Overview of the MetaRing Architecture *

Yoram Ofek
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598
e-mail: ofek at watson.ibm.com

ABSTRACT

The basic *MetaRing* architecture is a full-duplex ring providing fairness and spatial bandwidth reuse. Concurrent access and spatial bandwidth reuse enable simultaneous transmission over disjoint segments of the bidirectional ring. It therefore increases the potential throughput in each direction, by a factor of four or more. In this work, we overview the MetaRing principles:
(1) Distributed global fairness algorithm, a simple and robust mechanism based on a single control signal (i.e., one bit of information) that regulates the access to the ring.
(2) Protocol for service integration of: (i) synchronous or real-time traffic which is periodic and requires a connection set-up and which will have guaranteed bandwidth as well as bounded delay, and (ii) connectionless or asynchronous traffic with no real-time constraints that can use the remainder of the bandwidth. Integration is an important function for multi-media applications.
(3) Protocol and requirements for multi-ring and dual-bus MetaRing networks.
(4) Principles and requirements for interconnecting MetaRing with wide-area networks (WANs). We show that (i) the WAN-to-ring interconnection requires a separate queue for asynchronous traffic and relies on the use of the fairness mechanism for **internal flow control**, whereas (ii) the WAN-to-dual-bus configuration of the MetaRing network is simpler, since it does not require any buffering and does not rely on a fairness mechanism for **internal flow control**, furthermore; it is fault tolerant and has better synchronous traffic performance.

*Published in: *Computer Networks and ISDN Systems* Volume: 26, Number: 6-8, Pages: 817-830, March 1994.

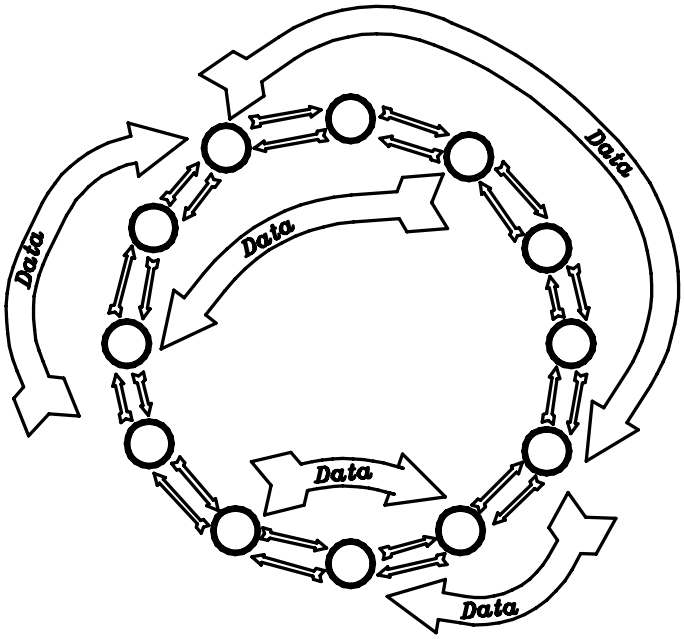

Figure 1: Concurrent Transmissions on a Full-duplex Ring with Spatial Reuse

1 Introduction

The main motivation for developing the *MetaRing* [12, 13, 27, 26, 10, 11] architecture is to increase the throughput of a ring-based local area network (e.g., [4, 31]) beyond its single link capacity by means of spatial bandwidth reuse. Spatial bandwidth reuse enables concurrent access in each direction of the ring by more than one node, as shown in Figure 1. However, uncontrolled access may cause starvation. This can happen if some nodes are constantly being "covered" by up-stream ring traffic, and thus, are not able to access the ring. Therefore, an efficient fairness control mechanism is critical.

To quantify this, assume that the network has n nodes and is under full load (i.e., at all times all nodes have packets or cells to send). Under an uniform destination distribution, the maximum distance for a packet to travel is $n/2$ hops, and the average distance is $n/4$ hops. Therefore, the spatial bandwidth reuse factor for one direction is four, i.e., on the average four nodes are able to transmit at the same time. As a result, the capacity of the full-duplex buffer insertion ring is eight times that of a single link, which is four times more than a dual token-ring. If the destination distribution is inversely proportional to the distance, then the average distance is $n/6$ hops (this means the spatial bandwidth reuse factor is six for each direction).

Fairness mechanisms for slotted rings with spatial bandwidth reuse were introduced in MAGNET [23] (Columbia University), Orwell [17] (British Telecom) and ATMR [30] (NTT). The fairness algorithms of these architectures operate using network-wide fairness cycles which may result in an idle time between successive fairness cycles. This idle time is sensitive to the ring propagation delays. More recently, there were two buffer insertion ring proposals from the IBM Zurich Research Laboratory: CRMA-II [36, 35] and BCMA [20], and a slotted ring proposal, D3Q, from ASCOM [2].

The fairness mechanism presented in the MetaRing network operates continuously and follows the natural ordering along the ring [13, 27]. Therefore, it is less sensitive to the ring propagation delay than MAGNET, Orwell, BCMA and ATMR, and it is also more versatile, since it can be used for multi-ring flow control, traffic integration, and flow control between a ring and wide area networks. The MetaRing fairness algorithm requires only a single bit of information, and therefore, if ATM cells are transmitted, this algorithm can be implemented by using only one of the four GFC (generic flow control) bits in the ATM cell header. Note that the four GFC bits are not sufficient for implementing the Orwell and ATMR fairness algorithms since they require the use of a unique node ID, which also makes their implementation more complex.

Since the buffer insertion (or slotted) ring access is always permitted, unless there is ring traffic, there can be no degradation in its efficiency as the bandwidth or physical size increases. All links can be kept at full utilization, at all times, provided that the nodes have enough data to transmit.

