Ipsilon's General Switch Management Protocol Specification
Version 1.1

Status of this Memo

   This memo provides information for the Internet community. This memo
   does not specify an Internet standard of any kind. Distribution of
   this memo is unlimited.

Abstract

   The General Switch Management Protocol (GSMP), is a general purpose
   protocol to control an ATM switch. GSMP allows a controller to
   establish and release connections across the switch; add and delete
   leaves on a point-to-multipoint connection; manage switch ports;
   request configuration information; and request statistics.

Table of Contents

1. Introduction

   The General Switch Management Protocol (GSMP), is a general purpose
   protocol to control an ATM switch. GSMP allows a controller to
   establish and release connections across the switch; add and delete
   leaves on a point-to-multipoint connection; manage switch ports;
   request configuration information; and request statistics. It also
   allows the switch to inform the controller of asynchronous events
   such as a link going down. GSMP runs across an ATM link connecting
   the controller to the switch, on a control connection (virtual
   channel) established at initialization. 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.

   A switch is assumed to contain multiple "ports". Each port is a
   combination of one "input port" and one "output port". Some GSMP
   requests refer to the port as a whole whereas other requests are
   specific to the input port or the output port. ATM cells arrive at
   the switch from an external communication link on incoming virtual
   channels at an input port. ATM cells depart from the switch to an
   external communication link on outgoing virtual channels from an
   output port. Virtual channels on a port or link are referenced by
   their virtual path and virtual channel identifiers (VPI/VCI). A
   virtual channel connection across a switch is formed by connecting an
   incoming virtual channel to one or more outgoing virtual channels.
   Virtual channel connections are referenced by the input port on which
   they arrive and the virtual path and virtual channel identifiers
   (VPI/VCI) of their incoming virtual channel.

   In general a virtual channel is established with a certain quality of
   service (QOS). Unfortunately this is an ill defined and changing
   concept as new ideas make their way into hardware. For this version
   of the GSMP protocol it is assumed that each virtual channel
   connection may be assigned a priority when it is established. It may
   be assumed that for virtual channel connections that share the same
   output port, an ATM cell on a connection with a higher priority is
   much more likely to exit the switch before an ATM cell on a
   connection with a lower priority if they are both in the switch at
   the same time. The number of priorities that each port of the switch
   supports may be obtained from the port configuration message.

   Switch ports are described by a 32 bit port number. The switch
   assigns port numbers and it may typically choose to structure the 32
   bits into sub-fields that have meaning to the physical structure of
   the switch (e.g. shelf, slot, port). In general, a port in the same
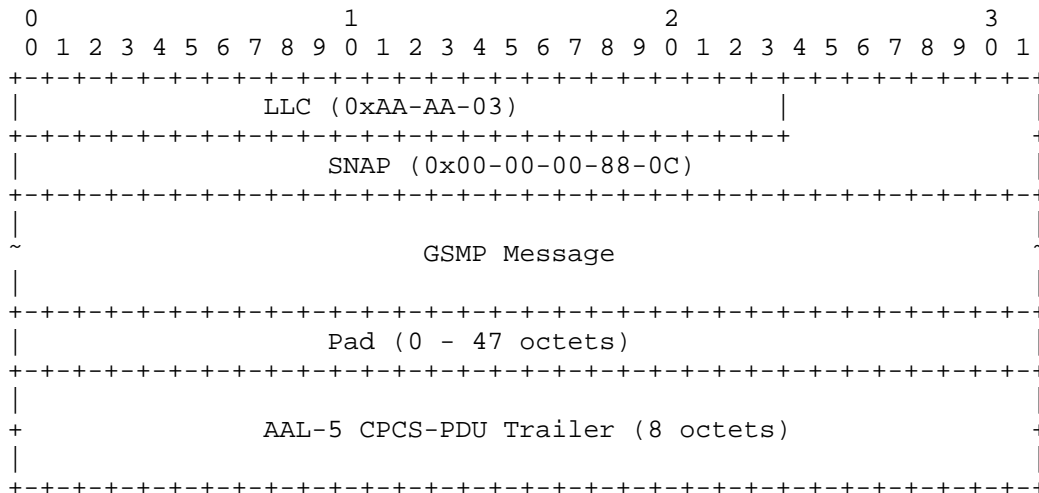   physical location on the switch will always have the same port

number, even across power cycles. The internal structure of the port
number is opaque to the GSMP protocol. However, by looking up the
product identity in a database, network management tools may discover
the partitioning of the port number and the physical meaning of the
sub-fields.

Each switch port also maintains a port session number assigned by the
switch. A connection management message or a port management message
with an incorrect port session number must be rejected. This allows
the controller to detect a link failure and to keep state
synchronized. The port session number of a port remains unchanged
while the port is continuously in the available state and the link
status is continuously up. When a port returns to the available state
after it has been unavailable or in any of the loopback states, or
when the line status returns to the up state after it has been down
or in test, or after a power cycle, its port session number will have
changed. Port session numbers should be assigned using some form of
random number.

GSMP also contains an adjacency protocol. The adjacency protocol is
used to synchronize state across the link, to discover the identity
of the entity at the other end of a link, and to detect when it
changes.

2. GSMP Packet Format

GSMP packets are variable length and are encapsulated directly in an
AAL-5 CPCS-PDU [I.363] with an LLC/SNAP header as illustrated:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    LLC (0xAA-AA-03)           |               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+               +
|                  SNAP (0x00-00-00-88-0C)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                        GSMP Message                           ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Pad (0 - 47 octets)                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+               AAL-5 CPCS-PDU Trailer (8 octets)               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

(The convention in the documentation of Internet Protocols [rfc1700]
is to express numbers in decimal and to picture data in "big-endian"
order. That is, fields are described left to right, with the most
significant octet on the left and the least significant octet on the
right. Whenever a diagram shows a group of octets, the order of
transmission of those octets is the normal order in which they are
read in English. Whenever an octet represents a numeric quantity the
left most bit in the diagram is the high order or most significant
bit. That is, the bit labeled 0 is the most significant bit.
Similarly, whenever a multi-octet field represents a numeric quantity
the left most bit of the whole field is the most significant bit.
When a multi-octet quantity is transmitted, the most significant
octet is transmitted first. This is the same coding convention as is
used in the ATM layer [I.361] and AAL-5 [I.363].)

The LLC/SNAP header contains the octets: 0xAA 0xAA 0x03 0x00 0x00
0x00 0x88 0x0C.

The maximum transmission unit (MTU) of the GSMP message is 1500
octets.

The default virtual channel for LLC/SNAP encapsulated messages is:

    VPI = 0
    VCI = 15.

GSMP is a master-slave protocol. The controller issues request
messages to the switch. Each request message indicates whether a
response is required from the switch and contains a transaction
identifier to enable the response to be associated with the request.
The switch replies with a response message indicating either a
successful result or a failure. There are four classes of GSMP
request-response message: Connection Management, Port Management,
Statistics, and Configuration. The switch may also generate
asynchronous Event messages to inform the controller of asynchronous
events. Event messages are not acknowledged by the controller. There
is also an adjacency protocol message used to establish
synchronization across the link and maintain a handshake.

For the request-response messages each message type has a format for
the request message and a format for the success response. Unless
otherwise specified a failure response message is identical to the
request message that caused the failure, with the Code field
indicating the nature of the failure. Event messages have only a
single format defined as they are not acknowledged by the controller.

Except for the adjacency protocol message, no GSMP messages may be
sent across the link until the adjacency protocol has achieved

synchronization, and all GSMP messages received on a link that does
not currently have state synchronization must be discarded.

All GSMP messages, except the adjacency protocol message, have the
following format:
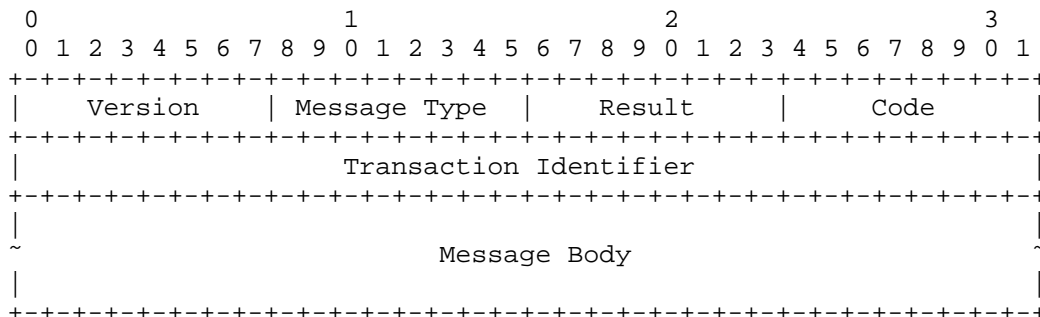
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Version     | Message Type  |    Result     |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Transaction Identifier                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                         Message Body                          ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Version

        The GSMP protocol version number, currently Version = 1. It
        should be set by the sender of the message to the GSMP
        protocol version that the sender is currently running.

Message Type

        The GSMP message type. GSMP messages fall into five
        classes: Connection Management, Port Management,
        Statistics, Configuration, and Events. Each class, except
        for port management, has a number of different message
        types. In addition, one Message Type is allocated to the
        adjacency protocol.

Result

        Field in a connection management request message or a port
        management request message, is used to indicate whether a
        response is required to the request message if the outcome
        is successful. A value of "NoSuccessAck" indicates that the
        request message does not expect a response if the outcome
        is successful, and a value of "AckAll" indicates that a
        response is expected if the outcome is successful. In both
        cases a failure response will be generated if the request
        fails. This facility reduces the traffic in the case where
        the controller is simply checking that the state in the
        switch is correct. For all other request messages a value
        of "NoSuccessAck" in the request message is ignored and the
        request message is handled as if the field were set to
        "AckAll". In a response message the result field can have
        two values: "Success" and "Failure".

The encoding of the result field is:

            NoSuccessAck: Result = 1
            AckAll: Result = 2
            Success: Result = 3
            Failure: Result = 4.

The Result field is not used in an adjacency protocol
message and should be set to zero by the sender and ignored
by the receiver.

Code
         Field gives further information concerning the result in a
         response message. It is mostly used to pass an error code
         in a failure response but can also be used to give further
         information in a success response message or an event
         message. In a request message the code field is not used
         and is set to zero. In an adjacency protocol message the
         Code field is used to determine the function of the
         message.

