# Quality of Service Support for Recording and Playback of MBone Sessions in Heterogeneous IP/ATM Networks

*Jens Schmitt*[1], *Michael Zink*[1], *Lars Wolf*[1], *Ralf Steinmetz*[1,2]

1
Industrial Process and System Communications
Department of Electrical Engineering and Information Technology
Technical University of Darmstadt
Merckstr. 25 • D-64283 Darmstadt • Germany
2
GMD IPSI
German National Research Center
for Information Technology
Dolivostr. 15 • D-64283 Darmstadt • Germany

email: {Jens.Schmitt,Michael.Zink,Lars.Wolf,Ralf.Steinmetz}@kom.th-darmstadt.de

## Abstract

Many video and audio sessions are transmitted via the Multicast Backbone (MBone) medium day by day. The Integrated Services architecture of the Internet will make it now possible to transmit and receive these sessions in a 'guaranteed' quality of service. To allow this in heterogeneous networks, consisting of, e.g. Ethernet and ATM, an interaction between the QoS architectures of the Internet and ATM is necessary. However, they are very different, so mapping them is a difficult task.

In this paper, the implementation of an interaction approach for the QoS architectures developed for the Internet and for ATM, and the extension of the MBone VCR for recording and playing back MBone sessions in a higher quality, is described. This system allows the QoS supported recording and off-line playback of MBone session in heterogeneous IP/ATM networks.

**Keywords**: Heterogeneous Networks, IP/ATM, Integrated Services, QoS, ATM, MBone, MBone VCR, RSVP over ATM.

## 1 Introduction

The MBone VCR [Holf95] was one of the first MBone tools that allowed the recording and offline playback of sessions that were transmitted over the MBone. This allows users to record sessions, which they are not able to watch live, and play them back at any time. An interesting application of this tool is for example the recording of a lecture at a University that was transmitted via the MBone, and playing back that session at a different location, at a different time and as often as wanted.

With the current version of the MBone VCR [RVIC97] sessions can only be recorded and played back in best-effort quality. Using the Internet Integrated Services (IIS) [BrCS94], it is possible to reach a higher quality for transmissions on the Internet. Therefore it was our goal to extend the functionality of the MBone VCR to use these new services. This extended MBone VCR makes it possible to cooperate with other MBone tools that were extended for the use of the IIS, like the rsvp-vic [MaMS97], [RVIC97].

The new services are implemented in the network layer (packet level), but we believe in order to make them work efficiently and effectively it is necessary that they are supported by a QoS-active link layer. In our point of view ATM will become more and more popular in the backbones of MANs and WANs, and becomes thus an important link layer technology from the Internet perspective. This trend implies that in the near future ATM and IP networks will coexist and ATM networks will be

This work is supported in part by Deutsche Telekom AG, Darmstadt, Germany, and by Volkswagen-Stiftung, Hannover, Germany.

used to interconnect IP networks. To provide QoS support in heterogeneous IP/ATM networks, an interaction between the QoS architectures of the Internet and ATM has to be implemented.

We decided to implement ‘RSVP over ATM’ to solve this problem, that means we based our design on an extension of the Classical IP over ATM model [Laub94]. By overlaying the RSVP on an ATM network it is possible for IP applications to request a certain QoS through the use of RSVP. The information delivered by RSVP is used by the ATM subnetwork to setup a VC whose QoS characteristics should approximate the ones signalled by RSVP as closely as possible. An alternative would have been to use native ATM services for the QoS extension of the MBone VCR. Obvious drawbacks of this approach would have been that the ATM-capable MBone VCR could only run on ATM workstations, and that the whole data communication part of the MBone VCR would have had to be rewritten. In contrast to this, our approach of extending the MBone VCR by RSVP leaves the data communication part unchanged and furthermore does not restrict the use of the MBone VCR to a homogeneous ATM network.

In our prototypical system the MBone VCR signals the desired QoS to RSVP which is afterwards mapped to ATM QoS parameters. Depending on the ATM QoS parameters a QoS VC will be set up on which the appropriate data is sent.

While we give an overview about related work in the next section, section 3 deals with our implementation of RSVP over ATM. The fourth section describes how the MBone VCR had to be enhanced to become an RSVP-aware application and in section 5 the interaction of all components of our prototypical system is illustrated by some typical scenarios. Section 6 then summarizes our work and gives some thoughts on possible future work.

