

Design and evaluation of an advance reservation protocol on top of RSVP

A. Schill, S. Kühn , F. Breiter

*Dresden University of Technology; Chair for Computernetworks
01062 Dresden, Germany, call: 049351/4638261
schill/kuehn/breiter@ibdr.inf.tu-dresden.de*

Abstract

Existing reservation protocols such as RSVP implement so-called immediate reservations (ImRe) that are requested and granted just when the resources are actually needed. This paper describes the design, implementation and evaluation of a novel advance reservation protocol. Its major property is that resources are reserved well in advance to the actual usage phase. This way, the probability of resource availability can be increased, and network utilization can be improved. Specific aspects covered in this paper are the mapping of ReRA (resource reservation in advance) onto RSVP, the evaluation and selection of an appropriate call admission control strategy, the internal management of advance reservations, the mapping of the approach onto ATM, as well as its implementation and evaluation.

Keywords:

ATM, RSVP, resource reservation in advance, ReRA, quality of service

1 INTRODUCTION

Considering, for instance, the „real world“, activities such as a meeting in a conferencing room are scheduled for a specific time. With this example, reservation means that limited resources are reserved a certain time in advance to get an assurance that the resources are available at the requested time. The definition of reservation in current QoS based networks like ATM covers another kind of semantics: reservations are being performed in conjunction with a connection establishment; this means that reservations are done at the time when the network resources are actually needed. This is an element of uncertainty, because in the worst case the required resources are used by other applications and therefore the application request must be rejected. However, using ReRA, a future lack of resources during actual reservations can be avoided.

From our point of view, it shall be possible to schedule applications, e.g. an important videoconference with several partners, for a given time in the future. The system shall then calculate and virtually reserve the required resources for the

specific time, however without immediately blocking them. In order to grant the required QoS, it is necessary to provide mechanisms of resource reservation in advance as part of reservation protocols.

Resource reservation in advance allows a more flexible handling of applications by decoupling the starting time of the service from the time the service request is made. This is implemented by our ReRA mechanisms as an extension of existing reservation protocols (RSVP on top of ATM). As an application example, these mechanisms can support the computation of the earliest time for Scalable Video on Demand (SVoD); so the user can play the movie with the desired QoS at the earliest possible time when the required resources are available. (Hafid, 1997) shows that SVoD achieves high resource utilization and notably decreases the blocking probability of user requests in comparison with typical VoD.

As most current and future networks are not purely based on ATM but consist of a mix of various technologies (ATM, FDDI, Ethernet etc.), higher-level protocols on top of ATM such as IP are necessary to enable heterogeneous interoperability. To achieve QoS guarantees, reservation protocols such as RSVP (Resource ReReservation Protocol) are necessary at this level. In order to address advance reservations, we present an implementation of a ReRA signaling mechanism in RSVP. Moreover, an admission control strategy combining ATM and ReRA algorithms are discussed and simulation results are presented.

The paper is organized as follows: In section 2 we shortly discuss differences between typical immediate and advance reservations and the suitability of extending RSVP with ReRA mechanism. Moreover, related work concerning resource reservation in advance is outlined. Section 3 describes the general model and its mapping onto the communication mechanisms in RSVP in detail with references to specific problems and associated solutions. In view of using ReRA on top of a RSVP/ATM protocol stack, an admission control algorithm will be discussed in section 4. In section 5, implementation aspects of ReRA as an RSVP extension are presented as a synergy of the former considerations.

2 FOUNDATIONS

ATM networks (Alles, 1995; ATM1, 1995; ATM2, 1995; Cidon, 1995; Partridge, 1994) with traffic classes such as constant bit rate (CBR), variable bit rate (VBR) and available bit rate (ABR) (Konst, 1995) offer quality of service and traffic parameters in order to specify and guarantee QoS. Resource reservations in terms of bandwidth, buffers and other resources performed accordingly. QoS requirements, however, can not be specified explicitly with conventional IP over ATM, multiprotocol over ATM (MPOA) or LAN emulation services (Jeffries, 1994; Laubach, 1994).

However, emerging protocols such as RSVP (Resource Reservation Protocol) (Braden, 1995; Bordon, 1995) or the older ST-II approach (Delgrossi, 1994) are able to map transport-level QoS requirements onto ATM (Crowcroft, 1995). As opposed to the use of native ATM, this facilitates the integration of heterogeneous networks (Malamud, 1992), for example including Fast Ethernet, Gigabit Ethernet, FDDI and other technologies. Together with the emerging IPv6 protocol (Gilligan,

1995), RSVP will form the basis for the Integrated Services Internet and for Internet-II with its gigabit links. Higher-level quality of service architectures enable a further abstraction of QoS specifications, facilitating the implementation of QoS-aware applications (Campbell, 1994; Vogel, 1995).

However, the reservation approach of RSVP and related protocols attempts to reserve resources just at the point in time when they are actually required. No advance planning or resource scheduling is possible this way. Therefore, the likelihood of receiving a rejection of required reservations by the network increases. As an interesting solution to this problem, several authors have suggested advance reservation techniques. Basic concepts of advance reservation are discussed in (Reinhard, 1994), with a consideration of existing reservation protocols presented in (Reinhard, 1995). A detailed discussion of major issues and trade-offs of advance reservation techniques is also given in (Wolf, 1995). More recently, simulation studies (Degermark, 1997; Ferrari, 1997) have been conducted in order to investigate performance effects concerning the duration of reservations and the advance booking time. (Ferrari, 1997) also emphasizes the coexistence of advance reservations and immediate reservations within the same environment. Moreover, (Degermark, 1997) presents initial considerations concerning the mapping of advance reservation protocols onto RSVP.

Our work is based on the conceptual considerations of these related approaches. However, we particularly emphasize the following specific issues:

Admission control: We investigate several alternative admission control strategies and adapt the selected approach to the requirements of ReRA.

