IEEE 802.17 Resilient Packet Ring Tutorial

Fredrik Davik, Simula Research Laboratory and Ericsson Research
Mete Yilmaz, Cisco Systems
Stein Gjessing, Simula Research Laboratory
Necdet Uzun, Cisco Systems

ABSTRACT

IEEE Working Group 802.17 is standardizing a new ring topology network architecture, called the Resilient Packet Ring, to be used mainly in metropolitan and wide area networks. This article presents a technology background, gives an overview, and explains some of the design choices behind RPR. Some major architectural features are illustrated and compared by showing performance evaluation results using the RPR simulator developed at Simula Research Laboratory using the OPNET Modeler simulation environment.

INTRODUCTION

The Resilient Packet Ring (RPR, IEEE 802.17) is a ring-based network protocol being standardized by IEEE [1]. Packet ring-based data networks were pioneered by the Cambridge Ring [2], and followed by other important network architectures, notably MetaRing [3], Token Ring [4], FDDI [5], ATMR [6], and CRMA-II [7].

Rings are built using several point-to-point connections. When the connections between the stations are bidirectional, rings allow for resilience (a frame can reach its destination even in the presence of a link failure). A ring is also simpler to operate and administrate than a complex mesh or an irregular network.

Networks deployed by service providers in MANs or WANs are often based on SONET/SDH rings. Many SONET rings consist of a dual-ring configuration in which one of the rings is used as the backup ring, and remains unused during normal operation, utilized only in the case of failure of the primary ring. The static bandwidth allocation and network monitoring requirements increase the total cost of a SONET network. While Gigabit Ethernet does not require static allocation and provides cost advantages, it cannot provide desired features such as fairness and auto-restoration.

Since RPR is being standardized in the IEEE 802 LAN/MAN families of network protocols, it can inherently bridge to other IEEE 802 networks and mimic a broadcast medium. RPR implements a medium access control (MAC) protocol for access to the shared ring communication medium that has a client interface similar to that of Ethernet's.

The rest of this article is organized as follows. Ring network basics and RPR station design are discussed. The so-called fairness algorithm is presented, and topology discovery, resilience, and bridging are then discussed. Finally, frame formats are outlined and a conclusion is given. In order to demonstrate different operational modes, some performance figures are included and discussed. The scenarios have been executed on the RPR simulator model developed at Simula Research Laboratory and implemented in OPNET Modeler [8], according to the latest RPR draft standard of November 2003 (v. 3.0).

RING NETWORK BASICS

In unicast addressing (broadcast will be covered later), frames are added onto the ring by a sender station that also decides on which of the two counterrotating rings (called ringlet 0 and ringlet 1 in RPR) the frame should travel to the receiving station. If a station does not recognize the destination address in the frame header, the frame is forwarded to the next station on the ring. In RPR, the transit methods supported are cut-through (the station starts to forward the frame before it is completely received) and store-and-forward.

To prevent frames with a destination address recognized by no station on the ring from circulating forever, a time to live (TTL) field is decremented by all stations on the ring.

When an RPR station is the receiver of a frame, it removes the frame completely from the ring instead of just copying the contents of the frame and letting the frame traverse the ring back to the sender. When the receiving station removes the frame from the ring, the bandwidth otherwise consumed by this frame on the path back to the source is available for use by other sending stations. This is generally known as spatial reuse.

0163-6804/04/$20.00 © 2004 IEEE

Figure 1 shows an example scenario where spatial reuse is obtained on the outer ring; station 2 is transmitting to station 4 at the same time as station 6 is transmitting to station 9.

The ring access method is an important design choice. A token may circulate the ring so that the station holding the token is the only station allowed to send (as in Token Ring). An alternative access method, called a buffer insertion ring, was developed as early as 1974 [9, 10], and utilized later in protocols like MetaRing [3], CRMA-II [7], SCI [11], and SRP [12].

Every station on the ring has a buffer (called a *transit queue*; Fig. 1) in which frames transiting the station may be temporarily queued. The station must act according to two simple rules. The first principle is that the station may only start to add a packet if the transit queue is empty and there are no frames in transit. Second, if a transiting frame arrives after the station has started to add a frame, this transiting frame is temporarily stored (for as long as it takes to send the added frame) in the transit queue.

Obviously, these two simple principles need some improvement to make up a full working protocol that distributes bandwidth fairly. How this is achieved in RPR will be revealed in the next sections.

