

## Token Ring Extensions to the Remote Network Monitoring MIB

### Status of this Memo

This RFC 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" for the standardization state and status of this protocol. Distribution of this memo is unlimited.

### Abstract

This memo defines extensions to the Remote Network Monitoring MIB for managing 802.5 Token Ring networks.

The Remote Network Monitoring MIB, RFC 1271, defines a framework for remote monitoring functions implemented on a network probe. That MIB defines objects broken down into nine functional groups. Some of those functional groups, the statistics and the history groups, have a view of the data-link layer that is specific to the media type and require specific objects to be defined for each media type. RFC 1271 defined those specific objects necessary for Ethernet. This companion memo defines those specific objects necessary for Token Ring LANs.

In addition, this memo defines some additional monitoring functions specifically for Token Ring. These are defined in the Ring Station Group, the Ring Station Order Group, the Ring Station Configuration Group, and the Source Routing Statistics Group.

### Table of Contents

1. The Network Management Framework .....	2
2. Guidelines for implementing RFC1271 objects on a Token Ring network .....	3
2.1 Host Group .....	3
2.2 Matrix Group .....	3
2.3 Filter Group .....	3
2.4 Other comments .....	4
3. Overview of the RMON Token Ring Extensions MIB .....	4
3.1 The Token Ring Statistics Groups .....	4
3.2 The Token Ring History Groups .....	5
3.3 The Token Ring Ring Station Group .....	5

3.4 The Token Ring Ring Station Order Group .....	5
3.5 The Token Ring Ring Station Config Group .....	5
3.6 The Token Ring Source Routing Group .....	5
4. Terminology .....	5
5. Definitions .....	6
5.1 The Token Ring Mac-Layer Statistics Group .....	6
5.2 The Token Ring Promiscuous Statistics Group .....	14
5.3 The Token Ring Mac-Layer History Group .....	19
5.4 The Token Ring Promiscuous History Group .....	27
5.5 The Token Ring Ring Station Group .....	32
5.6 The Token Ring Ring Station Order Group .....	41
5.7 The Token Ring Ring Station Config Group .....	43
5.8 The Token Ring Source Routing Group .....	47
6. References .....	54
7. Acknowledgments .....	55
8. Security Considerations .....	55
9. Author's Address .....	55

## 1. The Network Management Framework

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

STD 16, RFC 1155 [1] which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management. STD 16, RFC 1212 [2] defines a more concise description mechanism, which is wholly consistent with the SMI.

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

STD 15, RFC 1157 [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 STD 16, RFC 1212'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[5] language is used for this purpose. However, STD 16, RFC 1155 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. Guidelines for implementing RFC1271 objects on a Token Ring network

Wherever a MacAddress is to be used in this MIB the source routing bit is stripped off. The resulting address will be consistently valid for all packets sent by a particular node.

### 2.1. Host Group

Only Token Ring isolating errors will increment the error counter for a particular hostEntry. The isolating errors are: LineErrors, BurstErrors, ACErrors, InternalErrors, and AbortErrors. ACErrors will increment the error counter only for the nearest upstream neighbor of the station reporting the error. LineErrors and BurstErrors will increment the error counters for the station reporting the error and its neighbor upstream neighbor. InternalErrors and AbortErrors will increment the error counter for the station reporting the error only. In addition, congestionErrors will also be counted for each hostEntry. These errors will be incremented in the host entry of the station that reports the errors in an error report frame.

The hostOutPkts and hostOutOctets counters shall not be incremented for packets with errors.

### 2.2. Matrix Group

Error counters are never incremented.

### 2.3. Filter Group

The following conditions make up the status bitmask for token ring networks:

bit #	Error
3	First packet after some packets were dropped
4	Packet with the Frame Copied Bit set
5	Packet with the Address Recognized Bit set

For the purpose of the packet match algorithm, the filters assume a 32 byte RIF field. Thus, when matching, the filter is compared to the packet starting at the AC byte of the packet, until the end of the RIF field; then the unused RIF bytes in the filter are skipped and matching proceeds from that point. Any filter "care" bits in the RIF that don't correspond to bytes in the input packet will cause the filter to fail.

#### 2.4. Other comments

Because soft error report packets may be sent with assured delivery, some errors may be accidentally reported twice on devices that perform the RMON function promiscuously.

### 3. Overview of the RMON Token Ring Extensions MIB

The Remote Network Monitoring MIB, RFC 1271, defines a framework for remote monitoring functions implemented on a network probe. That MIB defines objects broken down into nine functional groups. Some of those functional groups, the statistics and the history groups, have a view of the data-link layer that is specific to the media type and require specific objects to be defined for each media type. RFC 1271 defined those specific objects necessary for Ethernet. This MIB defines contains four groups that define those specific objects necessary for Token Ring LANs.

In addition, this memo defines some additional monitoring functions specifically for Token Ring. These are defined in the Ring Station Group, the Ring Station Order Group, the Ring Station Configuration Group, and the Source Routing Statistics Group.

#### 3.1. The Token Ring Statistics Groups

The Token Ring statistics groups contain current utilization and error statistics. The statistics are broken down into two groups, the Token Ring Mac-Layer Statistics Group and the Token Ring Promiscuous Statistics Group. The Token Ring Mac-Layer Statistics Group collects information from Mac Layer, including error reports for the ring and ring utilization of the Mac Layer. The Token Ring Promiscuous Statistics Group collects utilization statistics from data packets collected promiscuously.

### 3.2. The Token Ring History Groups

The Token Ring History Groups contain historical utilization and error statistics. The statistics are broken down into two groups, the Token Ring Mac-Layer History Group and the Token Ring Promiscuous History Group. The Token Ring Mac-Layer History Group collects information from Mac Layer, including error reports for the ring and ring utilization of the Mac Layer. The Token Ring Promiscuous History Group collects utilization statistics from data packets collected promiscuously.

### 3.3. The Token Ring Ring Station Group

The Token Ring Ring Station Group contains statistics and status information associated with each Token Ring station on the local ring. In addition, this group provides status information for each ring being monitored.

### 3.4. The Token Ring Ring Station Order Group

The Token Ring Ring Station Order Group provides the order of the stations on monitored rings.

### 3.5. The Token Ring Ring Station Config Group

The Token Ring Ring Station Config Group manages token ring stations through active means. Any station on a monitored ring may be removed or have configuration information downloaded from it.

### 3.6. The Token Ring Source Routing Group

The Token Ring Source Routing Group contains utilization statistics derived from source routing information optionally present in token ring packets.

## 4. Terminology

The 802.5 specification [7] defines the term "good frame" as a frame that is bounded by a valid SD and ED, is an integral number of octets in length, is composed of only 0 and 1 bits between the SD and the ED, has the FF bits of the GC field equal to 00 or 01, has a valid FCS, and has a minimum of 18 octets between the SD and the ED. This document will use the term "good frame" in the same manner.