Management of reservations: We analyze different mechanisms for supporting coexistence of advance and immediate reservations. We also propose mechanisms for optimizing resource utilization based on flexible periods of validity of reservation requests. Moreover, in case of rejection of advance reservations, we propose appropriate alternatives to the applications.

Validation by simulation and implementation: In addition to further simulation studies concerning admission control and resource utilization, we are also able to present first results of a running implementation of ReRA in order to validate our concepts. Finally, we attempt to integrate our model with the existing policy control of RSVP (Herzog, 1996; Herzog, 1997).

In the following section, our reservation model and protocol is being presented, followed by details of the call admission control and the implementation.

3 RESERVATION MODEL AND PROTOCOL

3.1 General Model

Advance reservations are characterized by the announcement of events for a given time in the future. So one of the necessary parameters is the starting time, that means the time the resources will be needed. Furthermore, it seems to be meaningful to specify a duration which describes how long the reservation will be alive. Using this information the admission control is able to recognize whether the reservation is overlapping with others or not.

Based on these general ReRA characteristics our model is shown in Figure 1 describing the extended communication to set up a resource reservation in advance. Three different phases can be distinguished which are closely connected with our general model. We use these phases in analogy to (Reinhard, 1995). Moreover, in our detailed view we have introduced some time marks at which messages are sent or actions are performed within the phases.

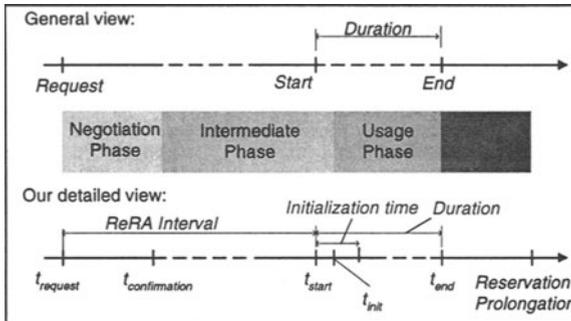


Figure 1: General Model

To be more precise, at the point t_{request} the client issues a request to specify how much of the resource capacities have to be reserved. It also specifies the points in time that define beginning (t_{start}) and duration (d_{res}) of the reservation. After completed negotiation and admission control, the negotiation phase is finished

by getting a confirmation message (t_{confirm}).

After a successful negotiation, the users might want to re-negotiate the communication resources. This covers the cancellation or an increase or decrease of the requested resources. If the changes are acceptable or the reservation was successfully released, the service provider acknowledges it. Re-negotiation concerns not only the traffic and QoS parameters of the reserved connection but also the starting point and duration. A re-negotiation can be performed until t_{end} is reached.

When the start time is reached, the activation of the requested resources has to be performed within the initialization time. If the client is not sending a demand message by then, the reservation will be aged and all state information corresponding to it will be removed. We prefer this additional communication at t_{start} because last changes should be possible and resources are only hard and physically reserved if they are really needed. In contrast to (Wolf, 1995) we don't need a special preparation mechanism for the resource usage phase, for example for preempting running applications with immediate reservations. We just have to convert a logical reservation into a physical one as the resources are already available.

3.2 RSVP protocol extensions

In the following essential changes and extensions of RSVP to support ReRA will be described. In particular, this includes:

- the negotiation of ReRA information,
- the modification of messages to carry the additional information (ReRA object),
- new RSVP-API functions and
- a special handling of RSVP in case of short and longer termed router failures before the starting time of the advance reservation is reached.

Moreover, to realize a complete extension of the RSVP specification to support

ReRA it also includes an extension of: the state control blocks (Path-, Reservation- and Traffic Control blocks) to support the pseudo hard state concept (see below) and to hold the new reservation parameters, the message processing rules and the specific RSVP interfaces (e.g. traffic control API).

Negotiation of ReRA information

To describe the negotiation of ReRA information, we have introduced service primitives which are derived from the general model:

ResvReq/ ResvInd,
 ResvRsp/ ResvConf,
 ResvModReq/ ResvModInd,
 ResvModRsp/ ResvModConf,
 ActivateReq/ ActivateInd.

These service primitives were completed to announce a future data flow:

AnnounceReq/ AnnounceInd,
 AnnModReq/ AnnModInd.

Looking at the communication (see Figure 2) we can recognize that it is possible to map the service primitives of the advance reservation to appropriate RSVP messages. So we use PATH messages to announce the future data flow and RESV messages to setup an advance reservation for that flow. In this way we don't have to introduce new RSVP messages. Merely an extension of the currently defined messages is needed to carry the additional parameters.

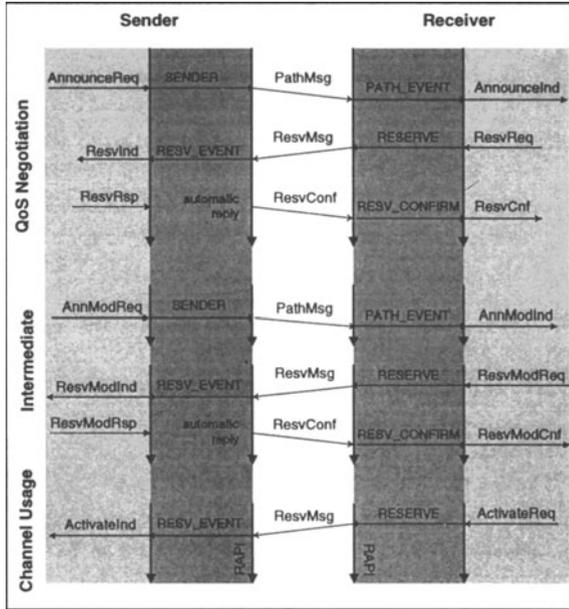


Figure 2: Mapping of ReRA primitives to RSVP messages

ReRA - Object

The negotiation of ReRA information is done by sending PATH and RESV messages carrying an additional instance of a new RSVP (Figure 3). As the ReRA object is an additional and optional object in an RSVP message, a result of this solution is that the mechanisms for sending and receiving RSVP messages need not

Object Length	Class-Num (16)	C-Type (1)
Flags	Option Mask	ID equal/part of PRID (sid) (rid)
	Option 1 (rera_start)	
	Option 2 (rera_end)	
	Option 3 (initialization_time)	

Figure 3: ReRA Object

to be changed and the semantics of the messages will be preserved for ImRe.

