Internet Engineering Task Force
Internet-Draft
draft-kim-llrmp-00.ps

P. Kim
HP Laboratories Bristol
Expires: 12/96

Link Level Resource Management Protocol (LLRMP)

Protocol Specification - Version 0

June 12, 1996

Status of 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 inappropriate to use Internet-Drafts as reference material or to cite them other than as "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 ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Abstract

This memo describes the LLRMP, a lightweight link level signalling protocol used to setup resources in shared medium and bridged or switched LANs. Network layer resource management protocols like RSVP or STII, or a local network manager, may invoke the LLRMP to request a certain quality of service over a bridged LAN.

The LLRMP protocol supports a distributed admission control over each shared medium segment within the bridged LAN. Each node running the protocol has knowledge about the resources reserved on its local network segment. Resources are reserved independently on each segment, as a LLRMP reservation message travels along the data path from the source to the receiver.

1. Introduction

The Link Level Resource Management Protocol (LLRMP) was designed to setup resources in local, shared medium and bridged/switched networks. This document describes the protocol properties and mechanisms used to achieve this. Even though the protocol is discussed based on examples which assume Demand-Priority networks, it does not take any advantage of technology specific features. The LLRMP can be used to dynamically reserve resources in heterogeneous bridged/switched networks composed of segments of a different technology e.g. Ethernet, Token Ring, FDDI and 100VG-AnyLAN. The reservation setup for those segments will only differ with regard to link specific information to be carried by the LLRMP and the actual admission control algorithm applied to control the access to the link.

Network layer resource management protocols like RSVP or STII may invoke the LLRMP to request a certain quality of service over a particular bridged LAN. After receiving such a request the LLRMP distributes the required per-flow information to all bridges and switches along the data path between the source and all receiver(s), and initiates admission control on each of the intermediate links. Resources are reserved when necessary and available.

The LLRMP is a new link level protocol which uses a new *ethertype* for carrying control information between end-hosts and bridges/switches. The protocol runs on end-hosts and on bridges. However, for example on 100VGAny-LAN networks, only end-hosts which use the high priority service need to be updated. Nodes which only send with normal priority are not affected. The deployment of the LLRMP protocol in bridges can be performed gradually since the protocol can work transparently through bridges and switches which do not support it. This would allow an administrator to initially control resources on bottleneck segments within the bridged LAN.

The LLRMP is designed independent of any network layer resource management. This allows the LLRMP to simultaneously serve requests from different upper layer management entities e.g. RSVP, STII or any local network management system. Reservations are established and torn down dynamically. The protocol also dynamically adapts to topology changes. Protocol state is usually only maintained in bridges along the data path and not in every bridge on the LAN. We assume standard IEEE 802.1 learning bridges [8]. In order to forward control-messages efficiently, the LLRMP entity on a bridge needs access to the local MAC address table. However, the protocol does not perform any routing functionality. This is assumed to be carried out by other link layer mechanisms e.g. the standard learning process.

The LLRMP protocol supports a distributed admission control over each shared medium segment within the bridged LAN. Each node running the protocol has complete knowledge about the reserved resources on its local network segment. This allows the local node to reject reservation requests without transmitting a control message if there are insufficient link resources. Resources are reserved independently on each segment, on a hop-by-hop basis, as a LLRMP reservation message travels along the data path from the source to the receiver. Unicast and multicast flows are supported.

Throughout this draft, we expect the reader to be familar with RSVP [1], STII[2] and the current drafts for advanced services e.g. [4 - 6] proposed in the Integrated Services IETF working group. The remainder of this document is organized as follows. Section 2 provides an overview of the system and lists the key features of the LLRMP. This is followed by a discussion on the LLRMP message types, the receiver- and the reservation model used. We then describe the protocol mechanisms and present the rules for forwarding control packets. Section 3 discusses service mapping issues. The relationship of LLRMP to network layer resource management protocols like RSVP and STII, and the support of RSVP reservation styles is explained in section 4. Section 5 provides the Functional Specification.

2. System Model and LLRMP Features

We assume a LAN architecture which may have a pure shared, pure switched, or shared and switched topology. Heterogeneous technologies are assumed to be used for different segments of the bridged/ switched LAN. An example for a bridged LAN is illustrated in Figure 1. It consists of a shared backbone segment and several multiport bridges connecting workgroup segments to the backbone. A router and a few special hosts e.g. servers are connected directly to the backbone. Since a pure switched network topology is a special case of a shared one, we discuss the operation of the LLRMP protocol in a shared environment as the more general case.