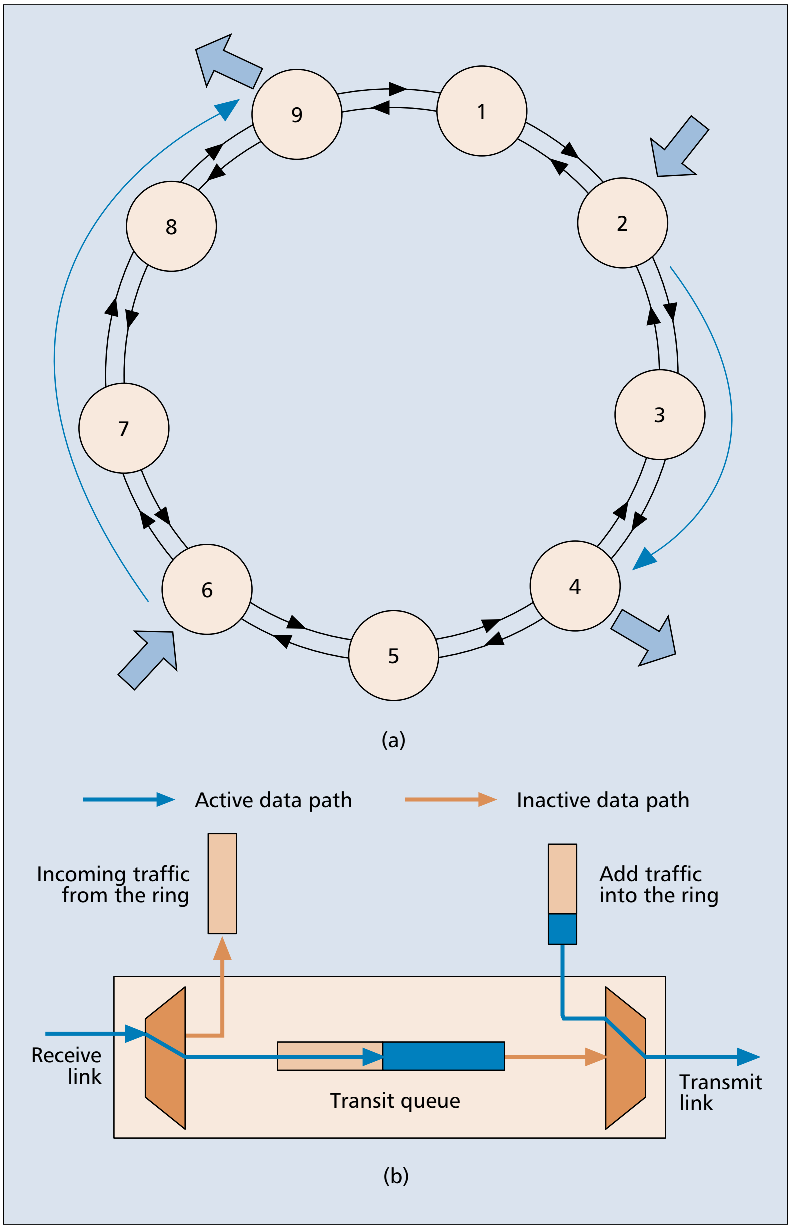
Station Design and Packet Priority

The stations on the RPR ring implement a MAC protocol that controls the stations' access to the ring communication medium. Several physical layer interfaces (reconciliation sublayers) for Ethernet (called PacketPHYs) and SONET/SDH are defined. The MAC entity also implements access points clients can call in order to send and receive frames and status information.

RPR provides a three-level class-based traffic priority scheme. The objectives of the class-based scheme are to let class A be a low-latency low-jitter class, class B be a class with predictable latency and jitter, and class C be a best effort transport class. It is worthwhile to note that the RPR ring does not discard frames to resolve congestion. Hence, when a frame has been added onto the ring, even if it is a class C frame, it will eventually arrive at its destination.

Class A traffic is divided into classes A0 and A1, and class B traffic is divided into classes B-CIR (committed information rate) and B-EIR (excess information rate). The two traffic classes C and B-EIR are called fairness eligible (FE), because such traffic is controlled by the fairness algorithm described in the next section.