For multi-media purposes, we show how to integrate two basic classes of traffic services: (i) *synchronous* or real-time traffic that requires bounded delay and guaranteed bandwidth, and (ii) *asynchronous* traffic with no real-time constraints but with fairness requirements. This integration mechanism is functionally equivalent to the TIMED-TOKEN protocol in FDDI [18, 3, 31]. This protocol together with the asynchronous fairness still maintains round-robin fairness with spatial bandwidth reuse for the asynchronous traffic. The integration protocol has the important property that unused reserved capacity for synchronous traffic can be used by the asynchronous traffic.

The *MetaRing* network was prototyped at the IBM T. J. Watson Research Center in 1989. This prototype supports the transmission of variable size packets at 100 Mb/s link speed with an aggregate throughput of 700 Mb/s. A Gigabit version of the MetaRing is currently being implemented as part of the IBM participation in the Aurora testbed which is part of the NSF/DARPA Gigabit Networking Program [1].

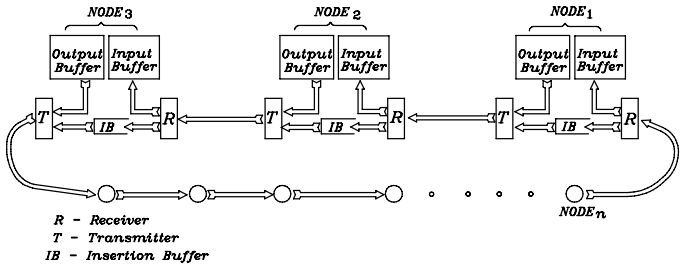

Figure 2: Buffer Insertion Ring (one direction)

In Section 2, the basic MetaRing principles are reviewed. Based on these principles we present in Section 3 various possible network configurations. The conclusions are given in Section 4.

2 Basic Principles of Operation

In this section, we describe the basic principles of the MetaRing: (i) access control, (ii) hardware control signals, (iii) fairness of asynchronous data traffic, and (iv) integration of synchronous and asynchronous traffic.

2.1 Access Control with Spatial Bandwidth Reuse

The MetaRing can operate under two basic access control modes: buffer insertion for variable size packets or slotted for fixed size cells [19, 21, 24]. In both modes, the packets or cells are removed by their destinations to provide spatial bandwidth reuse.

Buffer insertion is a random and distributed access technique. On the receiving side of each link, there is an insertion buffer (IB) which can store at least one maximal size packet, as shown in Figure 2. A node may start to transmit a packet at any time as long as its insertion buffer is empty. If ring traffic arrives when the node is in the middle of a packet transmission, then this traffic will be delayed in the insertion buffer until the packet transmission is completed. The node cannot transmit another packet until the insertion buffer becomes idle again. Thus, non-preemptive priority is given to the ring traffic. If the insertion buffer of a node is idle, the ring traffic is **cut-through** the insertion buffer. This means that a packet does not have to be completely received before it is forwarded [22].

When operated in the slotted mode, each slot starts with a **busy-bit**. If this bit is 0, the slot is empty, and if it is 1, the slot is full. A node can transmit a cell only if it receives an

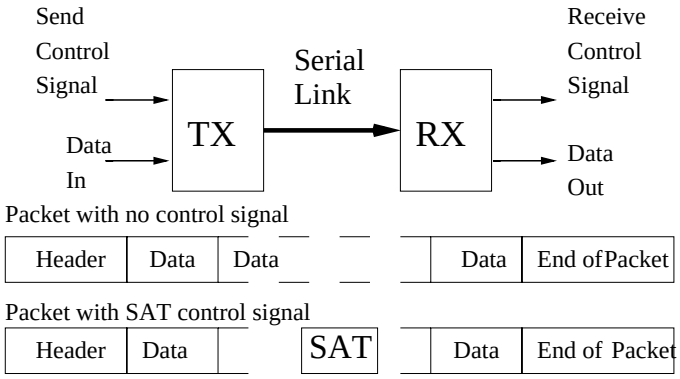

Figure 3: The Preemptive/Nondestructive Control Signal Mechanism

empty slot. The cell is removed by the destination node, and then the slot becomes empty.

The motivation for a slotted mode is to minimize the delay between source and destination. The "price" of this change is that the cell size must be fixed, i.e., in every slot we put a single cell. The network hardware interface and access control algorithms for the buffer insertion and slotted modes are basically the same. The main difference between the two modes is that the receiving host interface, in the slotted mode, should reassemble the variable size packets from the fixed size cells, which can be a complex function.

2.2 Hardware Control Signals

The hardware control signals are used to implement time critical control functions that must operate fast. These signals use the same physical medium as the data, and can be used to improve fairness, to enable traffic integration and to prevent insertion buffer overflow. The following two characteristics ensure a small delay for the control signals: (i) short - only a few characters (possibly one), and (ii) preemptive resume priority - i.e., it can be sent in the middle of a data packet without damaging the data packet which it preempts, as illustrated in Figure 3.

Each control signal can be followed by a predefined number of parameters. The different control signals form different control channels over the transmission links. Thus, over the full-duplex ring one data channel in each direction and one or more control channels in each direction are virtually constructed. Each control channel is associated with one data channel. There are two cases: (i) a control channel associated with a data channel in the opposite direction. In this case, the data is sent down-stream and the corresponding control signals are sent up-stream, and (ii) a control channel associated with a data channel in the same

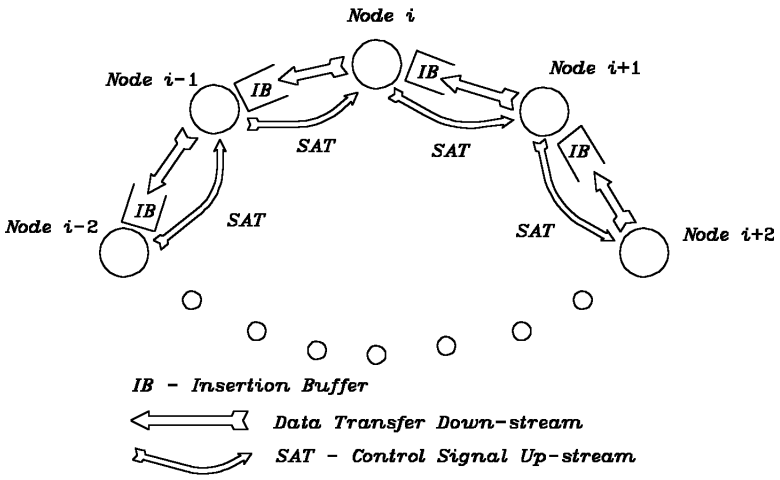

Figure 4: The Basic Mechanism (one direction) on Full-duplex Ring

direction.

