

Network Working Group  
Request for Comments: 3295  
Category: Standards Track

H. Sjostrand  
ipUnplugged  
J. Buerkle  
Nortel Networks  
B. Srinivasan  
Cplane  
June 2002

## Definitions of Managed Objects for the General Switch Management Protocol (GSMP)

### 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.

### Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

### Abstract

This memo defines a portion of the Management Information Base (MIB) for the use with the network management protocols in the Internet community. In particular, it describes managed objects for the General Switch Management Protocol (GSMP).

### Table of Contents

1. Introduction.....	2
2. The SNMP Management Framework.....	2
3. Structure of the MIB.....	3
3.1 Overview.....	3
3.2 Scope.....	4
3.3 MIB guideline.....	4
3.4 MIB groups.....	5
3.4.1 GSMP Switch Controller group.....	5
3.4.2 GSMP Switch group.....	6
3.4.3 GSMP Encapsulation groups.....	6
3.4.4 GSMP General group.....	7
3.4.5 The GSMP Notifications Group.....	7
3.5 Textual Conventions.....	8
4. GSMP MIB Definitions.....	9
5. Acknowledgments.....	42

6. References.....	42
7. Intellectual Property Rights.....	44
8. Security Considerations.....	45
9. Authors' Addresses.....	46
10. Full Copyright Statement.....	47

## 1. Introduction

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects for the General Switch Management Protocol (GSMP).

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

## 2. The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- \* An overall architecture, described in RFC 2571 [RFC2571].
- \* Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIV1 and is described in STD 16, RFC 1155 [RFC1155], STD 16, RFC 1212 [RFC1212], and RFC 1215 [RFC1215]. The second version, called SMIV2, is described in STD 58, RFC 2578 [RFC2578], RFC 2579 [RFC2579], and RFC 2580 [RFC2580].
- \* Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and is described in STD 15, RFC 1157 [RFC1157]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and is described in RFC 1901 [RFC1901] and RFC 1906 [RFC1906]. The third version of the message protocol is called SNMPv3 and is described in RFC 1906 [RFC1906], RFC 2572 [RFC2572], and RFC 2574 [RFC2574].
- \* Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats are described in STD 15, RFC 1157 [RFC1157]. A second set of operations and associated PDU formats are described in 1905 [RFC1905].

- \* A set of fundamental applications described in RFC 2573 [RFC2573], and the view-based access control mechanism is described in RFC 2575 [RFC2575].

A more detailed introduction to the current SNMP Management Framework can be found in RFC 2570 [RFC2570].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIV2. A MIB conforming to the SMIV1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine readable information in SMIV2 will be converted into textual descriptions in SMIV1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.

### 3. Structure of the MIB

This memo defines a portion of the Management Information Base (MIB) for the use with network management protocols in the Internet community. In particular, it describes managed objects for the General Switch Management Protocol (GSMP), as defined in [RFC3292].

#### 3.1 Overview

The General Switch Management Protocol (GSMP) is a general purpose protocol to control a label switch. GSMP allows a controller to establish and release connections across the switch, to manage switch ports and to request configuration information or statistics. It also allows the switch to inform the controller of asynchronous events such as a link going down.

The GSMP protocol is asymmetric, the controller being the master and the switch being the slave. Multiple switches may be controlled by a single controller using multiple instantiations of the protocol over separate control connections. Also a switch may be controlled by more than one controller by using the technique of partitioning.

Each instance of a (switch controller, switch partition) adjacency is a session between one switch controller entity and one switch entity. The MIB provides objects to configure/setup these entities to form the GSMP sessions. It also provide objects to monitor these GSMP sessions.

### 3.2 Scope

The GSMP mib is a protocol mib. It contains objects to configure, monitor, and maintain the GSMP protocol entity. It does not provide any information learned via the protocol, such as "all ports config" information.

The relationships between virtual entities, such as Virtual Switch Entities, and "physical" entities, such as Switch Entities, falls outside of the management of GSMP. This also applies for the management of switch partitions. So this is excluded from the GSMP mib.

It is possible to configure which, and how many Switch Controllers are controlling one Switch since every potential session with the switch has to be represented with an Switch entity. It is, however, not possible to define that one Switch Controller shouldn't allow other Switch controllers to control the same switch or partition on the switch. It is assumed that there are mechanisms that synchronise controllers and the configuration of them. This is outside the scope of this mib.

### 3.3 MIB guideline

Two tables are used to configure potential GSMP sessions depending if you are acting as a GSMP switch controller or a GSMP switch. Each row in these tables initiates a GSMP session.

The entity ID is a 48-bit name that is unique within the operational context of the device. A 48-bit IEEE 802 MAC address, if available, MAY be used for the entity ID. If the Ethernet encapsulation is used, the entity ID MUST be the IEEE 802 MAC address of the interface on which the GSMP session is to be setup.

First, the encapsulation of the potential GSMP session shall be defined. If ATM is used, a row in the gsmAtmEncapTable has to be created with the index set to the entity ID. The specified resources should be allocated to GSMP. If TCP/IP is used, a row in the gsmTcpIpEncapTable has to be created with the index set to the entity ID. The specified port shall be allocated to GSMP. No special action is needed if ethernet encapsulation is used.

Then the entity information shall be defined. To create a Switch Entity, an entry in the gsmSwitchTable is created with the index set to the entity ID. To create a Switch Controller Entity, an entry in the gsmControllerTable is created with the index set to the entity ID.

When the row status of the GsmpControllerEntry or GsmpSwitchEntry is set to active (e.g., in the case with ATM or TCP/IP there are active rows with a corresponding entity ID), the adjacency protocol of GSMP is started.

Another table, the gsmpSessionTable, shows the actual sessions that are established or are in the process of being established. Each row represents a specific session between an Entity and a peer. This table carries information about the peer, the session, and parameters that were negotiated by the adjacency procedures. The gsmpSessionTable also contains statistical information regarding the session.

This creation order SHOULD be used by all GSMP managers. This is to avoid clash situations in multiple SNMP manager scenarios where different managers may create competing entries in the different tables.

Entities may very well be configured by other means than SNMP, e.g., the cli command. Such configured entities SHOULD be represented as entries in the tables of this mib and SHOULD be possible to query, and MAY be possible to alter with SNMP.

### 3.4 MIB groups

#### 3.4.1 GSMP Switch Controller group

The controller group is used to configure a potential GSMP session on a Switch Controller. A row in the gsmpControllerTable is created for each such session. If ATM or TCP/IP encapsulation is used, a corresponding row has to be created in these tables before the session adjacency protocol is initiated.

If ATM or TCP/IP is used, encapsulation data is defined in the corresponding encapsulation tables. If ethernet is used, the MAC address of the interface defined for the session is set by the Controller ID object.

The adjacency parameters are defined; such as

- Max supported GSMP version.
- Time between the periodic adjacency messages.
- Controller local port number and instance number.
- Whether partitions are being used and the partition ID for the specific partitions this controller is concerned with if partitions are used.
- The resynchronisation strategy for the session is specified.

The notification mapping is set to specify for with events the corresponding SNMP notifications are sent.

### 3.4.2 GSMP Switch group

The switch group is used to configure a potential GSMP session on a Switch. A row in the gsmpSwitchTable is created for each such session. If ATM or TCP/IP encapsulation is used, a corresponding row has to be created in these tables before the session adjacency protocol is initiated.

If ATM or TCP/IP is used, encapsulation data is defined in the corresponding encapsulation tables. If ethernet is used the MAC address of the interface defined for the session is set by the Switch ID object.

The adjacency parameters are defined; such as

- Max supported GSMP version
- Time between the periodic adjacency messages
- Switch Name, local port number, and instance number.
- Whether partitions are being used and the partition ID for this specific partition if partitions are used.
- The switch type could be set.
- The suggested maximum window size for unacknowledged request messages.

Also, a notification mapping is set to specify for with events the corresponding SNMP notifications are sent.

### 3.4.3 GSMP Encapsulation groups

The ATM Encapsulation Table and the TCP/IP Encapsulation Table provides a way to configure information that are encapsulation specific. The encapsulation data is further specified in [RFC3293].