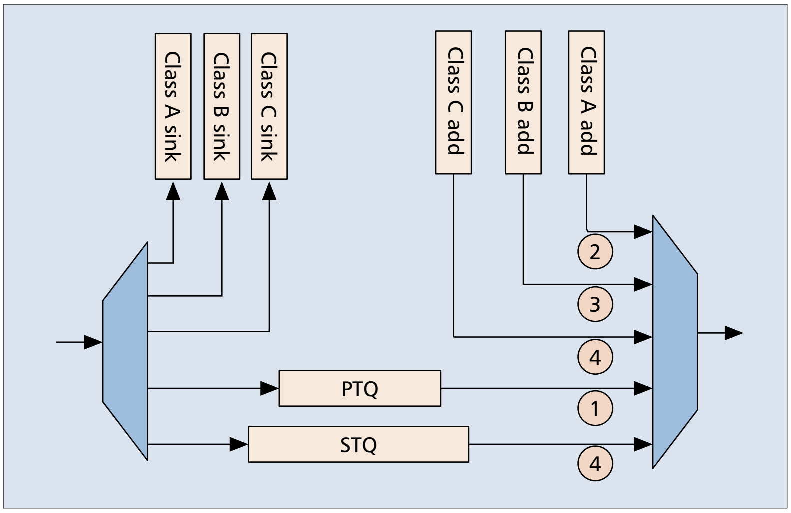
In order to fulfill the service guarantees for class A0, A1, and B-CIR traffic, bandwidth needed for these traffic classes is preallocated. Bandwidth preallocated for class A0 traffic is called *reserved* and can only be utilized by the station holding the reservation. Bandwidth preallocated for class A1 and B-CIR traffic is called *reclaimable*. Reserved bandwidth not in use is wasted. Bandwidth not preallocated and reclaimable bandwidth not in use may be used to send FE traffic.



■ **Figure 1.** *An RPR network: a) destination stripping and spatial reuse illustrated on the outer ring (ringlet 0); b) a station's attachment to only one ringlet, showing the insertion buffer or transit queue that stores frames in transit, while the station itself adds a frame.*

A station's reservation of class A0 bandwidth is broadcast on the ring using topology messages (topology discovery is discussed later). Having received such topology messages from all other stations on the ring, every station calculates how much bandwidth to reserve for class A0 traffic. The remaining bandwidth, called *unreserved rate*, can be used for all other traffic classes.

An RPR station implements several traffic shapers (for each ringlet) that limit and smooth the add and transit traffic. There is one shaper for each of A0, A1, and B-CIR as well as one for FE traffic. There is also a shaper for all transmit traffic other than class A0 traffic, called the *downstream shaper*. The downstream shaper ensures that the total transmit traffic from a station, other than class A0 traffic, does not exceed the unreserved rate. The other shapers are used



■ **Figure 2.** *The attachment to one ring by a dual-transit-queue station. The numbers in the circles give a very crude indication of transmit link priority.*

to limit the station's add traffic for the respective traffic classes.

The shapers for classes A0, A1, and B-CIR are preconfigured; the downstream shaper is set to the unreserved rate, while the FE shaper is dynamically adjusted by the fairness algorithm.

While a transit queue of one maximum transmission unit (MTU) is enough for buffering of frames in transit when the station adds a new frame into the ring, some flexibility for scheduling of frames from the add and transit paths can be obtained by increasing the size of the transit queue. For example, a station may add a frame even if the transit queue is not completely empty. Also, a larger queue may store lower-priority transit frames while the station is adding high-priority frames. The transit queue could have been specified as a priority queue, where frames with the highest priority are dequeued first. A simpler solution, adopted by RPR, is to optionally have two transit queues. Then high-priority transit frames (class A) are queued in the primary transit queue (PTQ), while class B and C frames are queued in the secondary transit queue (STQ). Forwarding from the PTQ has priority over the STQ and most types of add traffic. Hence, a class A frame traveling the ring will usually experience not much more than the propagation delay and some occasional transit delays waiting for outgoing packets to completely leave the station (RPR does not support preemption of packets). Figure 2 shows one ring interface with three add queues and two transit queues. The numbers in the circles indicate a crude priority on the transmit link.

An RPR station may have one transit queue only (the PTQ). In order for class A traffic to move quickly around the ring, the transit queues in all single transit queue stations should then be almost empty. This is achieved by letting transit traffic have priority over all add traffic, and requiring all class A traffic to be reserved (class A0). Hence, there will always be room for class A traffic, and class B has priority over class C add traffic, just as in a two-transit-queue station.

Figure 3 shows an example run where the latency of frames sent between two given stations on an RPR ring is measured. The stations on the ring have two transit queues. The ring is overloaded with random background class C traffic. Latency is measured from when a packet is ready to enter the ring (i.e., first in the add queue) until it arrives at the receiver. Notice how class A traffic keeps its low delay even when the ring is congested. Also notice how class B traffic still has low jitter under high load, while class C traffic experiences some very high delays.

An RPR ring may consist of both one- and two-transit-queue stations. The rules for adding and scheduling traffic are local to the station, and the fairness algorithm described below works for both station designs.