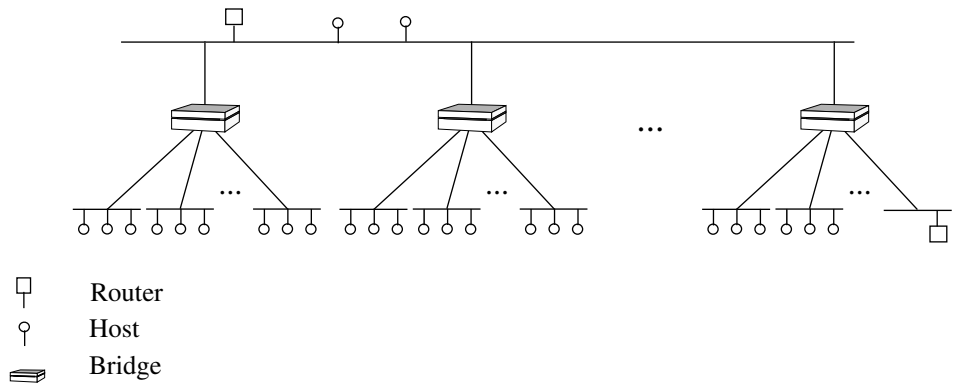

Figure 1:Example bridged LAN Topology.

We assume in the following discussion, without loss of generality, that advanced link layer services are built on top of prioritized medium access mechanisms as offered for example in FDDI, Token Ring or 100VGAny-LAN networks. Each host on a segment using such a prioritized access is expected to run the LLRMP. If a link technology cannot support several priority levels then the best effort traffic could be isolated e.g. by using a pure switched topology and priorities in switches.

On shared medium segments, rate regulation needs to be done in all hosts injecting prioritized data packets into a segment. The rate regulator controls the amount of prioritized data sent by the host in a certain time interval. This is controlled by the LLRMP. The transmission of best effort traffic is not restricted since it uses a separate output queue of lower priority. We expect the LLRMP and the rate

regulator to be implemented as part of the device driver of a LAN adapter card. The LLRMP entities of different adapter cards on the same node operate independently from each other.

Bridges are assumed to be able to classify data packets and forward them according to the service assigned to the corresponding session. A simple classification at the link layer for multicast flows might be based on the multicast destination address, assuming that multicast addresses are uniquely assigned within the bridged network. The classification for unicast flows is left for future discussion.

Whether or not rate regulators are required in bridges depends on the service, the reservation style and the link technology. For example, a controlled load service on bridged 100VGAny-LAN networks can be provided without rate regulators in bridges if: (1) the traffic is regulated at the source nodes which use the high priority access mechanism, and (2) distinct resources for each traffic source are allocated. If bridges do support rate regulation then this needs to be controlled by the LLRMP.

The LLRMP uses the soft state concept, as proposed in RSVP, for managing reservation state in hosts and bridges. Resources are reserved for simplex data streams. There is one important control message in the LLRMP protocol: the reservation request message. This message is used to create, modify, periodically refresh and tear down reservation state. In the absence of refreshes, the state automatically times out. Reservation request messages are sent using prioritized medium access e.g. the high priority service on Demand priority networks in order to minimize control packet loss. Resources are reserved independently on each segment in the data path, as the LLRMP request message travels through the bridged network. The reservation model is sender-based which ensures simplicity. Resources are requested at the node where the traffic enters the link.

The LLRMP includes two different strategies for admitting and rejecting reservation requests. The first uses an optimistic approach: after successfully performing admission control for a flow on the local node, the requested service is immediately enabled for data packets of the admitted flow. Possible reservation state inconsistencies on different nodes on the segment are resolved using a distributed reject mechanism whereby a service request can be rejected by any node on the same segment. The second approach is pessimistic. It uses an explicit acknowledgement message sent by a single dedicated node on each segment in order to permit the request. The requested service at the source node is only enabled after the source node has received the acknowledgement message. The dedicated node is automatically elected on each segment using a simple election mechanism.

Both admission strategies represent a compromise between control packet overhead and probability of service violation. For shared medium segments, we propose using the optimistic approach for services which do not provide hard guarantees e.g. a controlled load service, and the pessimistic approach for a guaranteed service. On full duplex switched segments, the pessimistic approach can be used for all types of service since the switch has complete control over the outgoing link.

Summarizing, the key features of the LLRMP protocol are:

a) The allocation concept is distributed, there is no central allocation manager for the entire bridged LAN. Each node using the protocol has complete knowledge about all reserved resources on its local network segment.

b) The protocol is basically a simple, single message protocol. Resources are reserved in *one pass*. Soft state is installed in hosts and bridges. The state is created, modified, torn down and refreshed by the source. Synchronization mechanisms between senders and receivers are assumed to be at a higher level than the link level (e.g. performed by RSVP, ST-II, or local network management).

c) The LLRMP protocol is independent of the network layer resource management mechanisms.

d) Resource management is supported for unicast and multicast data. Distinct or shared reservations can be created depending on (1) the reservation request from the user, (2) the ability of bridges to do policing and rate regulation.

e) The protocol supports heterogeneous receiver requests for the same multicast group. However it is assumed that the majority of receivers in the bridged LAN can be satisfied with the *receiver wild-card service* for this multicast group. Only special nodes e.g. routers and gateways might need different reservations because they connect receivers in the WAN.

f) LLRMP protocol state is only maintained along the data path between source and receiver and not in every bridge in the bridged network. This assumes standard learning bridges.

g) Reservation requests may be gathered into large packets in bridges to reduce the overall control traffic load.

h) The LLRMP protocol mechanisms are identical for end-nodes and bridges. However bridges are connected to multiple network segments. Bridges also have to make forwarding decisions for reservation request messages.

i) The LLRMP can operate in a heterogeneous bridged environment, where only a few bridges support the new protocol. Parts of the network which do not support the LLRMP are treated as a single logical segment by the LLRMP.