If ATM encapsulation is used, the interface and the virtual channel are specified.

If TCP/IP is used, the IP address and the port number are specified.

No special config data needed if Ethernet encapsulation is used.

This mib MAY be extended with new, standard or proprietary, GSMP encapsulation types. If a new encapsulation type needs to be added, it SHOULD be done in the form of a new table with the entity ID as an index. A row in that encapsulation table SHOULD be created before any row in a GSMP entity table is created that is using this new GSMP encapsulation.

#### 3.4.4 GSMP General group

The GSMP session table provides a way to monitor and maintain GSMP sessions.

The session is defined by a Switch Controller Entity and Switch Entity pair.

#### 3.4.5 The GSMP Notifications Group

The GSMP Notification Group defines notifications for GSMP entities. These notifications provide a mechanism for a GSMP device to inform the management station of status changes. Also a notification is defined for each type of GSMP events.

The group of notifications consists of the following notifications:

- gsmpSessionDown

This notification is generated when a session is terminating and also reports the final accounting statistics of the session.

- gsmpSessionUp

This notification is generated when a new session is established.

- gsmpSendFailureInd

This notification is generated when a message with a failure indication was sent. This means that this notification identifies a change to the gsmpSessionStatFailureInds object in a row of the gsmpSessionTable.

- gsmpReceivedFailureInd

This notification is generated when a message with a failure indication received. This means that this notification identifies a change to the gsmpSessionStatReceivedFailures object in a row of the gsmpSessionTable.

- gsmpPortUpEvent

This notification is generated when a Port Up Event is either received or sent.

- gsmpPortDownEvent

This notification is generated when a Port Down Event is either received or sent.

- gsmpInvalidLabelEvent

This notification is generated when an Invalid Label Event is either received or sent.

- gsmpNewPortEvent

This notification is generated when New Port Event either is received or sent.

- gsmpDeadPortEvent

This notification is generated when a Dead Port Event is either received or sent.

- gsmpAdjacencyUpdateEvent

This notification is generated when an Adjacency Update Event is either received or sent.

To disable or enable the sending of each notification, the bits in the bitmap are set to 0 or 1 in the Notification mapping objects in the Controller Entity or Switch Entity tables.

The GSMP notification map capability should not be seen as a duplication of the filter mechanism in the snmp notification originator application [RFC2573], but as a compliment, to configure the relation between GSMP events and the SNMP notifications already in the GSMP agent. SNMP notifications and GSMP events operate sometimes on a different timescale, and it may in some applications be devastating for a SNMP application to receive events for each GSMP events. E.g. the invalid label event in a ATM switch scenario may cause mass SNMP notification flooding if mapped to a SNMP notification.

### 3.5 Textual Conventions

The datatypes GsmpNameType, GsmpLabelType, GsmpVersion, GsmpPartitionType, and GsmpPartitionIdType are used as textual conventions in this document. These textual conventions are used for the convenience of humans reading the MIB. Objects defined using these conventions are always encoded by means of the rules that define their primitive type. However, the textual conventions have



special semantics associated with them. Hence, no changes to the SMI or the SNMP are necessary to accommodate these textual conventions which are adopted merely for the convenience of readers.

#### 4. GSMP MIB Definitions

GSMP-MIB DEFINITIONS ::= BEGIN

```

IMPORTS
    OBJECT-TYPE, MODULE-IDENTITY, NOTIFICATION-TYPE,
    Unsigned32, Integer32, mib-2
        FROM SNMPv2-SMI                      -- [RFC2578]
    RowStatus, TruthValue, TimeStamp,
    StorageType, TEXTUAL-CONVENTION
        FROM SNMPv2-TC                      -- [RFC2579]
    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF                    -- [RFC2580]
    ZeroBasedCounter32
        FROM RMON2-MIB                      -- [RFC2021]
    InterfaceIndex
        FROM IF-MIB                         -- [RFC2863]
    AtmVcIdentifier, AtmVpIdentifier
        FROM ATM-TC-MIB                    -- [RFC2514]
    InetAddressType, InetAddress, InetPortNumber
        FROM INET-ADDRESS-MIB ;            -- [RFC3291]

gsmpMIB MODULE-IDENTITY
    LAST-UPDATED "200205310000Z" -- May 31, 2002
    ORGANIZATION "General Switch Management Protocol (gsmp)
        Working Group, IETF"
    CONTACT-INFO
        "WG Charter:
        http://www.ietf.org/html.charters/gsmp-charter.html

        WG-email:      gsmp@ietf.org
        Subscribe:     gsmp-request@ietf.org
        Email Archive:  ftp://ftp.ietf.org/ietf-mail-archive/gsmp/

        WG Chair:      Avri Doria
        Email:          avri@acm.org

        WG Chair:      Kenneth Sundell
        Email:          ksundell@nortelnetworks.com

        Editor:        Hans Sjostrand
        Email:          hans@ipunplugged.com

```

Editor: Joachim Buerkle  
 Email: joachim.buerkle@nortelnetworks.com

Editor: Balaji Srinivasan  
 Email: balaji@cplane.com"

## DESCRIPTION

"This MIB contains managed object definitions for the  
 General Switch Management Protocol, GSMP, version 3"

REVISION "200205310000Z"

DESCRIPTION "Initial Version, published as RFC 3295"

::= { mib-2 98 }

gsmpNotifications	OBJECT IDENTIFIER ::= { gsmpMIB 0 }
gsmpObjects	OBJECT IDENTIFIER ::= { gsmpMIB 1 }
gsmpNotificationsObjects	OBJECT IDENTIFIER ::= { gsmpMIB 2 }
gsmpConformance	OBJECT IDENTIFIER ::= { gsmpMIB 3 }

--\*\*\*\*\*  
 -- GSMP Textual Conventions  
 --\*\*\*\*\*

GsmpNameType ::= TEXTUAL-CONVENTION

STATUS current

## DESCRIPTION

"The Name is a 48-bit quantity.

A 48-bit IEEE 802 MAC address, if  
 available, may be used."

SYNTAX OCTET STRING (SIZE(6))

GsmpPartitionType ::= TEXTUAL-CONVENTION

STATUS current

## DESCRIPTION

"Defining if partitions are used and how the partition id  
 is negotiated. "

SYNTAX INTEGER {  
     noPartition(1),  
     fixedPartitionRequest(2),  
     fixedPartitionAssigned(3)  
 }

GsmpPartitionIdType ::= TEXTUAL-CONVENTION

STATUS current

## DESCRIPTION

"A 8-bit quantity. The format of the Partition ID is not  
 defined in GSMP. If desired, the Partition ID can be  
 divided into multiple sub-identifiers within a single

partition. For example: the Partition ID could be subdivided into a 6-bit partition number and a 2-bit sub-identifier which would allow a switch to support 64 partitions with 4 available IDs per partition."

SYNTAX OCTET STRING (SIZE(1))

GsmVersion ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The version numbers defined for the GSMP protocol. The version numbers used are defined in the specifications of the respective protocol,  
 1 - GSMPv1.1 [RFC1987]  
 2 - GSMPv2.0 [RFC2397]  
 3 - GSMPv3 [RFC3292]  
 Other numbers may be defined for other versions of the GSMP protocol."

SYNTAX Unsigned32

GsmLabelType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The label is structured as a TLV, a tuple, consisting of a Type, a Length, and a Value. The structure is defined in [RFC 3292]. The label TLV is encoded as a 2 octet type field, followed by a 2 octet Length field, followed by a variable length Value field.  
 Additionally, a label field can be composed of many stacked labels that together constitute the label."

SYNTAX OCTET STRING

```
--*****
-- GSMP Entity Objects
--*****
```

```
--
-- Switch Controller Entity table
--
```

gsmControllerTable OBJECT-TYPE

SYNTAX SEQUENCE OF GsmControllerEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table represents the Switch Controller Entities. An entry in this table needs to be configured (created) before a GSMP session might be started."

::= { gsmObjects 1 }

## gsmpControllerEntry OBJECT-TYPE

SYNTAX GsmpControllerEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"An entry in the table showing the data for a specific Switch Controller Entity. If partitions are used, one entity corresponds to one specific switch partition. Depending of the encapsulation used, a corresponding row in the gsmpAtmEncapTable or the gsmpTcpIpEncapTable may have been created."