The definition of the class value (Class-Num 16; C-Type 1) has two reasons: first, to be compatible with other RSVP platforms and secondly to get error information from non ReRA nodes instead of dropping ReRA messages.

A ReRA object consists of a common header, followed by a object specific body. The first byte of the body includes the following *Flags*:

- **WILDCARD**: If this flag is set and the start and/or end time are not specified then the missing time parameters of the reservation will be filled automatically by the system based on the time parameters of the PATH message. This operation will also be enforced after a modification of the data flow start and end time.
- **ACTIVATE** (for sender and receiver): This flag is used, for example in case of a receiver, to distinguish between a normal RESV (refresh) message and a RESV message, which is send at the starting point to indicate that the user wishes to use the reserved resources (the latter complies with ActivateReq). If the receiver is not sending a RESV message (with this flag) within the initialization time, a RESV-TEAR message will be sent automatically and all state information corresponding to it will be removed. Automatically means, that we send a RESV-TEAR message instead of a RESV refresh message if the current time is greater than the starting time of the reservation.

The next byte within the ReRA object includes an *Option Mask* which is used to initiate a special handling of the information (option) specified in that class. Currently three options are defined (*re_ra_start*, *re_ra_end*, *initialization_time*)¹. Each option has a corresponding bit within the option mask. For example, in the RESV message, the *Option_1* and *Option_2* bit of the option mask is set to indicate that the object includes the points in time when the reservation starts and finishes. Furthermore, to change the end of an already requested reservation only the *Option_2* bit is set and the options field carries the new end time.

Moreover, an *ID* (identifier) is part of the object to realize a unique identification of a data flow and a reservation as part of a session. As it is possible that a sender can announce more than one data flow and a receiver can request more than one reservation per data flow², the sender and receiver have to number it consecutively in *Sid* (sender id) and *Rid* (receiver id).

Hence, a data flow is identified by: *<Session, Sender_Template, Sid>* and

a reservation is identified by: *<Session, Filter_Spec, Sid, Recv_Address, Rid>*

Another possible way is, that the ReRA object consists of two options including *re_ra_start* and *re_ra_end* (for identification) and two additional options including a new *re_ra_start* and *re_ra_end* for a modification. Using an ID to identify data flow and reservation, respectively, we don't have to include the *re_ra_start* and *re_ra_end* time in each message and so we reduce the RSVP overhead.

¹ Currently we use a fixed option length. In the future the option field will have a variable length.

² In a video conference for example.

Extending RSVP with such a mechanism is supporting and ideally complementing the newly introduced policy control of the IETF (Herzog, 1996; Herzog, 1996; Herzog, 1997). These policies follow the bilateral agreements model. The model assumes that network clouds (providers) contract with their closest point of contract to establish ground rules and arrangements for access control and accounting. The advanced reservation scenario is almost identical to the simple access control. Moreover, it is assumed that each verbal bilateral pre-registration is identified by a PRID (pre-registration ID). The policy data object defined by the IETF has the type of AR (Advanced Reservation) and the form PD (AR, PRID, UID).

Using our mechanism an automatic bilateral pre-registration is realized. Sending the ReRA request the object includes one additional option: the UID. Moreover, the PRID is defined by the identifier in the ReRA object, the session and addresses of sender and receiver. As result of these combined mechanisms the activation and an additional access control for authorized users of an advanced reservation will be performed by sending only the policy object in a RESV message.

New RSVP-API functions

The following functions are introduced to realize a more flexible handling of advanced reservation within the RSVP-API:

- **LEAVE:** This function allows the user to leave a session without releasing the soft states. It is used to inform the RSVP daemon that the application is not actually available for the next time until the application will make a modification or the starting point is reached. The application should save session information to be able to re-register by calling the session function.
- **TEARDOWN:** As we allow more than one reservation in one session (which can be distinguished by the start and end time of the reservation) this function can be called to remove a specific reservation.

Short and longer termed router failures

As the advance notice of an advance reservation request can take a long time, the occurrence of special failures and its removal have to be considered. This concerns among other things the actual availability of endsystems and the reaction to short and longer termed router failures (Figure

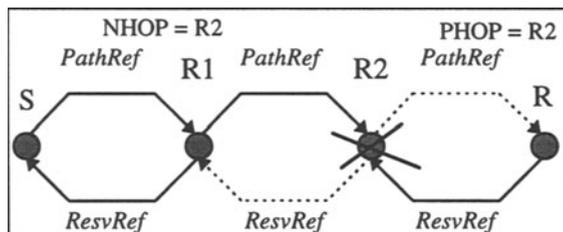


Figure 4: Router failure of router R2

4) which implies a changed topology of the net. In order to recognize and to handle such kind of failures we have developed a special algorithm. Based on the expired cleanup timer for each reservation, which is the result of missed RSVP refresh messages, longer termed failures are recognized and temporary router failure are ignored by increasing the lifetime of the soft states (see below).