## 2 Related Work

To satisfy the requirements of distributed multimedia applications in the Internet, as for example the MBone tools, the Internet Integrated Services (IIS) were introduced. These new services shall support hard real-time (guaranteed service) [RVIC97] and adaptive real-time (controlled load) [ShPG97] applications like e.g. vic. The provisioning of QoS for applications is done by the Resource reSerVation Protocol (RSVP) [BZBH+97], an extension of the Internet protocol suite.

Several tools for transmitting and receiving data via the MBone have been developed in the last few years (vic, vat, nv, ivs, sdr, wb,...). Unfortunately, today most of them are not able to work with the new services provided by IIS. One of these extended applications which support RSVP is the well known vic [RVIC97]. At present it is only possible to have high quality sessions on the MBone between a sending and a receiving rsvp-vic. A user can only receive live sessions with a desired QoS. Playing already stored sessions or recording sessions with a desired QoS with the available MBone tools is not possible at the moment.

Since ATM is becoming more and more important in the WAN and backbone area there has already been done some work in the field of interaction between the QoS architectures of the Internet and ATM, e.g. some architectural considerations have been made [BCDB95] [CBBB+97][SKB96], implementation guidelines have been developed [Berg97], and a few RSVP over ATM implementations have been conducted. One of these implementations is described in [BFGK95] which is suitable for the handling of a large number of RSVP receivers. Another implementation is described in [BrGi96] which however in contrast to the one above does not support multicast. These two implementations are taking the approach to map one IP Flow to one ATM VC which differs from a third implementation in which an IP flow multiplexing is performed [MaMS97]. This multiplexing allows the combination of several IP flows into one VC.

The main approach of all of this implementations was to extend the *Classical IP over ATM* model [Laub94] to support the transport of IP flows over ATM networks.

## 3 Implementation of RSVP over ATM

It was our goal to develop an RSVP over ATM implementation that allows applications running on top of IP the reservation of resources in a heterogeneous network environment. We decided to use the Solaris 2.5.1 implementation of RSVP [RSVP97] as a basis for our RSVP over ATM implementation and to extend it with an appropriate interface to ATM. Figure 1 illustrates the environment in which such an implementation is used.The RSVP over ATM implementation resides in the so-called Edge-Devices, which have connectivity to both worlds and therefore have to ’translate’ between themSection 3.1 describes the differences between RSVP and ATM signalling. How the mapping between the different QoS models in heterogeneous IP/ATM networks is achieved will be described in section 3.2. In section 3.3 the underlying model of our RSVP over ATM implementation is presented..

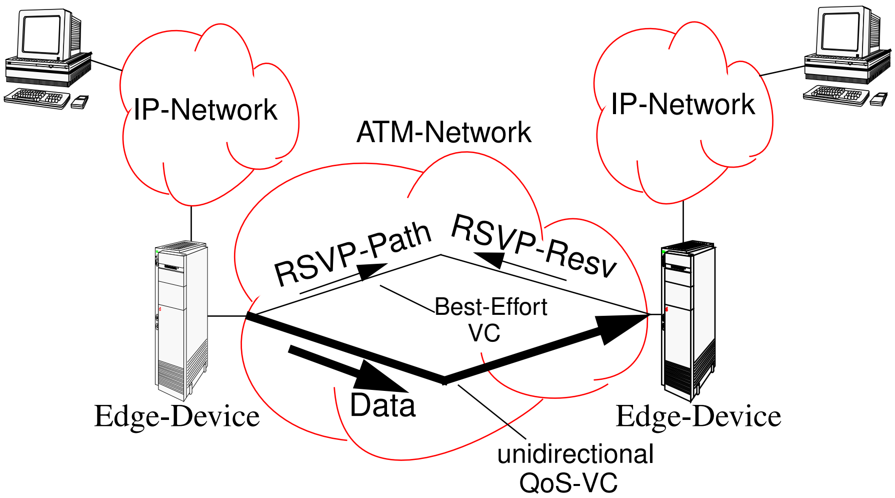


*Figure 1:* RSVP over ATM.