The RPR Fairness Algorithm

In the basic buffer insertion access method, a station may only send a frame if the transit queue is empty. Thus, it is very easy for a downstream station to be starved by upstream ones. In RPR, the solution to the starvation problem is to force all stations to behave according to a specified fairness algorithm. The objective of the fairness algorithm is to distribute unallocated and unused reclaimable bandwidth fairly among the contending stations and use this bandwidth to send class B-EIR and class C (FE) traffic.

When defining fair distribution of bandwidth, RPR enforces the principle that when the demand for bandwidth on a link is greater than the supply, the available bandwidth should be fairly distributed between the contending sender stations. A weight is assigned to each station, so a fair distribution of bandwidth need not be an equal one.

When the bandwidth on the transmit link of a station is exhausted, the link and station are said to be congested, and the fairness algorithm starts working. The definition of congestion is different for single- and dual-queue stations, but both types of stations are congested if the total transmit traffic is above certain thresholds. In addition, a single-queue station is congested if frames that are to be added have to wait a long time before they are forwarded, and a dual-queue station is congested if the STQ is filling up (and hence transit frames have to wait a long time before they are forwarded).

The most probable cause of congestion is the station itself and its immediate upstream neighbors. So, by sending a so-called fairness message upstream (on the opposite ring), the probable cause of the congestion is reached faster than by sending the fairness message downstream over the congested link. Figure 4 shows how the attachment to one ring asks the other attachment to queue and send a fairness message. In the following we focus on fairness on one ring. The fairness algorithm on the other ring works exactly the same way.

When a station becomes congested it calculates a first approximation to the fair rate by either dividing the available bandwidth between all upstream stations that are currently sending frames through this station, or using its own current add rate. This calculated value is sent upstream to all stations that are contributing to the congestion, and these stations must adjust their sending of FE traffic accordingly. The recipients of this message together with the originating station constitute a congestion domain.

There are two options specified for the fairness algorithm. In the *conservative* mode the congested station waits to send a new fair rate value until all stations in the congestion domain have adjusted to the fair rate, and this change is observed by the congested station itself. The estimate of the time to wait (called the fairness round-trip time, FRTT) is calculated by sending special control frames across the congestion domain. The new fair rate may be smaller or larger than the previous one, depending on the observed change.

In the *aggressive* mode, the congested station continuously (fairness packets are sent with a default interval of 100 μs) distributes a new approximation to the fair rate. When the station finally becomes uncongested, it sends a fairness message indicating no congestion. A station receiving a fairness message indicating no congestion will gradually increase its add traffic (assuming the station's demand is greater than what it is currently adding). In this way (if the traffic load is stable) the same station will become congested again after a while, but this time the estimated fair rate will be closer to the real fair rate, and hence the upstream stations in the congestion domain do not have to decrease their traffic rate as much as previously.

Figure 5 shows how the aggressive and conservative modes of the fairness algorithm work for a given scenario. Both scenarios are simulated for a 16-station ring, with 50 km 1 Gb/s links of which each station uses 1 percent for A0 traffic. All stations are dual-queue designs, and stations 1, 2, and 3 are sending to station 4. The traffic starts at time 1.0 s, and initially only station 3 is sending. At time 1.1 s, station 1 starts sending. Both of these flows are greedy class C flows, and both fairness methods are quick to share the bandwidth on the congested link (from station 3 to station 4) equally. At time 1.2 s, station 2 starts sending a 200 Mb/s flow (also class C frames to station 4). We see that the aggressive method very quickly (but after some oscillations) adapts to the new fair distribution of bandwidth. The conservative method, waiting one FRTT between each rate adjustment, uses more time to adjust to the new load.

At time 1.5 s, the traffic from station 1 stops. For both methods we see that some traffic from station 2 that has been queued is now being released; hence, there are an added number of packets received from station 2 at station 4. The aggressive method has some additional oscillations, but otherwise adjusts quickly to the new traffic pattern. The conservative method adjusts with fewer oscillations, but more slowly.

Topology Discovery

Topology discovery determines connectivity and the ordering of the stations around the ring. This is accomplished by collecting information about the stations and interconnecting links via the topology discovery protocol. The collected information is stored in the topology databases of each station.

