Aggregation of RSVP for IPv4 and IPv6 Reservations
draft-baker-rsvp-aggregation-01.txt

Abstract

A key problem in the design of RSVP version 1 is, as noted in its applicability statement, that it lacks facilities for aggregation of individual reserved sessions into a common class. The use of such aggregation is required for scalability.

This document describes the use of a single RSVP reservation to aggregate other RSVP reservations across a transit routing region, in a manner conceptually similar to the use of Virtual Paths in an ATM network. It proposes a way to dynamically create the aggregate reservation, classify the traffic for which the aggregate reservation applies, determine how much bandwidth is needed to achieve the requirement, and recover the bandwidth when the sub-reservations are no longer required. It also contains recommendations concerning algorithms and policies for predictive reservations.

1. Introduction

A key problem in the design of RSVP version 1 [RSVP] is, as noted in its applicability statement, that it lacks facilities for aggregation of individual reserved sessions into a common class. The use of such aggregation is recommended in [CSZ], and required for scalability.

The problem of aggregation may be addressed in a variety of ways. For example, it may sometimes be sufficient simply to mark reserved traffic with a suitable DSCP (e.g. EF), thus enabling aggregation of scheduling and classification state. It may also be desirable to install one or more aggregate reservations from ingress to egress of an "aggregation region" (defined below) where each aggregate reservation carries similarly marked packets from a large number of flows. This is to provide high levels of assurance that the end-to-end requirements of reserved flows will be met, while at the same time enabling reservation state to be aggregated.

Throughout, we will talk about "Aggregator" and "Deaggregator", referring to the routers at the ingress and egress edges of an aggregation region. Exactly how a router determines whether it should perform the role of aggregator or deaggregator is described below.

We will refer to the individual reserved sessions (the sessions we are attempting to aggregate) as "end-to-end" reservations ("E2E" for short), and to their respective Path/Resv messages as E2E Path/Resv messages. We refer to the the larger reservation (that which represents many E2E reservations) as an "aggregate" reservation, and its respective Path/Resv messages as "aggregate Path/Resv messages".

1.1. Problem Statement: Aggregation Of E2E Reservations

The problem of many small reservations has been extensively discussed, and may be summarized in the observation that each reservation requires a non-trivial amount of message exchange, computation, and memory resources in each router along the way. It would be nice to reduce this to a more manageable level where the load is heaviest and aggregation is possible.

Aggregation, however, brings its own challenges. In

particular, it reduces the level of isolation between individual flows, implying that one flow may suffer delay from the bursts of another. Synchronization of bursts from different flows may occur. However, there is evidence [CSZ] to suggest that aggregation of flows has no negative effect on the mean delay of the flows, and actually leads to a reduction of delay in the "tail" of the delay distribution (e.g. 99% percentile delay) for the flows. These benefits of aggregation to some extent offset the loss of strict isolation.

1.2. Proposed Solution

The solution we propose involves the aggregation of several E2E reservations that cross an "aggregation region" and share common ingress and egress routers into one larger reservation from ingress to egress. We define an "aggregation region" as a contiguous set of systems capable of performing RSVP aggregation (as defined following) along any possible route through this contiguous set.

Communication interfaces fall into two categories with respect to an aggregation region; they are "exterior" to an aggregation region, or they are "interior" to it. Routers that have at least one interface in the region fall into three categories with respect to a given RSVP session; they aggregate, they deaggregate, or they are between an aggregator and a deaggregator.

Aggregation depends on being able to hide E2E RSVP messages from routers inside the aggregation region. To achieve this end, the IP Protocol Number in the E2E reservation's PATH, PATH-TEAR, and RESV-CONFIRM messages is changed from RSVP (46) to RSVP-E2E-IGNORE (a new value, to be assigned) upon entering the aggregation region, and restored to RSVP at the deaggregator point. These messages are ignored (no state is stored and the message is forwarded as a normal IP datagram) by each router within the aggregation region whenever they are forwarded to an inside interface. Since the deaggregating router perceives the previous hop on such messages to be the aggregating router, RESV and other messages do not require this modification; they are unicast from system to system anyway.

