe chose not to count for this entry for
whatever reason. Most often, this event occurs when the probe
is out of some resources and decides to shed load from this
collection."
```

```
This count does not include packets that were not counted
```

because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { matrixControl2Entry 1 }
```

matrixControlCreateTime OBJECT-TYPE

SYNTAX LastCreateTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime when this control entry was last activated. This can be used by the management station to ensure that the table has not been deleted and recreated between polls."

```
::= { matrixControl2Entry 2 }
```

channel2Table OBJECT-TYPE

SYNTAX SEQUENCE OF Channel2Entry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

```
::= { filter 3 }
```

channel2Entry OBJECT-TYPE

SYNTAX Channel2Entry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

AUGMENTS { channelEntry }

```
::= { channel2Table 1 }
```

Channel2Entry ::= SEQUENCE {

channelDroppedFrames Counter32,

channelCreateTime LastCreateTime

}

channelDroppedFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for this entry for whatever reason. Most often, this event occurs when the probe

is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { channel2Entry 1 }
```

```
channelCreateTime OBJECT-TYPE
```

```
SYNTAX      LastCreateTime
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The value of sysUpTime when this control entry was last
activated. This can be used by the management station to
ensure that the table has not been deleted and recreated
between polls."
```

```
::= { channel2Entry 2 }
```

```
tokenRingMLStats2Table OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF TokenRingMLStats2Entry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Contains the RMON-2 augmentations to RMON-1."
```

```
::= { statistics 5 }
```

```
tokenRingMLStats2Entry OBJECT-TYPE
```

```
SYNTAX      TokenRingMLStats2Entry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Contains the RMON-2 augmentations to RMON-1."
```

```
AUGMENTS { tokenRingMLStatsEntry }
```

```
::= { tokenRingMLStats2Table 1 }
```

```
TokenRingMLStats2Entry ::= SEQUENCE {
```

```
tokenRingMLStatsDroppedFrames      Counter32,
```

```
tokenRingMLStatsCreateTime          LastCreateTime
```

```
}
```

```
tokenRingMLStatsDroppedFrames OBJECT-TYPE
```

```
SYNTAX      Counter32
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for this entry for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

```
::= { tokenRingMLStats2Entry 1 }
```

```
tokenRingMLStatsCreateTime OBJECT-TYPE
```

```
SYNTAX      LastCreateTime
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"The value of sysUpTime when this control entry was last activated. This can be used by the management station to ensure that the table has not been deleted and recreated between polls."

```
::= { tokenRingMLStats2Entry 2 }
```

```
tokenRingPStats2Table OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF TokenRingPStats2Entry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

"Contains the RMON-2 augmentations to RMON-1."

```
::= { statistics 6 }
```

```
tokenRingPStats2Entry OBJECT-TYPE
```

```
SYNTAX      TokenRingPStats2Entry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

"Contains the RMON-2 augmentations to RMON-1."

```
AUGMENTS { tokenRingPStatsEntry }
```

```
::= { tokenRingPStats2Table 1 }
```

```
TokenRingPStats2Entry ::= SEQUENCE {
```

```
tokenRingPStatsDroppedFrames Counter32,
```

```
tokenRingPStatsCreateTime      LastCreateTime
```

```
}
```

```
tokenRingPStatsDroppedFrames OBJECT-TYPE
```

```
SYNTAX      Counter32
```

MAX-ACCESS read-only
 STATUS current
 DESCRIPTION

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for this entry for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

::= { tokenRingPStats2Entry 1 }

tokenRingPStatsCreateTime OBJECT-TYPE

SYNTAX LastCreateTime
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION

"The value of sysUpTime when this control entry was last activated. This can be used by the management station to ensure that the table has not been deleted and recreated between polls."

::= { tokenRingPStats2Entry 2 }

ringStationControl2Table OBJECT-TYPE

SYNTAX SEQUENCE OF RingStationControl2Entry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

::= { tokenRing 7 }

ringStationControl2Entry OBJECT-TYPE

SYNTAX RingStationControl2Entry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

AUGMENTS { ringStationControlEntry }

::= { ringStationControl2Table 1 }