Transaction Identifier
         Used to associate a request message with its response
         message. For request messages the controller may select any
         transaction identifier. For response messages the
         transaction identifier is set to the value of the
         transaction identifier from the message to which it is a
         response. For event messages the transaction identifier
         should be set to zero. In the adjacency protocol the
         Transaction Identifier is not used. This field is not
         present in the adjacency protocol message.

3. Connection Management Messages

   Connection management messages are used by the controller to
   establish, delete, modify and verify connections across the switch.
   The Add Branch, Delete Branch, Delete Tree, Verify Tree, and Delete
   All connection management messages have the following format for both
   request and response messages:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    Version    |  Message Type  |   Result     |     Code     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                     Transaction Identifier                   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                     Port Session Number                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         Input Port                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | zero  |     Input VPI         |          Input VCI           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                        Output Port                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | zero  |     Output VPI        |          Output VCI          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |      Number of Branches       |   Reserved    |   Priority   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
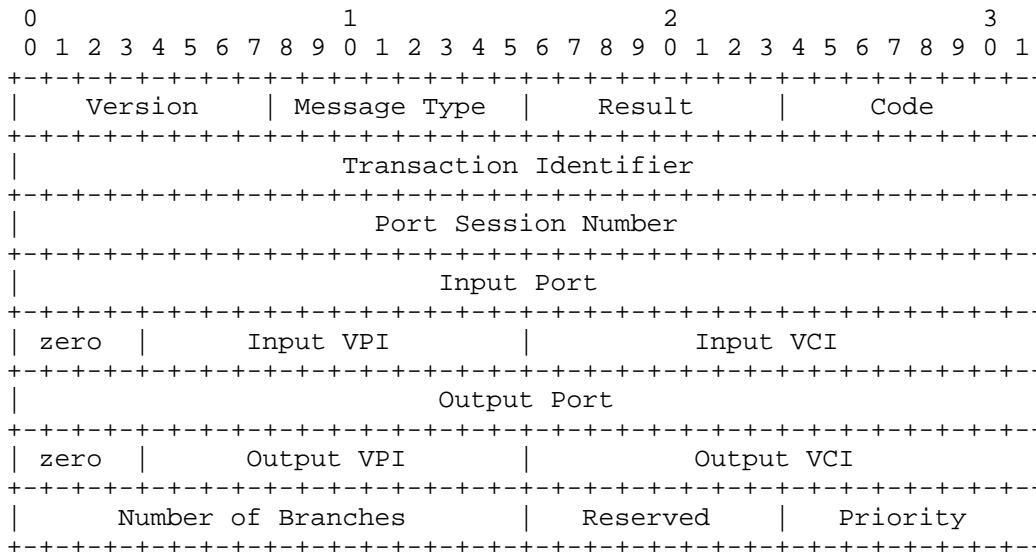
Port Session Number
        Field gives the session number of the input port. Each
        switch port maintains a Port Session Number assigned by the
        switch. The port session number of a port remains unchanged
        while the port is continuously in the Available state and
        the link status is continuously Up. When a port returns to
        the Available state after it has been Unavailable or in any
        of the Loopback states, or when the line status returns to
        the Up state after it has been Down or in Test, or after a
        power cycle, a new Port Session Number must be generated.
        Port session numbers should be assigned using some form of
        random number. The switch must reject any connection
        management request message that has an invalid Port Session
        Number for the port specified in the Input Port field by
        returning a failure response message with the Code field
        indicating, "Invalid port session number." The current port
        session number may be obtained using a configuration
        message.

   Input Port
        Indicates a switch input port. Switch ports are referenced
        by a 32 bit value assigned by the switch.

   Input VPI
        Identifies an ATM virtual path arriving at the switch input
        port indicated by the Input Port field.

Input VCI
        Identifies an ATM virtual channel arriving on the virtual
        path indicated by the Input VPI field at the switch input
        port indicated by the Input Port field.

Output Port
        Indicates a switch output port. Switch ports are
        referenced by a 32 bit value assigned by the switch.

Output VPI
        Identifies an outgoing virtual path departing from the
        switch output port indicated in the Output Port field.

Output VCI
        Identifies an outgoing virtual channel departing on the
        virtual path indicated by the Output VPI field from the
        switch output port indicated in the Output Port field.

Number of Branches
        Gives the number of output branches on a virtual channel
        connection. (A unicast connection will have one branch, a
        multicast connection will have two or more branches.) This
        field is only used in the Verify Tree message. In all
        other connection management messages this field should be
        set to zero by the sender and ignored by the receiver.

Reserved
        This field is not used. It is set to zero by the sender and
        ignored by the receiver.

Priority
        Gives the priority of the connection. The highest priority
        is numbered zero and the lowest priority is numbered "Q-1"
        where "Q" is the number of priorities that the output port
        can support. The ability to offer different qualities of
        service to different connections based upon their priority
        is assumed to be a property of the output port of the
        switch. It is assumed that for virtual channel connections
        that share the same output port, an ATM cell on a
        connection with a higher priority is much more likely to
        exit the switch before an ATM cell on a connection with a
        lower priority if they are both in the switch at the same
        time. The number of priorities that each output port can
        support is given in the Port Configuration message. If a
        connection request is received with a value in the priority
        field that the switch cannot support, the switch will
        assign the closest priority that it is capable of
        supporting. This field is only used in the Add Branch and

Move Branch messages. In all other connection management
messages this field should be set to zero by the sender and
ignored by the receiver.

If the result field of the request message is "AckAll" the switch
must reply to all connection management request messages with a
success response message or a failure response message. If the
result field of the request message is "NoSuccessAck" the switch must
only reply in the case of a failure.

A success response message must not be sent until the operation has
been successfully completed. For connection management messages the
success response message is a copy of the request message returned
with a Result field indicating success. The Code field is not used in
a connection management success response message and should be set to
zero. The failure response message is a copy of the request message
returned with a Result field indicating failure. The Code field is
used to pass the Failure Code in a connection management failure
response message. If the switch issues a failure response the
connection state within the switch must not be modified by the
request message that resulted in the failure.

No distinction is made between unicast connections and multicast
connections. The first Add Branch message for a particular Input
Port, Input VPI, and Input VCI will establish a unicast connection.
The second Add Branch message with the same Input Port, Input VPI,
and Input VCI fields will convert the connection to a multicast
connection with two branches. Subsequent Add Branch messages with the
same Input Port, Input VPI, and Input VCI fields will add further
branches to the multicast connection. Use of the Delete Branch
message on a multicast connection with two branches will result in a
unicast connection. Use of the Delete Branch message on a unicast
connection will delete the unicast connection. There is no concept of
a connection with zero output branches. All connections are
unidirectional, one input virtual channel to one or more output
virtual channels.

The connection management messages may be issued regardless of the
Port Status of the switch port. Connections may be established or
deleted when a switch port is in the Available, Unavailable, or any
of the Loopback states. However, all connection state on an input
port will be deleted when the port returns to the Available state
from any other state, i.e. when a Port Management message is received
for that port with the Function field indicating either Bring Up, or
Reset Input Port.

3.1 Add Branch Message

   The Add Branch message is a connection management message used to
   establish a virtual channel connection or to add an additional branch
   to an existing virtual channel connection. It may also be used to
   check the connection state stored in the switch. The connection is
   specified by the Input Port, Input VPI, and Input VCI fields. The
   output branch is specified by the Output Port, Output VPI, and Output
   VCI fields. The priority of the connection is specified by the
   Priority field. The Add Branch message is:

      Message Type = 16

   If the virtual channel connection specified by the Input Port, Input
   VPI, and Input VCI fields does not already exist, it must be
   established with the single output branch specified in the request
   message. The output branch should have the priority specified by the
   Priority field. If the Result field of the request message is
   "AckAll" a success response message must be sent upon successful
   establishment of the specified branch. The success response message
   must not be sent until the Add Branch operation has been completed.

   If the virtual channel connection specified by the Input Port, Input
   VPI, and Input VCI fields already exists, but the specified output
   branch does not, the new output branch must be added. The new output
   branch should have the priority specified by the Priority field. If
   the Result field of the request message is "AckAll" a success
   response message must be sent upon successful establishment of the
   specified branch. The success response message must not be sent until
   the Add Branch operation has been completed.

   If the virtual channel connection specified by the Input Port, Input
   VPI, and Input VCI fields already exists and the specified output
   branch also already exists, the priority of the connection, if
   different from the request message, should be changed to that in the
   request message. A success response message must be sent if the
   Result field of the request message is "AckAll". This allows the
   controller to periodically reassert the state of a connection or to
   change its priority. If the result field of the request message is
   "NoSuccessAck" a success response message should not be returned.
   This may be used to reduce the traffic on the control link for
   messages that are reasserting previously established state. For
   messages that are reasserting previously established state, the
   switch must always check that this state is correctly established in
   the switch hardware (i.e. the actual connection tables used to
   forward cells).

The behavior is undefined if the output virtual channel specified by
the Output Port, Output VPI, and Output VCI fields is already in use
by any connection other than that specified by the Input Port, Input
VPI, and Input VCI fields.

A failure response must be returned if the switch is unable to
establish the specified branch or if there is an error in any of the
fields of the request message. If a failure message is returned the
state of the switch must not have been modified by the request
message.

It should be noted that different switches support multicast in
different ways. There will be a limit to the total number of
multicast connections any switch can support, and possibly a limit on
the maximum number of branches that a multicast connection may
specify. Some switches also impose a limit on the number of
different VPI/VCI values that may be assigned to the output branches
of a multicast connection. Many switches are incapable of supporting
more than a single branch of any particular multicast connection on
the same output port. Specific failure codes are defined for some of
these conditions. If a switch sends a failure response to an Add
Branch message it must choose the most specific failure code.

3.2 Delete Branch Message

The Delete Branch message is a connection management message used to
delete a single branch of a virtual channel connection, or in the
case of the last branch, to delete the connection. The virtual
channel connection is specified by the Input Port, Input VPI, and
Input VCI fields. The specific branch is indicated by the Output
Port, Output VPI, and Output VCI fields. The Delete Branch message
is:

    Message Type = 17

If the Result field of the request message is "AckAll" a success
response message must be sent upon successful deletion of the
specified branch. The success response message must not be sent until
the delete branch operation has been completed and if possible, not
until all data on that branch, queued for transmission, has been
transmitted. A failure message indicating, "The specified connection
does not exist," must be sent if the connection specified by the
Input Port, Input VPI, and Input VCI fields does not exist. A failure
message indicating, "The specified branch does not exist," must be
sent if the connection specified by the Input Port, Input VPI, and
Input VCI fields exists but the branch specified by the Output Port,
Output VPI, and Output VCI fields does not exist.

