1. Status of this Memo

This document is an Internet Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are draft documents valid for a maximum of six months. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress."

Please check the I-D abstract listing contained in each Internet Draft directory to learn the current status of this or any other Internet Draft.
2. Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it defines objects for managing the Resource Reservation Protocol (RSVP) within the interface attributes defined in the Integrated Services Model. Thus, the Integrated Services MIB is directly relevant to and cross-referenced by this MIB. Comments should be made to the RSVP Working Group, rsvp@isi.edu.

This memo does not, in its draft form, specify a standard for the Internet community.
3. The SNMPv2 Network Management Framework

The SNMPv2 Network Management Framework consists of four major components. They are:

- **RFC 1441** which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management.

- **RFC 1213** defines MIB-II, the core set of managed objects for the Internet suite of protocols.

- **RFC 1445** which defines the administrative and other architectural aspects of the framework.

- **RFC 1448** which defines the protocol used for network access to managed objects.

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

3.1. Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, each object type is named by an OBJECT IDENTIFIER, an administratively assigned name. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the descriptor, to refer to the object type.

4. Overview

4.1. Textual Conventions

Several new data types are introduced as a textual convention in this MIB document. These textual conventions enhance the readability of the specification and can ease comparison with other specifications if appropriate. It should be noted that the introduction of these textual conventions has no effect on either the syntax nor the semantics of any managed
objects. The use of these is merely an artifact of the explanatory method used. Objects defined in terms of one of these methods are always encoded by means of the rules that define the primitive type. Hence, no changes to the SMI or the SNMP are necessary to accommodate these textual conventions which are adopted merely for the convenience of readers and writers in pursuit of the elusive goal of clear, concise, and unambiguous MIB documents.

4.2. Structure of MIB

The MIB is composed of the following sections:
   General Objects
   Session Statistics Table
   Session Sender Table
   Reservation Requests Received Table
   Reservation Requests Forwarded Table
   RSVP Active Flows Table
   RSVP Interface Attributes Table
   RSVP Neighbor Table

As a general rule, it is difficult in SNMP to describe arbitrarily long or complex messages; this MIB therefore seeks to describe the Path State Database and the Reservation State Database as though each flow and filter description received in an aggregate message had been received in a separate reservation message.

Thus, if a RESV message is received for session 224.1.2.3+UDP+4455 with two filter/flow spec groups describing a sender 1.2.3.4 and another sender 1.2.7.8, these two will show in the MIB as two separate rows: one for 224.1.2.3+UDP+4455 from 1.2.3.4 and the other for 224.1.2.3+UDP+4455 from 1.2.7.8.
5. Definitions

RSVP-MIB DEFINITIONS ::= BEGIN

IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, Gauge32, NOTIFICATION-TYPE,
Integer32, experimental FROM SNMPv2-SMI

TEXTUAL-CONVENTION, TruthValue, RowStatus, TimeStamp,
TestAndIncr, TimeInterval FROM SNMPv2-TC

MODULE-COMPLIANCE, OBJECT-GROUP FROM SNMPv2-CONF
BitRate, BurstSize FROM INTEGRATED-SERVICES-MIB
ifIndex, InterfaceIndex FROM IF-MIB;

rsvp MODULE-IDENTITY
LAST-UPDATED "9511030500Z" -- Fri Jun 14 15:30:52 PDT 1996
ORGANIZATION "IETF RSVP Working Group"
CONTACT-INFO
"      Fred Baker
Postal: Cisco Systems
      519 Lado Drive
      Santa Barbara, California 93111
Tel:    +1 805 681 0115
E-Mail: fred@cisco.com

John Krawczyk
Postal: Bay Networks, Inc
      2 Federal Street
      Billerica, Massachusetts 01821
Tel:    +1 508 436 3811
E-Mail: jj@BayNetworks.com"
DESCRIPTION
"The MIB module to describe the RSVP and In-
tegrated Services Protocol"
::= { experimental 71 }
rsvpObjects OBJECT IDENTIFIER ::= { rsvp 1 } -- tables
rsvpGenObjects OBJECT IDENTIFIER ::= { rsvp 2 } -- global objects
rsvpNotifications OBJECT IDENTIFIER ::= { rsvp 3 } -- traps
rsvpConformance OBJECT IDENTIFIER ::= { rsvp 4 } -- conformance
SessionNumber ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"The Session Number convention is used for numbers identifying sessions or saved PATH or RESV information. It is a number in the range returned by a TestAndIncr variable, having no protocol meaning whatsoever but serving instead as simple identifier.

The alternative was a very complex instance or instance object that became unwieldy."
SYNTAX INTEGER (0..2147483647)

Protocol ::= TEXTUAL-CONVENTION
DISPLAY-HINT "d"
STATUS current
DESCRIPTION
"The value of the IP Protocol field of an IP Datagram Header. This identifies the protocol layer above IP. For example, the value 6 is used for TCP and the value 17 is used for UDP. The values of this field are defined in the Assigned Numbers RFC."
SYNTAX INTEGER (1..255)

SessionType ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"The value of the C-Type field of a Session object, as defined in the RSVP specification. This value determines the lengths of octet strings and use of certain objects such as the 'port' variables. If the C-Type calls for an IP6 address, one would expect all source, destination, and next/previous hop addresses to be 16 bytes long, and for the ports to be UDP/TCP port numbers, for example."
SYNTAX INTEGER (1..255)
Port ::= TEXTUAL-CONVENTION
   DISPLAY-HINT "d"
   STATUS current
   DESCRIPTION
   "The value of the UDP or TCP Source or Destination Port field, IPSEC SPI, or other session discriminator. If it is not used, the value should be of length 0. This pair, when coupled with the IP Addresses of the source and destination system and the IP protocol field, uniquely identifies a data stream."
   SYNTAX   OCTET STRING

RsvpEncapsulation ::= TEXTUAL-CONVENTION
   STATUS current
   DESCRIPTION
   "This indicates the encapsulation that an RSVP Neighbor is perceived to be using."
   SYNTAX   INTEGER {
               ip (1),  -- IP Protocol 46
               udp (2), -- UDP Encapsulation
               both (3) -- neighbor is using both encapsulations
              }

RefreshInterval ::= TEXTUAL-CONVENTION
   DISPLAY-HINT "d"
   STATUS current
   DESCRIPTION
   "The number of milliseconds that are expected to elapse between refreshes of path or reservation state. Unrefreshed Path or reservation state is removed after a small multiple of this period."
   SYNTAX   INTEGER (0..’7FFFFFFF’h)
QosService ::= TEXTUAL-CONVENTION
  STATUS current
  DESCRIPTION
  "The class of service in use by a flow."
  SYNTAX INTEGER {
    controlledDelay (1), -- Controlled Delay
    guaranteedDelay (2), -- Guaranteed Delay
    predictiveDelay (3), -- Predictive Delay
    controlledLoad (5)   -- Controlled Load
  }