The token buckets (SENDER_TSPECs and FLOWSPECS) of E2E reservations are summed into the corresponding information elements in aggregate PATH and RESV messages. Aggregate PATH

messages are sent from the aggregator to the deaggregator(s) using RSVP's normal IP Protocol Number. Aggregate RESV messages are sent back from the deaggregator to the aggregator, thus establishing an aggregate reservation on behalf of the set of E2E flows that use this aggregator and deaggregator. There may be several such aggregate reservations between the same two routers, representing different classes of traffic; the aggregate reservation is therefore for the traffic marked with a particular DSCP Group.

1.3. Definitions

We define an "aggregation region" as a set of RSVP-capable routers for which E2E RSVP messages arriving on an outside interface of one router in the set would traverse one or more inside interfaces (of this and/or other routers in the set) before finally traversing an outside interface.

Such an E2E RSVP message is said to have crossed the aggregation region.

We define the "aggregating" router for this E2E flow as the first router that processes the E2E Path message as it enters the aggregation region (i.e., the one which forwards the message from an outside interface to an inside interface).

We define the "deaggregating" router for this E2E flow as the last router to process the E2E Path as it leaves the aggregation region (i.e., the one which forwards the message from an inside interface to an outside interface).

We define an "interior" router for this E2E flow as any router in the aggregation region which receives this message on an inside interface and forwards it to another inside interface. Interior routers perform neither aggregation nor deaggregation for this flow.

1.4. Detailed Aspects of Proposed Solution

A number of issues jump to mind in considering this model.

1.4.1. Traffic Classification Within The Aggregation region

One of the reasons that RSVP Version 1 did not identify a way to aggregate sessions was that there was not a clear way to classify the aggregate. With the development of the Differentiated Services architecture, this is at least partially resolved; traffic of a particular class can be marked with a given DSCP and so classified. We presume this model.

We presume that on each link en route, a queue, WDM color, or similar management component is set aside for all aggregated traffic of the same class, and that sufficient bandwidth is made available to carry the traffic that has been assigned to it. This bandwidth may be adjusted based on the total amount of aggregated reservation traffic assigned to the same class.

There are numerous options for exactly which Diff-serv PHBs might be used for different classes of traffic as it crosses the aggregation region. This is the "service mapping" problem described in [ISDS], and is applicable to situations broader than those described in this document. Arguments can be made for using either EF or one or more AF PHBs for aggregated traffic.

Independent of which PHB is used, care needs to be take in an environment where provisioned Diff-Serv and aggregated RSVP are used in the same network, to ensure that the total offered load for a single PHB does not exceed the link capacity allocated to that PHB. One solution to this is to reserve one of the four AF classes strictly for the aggregated reservation traffic while using other AF classes for provisioned Diff-Serv.

Therefore, while some RSVP reservation state per aggregate reservation is maintained inside the aggregation region, a single classification and scheduling state (e.g., a DSCP used for classifying traffic) is maintained per aggregate reservation class (rather than per aggregate reservation) inside the aggregation region. For example, if Guaranteed Service is represented by the EF DSCP throughout the aggregation region, there may be a reservation for each aggregator/deaggregator pair in each router, but only the EF DSCP need be inspected at each inside interface.

1.4.2. Deaggregator Determination

The first question is "How do we know which aggregate reservation a particular E2E flow should aggregate into?" To know that, we must know three things: its aggregating router, its deaggregating router, and (assuming DSCPs are used to differentiate among various reservations between the same two routers), the relevant DSCP.

The aggregator is trivial: we know that an E2E flow has arrived at an aggregator when its PATH message arrives at a router on an outside interface and must be forwarded on an inside interface.

The DSCP is equally easy, or at least it is in concept. The DSCP is chosen for an aggregate reservation based on locally configured policy, perhaps taking into account such factors as the intserv service class requested for the flow. (Some details in the exact point at which the DSCP can be determined are discussed below.)

The deaggregator is more involved. If an SPF routing protocol, such as OSPF or IS-IS, is in use, and if it has been extended to advertise information on Deaggregation roles, it can tell us the set of routers from which the deaggregator will be chosen. In principle, if the aggregator and deaggregator are in the same area, then the identity of the deaggregator could be determined from the link state database. However, this approach would not work in multi-area environments or for distance vector protocols.

One method for Deaggregator determination is manual configuration. With this method the network operator would configure the Aggregator and the Deaggregator with the necessary information.