3.3 Delete Tree Message

   The Delete Tree message is a connection management message used to
   delete an entire virtual channel connection. All remaining branches
   of the connection are deleted. The virtual channel connection is
   specified by the Input Port, Input VPI, and Input VCI fields. The
   Output Port, Output VPI, and Output VCI fields are not used in this
   message and their contents should be set to zero by the sender and
   ignored by the receiver. The Delete Tree message is:

      Message Type = 18

   If the Result field of the request message is "AckAll" a success
   response message must be sent upon successful deletion of the
   specified connection. The success message must not be sent until the
   delete operation has been completed and if possible, not until all
   data on the connection, queued for transmission, has been
   transmitted. A failure message indicating, "The specified connection
   does not exist," must be sent if the connection specified by the
   Input Port, Input VPI, and Input VCI fields does not exist.

3.4 Verify Tree Message

   The Verify Tree message is a connection management message used to
   verify the number of branches on a virtual channel connection. The
   virtual channel connection is specified by the Input Port, Input VPI,
   and Input VCI fields. The Output Port, Output VPI, and Output VCI
   fields are not used in this message and their contents should be set
   to zero by the sender and ignored by the receiver. The number of
   branches that the sender believes that this virtual channel
   connection should contain is given by the Number of Branches field.
   The Verify Tree message is:

      Message Type = 19

   If the Result field of the request message is "AckAll" a success
   response message must be sent if the receiver agrees that the actual
   number of branches of the specified virtual channel connection
   matches the number contained in the Number of Branches field of the
   request message. The failure response message, with the code field
   set to "Failure specific to the particular message type," must be
   sent if the actual number of branches of the specified virtual
   channel connection does not match the number contained in the Number
   of Branches field of the request message. In this failure response
   message the Number of Branches field must be changed to contain the
   actual number of branches of the specified virtual channel
   connection. A failure response message with the code field set to a
   different value must be used to indicate some other failure such as,

"The specified connection does not exist." In this case the Number of
Branches field will be the same as that of the request message.

The Verify Tree message can only be guaranteed to yield a correct
response when there are no other connection request messages or their
response messages pending for the specified connection. If this is
not the case the result of the Verify Tree message is undefined.

3.5 Delete All Message

The Delete All message is a connection management message used to
delete all connections on a switch input port. All connections that
arrive at the specified input port must be deleted. On completion of
the operation all dynamically assigned VPI/VCI values for the
specified port must be unassigned, i.e. there must be no virtual
connections established in the VPI/VCI space that GSMP controls on
this port. The Input VPI, Input VCI, Output Port, Output VPI, and
Output VCI fields are not used in this message and their contents are
ignored and unspecified. The Delete All message is"

        Message Type = 20

If the Result field of the request message is "AckAll" a success
response message must be sent upon completion of the operation. The
success response message must not be sent until the operation has
been completed.

3.6 Move Branch Message

The Move Branch connection management message has the following
format for both request and response messages:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    Version    | Message Type  |    Result     |     Code      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Transaction Identifier                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     Port Session Number                       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         Input Port                            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | zero  |       Input VPI       |           Input VCI           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      Old Output Port                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | zero  |     Old Output VPI    |        Old Output VCI         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      New Output Port                          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | zero  |    New Output VPI     |        New Output VCI         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      Reserved                 |   Priority    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
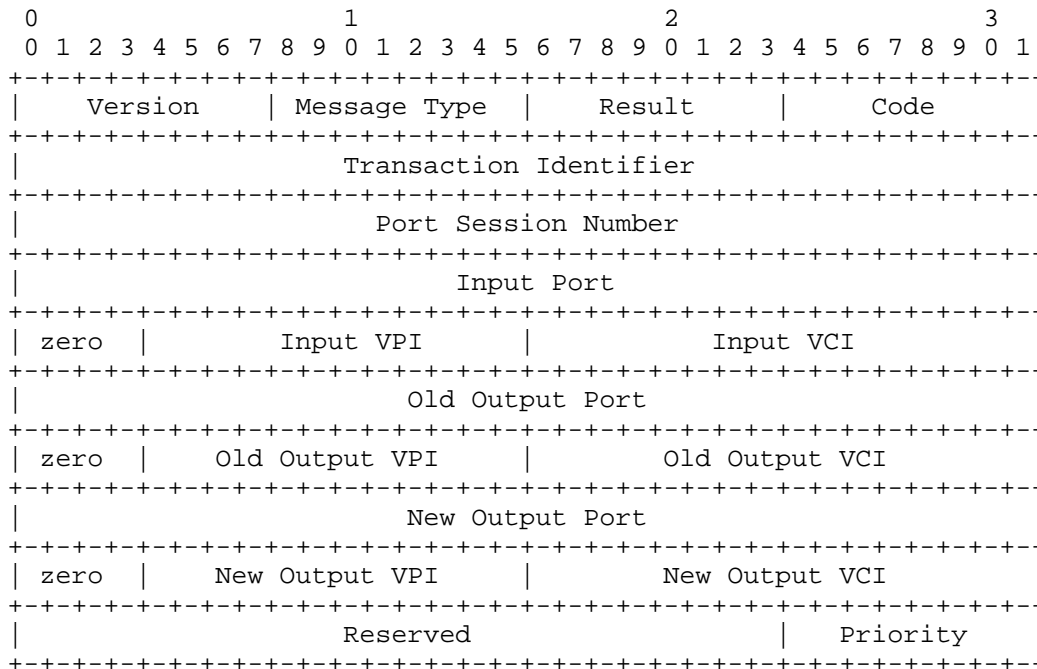
The Move Branch message is a connection management message used to
move a single output branch of a virtual channel connection from its
current output port, output VPI, and output VCI, to a new output
port, output VPI, and output VCI on the same virtual channel
connection. None of the other output branches are modified. When the
operation is complete the original output VPI/VCI on the original
output port will be deleted from the connection. The Move Branch
message is:

    Message Type = 22

If the virtual channel connection specified by the Input Port, Input
VPI, and Input VCI fields already exists, and the output branch
specified by the Old Output Port, Old Output VPI, and Old Output VCI
fields exists as a branch on that connection, the output branch
specified by the New Output Port, New Output VPI, and New Output VCI
fields is added to the connection and the branch specified by the Old
Output Port, Old Output VPI, and Old Output VCI fields is deleted. If
the Result field of the request message is "AckAll" a success
response message must be sent upon successful completion of the
operation. The success response message must not be sent until the
Move Branch operation has been completed.

If the virtual channel connection specified by the Input Port, Input
VPI, and Input VCI fields already exists, but the output branch
specified by the Old Output Port, Old Output VPI, and Old Output VCI
fields does not exist as a branch on that connection, a failure
response must be returned with the Code field indicating, "The
specified branch does not exist." The connection state of the switch
must not be modified in this case.

If the virtual channel connection specified by the Input Port, Input
VPI, and Input VCI fields does not exist, a failure response must be
returned with the Code field indicating, "The specified connection
does not exist." The connection state of the switch must not be
modified in this case.

The behavior is undefined if the output virtual channel specified by
the New Output Port, New Output VPI, and New Output VCI fields is
already in use by any connection.

A failure response will be returned if the switch is unable to
establish the specified branch or if there is an error in any of the
fields of the request message. If a failure message is returned the
state of the switch must not have been modified by the request
message.

4. Port Management Message

The Port Management message allows a port to be brought into service,
taken out of service, looped back, or reset. Only the Bring Up and
the Reset Input Port functions change the connection state
(established connections) on the input port. Only the Bring Up
function changes the value of the Port Session Number. If the Result
field of the request message is "AckAll" a success response message
must be sent upon successful completion of the operation. The success
response message must not be sent until the operation has been
completed. The Port Management Message is:

    Message Type = 32

The Port Management message has the following format for the request
and success response messages:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    | Message Type  |    Result     |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Transaction Identifier                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Port                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Port Session Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Event Sequence Number                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Event Flags  |   Duration    |           Function           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
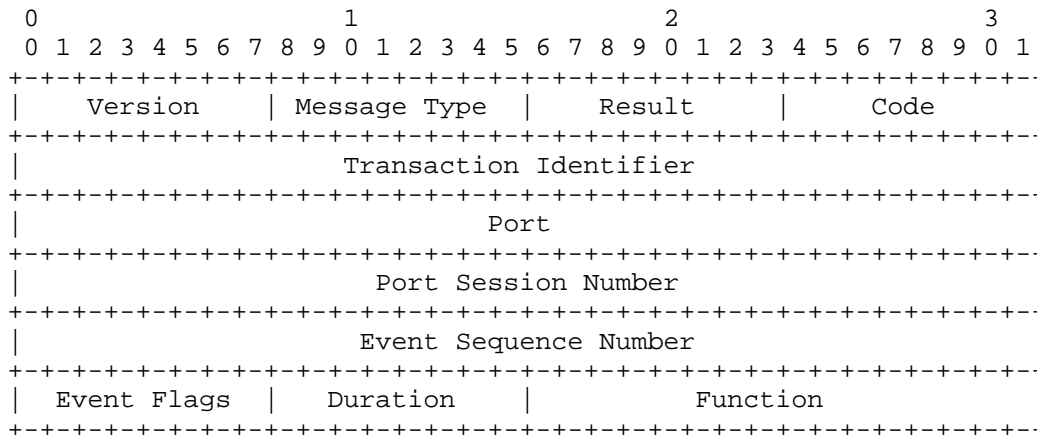
Port

        Gives the port number of the port to which the message
        applies.

Port Session Number

        Gives the current port session number for the port. If the
        Port Session Number in the request message does not match
        the current port session number of the port indicated by
        the Port field of the request message, a failure response
        must be returned with, "Invalid port session number,"
        indicated in the Code field. If the specified function
        requires a new Port Session Number to be generated the new
        Port Session Number must be given in the success response
        message. The Port Session Number must be generated using
        some form of random number.