Concerning the expiration of the lifetime there are two reasons. Either there is a router failure (crash, topology changes) or an endsystem is unplugged from the net. In the case of longer termed router failures a new route will be determined using the routing protocol. In this context it should be emphasized that current routing protocols are concerned only with topology information and not network resources. This can result in alternate paths and hence in rejection of advance reservation within the intermediate phase. In order to avoid it, route pinning has been suggested

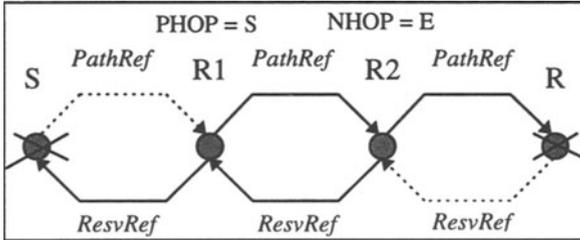


Figure 5: Unplugged endsystems

in (Crowley, 1996): „Route pinning means that an existing route with a reservation will not be replaced by a better one unless the existing one is no longer usable because of a topology change directly related to the existing route“.

If an endsystem is unplugged from the network (for instance a PC which was turned off; Figure 5), this results in missed PATH and RESV refresh messages and in a loss of the corresponding soft states of the neighbored router. In order to avoid this, the routers which are situated immediately behind the sender and the receiver, respectively, should react if no new refresh message arrives after the expired lifetime. Because these routers take over the function of the endsystems in sending PATH and RESV refresh messages, they will be called proxy routers. Moreover, each router has to determine if the next host is the receiver downstream or the sender upstream, itself:

- NHOP = receiver address³ (if RESV refresh message is expected) or
- PHOP = sender address (if PATH refresh message is expected).

If one of these cases is fulfilled, this results in holding the soft states in the router. More precisely, these soft states are kept and only removed if the user sends a teardown message or if the starting time of the reservation is already exceeded. Therefore, the states are called pseudo hard states and so we are able to deal with unplugged endsystems.

To avoid that soft states which hold the ReRA information will be deleted caused by temporary router failure we use a modified refreshing mechanism. Considering the soft state approach of RSVP, the periodic sending of PATH and RESV messages (default: every 30 seconds) produces a considerable overhead. The refreshing of the states is controlled by the refresh time R and the lifetime L . The relationship between R and L is expressed by the following equation: $L \geq (K + 0.5) * 1.5 * R$ ($K-1$ successive messages may be lost without state information being deleted) and is shown in Figure 6.

³ To provide the receiver address in unicast/multicast sessions the RESV message has to include always a RESV_CONFIRM object.

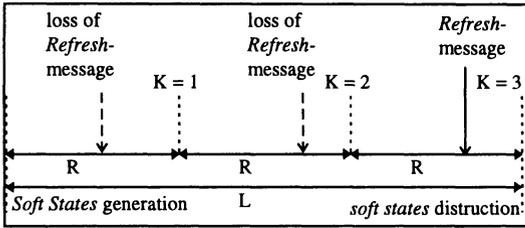


Figure 6: Refresh-time and Lifetime

From our point of view, L should increased in such a way that reservations are insensitive to temporary router failures and that the overhead of RSVP messages is minimized. This can be achieved by an adequate increase of R and K . K is a constant value, which should be

configured manually (e.g. as a parameter for the RSVP daemon) or via management functions. R is calculated when the ReRA connection is requested and is reduced step by step until the starting time is reached. At this time, R is set to the default value. Assuming that R is calculated by $R = \text{standard_refresh_time}(30s) + (\text{starting_time} - \text{current_time}) / D$, R is dependent on the current time and a granularity D . It should be emphasized that concrete values of R and K have to be determined based on future measurements in our experimental environment.

On the other hand concerning receivers which have joined a multicast group during the intermediate phase the increasing of R and K results in delayed receiving of PATH messages. In order to avoid this, RSVP calls the mrouterd daemon periodically. Based on the information about new receiver addresses joined to a multicast session, RSVP sends PATH messages to the multicast group immediately. Moreover, it is possible that a receiver can leave a multicast group during the intermediate phase (e.g. in case of unplugged endsystems). This results in loss of refresh messages following the path to the receiver. In our implementation all multicast information of advanced reservation are held within the soft states as an additional list of outgoing interfaces.

4 CALL ADMISSION CONTROL

4.1 Sharing versus Partitioning

The design of a suitable ATM traffic control is considered as a fundamental challenge for the success of the ReRA mechanism. The primary role of traffic control is to protect the network and the user in order to achieve predefined network performance objectives. Dealing with advance reservation, an additional role of traffic control is to optimize the use of network resources for the purpose of achieving: (1) sufficient network efficiency, that means the effectiveness in utilizing bandwidth, (2) a minimal blocking probability and (3) a fair handling of the advance and the immediate reservations. Obviously, it seems to be difficult to combine these features in such a way that the Call Admission Control (CAC) will achieve all these goals in an optimal way. So in (Schill, 1997) we have considered influence parameters and their effects on the characteristics of the CAC as a first step.

Basically, there are the following policies of resource awarding or administration to handle the two different kinds of reservations: sharing and partitioning. Sharing the resources allows the awarding of the whole bandwidth for example in the case of an aggressive access to ReRA connections. A partitioning of the bandwidth for

instance for ImRe and ReRA is heading towards an equitable handling of the single reservations as the bandwidth is dedicated to each reservation type. But, assuming an adverse allocation of resources between ImRe and ReRA, it can result in an inefficient utilization of bandwidth caused by the two boundaries.

A dynamical partitioning (Ferrari1, 1997) seems to be a reasonable trade-off. A partitioning with a moveable boundary means, that each reservation has its own partition, but with the option of using available resources of the other partition if the own partition is completely allocated. But considering this in more detail, it resembles sharing across the whole bandwidth or at least sharing of bandwidth in between exclusive partitions (based on watermarks to protect e.g. the advance partition; see Figure 7), respectively. As an appropriate value of the watermarks for delineating exclusive partitions is not easy to determine, sharing the whole bandwidth seems to be the most common way.

Based on sharing we have developed a basic admission control for an ATM network.

Dealing with the RSVP/ ATM protocol stack the CAC of the QoS based ATM and its specific service classes have to be investigated for a ReRA extension. In view of the growing importance of VBR and ABR services, we consider a suitable ATM VBR algorithm in the next section which takes into account the statistical multiplexing of variable flows.

