

**Support of Shortcuts for RSVP Flows Over ATM**  
**draft-guerin-iss11-rsvp-shortcut-00.txt**

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, and 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."

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the internet-drafts Shadow Directories on ds.internic.net (US East Coast), nic.nordu.net (Europe), ftp.isi.edu (US West Coast), or munnari.oz.au (Pacific Rim).

Abstract

Support for RSVP flows across an ATM network has been defined in [BB97, GB97, Ber97a]. Although the use of shortcuts was mentioned in [BB97, Sec. 3.7], and [Ber97a, Sec. 3.6] the current specifications do not address this case in sufficient details so as to allow interoperable implementations. The purpose of this document is to identify issues in using ATM shortcuts for RSVP flows. For purposes of illustrations, the NHRP protocol [LKPC97] will be assumed as the mechanism used to discover shortcuts, and a possible interface between RSVP and NHRP will be outlined. However, it should be noted that other shortcut mechanisms are possible, e.g., PAR [J96], and the general conclusions of this document should also hold in such cases. Since the issue of shortcut discovery is significantly more complex in the multicast case, for which in most instances has not been defined, e.g., NHRP, this document is limited to the case of unicast flows.



## Contents

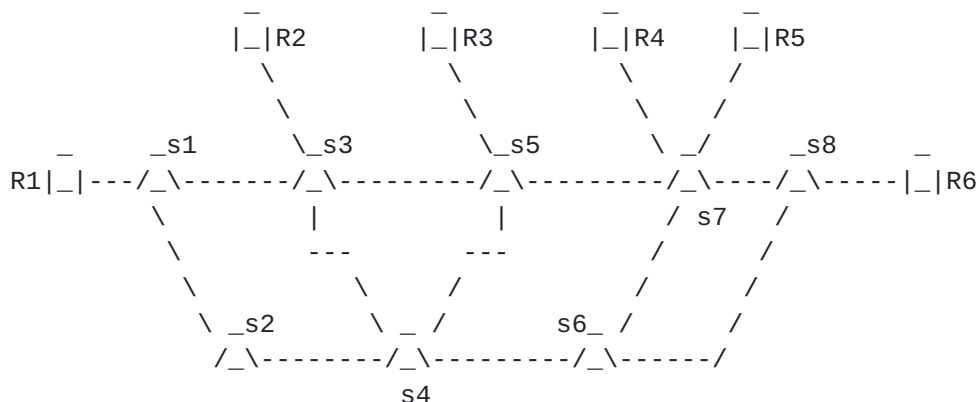
Status of This Memo	i
Abstract	i
1. Overview and Issue	1
<a href="#">1.1</a> . General Design Principles . . . . .	<a href="#">3</a>
2. RSVP Shortcut Interactions	5
<a href="#">2.1</a> . RSVP Control Message Handling . . . . .	<a href="#">5</a>
<a href="#">2.2</a> . Data VC Operations . . . . .	<a href="#">7</a>
<a href="#">2.3</a> . Error Handling Scenarios . . . . .	<a href="#">8</a>
3. Sample RSVP-NHRP Interface	9
4. Interaction with MPOA	11
5. Example of RSVP Shortcut	12



## 1. Overview and Issue

In order to understand the implications of establishing and using shortcuts for RSVP flows across ATM networks, it is helpful to first review how shortcuts are established and used. For purposes of illustration, we focus on the case of the NHRP protocol (Next-Hop Routing Protocol) [[LKPC97](#)]. However, most of the conclusions would also hold when using other shortcut establishment mechanisms, e.g., PAR [[J96](#)].

NHRP is a next hop discovery protocol, that allows progressive discovery of the layer 2 addresses of layer 3 next hops for a given network (IP) destination address. In other words, NHRP allows a network layer protocol such as IP to propagate a query to determine whether, for a given destination (or sets of destinations), some of the network layer routing hops can be bypassed by establishing a shortcut across the layer 2 (ATM) topology. This is illustrated in Figure 1, where ATM switches are represented with triangles, while rectangles are used for IP routers.