DelayClass ::= TEXTUAL-CONVENTION
  DISPLAY-HINT "d"
  STATUS current
  DESCRIPTION
  "The class of delay used by the Controlled Delay Service."
  SYNTAX INTEGER (1..3)

MessageSize ::= TEXTUAL-CONVENTION
  DISPLAY-HINT "d"
  STATUS current
  DESCRIPTION
  "The size of a message in bytes. This is used to specify the minimum and maximum size of a message along an integrated services route."
  SYNTAX INTEGER (0..'7FFFFFFF'h)
The RSVP Session Statistics Database displays statistics relating to the number of senders and receivers in each session.

rsvpSessionNewIndex OBJECT-TYPE
SYNTAX TestAndIncr
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"This object is used to assign values to rsvpSessionNumber as described in ‘Textual Conventions for SNMPv2’. The network manager reads the object, and then writes the value back in the SET that creates a new instance of rsvpSessionEntry. If the SET fails with the code ‘inconsistentValue’, then the process must be repeated; If the SET succeeds, then the object is incremented, and the new instance is created according to the manager’s directions."
::= { rsvpGenObjects 1 }

rsvpSessionTable OBJECT-TYPE
SYNTAX SEQUENCE OF RsvpSessionEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Statistics concerning the sessions seen by a given system."
::= { rsvpObjects 1 }

rsvpSessionEntry OBJECT-TYPE
SYNTAX RsvpSessionEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Statistics concerning a single RSVP session seen by a given system."
INDEX { rsvpSessionNumber }
::= { rsvpSessionTable 1 }
RsvpSessionEntry ::=  
SEQUENCE {
  rsvpSessionNumber           SessionNumber,  
  rsvpSessionType             SessionType,    
  rsvpSessionDestAddr         OCTET STRING,   
  rsvpSessionProtocol         Protocol,       
  rsvpSessionPort             Port,          
  rsvpSessionSenders          Gauge32,        
  rsvpSessionReceivers        Gauge32,        
  rsvpSessionRequests         Gauge32
}

rsvpSessionNumber OBJECT-TYPE
SYNTAX      SessionNumber
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  
"The number of this session. This is for SNMP
Indexing purposes only and has no relation to
any protocol value."
::= { rsvpSessionEntry 1 }

rsvpSessionType OBJECT-TYPE
SYNTAX      SessionType
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The type of session (IP4, IP6, IP6 with flow
information, etc), and therefore the structure
of the information."
::= { rsvpSessionEntry 2 }

rsvpSessionDestAddr OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The Destination Address used used by for all
senders in this session."
::= { rsvpSessionEntry 3 }
rsvpSessionProtocol OBJECT-TYPE
SYNTAX Protocol
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The IP Protocol used by a session."
::= { rsvpSessionEntry 4 }

rsvpSessionPort OBJECT-TYPE
SYNTAX Port
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The UDP or TCP port number used as a destination port for all senders in this session. A value of zero indicates that the IP protocol in use, specified by rsvpSessionProtocol, does not have ports."
::= { rsvpSessionEntry 5 }

rsvpSessionSenders OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of senders currently known to be part of this session."
::= { rsvpSessionEntry 6 }

rsvpSessionReceivers OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of reservations being requested of this system for this session."
::= { rsvpSessionEntry 7 }
rsvpSessionRequests OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of reservation requests this system is sending upstream for this session."
::= { rsvpSessionEntry 8 }

