

# SERVICE RESTORATION IN SWITCHED ETHERNET NETWORKS

## Introduction

Over the past couple of decades, along with an increase in the speed of data communications networks, there has been an increase in the expectation of availability of those networks.

Service providers have been selling data services to their subscribers with guarantees of performance with several dimensions (Service Level Agreements) including availability. Increasingly, the nominal availability has been set at “five 9’s” or 99.999%. This translates into downtime of 5¼ minutes per year (or about 27 seconds a month).

Many service providers have begun using Ethernet to deliver new services, such as voice and video. These services require very low levels of frame loss in order to provide an acceptable end-user experience. MPEG4 delivery of standard definition TV sends about 100 frames per second, or one frame every 10 ms. Loss of a single frame can result in a noticeable artifact on screen.

These availability requirements substantially reduce the acceptable recovery time from network failures. In the past, if a network link went down it would often take minutes or even hours to detect the failure, determine the cause of the failure, and take recovery action. Clearly in the face of five 9’s of availability this activity has to happen in well under a second. In fact, with the deployment of automatic protection switching in SONET/SDH networks the expectation is that recovery can happen in 50ms or less.

As packet network technology (i.e., Ethernet) has begun to augment or replace SONET/SDH in service provider deployments, the expectation of 50ms failure recovery has been maintained. Ethernet, however, is not SONET/SDH and the structure and design of Ethernet-based networks are different than SONET/SDH. As a result, achieving 50ms failure recovery in Ethernet has been slow to happen.

The goal of this white paper is to describe the various protection switching techniques that are available for Ethernet and the applicability of these techniques to carrier networks. We briefly discuss protection switching in SONET/SDH to provide some background and then continue on to discuss the various protection switching mechanisms available in Ethernet networks.

## Detection and Restoration

Recovery from a network failure has two primary components: *detection* and *restoration*.

Before recovery can be effected, the failure has to be detected. Sometimes this is easy such as when a fiber is pulled out and signal is lost. Often it is more difficult such as when framing is lost due to a line error or if the signal is degrading. Clearly, without knowledge of a failure the system can’t begin to respond. In addition to having well-defined criteria for deciding that there has been a failure, the systems must be designed to get fast notification. Polling a link once a second to see if it has a loss of signal will not allow the system to recover in 50ms and polling very frequently could have an impact on system performance.

Ethernet has traditionally made detection of failures difficult. Early 10Mbps and 100Mbps Ethernet did not transmit anything between frames, therefore it was impossible to detect failures unless somebody was transmitting. You had to wait.

Fiber optic Ethernet improved the situation by allowing detection of loss of light (LOS) very quickly—but more subtle issues such as a failure on the remote node or a degraded signal were still difficult to diagnose.

On the other hand SONET/SDH and OTN<sup>1</sup> send a continual stream of frames regardless of whether there is data to send. This provides the ability to detect more subtle failures through the loss of frame synchronization (LOF). When SONET/SDH (based on the A1/A2 bytes) and OTN (based on the FAS bytes) lose synchronization with their frame structure they declare an out of frame condition (OOF). If an OOF condition persists for 3ms they declare a loss of frame (LOF) condition.

OTN also has strong forward error correction (FEC) which provides two additional benefits. First, many errors can be corrected on the fly reducing the overall frame loss (and/or increasing the distance that can be driven). The

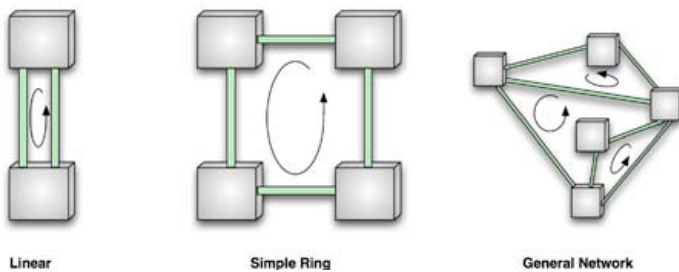
<sup>1</sup> OTN is the Optical Transport Network defined in ITU standards G.872 and G.709. OTN uses a fixed frame size of 16320 bytes and has several frame rates defined to carry OCn/STMn payloads. OTU2 operates at 10.709Gbps and can carry an OC192/STM64 payload or a 10Gbps Ethernet.

FEC also allows the system to measure the bit error rate<sup>2</sup> (BER). The BER allows the system to detect a *degrading* link rather than just a failed link allowing a preemptive protection switch before the link fails completely or the error rate becomes unacceptable.

Once the failure is detected the system/network can respond i.e., restoration can occur.