Routed path: R1-(s1-s3)-R2-(s3-s5)-R3-(s5-s7)-R4-(s7)-R5-(s7-s8)-R6

Shortcut: R1-(s1-s2-s4-s6-s8)-R6

Figure 1: Routed Path vs Shortcut.

Irrespective of whether they are obtained through NHRP or some other means, the potential benefits of shortcuts are to off-load network layer processing of the packets from the routing hops being bypassed, and also to possibly achieve better delay and/or throughput performance for the data flow between the source and the destination. In the case of RSVP flows, another benefit is the opportunity to



better leverage the layer 2 topology to support QoS requirements, i.e., improve the likelihood that the necessary resources (bandwidth and buffer) are available.

There are two main aspects to the use of ATM shortcuts for RSVP. The first concerns the internal (within the router or the RSVP-ATM gateway) interactions between RSVP and the shortcut establishment mechanism, e.g., NHRP. The second, and possibly more important, aspect deals with the external flows associated with the establishment of shortcuts, and in particular their coexistence with existing uses of shortcut mechanisms, i.e., for non-RSVP flows.

In many implementations, the decision to establish a shortcut for some best effort traffic flow is traffic dependent. For example, a shortcut request is triggered only when the traffic volume to the destination exceeds a certain threshold, and thus warrants the overhead (signaling and support of an additional VC) of setting up the shortcut. In addition, how long the shortcut remains in effect is also typically traffic dependent, so that a shortcut will be taken down if the amount of traffic falls below a given threshold, or after some time-out period. This means that the "routes" associated with shortcuts, in addition to violating network protocol rules by crossing subnet boundaries, are also temporary in nature. As a result, routes associated with shortcuts will normally NOT show-up in the "standard" IP routing table, and will only be reflected in the forwarding information base of the router.

The limited visibility, to IP routing, of shortcuts has some implications for RSVP. Specifically, when RSVP queries IP routing to obtain the NHOP to which to send a packet, the route that is returned is the default (hop-by-hop) path and not the shortcut path. As a result, RSVP control packets will flow on the default hop-by-hop path, while the data packets of the associated flow will be forwarded on the existing shortcut, if any, together with the rest of the default traffic to that destination. However, if and when a reservation is made and the classifier in the IP forwarding fastpath is updated at the ingress router, the RSVP data packets will start being forwarded on the routed path which is where the reservation was made. In some instances, this may have the unfortunate effect that the user may see lower performance (higher delay) once the reservation is in place. This is clearly not desirable, and further motivates the need for allowing RSVP flows to also use shortcuts.

As alluded to earlier, this requires the definition of an interface between RSVP and the shortcut mechanism, that will support the necessary interactions. In particular, RSVP, or the RSVP-ATM gateway, must be able to query the shortcut mechanism, and to request the establishment of a new shortcut VC. In addition, the management



of shortcuts for both RSVP and best-effort traffic to the same destination needs to be coordinated, and this requires additional interactions between RSVP and the shortcut mechanism. Before proceeding with more detailed discussions on each of these aspects, we briefly review some of the general design issues and choices in using shortcuts for RSVP flows.

### **1.1. General Design Principles**

The main design principle followed in this document to support shortcuts for RSVP flows is:

Separate but consistent shortcuts for RSVP data and control messages.

There are several implications of this principle.

1. Receipt of an RSVP PATH message will trigger the establishment of a best-effort shortcut unless one is already in place for the same destination.