### 3.1 Differences between RSVP and QoS signalling in ATM

The RSVP and QoS signalling in ATM [ATMF96a][ATMF96b] are very different. These differences complicate any RSVP over ATM implementation:

- **Initiation of a Reservation**
  The actual reservation in RSVP is performed when the *Resv* message arrives which is initiated by the receiver. In opposite to RSVP a reservation in ATM is done by the sender during VC setup.
- **State in the Network Nodes**
  In RSVP a route between sender and receiver can change and RSVP will adapt to this route change (soft state). If by any reason a route in ATM has to be changed, the connection has to be cancelled and a new one is setup (hard state).
- **Time of Reservation**
  In ATM data transmission and resource reservation are synchronized, i.e. the reservation is done during VC setup. In contrast to ATM a reservation in RSVP can be done asynchronously to data transmission.
- **Variation of an Existing Reservation**
  If a reservation has to be changed in ATM the existing VC must be closed and a new one has to be set up. In RSVP a reservation can be changed during data transmission. RSVP allows the dynamic change of QoS which is in contrast to ATM where the requested QoS is static.
- **Uni/bidirectional Flows/Connections**
  An RSVP reservation is always for unidirectional flows which means that a reservation is performed for data flowing from a sender to a receiver. ATM reservations are bidirectional for a data connection from a sender to a receiver and vice versa.
- **Heterogeneity**
  In opposite to RSVP heterogeneous receivers are not supported in ATM because the signalling allows only point-to-multipoint VCs with a uniform reservation (homogenous QoS)

Table 1 summarizes the differences between RSVP and QoS signalling in ATM..

| Characteristic | RSVP | ATM |
|---|---|---|
| Reservation style | Receiver-oriented | Sender-oriented |
| State in nodes | Soft state | Hard state |
| Reservation setup | Independent from connection setup | Synchronous with connection setup |
| Modification of existing reservations | Dynamic | Static |
| Uni-/bidirectional | Unidirectional | Bidirectional |
| Heterogeneity | Support for heterogeneous receivers | Same QoS for all receivers |

**Table 1: Differences between RSVP and QoS signalling in ATM**

One of the tasks of an RSVP over ATM implementation is to solve the problems that are caused by these differences.

### 3.2 QoS Support

In order to provide QoS support for applications running on top of TCP/IP or UDP/IP on heterogeneous IP/ATM networks a mapping between IIS' QoS and the ATM QoS parameters must be implemented in the Edge-Devices. With this mapping it is possible to setup a VC which characteristics comply to the ones signalled by RSVP. The RSVP over ATM implementation has to perform the following steps for the mapping between the different QoS parameters:

- Mapping between the different IIS and ATM service classes as described in [GaBo97] and shown in Figure 2.
- Translation between the values of the QoS parameter from bytes (IIS) to cells (ATM).

The mapping shown in Figure 2 can only be seen as the 'theoretical optimum' because VBR and ABR are not yet implemented in most of the ATM hardware.

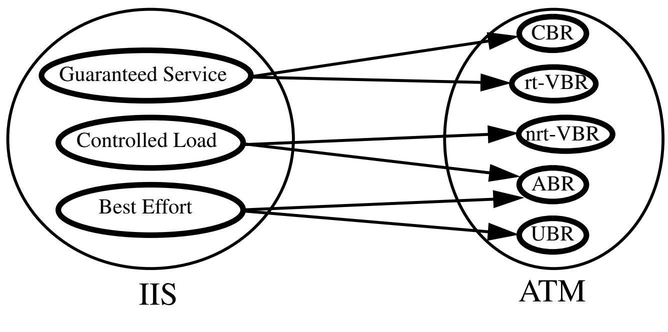


*Figure 2:* Mapping of QoS Service Classes.

### 3.3 Architecture of the RSVP over ATM implementation

We extended RSVP by an interface to ATM which is responsible for the VC management. With this new interface, a flow, for which a reservation has been made, will be sent over a VC with a QoS corresponding to the one signalled by the RSVP reservation (QoS-VC) instead of a *Best-Effort*-VC as in the *Classical IP over ATM* model. That means cell-level scheduling is used in order to avoid the duplication of functionality already provided by ATM in the network/IP layer, i.e. we try to maximize the use of ATM facilities in order to alleviate the task of the network/IP layer.

In our implementation we used the Fore-API [FORE95] to realize this interface including the VC management. The library routines of the Fore API provide a connection-oriented client and server model. For transmitting data between client and server a connection has to be established first. Therefore we also had to implement a so-called ATM-Connection-Server that is responsible for the connection setup and receiving of data on the egress Edge-Device. The client of this design is part of the RSVP over ATM implementation.

The steps that have to be performed for a reservation and data transmission of a flow are as follows:

- If a *Resv* message arrives at the ingress Edge-Device, a QoS-VC to the egress Edge-Device is setup.
- After a successful setup of the QoS-VC data belonging to the flow will be sent over this VC and no longer over Classical IP's Best-Effort-VC.
- On the egress Edge-Device the ATM-Connection-Server transfers the data to the traditional LAN interface (e.g. Ethernet, Token Ring), in case the application is not running on the ATM interface of the Edge-Device.