Another method allows automatic Deaggregator determination and corresponding Aggregator notification. When the E2E RSVP PATH message transits from an inside interface to an outside interface, the deaggregating router must advise the aggregating router of the correlation between itself and the flow. This has the nice attribute of not being specific to the routing protocol. It also has the property of automatically adjusting to route changes. For instance, if because of a topology change, another Deaggregator is now on the shortest path, this method will automatically identify the new Deaggregator and swap to it.

1.4.3. Size of Aggregate Reservations

A range of options exist for determining the size of the aggregate reservation, presenting a tradeoff between simplicity and scalability. Simplistically, the size of the aggregate reservation needs to be greater than or equal to the sum of the bandwidth of the E2E reservations it aggregates, and its burst capacity must be greater than or equal to the sum of their burst capacities. However, if followed religiously, this leads us to change the bandwidth of the aggregate reservation each time an underlying E2E reservation changes, which loses one of the key benefits of aggregation, the reduction of message processing cost in the aggregation region.

We assume, therefore, that there is some policy, not defined in this specification (although sample policies are suggested which have the necessary characteristics). This policy maintains the amount of bandwidth on a given aggregate reservation at an amount greater than or equal to the sum of the bandwidths of its underlying E2E reservations, while endeavoring to change it infrequently. This may require some level of trend analysis. If there is a significant probability that in the next interval of time the current aggregate reservation will be exhausted, the router must predict the necessary bandwidth and request it. If the router has a significant amount of bandwidth reserved but has very little probability of using it, the policy may be to predict the amount of bandwidth required and release the excess.

This policy is likely to benefit from introduction of some hysteresis (i.e. ensure that the trigger condition for aggregate reservation size increase is sufficiently different from the trigger condition for aggregate reservation size decrease) to avoid oscillation in stable conditions.

Clearly, the definition and operation of such policies are as much business issues as they are technical, and are out of the scope of this document.

1.4.4. Intra-domain Routes

RSVP directly handles route changes, in that reservations follow the routes that their data follow. This follows from the property that PATH messages contain the same IP source and destination address as the data flow for which a reservation

is to be established. However, since we are now making aggregate reservations by sending a PATH message from an aggregating to a deaggregating router, the reserved (E2E) data packets no longer carry the same IP addresses as the relevant (aggregate) PATH message. The issue becomes one of making sure that data packets for reserved flows follow the same path as the PATH message that established Path state for the aggregate reservation. Several approaches are viable.

First, the data may be tunneled from aggregator to deaggregator, using technologies such as IP-in-IP tunnels, GRE tunnels, MPLS labeled tunnels, and so on. These each have particular advantages, especially MPLS, which allows traffic engineering. They each also have some cost in link overhead and configuration complexity.

If data is not tunneled, then we are depending a characteristic of IP best metric routing , which is that if the route from A to Z includes the path from H to L, and the best metric route was chosen all along the way, then the best metric route was chosen from H to L. Therefore, an aggregate path message which crosses a given aggregator and deaggregator will of necessity use the best path between them.

If this is a single path, the problem is solved. If it is a multi-path route, then we are forced to determine, perhaps by measurement, what proportion of the traffic for a given E2E reservation is passing along each of the paths, and assure ourselves of sufficient bandwidth for the present use. A simple, though inelegant, way of doing this is to reserve the total capacity of the aggregate route down each path.

For this reason, we believe it is advantageous to use one of the above-mentioned tunneling mechanisms in cases where multi-path routes may exist.

1.4.5. Inter-domain Routes

The case of inter-domain routes differs somewhat from the intra-domain case just described. Specifically, best-path considerations do not apply, as routing is by a combination of routing policy and shortest AS path rather than simple best metric.

In the case of inter-domain routes, data traffic belonging to different E2E sessions (but the same aggregate session) may

not enter an aggregation region via the same aggregator interface, and/or may not leave via the same deaggregator interface. It is possible that we could identify this occurrence in some central system which sees the reservation information for both of the apparent sessions, but it is not clear that we could determine a priori how much traffic went one way or the other apart from measurement.

We simply note that this problem can occur and needs to be allowed for in the implementation. We recommend that each such e2e reservation be summed into its appropriate aggregate reservation, even though this involves over-reservation.