## 5. Definitions

```
TOKEN-RING-RMON-MIB DEFINITIONS ::= BEGIN

IMPORTS
    Counter, TimeTicks                FROM RFC1155-SMI
    OBJECT-TYPE                       FROM RFC-1212
    OwnerString, EntryStatus,        -- Textual Conventions
    rmon, statistics, history        FROM RFC1271-MIB;

-- All representations of MAC addresses in this MIB
-- Module use, as a textual convention (i.e. this
-- convention does not affect their encoding), the
-- data type:

MacAddress ::= OCTET STRING (SIZE (6)) -- a 6 octet
                                           -- address in
                                           -- the "canonical"
                                           -- order
-- defined by IEEE 802.1a, i.e., as if it were
-- transmitted least significant bit first, even though
-- 802.5 (in contrast to other 802.x protocols) requires
-- MAC addresses to be transmitted most significant bit
-- first.

TimeInterval ::= INTEGER
    -- A period of time, measured in units of 0.01 seconds.

-- This MIB module uses the extended OBJECT-TYPE macro as
-- defined in [2].

-- Token Ring Remote Network Monitoring MIB

tokenRing                OBJECT IDENTIFIER ::= { rmon 10 }

-- The Token Ring Mac-Layer Statistics Group
--
-- Implementation of this group is optional

tokenRingMLStatsTable OBJECT-TYPE
    SYNTAX SEQUENCE OF TokenRingMLStatsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of Mac-Layer Token Ring statistics
```

```

        entries."
 ::= { statistics 2 }

tokenRingMLStatsEntry OBJECT-TYPE
    SYNTAX TokenRingMLStatsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A collection of Mac-Layer statistics kept for a
         particular Token Ring interface."
    INDEX { tokenRingMLStatsIndex }
 ::= { tokenRingMLStatsTable 1 }

-- As an example, an instance of the
-- tokenRingMLStatsMacOctets object
-- might be named tokenRingMLStatsMacOctets.1

TokenRingMLStatsEntry ::= SEQUENCE {
    tokenRingMLStatsIndex                INTEGER,
    tokenRingMLStatsDataSource           OBJECT IDENTIFIER,
    tokenRingMLStatsDropEvents           Counter,
    tokenRingMLStatsMacOctets            Counter,
    tokenRingMLStatsMacPkts              Counter,
    tokenRingMLStatsRingPurgeEvents     Counter,
    tokenRingMLStatsRingPurgePkts       Counter,
    tokenRingMLStatsBeaconEvents        Counter,
    tokenRingMLStatsBeaconTime          TimeInterval,
    tokenRingMLStatsBeaconPkts          Counter,
    tokenRingMLStatsClaimTokenEvents    Counter,
    tokenRingMLStatsClaimTokenPkts     Counter,
    tokenRingMLStatsNAUNChanges         Counter,
    tokenRingMLStatsLineErrors           Counter,
    tokenRingMLStatsInternalErrors      Counter,
    tokenRingMLStatsBurstErrors          Counter,
    tokenRingMLStatsACErrors             Counter,
    tokenRingMLStatsAbortErrors          Counter,
    tokenRingMLStatsLostFrameErrors     Counter,
    tokenRingMLStatsCongestionErrors    Counter,
    tokenRingMLStatsFrameCopiedErrors   Counter,
    tokenRingMLStatsFrequencyErrors     Counter,
    tokenRingMLStatsTokenErrors         Counter,
    tokenRingMLStatsSoftErrorReports    Counter,
    tokenRingMLStatsRingPolleEvents     Counter,
    tokenRingMLStatsOwner                OwnerString,
    tokenRingMLStatsStatus               EntryStatus
}

```

```

tokenRingMLStatsIndex OBJECT-TYPE
    SYNTAX INTEGER (1..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of this object uniquely identifies this
         tokenRingMLStats entry."
    ::= { tokenRingMLStatsEntry 1 }

tokenRingMLStatsDataSource OBJECT-TYPE
    SYNTAX OBJECT IDENTIFIER
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "This object identifies the source of the data
         that this tokenRingMLStats entry is configured to
         analyze. This source can be any tokenRing
         interface on this device. In order to identify a
         particular interface, this object shall identify
         the instance of the ifIndex object, defined in
         MIB-II [3], for the desired interface. For
         example, if an entry were to receive data from
         interface #1, this object would be set to
         ifIndex.1.

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

        This object may not be modified if the associated
        tokenRingMLStatsStatus object is equal to
        valid(1)."
    ::= { tokenRingMLStatsEntry 2 }

tokenRingMLStatsDropEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of events in which packets were
         dropped by the probe due to lack of resources.
         Note that this number is not necessarily the
         number of packets dropped; it is just the number
         of times this condition has been detected. This
         value is the same as the corresponding
         tokenRingPStatsDropEvents."
    ::= { tokenRingMLStatsEntry 3 }

```

```
tokenRingMLStatsMacOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of octets of data in MAC packets
        (excluding those that were not good frames)
        received on the network (excluding framing bits
        but including FCS octets)."
```

```
 ::= { tokenRingMLStatsEntry 4 }
```

```
tokenRingMLStatsMacPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of MAC packets (excluding
        packets that were not good frames) received."
```

```
 ::= { tokenRingMLStatsEntry 5 }
```

```
tokenRingMLStatsRingPurgeEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of times that the ring enters
        the ring purge state from normal ring state. The
        ring purge state that comes in response to the
        claim token or beacon state is not counted."
```

```
 ::= { tokenRingMLStatsEntry 6 }
```

```
tokenRingMLStatsRingPurgePkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of ring purge MAC packets
        detected by probe."
```

```
 ::= { tokenRingMLStatsEntry 7 }
```

```
tokenRingMLStatsBeaconEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of times that the ring enters a
        beaming state (beaconFrameStreamingState,
        beaconBitStreamingState,
```

```
        beaconSetRecoveryModeState, or
        beaconRingSignalLossState) from a non-beaconing
        state. Note that a change of the source address
        of the beacon packet does not constitute a new
        beacon event."
 ::= { tokenRingMLStatsEntry 8 }

tokenRingMLStatsBeaconTime OBJECT-TYPE
    SYNTAX TimeInterval
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total amount of time that the ring has been
        in the beaconing state."
 ::= { tokenRingMLStatsEntry 9 }

tokenRingMLStatsBeaconPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of beacon MAC packets detected
        by the probe."
 ::= { tokenRingMLStatsEntry 10 }

tokenRingMLStatsClaimTokenEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of times that the ring enters
        the claim token state from normal ring state or
        ring purge state. The claim token state that
        comes in response to a beacon state is not
        counted."
 ::= { tokenRingMLStatsEntry 11 }

tokenRingMLStatsClaimTokenPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of claim token MAC packets
        detected by the probe."
 ::= { tokenRingMLStatsEntry 12 }
```

```
tokenRingMLStatsNAUNChanges OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of NAUN changes detected by the
        probe."
    ::= { tokenRingMLStatsEntry 13 }

tokenRingMLStatsLineErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of line errors reported in error
        reporting packets detected by the probe."
    ::= { tokenRingMLStatsEntry 14 }

tokenRingMLStatsInternalErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of adapter internal errors
        reported in error reporting packets detected by
        the probe."
    ::= { tokenRingMLStatsEntry 15 }

tokenRingMLStatsBurstErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of burst errors reported in
        error reporting packets detected by the probe."
    ::= { tokenRingMLStatsEntry 16 }

tokenRingMLStatsACErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of AC (Address Copied) errors
        reported in error reporting packets detected by
        the probe."
    ::= { tokenRingMLStatsEntry 17 }
```

```
tokenRingMLStatsAbortErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of abort delimiters reported in
        error reporting packets detected by the probe."
    ::= { tokenRingMLStatsEntry 18 }

tokenRingMLStatsLostFrameErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of lost frame errors reported in
        error reporting packets detected by the probe."
    ::= { tokenRingMLStatsEntry 19 }

tokenRingMLStatsCongestionErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of receive congestion errors
        reported in error reporting packets detected by
        the probe."
    ::= { tokenRingMLStatsEntry 20 }

tokenRingMLStatsFrameCopiedErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of frame copied errors reported
        in error reporting packets detected by the probe."
    ::= { tokenRingMLStatsEntry 21 }

tokenRingMLStatsFrequencyErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of frequency errors reported in
        error reporting packets detected by the probe."
    ::= { tokenRingMLStatsEntry 22 }
```

```
tokenRingMLStatsTokenErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of token errors reported in
        error reporting packets detected by the probe."
    ::= { tokenRingMLStatsEntry 23 }

tokenRingMLStatsSoftErrorReports OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of soft error report frames
        detected by the probe."
    ::= { tokenRingMLStatsEntry 24 }

tokenRingMLStatsRingPollEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of ring poll events detected by
        the probe (i.e. the number of ring polls initiated
        by the active monitor that were detected)."
    ::= { tokenRingMLStatsEntry 25 }

tokenRingMLStatsOwner OBJECT-TYPE
    SYNTAX OwnerString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "The entity that configured this entry and is
        therefore using the resources assigned to it."
    ::= { tokenRingMLStatsEntry 26 }

tokenRingMLStatsStatus OBJECT-TYPE
    SYNTAX EntryStatus
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "The status of this tokenRingMLStats entry."
    ::= { tokenRingMLStatsEntry 27 }
```

```

-- The Token Ring Promiscuous Statistics Group
--
-- Implementation of this group is optional

tokenRingPStatsTable OBJECT-TYPE
    SYNTAX SEQUENCE OF TokenRingPStatsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of promiscuous Token Ring statistics
        entries."
    ::= { statistics 3 }

tokenRingPStatsEntry OBJECT-TYPE
    SYNTAX TokenRingPStatsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A collection of promiscuous statistics kept for
        non-MAC packets on a particular Token Ring
        interface."
    INDEX { tokenRingPStatsIndex }
    ::= { tokenRingPStatsTable 1 }

-- As an example, an instance of the
-- tokenRingPStatsDataOctets object
-- might be named tokenRingPStatsDataOctets.1

TokenRingPStatsEntry ::= SEQUENCE {
    tokenRingPStatsIndex                INTEGER,
    tokenRingPStatsDataSource           OBJECT IDENTIFIER,
    tokenRingPStatsDropEvents           Counter,
    tokenRingPStatsDataOctets           Counter,
    tokenRingPStatsDataPkts             Counter,
    tokenRingPStatsDataBroadcastPkts   Counter,
    tokenRingPStatsDataMulticastPkts   Counter,
    tokenRingPStatsDataPkts18to63Octets Counter,
    tokenRingPStatsDataPkts64to127Octets Counter,
    tokenRingPStatsDataPkts128to255Octets Counter,
    tokenRingPStatsDataPkts256to511Octets Counter,
    tokenRingPStatsDataPkts512to1023Octets Counter,
    tokenRingPStatsDataPkts1024to2047Octets Counter,
    tokenRingPStatsDataPkts2048to4095Octets Counter,
    tokenRingPStatsDataPkts4096to8191Octets Counter,
    tokenRingPStatsDataPkts8192to18000Octets Counter,
    tokenRingPStatsDataPktsGreaterThan18000Octets Counter,
    tokenRingPStatsOwner                OwnerString,
    tokenRingPStatsStatus                EntryStatus
}