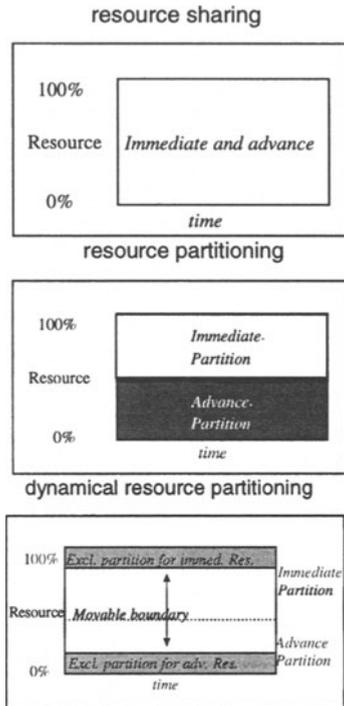


Figure 7: Policies of resource awarding

4.2 Administration of advance reservations

As a basis for introducing our ReRA ATM admission control as part of the next section, we briefly discuss the management or administration of advance reservations.

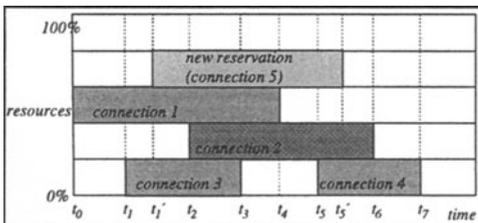


Figure 8: Administration of ReRA information

predefined and variable time intervals. We prefer the latter because no adaptations of start and end points as a multiple of granularity intervals is necessary. Moreover, we assume, that the number of variable intervals will be less than the account of

If a new reservation request arrives, the check for available resources has to be performed over the whole duration of the new connection. Resource changes are only caused by other reservation start or endpoints. So we have considered two methods of administration in interval tables:

predefined intervals in general. The reservation data base consists of a list of all intervals. Each interval is characterized by a begin, end and a list of overlapping reservations (Figure 8).

4.3 Equivalent Capacity

In general, the congestion control as an important area in ATM networks can be classified into preventive and reactive control. A combination is currently used in ATM: CBR and VBR use preventive schemes and ABR is based on a reactive scheme. Below, we examine the CAC algorithms of the preventive control which can be grouped in non-statical allocation (or peak rate allocation used for CBR) and a statical allocation (which makes economic sense dealing with bursty sources - VBR).

In view of extending one algorithm for advance reservation we have compared the following CAC schemes:

- Equivalent Capacity
- Heavy Traffic Approximation and
- Non-parametric Approach

The simulation comparisons of the performance of the above algorithms has shown, that the Equivalent Capacity achieves the highest statistical gains

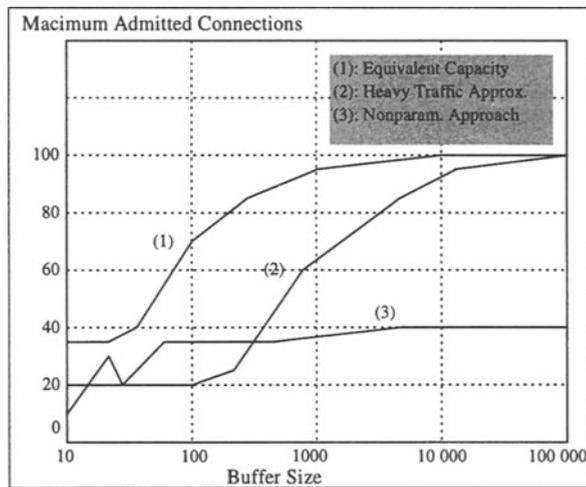


Figure 9: Comparison of admission control algorithms

(defined as the maximum number of admitted connections divided by the number of connections that can be accepted using peak rate allocation) depending on buffer size (Figure 9) and the required cell loss probability.

Table 1: Comparison of admission control algorithms

	<i>Equivalent Capacity</i>	<i>Heavy Traffic Approximation</i>	<i>Non-parametric Approach</i>
source description	interrupted fluid process	interrupted bernoulli process	no traffic model
traffic parameters	peak rate, average cell rate, average burst length, standard deviation	peak rate, average cell rate, average burst length,	peak rate, average cell rate
effectivity	high	high (large buffer)	low
computation overhead	low	low	higher

In Table 1 the three algorithms are compared relevant to some characteristics like their effectivity or traffic parameter description. As a major result of these considerations, we select the Equivalent Capacity (EC) as the basis of a ReRA extended algorithm for bursty (VBR) sources. In case of a new connection the EC of this single source can be obtained by solving the simplified equation (1) for c. The result c has to be less or equal the maximum link cell rate. In the case of N sources, and given that the buffer has the capacity K, the equivalent capacity is again the service rate C (2) which ensures that the cell loss is ε. As a simplification again, C (2) is the combination (minimum) of the Fluid Flow Approximation (which is including the new one) and the Stationary

Equivalent Capacity:
 statistical multiplexing of variable flows
 with $c = R \frac{a - K + \sqrt{(a - K)^2 + 4a\rho K}}{2a}$ (1)
 and
 $C = \text{MIN} (C_{(F)}, \sum_{i=1}^N C_{(S)} = m + \alpha\sigma)$; (2)
 $a = \ln(1/\epsilon)b(1-\rho)R$
 $\alpha \approx \sqrt{2\ln(1/\epsilon) - \ln 2\pi}$
 m: average cell rate K: buffer
 b: average burst length R: peak rate
 σ: standard deviation ε: loss probability
 $\rho = \frac{m}{R}$: source utilization

Figure 10: Equivalent Capacity

the EC-sum of all running connections Approximation which has been observed to follow approximately the Normal distribution supposing a high number of connections. The new connection will be accepted if C (2) is less or equal the maximum link capacity (Figure 10). Based on own simulation results (Figure 11), the following Table 2 summarizes why it is useful to determine the minimum of the Fluid-Flow approximation and the Stationary approximation.