-- Note that this is a read only table. If the corresponding
-- rsvpSenderTable, rsvpResvTable, and rsvpReqTable entries all
-- are removed, this entry goes away also.
The RSVP Session Sender Database contains the information displayed by senders regarding their potential contribution to session data content. It is in essence a list of the valid PATH messages that the RSVP Router or Host is receiving.

```
rsvpSenderNewIndex OBJECT-TYPE
SYNTAX      TestAndIncr
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
"This object is used to assign values to rsvpSenderNumber as described in 'Textual Conventions for SNMPv2'. The network manager reads the object, and then writes the value back in the SET that creates a new instance of rsvpSenderEntry. If the SET fails with the code 'inconsistentValue', then the process must be repeated; If the SET succeeds, then the object is incremented, and the new instance is created according to the manager’s directions."
::= { rsvpGenObjects 2 }

rsvpSenderTable OBJECT-TYPE
SYNTAX      SEQUENCE OF RsvpSenderEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"Information describing the state information displayed by senders in PATH messages."
::= { rsvpObjects 2 }

rsvpSenderEntry OBJECT-TYPE
SYNTAX      RsvpSenderEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"Information describing the state information displayed by a single sender’s PATH message."
INDEX { rsvpSessionNumber, rsvpSenderNumber }
::= { rsvpSenderTable 1 }

RsvpSenderEntry ::=
SEQUENCE {
  rsvpSenderNumber  SessionNumber,
  rsvpSenderType    SessionType,
  rsvpSenderDestAddr OCTET STRING,
  rsvpSenderSenderAddr OCTET STRING,
  rsvpSenderProtocol Protocol,
  rsvpSenderDestPort Port,
  rsvpSenderPort    Port,
  rsvpSenderHopAddr OCTET STRING,
  rsvpSenderInterface InterfaceIndex,
  rsvpSenderTSpecRate BitRate,
  rsvpSenderTSpecPeakRate BitRate,
  rsvpSenderTSpecBurst BurstSize,
  rsvpSenderTSpecMinTU MessageSize,
  rsvpSenderTSpecMaxTU MessageSize,
  rsvpSenderInterval RefreshInterval,
  rsvpSenderRSVPHop TruthValue,
  rsvpSenderLastChange TimeStamp,
  rsvpSenderPolicy OCTET STRING,
  rsvpSenderAdspec OCTET STRING,
  rsvpSenderStatus RowStatus
}

rsvpSenderNumber OBJECT-TYPE
SYNTAX      SessionNumber
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  
"The number of this session. This is for SNMP Indexing purposes only and has no relation to any protocol value."
::= { rsvpSenderEntry 1 }

rsvpSenderType OBJECT-TYPE
SYNTAX      SessionType
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The type of session (IP4, IP6, IP6 with flow information, etc), and therefore the structure of the information."
::= { rsvpSenderEntry 2 }
rsvpSenderDestAddr OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The Address used as a destination address by all senders in this session."
::= { rsvpSenderEntry 3 }

rsvpSenderSenderAddr OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The Address used as a source address by this sender in this session."
::= { rsvpSenderEntry 4 }

rsvpSenderProtocol OBJECT-TYPE
SYNTAX Protocol
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The IP Protocol used by a session. For secure sessions, this indicates IP Security."
::= { rsvpSenderEntry 5 }

rsvpSenderDestPort OBJECT-TYPE
SYNTAX Port
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The UDP or TCP port number used as a destination port for all senders in this session. A value of zero indicates that the IP protocol in use, specified by rsvpSessionProtocol, does not have ports."
::= { rsvpSenderEntry 6 }
rsvpSenderPort OBJECT-TYPE
SYNTAX         Port
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION    "The UDP or TCP port number used as a source port for this sender in this session. A value of zero indicates that the IP protocol in use, specified by rsvpSessionProtocol, does not have ports."
::= { rsvpSenderEntry 7 }

rsvpSenderHopAddr OBJECT-TYPE
SYNTAX         OCTET STRING
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION    "The address used by the previous RSVP hop (which may be the original sender)."
::= { rsvpSenderEntry 8 }

rsvpSenderInterface OBJECT-TYPE
SYNTAX         InterfaceIndex
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION    "The ifIndex value of the interface on which this PATH message was most recently received."
::= { rsvpSenderEntry 9 }
rsvpSenderTSpecRate OBJECT-TYPE
SYNTAX      BitRate
UNITS       "kilobits per second"
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
 "The Average Bit Rate of the sender’s data stream, in kilobits/second. Within a transmission burst, the arrival rate may be as fast as rsvpSenderTSpecPeakRate (if supported by the service model); however, averaged across two or more burst intervals, the rate should not exceed rsvpSenderTSpecRate.

Note that this is a prediction, often based on the general capability of a type of codec or particular encoding; the measured average rate may be significantly lower."
 ::= { rsvpSenderEntry 10 }

rsvpSenderTSpecPeakRate OBJECT-TYPE
SYNTAX      BitRate
UNITS       "kilobits per second"
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
 "The Peak Bit Rate of the sender’s data stream, in kilobits/second. Traffic arrival is not expected to exceed this rate at any time, apart from the effects of jitter in the network. If not specified in the TSpec, this returns zero or noSuchValue."
 ::= { rsvpSenderEntry 11 }
rsvpSenderTSpecBurst OBJECT-TYPE
SYNTAX BurstSize
UNITS "bits"
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The size of the largest burst expected from the sender at a time."
::= { rsvpSenderEntry 12 }

rsvpSenderTSpecMinTU OBJECT-TYPE
SYNTAX MessageSize
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The minimum message size for this flow. The policing algorithm will treat smaller messages as though they are this size."
::= { rsvpSenderEntry 13 }

rsvpSenderTSpecMaxTU OBJECT-TYPE
SYNTAX MessageSize
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The maximum message size for this flow. The admission algorithm will reject TSpecs whose Maximum Transmission Unit, plus the interface headers, exceed the interface MTU."
::= { rsvpSenderEntry 14 }

rsvpSenderInterval OBJECT-TYPE
SYNTAX RefreshInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The interval between PATH receipt of refresh messages as advertised by the Previous Hop."
::= { rsvpSenderEntry 15 }
rsvpSenderRSVPHop OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"If TRUE, the node believes that the previous hop is an RSVP hop. If FALSE, the node believes that the previous hop may not be an RSVP hop."
 ::= { rsvpSenderEntry 16 }

rsvpSenderLastChange OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The time of the last change in this PATH message; This includes the first time it was sent, or time of the most recent change in parameters."
 ::= { rsvpSenderEntry 17 }

rsvpSenderPolicy OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The contents of the policy object, displayed as an uninterpreted string of octets, including the object header. In the absence of such an object, this should be of zero length."