1.4.6. Reservations for Multicast Sessions

Aggregating reservations for multicast sessions is significantly more complex than for unicast sessions. The first challenge is to construct a multicast tree for distribution of the aggregate Path messages which follows the same path as will be followed by the data packets for which the aggregate reservation is to be made. This is complicated by the fact that the path which is followed by a data packet may depend on many factors such as its source address, the choice of shared trees or source-specific trees, and the location of a rendezvous point for the tree.

Once the problem of distributing aggregate Path messages is solved, there are considerable problems in determining the correct amount of resources to reserve at each link along the multicast tree. Because of the amount of heterogeneity that may exist in an aggregate multicast reservation, it appears that it would be necessary to retain information about individual E2E reservations within the aggregation region to allocate resources correctly. Thus, we may end up with a complex set of procedures for forming aggregate reservations that do not actually reduce the amount of stored state significantly for multicast sessions. [BERSON] describes possible ways to reduce this state by using measurement-based admission control.

As noted above, there are several aspects to RSVP state, and our approach for unicast aggregates all forms of state: classification, scheduling, and reservation state. One possible approach to multicast is to focus only on aggregation of classification and scheduling state, which are arguably the most important because of their impact on the fast path. That

approach is the one described in the current draft.

1.4.7. Multi-level aggregation

Ideally, an aggregation scheme should be able to accommodate recursive aggregation, with aggregate reservations being themselves aggregated. Multi-level aggregation can be accomplished using the procedures described here and a simple extension to the protocol number swapping process.

We can consider E2E RSVP reservations to be at aggregation level 0. When we aggregate these reservations, we produce reservations at aggregation level 1. In general, level n reservations may be aggregated to form reservations at level n+1.

When an aggregating router receives an E2E Path, it swaps the protocol number from RSVP to RSVP-E2E-IGNORE. In addition, it should write the aggregation level (1, in this case) in the 2 byte field that is present (and currently unused) in the router alert option. In general, a router which aggregates reservations at level n to create reservations at level n+1 will write the number n+1 in the router alert field. A router which deaggregates level n+1 reservations will examine all messages with IP protocol number RSVP-E2E-IGNORE but will process the message and swap the protocol number back to RSVP only in the case where the router alert field carries the number n+1. For any other value, the message is forwarded unchanged. Interior routers ignore all messages with IP protocol number RSVP-E2E-IGNORE.

1.4.8. Reliability Issues

There are a variety of issues that arise in the context of aggregation that would benefit from some form of explicit acknowledgment mechanism for RSVP messages. For example, it is possible to configure a set of routers such that an E2E PATH of protocol type RSVP-E2E-IGNORE would be effectively "black-holed", if it never reached a router which was appropriately configured to act as a deaggregator. It could then travel all the way to its destination where it would probably be ignored due to its non-standard protocol number. This situation is not easy to detect. The aggregator can be sure this problem has not occurred if a Path Error message is received from the deaggregator (as described in detail below).

It can also be sure there is no problem if an E2E Resv is received. However, the fact that neither of these events has happened may only mean that no receiver wishes to reserve resources for this session, or it may mean that the PATH was black-holed. However, if a neighbor-to-neighbor acknowledgment mechanism existed, the aggregator would expect to receive an acknowledgment from the deaggregator, and would interpret the lack of a response as an indication that a problem of configuration existed. It could then refrain from aggregating this particular session. We note that such a reliability mechanism has been proposed for RSVP in [REFRESH] and propose that it be used here.

2. Elements of Procedure

To implement aggregation, we define a number of elements of procedure.

2.1. Receipt of E2E Path Message By aggregating router

The very first event is the arrival of the E2E PATH message at an outside interface of an aggregator. Standard RSVP procedures [RSVP] are followed for this, including consideration of what set of interfaces it needs to be forwarded onto. These interfaces comprise zero or more outside interfaces and zero or more inside interfaces.

Service on outside interfaces is handled as defined in [RSVP].

Service on inside interfaces is complicated by the fact that the message needs to be included in some number of aggregate reservations, but at this point it is not known which one, because the deaggregator is not known. Therefore, the E2E PATH message is forwarded on the inside interface(s) using the IP Protocol number RSVP-E2E-IGNORE, but in every other respect identically to the way it would be sent by an RSVP router that was not performing aggregation.