Event Sequence Number

        In the success response message gives the current value of
        the Event Sequence Number of the switch port indicated by
        the Port field. The Event Sequence Number is set to zero
        when the port is initialized and is incremented by one each
        time an asynchronous event is detected on that port that
        the switch would normally report via an Event message. If
        the Event Sequence Number in the success response differs
        from the Event Sequence Number of the most recent Event
        message received for that port, events have occurred that
        were not reported via an Event message. This is most likely
        to be due to the flow control that restricts the rate at
        which a switch can send Event messages for each port. In
        the request message this field is not used and should be
        set to zero by the sender and ignored by the receiver.

Event Flags

Field in the request message is used to reset the Event
Flags in the switch port indicated by the Port field. Each
Event Flag in a switch port corresponds to a type of Event
message. When a switch port sends an Event message it sets
the corresponding Event Flag on that port. The port is not
permitted to send another Event message of the same type
until the Event Flag has been reset. If the Function field
in the request message is set to "Reset Event Flags," for
each bit that is set in the Event Flags field, the
corresponding Event Flag in the switch port is reset.

The Event Flags field is only used in a request message
with the Function field set to "Reset Event Flags." For all
other values of the Function field, the Event Flags field
should be set to zero in the request message and must be
ignored by the receiver. In the success response message
the Event Flags field must be set to the current value of
the Event Flags for the port, after the completion of the
operation specified by the request message, for all values
of the Function field. Setting the Event Flags field to all
zeros in a "Reset Event Flags" request message allows the
controller to obtain the current state of the Event Flags
and the current Event Sequence Number of the port without
changing the state of the Event Flags.

The correspondence between the types of Event message and
the bits of the Event Flags field is as follows:

        Port Up: Bit 0, (most significant bit)
        Port Down: Bit 1,
        Invalid VPI/VCI: Bit 2,
        New Port: Bit 3,
        Dead Port: Bit 4.

Duration

Is the length of time, in seconds, that any of the loopback
states remain in operation. When the duration has expired
the port will automatically be returned to service. If
another Port Management message is received for the same
port before the duration has expired, the loopback will
continue to remain in operation for the length of time
specified by the Duration field in the new message. The
Duration field is only used in request messages with the
Function field set to Internal Loopback, External Loopback,
or Bothway Loopback. In all other request messages it
should be set to zero by the sender and ignored by the
receiver.

Function

> Specifies the action to be taken. The specified action will
> be taken regardless of the current status of the port
> (Available, Unavailable, or any Loopback state). The
> defined values of the Function field are:

Bring Up:

> Function = 1. Bring the port into service. All
> connections that arrive at the specified input port
> must be deleted and a new Port Session Number must be
> selected using some form of random number. On
> completion of the operation all dynamically assigned
> VPI/VCI values for the specified input port must be
> unassigned, i.e. no virtual connections will be
> established in the VPI/VCI space that GSMP controls on
> this input port. The Port Status of the port
> afterwards will be Available.

Take Down:

> Function = 2. Take the port out of service. Any cells
> received at this port will be discarded. No cells will
> be transmitted from this port. The Port Status of the
> port afterwards will be Unavailable. The behavior is
> undefined if the port over which the GSMP protocol is
> running is taken down.

Internal Loopback:

> Function = 3. Cells arriving at the output port from
> the switch fabric are looped through to the input port
> to return to the switch fabric. All of the ATM
> functions of the input port above the PHY layer, e.g.
> header translation, are performed upon the looped back
> cells. The Port Status of the port afterwards will be
> Internal Loopback.

External Loopback:

> Function = 4. Cells arriving at the input port from
> the external communications link are immediately
> looped back to the communications link at the physical
> layer without entering the input port. None of the ATM
> functions of the input port above the PHY layer are
> performed upon the looped back cells. The Port Status
> of the port afterwards will be External Loopback.

Bothway Loopback:

> Function = 5. Both internal and external loopback are
> performed. The Port Status of the port afterwards will
> be Bothway Loopback.

Reset Input Port:
    Function = 6. All connections that arrive at the
    specified input port must be deleted and the input and
    output port hardware re-initialized. On completion of
    the operation all dynamically assigned VPI/VCI values
    for the specified input port must be unassigned, i.e.
    no virtual connections will be established in the
    VPI/VCI space that GSMP controls on this input port.
    The Port Session Number is not changed by the Reset
    Input Port function. The Port Status of the port
    afterwards will be Unavailable.

Reset Event Flags:
    Function = 7. For each bit that is set in the Event
    Flags field, the corresponding Event Flag in the
    switch port must be reset. The Port Status of the port
    is not changed by this function.

5. Statistics Messages

   The statistics messages permit the controller to request the values
   of various hardware counters associated with the switch input and
   output ports, and virtual channels. Two classes of statistics message
   are defined: the VC Activity Message, and the Port and VC Statistics
   Messages. The VC Activity message is used to determine whether one or
   more specific VCs have recently been carrying traffic. The Port and
   VC Statistics message is used to query the various port and VC
   specific traffic and error counters.

5.1 VC Activity Message

   The VC Activity message is used to determine whether one or more
   specific VCs have recently been carrying traffic. The VC Activity
   message contains one or more VC Activity records. Each VC Activity
   record is used to request and return activity information concerning
   a single virtual connection. Each VC is specified by its input port,
   input VPI, and input VCI. These are specified in the Input Port,
   Input VPI, and Input VCI fields of each VC Activity record. Two
   forms of activity detection are supported. If the switch supports per
   VC traffic accounting the current value of the traffic counter for
   each specified VC must be returned. The units of traffic counted are
   not specified but will typically be either cells or frames. The
   controller must compare the traffic counts returned in the message
   with previous values for each of the specified VCs to determine
   whether each VC has been active in the intervening period. If the
   switch does not support per VC traffic accounting, but is capable of
   detecting per-VC activity by some other unspecified means, the result

may be indicated for each VC using the Flags field. The VC Activity
message is:

    Message Type = 48

The VC Activity request and success response messages have the
following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    | Message Type  |    Result     |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Transaction Identifier                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Number of Records       |            Reserved           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                     VC Activity Records                       ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
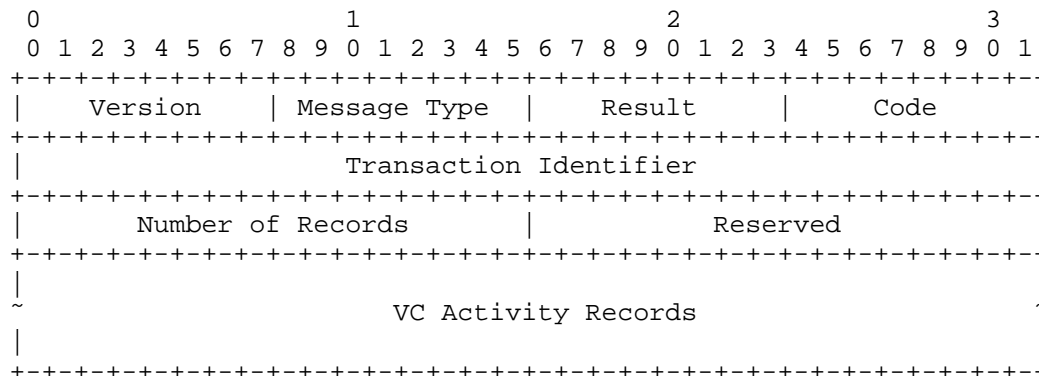
Number of Records
        Field specifies the number of VC Activity records to
        follow. The maximum number of VC Activity records permitted
        in a single VC Activity message is 120.

Reserved
        Field is not used. It is set to zero by the sender and
        ignored by the receiver.

Each VC Activity Record has the following format:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Input Port                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Flags |      Input VPI        |            Input VCI          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                       VC Traffic Count                        +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
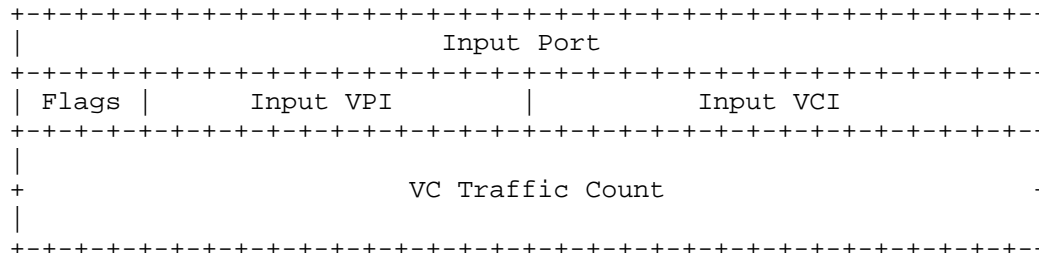
Input Port
        Identifies the port number of the input port on which the
        VC of interest arrives in order to identify the VC
        (regardless of whether the traffic count for the VC is
        maintained on the input port or the output port).

   Input VPI
   Input VCI
            Fields identify the specific virtual channel for which
            statistics are being requested.

   Flags

            In the request message this field is unused, it should be
            set to zero by the sender and ignored by the receiver. In
            the success response message bit 0 (msb) of the Flags field
            is used to indicate an invalid VC Activity record. This bit
            must be zero if any of the fields in this VC Activity
            record are invalid, if the input port specified by the
            Input Port field does not exist, or if the specified
            connection does not exist. If this bit is zero in a success
            response message bits 1 and 2 of the Flags field and the VC
            Traffic Count field are undefined. If bit 0 of the flags
            field is set, the VC Activity record is valid, and bits 1
            and 2 of the Flags field in the VC Activity record are used
            as follows:

                Bit 1 of the Flags field: if set, indicates that the
                value in bit 2 of the Flags field is valid; if zero,
                indicates that the value in the VC Traffic Count field
                is valid.

                If bit 1 of the Flags field is set, bit 2 of the Flags
                field, if set, indicates that there has been some
                activity on this virtual channel since the last VC
                Activity message for this virtual channel.

                If bit 1 of the Flags field is set, bit 2 of the Flags
                field, if zero, indicates that there has been no
                activity on this virtual channel since the last VC
                Activity message for this virtual channel.

                Bit 3 of the Flags field is not used, it should be set
                to zero by the sender and ignored by the receiver.