Figure 3 illustrates the setup and disconnection for an RSVP reservation.

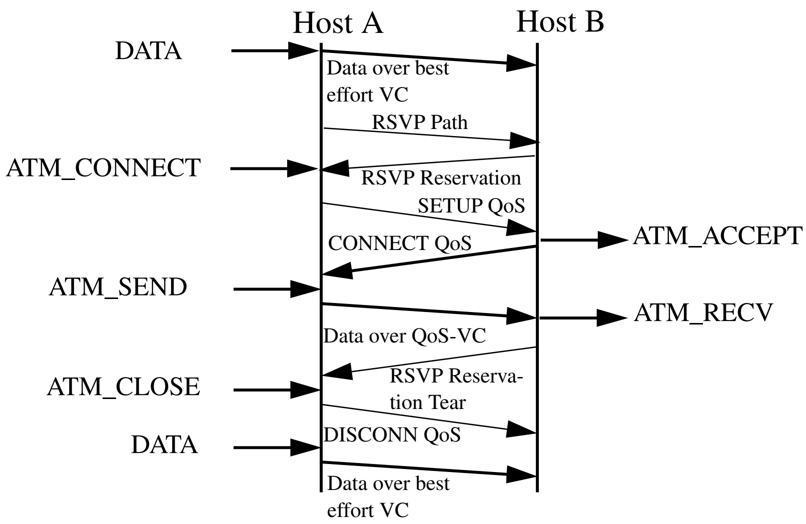


*Figure 3:* Setup and Disconnect of an RSVP Reservation.

To identify data that is belonging to a flow, a table exists in each Edge-Device. A new entry will be inserted in the table if a reservation arrives at the Edge-Device. Since there is no flow-label in the IPv4 header this entry contains the source and destination address, the source and destination port number and also the protocol ID of the flow. If the reservation is torn down the corresponding entry in the table will be deleted. This table is used by the *datamanagement* that is running on the Edge-Device to find out whether an IP packet arriving on the Ethernet interface belongs to a flow or not.

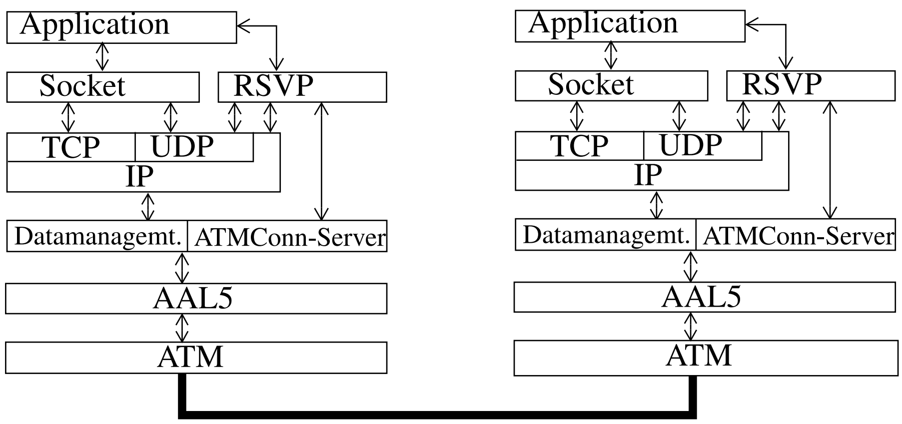


*Figure 4:* Extended Architecture.

The interaction between the extended RSVP implementation, the ATM-Connection-Server and the *datamanagement* in the protocol hierarchy is shown in Figure 4. In contrast to other RSVP over ATM [MaMS97] implementations we decided not to implement the VC management into the CBQ[FlJa95], the packet classifying and scheduling module supplied with [RSVP97], because ATM has its own traffic management which performs almost the same procedures as implemented in the CBQ. However in ATM it is done in hardware on the cell-level and is therefore much more efficient than a software scheme. The only advantage of a CBQ module ‘in front of’ ATM could be the possibility of doing a two-level scheduling. However, it was not our aim to investigate different strategies to assign groups of RSVP flows to ATM VCs and therefore we chose a one-to-one mapping between RSVP flows and ATM VCs.