2.2. Handling Of E2E Path Message By Interior Routers

At this point, the e2e Path message traverses zero or more interior routers. Interior routers receive the e2e Path message on an inside interface and forward it on another inside interface. The Router Alert IP Option alerts interior routers to check internally, but they find that the IP Protocol is RSVP-E2E-IGNORE and the next hop interface is inside. As such, they simply forward it as a normal IP datagram.

2.3. Receipt of E2E Path Message By Deaggregating router

The E2E PATH message finally arrives at a deaggregating router, which receives it on an inside interface and forwards it on an outside interface. Again, the Router Alert IP Option alerts it to intercept the message, but this time the IP Protocol is RSVP-E2E-IGNORE and the next hop interface is an

outside interface.

At this point, the deaggregating router associates the flow with an aggregate reservation. This selection is done on the basis of policy, and may take into account not only the aggregating router (whose IP Address may be found in the RSVP Hop Object) but other information about the flow. If no such aggregate reservation exists and the router is so configured, it may generate a PATH ERROR with code NEW-AGGREGATE-NEEDED back to the aggregating router. This should not result in any reservation being taken down, but may result in the aggregating router initiating the necessary aggregate path message, as described in the following section.

The deaggregating router changes the e2e Path message's IP Protocol from RSVP-E2E-IGNORE to IP Protocol RSVP, updates the ADSPEC of the e2e Path using information accumulated by the aggregate Path ADSPEC (if an aggregate Path has been received), and the E2E PATH message is forwarded towards its intended destination. To enable correct updating of the ADSPEC, a deaggregating router may wait for the arrival of an aggregate Path before forwarding the E2E Path.

2.4. **Initiation of New Aggregate Path Message By Aggregating** router

The aggregating router is responsible to include the SENDER_TSPEC information from individual E2E Path messages in the SENDER_TSPEC of the aggregate Path message it sends to its deaggregating router. The aggregating router may know that an E2E session is associated with a given deaggregator when one of two events occurs: it receives a PATH ERROR message with the error code NEW-AGGREGATE-NEEDED from the deaggregator, or it receives an E2E RESV message from the deaggregator. In the latter case, the RESV contains a DCLASS object [DCLASS] indicating which DSCP the deaggregator believes that the E2E flow belongs in. In the former case, the aggregator must make its own determination of a suitable DSCP based on the information in the E2E Path message(s) being aggregated and using locally available policy information. The identity of the deaggregator itself is found in either the ERROR SPECIFICATION of the Path Error message or the RSVP HOP object of the RESV.

On receipt of either message, if no corresponding aggregate

path state exists from the aggregator to the deaggregator for a session with the appropriate DSCP, and the aggregator is configured to do so, the aggregator should generate an aggregate PATH message for the aggregate reservation. The destination address of the aggregate PATH message is the address of the deaggregating router, and the message is sent with IP protocol number RSVP.

2.5. Handling of E2E RESV Message by Deaggregating Router

Having sent the E2E PATH message on toward the destination, the deaggregator must now expect to receive an E2E RESV for the session. On receipt, its responsibility is to assure itself that there is sufficient bandwidth reserved within the aggregation region to support the new E2E reservation, and if there is, then to forward the E2E RESV to the aggregating router.

If there is insufficient bandwidth reserved, it should follow the normal RSVP procedures [RSVP] for a reservation being placed with insufficient bandwidth to support the reservation. It may also immediately attempt to increase the aggregate reservation that is supplying bandwidth by increasing the size of the flowspec that it includes in the aggregate RESV that it sends upstream.

When sufficient bandwidth is available, it may simply send the E2E RESV message with IP Protocol RSVP to the aggregating router. This message should, in addition to other data, contain the DCLASS object to indicate which DSCP the deaggregating router expects the aggregator to use. The choice of DSCP may be made based on a combination of information in the received E2E RESV and local policy. An example policy might dictate a certain DSCP for Guaranteed Service and another DSCP for Controlled Load. The de-aggregator will also add the token bucket from the FLOWSPEC object into its internal understanding of how much of that reservation is in use.

2.6. Initiation of New Aggregate RESV Message By Deaggregating Router

Upon receiving an E2E RESV message on an outside interface, and having determined the appropriate DSCP for the session, the deaggregator looks for corresponding path state for a

session with the chosen DSCP. If aggregate PATH state exists, but no aggregate RESV state exists, the deaggregator creates an aggregate RESV and sets its initial request to a value not smaller than the requirement of the E2E reservation it is supporting.