2.1 LLRMP Message Types

The LLRMP contains five control messages used to manage and maintain a consistent reservation state on all nodes on the segment. Note that most of these message types are only used for error state notification and recovery. If the protocol state is consistent then resources are reserved, modified and torn down with a *single* control message: the *reservation request* message. This assumes the optimistic setup approach. Reservations that exceed the resource limit are rejected using a *reject* message.

In order to periodically force all nodes to report their resources allocated, *query* messages are periodically multicasted. An *error* message reports the case when sufficient resources could not be reserved on a particular segment within the data path. If the reservation setup uses the pessimistic setup approach then this requires one additional message: the *acknowledgment* message.

2.1.1 The Reservation Request Message

The LLRMP performs a sender-based reservation setup. Resources are reserved on each intermediate segment when the reservation request message travels along the data path from the source to the receiver. A LLRMP reservation request contains a *service_id*, a flow specification *FSpec*, a traffic specification *TSpec*, and one or more receiver specifications *RecvSpec*.

service_id

The *service_id* selects the LLRMP service requested. This might for example identify the controlled load service. However, the end-to-end service provided is determined by the quality of service provided on the intermediate links along the data path. Different link technologies may offer different services so service parameters are mapped by the LLRMP whenever the link technology along the data path changes.

FSpec

The flow specification *FSpec* is used to identify the flow and to classify the corresponding data packets. Filter information from the *FSpec* is passed to the local packet classifier on each bridge in the data path. The filter may select parts of the link level header such as the MAC destination address and/or header fields of higher level protocols. We first assume a simple classification using only multicast MAC addresses.

TSpec

The *TSpec* characterizes the data stream injected into the network. It determines the resources reserved during admission control. If a reservation request is admitted, the *TSpec* is passed to the local rate regulator which enforces the conformance of the data flow to the traffic specification.

The *TSpec* is service specific. Its format and parameters are defined by the IETF Integrated Services working group. However the *TSpec* may optionally also contain link specific parameters required for admission control, for example data flow characteristics measured at link layer at the entrance of a segment. The distributed admission control strategy used in the LLRMP requires this information to be carried from the source to all other nodes on the local segment.

RecvSpec

The receiver specification *RecvSpec* identifies a single receiver or a group of receivers. A single *RecvSpec* consists of a receiver identifier *receiver_id* and the service request *RSpec,* requested by that receiver. In the unicast case, the *receiver_id* contains the unicast MAC address of the receiver. If resources are requested for a multicast group then the *receiver_id* can either contain the unicast MAC address of a special member of the group e.g. of a router, or a *receiver wildcard*. The *receiver wildcard* selects all receivers of a multicast group in the bridged LAN.

The service request *RSpec* is also service specific. The format is defined by the IETF Integrated Services working group. Note that the LLRMP has to understand *TSpec*s and *RSpecs* because it merges reservations in case the reservation type is shared.

2.1.2 The Reject Message

A LLRMP reject message is only sent if a node made an accidental over limit reservation. The message carries the *FSpec*, the *TSpec* and an *error_code*.

FSpec

The *FSpec* in reject message is used to identify the flow and the corresponding originator of the reservation request.

TSpec

The *TSpec* describes the traffic characteristic, the source node on the local segment is allowed to send. This ensures that a node keeps its old reservations when a new request for more resources is made. Note that reject messages are only used on shared segments. In a full duplex switched environment reject messages are never sent since each switch has complete control over the resources of each outgoing link.

error_code

The *error_code* specifies the reason why the reservation request failed.

2.1.3 The Query Message

The query message is primarily used by a node to quickly learn the reservation state on a local segment because all nodes receiving a query will report their reservation state within a defined time interval. The query message only consists of the general LLRMP header.

The parameters carried in the *acknowledgment* and *error* message are defined in a future draft. Note that a LLRMP control message may contain multiple reservation requests or rejects. This is especially useful when several reservations from the same source node e.g. a bridge need to be refreshed.

2.2 Receiver Model

A LLRMP reservation request message may include a single *RecvSpec*, or a list of *RecvSpecs*. This allows the protocol to differentiate different receivers of the same multicast group. For each *receiver_id* specified in the reservation request, the LLRMP reserves at least the corresponding *RSpec* along the data path to that receiver. The default quality of service specified in the receiver wildcard *RecvSpec* entry is applied to all receivers which are not in the list.