2.3 Global Fairness on a Full-duplex Ring

Global fairness algorithms view each direction of the ring as a single shared communication resource. The objective of such an algorithm is to ensure that all nodes have equal opportunity to access the network. In contrast, local fairness algorithms view each link as a communication resource and the whole ring as a multiplicity of communication resources. The local fairness is somewhat more complex, but provides better throughput under non-symmetric traffic patterns. Local fairness algorithms for the MetaRing are described in [7, 6, 9], another local fairness algorithms was introduced in [33].

To achieve global fairness, the access to each direction of the ring is regulated by a hardware control signal, called SAT (comes from the word SATisfied), which circulates in the opposite direction to the data traffic it regulates [13, 12] (see Figure 4). Circulating the SAT control signal in the opposite direction enables better exploitation of the potential spatial bandwidth reuse of the full-duplex ring.

In principle, the node forwards the SAT signal up-stream without any delay, unless it is not SATisfied or "starved." By "starved" we mean that the node could not send the permitted number of data units (cells or bytes) since the last time it had forwarded the SAT signal. More specific, the node is SATisfied if between two visits of the SAT signal the node has sent at least l data units or if its output buffer is empty. If the node is not SATisfied, it will hold the SAT until it is SATisfied and then forward the SAT up-stream. After a node forwards a SAT,

it can send up to k more data units, before receiving and forwarding again the SAT signal, $k \geq l$ (in a symmetric case $k = l$).

The following is the description of the global fairness algorithm, which uses a single variable COUNT - to count the number of data units that has been transmitted. The algorithm has two parts: send packet and forward SAT.

Send Packet Algorithm:

The node can transmit a packet from its output buffer when it is not empty, only if the following two conditions hold:
(i) the variable COUNT is smaller than k, and
(ii) the insertion buffer is empty.
After the node has transmitted the packet, COUNT is incremented by the transmitted amount of data units.

Forward SAT Algorithm:

After receiving the SAT signal, the node will forward the SAT if either:
(i) its variable COUNT is equal to or greater than l, or
(ii) its output buffer is empty.
The node will hold the SAT if its variable COUNT is smaller than l and the output buffer is not empty.
The node will hold the SAT until COUNT becomes l (after l data units have been transmitted).
If during the time in which the node holds the SAT, another SAT arrives, the second SAT will be discarded.
After the node forwards the SAT, it will set the COUNT to zero.

In the slotted mode the global fairness algorithm is implemented by designating a single bit, at the beginning of each slot, as a *SAT-bit*. When a node wants to send a SAT signal to a neighboring node it will set the *SAT-bit* to 1, and otherwise to 0. If we are transmitting ATM cells, in either buffer insertion mode or slotted mode, the SAT signal can be implemented by using one of the four GFC (generic flow control) bits in the ATM cell header. Then when a node wants to send a SAT signal to a neighboring node it will set this bit to 1, and otherwise to 0.

There are several possible variations on this global fairness algorithm, for details see [13]. Performance studies of the MetaRing with different parameters can be found in [5, 13, 28], which demonstrate the high performance and high efficiency of the MetaRing architecture. An algorithm for ensuring that there is only a single SAT signal in each direction of a dual-ring,

is described in [25], which is a self-stabilize mechanism.

2.4 Synchronous and Asynchronous Integration

This section describes a mechanism for the **fair integration** of two types of traffic ([28, 37]): (i) synchronous or real-time traffic that requires a connection or reservation set-up and that will be guaranteed a given bandwidth and bounded delay, and (ii) asynchronous traffic with no real-time constraints that can use the remainder of the bandwidth.

The simplest integration method is to let the SAT signal held by those nodes that have outstanding synchronous traffic. This method is suitable only for relatively small rings. The following mechanism is more robust. It is based on four control signals SAT, ASYNC-EN(GR), ASYNC-EN(YL) and ASYNC-EN(RD). The ASYNC-EN signals (Asynchronous Enable - Green, Yellow, Red) are used for enabling and disabling the integration of the asynchronous traffic. The SAT is used for ensuring global fairness of the asynchronous traffic, as it was described in the previous section [1].

The integration is achieved by the following principles:

1. The synchronous traffic is reserved by a call set-up protocol.
2. Each node has two queues: one for synchronous and one for asynchronous traffic. All the reserved traffic is buffered in the synchronous queue.
3. For accessing the ring, traffic in the synchronous queue always has priority over traffic in the asynchronous queue.
4. Unused reserved capacity can be used for asynchronous traffic. This ensures high utilization even if the reservation is made on the basis of peak-rate.
5. The node can transmit traffic from the synchronous queue whenever the ring is idle (insertion buffer empty or empty slot arrives), regardless of its asynchronous queue state. For example, a node that holds the SAT signal, because it is not satisfied, will first send traffic from the synchronous queue and only then send its asynchronous quota and release the SAT.

[1] A different integration method is described in [13, 12], that method is using only two control signals: SAT and ASYNC-EN.

This last principle is very important. It basically states that the reserved traffic is transmitted even if there is no SAT signal in the system. Thus in the case of a SAT failure, the access of the reserved traffic will not be stopped (only the non-reserved asynchronous traffic is stopped during the recovery procedure.)