If no aggregate PATH state exists for the appropriate DSCP, this may be because the aggregator has not yet responded to the arrival of the E2E RESV sent in the preceding step. To avoid deadlock while waiting for a response, it would be desirable to use the acknowledgment mechanisms described in [REFRESH].

Once the deaggregator has the aggregate PATH message, then it sends an aggregate RESV message toward the aggregator (i.e., to the previous hop), using the AGGREGATED-RSVP session and filter specifications. Since the DSCP is in the SESSION object, the DCLASS is unnecessary. The message should be reliably delivered using the mechanisms in [REFRESH] or, alternatively, the CONFIRM object may be used, to assure that the aggregate RESV does indeed arrive and is granted. This enables the deaggregator to determine that the requested bandwidth is available to allocate to the E2E flows it supports.

2.7. Handling of Aggregate RESV Message by Interior Routers

The aggregate RESV message is handled in essentially the same way as defined in [RSVP]. The Session object contains the address of the deaggregating router (or the group address for the session in the case of multicast) and the DSCP that has been chosen for the session. The Filterspec object identifies the aggregating router. These routers perform admission control and resource allocation as usual and send the aggregate RESV on towards the aggregator.

2.8. Handling of E2E RESV Message by Aggregating Router

The E2E RESV message is the final confirmation to the aggregating router that a proportion of a given aggregate's bandwidth has been reserved. At this point, it should assure itself that the E2E reservation is associated with an appropriate aggregate, that the aggregator and deaggregator expectations synchronize, and that all things are in place. In particular, it needs to ensure that the DCLASS carried in the

E2E RESV matches the DSCP for an aggregate session to that deaggregator; if not, it needs to create a new aggregate Path for the appropriate DSCP and send it to the deaggregator. It should also assure itself that the SENDER_TSPEC from the E2E PATH message has been accumulated into the appropriate aggregate PATH message. Under normal circumstances, this is the only way it will be informed of this association. It should now forward the E2E RESV to its previous hop, following normal RSVP processing rules [RSVP].

2.9. Removal of E2E Reservation

E2E reservations are removed in the usual way via PATH TEAR, RESV TEAR, timeout, or as the result of an error condition. When they are removed, their FLOWSPEC information must also be removed from the allocated portion of the aggregate reservation. This same bandwidth may be re-used for other traffic in the near future. When E2E PATH messages are removed, their SENDER_TSPEC information must also be removed from the aggregate PATH.

2.10. Removal of Aggregate Reservation

Should an aggregate reservation go away (presumably due to a configuration change, route change, or policy event), the E2E reservations it supports are no longer active. They must be treated accordingly.

2.11. Handling of Data On Reserved E2E Flow by Aggregating Router

Prior to establishment that a given E2E flow is part of a given aggregate, the flow's data should be treated as general best effort traffic by whatever policies prevail for such. Generally, this will mean being given the same throughput behavior as non-essential traffic. However, upon establishing that, the aggregating router is responsible to mark any related traffic with the correct DSCP and forward it in the manner appropriate to traffic on that reservation. This may imply forwarding it to a given IP next hop, or piping it down a given link layer circuit, tunnel, or MPLS label switched path.

2.12. Procedures for Multicast Sessions

Because of the difficulties of aggregating multicast sessions described above, we focus on the aggregation of scheduling and classification state in the multicast case. The main difference between the multicast and unicast cases is that rather than sending an aggregate Path message to the unicast address of a single deaggregating router, in the multicast case we send the "aggregate" Path message to the same group address as the E2E session. This ensures that the aggregate Path message follows the same route as the E2E Path. This difference between unicast and multicast is reflected in the Session objects defined below. A consequence of this approach is that we continue to have reservation state per multicast session inside the aggregation region.

3. Protocol Elements

3.1. IP Protocol RSVP-E2E-IGNORE

This specification presumes the assignment of a protocol type RSVP-E2E-IGNORE, whose number is at this point TBD. This is used only on messages which require a router alert (PATH, PATH ERROR, and RESV CONFIRM), and signifies that the message must be treated one way when copied to an inside interface, and another way when copied to an outside interface.

3.2. Path Error Code

A PATH ERROR code NEW-AGGREGATE-NEEDED is presumed. This value does not signify that a terminal error has occurred, but that an action is required of the aggregating router to avoid an error condition in the near future.