We expect that most receivers of the same multicast group can be served with the *receiver wildcard* service and so the list of *RecvSpecs* will typically be small. This is because most multicast receivers in the bridged LAN encounter the same network conditions and may thus be satisfied with the same quality of service. Only a few special nodes like routers or gateways might need a more stringent service e.g. a lower delay bound, because they connect multicast receivers in the WAN. In this case a large fraction of the end-to-end delay is consumed in the WAN which may then impose a tighter delay bound on the LAN. However a single reservation message can, depending on the service, accommodate about 100 different *RecvSpecs*, which seems to be sufficient for existing LANs which usually have only one

or two connection points to the outside world. If a larger number is required, then the reservation request can be split over multiple packets

We chose the sender based reservation setup because it allows resources to be reserved with only a single message. This simplifies the design and ensures a fast setup. The protocol scales well as long as the list of *RecvSpecs* within the reservation request message is sufficiently small per multicast group.

2.3 Reservation Model

Resources are reserved in *one pass* by the LLRMP. There is no mechanism for end-hosts to adjust reservations on single segments within the data path e.g. to decrease the delay bound on one link and relax the request on another.

A heterogeneous reservation model would require additional control messages travelling along the reverse data path from the receivers back to the data sources. Whether such a reservation model is required in bridged/switched LANs is left for further study.

However, the LLRMP reservation request message may carry service parameters e.g. the end-to-end delay, the bridge-hop count, etc. . This would allow support of a heterogeneous reservation model similar to the *One Pass with Advertising* (OPWA) scheme presented in [7].

2.4 Protocol Mechanisms

2.4.1 Reservation and Reject Mechanism - The Optimistic Approach

Whenever the LLRMP receives a reservation request for a segment, it first performs admission control using its current reservation data base. If the test fails then the reservation request is immediately rejected without any signalling on the network. If the admission control was successful, then the LLRMP updates its local data base and sends a reservation request message onto the local segment. It also informs its local classifier and rate regulator about the existence of the new flow. After the classifier has installed the filter for the new flow, all data packets matching this filter are transmitted using the specified medium access mechanism e.g. a certain priority level specified by the admission control scheme.

The reservation request message is multicasted to all other LLRMP entities on the local segment using the highest priority medium access mechanism available on the segment. The message is addressed to a well-known, LLRMP specific multicast address. After receiving the request, each node itself does admission control for the request using its own view of the total reserved resources and, if successful, updates its local data base.

If the admission control fails then a node assembles a reject message and schedules it for transmission within a random time interval between > 0 and *TD* seconds. This process happens on all nodes. However there is one node on the segment which is allowed to send its reject immediately. This is the elected *arbiter* of the network segment.

Whenever other nodes receive a reject from the segment arbiter, they cancel the reject message they have scheduled, if they have one. This mechanism ensures that requests on a segment are usually rejected by just one node: the arbiter. However all other nodes set up a reject message in case the arbiter fails or has not received the request. After receiving a reject from the network, the originator of the request updates its data base and reports the error to the calling application.

Note that LLRMP-aware bridges do not flood reservation request messages. They usually only forward request messages along the data path using the forwarding rules provided in section 2.4.6 .

Robustness of The Optimistic Approach

Whenever the reservation state on all nodes on the segment is consistent, then reservation requests which cannot be served due to a lack of resources are rejected by the admission control on the local node. No signalling is carried out. However, if reservation messages go missing, or service requests are made at the same time on different nodes, then the reservation state in the network becomes inconsistent. This might affect the service of all flows using a certain priority level.

The Optimistic Approach can tolerate the loss of a single control message without any impact. Two or even more reservation requests can go missing at several nodes (e.g. due to congestion) as long as there is at least one node on the segment which has received all the messages. This doesn't have to be the arbiter since all nodes are able to reject reservation requests. Any failure of the arbiter is resolved by the arbiter election process discussed later in section 2.4.4 . If the arbiter crashes then a new node is automatically elected.

Inconsistencies of the reservation state are resolved by the refresh mechanism. Reservation messages periodically update the status information on all nodes in the network. The frequency of the refresh is determined by a query interval which is discussed later.

The LLRMP reserves some resources on each segment for its control traffic. The probability that reservation messages go missing is very small since control messages are transmitted over a single segment using a high priority medium access mechanism. In bridges and switches, control messages are passed to the processor queue and not to a (possibly congested) link output queue. It should be emphasized that with the optimistic approach, the quality of the link level service is only affected if over-limit reservations are made during inconsistent reservation state conditions.

2.4.2 Reservation and Reject Mechanism - The Pessimistic Approach

The pessimistic approach uses an explicit acknowledgment of reservation requests to prevent any possible service violation due to inconsistent reservation state on different nodes on the segment. As in the previous approach, after invocation, the LLRMP first performs admission control using the current local reservation data base. Requests are immediately rejected if this test fails. If the admission control was successful, then the LLRMP updates the local data base and sends a reservation request message onto the local segment. However, unlike the optimistic approach, the local classifier and the local rate regulator are not updated until the request was acknowledged by the segment arbiter.

After receiving the request message all nodes, except the segment arbiter, immediately update their local data base. Admission control for the request is actually carried out by the arbiter, using its own view of all the resources reserved on the segment. If the test fails then the arbiter sends a reject message back. The reject is also multicasted, so that all nodes on the segment can adjust their local data base according to the *TSpec* received within the reject message. The originator of the reservation request additionally informs the calling application, or reports an error back to the source if the originator was a bridge.