   VC Traffic Count
            Field is unused in the request message, it should be set to
            zero by the sender and ignored by the receiver. In the
            success response message, if the switch supports per-VC
            traffic counting, the VC Traffic Count field must be set to
            the value of a free running, VC specific, 64 bit traffic
            counter counting traffic flowing across the specified
            virtual channel. The value of the traffic counter is not
            modified by reading it. If per-VC traffic counting is
            supported, the switch must report the VC Activity result

using the traffic count rather than using bit 2 of the
Flags field.

The format of the failure response is the same as the request message
with the Number of Records field set to zero and no VC Activity
records returned in the message. If the switch is incapable of
detecting per-VC activity, a failure response must be returned
indicating, "The specified request is not implemented on this
switch."

5.2 Port and VC Statistics Messages

The Port and VC Statistics messages are used to query the various
port and VC specific traffic and error counters.

The Port and VC Statistics request messages have the following
format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    | Message Type  |    Result     |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Transaction Identifier                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Port                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| zero  |         VPI           |              VCI              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
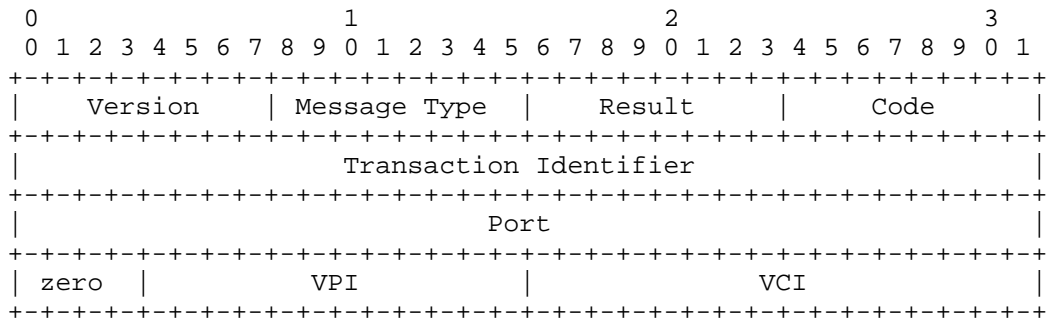
Port

        Identifies the port number of the port for which statistics
        are being requested.

VPI
VCI

        Fields identify the specific virtual channel for which
        statistics are being requested. For requests that do not
        require a virtual channel to be specified these fields
        should be set to zero in the request and ignored by the
        receiver.

The success response messages for the port and VC statistics group
have the following format:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    Version    |  Message Type |     Result    |     Code      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     Transaction Identifier                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                             Port                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | zero  |          VPI          |             VCI               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                        Input Cell Count                       +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                       Input Frame Count                       +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                    Input Cell Discard Count                   +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                   Input Frame Discard Count                   +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                     Input HEC Error Count                     +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                   Input Invalid VPI/VCI Count                 +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                       Output Cell Count                       +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                      Output Frame Count                       +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                   Output Cell Discard Count                   +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
```
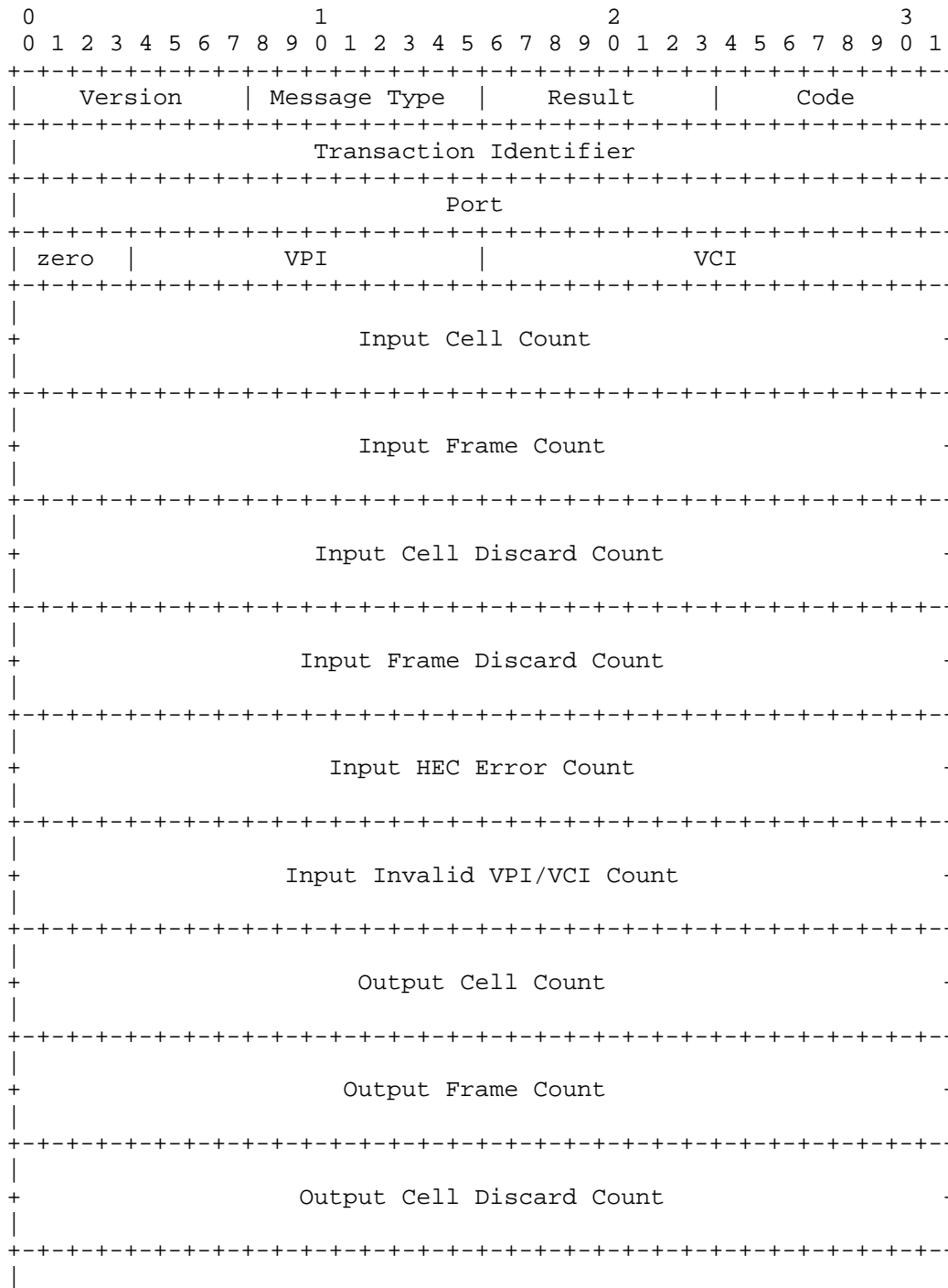
```
     +                    Output Frame Discard Count                  +
     |                                                                |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

     Port
     VPI/VCI
             Fields are the same as those of the request message.

     Input Cell Count
     Output Cell Count
             Each gives the value of a free running 64 bit counter
             counting cells arriving at the input or departing from the
             output respectively. In response to a Port Statistics
             message the count will be on a per port basis and in
             response to a VC Statistics message the count will be on a
             per VC basis.

     Input Frame Count
     Output Frame Count
             Each gives the value of a free running 64 bit counter
             counting frames (packets) arriving at the input or
             departing from the output respectively. In response to a
             Port Statistics message the count will be on a per port
             basis and in response to a VC Statistics message the count
             will be on a per VC basis.

     Input Cell Discard Count
     Output Cell Discard Count
             Each gives the value of a free running 64 bit counter
             counting cells discarded due to queue overflow on an input
             port or on an output port respectively. In response to a
             Port Statistics message the count will be on a per port
             basis and in response to a VC Statistics message the count
             will be on a per VC basis.

     Input Frame Discard Count
     Output Frame Discard Count
             Each gives the value of a free running 64 bit counter
             counting frames discarded due to queue overflow on an input
             port or on an output port respectively. In response to a
             Port Statistics message the count will be on a per port
             basis and in response to a VC Statistics message the count
             will be on a per VC basis.

     HEC Error Count
             Gives the value of a free running 64 bit counter counting
             cells discarded due to header checksum errors on arrival at
             an input port.

Invalid VPI/VCI Count
          Gives the value of a free running 64 bit counter counting
          cells discarded because their VPI/VCI is invalid on arrival
          at an input port. An incoming VPI/VCI is invalid if no
          connection is currently established having that value of
          VPI/VCI.

5.2.1 Port Statistics Message

   The Port Statistics message requests the statistics for the switch
   port specified in the Port field. The contents of the VPI/VCI field
   in the Port Statistics request message are ignored. All of the count
   fields in the success response message refer to per-port counts
   regardless of the virtual channels to which the cells belong. Any of
   the count fields in the success response message not supported by the
   port will be set to zero. The Port Statistics message is:

      Message Type = 49

5.2.2 VC Statistics Message

   The VC Statistics message requests the statistics for the virtual
   channel specified in the VPI/VCI field that arrives on the switch
   input port specified in the Port field. All of the count fields in
   the success response message refer only to the specified virtual
   channel. The HEC Error Count and Invalid VPI/VCI Count fields are not
   VC specific and are set to zero. Any of the other count fields not
   supported on a per virtual channel basis will be set to zero in the
   success response message. The VC Statistics message is:

      Message Type = 50

6. Configuration

   The configuration messages permit the controller to discover the
   capabilities of the switch. Three configuration request messages have
   been defined: Switch, Port, and All Ports.

All configuration request messages have the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    | Message Type  |     Result    |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Transaction Identifier                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Port                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
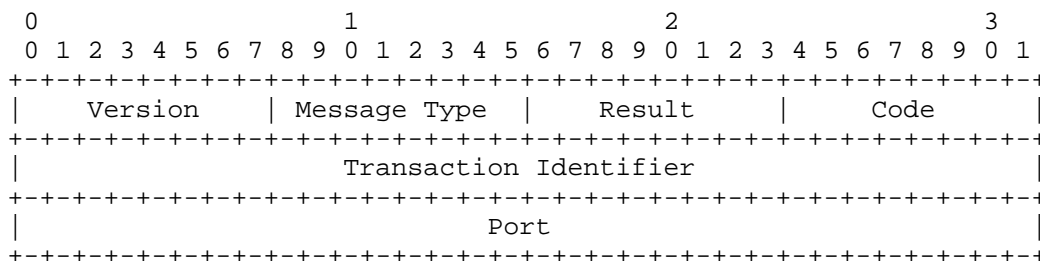
Port

> Identifies the port number for which configuration
> information is being requested. If the Port field is not
> required by the message it is set to zero by the sender and
> ignored by the receiver.