```
RingStationControl2Entry ::= SEQUENCE {
    ringStationControlDroppedFrames Counter32,
    ringStationControlCreateTime    LastCreateTime
}
```

ringStationControlDroppedFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The total number of frames which were received by the probe and therefore not accounted for in the *StatsDropEvents, but for which the probe chose not to count for this entry for whatever reason. Most often, this event occurs when the probe is out of some resources and decides to shed load from this collection.

This count does not include packets that were not counted because they had MAC-layer errors.

Note that, unlike the dropEvents counter, this number is the exact number of frames dropped."

::= { ringStationControl2Entry 1 }

ringStationControlCreateTime OBJECT-TYPE

SYNTAX LastCreateTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime when this control entry was last activated. This can be used by the management station to ensure that the table has not been deleted and recreated between polls."

::= { ringStationControl2Entry 2 }

sourceRoutingStats2Table OBJECT-TYPE

SYNTAX SEQUENCE OF SourceRoutingStats2Entry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

::= { tokenRing 8 }

sourceRoutingStats2Entry OBJECT-TYPE

SYNTAX SourceRoutingStats2Entry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Contains the RMON-2 augmentations to RMON-1."

AUGMENTS { sourceRoutingStatsEntry }

::= { sourceRoutingStats2Table 1 }

SourceRoutingStats2Entry ::= SEQUENCE {
 sourceRoutingStatsDroppedFrames Counter32,

```
    sourceRoutingStatsCreateTime      LastCreateTime
}

sourceRoutingStatsDroppedFrames OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of frames which were received by the probe
        and therefore not accounted for in the *StatsDropEvents, but
        for which the probe chose not to count for this entry for
        whatever reason.  Most often, this event occurs when the probe
        is out of some resources and decides to shed load from this
        collection.

        This count does not include packets that were not counted
        because they had MAC-layer errors.

        Note that, unlike the dropEvents counter, this number is the
        exact number of frames dropped."
    ::= { sourceRoutingStats2Entry 1 }

sourceRoutingStatsCreateTime OBJECT-TYPE
    SYNTAX      LastCreateTime
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value of sysUpTime when this control entry was last activated.
        This can be used by the management station to ensure that the
        table has not been deleted and recreated between polls."
    ::= { sourceRoutingStats2Entry 2 }

filter2Table OBJECT-TYPE
    SYNTAX      SEQUENCE OF Filter2Entry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Provides a variable-length packet filter feature to the
        RMON-1 filter table."
    ::= { filter 4 }

filter2Entry OBJECT-TYPE
    SYNTAX      Filter2Entry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Provides a variable-length packet filter feature to the
        RMON-1 filter table."
```

```
AUGMENTS { filterEntry }
 ::= { filter2Table 1 }
```

```
Filter2Entry ::= SEQUENCE {
  filterProtocolDirDataLocalIndex      Integer32,
  filterProtocolDirLocalIndex          Integer32
}
```

```
filterProtocolDirDataLocalIndex OBJECT-TYPE
```

```
SYNTAX      Integer32 (0..2147483647)
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"When this object is set to a non-zero value, the filter that it is associated with performs the following operations on every packet:

- 1) - If the packet doesn't match the protocol directory entry identified by this object, discard the packet and exit (i.e., discard the packet if it is not of the identified protocol).
- 2) - If the associated filterProtocolDirLocalIndex is non-zero and the packet doesn't match the protocol directory entry identified by that object, discard the packet and exit
- 3) - If the packet matches, perform the regular filter algorithm as if the beginning of this named protocol is the beginning of the packet, potentially applying the filterOffset value to move further into the packet."

```
DEFVAL { 0 }
```

```
::= { filter2Entry 1 }
```

```
filterProtocolDirLocalIndex OBJECT-TYPE
```

```
SYNTAX      Integer32 (0..2147483647)
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

"When this object is set to a non-zero value, the filter that it is associated with will discard the packet if the packet doesn't match this protocol directory entry."