```

```

}

tokenRingPStatsIndex OBJECT-TYPE
    SYNTAX INTEGER (1..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of this object uniquely identifies this
         tokenRingPStats entry."
    ::= { tokenRingPStatsEntry 1 }

tokenRingPStatsDataSource OBJECT-TYPE
    SYNTAX OBJECT IDENTIFIER
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "This object identifies the source of the data
         that this tokenRingPStats entry is configured to
         analyze. This source can be any tokenRing
         interface on this device. In order to identify a
         particular interface, this object shall identify
         the instance of the ifIndex object, defined in
         MIB-II [3], for the desired interface. For
         example, if an entry were to receive data from
         interface #1, this object would be set to
         ifIndex.1.

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

        This object may not be modified if the associated
        tokenRingPStatsStatus object is equal to
        valid(1)."
    ::= { tokenRingPStatsEntry 2 }

tokenRingPStatsDropEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of events in which packets were
         dropped by the probe due to lack of resources.
         Note that this number is not necessarily the
         number of packets dropped; it is just the number
         of times this condition has been detected. This
         value is the same as the corresponding
         tokenRingMLStatsDropEvents"

```

```
 ::= { tokenRingPStatsEntry 3 }

tokenRingPStatsDataOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of octets of data in good frames
         received on the network (excluding framing bits
         but including FCS octets) in non-MAC packets."
    ::= { tokenRingPStatsEntry 4 }

tokenRingPStatsDataPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of non-MAC packets in good
         frames. received."
    ::= { tokenRingPStatsEntry 5 }

tokenRingPStatsDataBroadcastPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
         that were directed to an LLC broadcast address
         (0xFFFFFFFF or 0xC000FFFFFFFF)."
```

```
        "The total number of good non-MAC frames received
        that were between 18 and 63 octets in length
        inclusive, excluding framing bits but including
        FCS octets."
 ::= { tokenRingPStatsEntry 8 }

tokenRingPStatsDataPkts64to127Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        that were between 64 and 127 octets in length
        inclusive, excluding framing bits but including
        FCS octets."
 ::= { tokenRingPStatsEntry 9 }

tokenRingPStatsDataPkts128to255Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        that were between 128 and 255 octets in length
        inclusive, excluding framing bits but including
        FCS octets."
 ::= { tokenRingPStatsEntry 10 }

tokenRingPStatsDataPkts256to511Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        that were between 256 and 511 octets in length
        inclusive, excluding framing bits but including
        FCS octets."
 ::= { tokenRingPStatsEntry 11 }

tokenRingPStatsDataPkts512to1023Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        that were between 512 and 1023 octets in length
        inclusive, excluding framing bits but including
        FCS octets."
```

```
 ::= { tokenRingPStatsEntry 12 }

tokenRingPStatsDataPkts1024to2047Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
         that were between 1024 and 2047 octets in length
         inclusive, excluding framing bits but including
         FCS octets."
    ::= { tokenRingPStatsEntry 13 }

tokenRingPStatsDataPkts2048to4095Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
         that were between 2048 and 4095 octets in length
         inclusive, excluding framing bits but including
         FCS octets."
    ::= { tokenRingPStatsEntry 14 }

tokenRingPStatsDataPkts4096to8191Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
         that were between 4096 and 8191 octets in length
         inclusive, excluding framing bits but including
         FCS octets."
    ::= { tokenRingPStatsEntry 15 }

tokenRingPStatsDataPkts8192to18000Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
         that were between 8192 and 18000 octets in length
         inclusive, excluding framing bits but including
         FCS octets."
    ::= { tokenRingPStatsEntry 16 }

tokenRingPStatsDataPktsGreaterThan18000Octets OBJECT-TYPE
    SYNTAX Counter
```

```

ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of good non-MAC frames received
    that were greater than 18000 octets in length,
    excluding framing bits but including FCS octets."
 ::= { tokenRingPStatsEntry 17 }

tokenRingPStatsOwner OBJECT-TYPE
    SYNTAX OwnerString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "The entity that configured this entry and is
        therefore using the resources assigned to it."
    ::= { tokenRingPStatsEntry 18 }

tokenRingPStatsStatus OBJECT-TYPE
    SYNTAX EntryStatus
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "The status of this tokenRingPStats entry."
    ::= { tokenRingPStatsEntry 19 }

-- The Token Ring History Groups

-- When an entry in the historyControlTable is created that
-- identifies a token ring interface as its
-- historyControlDataSource, the probe shall create
-- corresponding entries in the tokenRingMLHistoryTable
-- and/or the tokenRingPHistoryTable, depending on which
-- groups it supports.

-- The Token Ring Mac-Layer History Group
--
-- Implementation of this group is optional.
-- Implementation of this group requires implementation of
-- the historyControl group from RFC1271.

tokenRingMLHistoryTable OBJECT-TYPE
    SYNTAX SEQUENCE OF TokenRingMLHistoryEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of Mac-Layer Token Ring statistics

```

```

        entries."
 ::= { history 3 }

tokenRingMLHistoryEntry OBJECT-TYPE
    SYNTAX TokenRingMLHistoryEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A collection of Mac-Layer statistics kept for a
         particular Token Ring interface."
    INDEX { tokenRingMLHistoryIndex,
            tokenRingMLHistorySampleIndex }
 ::= { tokenRingMLHistoryTable 1 }

-- As an example, an instance of the
-- tokenRingMLHistoryMacOctets
-- object might be named tokenRingMLHistoryMacOctets.1.27

TokenRingMLHistoryEntry ::= SEQUENCE {
    tokenRingMLHistoryIndex                INTEGER,
    tokenRingMLHistorySampleIndex          INTEGER,
    tokenRingMLHistoryIntervalStart        TimeTicks,
    tokenRingMLHistoryDropEvents           Counter,
    tokenRingMLHistoryMacOctets            Counter,
    tokenRingMLHistoryMacPkts              Counter,
    tokenRingMLHistoryRingPurgeEvents      Counter,
    tokenRingMLHistoryRingPurgePkts        Counter,
    tokenRingMLHistoryBeaconEvents         Counter,
    tokenRingMLHistoryBeaconTime           TimeInterval,
    tokenRingMLHistoryBeaconPkts           Counter,
    tokenRingMLHistoryClaimTokenEvents     Counter,
    tokenRingMLHistoryClaimTokenPkts       Counter,
    tokenRingMLHistoryNAUNChanges          Counter,
    tokenRingMLHistoryLineErrors           Counter,
    tokenRingMLHistoryInternalErrors        Counter,
    tokenRingMLHistoryBurstErrors           Counter,
    tokenRingMLHistoryACErrors              Counter,
    tokenRingMLHistoryAbortErrors          Counter,
    tokenRingMLHistoryLostFrameErrors       Counter,
    tokenRingMLHistoryCongestionErrors      Counter,
    tokenRingMLHistoryFrameCopiedErrors     Counter,
    tokenRingMLHistoryFrequencyErrors       Counter,
    tokenRingMLHistoryTokenErrors          Counter,
    tokenRingMLHistorySoftErrorReports      Counter,
    tokenRingMLHistoryRingPollevents       Counter,
    tokenRingMLHistoryActiveStations        INTEGER
}