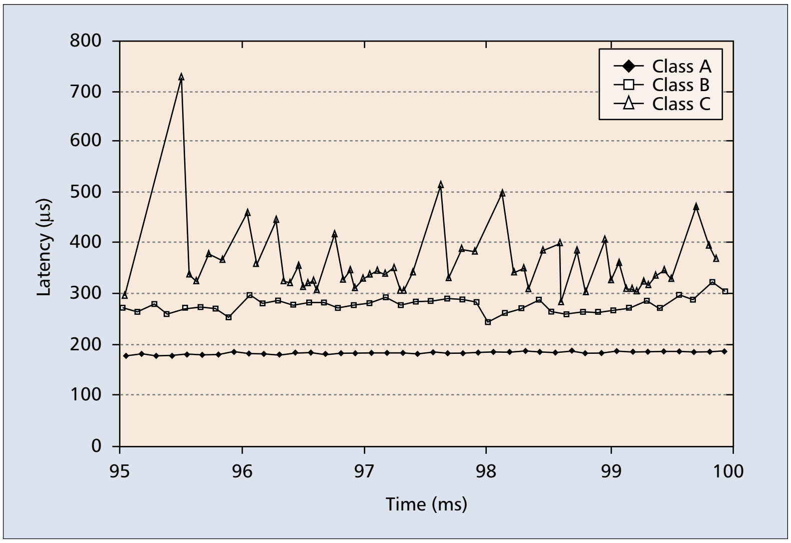


Figure 3. *Frame latency from station 1 to station 7 on a 16-station overloaded ring. The propagation and minimum frame latency is 180 μs.*

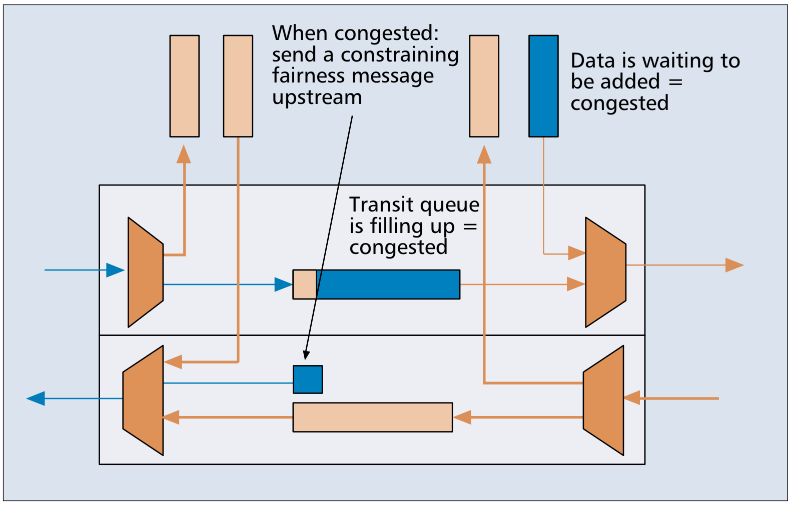

When congested: send a constraining fairness message upstream
Data is waiting to be added = congested
Transit queue is filling up = congested

Figure 4. *When a station becomes congested it sends a fairness message upstream.*

At system initialization, all stations send control frames, called topology discovery messages, containing their own status, around the ring. Topology messages are always sent all the way around the ring, on both ringlets, with an initial TTL equal to 255 (the maximum number of stations). All other stations on the ring receive these frames, and since the TTL is decremented by one for each station passed, all stations will be able to compute a complete topology image.

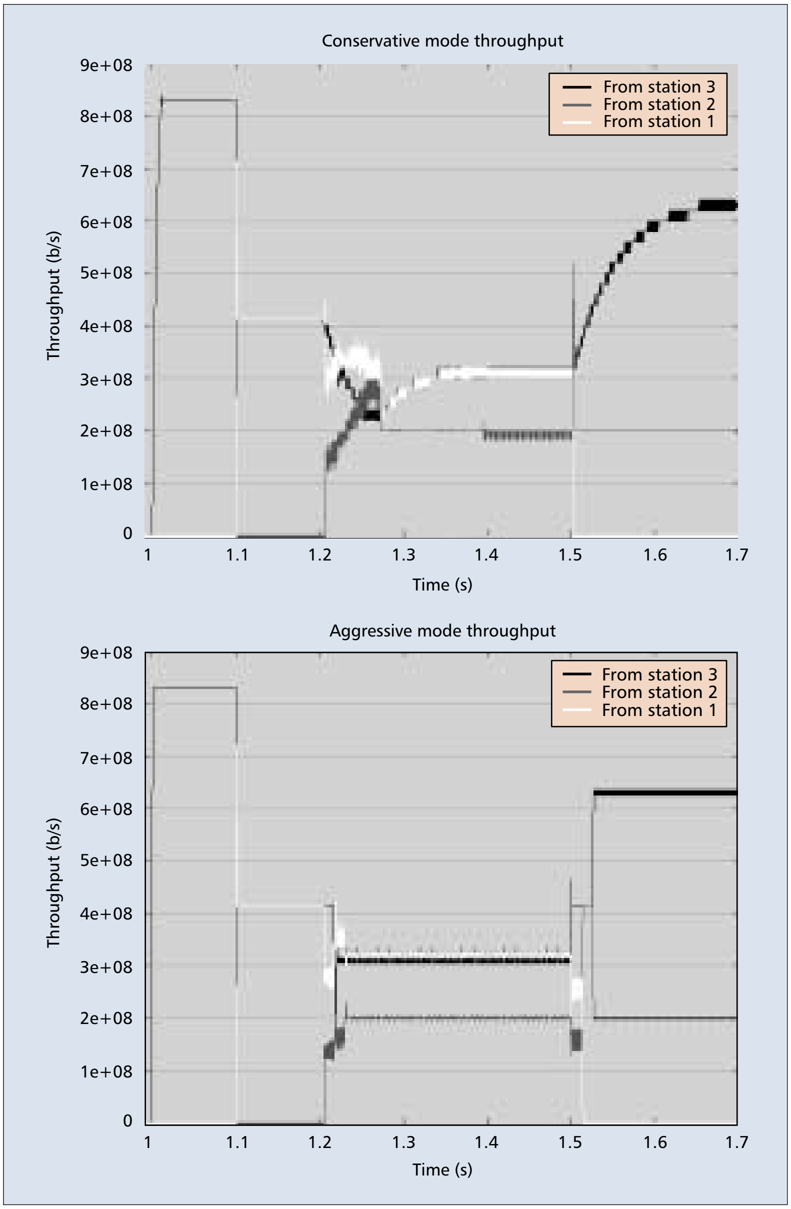
When a new station is inserted into a ring, or a station detects a link failure, it will immediately transmit a topology discovery message. If any station receives a topology message inconsistent with its current topology image, it will also immediately transmit a new topology message (always containing only the station's own status). Thus, the first station that notices a change starts a ripple effect, resulting in all stations transmitting their updated status information and rebuilding their topology images.