```
DEFVAL { 0 }
```

```
::= { filter2Entry 2 }
```

```
-- Conformance Macros
```

```
rmon2MIBCompliances OBJECT IDENTIFIER ::= { rmonConformance 1 }
```

```
rmon2MIBGroups      OBJECT IDENTIFIER ::= { rmonConformance 2 }
```

```

rmon2MIBCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION
    "Describes the requirements for conformance to
    the RMON2 MIB"
  MODULE -- this module
    MANDATORY-GROUPS { protocolDirectoryGroup,
                        protocolDistributionGroup,
                        addressMapGroup,
                        nlHostGroup,
                        nlMatrixGroup,
                        usrHistoryGroup,
                        probeInformationGroup }

    GROUP rmon1EnhancementGroup
  DESCRIPTION
    "The rmon1EnhancementGroup is mandatory for systems which
    implement RMON [RFC1757]"
  ::= { rmon2MIBCompliances 1 }

rmon2MIBApplicationLayerCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION
    "Describes the requirements for conformance to
    the RMON2 MIB with Application Layer Enhancements."
  MODULE -- this module
    MANDATORY-GROUPS { protocolDirectoryGroup,
                        protocolDistributionGroup,
                        addressMapGroup,
                        nlHostGroup,
                        nlMatrixGroup,
                        alHostGroup,
                        alMatrixGroup,
                        usrHistoryGroup,
                        probeInformationGroup }

    GROUP rmon1EnhancementGroup
  DESCRIPTION
    "The rmon1EnhancementGroup is mandatory for systems which
    implement RMON [RFC1757]"
  ::= { rmon2MIBCompliances 2 }

protocolDirectoryGroup OBJECT-GROUP
  OBJECTS { protocolDirLastChange,
            protocolDirLocalIndex, protocolDirDescr,
            protocolDirType, protocolDirAddressMapConfig,
            protocolDirHostConfig, protocolDirMatrixConfig,

```

```
        protocolDirOwner, protocolDirStatus }
STATUS    current
DESCRIPTION
    "Lists the inventory of protocols the probe has the capability
    of monitoring and allows the addition, deletion, and
    configuration of entries in this list."
 ::= { rmon2MIBGroups 1 }

protocolDistributionGroup OBJECT-GROUP
OBJECTS { protocolDistControlDataSource,
           protocolDistControlDroppedFrames,
           protocolDistControlCreateTime,
           protocolDistControlOwner, protocolDistControlStatus,
           protocolDistStatsPkts, protocolDistStatsOctets }
STATUS    current
DESCRIPTION
    "Collects the relative amounts of octets and packets for the
    different protocols detected on a network segment."
 ::= { rmon2MIBGroups 2 }

addressMapGroup OBJECT-GROUP
OBJECTS { addressMapInserts, addressMapDeletes,
           addressMapMaxDesiredEntries,
           addressMapControlDataSource,
           addressMapControlDroppedFrames,
           addressMapControlOwner, addressMapControlStatus,
           addressMapPhysicalAddress,
           addressMapLastChange }
STATUS    current
DESCRIPTION
    "Lists MAC address to network address bindings discovered by
    the probe and what interface they were last seen on."
 ::= { rmon2MIBGroups 3 }

nlHostGroup OBJECT-GROUP
OBJECTS { nlHostControlDataSource,
           nlHostControlNlDroppedFrames, nlHostControlNlInserts,
           nlHostControlNlDeletes,
           nlHostControlNlMaxDesiredEntries,
           nlHostControlAlDroppedFrames, nlHostControlAlInserts,
           nlHostControlAlDeletes,
           nlHostControlAlMaxDesiredEntries, nlHostControlOwner,
           nlHostControlStatus, nlHostInPkts, nlHostOutPkts,
           nlHostInOctets, nlHostOutOctets,
           nlHostOutMacNonUnicastPkts, nlHostCreateTime }
STATUS    current
DESCRIPTION
    "Counts the amount of traffic sent from and to each network
```

address discovered by the probe. Note that while the hlHostControlTable also has objects that control an optional alHostTable, implementation of the alHostTable is not required to fully implement this group."