```

```
tokenRingMLHistoryIndex OBJECT-TYPE
    SYNTAX INTEGER (1..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The history of which this entry is a part. The
        history identified by a particular value of this
        index is the same history as identified by the
        same value of historyControlIndex."
    ::= { tokenRingMLHistoryEntry 1 }

tokenRingMLHistorySampleIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "An index that uniquely identifies the particular
        Mac-Layer sample this entry represents among all
        Mac-Layer samples associated with the same
        historyControlEntry. This index starts at 1 and
        increases by one as each new sample is taken."
    ::= { tokenRingMLHistoryEntry 2 }

tokenRingMLHistoryIntervalStart OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    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."
    ::= { tokenRingMLHistoryEntry 3 }

tokenRingMLHistoryDropEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of events in which packets were
```

dropped by the probe due to lack of resources during this sampling interval. Note that this number is not necessarily the number of packets dropped, it is just the number of times this condition has been detected."

```
::= { tokenRingMLHistoryEntry 4 }
```

tokenRingMLHistoryMacOctets OBJECT-TYPE  
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "The total number of octets of data in MAC packets (excluding those that were not good frames) received on the network during this sampling interval (excluding framing bits but including FCS octets)."

```
::= { tokenRingMLHistoryEntry 5 }
```

tokenRingMLHistoryMacPkts OBJECT-TYPE  
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "The total number of MAC packets (excluding those that were not good frames) received during this sampling interval."

```
::= { tokenRingMLHistoryEntry 6 }
```

tokenRingMLHistoryRingPurgeEvents OBJECT-TYPE  
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "The total number of times that the ring entered the ring purge state from normal ring state during this sampling interval. The ring purge state that comes from the claim token or beacon state is not counted."

```
::= { tokenRingMLHistoryEntry 7 }
```

tokenRingMLHistoryRingPurgePkts OBJECT-TYPE  
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "The total number of Ring Purge MAC packets detected by the probe during this sampling

```
        interval."
 ::= { tokenRingMLHistoryEntry 8 }

tokenRingMLHistoryBeaconEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of times that the ring enters a
        beaoning state (beaconFrameStreamingState,
        beaconBitStreamingState,
        beaconSetRecoveryModeState, or
        beaconRingSignalLossState) during this sampling
        interval. Note that a change of the source
        address of the beacon packet does not constitute a
        new beacon event."
 ::= { tokenRingMLHistoryEntry 9 }

tokenRingMLHistoryBeaconTime OBJECT-TYPE
    SYNTAX TimeInterval
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The amount of time that the ring has been in the
        beaoning state during this sampling interval."
 ::= { tokenRingMLHistoryEntry 10 }

tokenRingMLHistoryBeaconPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of beacon MAC packets detected
        by the probe during this sampling interval."
 ::= { tokenRingMLHistoryEntry 11 }

tokenRingMLHistoryClaimTokenEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of times that the ring enters
        the claim token state from normal ring state or
        ring purge state during this sampling interval.
        The claim token state that comes from the beacon
        state is not counted."
 ::= { tokenRingMLHistoryEntry 12 }
```

```
tokenRingMLHistoryClaimTokenPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of claim token MAC packets
         detected by the probe during this sampling
         interval."
    ::= { tokenRingMLHistoryEntry 13 }

tokenRingMLHistoryNAUNChanges OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of NAUN changes detected by the
         probe during this sampling interval."
    ::= { tokenRingMLHistoryEntry 14 }

tokenRingMLHistoryLineErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of line errors reported in error
         reporting packets detected by the probe during
         this sampling interval."
    ::= { tokenRingMLHistoryEntry 15 }

tokenRingMLHistoryInternalErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of adapter internal errors
         reported in error reporting packets detected by
         the probe during this sampling interval."
    ::= { tokenRingMLHistoryEntry 16 }

tokenRingMLHistoryBurstErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of burst errors reported in
         error reporting packets detected by the probe
         during this sampling interval."
    ::= { tokenRingMLHistoryEntry 17 }
```

tokenRingMLHistoryACErrors OBJECT-TYPE

SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The total number of AC (Address Copied) errors reported in error reporting packets detected by the probe during this sampling interval."

::= { tokenRingMLHistoryEntry 18 }

tokenRingMLHistoryAbortErrors OBJECT-TYPE

SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The total number of abort delimiters reported in error reporting packets detected by the probe during this sampling interval."

::= { tokenRingMLHistoryEntry 19 }

tokenRingMLHistoryLostFrameErrors OBJECT-TYPE

SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The total number of lost frame errors reported in error reporting packets detected by the probe during this sampling interval."

::= { tokenRingMLHistoryEntry 20 }

tokenRingMLHistoryCongestionErrors OBJECT-TYPE

SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The total number of receive congestion errors reported in error reporting packets detected by the probe during this sampling interval."

::= { tokenRingMLHistoryEntry 21 }

tokenRingMLHistoryFrameCopiedErrors OBJECT-TYPE

SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION

"The total number of frame copied errors reported in error reporting packets detected by the probe during this sampling interval."

```
 ::= { tokenRingMLHistoryEntry 22 }
```

```
tokenRingMLHistoryFrequencyErrors OBJECT-TYPE
```

```
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
```

```
    "The total number of frequency errors reported in
    error reporting packets detected by the probe
    during this sampling interval."
```

```
 ::= { tokenRingMLHistoryEntry 23 }
```

```
tokenRingMLHistoryTokenErrors OBJECT-TYPE
```

```
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
```

```
    "The total number of token errors reported in
    error reporting packets detected by the probe
    during this sampling interval."
```

```
 ::= { tokenRingMLHistoryEntry 24 }
```

```
tokenRingMLHistorySoftErrorReports OBJECT-TYPE
```

```
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
```

```
    "The total number of soft error report frames
    detected by the probe during this sampling
    interval."
```

```
 ::= { tokenRingMLHistoryEntry 25 }
```

```
tokenRingMLHistoryRingPollEvents OBJECT-TYPE
```

```
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
```

```
    "The total number of ring poll events detected by
    the probe during this sampling interval."
```

```
 ::= { tokenRingMLHistoryEntry 26 }
```

```
tokenRingMLHistoryActiveStations OBJECT-TYPE
```

```
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
```

```
    "The maximum number of active stations on the ring
    detected by the probe during this sampling
```

```

        interval."
 ::= { tokenRingMLHistoryEntry 27}

-- The Token Ring Promiscuous History Group
--
-- Implementation of this group is optional.
-- Implementation of this group requires the implementation
-- of the historyControl group from RFC1271.

tokenRingPHistoryTable OBJECT-TYPE
    SYNTAX SEQUENCE OF TokenRingPHistoryEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of promiscuous Token Ring statistics
        entries."
    ::= { history 4 }

tokenRingPHistoryEntry OBJECT-TYPE
    SYNTAX TokenRingPHistoryEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A collection of promiscuous statistics kept for a
        particular Token Ring interface."
    INDEX { tokenRingPHistoryIndex,
            tokenRingPHistorySampleIndex }
    ::= { tokenRingPHistoryTable 1 }

-- As an example, an instance of the
-- tokenRingPHistoryDataPkts object
-- might be named tokenRingPHistoryDataPkts.1.27

TokenRingPHistoryEntry ::= SEQUENCE {
    tokenRingPHistoryIndex                INTEGER,
    tokenRingPHistorySampleIndex          INTEGER,
    tokenRingPHistoryIntervalStart        TimeTicks,
    tokenRingPHistoryDropEvents           Counter,
    tokenRingPHistoryDataOctets           Counter,
    tokenRingPHistoryDataPkts             Counter,
    tokenRingPHistoryDataBroadcastPkts    Counter,
    tokenRingPHistoryDataMulticastPkts    Counter,
    tokenRingPHistoryDataPkts18to63Octets Counter,
    tokenRingPHistoryDataPkts64to127Octets Counter,
    tokenRingPHistoryDataPkts128to255Octets Counter,
    tokenRingPHistoryDataPkts256to511Octets Counter,
    tokenRingPHistoryDataPkts512to1023Octets Counter,