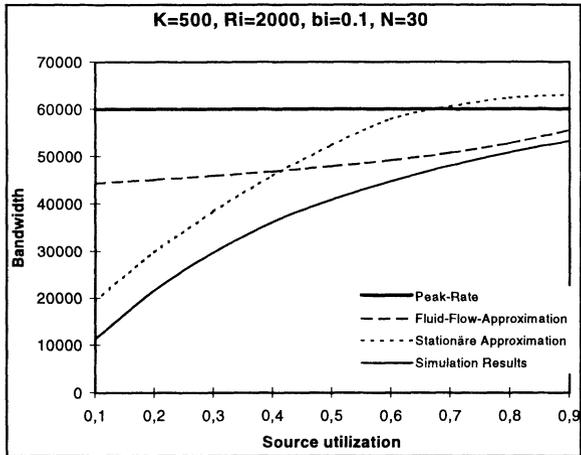


Figure 11: Simulation results

Table 2: Comparison of fluid flow and stationary approximation

Fluid-Flow Approximation is more suitable for:	Stationary Approximation is more suitable for:
small number of sources	higher number of sources
higher utilization of the source	smaller utilization of the source
shorter burst length	longer burst length
smaller peak rates	higher peak rates

As an example, Table 2 presents the suitability of both approximations and the peak rate allocation to draw a parallel with the actual simulation results. While varying the utilization of the 30 sources (namely the average rate) and keeping the buffer size b , the peak rate and burst length constant, the Stationary Approximation is a better approach to the simulation results (basing on a IFP source model) until reaching a source utilization of 0.4. Coming closer to a 100 % utilization (or to R) the Fluid-Flow approximation is the better approach. In principle, it can be said that both reservations are complementary to each other. So the calculated minimum of both registers the highest statistical gain.

4.4 An ATM ReRA-CAC Algorithm based on Sharing

Sharing the resources between ImRe and ReRA, there is a problem with the currently undefined duration of immediate reservation. Retaining these characteristics of ImRe, pre-emption has to be performed to realize that enough resources will be available looking ahead for the connection reserved in advance. As pre-emption is not desirable in most cases, we prefer ImRe with (whenever possible) a given duration. Otherwise ImRe with a wildcard duration is limited by the minimum advance notice. From this, it follows that an exact separation between ImRe and ReRA is possible.

The following admission control is a combination of the above considerations. That means we use sharing as the resource awarding strategy in combination with the equivalent capacity for variable flows. In general, each new reservation request in ATM is characterized by:

reservation_request := (reservation_time, source_traffic_descriptor)
 source_traffic_descriptor := (PCR, SCR, MBS, CLR, MaxCTD)
 reservation_time := (start, duration)

Using the equivalent capacity for VBR streams in ATM, an appropriate mapping of the ATM specific parameters onto the algorithm parameters have to be performed (with $D^2 = m(R-m)$):

ATM-Parameter	PCR	SCR	MBS	MaxCTD	CLR
CAC-Parameter	R	m, D^2	b	K	CLP

The average burst size b is calculated as follows: $b[s] = \frac{MBS[Cell]}{PCR[Cell/s]}$. Moreover,

the dimension of the buffer capacity depends on the maximum cell delay variation (Saito, 1994) supposing a FIFO buffer:

$$MaxCTD[s] \geq \frac{K[Cell] * Zellenlänge[bit / Cell]}{Linkkapazität[bit / s]} \quad \text{and: } CLP = CLR = \epsilon.$$

The following CAC steps are illustrated in Figure 12:

(A) Assuming, there is a data base with M intervals I_0, I_1, \dots, I_M . The first one begins with time 0 (current time), the last one is limited with the maximum advance notice.

(B) Moreover, a new reservation request is given, whereby ImRe is handled as a special case of ReRA with starting time 0 and a duration less or equal the minimum advance notice.

(C) The CAC determines all intervals $I_p \dots I_Q$, which are overlapped by the duration of the new request. Moreover, for all intervals $I_p \dots I_Q$ the CAC is performed.

Computation of the equivalent capacity in case of variable bit rates:

```
if res_request == VBR
```

- Assume N is the number of connections in the current interval. The fluid flow approximation $C_F(N+1)$ is calculated for N+1 connections including the new one:

$$C_F(N+1) = C_F(N) + C_{F,N+1}$$

- Moreover, the stationary approximation for N+1 connections is computed:

$$C_S(N+1) = m(N+1) + \alpha \cdot \delta(N+1)$$

- Finally, the equivalent capacity is the result of the combination of both approximations:

$$C_E(N+1) = \min\{C_F(N+1), C_S(N+1)\};$$

```
else //res_request is CBR
```

$$C_S(N+1) = C_F(N) + res_request.PCR;$$

(D) In case of a resource shortage the appropriate interval is tagged. If one of the intervals is tagged, the new request is rejected, otherwise accepted.

Accepting the new request causes an updating of the interval table on the appropriate host or intermediate node as described in section 4.2. If a rejection on a subsequent node occurs (Release) all interval tables on the way to the sender have to be changed to the state as it was, before the new request has arrived.