INDEX { gsmpControllerEntityId }

::= { gsmpControllerTable 1 }

## GsmpControllerEntry ::= SEQUENCE {

gsmpControllerEntityId	GsmpNameType,
gsmpControllerMaxVersion	GsmpVersion,
gsmpControllerTimer	Unsigned32,
gsmpControllerPort	Unsigned32,
gsmpControllerInstance	Unsigned32,
gsmpControllerPartitionType	GsmpPartitionType,
gsmpControllerPartitionId	GsmpPartitionIdType,
gsmpControllerDoResync	TruthValue,
gsmpControllerNotificationMap	BITS,
gsmpControllerSessionState	INTEGER,
gsmpControllerStorageType	StorageType,
gsmpControllerRowStatus	RowStatus

}

## gsmpControllerEntityId OBJECT-TYPE

SYNTAX GsmpNameType

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"The Switch Controller Entity Id is unique within the operational context of the device."

::= { gsmpControllerEntry 1 }

## gsmpControllerMaxVersion OBJECT-TYPE

SYNTAX GsmpVersion

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The max version number of the GSMP protocol being used in this session. The version is negotiated by the adjacency protocol."

DEFVAL { 3 }

```
 ::= { gsmpControllerEntry 2 }

gsmpControllerTimer OBJECT-TYPE
    SYNTAX      Unsigned32(1..255)
    UNITS       "100ms"
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The timer specifies the nominal time between
        periodic adjacency protocol messages. It is a constant
        for the duration of a GSMP session. The timer is
        specified in units of 100ms."
    DEFVAL { 10 }
    ::= { gsmpControllerEntry 3 }

gsmpControllerPort OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The local port number for the Switch Controller
        Entity."
    REFERENCE
        "General Switch Management Protocol V3: Section 3.1.2"
    ::= { gsmpControllerEntry 4 }

gsmpControllerInstance OBJECT-TYPE
    SYNTAX      Unsigned32(1..16777215)
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The instance number for the Switch Controller
        Entity. The Instance number is a 24-bit number
        that should be guaranteed to be unique within
        the recent past and to change when the link
        or node comes back up after going down. Zero is
        not a valid instance number. "
    ::= { gsmpControllerEntry 5 }

gsmpControllerPartitionType OBJECT-TYPE
    SYNTAX      GsmpPartitionType
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "A controller can request the specific partition identifier
        to the session by setting the Partition Type to
        fixedPartitionRequest(2). A controller can let the switch
        decide whether it wants to assign a fixed partition ID or
```

not, by setting the Partition Type to noPartition(1)."  
 ::= { gsmpControllerEntry 6 }

gsmpControllerPartitionId OBJECT-TYPE

SYNTAX GsmpPartitionIdType

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The Id for the specific switch partition that this Switch Controller is concerned with.

If partitions are not used or if the controller lets the switch assigns Partition ID, i.e Partition Type = noPartition(1), then this object is undefined."

::= { gsmpControllerEntry 7 }

gsmpControllerDoResync OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This object specifies whether the controller should resynchronise or reset in case of loss of synchronisation.

If this object is set to true then the Controller should resync with PFLAG=2 (recovered adjacency)."

DEFVAL { true }

::= { gsmpControllerEntry 8 }

gsmpControllerNotificationMap OBJECT-TYPE

SYNTAX BITS {

sessionDown(0),  
 sessionUp(1),  
 sendFailureIndication(2),  
 receivedFailureIndication(3),  
 portUpEvent(4),  
 portDownEvent(5),  
 invalidLabelEvent(6),  
 newPortEvent(7),  
 deadPortEvent(8),  
 adjacencyUpdateEvent(9)

}

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This bitmap defines whether a corresponding SNMP notification should be sent if a GSMP event is received by the Switch Controller. If the bit is set to 1 a notification should be sent. The handling and filtering of the SNMP notifications are then further specified in the

```

        SNMP notification originator application. "
    DEFVAL {{ sessionDown, sessionUp,
        sendFailureIndication, receivedFailureIndication }}
    ::= { gsmpControllerEntry 9 }

gsmpControllerSessionState OBJECT-TYPE
    SYNTAX          INTEGER {
                                null(1),
                                synsent(2),
                                synrcvd(3),
                                estab(4)
                            }
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The state for the existing or potential session that
        this entity is concerned with.
        The NULL state is returned if the proper encapsulation
        data is not yet configured, if the row is not in active
        status or if the session is in NULL state as defined in
        the GSMP specification."
    ::= { gsmpControllerEntry 10}

gsmpControllerStorageType OBJECT-TYPE
    SYNTAX          StorageType
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "The storage type for this controller entity.
        Conceptual rows having the value 'permanent' need not allow
        write-access to any columnar objects in the row."
    DEFVAL { nonVolatile }
    ::= { gsmpControllerEntry 11 }

gsmpControllerRowStatus OBJECT-TYPE
    SYNTAX          RowStatus
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "An object that allows entries in this table to
        be created and deleted using the
        RowStatus convention.
        While the row is in active state it's not
        possible to modify the value of any object
        for that row except the gsmpControllerNotificationMap
        and the gsmpControllerRowStatus objects."
    ::= { gsmpControllerEntry 12 }

```

```

--
-- Switch Entity table
--

gsmpSwitchTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF GsmpSwitchEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table represents the Switch
        Entities. An entry in this table needs to be configured
        (created) before a GSMP session might be started."
    ::= { gsmpObjects 2 }

gsmpSwitchEntry OBJECT-TYPE
    SYNTAX          GsmpSwitchEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in the table showing
        the data for a specific Switch
        Entity. If partitions are used, one entity
        corresponds to one specific switch partition.
        Depending of the encapsulation used,
        a corresponding row in the gsmpAtmEncapTable or the
        gsmpTcpIpEncapTable may have been created."
    INDEX { gsmpSwitchEntityId }
    ::= { gsmpSwitchTable 1 }

GsmpSwitchEntry ::= SEQUENCE {
    gsmpSwitchEntityId          GsmpNameType,
    gsmpSwitchMaxVersion        GsmpVersion,
    gsmpSwitchTimer             Unsigned32,
    gsmpSwitchName              GsmpNameType,
    gsmpSwitchPort              Unsigned32,
    gsmpSwitchInstance          Unsigned32,
    gsmpSwitchPartitionType     GsmpPartitionType,
    gsmpSwitchPartitionId       GsmpPartitionIdType,
    gsmpSwitchNotificationMap   BITS,
    gsmpSwitchSwitchType        OCTET STRING,
    gsmpSwitchWindowSize        Unsigned32,
    gsmpSwitchSessionState      INTEGER,
    gsmpSwitchStorageType       StorageType,
    gsmpSwitchRowStatus          RowStatus
}

gsmpSwitchEntityId OBJECT-TYPE
    SYNTAX          GsmpNameType

```



MAX-ACCESS not-accessible  
STATUS current  
DESCRIPTION  
    "The Switch Entity Id is unique  
    within the operational context of the device. "  
 ::= { gsmpSwitchEntry 1 }

gsmpSwitchMaxVersion OBJECT-TYPE  
SYNTAX GsmVersion  
MAX-ACCESS read-create  
STATUS current  
DESCRIPTION  
    "The max version number of the GSMP protocol being  
    supported by this Switch. The version is negotiated by  
    the adjacency protocol."  
DEFVAL { 3 }  
 ::= { gsmpSwitchEntry 2 }

gsmpSwitchTimer OBJECT-TYPE  
SYNTAX Unsigned32(1..255)  
UNITS "100ms"  
MAX-ACCESS read-create  
STATUS current  
DESCRIPTION  
    "The timer specifies the nominal time between  
    periodic adjacency protocol messages. It is a constant  
    for the duration of a GSMP session. The timer is  
    specified in units of 100ms."  
DEFVAL { 10 }  
 ::= { gsmpSwitchEntry 3 }

gsmpSwitchName OBJECT-TYPE  
SYNTAX GsmNameType  
MAX-ACCESS read-create  
STATUS current  
DESCRIPTION  
    "The name of the Switch. The first three octets must be an  
    Organisationally Unique Identifier (OUI) that identifies  
    the manufacturer of the Switch. This is by default set to  
    the same value as the gsmpSwitchId object if not  
    separately specified. "  
 ::= { gsmpSwitchEntry 4 }

gsmpSwitchPort OBJECT-TYPE  
SYNTAX Unsigned32  
MAX-ACCESS read-create  
STATUS current  
DESCRIPTION

"The local port number for this Switch Entity."