 ::= { rsvpSenderEntry 18 }
rsvpSenderAdspec OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The contents of the Advertising Specification object, displayed as an uninterpreted string of octets, including the object header. In the absence of such an object, this should be of zero length."
 ::= { rsvpSenderEntry 19 }

rsvpSenderStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"'valid' for all active PATH messages. This object may be used to install static PATH information or delete PATH information."
 ::= { rsvpSenderEntry 20 }
The RSVP Reservation Requests Received Table contains the information displayed by receivers regarding their needs with respect to sessions and senders. It is in essence a list of the valid RESV messages that the RSVP Router or Host is receiving.

rsvpResvNewIndex OBJECT-TYPE
SYNTAX      TestAndIncr
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
"This object is used to assign values to rsvpResvNumber as described in ‘Textual Conven-
tions for SNMPv2’. The network manager reads the object, and then writes the value back in
the SET that creates a new instance of rsvpResvEntry. If the SET fails with the code
'inconsistentValue’, then the process must be repeated; If the SET succeeds, then the object
is incremented, and the new instance is created according to the manager’s directions."
 ::= { rsvpGenObjects 3 }

rsvpResvTable OBJECT-TYPE
SYNTAX      SEQUENCE OF RsvpResvEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"Information describing the state information displayed by receivers in RESV messages."
 ::= { rsvpObjects 3 }

rsvpResvEntry OBJECT-TYPE
SYNTAX      RsvpResvEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
"Information describing the state information displayed by a single receiver’s RESV message
concerning a single sender."
INDEX { rsvpSessionNumber, rsvpResvNumber }
 ::= { rsvpResvTable 1 }
RsvpResvEntry ::= 
SEQUENCE {
    rsvpResvNumber SessionNumber, 
rsvpResvType SessionType, 
rsvpResvDestAddr OCTET STRING, 
rsvpResvSenderAddr OCTET STRING, 
rsvpResvProtocol Protocol, 
rsvpResvDestPort Port, 
rsvpResvPort Port, 
rsvpResvHopAddr OCTET STRING, 
rsvpResvInterface InterfaceIndex, 
rsvpResvService QosService, 
rsvpResvTSpecRate BitRate, 
rsvpResvTSpecPeakRate BitRate, 
rsvpResvTSpecBurst BurstSize, 
rsvpResvTSpecLevel DelayClass, 
rsvpResvTSpecMinTU MessageSize, 
rsvpResvTSpecMaxTU MessageSize, 
rsvpResvInterval RefreshInterval, 
rsvpResvScope OCTET STRING, 
rsvpResvShared TruthValue, 
rsvpResvExplicit TruthValue, 
rsvpResvRSVPHop TruthValue, 
rsvpResvLastChange TimeStamp, 
rsvpResvPolicy OCTET STRING, 
rsvpResvStatus RowStatus
}

rsvpResvNumber OBJECT-TYPE
SYNTAX SessionNumber
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"The number of this session.  This is for SNMP
Indexing purposes only and has no relation to
any protocol value."
::= { rsvpResvEntry 1 }
rsvpResvType OBJECT-TYPE
SYNTAX      SessionType
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The type of session (IP4, IP6, IP6 with flow information, etc), and therefore the structure of the information."
 ::= { rsvpResvEntry 2 }

rsvpResvDestAddr OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The Address used as a destination address by all senders in this session."
 ::= { rsvpResvEntry 3 }

rsvpResvSenderAddr OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The Address used as a source address by this sender in this session. The value of all zeroes indicates ‘any sender’."
 ::= { rsvpResvEntry 4 }

rsvpResvProtocol OBJECT-TYPE
SYNTAX      Protocol
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The IP Protocol used by a session. for secure sessions, this indicates IP Security."
 ::= { rsvpResvEntry 5 }
rsvpResvDestPort OBJECT-TYPE
SYNTAX Port
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The UDP or TCP port number used as a destination port for all senders in this session. A value of zero indicates that the IP protocol in use, specified by rsvpSessionProtocol, does not have ports."
::= { rsvpResvEntry 6 }

rsvpResvPort OBJECT-TYPE
SYNTAX Port
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The UDP or TCP port number used as a source port for this sender in this session. A value of zero indicates that the IP protocol in use, specified by rsvpSessionProtocol, does not have ports or the port is being ignored."
::= { rsvpResvEntry 7 }

rsvpResvHopAddr OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The address used by the next RSVP hop (which may be the ultimate receiver)."
::= { rsvpResvEntry 8 }

rsvpResvInterface OBJECT-TYPE
SYNTAX InterfaceIndex
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The ifIndex value of the interface on which this RESV message was most recently received."
::= { rsvpResvEntry 9 }
rsvpResvService OBJECT-TYPE
SYNTAX QosService
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The QoS Service classification requested by the receiver."
::= { rsvpResvEntry 10 }

rsvpResvTSpecRate OBJECT-TYPE
SYNTAX BitRate
UNITS "kilobits per second"
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Average Bit Rate of the sender’s data stream, in kilobits/second. Within a transmission burst, the arrival rate may be as fast as rsvpResvTSpecPeakRate (if supported by the service model); however, averaged across two or more burst intervals, the rate should not exceed rsvpResvTSpecRate.

Note that this is a prediction, often based on the general capability of a type of codec or particular encoding; the measured average rate may be significantly lower."
::= { rsvpResvEntry 11 }

rsvpResvTSpecPeakRate OBJECT-TYPE
SYNTAX BitRate
UNITS "kilobits per second"
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Peak Bit Rate of the sender’s data stream, in kilobits/second. Traffic arrival is not expected to exceed this rate at any time, apart from the effects of jitter in the network. If not specified in the TSpec, this returns zero or noSuchValue."
::= { rsvpResvEntry 12 }
rsvpResvTSpecBurst OBJECT-TYPE
SYNTAX     BurstSize
UNITS      "bits"
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
 "The size of the largest burst expected from
the sender at a time.

If this is less than the sender's advertised
burst size, the receiver is asking the network
to provide flow pacing beyond what would be
provided under normal circumstances. Such pac-
ing is at the network's option."
 ::= { rsvpResvEntry 13 }

rsvpResvTSpecLevel OBJECT-TYPE
SYNTAX     DelayClass
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
 "If the service is predictive or controlled-
delay, this is the service level that is being
requested. Otherwise, it is zero, or the agent
may return noSuchValue."
 ::= { rsvpResvEntry 14 }

rsvpResvTSpecMinTU OBJECT-TYPE
SYNTAX     MessageSize
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
 "The minimum message size for this flow. The
policing algorithm will treat smaller messages