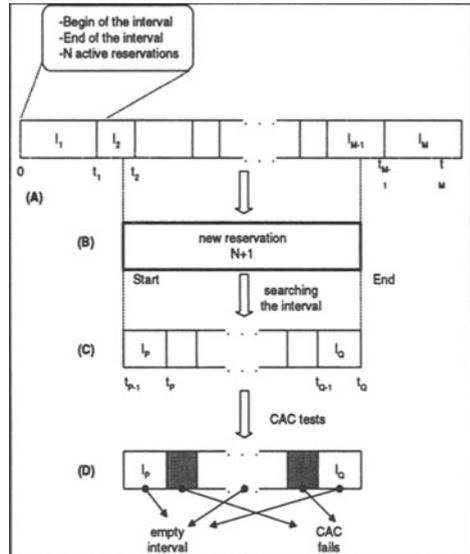


Figure 12: CAC algorithm

Problems of Resource Fragmentation

The introduction of advance reservation is closely connected with the fragmentation of resources. In order to optimize the utilization of bandwidth for reservation request, two ways are discussed in the following. To cope with the above aim we have introduced an extended time model for ReRA (Figure 13). So each reservation is not only characterized by a starting point and a duration, but a period of validity. This is useful dealing with Scaleable Video on Demand Systems. The user specifies the time of validity in addition to the duration and the desired beginning of the requested video, for example 5 till 10 p.m. The advantages of this modified ReRA model are an optimized utilization of bandwidth and a lower blocking probability.

To be able to inform the user about alternative reservations (accurate beginnings and ends), an overview about the current reservation situation of all nodes along the route must be gathered. This is done by overlapping the reservation situations for the period of validity of all nodes. Alternatives are determined as follows:

- a list of best matches with the desired bandwidth and duration,
- a bandwidth alternative if the requested resources are not available, and
- a time alternative, if the desired duration cannot be satisfied.

As this model is not part of the current signalling model in ATM, we expect to extend the signaling for sending an ADSPEC like in RSVP. This way, the pessimistic user can gather the desired reservation information before the actual request (SETUP) is sent.

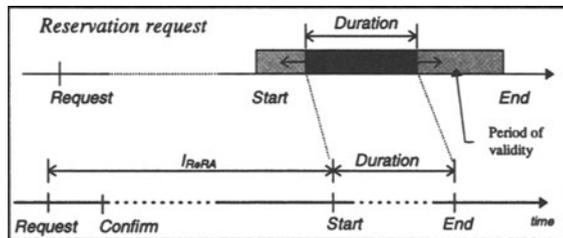


Figure 13: Extended time model for ReRA

In addition to the „ADSPEC“ suggestions, a bandwidth and duration alternative is given to the user even after a rejection on a single node. This can be used for calculating and scheduling alternative reservations.

4.5 Deferred Acknowledgements

To optimize the bandwidth utilization and to reduce blocking probability, an investigation of immediate versus deferred acknowledgement of reservations is interesting. Generally considered, there are two models based on the time when scheduling decisions are made. In the case of immediate acknowledgement, scheduling decisions are made as soon as a request arrives. The other model is based on delayed acknowledgements, only applicable to reservations in advance. Every request has an associated decision point which is given by the system. Focusing on a reduced blocking probability, for example, the following simple heuristics can be used. We assume, that the bandwidth requests of all reservations with a synchronous starting point are sorted in an ascending way. Accepting all reservations beginning with the smallest resource request until the whole bandwidth is awarded, results in a reduced blocking probability. Obviously, assuming synchronous starting points of an amount of reservations is an abstract model and only conceivable with relatively coarse-grained time intervals. In the future, we also have to investigate and develop reasonable heuristics to achieve a better utilization of resources by a special handling of requests according to their attributes (e.g. bandwidth requirements).

5 IMPLEMENTATION ARCHITECTURE AND DETAILS

Our implementation environment consists of several workstations of type DEC 3000/300 and DEC 3000/700 which are connected with our DEC GigaSwitch/ATM via fibre optic links.

Within our development work, we use a RSVP (version 1 Release 4.0a7 of the Information Science Institute (ISI)) implementation on top of ATM on the Digital

Unix operating system. As part of the IP/IPng convergence modules (CM), and hence part of the ATM subsystem, the RSVP specific API supports real time handling and transport of IP/IPng flows. The API is able to receive reservation information from a local RSVP daemon. Based on information contained in a given flow specification, a new ATM VC reservation is performed after completing the mapping of service classes and parameters. The classification of the IP/IPng flows belonging to a dedicated reserved reservation (virtual channel) is done in the IP/IPng CM, too. The mapping of RSVP onto ATM specific connection parameters has been checked by using the multi-generator (MGEN) utility. After specifying the RSVP parameters the toolset generates real-time traffic patterns so that the network can be loaded in a variety of ways with constant and poisson distributed streams. The following figures 14 and 15 show one flow per vc measured on the sending and receiving side (part I), and the second stream mapped into another vc (part II).

A comparison of the reserved rate and measured rate on the sending node (always including UDP-IP and ATM overhead) shows that the reservation overhead (inaccurate mapping) is about 389 Bit/s, that means the actual throughput is 99.996% of the reservation. But this is a result of the packet generation, too, which is not actually conforming to the reserved cell stream per second. As expected, considering both reservations at the same time, no influences on the required QoS of each flow have been considered.

At present, our ReRA-RSVP version supports an extended API between the RSVP daemon and the

application. In view of the ReRA object, the wildcard and activation flags are supported. The performed tests have shown that it is possible to extend RSVP in such a way described in section 3. With exception of the proxy router model, all extensions have been tested. Solely, the traffic control has not been extended as the implementation of ReRA in ATM is work in progress.

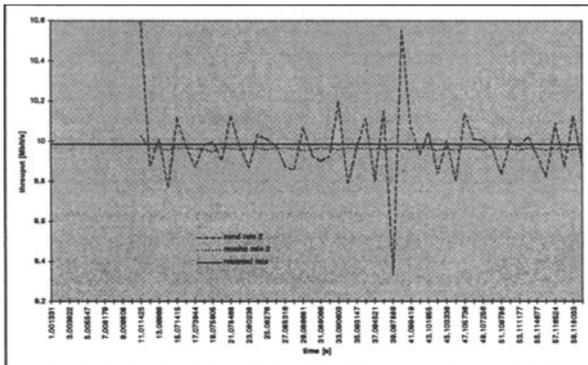


Figure 15: RSVP over ATM measurement - part II

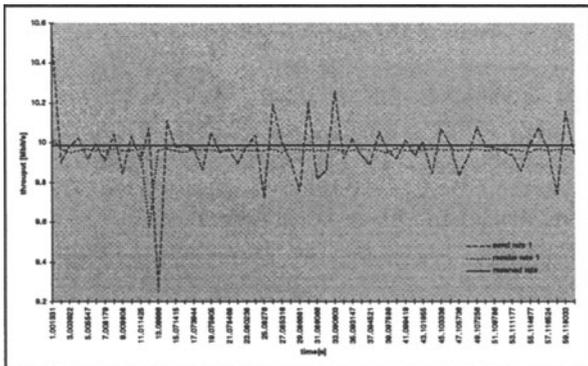


Figure 14: RSVP over ATM measurement - part I

6 CONCLUSIONS

As a main contribution of this paper, a general model of advance reservation was described and an enhancement of RSVP and ATM was presented to support reservation in advance.