Note that in the case where a shortcut is already in place, it may be necessary to verify that this existing shortcut is adequate, i.e., whether or not a more appropriate shortcut might be available for the destination address of the RSVP flow. This implies that for every new flow, RSVP should always check for the most appropriate shortcut to the flow's destination. In the case of NHRP, in order to ensure that the most appropriate shortcut possible is indeed returned, e.g., that the query is propagated as far as possible, some extensions to the current NHRP operation are desirable. Specifically, it would be useful to introduce and use QoS fields in NHRP queries (1) to carry the traffic information (TSpec) present in the RSVP PATH message. This information could then be used at each (NHRP) node as a "hint" to keep forwarding the query, so as to yield the "longest" possible shortcut consistent with the QoS requirements. For example, a bandwidth threshold may be used to decide which flows to forward over the hop-by-hop path and which to send over a separate shortcut.

- 
- 1. Note that such fields were present in earlier versions of the NHRP draft, but were dropped because their use was not fully specified at the time.**



It should be noted that the availability of a best-effort shortcut VC means that the data packets associated with the RSVP flow will also be able, if permitted (more on this in [Section 2](#)), to use the short and, therefore, bypass intermediate router hops as well.

2. In order to allow the eventual establishment of a QoS shortcut VC on which to forward the data packets of an RSVP flow, the PATH messages must be forwarded over a shortcut VC with the same endpoint.
3. Best-effort shortcuts should not be taken down as long as any (associated) RSVP flow remains active, i.e., have a valid path state.

This is required to ensure consistency between the forwarding of RSVP control and data packets. It means that the best-effort shortcut VC must stay in place as long as it is needed to forward RSVP control messages. In particular, the best-effort shortcut VC should not be taken down because of low traffic volume as is often the case in current implementations of shortcut mechanisms, e.g., NHRP. Note that a best-effort shortcut can be used to forward RSVP control messages associated with multiple RSVP flows, i.e., when the associated shortcut end-point is the same for all the flows. As a result, the shortcut must remain in place at least until all the flows have been removed.

4. Upon receipt of a RESV message, a new QoS shortcut VC is setup with the same end-points as the existing best-effort shortcut, but with the appropriate QoS characteristics based on the content of the RESV message.
5. A QoS shortcut VC should remain active as long as the associated RSVP reservation does.

Implicit here is the fact that "management" of a QoS shortcut is closely coupled to the state of the associated RSVP flow. However, responsibility for actually managing the QoS shortcut VCs can lie either with RSVP or with the shortcut mechanism itself. For example, in the case of NHRP which is already responsible for managing best-effort shortcuts, one may want to extend this to also include management of QoS VCs. On the other hand, in order to easily support multiple shortcut mechanisms, it might be desirable to keep management of QoS shortcuts under the responsibility of RSVP, or rather of the "ATM gateway" responsible for establishing VCs on behalf of RSVP. Note that in this case, it is still necessary to maintain coordination between the best-effort and QoS shortcut VCs as outlined above.



In the next two sections, we elaborate on the interactions between RSVP and the shortcut mechanism, in terms of both external and internal exchanges. In [Section 2](#), we specify more precisely the implications of RSVP shortcuts in terms of the associated message flows between entities on each end of the shortcut. In [Section 3](#), we illustrate, for the case of NHRP, a possible interface between RSVP and the shortcut mechanism.

## **[2. RSVP Shortcut Interactions](#)**

This section covers the interactions between hosts and/or routers that take place in order to enable the use of shortcuts for RSVP flows. In particular, it describes how and which VCs are used by RSVP control messages, e.g., PATH and RESV messages, how and when shortcut VCs are established to support RSVP flows, and finally how certain VC setup error scenarios are handled.

### **[2.1. RSVP Control Message Handling](#)**

In order to properly coordinate the forwarding of RSVP data packets on a shortcut VC with the associated RSVP control messages, it is useful to distinguish between downstream and upstream RSVP control messages. According to the RSVP specification [BZB+97], downstream messages, e.g., PATH messages, flow from the sender to the receiver(s), while upstream messages, e.g., RESV messages, flow hop-by-hop (from next hop to previous hop) back towards the sender(s). As mentioned earlier, in this draft we limit ourselves to unicast flows (single ingress and egress points for the ATM network).