6.1 Switch Configuration Message

The Switch Configuration message requests the global (non port-
specific) configuration for the switch. The Switch Configuration
message is:

     Message Type = 64

The Port field is not used in the request message and is set to zero.

The Switch Configuration success response message has the following
format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    | Message Type  |     Result    |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Transaction Identifier                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Firmware Version Number    |            Reserved          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Switch Type          |                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                              +
|                          Switch Name                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
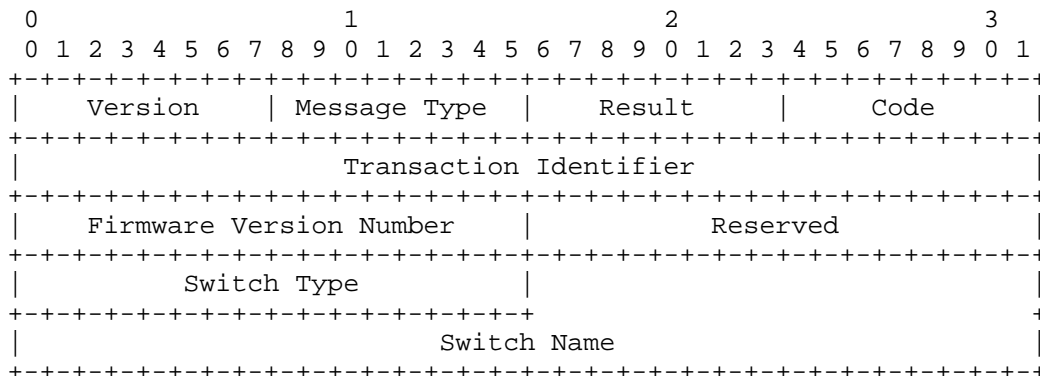
Firmware Version Number

> The version number of the switch control firmware
> installed.

    Reserved
            Field is not used. It is set to zero by the sender and
            ignored by the receiver.

    Switch Type
            A 16 bit field allocated by the manufacturer of the switch.
            (For these purposes the manufacturer of the switch is
            assumed to be the organization identified by the OUI in the
            Switch Name field.) The Switch Type identifies the product.
            When the Switch Type is combined with the OUI from the
            Switch Name the product is uniquely identified. Network
            Management may use this identification to obtain product
            related information from a database.

    Switch Name
            A 48 bit quantity that is unique within the operational
            context of the device. A 48 bit IEEE 802 MAC address, if
            available, may be used as the Switch Name. The most
            significant 24 bits of the Switch Name must be an
            Organizationally Unique Identifier (OUI) that identifies
            the manufacturer of the switch.

6.2 Port Configuration Message

    The Port Configuration message requests the switch for the
    configuration information of a single switch port. The Port field in
    the request message specifies the port for which the configuration is
    requested. The Port Configuration message is:

        Message Type = 65.

    The Port Configuration success response message has the following
    format:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    Version    | Message Type  |    Result     |     Code      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     Transaction Identifier                    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Port                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      Port Session Number                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | zero  |      Min VPI          | zero  |       Max VPI         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |        Min VCI                |           Max VCI             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         Cell Rate                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Port Status  |  Port Type    |  Line Status  |  Priorities   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
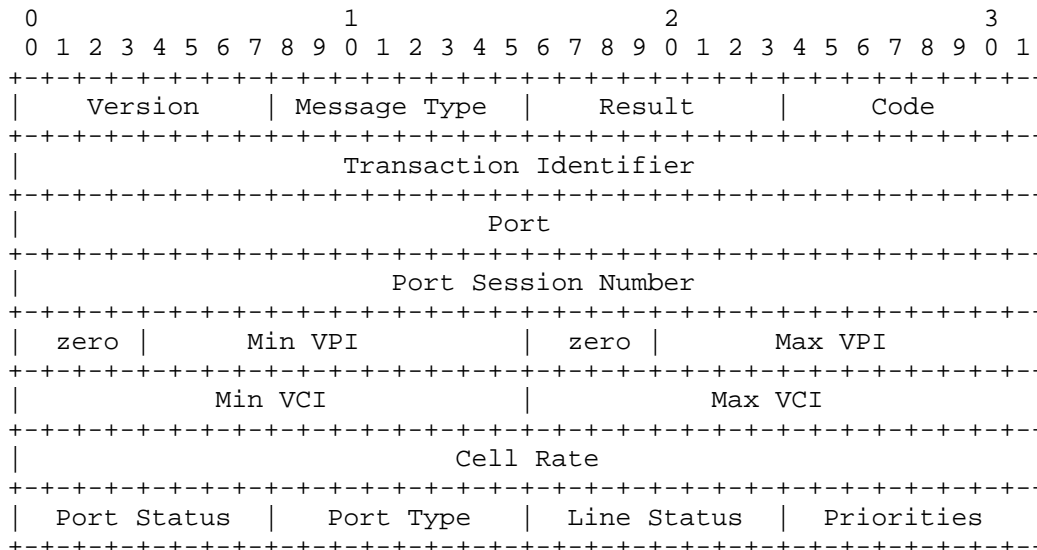
Port

> The switch port to which the configuration information
> refers. Configuration information relating to both the
> input and the output sides of the switch port is given.
> Port numbers are 32 bits wide and allocated by the switch.
> The switch may choose to structure the 32 bits into sub
> fields that have meaning to the physical structure of the
> switch hardware (e.g. shelf, slot, interface).

Port Session Number

> The current Port Session Number for the specified port.
> Each switch port maintains a Port Session Number assigned
> by the switch. The Port Session Number of a port remains
> unchanged while the port is continuously in the Available
> state. When a port returns to the Available state after it
> has been Unavailable, or after a power cycle, its Port
> Session Number must be changed, preferably using some form
> of random number.

Min VPI

> The minimum value of dynamically assigned incoming VPI that
> the connection table on the input port can support and may
> be controlled by GSMP.

Max VPI

> The maximum value of dynamically assigned incoming VPI that
> the connection table on the input port can support and may
> be controlled by GSMP. It is assumed that the input port

can handle all values of VPI within the range Min VPI to
Max VPI inclusive and that GSMP may control all values
within this range. If the switch does not support virtual
paths it is acceptable for both Min VPI and Max VPI to
specify the same value, most likely zero.

Min VCI

The minimum value of dynamically assigned incoming VCI that
the connection table on the input port can support and may
be controlled by GSMP.

Max VCI

The maximum value of dynamically assigned incoming VCI that
the connection table on the input port can support and may
be controlled by GSMP. It is assumed that the input port
can handle all values of VCI within the range Min VCI to
Max VCI inclusive for each of the virtual paths in the
range Min VPI to Max VPI inclusive and that GSMP may
control all values within this range.

Cell Rate

A measure of the bandwidth of the port. It is the rate of
cells arriving at or departing from the port in cells/s. It
is assumed that both input port and output port have the
same cell rate.

Port Status

Gives the administrative state of the port. The defined
values of the Port Status field are:

Available:
    Port Status = 1. The port is available to both send
    and receive cells. When a port changes to the
    Available state from any other administrative state,
    all dynamically assigned virtual connections must be
    cleared and a new Port Session Number must be
    generated.

Unavailable:
    Port Status = 2. The port has intentionally been taken
    out of service. No cells will be transmitted from this
    port. No cells will be received by this port.

Internal Loopback:
    Port Status = 3. The port has intentionally been taken
    out of service and is in internal loopback: cells
    arriving at the output port from the switch fabric are
    looped through to the input port to return to the

switch fabric. All of the ATM functions of the input
port above the PHY layer, e.g. header translation, are
performed upon the looped back cells.

External Loopback:
Port Status = 4. The port has intentionally been taken
out of service and is in external loopback: cells
arriving at the input port from the external
communications link are immediately looped back to the
communications link at the physical layer without
entering the input port. None of the ATM functions of
the input port above the PHY layer are performed upon
the looped back cells.

Bothway Loopback:
Port Status = 5. The port has intentionally been taken
out of service and is in both internal and external
loopback.

Port Type
The type of physical transmission interface for this port.
The values for this field are given by the IANAifTYPE
object from the MIB defined for the IANAifTYPE-MIB
specified in RFC 1573 [rfc1573]. Example values are: SONET
or SDH (39), DS-3 (30).

Line Status
The status of the physical transmission medium connected to
the port. The defined values of the Line Status field are:

Up:
Line Status = 1. The line is able to both send and
receive cells. When the Line Status changes to Up
from either the Down or Test states, a new Port
Session Number must be generated.

Down:
Line Status = 2. The line is unable either to send or
receive cells or both.

Test:
Line Status = 3. The port or line is in a test mode,
for example, power-on test.

Priorities
The number of different priorities that this output port
can assign to virtual channel connections. Zero is invalid
in this field. If an output port is able to support "Q"

> priorities, the highest priority is numbered zero and the
> lowest priority is numbered "Q-1". The ability to offer
> different qualities of service to different connections
> based upon their priority is assumed to be a property of
> the output port of the switch. It may be assumed that for
> virtual channel connections that share the same output
> port, an ATM cell on a connection with a higher priority is
> much more likely to exit the switch before an ATM cell on a
> connection with a lower priority if they are both in the
> switch at the same time.

6.3 All Ports Configuration Message

   The All Ports Configuration message requests the switch for the
   configuration information of all of its ports. The All Ports
   Configuration message is:

      Message Type = 66

   The Port field is not used in the request message and is set to zero.