Special emphasis was put on the problem of signalling the appropriate ReRA information within the integrated services Internet. Most importantly, the basic structure of RSVP could be maintained by just introducing additional RSVP classes with associated objects. This way, smooth migration of an existing network using RSVP towards ReRA is enabled. Moreover, a simple admission control algorithm for the ReRA algorithm within an ATM environment was designed. Based on the Equivalent Capacity approach and additional internal management mechanisms, a relatively high number of accepted connections and a correspondingly high utilization of resources can be achieved. This was confirmed by initial simulation results. The associated implementation of ReRA via RSVP has also been discussed and has yielded an acceptable performance for data streams using reserved bandwidth.

The presented concepts for resource reservation in advance are also under refinement and we expect to have further implementation experiences soon. To improve the performance of the CAC we are interested in considering neural network structures as connection admission control in ATM environments dealing with ReRA mechanisms. As the implementation of ReRA in RSVP is finished, our ongoing implementation work is aimed at finishing the realization of an extended signaling and admission control in ATM.

7 REFERENCES

- Alles, A., „ATM Internetworking“, Cisco Systems, Inc., 1995
- ATM1, „UNI Signaling 4.0“, ATM Forum 94-1019R6, Work in Progress, 1995
- ATM2, „Traffic Management Specification Version 4.0“, ATM Forum 95-0013R6, Work in Progress, 1995
- Bordon, M., Crawley, E., Davie, B., Batsell, S., „Integration of Real-time Services in an IP-ATM Network Architecture“, RFC 1821, August 1995
- Braden, R., Zhang, L., Berson, S., Herzog, S., Jamin, S., „Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification“, RFC 2205, September 1997
- Campbell, A., Coulson, G., Hutchison, D., „A Quality of Service Architecture“, Computer Communication Review, Vol. 1, No. 2, April 1994, pp. 6-27
- Cidon, L., Guerin, R., Khamisy, A., „An Investigation of Application Level Performance in ATM Networks“, IEEE Infocom, Boston, 1995, pp. 845-852
- Crowcroft, J., Wang, Z., Smith, A., „A Rough Comparison of the IETF and ATM Services Models“, 1995
- Crowley, Zhang, Sanchez, Salkewicz, „Quality of Service Extension to OSPF“, Internet Draft, draft-zhang-qos-qospf-00.txt, 1996

- Delgrossi, L., Herrtwich, R.G., Hoffmann, F., „An Implementation of ST-II for the Heidelberg Transport System“, *Internet-working: Research and Experience*, Vol. 5, 1994, pp. 43-69
- Degermark, M., Köhler, T., Pink, S., Schelén, O., „Advance Reservation for Predictive Service in the Internet“, *ACM Multimedia Systems*, Vol. 5, 1997, pp. 177-186
- Ferrari, D., Gupta A., Ventre, G., „Distributed advance Reservation of real-time connections“, *ACM Multimedia Systems*, Vol. 5, 1997, pp. 187-198
- Gilligan, R., Nordmark, E., „Transition Mechanism for IPv6 Hosts and Routers“, *Work in Progress*, 1995
- Hafid, A., „Providing a Scalable Video-on-Demand System using Future Reservation of Resources and Multicast Communications“, *IWQOS '97*, New York, May 1997
- Herzog, S., „RSVP Extensions for Policy Control“, *Internet Draft*, Apr. 1997, draft-ietf-rsvp-policy-ext-02.ps
- Jeffries, R., „ATM LAN Emulation“, *Data Communications*, No. 9, 1994, pp. 95-100
- Konstantoulakis, G., Stassinopoulos, G., „Transfer of Data over ATM Networks Using Available Bit Rate (ABR)“, *IEEE Symposium on Computers and Communications*, Alexandria, Egypt, 1995, pp. 2-8
- Laubach, M., „Classical IP and ARP over ATM“, *RFC 1577*, January 1994
- Malamud, C., „Interoperability in Today's Computer Networks, Prentice-Hall, Englewood Cliffs, N. J., 1992
- Partridge, C., „Gigabit Networking“, Addison Wesley Publishing Company, 1994
- Reinhardt, W., „Advance Reservation of Network Resources for Multimedia Applications“, *Proceedings of the 2nd IWACA94*, Heidelberg, 1994
- Reinhardt, W., „Advance Resource Reservation and its Impact on Reservation Protocols“, *TR*, 1995
- Saito, H., „Teletraffic Technologies in ATM networks“, Artech House 94
- Schill A., Kuehn S., Breiter F., „Resource Reservation in Advance in heterogeneous networks with partial ATM infrastructures“, *INFOCOM 97*, Japan, 1997
- Vogel, A., Kerherve, B., Bochmann, G., Gecsei, J., „Distributed Multimedia and QoS: A Survey“, *IEEE Multimedia*, Vol. 2, No. 2, 1995, pp. 10-19
- Wolf, L.C., Delgrossi, L., Steinmetz, R., Schaller, S., Wittig, H., „Issues of Reserving Resources in Advance“, *IBM European Network Center Heidelberg, Technical Report 43.9503*, 1995