The mapping between the IIS and ATM service classes could not be realized as described in chapter 3.2 because the Fore API does not support the reservation of resources in terms of UNI 3.0/3.1 or 4.0 service categories and parameters, but uses its own, simplified QoS model. For requesting resources three different parameters are offered by the Fore API. The values of these parameter have to be specified in kbits/s which makes a translation between bytes and cells unnecessary. On the table below the mapping between the API and IIS parameters is shown.

| Fore API | Guaranteed Service | Controlled Load |
|---|---|---|
| peak_bandwidth | guaranteed rate (R) | token bucket rate (r) |
| mean_bandwidth | token bucket rate (r) | token bucket rate (r) |
| mean_burst | token bucket depth (b) | token bucket depth (b) |

**Table 2: Mapping of QoS parameters.**

For signalling the desired QoS to RSVP the applications can simply use an API which acts as an interface to the RSVP daemon. How the interaction between the application and RSVP is realized will be shown in section 4.2.

## 4 The Extended MBone VCR

The MBone VCR is an application that allows the recording and playback of video and audio sessions that are transmitted via the MBone. But with the current version of the MBone VCR it is not possible to use the Internet Integrated Services which make it possible to record or playback data in a higher quality than best effort. However, especially for the recording case a guaranteed quality of service is very desirable, since the recorded session might be played back many times and can only be recorded once as is the case for example for a lecture. Due to these considerations we decided to implement an interface to RSVP into the MBone VCR. Section 4.1 gives an overview of the RSVP API and in section 4.2 the extended VCR is described.

### 4.1 RAPI the RSVP API

The RAPI is the standard application programmers interface of RSVP that can be used by applications for the invocation of RSVP services [BrHo97]. In other words, with RAPI applications have the opportunity to request QoS. RAPI is implemented as a client library that can be linked with an application. The most important functions contained in this library are shown below:.

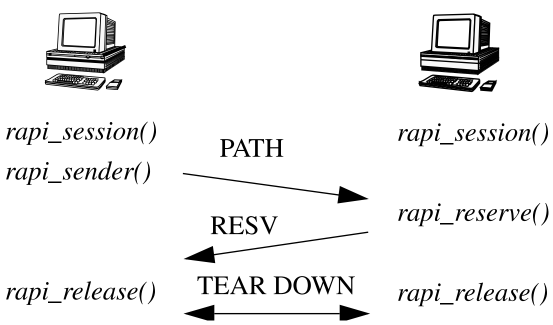


*Figure 5:* RAPI Functions.

- *rapi_session():* creates a RAPI Session.
- *rapi_sender():* a sending application must issue a *rapi_sender()* call to initiate the sending of an RSVP *Path* message.
- *rapi_reserve():* a receiving application uses *rapi_reserve* for making a reservation.
- *rapi_release():* removes an existing reservation.

In Figure 5 it is shown which functions have to be called by a sending and a receiving application.

### 4.2 .QoS Functions of the VCR

With the RAPI described above it is possible to give the VCR the functionality to record or playback sessions with a requested QoS. Because of the two modes of the VCR (sending or receiving data) it has to be possible to initiate either a *Path* or a *Resv* message. The extended VCR has the functionality of both the sender and receiver as they are shown in Figure 5. Its user interface is shown in Figure 6.

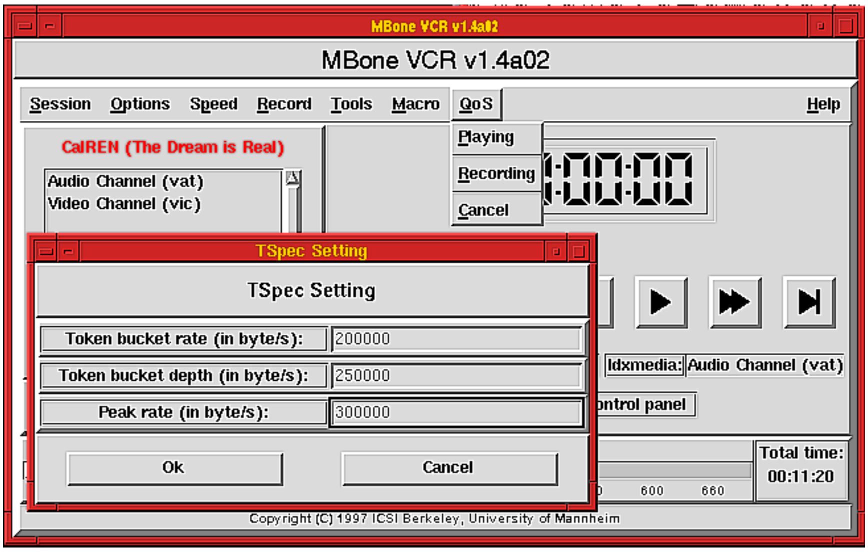


*Figure 6:* User Interface of the Extended VCR.