REFERENCE

"General Switch Management Protocol V3: Section 3.1.2"

::= { gsmpSwitchEntry 5 }

gsmpSwitchInstance OBJECT-TYPE

SYNTAX Unsigned32(1..16777215)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The instance number for the Switch Entity.

The Instance number is a 24-bit number that should be guaranteed to be unique within the recent past and to change when the link or node comes back up after going down. Zero is not a valid instance number."

::= { gsmpSwitchEntry 6 }

gsmpSwitchPartitionType OBJECT-TYPE

SYNTAX GsmpPartitionType

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A switch can assign the specific partition identifier to the session by setting the Partition Type to fixedPartitionAssigned(3). A switch can specify that no partitions are handled in the session by setting the Partition Type to noPartition(1)."

::= { gsmpSwitchEntry 7 }

gsmpSwitchPartitionId OBJECT-TYPE

SYNTAX GsmpPartitionIdType

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The Id for this specific switch partition that the switch entity represents. If partitions are not used, i.e. Partition Type = noPartition(1), then this object is undefined."

::= { gsmpSwitchEntry 8 }

gsmpSwitchNotificationMap OBJECT-TYPE

SYNTAX BITS {  
     sessionDown(0),  
     sessionUp(1),  
     sendFailureIndication(2),  
     receivedFailureIndication(3),  
     portUpEvent(4),

```

        portDownEvent(5),
        invalidLabelEvent(6),
        newPortEvent(7),
        deadPortEvent(8),
        adjacencyUpdateEvent(9)
    }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "This bitmap defines whether a corresponding SNMP
    notification should be sent if an GSMP event is sent
    by the Switch Entity. If the bit is set to 1 a
    notification should be sent. The handling and filtering of
    the SNMP notifications are then further specified in the
    SNMP notification originator application. "
DEFVAL {{ sessionDown, sessionUp,
        sendFailureIndication, receivedFailureIndication }}
::= { gsmpSwitchEntry 9 }

gsmpSwitchSwitchType OBJECT-TYPE
SYNTAX          OCTET STRING (SIZE(2))
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "A 16-bit field allocated by the manufacturer
    of the switch. The Switch Type
    identifies the product. When the Switch Type is combined
    with the OUI from the Switch Name the product is
    uniquely identified. "
::= { gsmpSwitchEntry 10 }

gsmpSwitchWindowSize OBJECT-TYPE
SYNTAX          Unsigned32(1..65535)
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The maximum number of unacknowledged request messages
    that may be transmitted by the controller without the
    possibility of loss. This field is used to prevent
    request messages from being lost in the switch because of
    overflow in the receive buffer. The field is a hint to
    the controller."
::= { gsmpSwitchEntry 11 }

gsmpSwitchSessionState OBJECT-TYPE
SYNTAX          INTEGER {
                                null(1),
                                synsent(2),

```

```

                                synrcvd(3),
                                estab(4)
                                }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The state for the existing or potential session that
    this entity is concerned with.
    The NULL state is returned if the proper encapsulation
    data is not yet configured, if the row is not in active
    status or if the session is in NULL state as defined in
    the GSMP specification."
 ::= { gsmpSwitchEntry 12}

gsmpSwitchStorageType OBJECT-TYPE
    SYNTAX      StorageType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The storage type for this switch entity.
        Conceptual rows having the value 'permanent' need not allow
        write-access to any columnar objects in the row."
    DEFVAL { nonVolatile }
    ::= { gsmpSwitchEntry 13 }

gsmpSwitchRowStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "An object that allows entries in this table to
        be created and deleted using the
        RowStatus convention.
        While the row is in active state it's not
        possible to modify the value of any object
        for that row except the gsmpSwitchNotificationMap
        and the gsmpSwitchRowStatus objects."
    ::= { gsmpSwitchEntry 14 }

--*****
-- GSMP Encapsulation Objects
--*****

--
-- GSMP ATM Encapsulation Table
--

gsmpAtmEncapTable OBJECT-TYPE

```

```

SYNTAX          SEQUENCE OF GsmAtmEncapEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "This table contains the atm encapsulation data
    for the Controller or Switch that uses atm aal5 as
    encapsulation. "
 ::= { gsmpObjects 3 }

```

#### gsmpAtmEncapEntry OBJECT-TYPE

```

SYNTAX          GsmAtmEncapEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "An entry in the table showing
    the encapsulation data for a specific
    Switch Controller entity or Switch entity."
INDEX { gsmpAtmEncapEntityId }
 ::= { gsmpAtmEncapTable 1 }

```

```

GsmAtmEncapEntry ::= SEQUENCE {
    gsmpAtmEncapEntityId      GsmNameType,
    gsmpAtmEncapIfIndex       InterfaceIndex,
    gsmpAtmEncapVpi           AtmVpIdentifier,
    gsmpAtmEncapVci           AtmVcIdentifier,
    gsmpAtmEncapStorageType   StorageType,
    gsmpAtmEncapRowStatus     RowStatus
}

```

#### gsmpAtmEncapEntityId OBJECT-TYPE

```

SYNTAX          GsmNameType
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "The Controller Id or Switch Id that is unique
    within the operational context of the device. "
 ::= { gsmpAtmEncapEntry 1 }

```

#### gsmpAtmEncapIfIndex OBJECT-TYPE

```

SYNTAX          InterfaceIndex
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The interface index for the virtual channel over which
    the GSMP session is established, i.e., the GSMP control
    channel for LLC/SNAP encapsulated GSMP messages on an
    ATM data link layer."
 ::= { gsmpAtmEncapEntry 2 }

```

```
gsmpAtmEncapVpi OBJECT-TYPE
    SYNTAX      AtmVpIdentifier
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        " The VPI value for the virtual channel over which the
          GSMP session is established, i.e., the GSMP control
          channel for LLC/SNAP encapsulated GSMP messages on an
          ATM data link layer."
    DEFVAL { 0 }
    ::= { gsmpAtmEncapEntry 3 }

gsmpAtmEncapVci OBJECT-TYPE
    SYNTAX      AtmVcIdentifier
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        " The VCI value for the virtual channel over which the
          GSMP session is established, i.e., the GSMP control
          channel for LLC/SNAP encapsulated GSMP messages on an
          ATM data link layer."
    DEFVAL { 15 }
    ::= { gsmpAtmEncapEntry 4 }

gsmpAtmEncapStorageType OBJECT-TYPE
    SYNTAX      StorageType
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The storage type for this entry. It should have the same
          value as the StorageType in the referring Switch
          Controller entity or Switch entity."
    DEFVAL { nonVolatile }
    ::= { gsmpAtmEncapEntry 5 }

gsmpAtmEncapRowStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "An object that allows entries in this table to
          be created and deleted using the
          RowStatus convention.
          While the row is in active state it's not
          possible to modify the value of any object
          for that row except the gsmpAtmEncapRowStatus object."
    ::= { gsmpAtmEncapEntry 6 }
```

```

--
-- GSMP TCP/IP Encapsulation Table
--

gsmpTcpIpEncapTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF GsmpTcpIpEncapEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table contains the encapsulation data
        for the Controller or Switch that uses TCP/IP as
        encapsulation."
    ::= { gsmpObjects 4 }

gsmpTcpIpEncapEntry OBJECT-TYPE
    SYNTAX          GsmpTcpIpEncapEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in the table showing
        the encapsulation data for a specific
        Switch Controller entity or Switch entity."
    INDEX { gsmpTcpIpEncapEntryId }
    ::= { gsmpTcpIpEncapTable 1 }

GsmpTcpIpEncapEntry ::= SEQUENCE {
    gsmpTcpIpEncapEntryId          GsmpNameType,
    gsmpTcpIpEncapAddressType      InetAddressType,
    gsmpTcpIpEncapAddress          InetAddress,
    gsmpTcpIpEncapPortNumber       InetPortNumber,
    gsmpTcpIpEncapStorageType      StorageType,
    gsmpTcpIpEncapRowStatus        RowStatus
}

gsmpTcpIpEncapEntryId OBJECT-TYPE
    SYNTAX          GsmpNameType
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "The Controller or Switch Id is unique
        within the operational context of the device. "
    ::= { gsmpTcpIpEncapEntry 1 }

gsmpTcpIpEncapAddressType OBJECT-TYPE
    SYNTAX          InetAddressType
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION

```

```

    "The type of address in gsmpTcpIpEncapAddress."
 ::= { gsmpTcpIpEncapEntry 2 }

```

```

gsmpTcpIpEncapAddress OBJECT-TYPE
    SYNTAX      InetAddress
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The IPv4 or IPv6 address used for
         the GSMP session peer."
 ::= { gsmpTcpIpEncapEntry 3 }

```

```

gsmpTcpIpEncapPortNumber OBJECT-TYPE
    SYNTAX      InetPortNumber
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The TCP port number used for the TCP session
         establishment to the GSMP peer."
    DEFVAL { 6068 }
 ::= { gsmpTcpIpEncapEntry 4 }

```

```

gsmpTcpIpEncapStorageType OBJECT-TYPE
    SYNTAX      StorageType
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "The storage type for this entry. It should have the same
         value as the StorageType in the referring Switch
         Controller entity or Switch entity."
    DEFVAL { nonVolatile }
 ::= { gsmpTcpIpEncapEntry 5 }

```

```

gsmpTcpIpEncapRowStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "An object that allows entries in this table to
         be created and deleted using the
         RowStatus convention.
         While the row is in active state it's not
         possible to modify the value of any object
         for that row except the gsmpTcpIpEncapRowStatus object."
 ::= { gsmpTcpIpEncapEntry 6 }

```

```

--*****
-- GSMP Session Objects

```



```
--*****
```

```
--
```

```
-- GSMP Session table
```

```
--
```

```
gsmpSessionTable OBJECT-TYPE
```

```
    SYNTAX          SEQUENCE OF GsmpSessionEntry
```

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

```
    STATUS          current
```

```
    DESCRIPTION
```

```
        "This table represents the sessions between
        Controller and Switch pairs. "
```

```
    ::= { gsmpObjects 5 }
```

```
gsmpSessionEntry OBJECT-TYPE
```

```
    SYNTAX          GsmpSessionEntry
```

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

```
    STATUS          current
```

```
    DESCRIPTION
```

```
        "An entry in the table showing
        the session data for a specific Controller and
        Switch pair. Also, statistics for this specific
        session is shown."
```

```
    INDEX { gsmpSessionThisSideId, gsmpSessionFarSideId }
```

```
    ::= { gsmpSessionTable 1 }
```

```
GsmpSessionEntry ::= SEQUENCE {
```

```
    gsmpSessionThisSideId
```

```
    GsmpNameType,
```

```
    gsmpSessionFarSideId
```

```
    GsmpNameType,
```

```
    gsmpSessionVersion
```

```
    GsmpVersion,
```

```
    gsmpSessionTimer
```

```
    Integer32,
```

```
    gsmpSessionPartitionId
```

```
    GsmpPartitionIdType,
```

```
    gsmpSessionAdjacencyCount
```

```
    Unsigned32,
```

```
    gsmpSessionFarSideName
```

```
    GsmpNameType,
```

```
    gsmpSessionFarSidePort
```

```
    Unsigned32,
```

```
    gsmpSessionFarSideInstance
```

```
    Unsigned32,
```

```
    gsmpSessionLastFailureCode
```

```
    Unsigned32,
```

```
    gsmpSessionDiscontinuityTime
```

```
    TimeStamp,
```

```
    gsmpSessionStartUptime
```

```
    TimeStamp,
```

```
    gsmpSessionStatSentMessages
```

```
    ZeroBasedCounter32,
```

```
    gsmpSessionStatFailureInds
```

```
    ZeroBasedCounter32,
```

```
    gsmpSessionStatReceivedMessages
```

```
    ZeroBasedCounter32,
```

```
    gsmpSessionStatReceivedFailures
```

```
    ZeroBasedCounter32,
```

```
    gsmpSessionStatPortUpEvents
```

```
    ZeroBasedCounter32,
```

```
    gsmpSessionStatPortDownEvents
```

```
    ZeroBasedCounter32,
```

```
    gsmpSessionStatInvLabelEvents
```

```
    ZeroBasedCounter32,
```

```
    gsmpSessionStatNewPortEvents
```

```
    ZeroBasedCounter32,
```

```
gsmpSessionStatDeadPortEvents      ZeroBasedCounter32,  
gsmpSessionStatAdjUpdateEvents     ZeroBasedCounter32  
}
```

gsmpSessionThisSideId OBJECT-TYPE

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

DESCRIPTION

"This side ID uniquely identifies the entity that this session relates to within the operational context of the device. "

::= { gsmpSessionEntry 1 }

gsmpSessionFarSideId OBJECT-TYPE

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

DESCRIPTION

"The Far side ID uniquely identifies the entity that this session is established against. "

::= { gsmpSessionEntry 2 }

gsmpSessionVersion OBJECT-TYPE

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

DESCRIPTION

"The version number of the GSMP protocol being used in this session. The version is the result of the negotiation by the adjacency protocol."

::= { gsmpSessionEntry 3 }

gsmpSessionTimer OBJECT-TYPE

```
SYNTAX          Integer32  
UNITS           "100ms"  
MAX-ACCESS      read-only  
STATUS          current
```

DESCRIPTION

"The timer specifies the time remaining until the adjacency timer expires. The object could take negative values since if no valid GSMP messages are received in any period of time in excess of three times the value of the Timer negotiated by the adjacency protocol loss of synchronisation may be declared. The timer is specified in units of 100ms."

::= { gsmpSessionEntry 4 }

gsmpSessionPartitionId OBJECT-TYPE  
SYNTAX GsmpPartitionIdType  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
    "The Partition Id for the specific switch partition that  
    this session is concerned with."  
 ::= { gsmpSessionEntry 5 }

gsmpSessionAdjacencyCount OBJECT-TYPE  
SYNTAX Unsigned32(1..255)  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
    "This object specifies the current number of adjacencies  
    that are established with controllers and the switch  
    partition that is used for this session. The value  
    includes this session."  
 ::= { gsmpSessionEntry 6 }

gsmpSessionFarSideName OBJECT-TYPE  
SYNTAX GsmpNameType  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
    "The name of the far side as advertised in the adjacency  
    message."  
 ::= { gsmpSessionEntry 7 }

gsmpSessionFarSidePort OBJECT-TYPE  
SYNTAX Unsigned32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
    "The local port number of the link across which the  
    message is being sent."  
REFERENCE  
    "General Switch Management Protocol V3: Section 3.1.2"  
 ::= { gsmpSessionEntry 8 }

gsmpSessionFarSideInstance OBJECT-TYPE  
SYNTAX Unsigned32(1..16777215)  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
    "The instance number used for the link during this  
    session. The Instance number is a 24-bit number  
    that should be guaranteed to be unique within

the recent past and to change when the link or node comes back up after going down. Zero is not a valid instance number."  
 ::= { gsmpSessionEntry 9 }

gsmpSessionLastFailureCode OBJECT-TYPE  
 SYNTAX Unsigned32(0..255)  
 MAX-ACCESS read-only  
 STATUS current  
 DESCRIPTION  
 "This is the last failure code that was received over this session. If no failure code have been received, the value is zero."  
 ::= { gsmpSessionEntry 10 }

gsmpSessionDiscontinuityTime OBJECT-TYPE  
 SYNTAX TimeStamp  
 MAX-ACCESS read-only  
 STATUS current  
 DESCRIPTION  
 "The value of sysUpTime on the most recent occasion at which one or more of this session's counters suffered a discontinuity. If no such discontinuities have occurred since then, this object contains the same timestamp as gsmpSessionStartUptime ."  
 ::= { gsmpSessionEntry 11 }