   The All Ports Configuration success response message has the
   following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    |  Message Type |     Result    |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Transaction Identifier                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Number of Records        |      Port Record Length       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                         Port Records                          ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
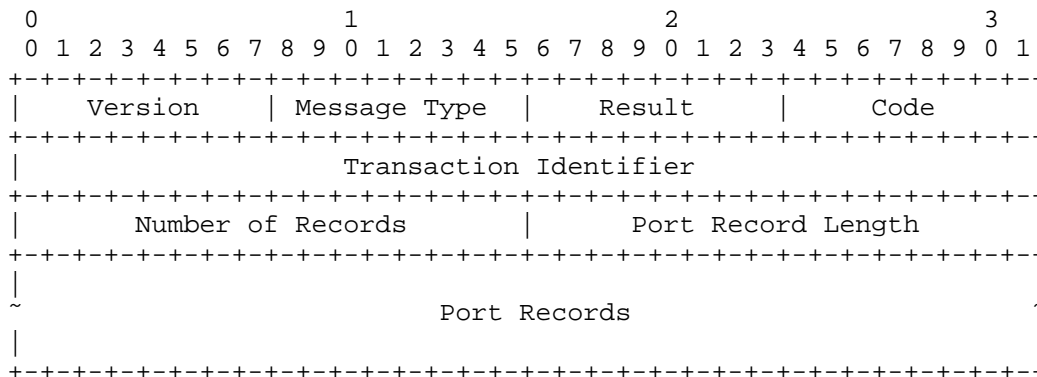
   Number of Records
           Field gives the number of Port Records to follow in the
           message. The maximum number of port records allowed in a
           single All Ports Configuration success response is 64. If a
           switch has more than 64 ports it must send them in multiple
           success response messages.

   Port Record Length
           Field gives the length of each port record in bytes. This
           is currently 24 but the Port Record Length field allows for

the future definition of further fields at the end of the
port record while preserving compatibility with earlier
versions of the protocol.

Port Records follow in the remainder of the message. Each port record
has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Port                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Port Session Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  zero |      Min VPI          |  zero |        Max VPI        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Min VCI              |          Max VCI               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Cell Rate                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Port Status  |  Port Type   |  Line Status  |  Priorities   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
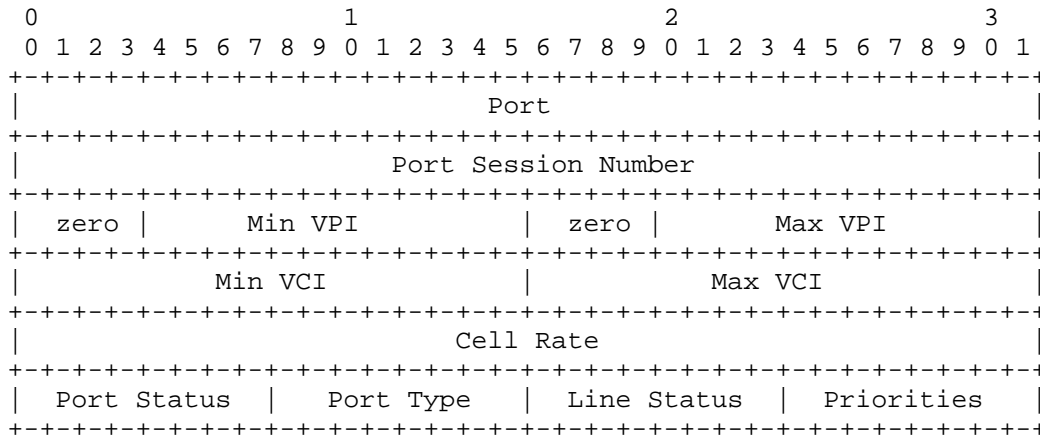
The definition of the fields in the port record is exactly the same
as that of the Port Configuration message.

7. Event Messages

Event messages allow the switch to inform the controller of certain
asynchronous events. Event messages are not acknowledged. The Result
field and the Code field in the message header are not used and
should be set to zero. Event messages are not sent during
initialization. Event messages have the following format:

```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |    Version    | Message Type  |    Result     |     Code      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                    Transaction Identifier                     |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                            Port                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                     Port Session Number                       |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                    Event Sequence Number                      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | zero  |        VPI          |            VCI                  |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
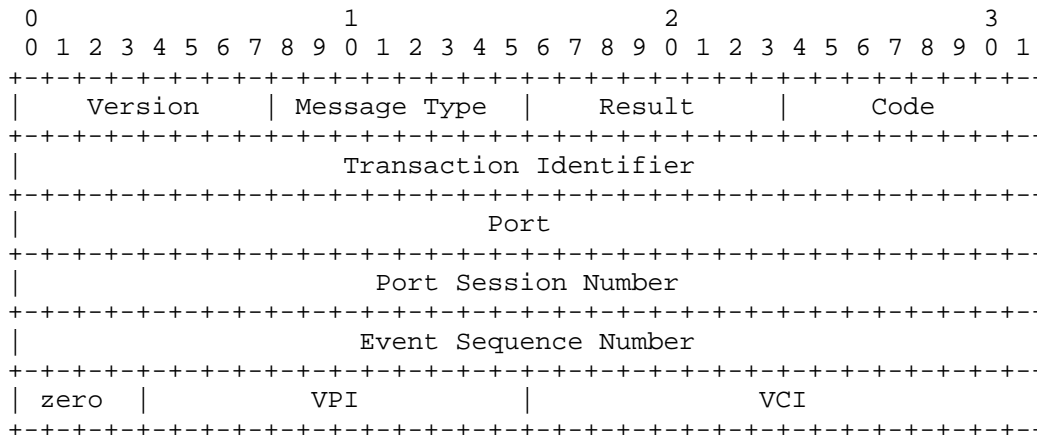
Port

        Field gives the switch port to which the event message
        refers.

Port Session Number

        The current Port Session Number for the specified port.

Event Sequence Number

        The current value of the Event Sequence Number for the
        specified port. The Event Sequence Number is set to zero
        when the port is initialized and is incremented by one each
        time an asynchronous event is detected on that port that
        the switch would normally report via an Event message. The
        Event Sequence Number must be incremented each time an
        event occurs even if the switch is prevented from sending
        an Event message due to the action of the flow control.

VPI/VCI

        Field gives the VPI/VCI to which the event message refers.
        If this field is not required by the event message it is
        set to zero.

Each switch port must maintain an Event Sequence Number and a set of
Event Flags, one Event Flag for each type of Event message. When a
switch port sends an Event message it must set the Event Flag on that
port corresponding to the type of the event. The port is not
permitted to send another Event message of the same type until the
Event Flag has been reset. Event Flags are reset by the "Reset Event
Flags" function of the Port Management message. This is a simple flow
control preventing the switch from flooding the controller with event
messages. The Event Sequence Number of the port must be incremented
every time an event is detected on that port even if the port is

prevented from reporting the event due to the action of the flow
control. This allows the controller to detect that it has not been
informed of some events that have occurred on the port due to the
action of the flow control.

7.1 Port Up Message

The Port Up message informs the controller that the Line Status of a
port has changed from either the Down or Test state to the Up state.
When the Line Status of a switch port changes to the Up state from
either the Down or Test state a new Port Session Number must be
generated, preferably using some form of random number. The new Port
Session Number is given in the Port Session Number field. The VPI/VCI
field is not used and is set to zero. The Port Up message is:

     Message Type = 80

7.2 Port Down Message

The Port Down message informs the controller that the Line Status of
a port has changed from the Up state to the Down state. This message
will be sent to report link failure if the switch is capable of
detecting link failure. The port session number that was valid before
the port went down is reported in the Port Session Number field. The
VPI/VCI field is not used and is set to zero. The Port Down message
is:

     Message Type = 81

7.3 Invalid VPI/VCI Message

The Invalid VPI/VCI message is sent to inform the controller that one
or more cells have arrived at an input port with a VPI/ VCI that is
currently not allocated to an assigned connection. The input port is
indicated in the Port field, and the VPI/VCI in the VPI/VCI field.
The Invalid VPI/VCI message is:

     Message Type = 82

7.4 New Port Message

The New Port message informs the controller that a new port has been
added to the switch. The port number of the new port is given in the
Port field. A new Port Session Number must be assigned, preferably
using some form of random number. The new Port Session Number is
given in the Port Session Number field. The state of the new port is
undefined so the VPI/VCI field is not used and is set to zero. The
New Port message is:

        Message Type = 83

7.5 Dead Port Message

   The Dead Port message informs the controller that a port has been
   removed from the switch. The port number of the port is given in the
   Port field. The Port Session Number that was valid before the port
   was removed is reported in the Port Session Number field. The
   VPI/VCI fields are not used and are set to zero. The Dead Port
   message is:

        Message Type = 84

8. Adjacency Protocol

   The adjacency protocol is used to synchronize state across the link,
   to discover the identity of the entity at the other end of a link,
   and to detect when it changes. No GSMP messages other than those of
   the adjacency protocol may be sent across the link until the
   adjacency protocol has achieved synchronization.

8.1 Packet Format

   The adjacency protocol is:

        Message Type = 10

   All GSMP messages belonging to the adjacency protocol have the
   following structure:
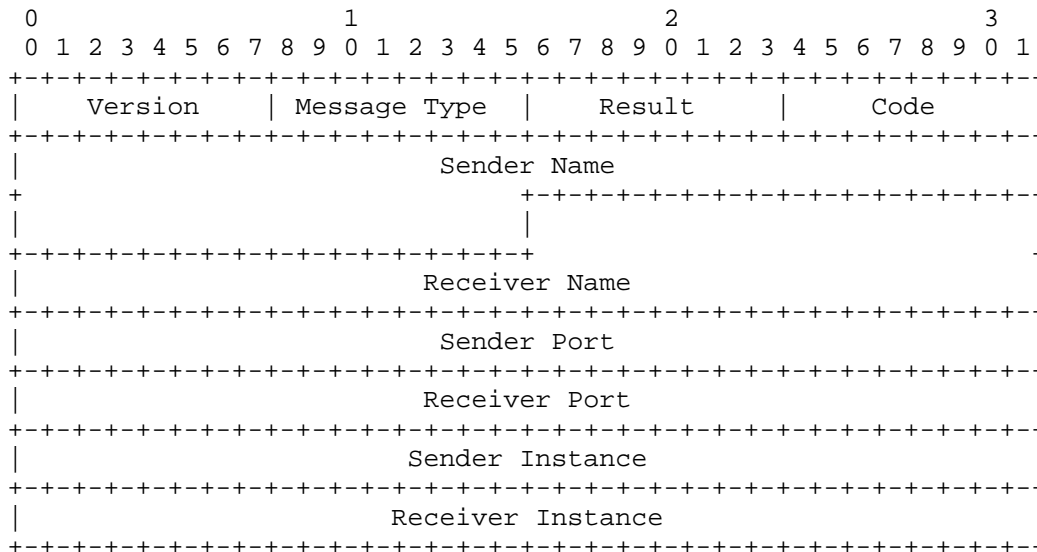
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    | Message Type  |    Result     |     Code      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sender Name                            |
+                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                        Receiver Name                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sender Port                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Receiver Port                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sender Instance                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Receiver Instance                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Version

        The GSMP protocol version number, currently Version = 1. It
        should be set by the sender of the message to the GSMP
        protocol version that the sender is currently running.

Result

        Field is not used in the adjacency protocol. It should be
        set to zero by the sender and ignored by the receiver.

Code

        Field specifies the function of the message. Four Codes are
        defined for the adjacency protocol:

            SYN: Code = 1
            SYNACK: Code = 2
            ACK: Code = 3
            RSTACK: Code = 4.

Sender Name

        For the SYN, SYNACK, and ACK messages, is the name of the
        entity sending the message. The Sender Name is a 48 bit
        quantity that is unique within the operational context of
        the device. A 48 bit IEEE 802 MAC address, if available,
        may be used for the Sender Name. For the RSTACK message,
        the Sender Name field is set to the value of the Receiver
        Name field from the incoming message that caused the RSTACK
        message to be generated.

Receiver Name

For the SYN, SYNACK, and ACK messages, is the name of the
entity that the sender of the message believes is at the
far end of the link. If the sender of the message does not
know the name of the entity at the far end of the link,
this field should be set to zero. For the RSTACK message,
the Receiver Name field is set to the value of the Sender
Name field from the incoming message that caused the RSTACK
message to be generated.

Sender Port

For the SYN, SYNACK, and ACK messages, is the local port
number of the link across which the message is being sent.
Port numbers are locally assigned 32 bit values. For the
RSTACK message, the Sender Port field is set to the value
of the Receiver Port field from the incoming message that
caused the RSTACK message to be generated.

Receiver Port

For the SYN, SYNACK, and ACK messages, is what the sender
believes is the local port number for the link, allocated
by the entity at the far end of the link. If the sender of
the message does not know the port number at the far end of
the link, this field should be set to zero. For the RSTACK
message, the Receiver Port field is set to the value of the
Sender Port field from the incoming message that caused the
RSTACK message to be generated.

Sender Instance

For the SYN, SYNACK, and ACK messages, is the sender's
instance number for the link. It is used to detect when the
link comes back up after going down or when the identity of
the entity at the other end of the link changes. The
instance number is a 32 bit number that is 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. For the RSTACK message, the Sender
Instance field is set to the value of the Receiver Instance
field from the incoming message that caused the RSTACK
message to be generated.

Receiver Instance

For the SYN, SYNACK, and ACK messages, is what the sender
believes is the current instance number for the link,
allocated by the entity at the far end of the link. If the
sender of the message does not know the current instance
number at the far end of the link, this field should be set
to zero. For the RSTACK message, the Receiver Instance

                    field is set to the value of the Sender Instance field from
                    the incoming message that caused the RSTACK message to be
                    generated.

8.2 Procedure

   The adjacency protocol is described by the rules and state tables
   given in this section.

   The rules and state tables use the following operations:

   o The "Update Peer Verifier" operation is defined as storing the
     values of the Sender Instance, Sender Port, and Sender Name fields
     from a SYN or SYNACK message received from the entity at the far
     end of the link.

   o The procedure "Reset the link" is defined as:

          1. Generate a new instance number for the link
          2. Delete the peer verifier (set to zero the values of Sender
             Instance, Sender Port, and Sender Name previously stored by
             the Update Peer Verifier operation)
          3. Send a SYN message
          4. Enter the SYNSENT state

   o The state tables use the following Boolean terms and operators:

       A   The Sender Instance in the incoming message matches the
            value stored from a previous message by the "Update Peer
            Verifier" operation.

       B   The Sender Instance, Sender Port, and Sender Name fields in
            the incoming message match the values stored from a
            previous message by the "Update Peer Verifier" operation.

       C   The Receiver Instance, Receiver Port, and Receiver Name
            fields in the incoming message match the values of the
            Sender Instance, Sender Port, and Sender Name currently
            sent in outgoing SYN, SYNACK, and ACK messages.

      "&&" Represents the logical AND operation

      "||" Represents the logical OR operation

      "!" Represents the logical negation (NOT) operation.

   o A timer is required for the periodic generation of SYN, SYNACK,
     and ACK messages. The period of the timer is unspecified but a
     value of one second is suggested.

     There are two independent events: the timer expires, and a packet
     arrives. The processing rules for these events are:

        Timer Expires: Reset Timer
                         If state = SYNSENT Send SYN
                         If state = SYNRCVD Send SYNACK
                         If state = ESTAB   Send ACK

        Packet Arrives: If incoming message is an RSTACK
                            If A && C && !SYNSENT
                               Reset the link
                            Else Discard the message
                         Else the following State Tables.

   o State synchronization across a link is considered to be achieved
     when the protocol reaches the ESTAB state.

     State Tables

State: SYNSENT