It was our intention that the request for QoS should be user-driven and not automated by the application (at least as far as the user can give more accurate information). Therefore a new item (*QoS*) was inserted in the top menu of the VCR as shown in Figure 6. By inserting this new item it is the users decision whether the quality for the session is adequate without a QoS request or not. The menu *QoS* includes the topics *Playing* and *Recording,* which are essentially synonyms for initiating a *Path* or *Resv* message. By choosing *Playing* the user gets a menu in which he or she is able to determine the QoS which is required to receive the data in the best possible quality. At the moment there are three parameters that can be determined by the user: *token bucket rate, token bucket depth* and *peak rate*. The furtherly needed parameters are automatically set by the implementation. We felt that it would make no sense to let the user determine all parameters which are needed for a *TSpec/RSpec,* since parameters like minimum policed unit, maximum packet size or slack are meaningless to the user. One could argue that this might also be the case for the user-selectable parameters mentioned above, however it should be pointed out that our work was not concerned with finding an adequate QoS user interface which allows users to request QoS in a very simple and intuitive way. We are aware that there is at least one layer of abstraction missing in order for a layman user to cope with the QoS parameters setting.

If the topic *Recording* is chosen almost the same window as in the *Playing* case will be shown. But in this case the fields *token bucket rate, token bucket depth* and *peak rate* are initialized with the values received from a *Path* message. These values can be seen as a proposal which can be edited by a user if he or she wants to request a higher or lower QoS.After insertion of the three parameters a *Path* or *Resv* message will be initiated through the extended VCR and signalled to RSVP by a RAPI call. In the case of a *Path* message the addresses of the receivers will be determined from the session information that is stored with each session. A third topic in the *QoS* menu is called *Cancel*, which allows a user to cancel an existing QoS request.

In addition to the extended VCR another RSVP capable application is necessary to allow QoS supported sessions. As mentioned above, one of these already extended applications is the rsvp-vic, which allows the request for QoS in a way that it takes the values included in the received *Path* message to form a *Resv* message.

## 5 Scenarios

To illustrate the interaction between the extended VCR, the RSVP daemon, our RSVP over ATM implementation, and the rsvp-vic, we describe some typical interactions. In each case we used the rsvp-vic as sending or receiving application to communicate with the extended VCR. In the case that the rsvp-vic recognizes that it is running on an RSVP-supported computer a *Path* message will be automatically setup in the sending case. In contrast to the extended VCR the parameters for this message can not be set by the user. In the receiving case a *Reserve* button exists in the actual window that allows the user to generate a QoS request. The parameters for the *Resv* message are adopted from the incoming *Path* message, again without invocation of the user.

### 5.1 VCR in Playing Mode

In the first scenario (scenario A, Figure 7) the VCR acts as a sender that distributes a session which was recorded before. With the extended VCR there is now the opportunity to distribute this session with a certain QoS. Before or during the session the user is able to generate an appropriate *Path* message, that will be propagated to all possible receivers. Depending on the IP address that is stated in the session information of a recorded session the *Path* message will be distributed as unicast or multicast message. The IP address can be edited and therefore it can be determined by the user if this session should be transmitted as unicast or multicast. The *Path* message generated by the VCR will be forwarded from RSVP router to RSVP router until it reaches its destination. In the case of an ATM backbone in between, the *Path* message will be transmitted via Fore IP [FORE96] (a proprietary modified version of *Classical IP over ATM*) over the existing best effort VC. In the ingress and egress Edge-Devices no further processing on the *Path* messages has to be done than in a standard RSVP router.

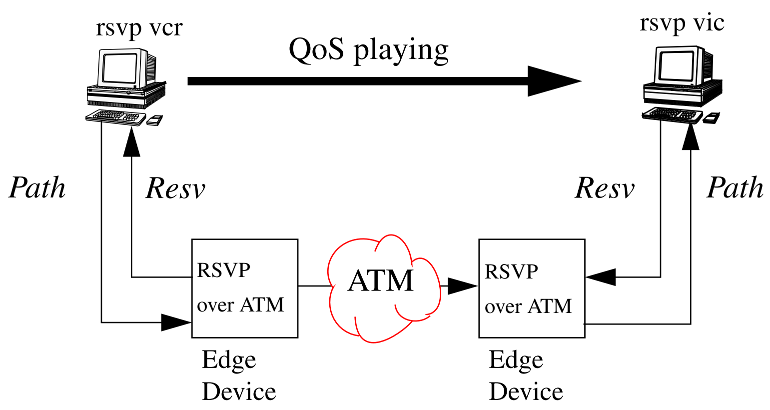


*Figure 7:* Scenario A.