gsmpSessionStartUptime OBJECT-TYPE  
 SYNTAX TimeStamp  
 MAX-ACCESS read-only  
 STATUS current  
 DESCRIPTION  
 " The value of sysUpTime when the session came to established state."  
 ::= { gsmpSessionEntry 12 }

gsmpSessionStatSentMessages OBJECT-TYPE  
 SYNTAX ZeroBasedCounter32  
 MAX-ACCESS read-only  
 STATUS current  
 DESCRIPTION  
 "The number of messages that have been sent in this session. All GSMP messages pertaining to this session after the session came to established state SHALL be counted, also including adjacency protocol messages and failure response messages.  
 When the counter suffers any discontinuity, then the gsmpSessionDiscontinuityTime object indicates when it

```
        happened."
 ::= { gsmpSessionEntry 13 }

gsmpSessionStatFailureInds OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of messages that have been sent with a
        failure indication in this session. Warning messages
        SHALL NOT be counted.
        When the counter suffers any discontinuity, then
        the gsmpSessionDiscontinuityTime object indicates when it
        happened."
    REFERENCE
        "General Switch Management Protocol V3: Section 12.1"
 ::= { gsmpSessionEntry 14 }

gsmpSessionStatReceivedMessages OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of messages that have been received in
        this session. All legal GSMP messages pertaining to this
        session after the session came to established state SHALL
        be counted, also including adjacency protocol messages
        and failure response messages.
        When the counter suffers any discontinuity, then
        the gsmpSessionDiscontinuityTime object indicates when it
        happened."
 ::= { gsmpSessionEntry 15 }

gsmpSessionStatReceivedFailures OBJECT-TYPE
    SYNTAX      ZeroBasedCounter32
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "The number of messages that have been received in
        this session with a failure indication. Warning messages
        SHALL NOT be counted.
        When the counter suffers any discontinuity, then
        the gsmpSessionDiscontinuityTime object indicates when it
        happened."
    REFERENCE
        "General Switch Management Protocol V3: Section 12.1"
 ::= { gsmpSessionEntry 16 }
```

**gsmpSessionStatPortUpEvents OBJECT-TYPE**

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of Port Up events that have been sent or received on this session.  
When the counter suffers any discontinuity, then the gsmpSessionDiscontinuityTime object indicates when it happened."

## REFERENCE

"General Switch Management Protocol V3: Section 9.1"  
::= { gsmpSessionEntry 17 }

**gsmpSessionStatPortDownEvents OBJECT-TYPE**

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of Port Down events that have been sent or received on this session.  
When the counter suffers any discontinuity, then the gsmpSessionDiscontinuityTime object indicates when it happened."

## REFERENCE

"General Switch Management Protocol V3: Section 9.2"  
::= { gsmpSessionEntry 18 }

**gsmpSessionStatInvLabelEvents OBJECT-TYPE**

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of Invalid label events that have been sent or received on this session.  
When the counter suffers any discontinuity, then the gsmpSessionDiscontinuityTime object indicates when it happened."

## REFERENCE

"General Switch Management Protocol V3: Section 9.3"  
::= { gsmpSessionEntry 19 }

**gsmpSessionStatNewPortEvents OBJECT-TYPE**

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of New Port events that have been sent or

received on this session.  
 When the counter suffers any discontinuity, then  
 the gsmpSessionDiscontinuityTime object indicates when it  
 happened."

## REFERENCE

"General Switch Management Protocol V3: Section 9.4"  
 ::= { gsmpSessionEntry 20 }

## gsmpSessionStatDeadPortEvents OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of Dead Port events that have been sent or  
 received on this session.  
 When the counter suffers any discontinuity, then  
 the gsmpSessionDiscontinuityTime object indicates when it  
 happened."

## REFERENCE

"General Switch Management Protocol V3: Section 9.5"  
 ::= { gsmpSessionEntry 21 }

## gsmpSessionStatAdjUpdateEvents OBJECT-TYPE

SYNTAX ZeroBasedCounter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of Adjacency Update events that have been sent  
 or received on this session.  
 When the counter suffers any discontinuity, then  
 the gsmpSessionDiscontinuityTime object indicates when it  
 happened."

## REFERENCE

"General Switch Management Protocol V3: Section 9.6"  
 ::= { gsmpSessionEntry 22 }

```
-- *****
-- GSMP Notifications
-- *****
```

```
--
-- Notification objects
--
```

## gsmpEventPort OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS accessible-for-notify

```

STATUS          current
DESCRIPTION
    "This object specifies the Port Number that is
    carried in this event."
 ::= { gsmpNotificationsObjects 1 }

gsmpEventPortSessionNumber OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   accessible-for-notify
    STATUS      current
    DESCRIPTION
        "This object specifies the Port Session Number that is
        carried in this event."
    ::= { gsmpNotificationsObjects 2 }

gsmpEventSequenceNumber OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS   accessible-for-notify
    STATUS      current
    DESCRIPTION
        "This object specifies the Event Sequence Number that is
        carried in this event."
    ::= { gsmpNotificationsObjects 3 }

gsmpEventLabel OBJECT-TYPE
    SYNTAX      GsmplabelType
    MAX-ACCESS   accessible-for-notify
    STATUS      current
    DESCRIPTION
        "This object specifies the Label that is
        carried in this event."
    ::= { gsmpNotificationsObjects 4 }

--
-- Notifications
--

gsmpSessionDown NOTIFICATION-TYPE
    OBJECTS {
        gsmpSessionStartUptime,
        gsmpSessionStatSentMessages,
        gsmpSessionStatFailureInds,
        gsmpSessionStatReceivedMessages,
        gsmpSessionStatReceivedFailures,
        gsmpSessionStatPortUpEvents,
        gsmpSessionStatPortDownEvents,
        gsmpSessionStatInvLabelEvents,

```



```

        gsmpSessionStatNewPortEvents,
        gsmpSessionStatDeadPortEvents,
        gsmpSessionStatAdjUpdateEvents
    }

```

STATUS current

DESCRIPTION

"When it has been enabled, this notification is generated whenever a session is taken down, regardless of whether the session went down normally or not. Its purpose is to allow a management application (primarily an accounting application) that is monitoring the session statistics to receive the final values of these counters, so that the application can properly account for the amounts the counters were incremented since the last time the application polled them. The gsmpSessionStartUptime object provides the total amount of time that the session was active.

This notification is not a substitute for polling the session statistic counts. In particular, the count values reported in this notification cannot be assumed to be the complete totals for the life of the session, since they may have wrapped while the session was up.

The session to which this notification applies is identified by the gsmpSessionThisSideId and gsmpSessionFarSideId which could be inferred from the Object Identifiers of the objects contained in the notification.

An instance of this notification will contain exactly one instance of each of its objects, and these objects will all belong to the same conceptual row of the gsmpSessionTable."

```
 ::= { gsmpNotifications 1 }
```

gsmpSessionUp NOTIFICATION-TYPE

```

    OBJECTS {
        gsmpSessionFarSideInstance
    }

```

STATUS current

DESCRIPTION

"When it has been enabled, this notification is generated when new session is established.