```
+==============================================================================+
|     Condition      |                 Action                 | New State |
+====================+========================================+===========+
|    SYNACK && C     |  Update Peer Verifier; Send ACK        |   ESTAB   |
+--------------------+----------------------------------------+-----------+
|    SYNACK && !C    |             Send RSTACK                |  SYNSENT  |
+--------------------+----------------------------------------+-----------+
|       SYN          |  Update Peer Verifier; Send SYNACK     |  SYNRCVD  |
+--------------------+----------------------------------------+-----------+
|       ACK          |             Send RSTACK                |  SYNSENT  |
+==============================================================================+
```

State: SYNRCVD

```
+======================================================================+
|     Condition      |               Action              | New State |
+====================+===================================+===========+
|   SYNACK && C      | Update Peer Verifier; Send ACK    |   ESTAB   |
+--------------------+-----------------------------------+-----------+
|   SYNACK && !C     |           Send RSTACK             |  SYNRCVD  |
+--------------------+-----------------------------------+-----------+
|        SYN         | Update Peer Verifier; Send SYNACK |  SYNRCVD  |
+--------------------+-----------------------------------+-----------+
|   ACK && B && C    |           Send ACK                |   ESTAB   |
+--------------------+-----------------------------------+-----------+
|  ACK && !(B && C)  |           Send RSTACK             |  SYNRCVD  |
+======================================================================+
```

State: ESTAB

```
+======================================================================+
|     Condition      |               Action              | New State |
+====================+===================================+===========+
|   SYN || SYNACK    |         Send ACK (note 1)         |   ESTAB   |
+--------------------+-----------------------------------+-----------+
|   ACK && B && C    |         Send ACK (note 1)         |   ESTAB   |
+--------------------+-----------------------------------+-----------+
|  ACK && !(B && C)  |           Send RSTACK             |   ESTAB   |
+======================================================================+
```

   Note 1: No more than one ACK should be sent within any time period of
   length defined by the timer.

9. Failure Response Messages

   A failure response message is formed by returning the request message
   that caused the failure with the Result field in the header
   indicating failure (Result = 4) and the Code field giving the failure
   code. The failure code specifies the reason for the switch being
   unable to satisfy the request message. A failure code of 16 is used
   for a failure that is specific to the particular request message and
   its meaning is defined within the text describing that message. The
   following failure codes are defined:

      1: Unspecified reason not covered by other failure codes.
      2: Invalid request message.
      3: The specified request is not implemented on this switch.
      4: Invalid port session number.
      5: One or more of the specified ports does not exist.

6: One or more of the specified ports is down.
7: One or more of the specified VPIs or VCIs is out of range on
    one or more of the requested ports.
8: The specified connection does not exist.
9: The specified branch does not exist.
10: A branch belonging to the specified multicast connection is
    already established on the specified output port and the
    switch cannot support more than a single branch of any
    multicast connection on the same output port.
11: The limit on the maximum number of multicast connections that
    the switch can support has been reached.
12: The limit on the maximum number of branches that the
    specified multicast connection can support has been reached.
13: Unable to assign the requested VPI/VCI value to the requested
    branch on the specified multicast connection.
14: General problem related to the manner in which multicast is
    supported by the switch.
15: Out of resources (e.g. memory exhausted, etc.).
16: Failure specific to the particular message type.

REFERENCES

   [I.361]   "B-ISDN ATM Layer Specification," International
              Telecommunication Union, ITU-T Recommendation I.361, Mar.
              1993.

   [I.363]   "B-ISDN ATM Adaptation Layer (AAL) Specification,"
              International Telecommunication Union, ITU-T Recommendation
              I.363, Mar. 1993.

   [rfc1700] "Assigned Numbers," STD 2, RFC 1700, October 1994.

   [rfc1573] "Evolution of the Interfaces Group of MIB-II," RFC 1573,
              January 1994.

SECURITY CONSIDERATIONS

   Security issues are not discussed in this document.

AUTHORS' ADDRESSES

   Peter Newman   Phone: +1 (415) 846-4603
   Ipsilon Networks, Inc.   Email: pn@ipsilon.com

   W. L. Edwards, Chief Scientist   Phone: +1 (913) 534 5334
   Sprint   Email: texas@sprintcorp.com

   Robert M. Hinden   Phone: +1 (415) 846-4604
   Ipsilon Networks, Inc.   Email: hinden@ipsilon.com

   Eric Hoffman   Phone: +1 (415) 846-4610
   Ipsilon Networks, Inc.   Email: hoffman@ipsilon.com

   Fong Ching Liaw   Phone: +1 (415) 846-4607
   Ipsilon Networks, Inc.   Email: fong@ipsilon.com

   Tom Lyon   Phone: +1 (415) 846-4601
   Ipsilon Networks, Inc.   Email: pugs@ipsilon.com

   Greg Minshall   Phone: +1 (415) 846-4605
   Ipsilon Networks, Inc.   Email: minshall@ipsilon.com

Ipsilon Networks, Inc. is located at:

    2191 East Bayshore Road
    Suite 100
    Palo Alto, CA 94303
    USA

Sprint is located at:

    Sprint
    Sprint Technology Services - Long Distance Division
    9300 Metcalf Avenue
    Mailstop KSOPKB0802
    Overland Park, KS 66212-6333
    USA