2.4.1 Distributed Reservation and Synchronous Access

The distributed reservation is the mechanism which guarantees bandwidth for transferring synchronous traffic over the ring. For the reservation or connection set-up mechanism, we assume the following notations:

1. T_c - is the periodic time cycle of synchronous data transfers (in seconds).
2. BW - the data transmission rate (in bits per second).
3. p - the basic data units (in bits); in the slotted mode this is the slot length in bits.
4. c - is the number of data units that can be transmitted over each transmission link in every time cycle, where $c = \frac{T_c BW}{p}$.
5. ρ - the maximum fraction of synchronous traffic ($0 \leq \rho < 1$).

When a node tries to reserve bandwidth for real-time transmission, it performs the following protocol.

The set-up protocol performed by a source node:

1. Determine how many data units are needed in one time cycle, say u.
2. Determine the transmission direction, which determines the reservation path.
3. Send reservation requests for u data units to all nodes along this reservation path.
4. If positive acknowledgements are received by the source from all the nodes along the reservation path, then this connection becomes effective. Else the source node sends a release request of u data units to all nodes along this reservation path.

Each node maintains a variable RESERVE, which indicates how many data units have been reserved. The RESERVE variable should be less than ρc, therefore:

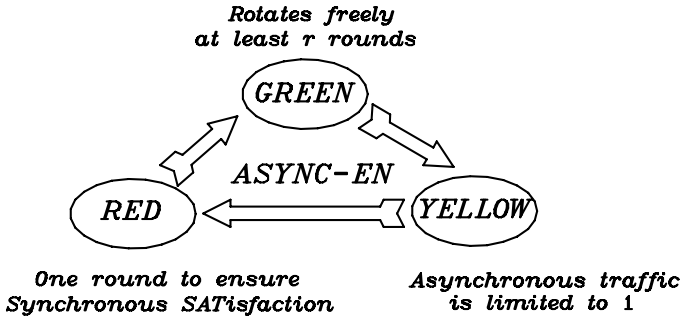

Figure 5: The Integration Signalling

- When a node receives a reservation request for u data units and if RESERVE+$u < \rho c$, then RESERVE=RESERVE+l and a positive acknowledgement is returned to the source node, else RESERVE=RESERVE+u and a negative acknowledgement is returned.
- When a node receives a release request for u data units then: RESERVE=RESERVE-u.

After the set-up is completed successfully, the reserved traffic is transmitted before asynchronous traffic. The reserved traffic will be queued only if the link is busy (the synchronous traffic is buffered in the SYNC-QUEUE).

2.4.2 Integration Protocol

The integration protocol uses the ASYNC-EN control signal, which has three different attributes: GREEN (GR), YELLOW (YL) and RED (RD), see Figure 5. The basic principle of the integration protocol is to periodically halt the asynchronous traffic, if necessary. The three attributes constitute three control signals with the following relationships (as shown in Figure 5):

1. ASYNC-EN(GR): the ASYNC-EN(GR) control signal is used for realizing a distributed timer on each ring interface (each direction has a separate identical mechanism), and for enabling and disabling the asynchronous traffic. Under normal condition, the ASYNC-EN(GR) rotates around the ring freely, i.e., each node will forward the ASYNC-EN(GR) immediately after receiving it. As a result, the rotation time of this signal is about the propagation delay around the ring, T_{RING}. We define a parameter T_{min} which is equal to the time for r free rotations of the ASYNC-EN(GR) around the ring ($T_{min} = rT_{RING}$, $r \geq 0$).

2. ASYNC-EN(YL): after the ASYNC-EN(GR) has completed at least r rounds a node that has a back-log of real-time traffic, can change the control signal attribute from GREEN to YELLOW. When nodes see the ASYNC-EN(YL) signal they cannot start to transmit new asynchronous packets into the ring. The YELLOW signal is transferred unconditionally until it reaches its origin node which then changes its attribute from YELLOW to RED.

3. ASYNC-EN(RD): the RED signal is transferred once around the ring. A node forwards the ASYNC-EN(RD) signal to its up-stream neighbor if it has no back-log of real-time traffic, i.e., its real-time traffic is satisfied, otherwise it holds the ASYNC-EN(RD) signal until it has no back-log of real-time traffic. When the RED signal returns to its origin node it will change its attribute back to GREEN. The GREEN signal should complete at least r rounds ($r \geq 0$) before the cycle, in Figure 5, can start again.

The synchronous back-log condition:

Synchronous traffic in a node is considered to be back-logged if it has been waiting in the synchronous transmission queue for more than a predefined time threshold. This time threshold is measured in terms of round trip delays on the ring, T_{RING}.

In [28, 37] we present and discuss in details the performance characteristics of this protocol, which demonstrates the effectiveness and high efficiency of this integration protocol.

3 Network Configurations

In this section several possible network configurations are described that can be constructed based on the MetaRing principles. First it is shown how the full-duplex ring can gracefully degrade to a multiple of bus segments, then it is shown how multiple rings can be connected together. Finally it is explained how the MetaRing can be connected to a wide area network.

3.1 Dual-bus Segments

When the full-duplex ring suffers link or node failures, the ring becomes disconnected, since we assume that even if only one direction of a full-duplex link is faulty, the other direction is declared faulty as well. Thus, if one or more failures occur, the full-duplex ring is transformed to a network consisting of one or more dual-bus segments.

The SAT-SAT' Mechanism

On a full-duplex bus, the SAT signal cannot go around in cycles. Therefore, when a SAT signal arrives at an edge node of the dual-bus, it will be sent back as a different control signal SAT' (in the opposite direction). When a corresponding edge node receives a SAT' it will send a regular SAT control signal in the opposite direction. The SAT' cannot be held by a node. Thus it is forwarded on the bus without nodes delaying it, so it will reach the other side of the bus in a time corresponding to the propagation delay.