The topology database includes not only the ordering of the stations around the ring and the protection status of the stations (describing its connected links, with status signal fail, signal degrade, or idle), but also the attributes of the stations and the RTTs to all the other stations on the ring.

Once the topology information has become stable, meaning the topology image does not change during a specified time period, a consistency check will be performed. For example, the station will make sure that the information collected on one ringlet matches that on the other.

Even under stable and consistent conditions, stations will continue to periodically transmit topology discovery messages in order to provide robustness for the operation of the ring.

When the client submits a frame to the MAC without specifying which ringlet to use, the MAC uses the topology database to find the shortest path. Information in the topology database is also used when calculating the FRTT in the conservative mode of the fairness algorithm.



■ **Figure 5.** *Dynamic traffic handled by the conservative and aggressive fairness algorithms (number of bits per second as received by station 4).*

Resilience

As described in the previous section, as soon as a station recognizes that one of its links or a neighbor station has failed, it sends out topology messages. When a station receives such a message indicating that the ring is broken, it starts to send frames in the only viable direction to the receiver. This behavior, which is mandatory in RPR, is called *steering*.

The IEEE 802 family of networks have a default packet mode, called *strict* in RPR. This means that packets should arrive in the same order as they are sent. To achieve in-order delivery of frames following a link or station failure, all stations stop adding packets and discard all transit frames until their new topology image is stable and consistent. Only then will stations start to steer packets onto the ring.

The time it takes for this algorithm to converge — that is from when the failure is observed by one station until all stations have a stable and consistent topology databases and can steer new frames — is the restoration time of the ring. The RPR standard mandates a restoration time below 50 ms. To accomplish this goal, several design decisions must be considered, including ring circumference, number of stations, and speed of execution inside each station.

RPR optionally defines a packet mode called *relaxed*, meaning it is tolerable that these packets arrive out of order. Such packets may be steered immediately after a failure has been detected and before the database is consistent. Relaxed frames will not be discarded from the transit queues either.

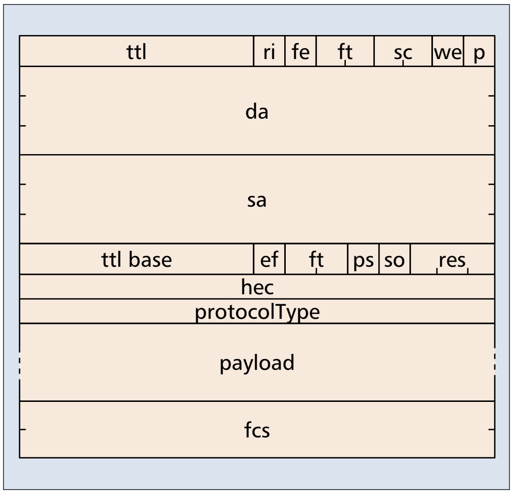
When a station detects that a link or its adjacent neighbor has failed, the station may optionally wrap the ring at the breakpoint (called *wrapping*) and immediately send frames back in the other direction (on the other ringlet) instead of discarding them. Frames not marked eligible for wrapping are always discarded at a wrap point.