Ensuring consistent RSVP reservation states and forwarding of RSVP data packets on a shortcut VC, requires proper coordination between the shortcut end-points and the forwarding of RSVP control messages. In particular, as mentioned in [Section 1.1](#), it is key that downstream messages, i.e., PATH messages, have previous and next hops that match the end-points of the shortcut. For example, in Figure 1, when the shortcut R1-R6 is used for an RSVP data flow, the associated RSVP control messages sent by R1 must be sent directly to R6 without using the intermediate routed path R2-R3-R4-R5. If messages passed in the downstream direction are sent hop-by-hop and the data is sent via a shortcut, the receiver may not be able to properly handle the data and extra resources may be allocated in the intermediate routers R2, R3, R4, and R5. This restriction does not apply to messages sent in the upstream direction, i.e., RESV messages, that can be sent on either the shortcut or the hop-by-hop paths (see [[Ber97a](#)] for details).



Given the importance of ensuring that downstream control messages are delivered directly to the same endpoint as that of the shortcut, we now review the different cases that must be considered and the appropriate actions for each of them. There are three different scenarios that a sender (ATM ingress point) can face:

1. A shortcut to the (session) destination does not exist for best-effort traffic, and establishment of such a best-effort shortcut to this destination is administratively permitted.
2. A shortcut already exists for best-effort traffic to the (session) destination.
3. A shortcut to the (session) destination does not exist for best-effort traffic, and establishment of such a best-effort shortcut to this destination is administratively prohibited.

Note that we assume that administrative control of shortcuts is part of the shortcut support mechanisms, such as NHRP, and is outside the scope of this document. Furthermore, it is assumed that RSVP can be notified when a best-effort shortcut is administratively prohibited.

The first case occurs when no shortcut exists and the sender is administratively permitted to create a shortcut VC for use by best-effort data traffic to the same destination address as the RSVP control message. In this case, there are two possible approaches that the sender can follow. A first approach is to setup the "best-effort" VC at the time the first control message (PATH) arrives, and wait until the best-effort shortcut is established before forwarding the message downstream. The second approach is to immediately forward the control message (PATH) via the hop-by-hop path, and to proceed in parallel with the establishment of the best-effort shortcut. Once the "best-effort" shortcut is established, forwarding of control messages can then be switched over to the shortcut. Downstream RSVP processes will detect the path change and update their internal state appropriately. This latter approach has the disadvantage that additional resources may be allocated along the hop-by-hop path prior to the shortcut being established. In either case, the newly established shortcut VC must be maintained for as long as there remains RSVP control traffic to be forwarded downstream, i.e., the VC should not be timed out or taken down for lack of activity as long as the RSVP PATH state remains in effect.

The second case occurs when a shortcut already exists for use by best-effort data traffic to the same destination address as an RSVP control message, i.e., the entry in the forwarding information base of the router, that corresponds to the destination address of the



RSVP message, indicates that it is a shortcut. In this case, the first step that needs to be performed is to "revalidate" the existing shortcut. As stated earlier, this revalidation is necessary as in some instances the shortcut mechanism may select different end-points for a shortcut depending on the nature of the traffic that is to use the shortcut. Revalidation of the shortcut also depends on the type of shortcut mechanism used. In case the shortcut is not validated, the sender finds itself in a similar situation as that of the previous case. Assuming the shortcut has been validated, the sender can then simply forward downstream RSVP control messages on the existing shortcut VC. However, as before, the sender must also make sure that the existing best-effort shortcut is not released while still being needed to forward RSVP control messages.

The third and last case occurs when no shortcut exists, and the sender is administratively prohibited from creating a best-effort shortcut VC to the same destination address as the RSVP control messages. However, QoS shortcut VCs are allowed and could be used by RSVP flows. For example, this could be the case when crossing administrative domain boundaries. In order to enable the use of QoS shortcuts by RSVP flows in such instances, it is necessary to create a special VC to the shortcut endpoint for use solely by RSVP control messages that are to be sent downstream. As in the first case, the sender can forward the first PATH message downstream either before or after the VC has been successfully established. Similarly, the control shortcut VC must again be maintained for as long as there are RSVP control messages to be passed downstream.