The SAT-SAT' mechanism forms a virtual ring on the dual-bus segment, so that the fairness algorithm can continue to operate correctly with the SAT signal, as previously described in Section 2.3. The SAT-SAT' mechanism is performed dynamically, i.e., the network configuration can change from a ring to multiple bus segments and back during normal operation, whenever a link or a node fails or recovers. As a result of bus segments changing back to form a ring configuration, it can occur that one or more SAT' signals will rotate in the ring. Since there is no edge node to convert these SAT' signals back to regular SAT signals, the infinite rotation of these SAT' signals must be prevented.

Prevention of Infinite Looping of SAT'

In order to prevent infinite rotation of SAT', each node will have to detect this abnormal phenomenon. A node can detect this when it sees two successive SAT' signals with no SAT signal in between. In this case, the SAT' signal is eliminated.

This happens since, if there is only one SAT signal in one direction and one or more SAT' signals in the opposite direction, and the SAT' signals are rotating strictly faster than the SAT signal. (This is because the SAT' is transferred unconditionally without delay and the SAT is intermittently held by the fairness algorithm.) As a result, a node will encounter more SAT' visits than SAT visits, which means that there will be two SAT' visits with no SAT visit in between.

Another possible method for eliminating the SAT' signals is to detect when SAT' and SAT, in opposite directions, are crossing one another over a link or at some node. When it is detected the SAT' is eliminated. This method is called *phase crossing elimination* and a

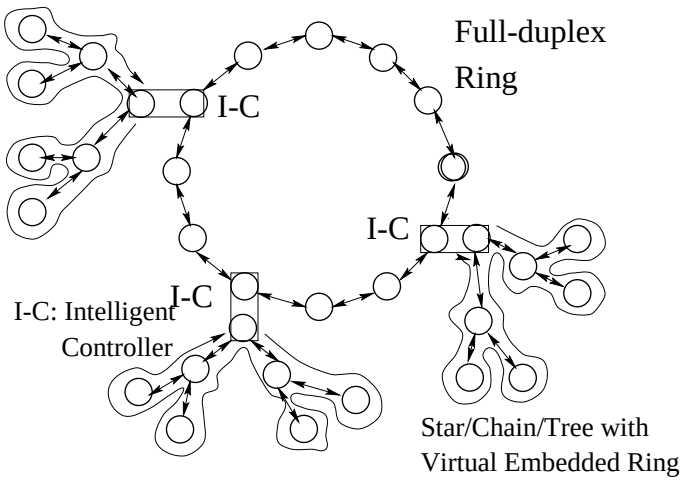

Figure 6: FDDI Look-alike Topology: Backbone with Virtual Rings

description on how it is implemented can be found in [29]. This method will operate correctly and effectively even if there are multiple SATs in the same direction, or even if the SAT and SAT' are rotating at the same speed. This method fails only if there is no SAT in the ring, which can be detected by a time-out mechanism.

3.2 Multi-ring Networks

The motivations for having a multi-ring structure are: (i) To match a logical and a physical network topology. For example, the FDDI physical structure is typically a backbone dual-ring with multiple trees or stars. (ii) To increase spatial bandwidth reuse and the aggregate throughput. Each ring in a multi-ring network deal with a cluster of users with a higher internal interaction than interaction with external users in other clusters. (iii) To enhance the fault tolerance of the system.

An example of a multi-ring structure is shown in Figure 6. It is based on a main full-duplex ring with additional secondary full-duplex rings and secondary unidirectional virtual rings that are embedded on trees or stars. In Figure 6, the secondary rings and trees are connected by switching nodes, which are called intelligent concentrators (I-C).

In the following, the asynchronous operation of the switching node is described. All other nodes operate as was previously described. In particular the fairness and internal flow control aspects of the system [26] are considered. In a related paper a new label-based source-routing for multi-ring networks [15, 16] is described.

3.2.1 Multiple Full-duplex Rings

It is assumed that each node on a ring follows the buffer insertion access and flow control principles which imply that traffic already on the ring has priority over the external traffic onto the ring. As a result, traffic that is transferred between rings may be queued or buffered at the switching or I-C nodes.

Each direction of a ring in the system is either clockwise (CW-ring) or counter-clockwise (CCW-ring). The route of a packet from source to destination travelling via multiple rings is either via only CW-rings or only CCW-rings. Thus, in multiple full-duplex ring configurations, there are always two possible routes between a source and destination: CW-route and CCW-route. It is assumed that the source node selects the route that is the shortest on the main ring. Since the main ring is potentially the bottleneck of the system, the traffic load on it should be minimized.

Fairness and internal flow control on multiple rings

There are two identical fairness mechanisms in the system, one for the CW-routes and the other for the CCW-routes. In the following discussion, we describe the fairness of the CW-routes. The traffic on the CW-rings is regulated by a SAT that is transferred counter-clockwise. Each ring has its own SAT signal. In order to regulate the data transfer between rings the SATs on adjacent rings should be synchronized. The SATs of the CW-rings and the CCW-rings are synchronized independently.

The SAT synchronization is performed by the switching node using the following algorithm (see Figure 7).

SAT Merge and Fork Algorithm in a Switching node:

- *If a SAT on the main (secondary) ring is received wait until the following conditions are met:*
  1. *the SAT from the secondary (main) ring arrives;*
  2. *the secondary flow condition 1 is SATisfied;*
  3. *the main flow condition 2 is SATisfied;*
- *If all three conditions are met, fork the SAT signals into the main and the secondary rings;*

The flow conditions are defined for the traffic between the rings.