as though they are this size."
 ::= { rsvpResvEntry 15 }
rsvpResvTSpecMaxTU OBJECT-TYPE
SYNTAX     MessageSize
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
"The maximum message size for this flow. The admission algorithm will reject TSpecs whose Maximum Transmission Unit, plus the interface headers, exceed the interface MTU."
::= { rsvpResvEntry 16 }

rsvpResvInterval OBJECT-TYPE
SYNTAX     RefreshInterval
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
"The interval between RESV receipt of refresh messages as advertised by the Next Hop."
::= { rsvpResvEntry 17 }

rsvpResvScope OBJECT-TYPE
SYNTAX     OCTET STRING
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
"The contents of the scope object, displayed as an uninterpreted string of octets, including the object header. In the absence of such an object, this should be of zero length."
::= { rsvpResvEntry 18 }

rsvpResvShared OBJECT-TYPE
SYNTAX     TruthValue
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
"If TRUE, a reservation shared among senders is requested. If FALSE, a reservation specific to this sender is requested."
::= { rsvpResvEntry 19 }

Baker and Krawczyk  Expires September 1996
rsprevExplicit OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"If TRUE, individual senders are listed using
Filter Specifications. If FALSE, senders are
listed in the Scope Object."
::= { rsprevEntry 20 }

rsprevRSVPHop OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"If TRUE, the node believes that the next hop
is an RSVP hop. If FALSE, the node believes
that the next hop may not be an RSVP hop."
::= { rsprevEntry 21 }

rsprevLastChange OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  
"The time of the last change in this reserva-
tion request; This includes the first time it
was received, or time of the most recent change
in parameters."
::= { rsprevEntry 22 }

rsprevPolicy OBJECT-TYPE
SYNTAX      OCTET STRING
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION  
"The contents of the policy object, displayed
as an uninterpreted string of octets, including
the object header. In the absence of such an
object, this should be of zero length."
::= { rsprevEntry 23 }
rsvpResvStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION "'valid' for all active RESV messages. This object may be used to install static RESV information or delete RESV information."
::= { rsvpResvEntry 24 }
The RSVP Reservation Requests Forwarded Table contains the information displayed by receivers regarding their needs with respect to sessions and senders. It is in essence a list of the valid RESV messages that the RSVP Router or Host is sending to its upstream neighbors.

rsvpResvFwdNewIndex OBJECT-TYPE
SYNTAX TestAndIncr
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This object is used to assign values to rsvpResvFwdNumber as described in ’Textual Conventions for SNMPv2’. The network manager reads the object, and then writes the value back in the SET that creates a new instance of rsvpResvFwdEntry. If the SET fails with the code ’inconsistentValue’, then the process must be repeated; If the SET succeeds, then the object is incremented, and the new instance is created according to the manager’s directions."
::= { rsvpGenObjects 4 }

rsvpResvFwdTable OBJECT-TYPE
SYNTAX SEQUENCE OF RsvpResvFwdEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "Information describing the state information displayed upstream in RESV messages."
::= { rsvpObjects 4 }
rsvpResvFwdEntry OBJECT-TYPE
SYNTAX RsvpResvFwdEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "Information describing the state information displayed upstream in an RESV message concerning a single sender."
INDEX { rsvpSessionNumber, rsvpResvFwdNumber }
::= { rsvpResvFwdTable 1 }

RsvpResvFwdEntry ::= SEQUENCE {
    rsvpResvFwdNumber               SessionNumber,
    rsvpResvFwdType                 SessionType,
    rsvpResvFwdDestAddr             OCTET STRING,
    rsvpResvFwdSenderAddr           OCTET STRING,
    rsvpResvFwdProtocol             Protocol,
    rsvpResvFwdDestPort             Port,
    rsvpResvFwdPort                 Port,
    rsvpResvFwdInterface            InterfaceIndex,
    rsvpResvFwdService              QosService,
    rsvpResvFwdTSpecRate            BitRate,
    rsvpResvFwdTSpecPeakRate        BitRate,
    rsvpResvFwdTSpecBurst           BurstSize,
    rsvpResvFwdTSpecLevel           DelayClass,
    rsvpResvFwdTSpecMinTU           MessageSize,
    rsvpResvFwdTSpecMaxTU           MessageSize,
    rsvpResvFwdInterval             RefreshInterval,
    rsvpResvFwdScope                OCTET STRING,
    rsvpResvFwdShared               TruthValue,
    rsvpResvFwdExplicit             TruthValue,
    rsvpResvFwdRSVPHop              TruthValue,
    rsvpResvFwdLastChange           TimeStamp,
    rsvpResvFwdPolicy               OCTET STRING,
    rsvpResvFwdStatus               RowStatus
}
rsvpResvFwdNumber OBJECT-TYPE
SYNTAX SessionNumber
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"The number of this session. This is for SNMP
Indexing purposes only and has no relation to
any protocol value."
 ::= { rsvpResvFwdEntry 1 }

rsvpResvFwdType OBJECT-TYPE
SYNTAX SessionType
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The type of session (IP4, IP6, IP6 with flow
information, etc), and therefore the structure
of the information."
 ::= { rsvpResvFwdEntry 2 }

rsvpResvFwdDestAddr OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Address used as a destination address by
all senders in this session."
 ::= { rsvpResvFwdEntry 3 }

rsvpResvFwdSenderAddr OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Address used as a source address by this
sender in this session. The value 0 designates
'any sender'."
 ::= { rsvpResvFwdEntry 4 }
Internet Draft             RSVP MIB                 March 1996

rsvpResvFwdProtocol OBJECT-TYPE
SYNTAX      Protocol
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
  "The IP Protocol used by a session. For secure
  sessions, this indicates IP Security."
 ::= { rsvpResvFwdEntry 5 }

rsvpResvFwdDestPort OBJECT-TYPE
SYNTAX      Port
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
  "The UDP or TCP port number used as a destination port for all senders in this session. A value of zero indicates that the IP protocol in use, specified by rsvpSessionProtocol, does not have ports."
 ::= { rsvpResvFwdEntry 6 }

rsvpResvFwdPort OBJECT-TYPE
SYNTAX      Port
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
  "The UDP or TCP port number used as a source port for this sender in this session. A value of zero indicates that the IP protocol in use, specified by rsvpSessionProtocol, does not have ports."
 ::= { rsvpResvFwdEntry 7 }

rsvpResvFwdInterface OBJECT-TYPE
SYNTAX      InterfaceIndex
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
  "The ifIndex value of the interface on which this RESV message was most recently sent."
 ::= { rsvpResvFwdEntry 8 }
rsvpResvFwdService OBJECT-TYPE
SYNTAX QosService