## **2.2. Data VC Operations**

Shortcut data VCs are setup and maintained in essentially the same fashion as described in [[Ber97a](#), [Ber97b](#)]. The only but significant difference is that the ATM called party used in VC setup matches the shortcut endpoint. As in the case of hop-by-hop operation, the setup of a QoS data VC is triggered by the receipt of the first RESV message. The fact that a shortcut will be used can be detected by realizing that the next hop, as per the IP source address in the RESV message, is on a different subnet, and by the fact that a "control shortcut VC" already exists to this next hop. When the RESV message is received by RSVP, a QoS VC to the shortcut endpoint is then initiated.

In establishing the QoS shortcut VC, the RSVP process should follow the same recommendations as stated in [[Ber97a](#)], specifically using independent VCs for each RSVP reservation. The RSVP process should also follow the same recommendations and requirements as stated



in [[Ber97b](#)]. This includes the ability to handle changes in QoS requests and negotiate appropriate encapsulation.

### **2.3. Error Handling Scenarios**

There are a number of error scenarios that need to be handled by RSVP when using shortcuts.

The first error scenario is when the shortcut VC used to forward RSVP control messages is abnormally released and cannot be reestablished. In this case there is no other choice but to send RSVP control messages hop-by-hop. When this occurs and no RSVP associated QoS VCs exists, no other special actions is required. However, when this occurs AND there are RSVP associated QoS VCs (there could be more than one), then those VCs MUST be released. The release of the associated data VCs is required to maintain the synchronization between forwarding and reservation states for the associated data flows.

A second error scenario occurs when the setup of the QoS data VC associated with an RSVP flow fails. In this case, a shortcut with the required QoS is not available, but a best-effort VC has been successfully established to forward the RSVP control messages. Since the desired QoS cannot be satisfied using a shortcut, the sender SHOULD stop sending the RSVP control messages over the best-effort shortcut and start sending them on the hop-by-hop path. This will eventually allow the reservation to be attempted on the hop-by-hop path. The best-effort shortcut VC used for control messages may be released or maintained based on its other uses. For example, it may still be used to forward RSVP control messages associated with other flows which did successfully setup a QoS shortcut to the same endpoint. Note that the sender may want to regularly retry establishing a shortcut based path, especially if the reservation is also unsuccessful on the hop-by-hop path.

The last error scenario to consider is that of a VC setup failure when processing a change in an RSVP reservation. This case is handled exactly the same way as is described in [[Ber97b](#), [Section 2.4](#)]. The old QoS VC MUST continue to be used. When the new reservation is greater than the old reservation, the reservation request MUST be answered with an error. When the new reservation is less than the old reservation, the request MUST be treated as if the modification was successful. Implementations SHOULD retry replacing the too large VC after some appropriate elapsed time.

In the next section, we provide an example of a possible interface between RSVP and the shortcut mechanisms, when NHRP is assumed for



the latter. This interface is only intended as a possible guideline and not meant to imply specific implementations requirements.

### 3. Sample RSVP-NHRP Interface

An interface between RSVP and NHRP should allow determination of appropriate shortcut endpoints for different RSVP flows, support coordination between best-effort and QoS VCs used for the same flow, and accommodate handling of error scenarios. In the context of the interface outlined in this section, we assume that RSVP (the RSVP-ATM gateway) is responsible for the establishment of QoS shortcut VCs. As a result, interactions between NHRP and RSVP are limited to dealing with best-effort VCs.