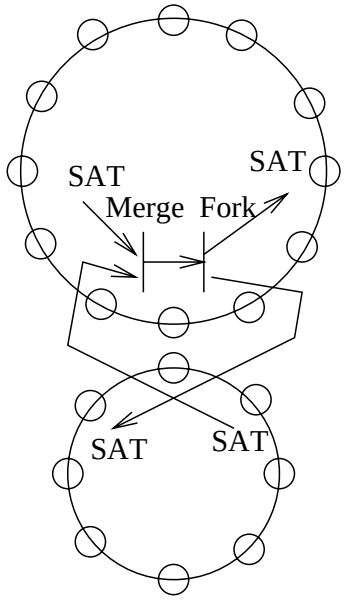

Figure 7: SAT Merge and Fork Synchronization

Secondary flow condition 1 - *the traffic from the secondary to the main ring is SATisfied, if all the cells/packets that were queued for the main ring when the previous SAT was forked, have been transmitted.*
Main flow condition 2 - *the traffic from the main to the secondary ring is SATisfied, if all the cells/packets that are currently queued for the secondary ring have been transmitted.*

3.2.2 Full-duplex with Secondary Trees

The inclusion of trees on the MetaRing architecture is desirable for two reasons: (i) the leaf nodes will have only one transmitter and one receiver (one full-duplex port) which will simplify their design, and (ii) adding remote nodes to the system is simplified, since it is possible to connect leaves directly to the switching node.

A unidirectional buffer-insertion virtual ring is embedded on the tree, as shown in Figure 6. This ring operates as described in Section 2. Note that this embedding is always possible, since the tree consists of full-duplex links.

The flow and fairness control between the tree and the main full-duplex ring can be performed in two ways. The simplest way is by defining virtual rings as all CW-rings or as all CCW-rings and applying the same algorithms as given in the previous section. In this method, the traffic to and among the trees will be done via one direction of the main ring, and not via

the shortest path.

The other alternative is that the virtual embedded ring will act both as a CW-ring and as a CCW-ring. The main problem in this case is that the instantaneous traffic rate into the virtual ring can be as much as twice its capacity. This will require more buffers in the switch (I-C) in order to avoid cell/packet loss. This alternative, although feasible, is not described in this paper and will be presented in a future work.

3.2.3 Switching Node Buffering Requirements

In this section, we analyze the buffering requirements on the switching node. it is shown that under arbitrary traffic pattern, it is possible to ensure that no cell/packet is lost as a result of buffer overflow. It is shown that the buffer size is bounded by the total transmission quota in the secondary ring.

We assume that a cell/packet is routed in this network via either all CW-rings or via all CCW-rings. In this discussion only the buffering requirements for the CW-routes are considered.

Let $q_{i,j}^{CW}$ be the transmission quota of node i on CW-ring j, and
$Q_j^{CW} = \sum_i q_{i,j}^{CW}$ be the total transmission quota on CW-ring j.

Buffering from secondary to main ring

The maximum amount of traffic that can flow into the switching node from the secondary ring between two successive forking of the SAT is $2Q_j^{CW}$. The factor of two comes from the summation of the current and previous quotas.

From the secondary flow condition 1 we see that the SAT is held at the switching node only for cells/packets that have been queued there at the last forking event of the SAT. As a result, in the worst case, the total buffering requirement is $3Q_j^{CW}$. Two quotas can be there when the current SAT is forked and the third quota can arrive during the current SAT rotation.

Buffering from main to secondary ring

In this case the worst scenario is when all the traffic on the secondary ring is local, i.e., nothing is switched into the main ring. From the main flow condition 2 it follows that when the SAT is forked the queue into the secondary ring is empty. The maximum amount of local secondary traffic that can cross the switching node is $2Q_j^{CW}$ (sum of previous and current quotas). Therefore, the maximum amount of traffic that can be blocked and queued from the main to the secondary ring is also $2Q_j^{CW}$.

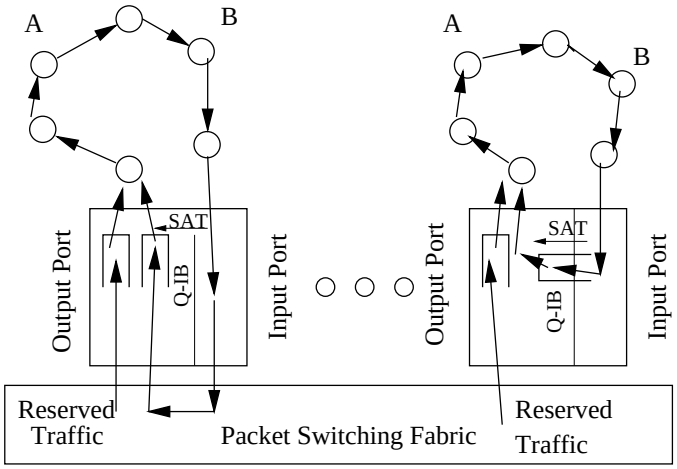

Figure 8: The Ring Loop is Closed via the Switch

3.3 Integration with Wide Area Networks

Interconnection with wide area networks (WANs) is an important configuration for current and future local and metropolitan area networks. In this section we examine possible configurations of the MetaRing with fast connection WANs, like ATM (B-ISDN). The problem is how to preserve, on one hand, the asynchronous LAN properties, and on the other hand, the synchronous connection (or bandwidth guaranteed) property of the fast connection networks. We assume that a similar synchronous connection set-up procedure, as in Section 2.4, is extended from the LAN to the WAN.

Two basic configurations are examined: (i) the LAN and WAN traffic are mixed via a switch, and (ii) the LAN is a dual-bus, and there is a clear separation between the LAN and WAN traffic.

The LAN loop is closed via a switch

The problem in this configuration is similar to the interconnection of multiple rings, as described in the previous section. Figures 8 and 9a, illustrate three possible methods to close the ring loop via a switch. One is via the high-speed switch, the second is via the full-duplex link adapter, and the third is via a three-way switch on a full-duplex ring (Figure 9a).