```
::= { rmon2MIBGroups 4 }
```

```
nlMatrixGroup OBJECT-GROUP
```

```
OBJECTS { hlMatrixControlDataSource,
           hlMatrixControlNlDroppedFrames,
           hlMatrixControlNlInserts, hlMatrixControlNlDeletes,
           hlMatrixControlNlMaxDesiredEntries,
           hlMatrixControlAlDroppedFrames,
           hlMatrixControlAlInserts, hlMatrixControlAlDeletes,
           hlMatrixControlAlMaxDesiredEntries,
           hlMatrixControlOwner, hlMatrixControlStatus,
           nlMatrixSDPkts, nlMatrixSDOctets, nlMatrixSDCreateTime,
           nlMatrixDSPkts, nlMatrixDSOctets, nlMatrixDSCreateTime,
           nlMatrixTopNControlMatrixIndex,
           nlMatrixTopNControlRateBase,
           nlMatrixTopNControlTimeRemaining,
           nlMatrixTopNControlGeneratedReports,
           nlMatrixTopNControlDuration,
           nlMatrixTopNControlRequestedSize,
           nlMatrixTopNControlGrantedSize,
           nlMatrixTopNControlStartTime,
           nlMatrixTopNControlOwner, nlMatrixTopNControlStatus,
           nlMatrixTopNProtocolDirLocalIndex,
           nlMatrixTopNSourceAddress, nlMatrixTopNDestAddress,
           nlMatrixTopNPktRate, nlMatrixTopNReversePktRate,
           nlMatrixTopNOctetRate, nlMatrixTopNReverseOctetRate }
```

```
STATUS current
```

```
DESCRIPTION
```

"Counts the amount of traffic sent between each pair of network addresses discovered by the probe. Note that while the hlMatrixControlTable also has objects that control optional alMatrixTables, implementation of the alMatrixTables is not required to fully implement this group."

```
::= { rmon2MIBGroups 5 }
```

```
alHostGroup OBJECT-GROUP
```

```
OBJECTS { alHostInPkts, alHostOutPkts,
           alHostInOctets, alHostOutOctets, alHostCreateTime }
```

```
STATUS current
```

```
DESCRIPTION
```

"Counts the amount of traffic, by protocol, sent from and to each network address discovered by the probe. Implementation of this group requires implementation of the Network Layer Host Group."

```
::= { rmon2MIBGroups 6 }
```

```
alMatrixGroup OBJECT-GROUP
```

```
OBJECTS { alMatrixSDPkts, alMatrixSDOctets, alMatrixSDCreateTime,
alMatrixDSPkts, alMatrixDSOctets, alMatrixDSCreateTime,
alMatrixTopNControlMatrixIndex,
alMatrixTopNControlRateBase,
alMatrixTopNControlTimeRemaining,
alMatrixTopNControlGeneratedReports,
alMatrixTopNControlDuration,
alMatrixTopNControlRequestedSize,
alMatrixTopNControlGrantedSize,
alMatrixTopNControlStartTime,
alMatrixTopNControlOwner, alMatrixTopNControlStatus,
alMatrixTopNProtocolDirLocalIndex,
alMatrixTopNSourceAddress, alMatrixTopNDestAddress,
alMatrixTopNAppProtocolDirLocalIndex,
alMatrixTopNPktRate, alMatrixTopNReversePktRate,
alMatrixTopNOctetRate, alMatrixTopNReverseOctetRate }
```

```
STATUS current
```

```
DESCRIPTION
```

```
"Counts the amount of traffic, by protocol, sent between each
pair of network addresses discovered by the
probe. Implementation of this group requires implementation of
the Network Layer Matrix Group."
```

```
::= { rmon2MIBGroups 7 }
```

```
usrHistoryGroup OBJECT-GROUP
```

```
OBJECTS { usrHistoryControlObjects,
usrHistoryControlBucketsRequested,
usrHistoryControlBucketsGranted,
usrHistoryControlInterval,
usrHistoryControlOwner, usrHistoryControlStatus,
usrHistoryObjectVariable, usrHistoryObjectSampleType,
usrHistoryIntervalStart, usrHistoryIntervalEnd,
usrHistoryAbsValue, usrHistoryValStatus }
```

```
STATUS current
```

```
DESCRIPTION
```

```
"The usrHistoryGroup provides user-defined collection of
historical information from MIB objects on the probe."
```

```
::= { rmon2MIBGroups 8 }
```

```
probeInformationGroup OBJECT-GROUP
```

```
OBJECTS { probeCapabilities,
probeSoftwareRev, probeHardwareRev, probeDateTime }
```

```
STATUS current
```

```
DESCRIPTION
```

```
"This group describes various operating parameters of the
```

```
    probe as well as controlling the local time of the probe."  
 ::= { rmon2MIBGroups 9 }
```

probeConfigurationGroup OBJECT-GROUP