3.3. SESSION Object

The SESSION object contains two values: the IP Address of the aggregate session destination, and the DSCP that it will use on the E2E data the reservation contains. For unicast sessions, the session destination address is address of the deaggregating router. For multicast sessions, the session destination is the multicast address of the E2E session (or sessions) being aggregated. The inclusion of the DSCP in the

session allows for multiple sessions toward the same address to be distinguished by their DSCP and queued separately. It also provides the means for aggregating scheduling and classification state. In the case where a session uses a pair of PHBs (e.g. AF11 and AF12), the DSCP used should represent the numerically smallest PHB (e.g. AF11). This follows the same naming convention described in [BRIM].

Session types are defined for IPv4 and IPv6 addresses.

o IP4 SESSION object: Class = SESSION,
C-Type = RSVP-AGGREGATE-IP4

```
+-------------+-------------+-------------+-------------+
|              IPv4 Session Address (4 bytes)           |
+-------------+-------------+-------------+-------------+
| /////////// |    Flags    |  /////////  |     DSCP    |
+-------------+-------------+-------------+-------------+
```

o IP6 SESSION object: Class = SESSION,
C-Type = RSVP-AGGREGATE-IP6

```
+-------------+-------------+-------------+-------------+
|                                                       |
+                                                       +
|                                                       |
+              IPv6 Session Address (16 bytes)          +
|                                                       |
+                                                       +
|                                                       |
+-------------+-------------+-------------+-------------+
| /////////// |    Flags    |  /////////  |     DSCP    |
+-------------+-------------+-------------+-------------+
```

3.4. SENDER_TEMPLATE Object

The SENDER_TEMPLATE object identifies the aggregating router for the aggregate reservation.

o IP4 SENDER_TEMPLATE object: Class = SENDER_TEMPLATE,
C-Type = RSVP-AGGREGATE-IP4

```
+-------------+-------------+-------------+-------------+
|                IPv4 Aggregator Address (4 bytes)      |
+-------------+-------------+-------------+-------------+
```

```
      o    IP6 SENDER_TEMPLATE object: Class = SENDER_TEMPLATE,
           C-Type = RSVP-AGGREGATE-IP6

           +-------------+-------------+-------------+-------------+
           |                                                       |
           +                                                       +
           |                                                       |
           +           IPv6 Aggregator Address (16 bytes)          +
           |                                                       |
           +                                                       +
           |                                                       |
           +-------------+-------------+-------------+-------------+
```

3.5. FILTER_SPEC Object

The FILTER_SPEC object identifies the aggregating router for the aggregate reservation, and is syntactically identical to the SENDER_TEMPLATE object.

4. **Policies and Algorithms For Predictive Management Of** Blocks Of Bandwidth

The exact policies used in determining how much bandwidth should be allocated to an aggregate reservation at any given time are beyond the scope of this document, and may be proprietary to the service provider in question. However, here we explore some of the issues and suggest approaches.

In short, the ideal condition is that the aggregate reservation always has enough resources to allocate to any flow reservation that requires its support, and never takes too much. Simply stated, but more difficult to achieve. Factors that come into account include significant times in the diurnal cycle: one may find that a large number of people start placing calls at 8:00 AM, even though the hour from 7:00 to 8:00 is dead calm. They also include recent history: if more people have been placing calls recently than have been finishing them, a prediction of the necessary bandwidth a few moments hence may call for more bandwidth than is currently allocated. Likewise, at the end of a busy period, we may find that the trend calls for declining reservation amounts.

We recommend a policy something along this line. At any given time, one should expect that the amount of bandwidth required for the aggregate reservation is the larger of the following:

(a) a requirement known a priori, such as from history of the diurnal cycle at a particular week day and time of day, and

(b) the trend line over recent history, with 90 or 99% statistical confidence.

We further expect that changes to that aggregate reservation would be made no more often than every few minutes, and ideally perhaps on larger granularity such as fifteen minute intervals or hourly. The finer the granularity, the greater the level of signaling required, while the coarser the granularity, the greater the chance for error, and the need to recover from that error.

In general, we expect that the aggregate reservation will not ever add up to exactly the sum of the reservations it supports, but rather will be an integer multiple of some block reservation size, which exceeds that value.