- RSVP\_shortcut\_resolve(flow\_ID, IP\_address, TSpec, return\_code, BE\_VC\_handle, endpoint\_ATM\_address)

This is a query from RSVP to NHRP to request the most appropriate shortcut possible towards reaching IP\_address. Note that if a shortcut already exists to the destination, NHRP may need to revalidate it based on the new TSpec specified in the call.

- \* flow\_ID: identifies the flow for future reference.
- \* IP\_address: unicast IP address of the destination specified in the PATH message.
- \* TSpec the traffic specification for the RSVP flow. This information may be carried in the QoS field of the NHRP request.
- \* return\_code: because resolution of the query and establishment of a new VC, if necessary, will take time, the return\_code can take three different values: success/pending/failed.
- \* BE\_VC\_handle: this value is only meaningful when return\_code = success, and identifies to RSVP the best-effort VC associated with the flow. Implicit here is the fact that both RSVP and NHRP are now aware that the states of the VC associated with BE\_VC\_handle and of the RSVP flow identified by flow\_id need to be coordinated. Note that a best effort VC could be associated with multiple flows, and NHRP could return the same or different handles for each flow.
- \* endpoint\_ATM\_address: ATM address of the endpoint to which the best-effort VC was established. RSVP will use this



address to subsequently request the setup of a QoS VC upon receipt of an associated RESV message.

- NHRP\_shortcut\_resolve(flow\_ID, return\_code, BE\_VC\_handle, endpoint\_ATM\_address)

This is an asynchronous response to an earlier query by RSVP, i.e., one for which return\_code = pending.

- \* flow\_ID: used by RSVP to correlate the response with the original query.
  - \* return\_code: is either success or failed.
  - \* BE\_VC\_handle: in case return\_code = success, identifies to RSVP the best-effort VC associated with the flow. Again, implicit here is the fact that both RSVP and NHRP are now aware that the states of the VC associated with BE\_VC\_handle and of the RSVP flow identified by flow\_id need to be coordinated. Note that a best effort VC could be associated with multiple flows, and NHRP could return the same or different handles for each flow.
  - \* endpoint\_ATM\_address: ATM address of the endpoint to which the best-effort VC was established. RSVP will use this address to subsequently request the setup of a QoS VC upon receipt of an associated RESV message.
- RSVP\_shortcut\_refresh (flow\_id, BE\_VC\_handle, return\_code)
- Used by RSVP to notify NHRP that an existing best-effort shortcut VC, previously established for a given RSVP flow, should be maintained independent of the volume of traffic it carries. Note that this function need not be supported through such an interface, and could be provided through local status bits that NHRP could query.
- \* flow\_id: initially assigned by RSVP to identify the flow.
  - \* BE\_VC\_handle: handle provided by NHRP after establishing the best-effort VC.
  - \* return\_code: can be success or an error code (tbd) in case RSVP is trying to refresh a non-existing VC.
- RSVP\_shortcut\_take\_down (flow\_id, BE\_VC\_handle)



Used by RSVP to notify NHRP that an existing best-effort shortcut VC, associated with a given flow, can be taken down. This need not trigger the actual take down of the VC in cases where the same best-effort VC is used to carry control messages of multiple RSVP flows, or if NHRP decides to keep the VC up to continue carrying best-effort data traffic. However, in the latter case, NHRP is responsible for removing any association between RSVP flows and the best-effort VC.

- \* flow\_id: initially assigned by RSVP to identify the flow.
  - \* BE\_VC\_handle: handle provided by NHRP after establishing the best-effort VC.
- NHRP\_BE\_VC\_failure(BE\_VC\_handle, error\_code)

Used by NHRP to notify RSVP of the failure of an existing best-effort VC. RSVP is responsible for taking down all associated QoS VCs.

- \* BE\_VC\_handle: handle originally used by NHRP to identify the best-effort VC.
- \* return\_code: indicates the type of failure if known.