The new session is identified by the gsmpSessionThisSideId and gsmpSessionFarSideId which could be inferred from the Object Identifier of the gsmpSessionFarSideInstance object

```
        contained in the notification."
 ::= { gsmpNotifications 2 }

gsmpSentFailureInd NOTIFICATION-TYPE
    OBJECTS {
        gsmpSessionLastFailureCode,
        gsmpSessionStatFailureInds
    }
    STATUS current
    DESCRIPTION
        "When it has been enabled, this notification is
        generated when a message with a failure indication was
        sent.

        The notification indicates a change in the value of
        gsmpSessionStatFailureInds. The
        gsmpSessionLastFailureCode contains the failure
        reason.

        The session to which this notification
        applies is identified by the gsmpSessionThisSideId and
        gsmpSessionFarSideId which could be inferred from the
        Object Identifiers of the objects contained in the
        notification."
 ::= { gsmpNotifications 3 }

gsmpReceivedFailureInd NOTIFICATION-TYPE
    OBJECTS {
        gsmpSessionLastFailureCode,
        gsmpSessionStatReceivedFailures
    }
    STATUS current
    DESCRIPTION
        "When it has been enabled, this notification is
        generate when a message with a failure indication
        is received.

        The notification indicates a change in the value of
        gsmpSessionStatReceivedFailures. The
        gsmpSessionLastFailureCode contains the failure
        reason.

        The session to which this notification
        applies is identified by the gsmpSessionThisSideId and
        gsmpSessionFarSideId which could be inferred from the
        Object Identifiers of the objects contained in the
        notification."
 ::= { gsmpNotifications 4 }
```

```
gsmpPortUpEvent NOTIFICATION-TYPE
  OBJECTS {
    gsmpSessionStatPortUpEvents,
    gsmpEventPort,
    gsmpEventPortSessionNumber,
    gsmpEventSequenceNumber
  }
  STATUS current
  DESCRIPTION
    "When it has been enabled, this notification is
    generated when a Port Up Event occurs.

    The notification indicates a change in the value of
    gsmpSessionStatPortUpEvents.

    The session to which this notification
    applies is identified by the gsmpSessionThisSideId and
    gsmpSessionFarSideId which could be inferred from the
    Object Identifier of the gsmpSessionStatPortUpEvents
    object contained in the notification."
 ::= { gsmpNotifications 5 }

gsmpPortDownEvent NOTIFICATION-TYPE
  OBJECTS {
    gsmpSessionStatPortDownEvents,
    gsmpEventPort,
    gsmpEventPortSessionNumber,
    gsmpEventSequenceNumber
  }
  STATUS current
  DESCRIPTION
    "When it has been enabled, this notification is
    generated when a Port Down Event occurs.

    The notification indicates a change in the value of
    gsmpSessionStatPortDownEvents.

    The session to which this notification
    applies is identified by the gsmpSessionThisSideId and
    gsmpSessionFarSideId which could be inferred from the
    Object Identifier of the gsmpSessionStatPortDownEvents
    object contained in the notification."
 ::= { gsmpNotifications 6 }

gsmpInvalidLabelEvent NOTIFICATION-TYPE
  OBJECTS {
    gsmpSessionStatInvLabelEvents,
    gsmpEventPort,
```

```
        gsmpEventLabel,
        gsmpEventSequenceNumber
    }
STATUS current
DESCRIPTION
    "When it has been enabled, this notification is
    generated when an Invalid Label Event occurs.

    The notification indicates a change in the value of
    gsmpSessionStatInvLabelEvents.

    The session to which this notification
    applies is identified by the gsmpSessionThisSideId and
    gsmpSessionFarSideId which could be inferred from the
    Object Identifier of the gsmpSessionStatInvLabelEvents
    object contained in the notification."
 ::= { gsmpNotifications 7 }

gsmpNewPortEvent NOTIFICATION-TYPE
    OBJECTS {
        gsmpSessionStatNewPortEvents,
        gsmpEventPort,
        gsmpEventPortSessionNumber,
        gsmpEventSequenceNumber
    }
STATUS current
DESCRIPTION
    "When it has been enabled, this notification is
    generated when a New Port Event occurs.

    The notification indicates a change in the value of
    gsmpSessionStatNewPortEvents.

    The session to which this notification
    applies is identified by the gsmpSessionThisSideId and
    gsmpSessionFarSideId which could be inferred from the
    Object Identifier of the gsmpSessionStatNewPortEvents
    object contained in the notification."
 ::= { gsmpNotifications 8 }

gsmpDeadPortEvent NOTIFICATION-TYPE
    OBJECTS {
        gsmpSessionStatDeadPortEvents,
        gsmpEventPort,
        gsmpEventPortSessionNumber,
        gsmpEventSequenceNumber
    }
STATUS current
```

## DESCRIPTION

"When it has been enabled, this notification is generated when a Dead Port Event occurs.

The notification indicates a change in the value of gsmpSessionStatDeadPortEvents.

The session to which this notification applies is identified by the gsmpSessionThisSideId and gsmpSessionFarSideId which could be inferred from the Object Identifier of the gsmpSessionStatDeadPortEvents object contained in the notification."

::= { gsmpNotifications 9 }

## gsmpAdjacencyUpdateEvent NOTIFICATION-TYPE

## OBJECTS {

gsmpSessionAdjacencyCount,  
gsmpSessionStatAdjUpdateEvents,  
gsmpEventSequenceNumber  
}

STATUS current

## DESCRIPTION

"When it has been enabled, this notification is generated when an Adjacency Update Event occurs.

The gsmpSessionAdjacencyCount contains the new value of the number of adjacencies that are established with controllers and the switch partition that is used for this session.

The notification indicates a change in the value of gsmpSessionStatAdjUpdateEvents.

The session to which this notification applies is identified by the gsmpSessionThisSideId and gsmpSessionFarSideId which could be inferred from the Object Identifier of the gsmpSessionAdjacencyCount or the gsmpSessionStatAdjUpdateEvents object contained in the notification."

::= { gsmpNotifications 10 }

```

--*****
-- GSMP Compliance
--*****

gsmpGroups          OBJECT IDENTIFIER ::= { gsmpConformance 1 }
gsmpCompliances     OBJECT IDENTIFIER ::= { gsmpConformance 2 }

gsmpModuleCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "The compliance statement for agents that support
        the GSMP MIB."
    MODULE -- this module
    MANDATORY-GROUPS { gsmpGeneralGroup
                        }
    GROUP gsmpControllerGroup
    DESCRIPTION
        "This group is mandatory for all Switch
        Controllers"

    GROUP gsmpSwitchGroup
    DESCRIPTION
        "This group is mandatory for all Switches"

    GROUP gsmpAtmEncapGroup
    DESCRIPTION
        "This group must be supported if ATM is used for GSMP
        encapsulation. "

    GROUP gsmpTcpIpEncapGroup
    DESCRIPTION
        "This group must be supported if TCP/IP is used for GSMP
        encapsulation. "

    OBJECT gsmpTcpIpEncapAddressType
    SYNTAX InetAddressType { unknown(0), ipv4(1), ipv6(2),
                            ipv4z(3), ipv6z(4) }
    DESCRIPTION
        "An implementation is only required to support
        'unknown(0)', and IPv4 addresses. Supporting addresses with
        zone index or IPv6 addresses are optional. Defining
        Internet addresses by using DNS domain names are not
        allowed."

    OBJECT gsmpTcpIpEncapAddress
    SYNTAX InetAddress (SIZE(0|4|8|16|20))
    DESCRIPTION
        "An implementation is only required to support

```

IPv4 addresses. Supporting addresses with zone index or IPv6 addresses are optional."

GROUP gsmpNotificationObjectsGroup  
DESCRIPTION

"This group must be supported if notifications are supported. "

GROUP gsmpNotificationsGroup  
DESCRIPTION

"This group must be supported if notifications are supported. "

::= { gsmpCompliances 1 }

-- units of conformance

gsmpGeneralGroup OBJECT-GROUP

OBJECTS {  
gsmpSessionVersion,  
gsmpSessionTimer,  
gsmpSessionPartitionId,  
gsmpSessionAdjacencyCount,  
gsmpSessionFarSideName,  
gsmpSessionFarSidePort,  
gsmpSessionFarSideInstance,  
gsmpSessionLastFailureCode,  
gsmpSessionDiscontinuityTime,  
gsmpSessionStartUptime,  
gsmpSessionStatSentMessages,  
gsmpSessionStatFailureInds,  
gsmpSessionStatReceivedMessages,  
gsmpSessionStatReceivedFailures,  
gsmpSessionStatPortUpEvents,  
gsmpSessionStatPortDownEvents,  
gsmpSessionStatInvLabelEvents,  
gsmpSessionStatNewPortEvents,  
gsmpSessionStatDeadPortEvents,  
gsmpSessionStatAdjUpdateEvents  
}

STATUS current

DESCRIPTION

"Objects that apply to all GSMP implementations."