By using rsvp-vic as receiving application a user can press the *Reserve* button that was described above which will initiate a *Resv* message and should consequently raise the quality of the transmitted session. It has to be seen as a disadvantage that the rsvp-vic directly adopts the information received from the *Path* message because the user has no chance to manipulate this information that results in the fact that he or she is not able to change the QoS even if it is not satisfying. By using this strategy, i.e. rsvp-vic choosing exactly what the sender tells him, RSVP's capabilities to support heterogeneous, receiver-oriented QoS is essentially destroyed. After initiating a *Resv* message by RAPI this message will be sent upstream towards the sender. In each router in between the requested resources will be reserved if possible. On the transition between IP and ATM the message will also be sent over the best effort VC as it is done with the *Path* message. After the *Resv* message has reached the ingress Edge-Device an appropriate QoS-VC will be setup as described in section 3.3. On the egress Edge-Device no further processing has to be performed by the RSVP over ATM implementation because the reservation has to be done on the Ethernet interface.

### 5.2 VCR in Recording Mode

In scenario B (Figure 8) the recording function of the VCR is used. With the extended VCR there will be the opportunity to record a session in a higher quality than best effort. To perform this operation the recordable data must be sent from an rsvp-capable application like the rsvp-vic. Now the roles of rsvp-vic and extended VCR are interchanged. The rsvp-vic is the sender that distributes video information from e.g. a video grabber or a camera. As described above the rsvp-vic automatically initiates a *Path* message when the distribution of data is started. The *Path* message will be handled in the same way as shown in scenario A. When the *Path* message reaches the receiver this will be signalled to the VCR by a RAPI upcall. Now it is possible to initiate a *Resv* message by the use of the new implemented features described in section 4.2. The user has the chance to accept the *TSpec* parameters that were offered by the *Path* message or to modify them according to his/her own QoS needs thereby allowing for heterogeneous receivers in contrast to rsvp-vic. The processing of the *Resv* message that is initiated by the VCR is done in the same manner as described in scenario A.

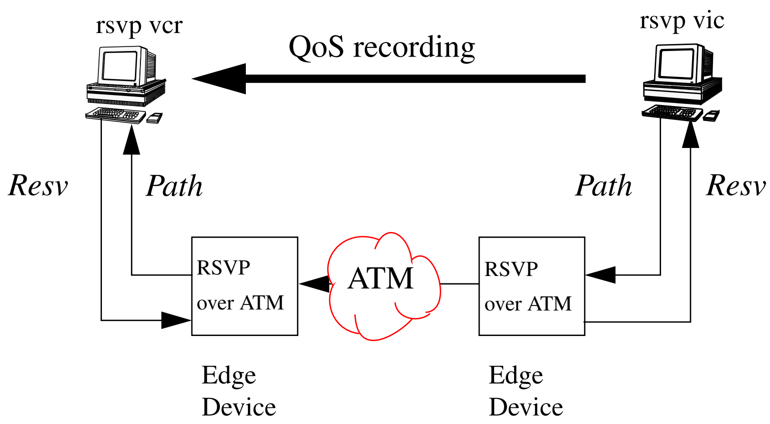


*Figure 8:* Scenario B.

## 6 Conclusion and Future Work

Much work has been done on the development of applications that allow the transmission of audio and video data. Since the introduction of the MBone VCR there is also the opportunity to record and play back such audio and video data transmissions. On the other hand considerable effort has been spent on the support of integrated services in the Internet as well as in ATM. Both worlds (Internet, ATM) do now offer support for integrated services but in very different ways. The increasing deployment of ATM especially in the backbone makes an integration between ATM and IP necessary. To support integrated services this integrating cannot satisfactorily be solved by *Classical IP over ATM*. In this case a more successful approach is the introduction of RSVP over ATM, which allows QoS support in heterogeneous IP/ATM networks.

It was our goal to show in a prototypical system how the support of Internet Integrated Services in heterogeneous IP/ATM networks for video and audio applications in the future may look like. Because there was no RSVP over ATM implementation available for our environment as we started our project, we had to implement our own RSVP over ATM prototype that could be used by the extended VCR and other RSVP capable applications.