#### **4. Interaction with MPOA**

In the ATM Forum's Multiprotocol Over ATM (MPOA) model, shortcuts are established between MPOA Clients (MPCs) on edge devices, or between an MPC and an NHRP Client (NHC). When an internetwork layer packet (e.g., IP packet) arrives over a shortcut VC to an MPOA edge device, the MPC looks up the internetwork layer destination address in the packet in a local cache to obtain DLL information to place on the packet. The DLL information is prepended to the packet and the frame is transferred to a LAN Emulation client for further handling. When using the scheme described in this draft for establishing shortcuts with QoS, it is possible that multiple shortcuts may be established to the same MPC with traffic from different RSVP flows to the same IP destination. Furthermore, it may be desirable to place different DLL headers for packets arriving over different VCs (e.g., with different values of priority in the token ring header, or in the IEEE 802.1p header), depending on the QoS required for the flow. The MPOA specification already allows for this possibility. The egress MPC simply needs to assign a different "tag" for each RSVP flow that it is terminating. This tag is carried in the packets arriving over the shortcuts to the edge device, and may be used to index in to



different entries in the local cache. Thus, the scheme described in this draft is compatible with MPOA as well.

It is for further study whether full PATH state needs to be stored on the edge devices or not. The MPOA Server (MPS) may be able to store the PATH state, thus allowing for building of simpler edge devices.

## 5. Example of RSVP Shortcut

In this last section, we briefly outline a possible set of events involved in the establishment of a shortcut for an RSVP flow. Referring to Figure 1, we assume a sending host H1 connected through a series of networks and routers to router R1 (ingress router to the ATM network), and a destination host H2 connected to router R6 (egress router from the ATM network) through again possibly a series of routers and networks.

- Host H1 starts sending PATH messages.
- First PATH message arrives at R1 which extracts the IP address of H2 and the TSpec of the flow and queries its local NHRP using `RSVP_shortcut_resolve()`.
- NHRP generates an NHRP query for the IP address of host H2 and specifies using the NHRP QoS field, that the query should be resolved in the most specific and feasible manner, i.e., the ATM address of router R6 should be returned. NHRP responds to the RSVP query with a `return_code = pending`.
- NHRP receives a response to its query. It could either correspond to an existing VC to R6 or require the establishment of a new VC. If the latter, NHRP proceeds with setting-up a new best-effort VC.
- NHRP informs, using `NHRP_shortcut_resolve()`, that the best effort VC to R6 has been established (or not established) by returning the appropriate `return_code`. In what follows, we assume the best effort VC has been successfully established, so that NHRP also returns the ATM address of R6 to RSVP and the handle corresponding to the VC.
- RSVP starts forwarding PATH messages on the newly establish shortcut VC to R6. Here we assume that R1 held back forwarding of PATH messages until after the establishment of the best-effort shortcut VC. Note that data packets for H2 will also be forwarded on the shortcut VC.



- PATH message arrives at R6 with its PHOP set to R1. R6 creates the appropriate path state and proceeds to forward the PATH message towards H2.
- H2 receives the PATH message. It decides to make a reservation and starts sending RESV messages.
- First RESV message arrives at R6 which (assuming the local reservation is successful), which forwards it to R1 since it has the IP address of R1 in its PHOP field.
- RESV message is received by RSVP at R1, which then requests the establishment of a QoS VC to R6 using the ATM address returned by NHRP in `NHRP_shortcut_resolve()`. The request specifies a service class and traffic parameters based on the information contained in the RESV message.
- Once notified of the success of the setup of the VC, RSVP proceeds to update the classifier at R1 to ensure that RSVP data packets are forwarded on the new VC. R1 also forwards the RESV message towards H1.
- As long as the associated RESV state remains alive, RSVP at R1 keeps the QoS VC to R6 up and also makes sure that the associated best-effort VC remains up.
- When the reservation is eventually taken down, say, by H2 issuing a `RESV_TEAR`, RSVP at R1 takes down the QoS VC to R6 and removes the associated classifier entry. Data packets start then being forwarded on the best effort shortcut VC to R6.
- As long as the PATH state remains alive, RSVP at R1 keeps refreshing the associated best-effort shortcut VC to R6 using `RSVP_shortcut_refresh()`.
- When the PATH state is removed, e.g., by receiving a `PATH_TEAR` initiated by H1, RSVP at R1 notifies NHRP using `RSVP_shortcut_take_down()` with the `flow_id` and `VC_handle` corresponding to the best effort shortcut VC associated with the flow. NHRP can then decide to either take the VC down or to keep it up.