```
OBJECTS { probeResetControl, probeDownloadFile,  
          probeDownloadTFTPServer, probeDownloadAction,  
          probeDownloadStatus,  
          serialMode, serialProtocol, serialTimeout,  
          serialModemInitString, serialModemHangUpString,  
          serialModemConnectResp, serialModemNoConnectResp,  
          serialDialoutTimeout, serialStatus,  
          netConfigIPAddress, netConfigSubnetMask,  
          netConfigStatus, netDefaultGateway,  
          trapDestCommunity, trapDestProtocol, trapDestAddress,  
          trapDestOwner, trapDestStatus,  
          serialConnectDestIpAddress, serialConnectType,  
          serialConnectDialString, serialConnectSwitchConnectSeq,  
          serialConnectSwitchDisconnectSeq,  
          serialConnectSwitchResetSeq,  
          serialConnectOwner, serialConnectStatus }
```

STATUS current

DESCRIPTION

"This group controls the configuration of various operating parameters of the probe."

```
::= { rmon2MIBGroups 10 }
```

rmon1EnhancementGroup OBJECT-GROUP

```
OBJECTS { historyControlDroppedFrames, hostControlDroppedFrames,  
          hostControlCreateTime, matrixControlDroppedFrames,  
          matrixControlCreateTime, channelDroppedFrames,  
          channelCreateTime, filterProtocolDirDataLocalIndex,  
          filterProtocolDirLocalIndex }
```

STATUS current

DESCRIPTION

"This group adds some enhancements to RMON-1 that help management stations."

```
::= { rmon2MIBGroups 11 }
```

rmon1EthernetEnhancementGroup OBJECT-GROUP

```
OBJECTS { etherStatsDroppedFrames, etherStatsCreateTime }
```

STATUS current

DESCRIPTION

"This group adds some enhancements to RMON-1 that help management stations."

```
::= { rmon2MIBGroups 12 }
```

rmon1TokenRingEnhancementGroup OBJECT-GROUP

```
OBJECTS { tokenRingMLStatsDroppedFrames,
```

```
        tokenRingMLStatsCreateTime,  
        tokenRingPStatsDroppedFrames, tokenRingPStatsCreateTime,  
        ringStationControlDroppedFrames,  
        ringStationControlCreateTime,  
        sourceRoutingStatsDroppedFrames,  
        sourceRoutingStatsCreateTime }  
STATUS    current  
DESCRIPTION  
    "This group adds some enhancements to RMON-1 that help  
    management stations."  
 ::= { rmon2MIBGroups 13 }  
END
```

7. Security Considerations

In order to implement this MIB, a probe must capture all packets on the locally-attached network, including packets between third parties. These packets are analyzed to collect network addresses, protocol usage information, and conversation statistics. Data of this nature may be considered sensitive in some environments. In such environments the administrator may wish to restrict SNMP access to the probe.

A probe implementing this MIB is likely to also implement RMON [RFC1757], which includes functions for returning the contents of captured packets, potentially including sensitive user data or passwords. It is recommended that SNMP access to these functions be restricted.

8. Appendix - TimeFilter Implementation Notes

1) Theory of Operation

The TimeFilter mechanism allows an NMS to reduce the number of SNMP transactions required for a 'table-update' operation. Polling of tables that incorporate a 'TimeFilter' INDEX can be reduced to a theoretical minimum (if used correctly). It can be easily implemented by an agent in a way independent of the number of NMS applications using the same time-filtered table.

Although the name 'TimeFilter' may imply that a history of change events is maintained by the agent, this is not the case. A time-filtered-value represents the current value of the object instance, not the 'saved' value at the time indicated by the TimeFilter INDEX value. Note that TimeFilter objects only appear in INDEX clauses (always not-accessible), so their value is never retrieved. By design, the actual value of a TimeFilter instance is not in itself meaningful (it's not a 'last-change-timestamp').

The TimeFilter is a boolean filtering function applied in internal Get* PDU processing. If the 'last-change-time' of the specified instance is less than the particular TimeFilter INDEX value, then the instance is considered 'not-present' (skipped for GetNext and GetBulk PDUs; 'noSuchInstance' or returned to the requester.

1.1) Agent Implementation of a Time-Filtered Table

In implementation, the time-filtered rows (one for each tick of sysUpTime) are only conceptual. The agent simply filters a real table based on:

- * the current value of sysUpTime
- * the TimeFilter value passed in the varbind
- * the last-update timestamp of each requested counter
(agent implementation requirement)

For example, to implement a time-filtered counter, an agent maintains a timestamp in a 32-bit storage location, initialized to zero. This is in addition to whatever instrumentation is needed for the counter.

Each time the counter is updated, the current value of sysUpTime is recorded in the associated timestamp. If this is not possible or practical, then a background polling process must 'refresh' the timestamp by sampling counter values and comparing them to recorded samples. The timestamp update must occur within 5 seconds of the actual change event.

When an agent receives a Get, GetNext, or GetBulk PDU requesting a time-filtered instance, the following agent has determined that the instance is within the MIB view indicated by the community string in the PDU.