Bridging

RPR supports bridging to other network protocols in the IEEE 802 family, and any station on an RPR ring may implement bridge functionality. Transporting Ethernet frames over RPR can provide resilience and class of service support.

RPR uses 48-bit source and destination MAC addresses in the same format as Ethernet (see the next section). When an Ethernet frame is bridged into an RPR ring, the bridge inserts RPR related fields into the Ethernet frame. Similarly, these fields will be removed if the frame moves from RPR (back) to Ethernet. An extended frame format is also defined in the standard for transport of Ethernet frames. In this format an RPR header encapsulates Ethernet frames.

When participating in the spanning tree protocol, RPR is viewed as one broadcast enabled subnet exactly like any other broadcast LAN. The ring structure is then not visible, and incurs no problem for the spanning tree protocol. The spanning tree protocol may not break the ring, but may disable one or more bridges connected to the ring.



■ **Figure 6.** *RPR basic data frame format.*

RPR implements broadcast by sending the frame all around the ring or sending the frame halfway on both ringlets. In the latter case the TTL field is initially set to a value so that it becomes zero and the packet is removed when it has traveled half the ring. Using broadcast, obviously no spatial reuse is achieved.

Since RPR can bridge to any other Ethernet, for example, Ethernet in the first mile (EFM), we envision Ethernets spanning all the way from the customer into the MAN or even the WAN. Whether such large and long-ranging Ethernets will be feasible or practical in the future is to be seen.

Another way to connect RPR to other data networks is to implement IP or layer 3 routers on top of the MAC clients. In this way RPR behaves exactly like any other Ethernet connected to one or more IP routers. Such IP routers should in the future also take advantage of the class-based packet priority scheme defined by RPR when they send quality of service constrained traffic over RPR.

Frame Formats

Data, fairness, control, and idle frames are the four different frame formats defined in the RPR standard. The following subsections introduce the important fields of these frames.

Data Frames

Data frames have two formats, basic and extended. Extended frame format is aimed at transparent bridging applications allowing easy egress processing and ingress encapsulation of other MAC frames. Using extended frame format also enables RPRs to eliminate out-of-order and duplicate bridged packets. The extended frame format is not described in this article. The basic data frame format is shown in Fig. 6.

Following is a short summary of RPR basic data frame fields:

ttl: The two-byte time to live field.
ri: The ring identifier bit defines into which ringlet the packet was initially inserted.
fe: The fairness eligible bit indicates that the packet has to abide by the rules of the fairness algorithm.
ft: The two-bit frame type: data, fairness, control, idle.
sc: The two-bit service class: A0, A1, B, C.
we: The wrap eligible bit indicates if the frame can be wrapped at a wrap node.
p: The parity bit is reserved for future use in data frames.
da: The six-byte destination address.
sa: The six-byte source address.
ttl base: This field is set to the initial value of the ttl field when the packet was first sourced into the ring. It is used for fast calculation of the number of hops a packet has traveled.
ef: The extended frame bit, indicating an extended frame format.
fi: The two-bit flooding indication is set on one or both ringlets when a frame is flooded.
ps: The passed source bit is set when passing its sender on the opposing ring after a wrap. The bit is used in detecting an error condition where a packet should have been stripped earlier.
so: The strict order bit, if set, identifies that the frame should be delivered to its destination in strict order.
res: A three-bit reserved field.
hec: The two-byte header error correction field protects the initial 16 bytes of the header.

Fairness Frames

The 16-byte fairness frame mainly provides the advertised fairRate and the source of the fairness frame. The information is used in the RPR fairness algorithm.

Control Frames

A control frame is similar to the data frame, but is distinguished by a designated ft field value, and its controlType field specifies the type of information carried. There are different types of control frames in RPR, such as topology and protection information, and operations, administration, and maintenance (OAM).

Idle Frames

Idles frames are utilized in order to compensate for rate mismatches between neighboring stations.

Conclusion

This article has discussed and explained the RPR architecture. It has shown how RPR has taken features from earlier ring-based protocols, and combined them into a novel and coherent architecture. Important parts covered in this article include the class-based priority scheme, station design, fairness, and resilience. Performance evaluations using the latest version of the draft standard demonstrate how the protocol behaves using different options. In particular, we have demonstrated how the aggressive fairness method acts very quickly in trying to adapt to a change in traffic load, while the conservative method has a more dampened response under varying load.