::= { gsmpGroups 1 }

gsmpControllerGroup OBJECT-GROUP

OBJECTS {  
gsmpControllerMaxVersion,

```
gsmpControllerTimer,
gsmpControllerPort,
gsmpControllerInstance,
gsmpControllerPartitionType,
gsmpControllerPartitionId,
gsmpControllerDoResync,
gsmpControllerNotificationMap,
gsmpControllerSessionState,
gsmpControllerStorageType,
gsmpControllerRowStatus
}
STATUS          current
DESCRIPTION
    "Objects that apply GSMP implementations of
    Switch Controllers."
::= { gsmpGroups 2 }
```

```
gsmpSwitchGroup OBJECT-GROUP
OBJECTS {
gsmpSwitchMaxVersion,
gsmpSwitchTimer,
gsmpSwitchName,
gsmpSwitchPort,
gsmpSwitchInstance,
gsmpSwitchPartitionType,
gsmpSwitchPartitionId,
gsmpSwitchNotificationMap,
gsmpSwitchSwitchType,
gsmpSwitchWindowSize,
gsmpSwitchSessionState,
gsmpSwitchStorageType,
gsmpSwitchRowStatus
}
STATUS          current
DESCRIPTION
    "Objects that apply GSMP implementations of
    Switches."
::= { gsmpGroups 3 }
```

```
gsmpAtmEncapGroup OBJECT-GROUP
OBJECTS {
gsmpAtmEncapIfIndex,
gsmpAtmEncapVpi,
gsmpAtmEncapVci,
gsmpAtmEncapStorageType,
gsmpAtmEncapRowStatus
}
STATUS          current
```



## DESCRIPTION

"Objects that apply to GSMP implementations that supports ATM for GSMP encapsulation."

::= { gsmpGroups 4 }

## gsmpTcpIpEncapGroup OBJECT-GROUP

OBJECTS {  
gsmpTcpIpEncapAddressType,  
gsmpTcpIpEncapAddress,  
gsmpTcpIpEncapPortNumber,  
gsmpTcpIpEncapStorageType,  
gsmpTcpIpEncapRowStatus  
}

STATUS current

## DESCRIPTION

"Objects that apply to GSMP implementations that supports TCP/IP for GSMP encapsulation."

::= { gsmpGroups 5 }

## gsmpNotificationObjectsGroup OBJECT-GROUP

OBJECTS {  
gsmpEventPort,  
gsmpEventPortSessionNumber,  
gsmpEventSequenceNumber,  
gsmpEventLabel  
}

STATUS current

## DESCRIPTION

"Objects that are contained in the notifications."

::= { gsmpGroups 6 }

## gsmpNotificationsGroup NOTIFICATION-GROUP

NOTIFICATIONS {  
gsmpSessionDown,  
gsmpSessionUp,  
gsmpSentFailureInd,  
gsmpReceivedFailureInd,  
gsmpPortUpEvent,  
gsmpPortDownEvent,  
gsmpInvalidLabelEvent,  
gsmpNewPortEvent,  
gsmpDeadPortEvent,  
gsmpAdjacencyUpdateEvent  
}

STATUS current

## DESCRIPTION

"The notifications which indicate specific changes in the value of objects gsmpSessionTable"

```
::= { gsmpGroups 7 }
```

END

## 5. Acknowledgments

The authors would like to thank Avri Doria and Kenneth Sundell for their contributions to this specification. Also thanks to David Partain, Michael MacFaden and Bert Wijnen who have contributed significantly with their SNMP expertise.

## 6. References

- [RFC1155] Rose, M. and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, RFC 1155, May 1990.
- [RFC1212] Rose, M. and K. McCloghrie, "Concise MIB Definitions", STD 16, RFC 1212, March 1991.
- [RFC1215] Rose, M., "A Convention for Defining Traps for use with the SNMP", RFC 1215, March 1991.
- [RFC1157] Case, J., Fedor, M., Schoffstall, M. and J. Davin, "Simple Network Management Protocol", STD 15, RFC 1157, May 1990.
- [RFC1901] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Introduction to Community-based SNMPv2", RFC 1901, January 1996.
- [RFC1905] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, January 1996.
- [RFC1906] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1906, January 1996.
- [RFC1987] Newman, P., Edwards, W., Hinden, R., Hoffman, E., Ching Liaw, F., Lyon, T. and Minshall, G., "Ipsilon's General Switch Management Protocol Specification," Version 1.1, RFC 1987, August 1996.
- [RFC2021] Waldbusser, S., "Remote Network Monitoring Management Information Base Version 2 using SMIV2", RFC 2021, January 1997.

- [RFC2026] Bradner, S., "The Internet Standards Process - Revision 3", BCP 9, RFC 2026, October 1996.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2397] Newman, P, Edwards, W., Hinden, R., Hoffman, E., Ching Liaw, F., Lyon, T. and Minshall, G., "Ipsilon's General Switch Management Protocol Specification," Version 2.0, RFC 2397, March 1998.
- [RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs.", BCP 26, RFC 2434, October 1998.
- [RFC2514] Noto, M., E. Spiegel, K. Tesink, "Definition of Textual Conventions and OBJECT-IDENTITIES for ATM Management", RFC 2514, February 1999.
- [RFC2570] Case, J., Mundy, R., Partain, D. and B. Stewart, "Introduction to Version 3 of the Internet-standard Network Management Framework", RFC 2570, April 1999.
- [RFC2571] Harrington, D., Presuhn, R. and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", RFC 2571, April 1999.
- [RFC2572] Case, J., Harrington D., Presuhn R. and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", RFC 2572, April 1999.
- [RFC2573] Levi, D., Meyer, P. and B. Stewart, "SNMP Applications", RFC 2573, April 1999.
- [RFC2574] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", RFC 2574, April 1999.
- [RFC2575] Wijnen, B., Presuhn, R. and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", RFC 2575, April 1999.
- [RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

- [RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Textual Conventions for SMIV2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Conformance Statements for SMIV2", STD 58, RFC 2580, April 1999.
- [RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB" RFC 2863, June 2000.
- [RFC3291] Daniele, M., Haberman, B., Routhier, S. and J., Schoenwaelder "Textual Conventions for Internet Network Addresses", RFC 3291, May 2002.
- [RFC3292] Doria, A., Hellstrand, F., Sundell, K. and T. Worster, "General Switch Management Protocol V3", RFC 3292, June 2002.
- [RFC3293] Worster, T., Doria, A. and J. Buerkle, "General Switch Management Protocol (GSMP) Packet Encapsulations for Asynchronous Transfer Mode (ATM), Ethernet and Transmission Control Protocol (TCP)", RFC 3293, June 2002.

## 7. Intellectual Property Rights

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

## 8. Security Considerations

Assuming that secure network management (such as SNMP v3) is implemented, the objects represented in this MIB do not pose a threat to the security of the network.

There are a number of management objects defined in this MIB that have a MAX-ACCESS clause of read-write and/or read-create. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations.

There are a number of managed objects in this MIB that may contain sensitive information. They are contained in the gsmpControllerTable and gsmpSwitchTable. It is thus important to control even GET access to these objects and possibly to even encrypt the values of these object when sending them over the network via SNMP. Not all versions of SNMP provide features for such a secure environment.

SNMPv1 by itself is not a secure environment. Even if the network itself is secure (for example by using IPSec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model RFC 2574 [RFC2574] and the View-based Access Control Model RFC 2575 [RFC2575] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, is properly configured to give access to the objects, only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

## 9. Authors' Addresses

Hans Sjostrand  
ipUnplugged  
P.O. Box 101 60  
S-121 28 Stockholm, Sweden

Phone: +46 8 725 5930  
EMail: hans@ipunplugged.com

Joachim Buerkle  
Nortel Networks Germany GmbH & Co. KG  
Hahnstrasse 37-39  
D-60528 Frankfurt am Main, Germany

Phone: +49 69 6697 3281  
EMail: joachim.buerkle@nortelnetworks.com

Balaji Srinivasan  
CPlane Inc.  
897 Kifer Road  
Sunnyvale, CA 94086

Phone: +1 408 789 4099  
EMail: balaji@cplane.com

## 10. Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