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The QoS Service classification requested."
::= { rsvpResvFwdEntry 9 }

rsvpResvFwdTSpecRate OBJECT-TYPE
SYNTAX BitRate
UNITS "kilobits per second"
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Average Bit Rate of the sender’s data
stream, in kilobits/second. Within a transmis-
sion burst, the arrival rate may be as fast as
rsvpResvFwdTSpecPeakRate (if supported by the
service model); however, averaged across two or
more burst intervals, the rate should not
exceed rsvpResvFwdTSpecRate.

Note that this is a prediction, often based on
the general capability of a type of codec or
particular encoding; the measured average rate
may be significantly lower."
::= { rsvpResvFwdEntry 10 }

rsvpResvFwdTSpecPeakRate OBJECT-TYPE
SYNTAX BitRate
UNITS "kilobits per second"
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Peak Bit Rate of the sender’s data stream,
in kilobits/second. Traffic arrival is not ex-
pected to exceed this rate at any time, apart
from the effects of jitter in the network. If
not specified in the TSpec, this returns zero
or noSuchValue."
::= { rsvpResvFwdEntry 11 }
rsvpResvFwdTSpecBurst OBJECT-TYPE
SYNTAX     BurstSize
UNITS      "bits"
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
 "The size of the largest burst expected from
 the sender at a time.

 If this is less than the sender’s advertised
 burst size, the receiver is asking the network
to provide flow pacing beyond what would be
provided under normal circumstances. Such pac-
ing is at the network’s option."
 ::= { rsvpResvFwdEntry 12 }

rsvpResvFwdTSpecLevel OBJECT-TYPE
SYNTAX     DelayClass
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
 "If the service is predictive or controlled-
delay, this is the service level that is being
requested. Otherwise, it is zero, or the agent
may return noSuchValue."
 ::= { rsvpResvFwdEntry 13 }

rsvpResvFwdTSpecMinTU OBJECT-TYPE
SYNTAX     MessageSize
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
 "The minimum message size for this flow. The
 policing algorithm will treat smaller messages
 as though they are this size."
 ::= { rsvpResvFwdEntry 14 }
rsvpResvFwdTSpecMaxTU OBJECT-TYPE
SYNTAX MessageSize
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The maximum message size for this flow. The admission algorithm will reject TSpecs whose Maximum Transmission Unit, plus the interface headers, exceed the interface MTU."
::= { rsvpResvFwdEntry 15 }

rsvpResvFwdInterval OBJECT-TYPE
SYNTAX RefreshInterval
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The interval between RESV receipt of refresh messages advertised to the Previous Hop."
::= { rsvpResvFwdEntry 16 }

rsvpResvFwdScope OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The contents of the scope object, displayed as an uninterpreted string of octets, including the object header. In the absence of such an object, this should be of zero length."
::= { rsvpResvFwdEntry 17 }

rsvpResvFwdShared OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"If TRUE, a reservation shared among senders is requested. If FALSE, a reservation specific to this sender is requested."
::= { rsvpResvFwdEntry 18 }
rsvpResvFwdExplicit OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"If TRUE, individual senders are listed using Filter Specifications. If FALSE, senders are listed in the Scope Object."
::= { rsvpResvFwdEntry 19 }

rsvpResvFwdRSVPHop OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"If TRUE, the node believes that the next hop is an RSVP hop. If FALSE, the node believes that the next hop may not be an RSVP hop."
::= { rsvpResvFwdEntry 20 }

rsvpResvFwdLastChange OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The time of the last change in this request; This includes the first time it was requested, or time of the most recent change in parameters requested."
::= { rsvpResvFwdEntry 21 }

rsvpResvFwdPolicy OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The contents of the policy object, displayed as an uninterpreted string of octets, including the object header. In the absence of such an object, this should be of zero length."
::= { rsvpResvFwdEntry 22 }
rsvpResvFwdStatus OBJECT-TYPE
SYNTAX     RowStatus
MAX-ACCESS read-create
STATUS     current
DESCRIPTION
    "'valid' for all active RESV messages. This
    object may be used to install static RESV in-
    formation or delete RESV information."
 ::= { rsvpResvFwdEntry 23 }
-- The RSVP Active Flows Database
-- lists all flows active on an outgoing interface, including
-- relevant attributes.

rsvpFlowTable OBJECT-TYPE
SYNTAX      SEQUENCE OF RsvpFlowEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
   "Information describing the reserved flows using the system’s interfaces."
::= { rsvpObjects 5 }

rsvpFlowEntry OBJECT-TYPE
SYNTAX      RsvpFlowEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
   "Information describing the use of a given interface by a given flow."
INDEX { rsvpFlowNumber }
::= { rsvpFlowTable 1 }

RsvpFlowEntry ::= SEQUENCE {
  rsvpFlowNumber                  SessionNumber,
  rsvpFlowType                    SessionType,
  rsvpFlowDestAddr                OCTET STRING,
  rsvpFlowSenderAddr              OCTET STRING,
  rsvpFlowProtocol                Protocol,
  rsvpFlowDestPort                Port,
  rsvpFlowPort                    Port,
  rsvpFlowInterface               InterfaceIndex,
  rsvpFlowRate                    BitRate,
  rsvpFlowBurst                   BurstSize,
  rsvpFlowWeight                  Integer32,
  rsvpFlowQueue                   Integer32,
  rsvpFlowMinTU                   MessageSize,
  rsvpFlowDontAsk                 TimeInterval,
  rsvpFlowStatus                  RowStatus
}
rsvpFlowNumber OBJECT-TYPE
SYNTAX SessionNumber
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"The number of this session. This is for SNMP
Indexing purposes only and has no relation to
any protocol value."
::= { rsvpFlowEntry 1 }

rsvpFlowType OBJECT-TYPE
SYNTAX SessionType
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The type of session (IP4, IP6, IP6 with flow
information, etc), and therefore the structure
of the information."
::= { rsvpFlowEntry 2 }

rsvpFlowDestAddr OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Address used as a destination address by
all senders in this session."
::= { rsvpFlowEntry 3 }

rsvpFlowSenderAddr OBJECT-TYPE
SYNTAX OCTET STRING
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The Address used as a source address by this
sender in this session."
::= { rsvpFlowEntry 4 }
rsvpFlowProtocol OBJECT-TYPE
SYNTAX      Protocol
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The IP Protocol used by a session. for secure
sessions, this indicates IP Security."
 ::= { rsvpFlowEntry 5 }

rsvpFlowDestPort OBJECT-TYPE
SYNTAX      Port
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The UDP or TCP port number used as a destina-
tion port for all senders in this session. A
value of zero indicates that the IP protocol in