```

```

tokenRingPHistoryDataPkts1024to2047Octets      Counter,
tokenRingPHistoryDataPkts2048to4095Octets      Counter,
tokenRingPHistoryDataPkts4096to8191Octets      Counter,
tokenRingPHistoryDataPkts8192to18000Octets     Counter,
tokenRingPHistoryDataPktsGreaterThan18000Octets Counter
}

tokenRingPHistoryIndex OBJECT-TYPE
    SYNTAX INTEGER (1..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The history of which this entry is a part. The
        history identified by a particular value of this
        index is the same history as identified by the
        same value of historyControlIndex."
    ::= { tokenRingPHistoryEntry 1 }

tokenRingPHistorySampleIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "An index that uniquely identifies the particular
        sample this entry represents among all samples
        associated with the same historyControlEntry.
        This index starts at 1 and increases by one as
        each new sample is taken."
    ::= { tokenRingPHistoryEntry 2 }

tokenRingPHistoryIntervalStart OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    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."
    ::= { tokenRingPHistoryEntry 3 }

```

```
tokenRingPHistoryDropEvents OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of events in which packets were
        dropped by the probe due to lack of resources
        during this sampling interval. Note that this
        number is not necessarily the number of packets
        dropped, it is just the number of times this
        condition has been detected."
    ::= { tokenRingPHistoryEntry 4 }

tokenRingPHistoryDataOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of octets of data in good frames
        received on the network (excluding framing bits
        but including FCS octets) in non-MAC packets
        during this sampling interval."
    ::= { tokenRingPHistoryEntry 5 }

tokenRingPHistoryDataPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval."
    ::= { tokenRingPHistoryEntry 6 }

tokenRingPHistoryDataBroadcastPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval that were directed
        to an LLC broadcast address (0xFFFFFFFF or
        0xC000FFFFFFFF)."
    ::= { tokenRingPHistoryEntry 7 }

tokenRingPHistoryDataMulticastPkts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
```

## DESCRIPTION

"The total number of good non-MAC frames received during this sampling interval that were directed to a local or global multicast or functional address. Note that this number does not include packets directed to the broadcast address."

::= { tokenRingPHistoryEntry 8 }

## tokenRingPHistoryDataPkts18to63Octets OBJECT-TYPE

SYNTAX Counter

ACCESS read-only

STATUS mandatory

## DESCRIPTION

"The total number of good non-MAC frames received during this sampling interval that were between 18 and 63 octets in length inclusive, excluding framing bits but including FCS octets."

::= { tokenRingPHistoryEntry 9 }

## tokenRingPHistoryDataPkts64to127Octets OBJECT-TYPE

SYNTAX Counter

ACCESS read-only

STATUS mandatory

## DESCRIPTION

"The total number of good non-MAC frames received during this sampling interval that were between 64 and 127 octets in length inclusive, excluding framing bits but including FCS octets."

::= { tokenRingPHistoryEntry 10 }

## tokenRingPHistoryDataPkts128to255Octets OBJECT-TYPE

SYNTAX Counter

ACCESS read-only

STATUS mandatory

## DESCRIPTION

"The total number of good non-MAC frames received during this sampling interval that were between 128 and 255 octets in length inclusive, excluding framing bits but including FCS octets."

::= { tokenRingPHistoryEntry 11 }

## tokenRingPHistoryDataPkts256to511Octets OBJECT-TYPE

SYNTAX Counter

ACCESS read-only

STATUS mandatory

## DESCRIPTION

"The total number of good non-MAC frames received during this sampling interval that were between

```
                256 and 511 octets in length inclusive, excluding
                framing bits but including FCS octets."
 ::= { tokenRingPHistoryEntry 12 }

tokenRingPHistoryDataPkts512to1023Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval that were between
        512 and 1023 octets in length inclusive, excluding
        framing bits but including FCS octets."
 ::= { tokenRingPHistoryEntry 13 }

tokenRingPHistoryDataPkts1024to2047Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval that were between
        1024 and 2047 octets in length inclusive,
        excluding framing bits but including FCS octets."
 ::= { tokenRingPHistoryEntry 14 }

tokenRingPHistoryDataPkts2048to4095Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval that were between
        2048 and 4095 octets in length inclusive,
        excluding framing bits but including FCS octets."
 ::= { tokenRingPHistoryEntry 15 }

tokenRingPHistoryDataPkts4096to8191Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval that were between
        4096 and 8191 octets in length inclusive,
        excluding framing bits but including FCS octets."
 ::= { tokenRingPHistoryEntry 16 }
```

```

tokenRingPHistoryDataPkts8192to18000Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval that were between
        8192 and 18000 octets in length inclusive,
        excluding framing bits but including FCS octets."
    ::= { tokenRingPHistoryEntry 17 }

```

```

tokenRingPHistoryDataPktsGreaterThan18000Octets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good non-MAC frames received
        during this sampling interval that were greater
        than 18000 octets in length, excluding framing
        bits but including FCS octets."
    ::= { tokenRingPHistoryEntry 18 }

```

```

-- The Token Ring Ring Station Group
--
-- Implementation of this group is optional
--

```

```

-- Although the ringStationTable stores entries only for
-- those stations physically attached to the local ring and
-- the number of stations attached to a ring is limited, a
-- probe may still need to free resources when resources
-- grow tight. In such a situation, it is suggested that
-- the probe free only inactive stations, and to
-- first free the stations that have been inactive for the
-- longest time.

```

```

ringStationControlTable OBJECT-TYPE
    SYNTAX SEQUENCE OF RingStationControlEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of ringStation table control entries."
    ::= { tokenRing 1 }

```

```

ringStationControlEntry OBJECT-TYPE
    SYNTAX RingStationControlEntry
    ACCESS not-accessible
    STATUS mandatory

```

```

DESCRIPTION
    "A list of parameters that set up the discovery of
    stations on a particular interface and the
    collection of statistics about these stations."
INDEX { ringStationControlIfIndex }
 ::= { ringStationControlTable 1 }

-- As an example, an instance of the
-- ringStationControlIfIndex object
-- might be named ringStationControlIfIndex.1

RingStationControlEntry ::= SEQUENCE {
    ringStationControlIfIndex          INTEGER,
    ringStationControlTableSize       INTEGER,
    ringStationControlActiveStations  INTEGER,
    ringStationControlRingState       INTEGER,
    ringStationControlBeaconSender    MacAddress,
    ringStationControlBeaconNAUN      MacAddress,
    ringStationControlActiveMonitor   MacAddress,
    ringStationControlOrderChanges    Counter,
    ringStationControlOwner            OwnerString,
    ringStationControlStatus           EntryStatus
}

ringStationControlIfIndex OBJECT-TYPE
    SYNTAX INTEGER (1..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of this object uniquely identifies the
        interface on this remote network monitoring device
        from which ringStation data is collected.  The
        interface identified by a particular value of this
        object is the same interface as identified by the
        same value of the ifIndex object, defined in MIB-
        II [3]."
    ::= { ringStationControlEntry 1 }

ringStationControlTableSize OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of ringStationEntries in the
        ringStationTable associated with this
        ringStationControlEntry."
    ::= { ringStationControlEntry 2 }