The arbiter returns an acknowledgement message if the request passed the admission control test. Upon receiving this acknowledgement, the originator of the request informs its local classifier and the rate regulator about the new flow. This enables the requested quality of service and the rate control mechanism for this flow.

The loss of a reservation request or reject message is covered by the periodic refresh mechanism. To improve the recovery time in case of control packet loss, every node decreases the refresh interval for all flows with locally admitted, but unacknowledged resources. As in the Optimistic Approach, any failure of the arbiter is resolved by the dynamic arbiter election process discussed later.

The pessimistic approach prevents any possible service violation due to inconsistent state conditions at the expense of an explicit acknowledgement for each new reservation request. However, the additional overhead seems to be acceptable since each segment in the bridged LAN has its own arbiter which keeps the overall approach decentralized. Furthermore, all nodes which are not the arbiter reject local reservation requests if their local admission control failed.

2.4.3 Reservation Refresh Mechanism

The reservation state on all nodes is aged out if not refreshed. Each LLRMP node periodically refreshes the reservation information held about it at all other nodes on the segment by sending a reservation message. All refreshes are synchronized by a query mechanism. The arbiter periodically sends a LLRMP query message onto the network. Whenever a node receives a query, it responds with a reservation message reporting the current local reservation state. The reservation request is scheduled within an arbitrary time interval of length *TD* to avoid congestion. Note that each node only reports its own flows and not the ones learned from other nodes on the segment.

The query mechanism allows a node to quickly learn the total reservation state on the local segment. It is also used to elect the segment arbiter.

2.4.4 Dynamic Segment Arbiter Election

By default, the arbiter is the node with the lowest MAC address on the segment. In bridged networks, each segment has its own arbiter which might be an end-host or a bridge.

The arbiter election uses the query mechanism. If an LLRMP node comes up, then it first assumes that it is the arbiter and thus queries the network by multicasting query messages. After receiving a query

with a lower MAC source address, a node will reschedule its own query timer to be longer than the current value. If the new node is itself the new arbiter then its query messages will update the query timer on all other nodes, if not, then it will itself receive a query with a lower address. If the arbiter fails, the query timer on other nodes runs out and finally another node becomes the arbiter.

Optionally a node can set an *arbiter flag* in the LLRMP header of its query messages. This forces all nodes which do not have the *arbiter flag* set themselves to update their query timer, even if they have a lower MAC address than the originator of the query message. The *arbiter flag* allows a dedicated node e.g. a network management station or a bridge to become the arbiter of the segment regardless of its MAC address.

2.4.5 Request Modifications and Tear Down

LLRMP allows nodes to dynamically change their reservations. As with new requests the availability of new resources is first checked by the admission control on the local node. After a successful admission, the new reservation state is distributed using the normal reservation request mechanism. Note that a node always keeps its reservations while it attempts to increase them.

Reservation messages are also used to release resources. Nodes which want to decrease or delete resources send a message with the new parameter set. If all resources for a flow are to be released then *NULL* values for all resource-parameters are distributed within the *TSpec* of the reservation request message. This allows resources to be torn down quickly.

2.4.6 Bridged Networks

The LLRMP protocol mechanisms are identical for end-nodes and bridges. However bridges are connected to multiple network segments and thus have to manage reservation state for each of their segments. For each reservation message received, bridges have to do the following tasks:

1. Perform admission control for the segment from which the request message was received. This assumes that the optimistic approach is used for the reservation setup. If the pessimistic approach is selected e.g. identified by the link level service used, then bridges only perform admission control if they are the arbiter for this segment. This was discussed in the previous section.
2. Look up the local MAC address table and decide whether and what reservation request needs to be forwarded onto other segments. This uses the unicast MAC addresses of the receivers, carried in the *RecvSpecs* of the reservation request message.
3. Perform admission control for each of the segments to which the request needs to be forwarded. If this test fails, send an error message back to the originator of the reservation request. Resources on the previous segments in the data path are not torn down by the error message. This is left to the originator who can then either: (1) use the existing reservations made, even if the setup to a single or several receivers failed, (2) change the reservation request and try again, or (3) tear all reservations down.
   If the admission control was successful, then the bridge assembles the corresponding reservation request message and forwards it onto the segment.

The forwarding decision is based on matching address information found in the reservation request message and in the MAC address table of bridges. The LLRMP makes its forwarding decision according to the rules listed below. First assume unicast:

1. A reservation request does not need to be forwarded to other segments if the receiver MAC address is registered in the MAC address table for the same port from which the reservation message was received (Leaf Rule).
2. If the bridge has a table entry for the receiver on a port which is different to the one from which the request was received, then it only forwards the entry through that port (Direct Path Rule).
3. If there is no receiver entry in the address table then the reservation request is forwarded to all segments, except the one from which the message was received (Flood Rule).