RPR is a new MAC-layer technology that may span into MANs and WANs. RPR can easily bridge to Ethernet, including access networks

Since RPR can bridge to any other Ethernet, for example Ethernet in the First Mile (EFM), we can envision Ethernets spanning all the way from the customer into the Metropolitan or even Wide Area Network. Whether such large and long-ranging Ethernets will be feasible or practical in the future, is to be seen.

RPR may differentiate traffic, so when used to implement IP links, it is able to help the IP routers implement the QoS-aware communication that is needed in a network that carries multimedia traffic.

like EFM. This makes it possible to perform layer 2 switching far into the backbone network, if such large link layer networks turn out to be practical. RPR may also do switching in the backbone network, by letting RPR implement virtual point-to-point links between the routers connected to the stations on the ring.

RPR may differentiate traffic, so when used to implement IP links it is able to help the IP routers implement the quality-of-service-aware communication needed in a network that carries multimedia traffic.

REFERENCES

[1] IEEE Draft P802.17, v. 3.0, "Resilient Packet Ring," Nov. 2003.

[2] R. M. Needham and A. J. Herbert, *The Cambridge Distributed Computing System*, Addison-Wesley, 1982.

[3] I. Cidon and Y. Ofek, "MetaRing — A Full-Duplex Ring with Fairness and Spatial Reuse," *IEEE Trans. Commun.*, vol. 41, no. 1, Jan. 1993.

[4] IEEE Std. 802.5-1989, "IEEE Standard for Token Ring."

[5] F. E. Ross, "Overview of FDDI: The Fiber Distributed Data Interface," *IEEE JSAC*, vol. 7, no. 7, Sept. 1989.

[6] ISO/IECJTC1SC6 N7873, "Specification of the ATMR Protocol (v. 2.0)," Jan. 1993.

[7] W. W. Lemppenau, H. R. van As, and H. R. Schindler, "Prototyping a 2.4 Gb/s CRMA-II Dual-Ring ATM LAN and MAN," *Proc. 6th IEEE Wksp. Local and Metro. Area Net.*, 1993.

[8] OPNET Modeler. http://www.opnet.com

[9] E. R. Hafner, Z. Nendal, and M. Tschanz, "A Digital Loop Communication System," *IEEE Trans. Commun.*, vol. 22, no. 6, June 1974.

[10] C. C. Reames and M. T. Liu, "A Loop Network for Simultaneous Transmission of Variable-length Messages," *ACM SIGARCH Comp. Architecture News*, *Proc. 2nd Annual Symp. Comp. Architecture*, vol. 3, no. 4, Dec. 1974.

[11] IEEE Std. 1596-1992, "IEEE Standard for a Scalable Coherent Interface (SCI)."

[12] D. Tsiang and G. Suwala, "The Cisco SRP MAC Layer Protocol," IETF Networking Group, RFC 2892, Aug. 2000.

BIOGRAPHIES

FREDRIK DAVIK (bjornfd@simula.no) received his B.S. from South Dakota School of Mines and Technology in 1996. He is currently working on simulation and performance studies of RPR at Simula Research Laboratory and Ericsson Research Norway toward completion of his Ph.D. at the University of Oslo.

METE YILMAZ (myilmaz@cisco.com) received his B.S. (1995) and M.S. (1999) degrees from Bogazici University, Istanbul, Turkey, in electrical engineering and computer engineering, respectively. He is currently pursuing his Ph.D. degree at New Jersey Institute of Technology, Newark, in computer engineering and is employed at Cisco Systems, San Jose, California. He has been working on communication networks for nine years. His research interests include quality of service, bandwidth reservation, and fairness algorithms.

STEIN GJESSING is a professor of computer science at Simula Research Laboratory and Department of Informatics, University of Oslo, Norway. He received his Ph.D. degree in 1985, and was head of the Department of Informatics, University of Oslo, from 1987 to 1991. From 1990 he participated in the development of IEEE 1596, Scalable Coherent Interface, and since January 2001 he has been a member of the RPR working group. His current research interests are within computer communications in general and quality of service related architectures and protocols in particular. From October 2003 to April 2004 he is a research scholar at the Computer Engineering Department, San Jose State University.

NECDET UZUN has over 15 years of experience in VLSI architecture design for high-speed communication networks. He has been involved in designing IP routers and switches, traffic shapers and schedulers, and fast IP lookup and classification architectures. He is currently with Cisco Systems. He was a founder of AuroraNetics, Inc. He worked as a consultant at Bellcore, as an industry professor at Polytechnic University and as an assistant professor at New Jersey Institute of Technology. He received B.S. and M.S. degrees from Istanbul Technical University, Turkey, and a Ph.D. degree from Polytechnic University, Brooklyn, New York, all in electrical engineering.