Since asynchronous traffic is unpredictable and bursty in "nature", the following are necessary requirements to prevent cell/packet loss:

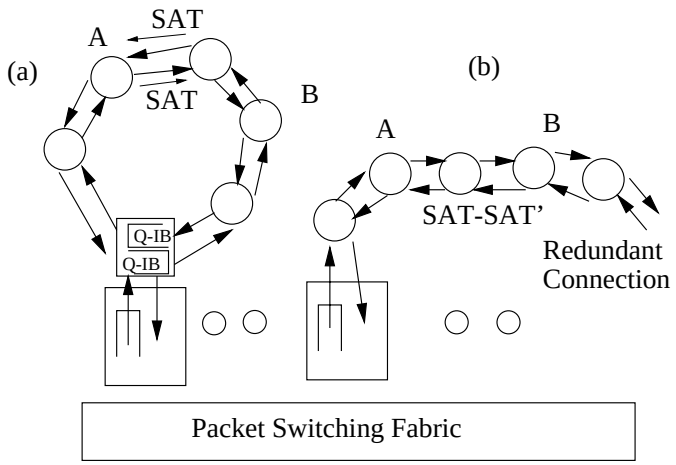

Figure 9: Dual-ring (a) and Dual-bus (b) Configurations

1. Priority to the synchronous connection traffic over the asynchronous LAN traffic.
2. Use of the SAT signal for **internal flow control** to integrate the LAN and WAN traffic, such that the switch controller is able to regulate the asynchronous LAN traffic.
3. Special buffer, **Q-IB** (in Figures 8 and 9a), for the LAN asynchronous traffic in order to preserve the no-loss property.

Switch internal flow condition: *the switch controller will hold the SAT signal if the* **Q-IB** *is not empty.*

It can be shown that the size of **Q-IB** should be twice the sum of the asynchronous quota of the ring's nodes: $S_{QI-B} = 2\sum_i q_i$. The factor of two comes from the summation of the current and previous quotas on the ring. Note that each direction of the dual-ring in Figure 9a is dealt independently. We also see that this kind of LAN-WAN interconnection is not possible without the SAT control signal.

Dual-bus to WAN connection

In Figure 9b, we show a configuration that preserves a clearer separation between the traffic on the dual-bus segment (Section 3.1) and the WAN traffic. In this case, the reserved synchronous traffic in the WAN has a "natural" up-stream priority over the LAN traffic, and no special buffering is required. Furthermore, the SAT is used only for fairness and not for internal flow-control between the LAN and WAN as it was the case in the previous configurations.

Redundant dual-bus to WAN configuration

The dual-bus configuration can be used for fault tolerant purposes. The other end of the bus can be connected to another switching node.

Both connections can be operational at the same time, which will make possible to use shortest path routing for the reserved synchronous traffic. In other words, in Figure 9b, if the distance from node B to A is shorter (number of hops) via WAN switches, then the connection will be set through the WAN. In this case, a source-routing [34, 8, 16] or label swapping [32, 14] will be used together with the LAN's self-routing method. Using the shortest path routing increases the potential spatial bandwidth reuse of this configuration.

4 Conclusions

In this paper a new LAN architecture, the MetaRing, is described. It utilizes the SAT control signal for asynchronous fairness. Furthermore, the following "value-added" functions can be realized by holding the SAT signal: (i) asynchronous and synchronous traffic integration, (ii) internal flow control for multi-ring interconnection, (iii) internal flow control for LAN-to-WAN interconnection and (iv) insertion buffer overflow prevention. In addition, the MetaRing has five basic routing modes [11]: (i) neighbor mode for initialization, (ii) point-to-point mode, (iii) broadcast mode, (iv) group multicast mode and (v) selective copy mode.

The MetaRing architecture unifies, in a simple manner, all the essential LAN properties:

- Immediate or random access under light load as in **Ethernet and DQDB**.
- A single node can almost fully load the ring as in **Token-rings and DQDB**.
- Fairness and asynchronous priority levels as in **IBM Token-ring**.
- Integration of synchronous and asynchronous traffic as in **FDDI**, but with a better fairness property.
- Transmission of variable size packets, in the buffer insertion mode, as in **Ethernet and Token-rings**.

The MetaRing implementation does not require new technology, and its design complexity is the same as other token rings (e.g., FDDI). However, with a better and more reliable performance. The potential aggregate throughput of the MetaRing is greater by a factor of four than a dual-token ring. Thus, this solution has much better cost effectiveness characteristics than token rings.

Acknowledgement

The author would like to thank Israel Cidon for his collaboration in the initial phases of the architecture, and to Hamid Ahmadi, Jeane Chen, Reuven Cohen, Alain Mayer, Rafail Ostrovsky, Adrian Segall, Khosrow Sohraby, Ho-Ting Wu and Moti Yung for their collaboration in other design aspects of the architecture. The author would also like to thank the anonymous reviewers for their helpful comments.

References

[1] Scientific American. Gigabit connection. *Scientific American*, pages 118–120, October 1990.

[2] R. Beeler, M. Potts, and S. Rao. D3q - the dynamic distributed dual queue. *ISS'90, XIII International Switching Symposium*, May 1990. Stockholm, Sweden.

[3] W. E. Burr. The FDDI optical data link. *IEEE Communication Magazine*, 24(5):18–23, May 1986.

[4] W. Bux, F. H. Closs, K. Kummerle H. J. Keller, and H. R. Mueller. Architecture and design of a reliable token-ring network. *IEEE J. on Selected Areas in Comm.*, SAC-1(5):756 – 765, November 1983.

[5] J. Chen, H. Ahmadi, and Y. Ofek. Performance study of a Gb/s MetaRing. *16th Conference on Local Computer Networks*, pages 136–147, October 1991.

[6] J. Chen, I. Cidon, and Y. Ofek. A local fairness algorithm for the MetaRing and its performance study. *GLOBECOM'92*, pages 1635–1641, 1992.

[7] J. Chen, I. Cidon, and Y. Ofek. A local fairness algorithm for gigabit LANs/MANs with spatial reuse. *IEEE J. on Selected Areas in Comm.*, 11(8):1183–1192, October 1993.