Multicast additionally requires the consideration of link layer multicast routing entries matching the multicast destination address of the session. We assume a multicast routing mechanism as currently being standardized in IEEE 802.1p.

If a *receiver wildcard* is specified in the request, then the reservation message is forwarded to all segments on which the multicast group is registered. The rules for unicast forwarding are additionally applied to all unicast receiver addresses specified within the list of *RecvSpecs* of the reservation request message. Whenever multiple receivers share a data path then the superset of their *RSpecs* is allocated. Note that a reservation request for a multicast session does not have to contain a receiver wildcard request. In this case, resources are only reserved along the data path towards group members explicitly listed in the reservation request message.

Whenever a reservation request contains multiple *RecvSpecs* and the forwarding rule (1) or (2) applies for one of the unicast MAC addresses listed in the reservation request message, then the corresponding *RecvSpec* entry is removed from the request message, if that message needs to be forwarded onto any of the segments on which the MAC address is not registered. This is illustrated later in Figure 2.

The forwarding process ensures that reservation messages are only forwarded along the data path between sender and receiver(s). This relies on MAC address entries in the MAC address table of the bridges. For reservation requests from RSVP and STII, these entries are likely to be there because: (1) network layer management control messages are periodically exchanged. These are *PATH* and *RESV* messages in RSVP and *HELLO* messages in ST-II. (2): applications with stringent service requirements are likely to be interactive, and will thus have a return control channel and/or a return data path.

However, the LLRMP still works when these address table entries are missing. In this case, reservation messages become forwarded to all segments and will reserve the requested resources on the entire LAN. This is equivalent to a network layer based allocation policy, where the link level structure of the network remains hidden. Eventually, data packets will flow causing entries to be made in MAC address tables of bridges and old reservations to time out on unused segments.

An Example

The LLRMP reservation setup in a bridged LAN is illustrated in Figure 2 for a multicast session using the group identifier *g*. There are three local multicast receivers *R1, R2, RT* and one data source *S1*. The receivers *R1* and *R2* are to be served with the receiver wildcard service *RSpec_W*. The router *RT* connects one or more receivers in the WAN. The local service requested for *RT* is *RSpec_RT*. We assume that *RSpec_RT* is more stringent than *RSpec_W* e.g. contains a lower delay bound, higher bandwidth request etc. . Symbolically the LLRMP reservation request can be represented by:

LLRMP_REQUEST (*service_id, FSpec, TSpec_S1, RecvSpec_RT, RecvSpec_W*)

where *service_id* is the requested service identifier, *FSpec* the flow specification, *TSpec_S1* the traffic characterisation of source *S1*, *RecvSpec_RT* the receiver specification for router *RT*, and *RecvSpec_W* the wildcard receiver specification. The Flow specification *FSpec* contains the address of *S1* and the group identifier *g*. This assumes a classification using only the multicast destination address for this example. The symbolic representation of the *FSpec* is:

FSpec (*Addr_S1 , g*)

where *Addr_S1* is the MAC address of source *S1* and *g* the multicast group. The receiver specifications for router *RT* and the wildcard entry can be represented by:

RecvSpec_RT (*Addr_RT, RSpec_RT*)
RecvSpec_W (**, RSpec_W*)

where *Addr_RT* denotes the unicast MAC address of router *RT* and the asterisk the wildcard receiver address. The LAN in Figure 2 consists of five segments and three bridges. The existing routing information is illustrated for each bridge port, e.g. the multicast group *g* is registered on port *p2* of bridge *B1*. Port *p3* of *B1* additionally holds an entry *Addr_RT* for the unicast MAC address of router RT.

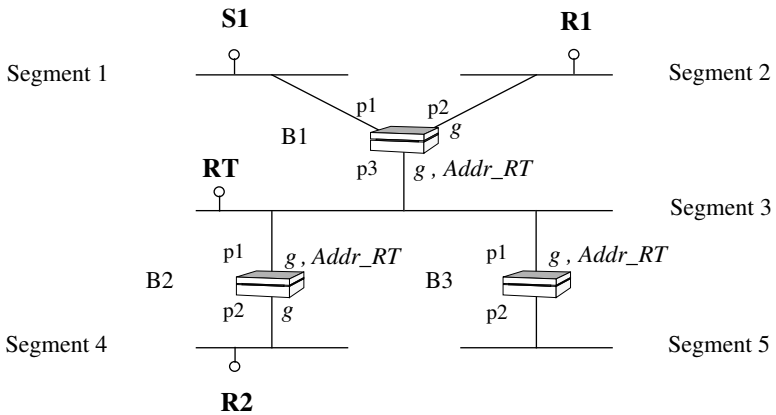

Figure 2:Example - LLRMP Reservation Setup in a bridged LAN.