#### Acknowledgment

The authors gratefully acknowledge inputs from Steve Nadas, Anoop Ghanwani, Steven Blake, Girish Bhat, and Ed Bowen.



## References

- [BB97] S. Berson and L. Berger. IP Integrated Services with RSVP over ATM ([draft-ietf-issll-atm-support-03.txt](#)). INTERNET-DRAFT, Internet Engineering Task Force, ISSLL Working Group, March 1997.
- [Ber97a] L. Berger. RSVP over ATM Implementation Guidelines ([draft-ietf-issll-atm-imp-guide-01.ps,txt](#)). INTERNET-DRAFT, Internet Engineering Task Force, ISSLL Working Group, July 1997.
- [Ber97b] L. Berger. RSVP over ATM Implementation Requirements ([draft-ietf-issll-atm-imp-req-00.ps,txt](#)). INTERNET-DRAFT, Internet Engineering Task Force, ISSLL Working Group, July 1997.
- [GB97] M. Garrett and M. Borden. Interoperation of Controlled Load and Guaranteed Service with ATM ([draft-ietf-issll-atm-mapping-02.txt](#)). INTERNET-DRAFT, Internet Engineering Task Force, ISSLL Working Group, March 1997.
- [J96] J. Jeffords (ed.). PNNI Augmented Routing (PAR) v1.0 Specification. ATM Forum BTD-PNNI-PAR-01.00, December 1996.
- [LKPC97] J. V. Luciani, D. Katz, D. Piscitello, and B. Cole. NBMA Next Hop Resolution Protocol NHRP ([draft-ietf-rolc-nhrp-11.txt](#)). INTERNET-DRAFT, Internet Engineering Task Force, ROLC Working Group, March 1997.
- [BZB+97] R. Braden (Ed.), L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource reSerVation Protocol (RSVP) version 1, functional specification ([draft-ietf-rsvp-spec-16.ps](#)). INTERNET-DRAFT, Internet Engineering Task Force - RSVP WG, June 1997.



## Authors' Address

Roch Guerin  
IBM T.J. Watson Research Center  
P.O. Box 704  
Yorktown Heights, NY 10598

EMAIL: [guerin@watson.ibm.com](mailto:guerin@watson.ibm.com)  
VOICE +1 914 784-7038  
FAX +1 914 784-6205

Liang Li  
IBM Corporation  
P. O. Box 12195  
Research Triangle Park, NC 27709

EMAIL: [lli@raleigh.ibm.com](mailto:lli@raleigh.ibm.com)  
VOICE: +1 919 254-5627  
FAX: +1 919 254-5483

Lou Berger  
FORE Systems  
6905 Rockledge Drive  
Suite 800  
Bethesda, MD 20817

EMAIL: [lberger@fore.com](mailto:lberger@fore.com)  
VOICE: +1 301 571 2534

Dilip Kandlur  
IBM T.J. Watson Research Center  
P.O. Box 704  
Yorktown Heights, NY 10598

EMAIL: [kandlur@watson.ibm.com](mailto:kandlur@watson.ibm.com)  
VOICE +1 914 784-7722  
FAX +1 914 784-6205

Vijay Srinivasan  
IBM Corporation  
P. O. Box 12195  
Research Triangle Park, NC 27709

EMAIL: [vijay@raleigh.ibm.com](mailto:vijay@raleigh.ibm.com)  
VOICE: +1 919 254-2730  
FAX: +1 919 254-5410