```

```
ringStationControlActiveStations OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of active ringStationEntries in the
         ringStationTable associated with this
         ringStationControlEntry."
    ::= { ringStationControlEntry 3 }

ringStationControlRingState OBJECT-TYPE
    SYNTAX INTEGER {
        normalOperation(1),
        ringPurgeState(2),
        claimTokenState(3),
        beaconFrameStreamingState(4),
        beaconBitStreamingState(5),
        beaconRingSignalLossState(6),
        beaconSetRecoveryModeState(7)
    }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The current status of this ring."
    ::= { ringStationControlEntry 4 }

ringStationControlBeaconSender OBJECT-TYPE
    SYNTAX MacAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The address of the sender of the last beacon
         frame received by the probe on this ring.  If no
         beacon frames have been received, this object
         shall be equal to six octets of zero."
    ::= { ringStationControlEntry 5 }

ringStationControlBeaconNAUN OBJECT-TYPE
    SYNTAX MacAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The address of the NAUN in the last beacon frame
         received by the probe on this ring.  If no beacon
         frames have been received, this object shall be
         equal to six octets of zero."
    ::= { ringStationControlEntry 6 }
```

```
ringStationControlActiveMonitor OBJECT-TYPE
    SYNTAX MacAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The address of the Active Monitor on this
        segment.  If this address is unknown, this object
        shall be equal to six octets of zero."
    ::= { ringStationControlEntry 7 }

ringStationControlOrderChanges OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of add and delete events in the
        ringStationOrderTable optionally associated with
        this ringStationControlEntry."
    ::= { ringStationControlEntry 8 }

ringStationControlOwner OBJECT-TYPE
    SYNTAX OwnerString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "The entity that configured this entry and is
        therefore using the resources assigned to it."
    ::= { ringStationControlEntry 9 }

ringStationControlStatus OBJECT-TYPE
    SYNTAX EntryStatus
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "The status of this ringStationControl entry.

        If this object is not equal to valid(1), all
        associated entries in the ringStationTable shall
        be deleted by the agent."
    ::= { ringStationControlEntry 10 }

ringStationTable OBJECT-TYPE
    SYNTAX SEQUENCE OF RingStationEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of ring station entries.  An entry will
        exist for each station that is now or has
```

```

        previously been detected as physically present on
        this ring."
 ::= { tokenRing 2 }

ringStationEntry OBJECT-TYPE
SYNTAX RingStationEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
    "A collection of statistics for a particular
    station that has been discovered on a ring
    monitored by this device."
INDEX { ringStationIfIndex, ringStationMacAddress }
 ::= { ringStationTable 1 }

-- As an example, an instance of the
-- ringStationStationStatus object might be named
-- ringStationStationStatus.1.16.0.90.0.64.131

RingStationEntry ::= SEQUENCE {
    ringStationIfIndex          INTEGER,
    ringStationMacAddress       MacAddress,
    ringStationLastNAUN         MacAddress,
    ringStationStationStatus    INTEGER,
    ringStationLastEnterTime    TimeTicks,
    ringStationLastExitTime     TimeTicks,
    ringStationDuplicateAddresses Counter,
    ringStationInLineErrors     Counter,
    ringStationOutLineErrors    Counter,
    ringStationInternalErrors   Counter,
    ringStationInBurstErrors    Counter,
    ringStationOutBurstErrors   Counter,
    ringStationACErrors         Counter,
    ringStationAbortErrors      Counter,
    ringStationLostFrameErrors  Counter,
    ringStationCongestionErrors Counter,
    ringStationFrameCopiedErrors Counter,
    ringStationFrequencyErrors  Counter,
    ringStationTokenErrors      Counter,
    ringStationInBeaconErrors   Counter,
    ringStationOutBeaconErrors  Counter,
    ringStationInsertions       Counter
}

ringStationIfIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory

```

```

DESCRIPTION
    "The value of this object uniquely identifies the
    interface on this remote network monitoring device
    on which this station was detected.  The interface
    identified by a particular value of this object is
    the same interface as identified by the same value
    of the ifIndex object, defined in MIB-II [3]."
 ::= { ringStationEntry 1 }

ringStationMacAddress OBJECT-TYPE
    SYNTAX MacAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The physical address of this station."
    ::= { ringStationEntry 2 }

ringStationLastNAUN OBJECT-TYPE
    SYNTAX MacAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The physical address of last known NAUN of this
        station."
    ::= { ringStationEntry 3 }

ringStationStationStatus OBJECT-TYPE
    SYNTAX INTEGER {
        active(1),      -- actively participating in ring poll.
        inactive(2),   -- Not participating in ring poll
        forcedRemoval(3) -- Forced off ring by network
                       -- management.
    }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The status of this station on the ring."
    ::= { ringStationEntry 4 }

ringStationLastEnterTime OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of sysUpTime at the time this station
        last entered the ring.  If the time is unknown,
        this value shall be zero."
    ::= { ringStationEntry 5 }

```

```
ringStationLastExitTime OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of sysUpTime at the time the probe
         detected that this station last exited the ring.
         If the time is unknown, this value shall be zero."
    ::= { ringStationEntry 6 }

ringStationDuplicateAddresses OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of times this station experienced a
         duplicate address error."
    ::= { ringStationEntry 7 }

ringStationInLineErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of line errors reported by this
         station in error reporting packets detected by the
         probe."
    ::= { ringStationEntry 8 }

ringStationOutLineErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of line errors reported in error
         reporting packets sent by the nearest active
         downstream neighbor of this station and detected
         by the probe."
    ::= { ringStationEntry 9 }

ringStationInternalErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of adapter internal errors
         reported by this station in error reporting
         packets detected by the probe."
```

```
 ::= { ringStationEntry 10 }

ringStationInBurstErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of burst errors reported by this
        station in error reporting packets detected by the
        probe."
    ::= { ringStationEntry 11 }

ringStationOutBurstErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of burst errors reported in
        error reporting packets sent by the nearest active
        downstream neighbor of this station and detected
        by the probe."
    ::= { ringStationEntry 12 }

ringStationACErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of AC (Address Copied) errors
        reported in error reporting packets sent by the
        nearest active downstream neighbor of this station
        and detected by the probe."
    ::= { ringStationEntry 13 }

ringStationAbortErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of abort delimiters reported by
        this station in error reporting packets detected
        by the probe."
    ::= { ringStationEntry 14 }

ringStationLostFrameErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
```

```
DESCRIPTION
    "The total number of lost frame errors reported by
    this station in error reporting packets detected
    by the probe."
 ::= { ringStationEntry 15 }

ringStationCongestionErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of receive congestion errors
        reported by this station in error reporting
        packets detected by the probe."
 ::= { ringStationEntry 16 }

ringStationFrameCopiedErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of frame copied errors reported
        by this station in error reporting packets
        detected by the probe."
 ::= { ringStationEntry 17 }

ringStationFrequencyErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of frequency errors reported by
        this station in error reporting packets detected
        by the probe."
 ::= { ringStationEntry 18 }

ringStationTokenErrors OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of token errors reported by this
        station in error reporting frames detected by the
        probe."
 ::= { ringStationEntry 19 }

ringStationInBeaconErrors OBJECT-TYPE
    SYNTAX Counter
```

```
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of beacon frames sent by this
    station and detected by the probe."
 ::= { ringStationEntry 20 }

ringStationOutBeaconErrors OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of beacon frames detected by the
    probe that name this station as the NAUN."
 ::= { ringStationEntry 21 }

ringStationInsertions OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The number of times the probe detected this
    station inserting onto the ring."
 ::= { ringStationEntry 22 }

-- The Token Ring Ring Station Order Group
--
-- Implementation of this group is optional
--

-- The ringStationOrderTable

ringStationOrderTable OBJECT-TYPE
SYNTAX SEQUENCE OF RingStationOrderEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
    "A list of ring station entries for stations in
    the ring poll, ordered by their ring-order."
 ::= { tokenRing 3 }

ringStationOrderEntry OBJECT-TYPE
SYNTAX RingStationOrderEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
    "A collection of statistics for a particular
```

```

        station that is active on a ring monitored by this
        device. This table will contain information for
        every interface that has a
        ringStationControlStatus equal to valid."
INDEX { ringStationOrderIfIndex,
        ringStationOrderOrderIndex }
 ::= { ringStationOrderTable 1 }

-- As an example, an instance of the
-- ringStationOrderMacAddress object might be named
-- ringStationOrderMacAddress.1.14

RingStationOrderEntry ::= SEQUENCE {
    ringStationOrderIfIndex          INTEGER,
    ringStationOrderOrderIndex      INTEGER,
    ringStationOrderMacAddress      MacAddress
}

ringStationOrderIfIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of this object uniquely identifies the
        interface on this remote network monitoring device
        on which this station was detected. The interface
        identified by a particular value of this object is
        the same interface as identified by the same value
        of the ifIndex object, defined in MIB-II [3]."
    ::= { ringStationOrderEntry 1 }

ringStationOrderOrderIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "This index denotes the location of this station
        with respect to other stations on the ring. This
        index is one more than the number of hops
        downstream that this station is from the rmon
        probe. The rmon probe itself gets the value one."
    ::= { ringStationOrderEntry 2 }

ringStationOrderMacAddress OBJECT-TYPE
    SYNTAX MacAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION

```

```

        "The physical address of this station."
 ::= { ringStationOrderEntry 3 }

-- The Token Ring Ring Station Config Group
--
-- Implementation of this group is optional.
-- The ring station config group manages token ring nodes
-- through active means.

ringStationConfigControlTable OBJECT-TYPE
    SYNTAX SEQUENCE OF RingStationConfigControlEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of ring station configuration control
         entries."
    ::= { tokenRing 4 }

ringStationConfigControlEntry OBJECT-TYPE
    SYNTAX RingStationConfigControlEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "This entry controls active management of stations
         by the probe. One entry exists in this table for
         each active station in the ringStationTable."
    INDEX { ringStationConfigControlIfIndex,
            ringStationConfigControlMacAddress }
    ::= { ringStationConfigControlTable 1 }

-- As an example, an instance of the
-- ringStationConfigControlRemove object might be named
-- ringStationConfigControlRemove.1.16.0.90.0.64.131

RingStationConfigControlEntry ::= SEQUENCE {
    ringStationConfigControlIfIndex      INTEGER,
    ringStationConfigControlMacAddress   MacAddress,
    ringStationConfigControlRemove      INTEGER,
    ringStationConfigControlUpdateStats  INTEGER
}

ringStationConfigControlIfIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of this object uniquely identifies the

```

```
interface on this remote network monitoring device
on which this station was detected. The interface
identified by a particular value of this object is
the same interface as identified by the same value
of the ifIndex object, defined in MIB-II [3]."
 ::= { ringStationConfigControlEntry 1 }

ringStationConfigControlMacAddress OBJECT-TYPE
SYNTAX MacAddress
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The physical address of this station."
 ::= { ringStationConfigControlEntry 2 }

ringStationConfigControlRemove OBJECT-TYPE
SYNTAX INTEGER {
    stable(1),
    removing(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Setting this object to 'removing(2)' causes a
    Remove Station MAC frame to be sent. The agent
    will set this object to 'stable(1)' after
    processing the request."
 ::= { ringStationConfigControlEntry 3 }

ringStationConfigControlUpdateStats OBJECT-TYPE
SYNTAX INTEGER {
    stable(1),
    updating(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Setting this object to 'updating(2)' causes the
    configuration information associate with this
    entry to be updated. The agent will set this
    object to 'stable(1)' after processing the
    request."
 ::= { ringStationConfigControlEntry 4 }
```

```

-- The ringStationConfig Table
--
-- Entries exist in this table after an active
-- configuration query has completed successfully for
-- a station. This query is initiated by the associated
-- ringStationConfigControlUpdateStats variable.

ringStationConfigTable OBJECT-TYPE
    SYNTAX SEQUENCE OF RingStationConfigEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of configuration entries for stations on a
         ring monitored by this probe."
    ::= { tokenRing 5 }

ringStationConfigEntry OBJECT-TYPE
    SYNTAX RingStationConfigEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A collection of statistics for a particular
         station that has been discovered on a ring
         monitored by this probe."
    INDEX { ringStationConfigIfIndex,
            ringStationConfigMacAddress }
    ::= { ringStationConfigTable 1 }

-- As an example, an instance of the
-- ringStationConfigLocation object might be named
-- ringStationConfigLocation.1.16.0.90.0.64.131

RingStationConfigEntry ::= SEQUENCE {
    ringStationConfigIfIndex          INTEGER,
    ringStationConfigMacAddress       MacAddress,
    ringStationConfigUpdateTime       TimeTicks,
    ringStationConfigLocation         OCTET STRING,
    ringStationConfigMicrocode        OCTET STRING,
    ringStationConfigGroupAddress     OCTET STRING,
    ringStationConfigFunctionalAddress OCTET STRING
}

ringStationConfigIfIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of this object uniquely identifies the

```

```
        interface on this remote network monitoring device
        on which this station was detected.  The interface
        identified by a particular value of this object is
        the same interface as identified by the same value
        of the ifIndex object, defined in MIB-II [3]."
 ::= { ringStationConfigEntry 1 }

ringStationConfigMacAddress OBJECT-TYPE
    SYNTAX MacAddress
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The physical address of this station."
 ::= { ringStationConfigEntry 2 }

ringStationConfigUpdateTime OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of sysUpTime at the time this
        configuration information was last updated
        (completely)."
```

```

 ::= { ringStationConfigEntry 6 }

ringStationConfigFunctionalAddress OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE(4))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "the functional addresses recognized by this
         station."
 ::= { ringStationConfigEntry 7 }

-- The Token Ring Source Routing group
--
-- Implementation of this group is optional.
-- The data in this group is collected from the source
-- routing information potentially present in any token ring
-- packet. This information will be valid only in a pure
-- source route bridging environment. In a transparent
-- bridging or a mixed bridging environment, this
-- information may not be accurate.

sourceRoutingStatsTable OBJECT-TYPE
    SYNTAX SEQUENCE OF SourceRoutingStatsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A list of source routing statistics entries."
 ::= { tokenRing 6 }

sourceRoutingStatsEntry OBJECT-TYPE
    SYNTAX SourceRoutingStatsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A collection of source routing statistics kept
         for a particular Token Ring interface."
    INDEX { sourceRoutingStatsIfIndex }
 ::= { sourceRoutingStatsTable 1 }

-- As an example, an instance of the
-- sourceRoutingStatsInFrames object might be named
-- sourceRoutingStatsInFrames.1

SourceRoutingStatsEntry ::= SEQUENCE {
    sourceRoutingStatsIfIndex          INTEGER,
    sourceRoutingStatsRingNumber      INTEGER,
    sourceRoutingStatsInFrames        Counter,

```

```

    -- in to our net

sourceRoutingStatsOutFrames          Counter,
    -- out from our net

sourceRoutingStatsThroughFrames     Counter,
    -- through our net

sourceRoutingStatsAllRoutesBroadcastFrames Counter,
sourceRoutingStatsSingleRouteBroadcastFrames Counter,
sourceRoutingStatsInOctets          Counter,
sourceRoutingStatsOutOctets         Counter,
sourceRoutingStatsThroughOctets     Counter,
sourceRoutingStatsAllRoutesBroadcastOctets Counter,
sourceRoutingStatsSingleRoutesBroadcastOctets Counter,
sourceRoutingStatsLocalLLCFrames    Counter,
sourceRoutingStats1HopFrames        Counter,
sourceRoutingStats2HopsFrames       Counter,
sourceRoutingStats3HopsFrames       Counter,
sourceRoutingStats4HopsFrames       Counter,
sourceRoutingStats5HopsFrames       Counter,
sourceRoutingStats6HopsFrames       Counter,
sourceRoutingStats7HopsFrames       Counter,
sourceRoutingStats8HopsFrames       Counter,
sourceRoutingStatsMoreThan8HopsFrames Counter,
sourceRoutingStatsOwner              OwnerString,
sourceRoutingStatsStatus             EntryStatus
}

sourceRoutingStatsIfIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The value of this object uniquely identifies the
        interface on this remote network monitoring device
        on which source routing statistics will be
        detected.  The interface identified by a
        particular value of this object is the same
        interface as identified by the same value of the
        ifIndex object, defined in MIB-II [3]."
    ::= { sourceRoutingStatsEntry 1 }

sourceRoutingStatsRingNumber OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION

```

"The ring number of the ring monitored by this entry. When any object in this entry is created, the probe will attempt to discover the ring number. Only after the ring number is discovered will this object be created. After creating an object in this entry, the management station should poll this object to detect when it is created. Only after this object is created can the management station set the sourceRoutingStatsStatus entry to valid(1)."

```
 ::= { sourceRoutingStatsEntry 2 }
```

sourceRoutingStatsInFrames OBJECT-TYPE  
 SYNTAX Counter  
 ACCESS read-only  
 STATUS mandatory  
 DESCRIPTION  
 "The count of frames sent into this ring from another ring."  
 ::= { sourceRoutingStatsEntry 3 }

sourceRoutingStatsOutFrames OBJECT-TYPE  
 SYNTAX Counter  
 ACCESS read-only  
 STATUS mandatory  
 DESCRIPTION  
 "The count of frames sent from this ring to another ring."  
 ::= { sourceRoutingStatsEntry 4 }

sourceRoutingStatsThroughFrames OBJECT-TYPE  
 SYNTAX Counter  
 ACCESS read-only  
 STATUS mandatory  
 DESCRIPTION  
 "The count of frames sent from another ring, through this ring, to another ring."  
 ::= { sourceRoutingStatsEntry 5 }

sourceRoutingStatsAllRoutesBroadcastFrames OBJECT-TYPE  
 SYNTAX Counter  
 ACCESS read-only  
 STATUS mandatory  
 DESCRIPTION  
 "The total number of good frames received that were All Routes Broadcast."  
 ::= { sourceRoutingStatsEntry 6 }

```
sourceRoutingStatsSingleRouteBroadcastFrames OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of good frames received that
         were Single Route Broadcast."
    ::= { sourceRoutingStatsEntry 7 }

sourceRoutingStatsInOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The count of octets in good frames sent into this
         ring from another ring."
    ::= { sourceRoutingStatsEntry 8 }

sourceRoutingStatsOutOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The count of octets in good frames sent from this
         ring to another ring."
    ::= { sourceRoutingStatsEntry 9 }

sourceRoutingStatsThroughOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The count of octets in good frames sent another
         ring, through this ring, to another ring."
    ::= { sourceRoutingStatsEntry 10 }

sourceRoutingStatsAllRoutesBroadcastOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of octets in good frames
         received that were All Routes Broadcast."
    ::= { sourceRoutingStatsEntry 11 }

sourceRoutingStatsSingleRoutesBroadcastOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
```

```
STATUS mandatory
DESCRIPTION
    "The total number of octets in good frames
    received that were Single Route Broadcast."
 ::= { sourceRoutingStatsEntry 12 }

sourceRoutingStatsLocalLLCFrames OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of frames received who had no
    RIF field (or had a RIF field that only included
    the local ring's number) and were not All Route
    Broadcast Frames."
 ::= { sourceRoutingStatsEntry 13 }

sourceRoutingStats1HopFrames OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of frames received whose route
    had 1 hop, were not All Route Broadcast Frames,
    and whose source or destination were on this ring
    (i.e. frames that had a RIF field and had this
    ring number in the first or last entry of the RIF
    field)."
 ::= { sourceRoutingStatsEntry 14 }

sourceRoutingStats2HopsFrames OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of frames received whose route
    had 2 hops, were not All Route Broadcast Frames,
    and whose source or destination were on this ring
    (i.e. frames that had a RIF field and had this
    ring number in the first or last entry of the RIF
    field)."
 ::= { sourceRoutingStatsEntry 15 }

sourceRoutingStats3HopsFrames OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
```

```
        "The total number of frames received whose route
        had 3 hops, were not All Route Broadcast Frames,
        and whose source or destination were on this ring
        (i.e. frames that had a RIF field and had this
        ring number in the first or last entry of the RIF
        field)."
```

```
 ::= { sourceRoutingStatsEntry 16 }
```

```
sourceRoutingStats4HopsFrames OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of frames received whose route
        had 4 hops, were not All Route Broadcast Frames,
        and whose source or destination were on this ring
        (i.e. frames that had a RIF field and had this
        ring number in the first or last entry of the RIF
        field)."
```

```
 ::= { sourceRoutingStatsEntry 17 }
```

```
sourceRoutingStats5HopsFrames OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of frames received whose route
        had 5 hops, were not All Route Broadcast Frames,
        and whose source or destination were on this ring
        (i.e. frames that had a RIF field and had this
        ring number in the first or last entry of the RIF
        field)."
```

```
 ::= { sourceRoutingStatsEntry 18 }
```

```
sourceRoutingStats6HopsFrames OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of frames received whose route
        had 6 hops, were not All Route Broadcast Frames,
        and whose source or destination were on this ring
        (i.e. frames that had a RIF field and had this
        ring number in the first or last entry of the RIF
        field)."
```

```
 ::= { sourceRoutingStatsEntry 19 }
```

```
sourceRoutingStats7HopsFrames OBJECT-TYPE
```

```
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of frames received whose route
    had 7 hops, were not All Route Broadcast Frames,
    and whose source or destination were on this ring
    (i.e. frames that had a RIF field and had this
    ring number in the first or last entry of the RIF
    field)."
```

```
 ::= { sourceRoutingStatsEntry 20 }
```

```
sourceRoutingStats8HopsFrames OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of frames received whose route
    had 8 hops, were not All Route Broadcast Frames,
    and whose source or destination were on this ring
    (i.e. frames that had a RIF field and had this
    ring number in the first or last entry of the RIF
    field)."
```

```
 ::= { sourceRoutingStatsEntry 21 }
```

```
sourceRoutingStatsMoreThan8HopsFrames OBJECT-TYPE
SYNTAX Counter
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of frames received whose route
    had more than 8 hops, were not All Route Broadcast
    Frames, and whose source or destination were on
    this ring (i.e. frames that had a RIF field and
    had this ring number in the first or last entry of
    the RIF field)."
```

```
 ::= { sourceRoutingStatsEntry 22 }
```

```
sourceRoutingStatsOwner OBJECT-TYPE
SYNTAX OwnerString
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "The entity that configured this entry and is
    therefore using the resources assigned to it."
```

```
 ::= { sourceRoutingStatsEntry 23 }
```

```
sourceRoutingStatsStatus OBJECT-TYPE
```

```
SYNTAX EntryStatus
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "The status of this sourceRoutingStats entry."
 ::= { sourceRoutingStatsEntry 24 }
```

END

## 6. References

- [1] Rose M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based internets", STD 16, RFC 1155, Performance Systems International, Hughes LAN Systems, May 1990.
- [2] Rose, M., and K. McCloghrie, Editors, "Concise MIB Definitions", STD 16, RFC 1212, Performance Systems International, Hughes LAN Systems, March 1991.
- [3] McCloghrie K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets", STD 17, RFC 1213, Performance Systems International, March 1991.
- [4] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", STD 15, RFC 1157, SNMP Research, Performance Systems International, Performance Systems International, MIT Laboratory for Computer Science, May 1990.
- [5] Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization. International Standard 8824, December, 1987.
- [6] Waldbusser, S., "Remote Network Monitoring Management Information Base", RFC 1271, CMU, November 1991.
- [7] Token Ring Access Method and Physical Layer Specifications, Institute of Electrical and Electronic Engineers, IEEE Standard 802.5-1989, 1989.

## 7. Acknowledgments

This document was produced by the Token Ring RMON MIB working group.

In addition, the author gratefully acknowledges the comments of the following individuals:

Andrew Bierman	Synoptics
Steve Bostock	Novell
Gary Ellis	Hewlett-Packard
Mike Erlinger	Aerospace Corporation
Robert Graham	Protools
Stephen Grau	Novell
Carl Hayssen	Ungermann-Bass
Jeff Hughes	Hewlett-Packard
Robin Iddon	AXON Networks
Ken Kutzler	Synoptics
To-Choi Lau	Novell
Carl Madison	Startek
Keith McCloghrie	Hughes Lan Systems
Rohit Mital	Protools
Keith Schomburg	IBM
Marshall Rose	Dover Beach Consulting
Mark Therieau	Microcom
Mark van der Pol	Hughes Lan Systems
Brian Wyld	Consultant

## 8. Security Considerations

Security issues are not discussed in this memo.

## 9. Author's Address

Steven Waldbusser  
Carnegie Mellon University  
4910 Forbes Ave.  
Pittsburgh, PA 15213

Phone: (412) 268-6628  
EMail: waldbusser@cmu.edu