The reservation setup starts on segment 1 when source *S1* performs admission control and multicasts a reservation request message to all other nodes on the segment. The reserved resources on segment 1 are *RSpec_RT* because *RSpec_RT > RSpec_W.*

After receiving the request message, bridge *B1* looks up its MAC address table and finds a multicast routing entry for group *g* on port *p2* and *p3*, and the unicast entry *Addr_RT* for *RT* on port *p3*. For entry *Addr_RT*, the forwarding rule (2) (Direct Path) applies, so the corresponding reservation *RSpec_RT* only needs to be reserved on port *p3*. Thus bridge *B1* allocates *RSpec_RT* for port *p3* and forwards the request message onto the corresponding segment *3*. Since *B1* has also a multicast routing entry on its port *p2*, it must also reserve resources for the corresponding segment *2*. However only resources according to *RSpec_W* need to be reserved so the request message forwarded onto segment *2* has the format:

LLRMP_REQUEST (*service_id, FSpec, TSpec_S1, RecvSpec_W*)

The receiver specification *RecvSpec_RT* has been removed since router *RT* is known to be reachable through port *p3* on *B1*.

Meanwhile, the reservation request message forwarded onto segment *3* travels to bridge *B2* and *B3*. On bridge *B2*, the forwarding rule (1) (Leaf Rule) applies for the unicast entry *Addr_RT.* However since the multicast group *g* is also registered on port *p2*, resources according to *RSpec_W* become reserved for segment *4*. The request message forwarded has the same format as the one forwarded onto segment 2. Bridge *B3* does not reserve any resources according to forwarding rule (1), because router *RT* and group *g* are registered on the port on which the request message arrives. Note that the reservation request message is not forwarded onto segment *5*. After the reservation setup is finished, the following resources are reserved:

Segment 1 : *RSpec_RT*
Segment 2 : *RSpec_W*
Segment 3 : *RSpec_RT*
Segment 4 : *RSpec_W*
Segment 5 : none

2.4.7 Dynamic Topology Changes

The LLRMP automatically recovers from dynamic topology changes using the reservation refresh mechanism. Whenever the bridged topology changes, reservation refreshes are forwarded along the new data path and will immediately reserve any missing resources. Old reservation state in bridges will time out since not refreshed.

2.4.8 Heterogeneous bridged LANs

The LLRMP can operate in a heterogeneous environment where only parts of the bridged LAN support the new protocol. Existing bridges and switches will forward LLRMP reservation messages through all

their ports since a new ethertype and a multicast address are used to exchange protocol status informations. Non LLRMP clouds between two LLRMP nodes/bridges are treated as one single logical segment for allocating resources.

3. Service Mapping

At invocation, the LLRMP receives the request for a certain service specified in the service identifier. During reservation setup, this service may be mapped to a link specific service whenever the link technology along the data path changes. Different technologies have different properties and might therefore provide different services. Resources for a particular service might also be completely utilized, when a request arrives, but other services might still have capacity available. A link technology may also only provide a guaranteed service and emulate other services by reserving link resources within the guaranteed service. The mapping rules between different services are for further drafts.

4. LLRMP Relationship to RSVP, ST-II

RSVP and ST-II are the resource setup protocols designed for an Integrated Services Internet. They are used by applications to request a specific quality of service from the network. After receiving a request, RSVP or ST-II reserve resources on each link along the way between source and receiver(s).

If a LAN is involved in the data path, then the request for the link is passed to the LLRMP. Reservation requests are made at the node where the traffic enters the bridged LAN. This node could be an end-host, a router or a gateway. The service calling conventions are straightforward for ST-II because it performs a sender initiated resource reservation. During reservation setup, an ST-II *CONNECT* message travels downstream from the source towards the receiver(s). It contains the flow specification of the source and a list of all targets. In contrast, RSVP uses a receiver based model. *PATH* messages carry the sender's traffic characteristic (*TSpec*) downstream. *RESV* messages are used for signalling the service request (*RSpec*) of the receiver(s). They travel upstream towards the source.

RSVP makes a reservation request for a flow to the LLRMP after is has received a *PATH* and a *RESV* message for this flow, because this provides the required information (*TSpec, RSpec*) needed to do the request.

4.1 Support of RSVP Reservation Styles

RSVP offers several reservation styles. These styles are defined according to the reservation type and the sender selection used. Reservations are either *shared*: when data packets from different senders use the same reservation on a link, or *distinct*, when resources are established for each sender of a session. The sender selection is either *explicit*: which requires a list of all selected senders, or a *wildcard* which implicitly selects all *senders* to a session. RSVP currently defines the *Fixed-Filter- (FF)*, the *Shared-Explicit- (SE)*, and the *Wildcard-Filter (WF)* style. Their mapping onto LLRMP mechanisms is discussed in the following:

4.1.1 Fixed-Filter Style Reservations

The *Fixed-Filter (FF)* style establishes a distinct reservation for explicit senders. The mapping of a *FF* reservation onto the LLRMP is straightforward. An example for a *Fixed-Filter* reservation was discussed for Figure 2 in section 2.3.6.