```
/* return TRUE if the object is present */
boolean time_filter_test (
    TimeFilter  last_modified_timestamp,
    TimeFilter  index_value_in_pdu )
{
    if (last_modified_timestamp < index_value_in_pdu)
        return FALSE;
    else
        return TRUE;
}
```

The agent applies this function regardless of the lastActivationTime of the conceptual row in question. In other words, counter discontinuities are ignored (i.e. conceptual row deleted and then re-created later). An agent should consider a object instance 'changed' when it is created (either at restart time for scalars and static objects, or row-creation-time for dynamic tables).

Note that using a timeFilter INDEX value of zero removes the filtering functionality, as the instance will always be

1.2) NMS Implementation of a Time-Filtered Table

The particular TimeFilter INDEX values used by an NMS reflect the polling interval of the NMS, relative to the particular agent's notion of sysUpTime.

An NMS needs to maintain one timestamp variable per agent (initialized to zero) for an arbitrary group of time-filtered MIB objects that are gathered together in the same PDU. Each time the Get* PDU is sent, a request for sysUpTime is included. The retrieved sysUpTime value is used as the timeFilter value in the next polling cycle. If a polling sweep of a time-filtered group of objects requires more than one SNMP transaction, then the sysUpTime value retrieved in the first GetResponse PDU of the polling sweep is saved as the next timeFilter value.

The actual last-update time of a given object is not indicated in the returned GetResponse instance identifier, but rather the timeFilter value passed in the Get*Request PDU is returned.

A "time-filtered get-next/bulk-sweep", done once per polling cycle, is a series of GetNext or GetBulk transactions, and is over when one of the following events occurs:

- 1) the TimeFilter index value returned in the GetResponse is different than the TimeFilter index value passed in the GetNext or GetBulk request. Counter values will still be returned beyond this point (until the last-change-time is reached), but most likely the same values will be returned.
- 2) the return PDU includes instances lexicographically greater than the objects expected (i.e. same GetNext semantics as if the TimeFilter wasn't there)
- 3) a noSuchName or other exception/error is returned.

Note that the use of a time-filtered table in combination with a GetRequest PDU neutralizes any optimization that otherwise might be achieved with the TimeFilter, because no PDU transactions are saved. Either the current time-filtered object-value is returned, or a 'noSuchInstance' exception (SNMPv1c) or 'noSuchName' error (SNMPv1) is returned.

If GetBulk PDUs are used, then the value selected for response PDUs generated by the agent, since duplicate entries (one per size. An appropriate of conceptual rows in the time-filtered table if known, or equal to the number of instances expected to fit in a GetResponse PDU without causing a 'tooBig' error from the agent.

2) TimeFilter Example

The following example demonstrates how an NMS and Agent might use a table with a TimeFilter object in the INDEX. A static table is assumed to keep the example simple, but dynamic tables can also be supported.

2.1) General Assumptions

```
fooEntry INDEX { fooTimeMark, fooIfIndex }
FooEntry = SEQUENCE {
    fooTimeMark    TimeFilter,
    fooIfIndex     Integer32,
    fooCounts      Counter32
}
```

The NMS polls the fooTable every 15 seconds and the baseline poll occurs when the agent has been up for 6 seconds, and the NMS has been up for 10 seconds.

There are 2 static rows in this table at system initialization
(fooCounts.0.1 and fooCounts.0.2).

Row 1 was updated as follows:

	SysUpTime	fooCounts.*.1 value	500
1	900	2	
	2300	3	

Row 2 was updated as follows:

SysUpTime	fooCounts.*.2 value
1100	1
1400	2

2.2) SNMP Transactions from NMS Perspective

Time nms-1000:

```
# NMS baseline poll -- get everything since last agent
restart
```

```
# TimeFilter == 0
```

```
get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
         fooCounts.0);
```