[8] I. Cidon and Inder S. Gopal. PARIS: An approach to integrated high-speed private networks. *Intern. J. on Digital and Analog Cabled Systems*, 1(2):77–86, April-June 1988.

[9] I. Cidon and Y. Ofek. Distributed fairness algorithm for local area networks with concurrent transmissions. In *the 3rd International Workshop on Distributed Algorithms*, pages 57–69. Springer Verlag Lecture Notes in Computer Science 392, September 1989. Also: IBM Research Report RC 15051, October 1989.

[10] I. Cidon and Y. Ofek. Fairness algorithm for full-duplex buffer insertion ring. *U.S. Patent*, 4926418, 1989.

[11] I. Cidon and Y. Ofek. MetaRing - a full-duplex ring with fairness and spatial reuse. *IBM Research Report*, RC 14961, September 1989.

[12] I. Cidon and Y. Ofek. MetaRing: A full-duplex ring with fairness and spatial reuse. *INFOCOM'90*, pages 969–981, 1990.

[13] I. Cidon and Y. Ofek. MetaRing - a full-duplex ring with fairness and spatial reuse. *IEEE Trans. on Comm.*, COM-41(1):110–120, January 1993.

[14] R. Cohen and Y. Ofek. Label swapping routing with self-termination. *INFOCOM'93*, 1993.

[15] R. Cohen, Y. Ofek, and A. Segall. A new label-based source routing in multi-ring networks. *The 3rd International Workshop on Protocols for High-Speed Networks (IFIP WG6.1/WG6.4)*, pages 69–84, 1992.

[16] R. Cohen, Y. Ofek, and A. Segall. A new label-based source routing for multi-ring networks. *IEEE T. on Networking*, 3(3):320–328, June 1995.

[17] R. M. Falconer and J. L. Adams. Orwell: a protocol for an integrated services local network. *British Telecom Technology Journal*, 3(4):27–35, October 1985.

[18] R. M. Grow. A timed token protocol for local area networks. *Electro/82, Boston, Massachusetts*, pages 1–7, 1982.

[19] E. R. Hafner, Z. Nenadal, and M. Tschanz. Integrated local communications - principles and realization. *Hasler Review*, 8(2):34–43, 1975.

[20] P. Heinzmann, H. R. Muller, D. A. Pitt, and H. R. van As. Buffer-insertion cell-synchronized multiple access (BCMA) on a slotted ring. *2nd International Conference on Local Communications Systems: LAN and PBX*, pages 223–248, June 1991. Palma - Balearic Islands, Spain.

[21] D. E. Hubber, W. Steinlin, and P. J. Wild. Silk: An implementation of a buffer insertion ring. *IEEE J. on Selected Areas in Comm.*, SAC-1(5):766–774, November 1983.

[22] P. Kermani and L. Kleinrock. Virtual cut-through: A new computer communication switching technique. *Computer Networks*, 3:267–286, September 1979.

[23] A. A. Lazar, A. T. Temple, and R. Gidron. MAGNET II: A metropolitan area network based on asynchronous time sharing. *IEEE J. on Selected Areas in Comm.*, SAC-8(8):1582–1594, October 1990.

[24] M. T. Liu and D. M. Rouse. A study of ring networks. *Proc. IFIP WG6.4/University of Kent Workshop on Ring Technology Based Local Area Networks*, pages 1–39, September 1983.

[25] A. Mayer, Y. Ofek, R. Ostrovsky, and M. Yung. Self-stabilizing symmetry breaking in constant-space. *ACM-STOC*, pages 667–678, 1992.

[26] Y. Ofek. Integration of multi-ring on the MetaRing architecture. *2nd IEEE Workshop on Future Trends of Distributed Computing Systems*, pages 190–196, 1990.

[27] Y. Ofek. Overview of the MetaRing Architecture. *Computer Networks and ISDN Systems*, 26(6-8):817–830, March 1994.

[28] Y. Ofek, K. Sohraby, and H. Wu. Integration of synchronous and asynchronous traffic on the MetaRing architecture and its analysis. *IEEE/ACM T. on Networking*, 5(1):111–121, February 1997.

[29] Y. Ofek and M. Yung. Efficient mechanism for fairness and deadlock-avoidance in high-speed networks. *The 4th International Workshop on Distributed Algorithms*, pages 192–212, September 1990.

[30] H. Ohnishi, N. Morita, and S. Suzuki. ATM ring potocol and performance. *ICC'89*, pages 394–398, 1989.

[31] F. E. Ross. FDDI - a tutorial. *IEEE Communication Magazine*, 24(5):10–17, May 1986.

[32] A. Segall, T. Barzilai, and Y. Ofek. Reliable multi-user tree setup with local identifiers. *IEEE J. on Selected Areas in Communications*, 9(9):1427–1439, December 1991.

[33] R. Simha and Y. Ofek. A starvation-free access protocol for a full-duplex buffer insertion ring local area network. *Computer Networks and ISDN Systems*, 21(2):109–120, 4 1991.

[34] C. A. Sunshine. Source routing in computer networks. *ACM Computer Communication Review*, 7(1):29–32, January 1977.

[35] H. R. van As, W. W. Lemppenau, and P. Zafiropulo. Performance of cRMA-II: A reservation-based fair media access protocol for Gbit/s lANs and mANs with buffer insertion. *EFOC/LAN'92*, pages 162–169, June 1992.

[36] H. R. van As, W. W. Lemppenau, P. Zafiropulo, and E. A. Zurfluh. CRMA-II: A Gbit/s MAC protocol for ring and bus networks with immediate access capability. *EFOC/LAN'91*, pages 262–277, June 1991.

[37] H. Wu, Y. Ofek, and K. Sohraby. Integration of synchronous and asynchronous traffic on the MetaRing architecture and its analysis. *ICC'92*, pages 147–153, 1992.