use, specified by rsvpSessionProtocol, does not
have ports."
 ::= { rsvpFlowEntry 6 }

rsvpFlowPort OBJECT-TYPE
SYNTAX      Port
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The UDP or TCP port number used as a source
port for this sender in this session. A value
of zero indicates that the IP protocol in use,
specified by rsvpSessionProtocol, does not have
ports."
 ::= { rsvpFlowEntry 7 }

rsvpFlowInterface OBJECT-TYPE
SYNTAX      InterfaceIndex
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
"The ifIndex value of the interface on which
this PATH message was most recently received."
 ::= { rsvpFlowEntry 8 }
rsvpFlowRate OBJECT-TYPE
SYNTAX    BitRate
UNITS     "kilobits per second"
MAX-ACCESS read-create
STATUS    current
DESCRIPTION
 "The Average Bit Rate of the sender's data stream, in Kilobits. The rate may be arbitrarily fast during a short interval such as the duration of a video frame. However, over any two such intervals it will not average faster than the average rate as transmitted by the sender."
 ::= { rsvpFlowEntry 9 }

rsvpFlowBurst OBJECT-TYPE
SYNTAX    BurstSize
UNITS     "bits"
MAX-ACCESS read-create
STATUS    current
DESCRIPTION
 "The size of the largest burst expected from the sender at a time.

If this is less than the sender's advertised burst size, the receiver is asking the network to provide flow pacing beyond what would be provided under normal circumstances. Such pacing is at the network's option."
 ::= { rsvpFlowEntry 10 }

rsvpFlowWeight OBJECT-TYPE
SYNTAX    Integer32
MAX-ACCESS read-create
STATUS    current
DESCRIPTION
 "The weight used to prioritize the traffic.  Note that the interpretation of this object is implementation-specific, as implementations vary in their use of weighting procedures."
 ::= { rsvpFlowEntry 11 }
rsvpFlowQueue OBJECT-TYPE
SYNTAX      Integer32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
  "The number of the queue used by this traffic. Note that the interpretation of this object is implementation-specific, as implementations vary in their use of queue identifiers."
 ::= { rsvpFlowEntry 12 }

rsvpFlowMinTU OBJECT-TYPE
SYNTAX      MessageSize
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
  "The minimum message size for this flow. The policing algorithm will treat smaller messages as though they are this size."
 ::= { rsvpFlowEntry 13 }

rsvpFlowDontAsk OBJECT-TYPE
SYNTAX      TimeInterval
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
  "A period of time, measured in hundredths of a second, during which the agent is precluded from sending a newFlow notification for this conversation.

If an authorizing application leaves rsvpFlowStatus in a state other than 'valid' and sets rsvpFlowDontAsk to a non-zero value, the agent must neither consider the flow active for admission or queuing purposes, or re-request authorization to install it, until the indicated amount of time has elapsed."
 ::= { rsvpFlowEntry 14 }
rsvpFlowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"'valid' for all active flows. This object may be used to install static classifier information, delete classifier information, or authorize such."
::= { rsvpFlowEntry 15 }
The RSVP Interface Attributes Database contains the RSVP-specific information for an interface. Information that is shared with other reservation procedures such as ST-II is in the Integrated Interface Attributes Database.

rsvpIfTable OBJECT-TYPE
SYNTAX SEQUENCE OF RsvpIfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "The RSVP-specific attributes of the system’s interfaces."
 ::= { rsvpObjects 6 }

rsvpIfEntry OBJECT-TYPE
SYNTAX RsvpIfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "The RSVP-specific attributes of the a given interface."
INDEX { ifIndex }
 ::= { rsvpIfTable 1 }

RsvpIfEntry ::= SEQUENCE {
  rsvpIfUdpNbrs Gauge32,
  rsvpIfIpNbrs Gauge32,
  rsvpIfNbrs Gauge32,
  rsvpIfEnabled TruthValue,
  rsvpIfUdpRequired TruthValue,
  rsvpIfRefreshBlockadeMultiple INTEGER,
  rsvpIfRefreshMultiple INTEGER,
  rsvpIfTTL INTEGER,
  rsvpIfRefreshInterval TimeInterval,
  rsvpIfRouteDelay TimeInterval,
  rsvpIfStatus RowStatus
}
rsvpIfUdpNbrs OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of neighbors perceived to be using only the RSVP UDP Encapsulation."
 ::= { rsvpIfEntry 1 }

rsvpIfIpNbrs OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of neighbors perceived to be using only the RSVP IP Encapsulation."
 ::= { rsvpIfEntry 2 }

rsvpIfNbrs OBJECT-TYPE
SYNTAX Gauge32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The number of neighbors currently perceived; this will exceed rsvpIfIpNbrs + rsvpIfUdpNbrs by the number of neighbors using both encap-
ulations."
 ::= { rsvpIfEntry 3 }

rsvpIfRefreshBlockadeMultiple OBJECT-TYPE
SYNTAX INTEGER {1..65536}
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The value of the RSVP value ‘Kb’, Which is the minimum number of refresh intervals that blockade state will last once entered."
DEFVAL { 4 }
 ::= { rsvpIfEntry 4 }
rsvpIfRefreshMultiple OBJECT-TYPE
SYNTAX INTEGER (1..65536)
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The value of the RSVP value 'K', which is the
number of refresh intervals which must elapse
(minimum) before a PATH or RESV message which
is not being refreshed will be aged out."
DEFVAL { 4 }
::= { rsvpIfEntry 5 }

rsvpIfTTL OBJECT-TYPE
SYNTAX INTEGER (0..255)
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The value of SEND_TTL used on this interface.
If set to zero, then the TTL in RSVP messages
is not overridden."
DEFVAL { 0 } -- which is to say, no override
::= { rsvpIfEntry 6 }

rsvpIfRefreshInterval OBJECT-TYPE
SYNTAX TimeInterval
UNITS "hundredths of a second"
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"The value of the RSVP value 'R', which is the
minimum period between refresh transmissions of
a given PATH or RESV message on an interface."
DEFVAL { 3000 } -- 30 seconds
::= { rsvpIfEntry 7 }
rsvpIfRouteDelay OBJECT-TYPE
SYNTAX TimeInterval
UNITS "hundredths of a second"
MAX-ACCESS read-create
STATUS current
DESCRIPTION "The approximate period from the time a route is changed to the time that a route is changed to the time a resulting message appears on the interface."
DEFVAL { 200 } -- 2 seconds ::= { rsvpIfEntry 8 }

rsvpIfEnabled OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-create
STATUS current
DESCRIPTION "If TRUE, RSVP is enabled on this Interface. If FALSE, RSVP is not enabled on this interface."
 ::= { rsvpIfEntry 9 }

rsvpIfUdpRequired OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-create
STATUS current
DESCRIPTION "If TRUE, manual configuration forces the use of UDP encapsulation on the interface. If FALSE, UDP encapsulation is only used if rsvpIfUdpNbrs is not zero."
 ::= { rsvpIfEntry 10 }
rsvpIfStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION "'valid' on interfaces that are configured for RSVP."
 ::= { rsvpIfEntry 11 }
The RSVP Neighbor Database lists the neighbors the RSVP process currently is receiving messages from.