```
returns:
```

```
sysUpTime.0 == 600
```

```
fooCounts.0.1 == 1 # incremented at time 500
```

```
fooCounts.0.2 == 0 # visible since created at time 0
```

Time nms-2500:

```
# NMS 1st poll
```

```
# TimeFilter index == 600
```

```
get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
         fooCounts.600);
```

```
returns:
```

```
sysUpTime.0 == 2100
```

```
fooCounts.600.1 == 2 # incremented at time 900
```

```
fooCounts.600.2 == 2 # incremented at times 1100 and
```

```
1400
```

```
fooCounts.601.1 == 2 # indicates end of sweep
```

```
Time nms-4000:
# NMS 2nd poll
# TimeFilter == 2100

get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
         fooCounts.2100);
returns:
    sysUpTime.0 == 3600
    fooCounts.2100.1 == 3 # incremented at time 2300
    fooCounts.2102.1 == 3 # indicates end-of-sweep

# the counter value for row 2 is not returned because
# it hasn't changed since sysUpTime == 2100.
# The next timetick value for row 1 is returned instead
```

```
Time nms-5500:
# NMS 3rd poll
# TimeFilter == 3600

get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
         fooCounts.3600);
returns:
    sysUpTime.0 == 5100
    some-instance-outside-the-fooTable == <don't care>
    some-instance-outside-the-fooTable == <don't care>

# no 'fooTable' counter values at all are returned
because
# neither counter has been updated since sysUpTime ==
3600
```

2.3) Transactions and TimeFilter Maintenance: Agent Perspective

```
Time agt-0:
# initialize fooTable
fooCounts.1 = 0; changed.1 = 0;
fooCounts.2 = 0; changed.2 = 0;

Time agt-500:
# increment fooCounts.1
++fooCounts.1; changed.1 = 500;
```

```
Time agt-600
# answer get-bulk
#   get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
#           fooCounts.0);
# (changed >= 0)
# return both counters

Time agt-900:
# increment fooCounts.1
++fooCounts.1; changed.1 = 900;

Time agt-1100:
# increment fooCounts.2
++fooCounts.2; changed.2 = 1100;

Time agt-1400:
# increment fooCounts.2
++fooCounts.2; changed.2 = 1400;

Time agt-2100
# answer get-bulk
# get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
#           fooCounts.600);
# (changed >= 600)
# return both counters

Time agt-2300:
# increment fooCounts.1
++fooCounts.1; changed.1 = 2300;

Time agt-3600:
# answer get-bulk
# get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
#           fooCounts.2100);
# (changed >= 2100)
# return only fooCounts.1 from the fooTable--twice

Time agt-5100:
# answer get-bulk
# get-bulk(nonRptrs=1, maxReps=2, sysUpTime.0,
#           fooCounts.3600);
# (changed >= 3600)
# return lexicographically-next two MIB instances
```

9. Acknowledgments

This document was produced by the IETF Remote Network Monitoring Working Group.

10. References

- [1] SNMPv2 Working Group, J. Case, K. McCloghrie, M. Rose, S. Waldbusser, "Structure and Identification of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)" RFC 1902, January 1996.
- [2] SNMPv2 Working Group, J. Case, K. McCloghrie, M. Rose, S. Waldbusser, "Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1903 January 1996.
- [3] McCloghrie, K., and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, RFC 1213, March 1991.
- [4] SNMPv2 Working Group, J. Case, K. McCloghrie, M. Rose, S. Waldbusser, "Protocol Operations for version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, January 1996.
- [5] McCloghrie, K., and F. Kastenholtz, "Evolution of the Interfaces Group of MIB-II", RFC 1573, January 1994.
- [6] Information processing systems -- Open Systems Interconnection -- Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization. International Standard 8824, (December, 1987).
- [7] Information processing systems -- Open Systems Interconnection -- Specification of Basic Encoding Rules for Abstract Notation One (ASN.1), International Organization for Standardization. International Standard 8825, (December, 1987).
- [8] Rose, M., Editor, "A Convention for Defining Traps for use with the SNMP", RFC 1215, March 1991.
- [9] Waldbusser, S., "Remote Network Monitoring Management Information Base", RFC 1757, February 1995.

[10] Waldbusser, S., "Token Ring Extensions to the Remote Network Monitoring MIB", RFC 1513, September 1993.

11. Author's Address

Steven Waldbusser
International Network Services

Phone: (415) 254-4251
EMail: waldbusser@ins.com