5. Security Considerations

Numerous security issues pertain to this document; for example, the loss of an aggregate reservation to an aggressor causes many calls to operate unreserved, and the reservation of a great excess of bandwidth may result in a denial of service. However, these issues are not confined to this extension: RSVP itself has them. We believe that the security mechanisms in RSVP address these issues as well.

6. IANA Considerations

Beyond allocating an IP Protocol, a PATH ERROR code, and an RSVP Addressing object "type", there are no IANA issues in this document. We do not define an object that will itself require assignment by IANA.

7. Acknowledgments

The authors acknowledge that published documents and discussion with several people, notably John Wroclawski, Steve Berson, and Andreas Terzis materially contributed to this draft. The design derives directly from an internet draft by Roch Guerin [GUERIN] and from Steve Berson's drafts on the subject. It is also influenced by the design in the diff-edge draft by Bernet et al [BERNET] and by the RSVP tunnels draft [TERZIS].

8. References

[CSZ]
Clark, D., S. Shenker, and L. Zhang, "Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism," in Proc. SIGCOMM'92, September 1992.

[IP] RFC 791, "Internet Protocol". J. Postel. Sep-01-1981.

[HOSTREQ]
RFC 1122, "Requirements for Internet hosts - communication layers". R.T. Braden. Oct-01-1989.

[FRAMEWORK]
Nichols, "Differentiated Services Operational Model and Definitions", 02/11/1998, draft-nichols-dsopdef-00.txt

[PRINCIPLES]
RFC 1958, "Architectural Principles of the Internet". B. Carpenter. June 1996.

[ASSURED]
Clark and Wroclawski, "An Approach to Service Allocation in the Internet", 08/04/1997, draft-clark-diff-svc-alloc-00.txt

[BROKER]
Nichols and Zhang, "A Two-bit Differentiated Services Architecture for the Internet", 12/23/1997, draft-nichols-diff-svc-arch-01.txt

[BERSON]
Berson and Vincent. "Aggregation of Internet Integrated Services State". draft-berson-rsvp-aggregation-00.txt, August 1998

[BRIM]
Brim and Carpenter. "Per Hop Behavior Identification Codes". draft-brim-diffserv-phbid-00.txt, April 1999.

[ISDS]
Bernet et al. "Integrated Services Operation Over Diffserv Networks". draft-ietf-issll-diffserv-rsvp-02.txt, June 1999.

[GUERIN]
Guerin, R., Blake, S. and Herzog, S.,"Aggregating RSVP based QoS Requests", Internet Draft, draft-guerin-aggreg-rsvp-00.txt, November 1997.

[RSVP]
Braden, R., Zhang, L., Berson, S., Herzog, S. and Jamin, S., "Resource Reservation Protocol (RSVP) Version 1 Functional Specification", RFC 2205, September 1997.

[BERNET]
Bernet, Y., Durham, D., and F. Reichmeyer, "Requirements of Diff-serv Boundary Routers", Internet Draft, draft-bernet-diffedge-01.txt, November, 1998.

[REFRESH]
Berger, L., Gan, D., and G. Swallow, "RSVP Refresh Reduction Extensions", Internet Draft, draft-berger-rsvp-refresh-reduct-02.txt, May 1999.

[TERZIS]
Terzis, A., Krawczyk, J., Wroclawski, J., and L. Zhang, "RSVP Operation Over IP Tunnels", Internet Draft, draft-ietf-rsvp-tunnel-04.txt, May 1999.

[DCLASS]
Bernet, Y., "Usage and Format of the DCLASS Object With RSVP Signaling", Internet Draft, draft-bernet-dclass-01.txt, June 1999.

9. Authors' Addresses

Fred Baker
Cisco Systems
519 Lado Drive
Santa Barbara, California 93111
Phone: (408) 526-4257
Email: fred@cisco.com

Carol Iturralde
Cisco Systems
250 Apollo Drive
Chelmsford MA,01824 USA
Phone: 978-244-8532
Email: cei@cisco.com

Francois Le Faucheur
Cisco Systems
291, rue Albert Caquot
06560 Valbonne, France
Phone: +33.1.6918 6266
Email: flefauch@cisco.com

Bruce Davie
Cisco Systems
250 Apollo Drive
Chelmsford MA,01824 USA
Phone: 978-244-8921
Email: bdavie@cisco.com