In the second step we extended the MBone VCR in a manner that allows to play and record sessions in a guaranteed quality. From our point of view these enhanced applications (e.g. rsvp-vic, enhanced VCR) will lead to an increasing use of the Internet for audio and video applications. This will also be supported by the raising use of ATM as network media in the backbone, since ATM promises to support QoS on the link layer even in a WAN environment.

We are currently testing the system in relation to performance and reliability to find out in which elements further optimization has to be done.

Future work could be to improve the RSVP over ATM implementation in the areas of QoS mapping, multicast support for heterogeneous receivers and clever schemes to solve the problem of supporting RSVP's dynamic QoS. A further area of future work in the application domain could be to develop a better QoS user interface for the extended VCR in order to make it more intuitive for a layman user.

## References


[ATMF96a] The ATM Forum: *Traffic Management (TM) Specification 4.0*, April 1996.

[ATMF96b] The ATM Forum: *ATM User-Network Interface Signalling Specification 4.0*, July 1996.

[Berg97] L.Berger: *RSVP over ATM Implementation Guidelines,* Internet Draft, November 1997.

[BFGK95] A.Birman, V.Firoiu, R.Guerin, D.Kandlur: *Provisioning of RSVP-based Services over a Large ATM Network,* IBM Research Report, October 1995.

[BCDB95] M.Borden, E.Crawley, B.Davie, S.Batsell: *Integration of Real-Time Services in an IP-ATM Network Architecture*, Internet RFC 1821, August 1995.

[BrCS94] R.Braden, D.Clark, S.Shenker: *Integrated Services in the Internet Architecture*, Internet Request for Comment 1633, 1994.

[BrGi96] T.Braun, S.Giorcelli: *Quality of Service Support for IP Flows over ATM,* Kommunikation in verteilten Systemen:GI/ITG Fachtagung, February 1997.

[BrHo97] R.Braden, D.Hoffman: *RAPI - An RSVP Application Programming Interface Vers. 5,* Internet Draft, October 1997.

[BZBH+97] R.Braden, L.Zhang, S.Berson, S.Herzog, S.Jamin: *Resource Reservation Protocol (RSVP) - Version 1 Functional Specification*, Internet Request for Comment 2205, September 1997.

[CBBB+97] E.Crawley, S.Berson., L.Berger, F.Baker, M.Borden, J.Krawczyk: *A Framework for Integrated Services and RSVP over ATM,* Internet Draft, November 1997.

[FlJa95] S.Floyd, V.Jacobson: *Link-Sharing and Resource Management Model for Packet Networks*, IEEE/ACM Transactions on Networking, Vol.3 No.4, August 1995.

[FORE95] Fore Systems: *Fore API Documentation*, Online Document: ftp://ftp.fore.com/pub/docs.

[FORE96] Fore Systems: *Fore IP overview, ForeThought 4.1 Release*, Fore Systems, October 1996.

[GaBo97] M.W.Garrett, M.Borden: *Interoperation of Controlled Load and Guaranteed Service with ATM,* Internet Draft, July 1997.

[Holf95] W.Holfelder: *MBone VCR - Video Conference Recording on the Mbone,* Proceedings of ACM Multimedia ‘95, October 1995.

[Laub94] M.Laubach: *Classical IP and ARP over ATM*, RFC 1577, January 1994.

[MaMS97] M.DeMarco, G.Meroni, L.Salgarelli, *Efficient Transport of IP Flows Across ATM Networks,* IEEE ATM ‘97 Workshop, May 1997.

[McJa95] S.McCanne, V.Jacobson: *vic: a flexible framework for packet video,* Proceedings of ACM Multimedia ‘95, October 1995.

[MVCR97] W.Holfelder, *The MBone VCR,* Software online: ftp://pi4.informatik.uni-mannheim.de/pub/mbone/vcr.

[RSVP97] SUN: *RSVP Software Release for Solaris,* Software Online:http://playground.Sun.COM/pub/rsvp/.

[RVIC97] ISI - Information Science Institute: *RSVP VIC,* Software Online: ftp://ftp.isi.edu/rsvp/release.

[ShPG97] S.Shenker, C.Partridge, R.Guerin: *Specification of Guaranteed Quality of Service*, Internet Request for Comment 2212, September 1997.

[SKB96] A.Schill, S.Kühn, F.Breiter: *Internetworking over ATM: Experiences with IP/IPng and RSVP*, Computer Networks and ISDN Systems, Vol. 28, 1996.

[Wroc97] J.Wroczlawski: *Specification of the Controlled-Load Network Element Service*, Internet Request for Comment 2211, September 1997.