For unicast flows, RSVP requests Fixed-Filter style reservations by passing the service identifier *service_id,* the flow specification *FSpec*, the traffic characterisation *TSpec*, and the receiver specification *RecvSpec_R* to the LLRMP. The receiver specification *RecvSpec_R* contains the receiver address *Addr_R* and the service request *RSpec_R* for that receiver. Symbolically, the service request can be represented by:

FF (*service_id, FSpec, TSpec, RecvSpec_R*)

For multicast, the service request can contain one or a list of receiver specifications. One *RecvSpec* may contain a receiver wildcard which specifies the default service for the group. A multicast service request can be represented by:

FF (*service_id, FSpec, TSpec, RecvSpec_R1, RecvSpec_R2, ... , RecvSpec_W*)

where *RecvSpec_R1*, *RecvSpec_R2*, ... , etc. denote the receiver specification for the receivers *R1, R2,* ... , etc. *RecvSpec_W* is the receiver wildcard service specification.

The LLRMP reserves *RSpec_R1* along the data path to *R1*, *RSpec_2* along the data to *R2*, ... , and *RSpec_W* for all other multicast receivers in the group. If parts of the data path overlap or the required routing information is not present in a particular bridge then the 'maximum' of *RSpec_1*, *RSpec_2*, ... , and *RSpec_W* is allocated, where the maximum denotes the most stringent service requirement. Note that the wildcard receiver specification *RecvSpec_W* can be omitted in the request.

4.1.2 Shared Reservations

Shared reservations are being studied. In this section we only discuss the basic concept used by the LLRMP to support shared reservations. The exact mechanisms are defined in a future draft. The discussion assumes that bridges support rate regulation on a per flow basis.

Shared reservations are made using the rule:

> In bridged networks, shared reservations can be made when flows from different sources share the same output port of a bridge/switch. If data packets are injected onto the same segment from different source nodes then distinct reservations have to be reserved.

This is because there is no sychronization mechanism between different nodes on existing link technologies which can ensure that the aggregate traffic injected by several nodes is smaller than the resources reserved.

Figure 3 illustrates a simple example with two data sources *S1* and *S2* and one receiver *R* in a single-hop bridged LAN topology. Both sources send to the multicast group *g*, which was joined by receiver *R*.

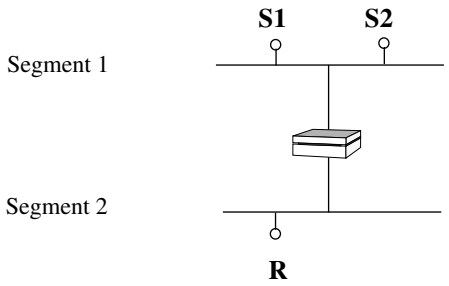

Figure 3:LLRMP Shared Reservation on a bridged LAN.

A reservation request is made at both sources *S1* and *S2*. Assume that the request specifies that *B* resources are to be shared between all sources sending to group *g*. This corresponds to a RSVP *Wild-card Filter* reservation. Then assume that the reservation request is first invoked at *S1*. The LLRMP reserves *B* resources on segment 1 and *B* resources on segment 2 for source *S1* . The reservation setup for that was already discussed.

If the reservation request is now invoked at source *S2* then the LLRMP will also reserve *B* resources for source *S2* on segment 1. Even though the request specified shared reservations, distinct reservations have to be made for *S1* and *S2* on segment 1. However the resources on segment 2 can be shared because the rate regulator on the bridge ensures that the aggregate traffic from *S1* and *S2* never exceeds the reservation *B*. Note that in this example, the reservation request message from *S2* does not have to be forwarded onto segment 2, since shared resources were reserved earlier by the reservation request message from source *S1*.

After the reservation setup is finished, the LLRMP has reserved *2B* resources on segment 1 and *B* resources on segment 2.

5. Functional Specification

The exact message formats are defined in a future draft.

6. References

[1] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, *Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification*, Internet-Draft draft-ietf-rsvp-spec-12.ps, May 1996.

[2] L. Delgrossi, L. Berger, *Internet Stream Protocol Version 2 (ST2), Protocol Specification - Version ST2+*, RFC 1819, August 1995.

[3] S. Shenker, C. Partridge, *Specification of Guaranteed Quality of Service*, Internet Draft, December 1995.

[4] S. Shenker, C. Partridge, J. Wroclawski, Editors, *Specification of Controlled Delay Quality of Service*, Internet Draft, November 1995.

[5] J. Wroclawski, Editor, *Specification of the Controlled-Load Network Element Service*, Internet Draft, November 1995.

[6] S. Shenker, C. Partridge, B. Davie, L. Breslau, Editors, *Specification of Predictive Quality of Service*, Internet Draft, 1995.

[7] S. Shenker, L. Breslau, *Two Issues in Reservation Establishment*, in Proc. of ACM SIGCOMM '95, pp. 14 - 26, Cambridge, MA, August 1995.

[8] *IEEE 802.1D, International Standard, Information technology - Telecommunications and information exchange between systems - Local area networks - Media access control (MAC) bridges*, ISO/IEC 10038:1993.

Authors Address

Peter Kim
Hewlett Packard Laboratories Bristol
Filton Road, Stoke Gifford
Bristol BS12 6QZ. U. K.
pk@hplb.hpl.hp.com
+44 117 922 8357