rsvpNbrTable OBJECT-TYPE
SYNTAX     SEQUENCE OF RsvpNbrEntry
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION
   "Information describing the Neighbors of an RSVP system."
::= { rsvpObjects 7 }

rsvpNbrEntry OBJECT-TYPE
SYNTAX     RsvpNbrEntry
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION
   "Information describing a single RSVP Neighbor."
INDEX { ifIndex, rsvpNbrAddress }
::= { rsvpNbrTable 1 }

RsvpNbrEntry ::= 
SEQUENCE {
   rsvpNbrAddress      OCTET STRING, 
   rsvpNbrProtocol     RsvpEncapsulation, 
   rsvpNbrStatus       RowStatus
}

rsvpNbrAddress OBJECT-TYPE
SYNTAX     OCTET STRING (SIZE(4..16))
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION
   "The IP4 or IP6 Address used by this neighbor."
::= { rsvpNbrEntry 1 }
rsvpNbrProtocol OBJECT-TYPE
SYNTAX       RsvpEncapsulation
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  
"The encapsulation being used by this neighbor."
::= { rsvpNbrEntry 2 }

rsvpNbrStatus OBJECT-TYPE
SYNTAX       RowStatus
MAX-ACCESS   read-create
STATUS       current
DESCRIPTION  
"'valid' for all neighbors. This object may be used to configure neighbors. In the presence of configured neighbors, the implementation may (but is not required to) limit the set of valid neighbors to those configured."
::= { rsvpNbrEntry 3 }
newFlow NOTIFICATION-TYPE
OBJECTS {
}
STATUS current
DESCRIPTION
"The newFlow trap indicates that the originating system has installed a new flow in its classifier, or (when reservation authorization is in view) is prepared to install such a flow in the classifier and is requesting authorization. The objects included with the Notification may be used to read further information using the RSVP MIB. Authorization or non-authorization may be enacted by a write to the variable rsvpFlowStatus."
::= { rsvpNotifications 1 }

lostFlow NOTIFICATION-TYPE
OBJECTS {
}
STATUS current
DESCRIPTION
"The lostFlow trap indicates that the originating system has removed a flow in its classifier."
::= { rsvpNotifications 2 }
-- conformance information

rsvpGroups OBJECT IDENTIFIER ::= { rsvpConformance 1 }
rsvpCompliances OBJECT IDENTIFIER ::= { rsvpConformance 2 }

-- compliance statements

rsvpCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION "The compliance statement. Note that the implementation of this module requires implementation of the Integrated Services MIB as well."
  MODULE -- this module
  MANDATORY-GROUPS {
    rsvpSessionGroup, rsvpSenderGroup, rsvpResvGroup,
    rsvpResvFwdGroup, rsvpFlowGroup, rsvpIfGroup,
    rsvpNbrGroup
  }
  ::= { rsvpCompliances 1 }

rsvpSessionGroup OBJECT-GROUP
  OBJECTS {
    rsvpSessionNewIndex, rsvpSessionType, rsvpSessionDestAddr,
    rsvpSessionProtocol, rsvpSessionPort, rsvpSessionSenders,
    rsvpSessionReceivers, rsvpSessionRequests
  }
  STATUS current
  DESCRIPTION "These objects are required for RSVP Systems."
  ::= { rsvpGroups 1 }
rsvpSenderGroup OBJECT-GROUP
    OBJECTS {
      rsvpSenderType, rsvpSenderDestAddr, rsvpSenderSenderAddr,
      rsvpSenderProtocol, rsvpSenderDestPort, rsvpSenderPort,
      rsvpSenderHopAddr, rsvpSenderInterface, rsvpSenderTSpecRate,
      rsvpSenderTSpecPeakRate, rsvpSenderTSpecBurst,
      rsvpSenderTSpecMinTU, rsvpSenderTSpecMaxTU, rsvpSenderInterval,
      rsvpSenderLastChange, rsvpSenderStatus, rsvpSenderRSVPHop,
      rsvpSenderPolicy, rsvpSenderAdspec, rsvpSenderNewIndex
    }
    STATUS  current
    DESCRIPTION
      "These objects are required for RSVP Systems."
    ::= { rsvpGroups 2 }

rsvpResvGroup OBJECT-GROUP
    OBJECTS {
      rsvpResvType, rsvpResvDestAddr, rsvpResvSenderAddr,
      rsvpResvProtocol, rsvpResvDestPort, rsvpResvPort,
      rsvpResvHopAddr, rsvpResvInterface, rsvpResvService,
      rsvpResvTSpecRate, rsvpResvTSpecBurst, rsvpResvTSpecPeakRate,
      rsvpResvTSpecMinTU, rsvpResvTSpecMaxTU, rsvpResvTSpecLevel,
      rsvpResvInterval, rsvpResvScope, rsvpResvShared,
      rsvpResvExplicit, rsvpResvRSVPHop, rsvpResvLastChange,
      rsvpResvPolicy, rsvpResvStatus, rsvpResvNewIndex
    }
    STATUS  current
    DESCRIPTION
      "These objects are required for RSVP Systems."
    ::= { rsvpGroups 3 }
rsvpResvFwdGroup OBJECT-GROUP
  OBJECTS {
    rsvpResvFwdType, rsvpResvFwdDestAddr, rsvpResvFwdSenderAddr,
    rsvpResvFwdProtocol, rsvpResvFwdDestPort, rsvpResvFwdPort,
    rsvpResvFwdInterface, rsvpResvFwdNewIndex, rsvpResvFwdService,
    rsvpResvFwdTSpecPeakRate, rsvpResvFwdTSpecMinTU,
    rsvpResvFwdTSpecMaxTU, rsvpResvFwdTSpecRate, 
    rsvpResvFwdTSpecBurst, rsvpResvFwdTSpecLevel, 
    rsvpResvFwdInterval, rsvpResvFwdScope, rsvpResvFwdShared, 
    rsvpResvFwdExplicit, rsvpResvFwdRSVPHop, rsvpResvFwdLastChange,
    rsvpResvFwdPolicy, rsvpResvFwdStatus
  }
  STATUS current
  DESCRIPTION
    "These objects are required for RSVP Systems."
  ::= { rsvpGroups 4 }

rsvpFlowGroup OBJECT-GROUP
  OBJECTS {
    rsvpFlowType, rsvpFlowDestAddr, rsvpFlowSenderAddr,
    rsvpFlowProtocol, rsvpFlowDestPort, rsvpFlowPort,
    rsvpFlowInterface, rsvpFlowDontAsk, rsvpFlowRate,
    rsvpFlowBurst, rsvpFlowWeight, rsvpFlowQueue, rsvpFlowMinTU,
    rsvpFlowStatus
  }
  STATUS current
  DESCRIPTION
    "These objects are required for RSVP Systems."
  ::= { rsvpGroups 5 }

rsvpIfGroup OBJECT-GROUP
  OBJECTS {
    rsvpIfUdpNbrs, rsvpIfIpNbrs, rsvpIfNbrs, rsvpIfEnabled,
    rsvpIfUdpRequired, rsvpIfRefreshBlockadeMultiple,
    rsvpIfRefreshMultiple, rsvpIfRefreshInterval, rsvpIfTTL,
    rsvpIfRouteDelay, rsvpIfStatus
  }
  STATUS current
  DESCRIPTION
    "These objects are required for RSVP Systems."
  ::= { rsvpGroups 6 }

Baker and Krawczyk   Expires September 1996
rsvpNbrGroup OBJECT-GROUP
   OBJECTS {
       rsvpNbrProtocol, rsvpNbrStatus
   }
STATUS current
DESCRIPTION
   "These objects are required for RSVP Systems."
 ::= { rsvpGroups 7 }
6. Acknowledgements

This document was produced by the RSVP Working Group.
7. References


Table of Contents

1 Status of this Memo ................................... 1
2 Abstract ............................................... 2
3 The SNMPv2 Network Management Framework ............ 3
3.1 Object Definitions .................................. 3
4 Overview ............................................. 3
4.1 Textual Conventions ................................ 3
4.2 Structure of MIB .................................... 4
5 Definitions ........................................... 5
5.1 RSVP Session Statistics Database .................... 8
5.2 RSVP Session Sender Database ....................... 12
5.3 RSVP Reservation Requests Database ................. 20
5.4 RSVP Reservation Requests Database ................. 29
5.3 RSVP Interface Flows Database ...................... 38
5.4 RSVP Interface Attributes Database ................. 44
5.5 RSVP Neighbor Database ............................ 49
5.4 Notifications ....................................... 51
6 Acknowledgements ..................................... 57
7 References ............................................ 58