Protection schemes vary in a number of ways. The most important distinction is topology. Some protection schemes protect a single line—a connection between two systems or nodes—while others protect a number of nodes that are connected together to form a more complex network. In all cases there is logically a loop—multiple paths between protected systems—although we don't always think of it as a loop. When protection is across more than two nodes, those nodes can be configured into a simple loop, usually referred to as a ring, or a more complex network as illustrated in the following diagram.

### Network Topologies



APS (Automatic Protection Switching) generically describes the various mechanisms used to recover from network failures.

### SONET Protection Switching

All of this high speed, 50ms protection/restoration started with SONET/SDH, so the discussion will start there. Starting with SONET/SDH allows the introduction of relevant terminology and many of the concepts in SONET/SDH APS will be seen in the various Ethernet APS mechanisms.

SONET/SDH protection falls into two categories: linear and ring.

Linear protection switching protects the connection between two nodes. There are two types of linear APS in SONET/SDH, 1+1 and 1:n.

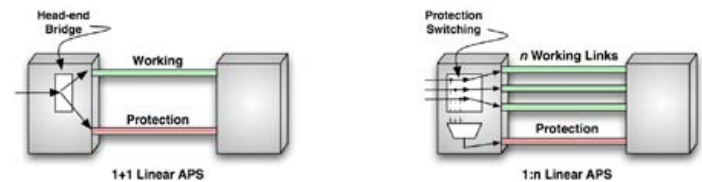
With 1+1 protection there are two physical (bidirectional) links between the nodes, a *working* link and a *protection* link. All transmitted traffic is sent over both links. This is called *head-end bridge*<sup>3</sup>. At the other end the receiver selects one of the two links based on configuration and real-time quality metrics (link

available, bit error rate, etc.). This is called *tail-end select*. If the selected link fails, the receiver just selects the *other* link and continues operating. Transmit and receive work independently (i.e. both sides are doing a head-end bridge and tail-end select).

The approach is simple. The nodes operate independently and no signaling or protocol is needed to recover from a failure. A further benefit is that restoration is almost immediate—once the fault is detected, the restoration time is very close to zero. The primary downside is that an entire extra facility is dedicated to protection. The available bandwidth is only 50% of the total provisioned facility.

When more than one working link<sup>4</sup> is connected between nodes, linear 1:n APS improves the link utilization at the cost of additional complexity. With this technique there are  $n$  working links and one protection link. If there is a link failure detected at the tail end (receiver), it must notify the transmitter and request that the link be bridged to the protection link. This technique improves link utilization (to  $\frac{n}{n+1}$ ) but requires signaling to occur. The restoration isn't quite instantaneous but it is still very fast. Linear 1+1 and 1:n APS are shown in the following diagram.

### SONET Linear Protection Switching



There are several types of ring-based APS in SONET/SDH. The distinctions are whether traffic is normally running in both directions around the ring, whether there is a single (bidirectional) link between ring nodes, and whether lines are protected or paths are protected. There are four common used techniques: Unidirectional Line-Switched Ring (ULSR), Unidirectional Path-Switched Ring (UPSR), Bidirectional Path-Switched Ring (BPSR), and Bidirectional Line-Switched Ring (BLSR). We will discuss ULSR because it is the simplest of the four.

With ULSR there is a single fiber pair between each node in the ring. This can be thought of as two rings, one clockwise and one counter-clockwise. All inserted traffic is sent on the *working* ring (counter-clockwise in the following diagram). This is what makes the technique *unidirectional*.

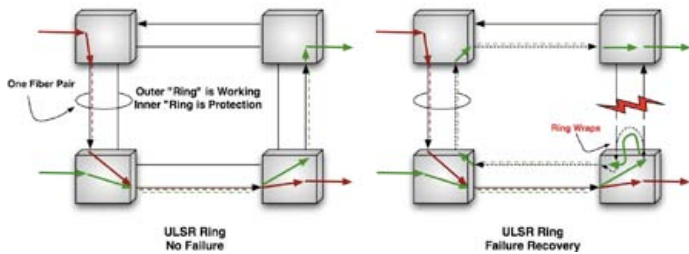
When a failure occurs, the node that detects the failure *wraps* the working ring onto the protection ring as shown in the following diagram. This ensures that all traffic can still reach all nodes.

<sup>2</sup> SONET/SDH also supports signal degrade detection through the use of Bit Interleaved Parity (BIP).

<sup>3</sup> Don't confuse this use of the term *bridge* with an Ethernet bridge (today called an Ethernet Switch).

<sup>4</sup> 1:n can be used if there is only one working link (1:1) but there is a benefit over 1+1 only when the protection link is qualitatively less desirable than the working link (e.g., higher latency, higher error rate, etc.)

### ULSR - Before and After



ULSR operates a bit differently. Each node bridges inserted traffic on a path/tributary basis onto both rings as with a head-end bridge. The receiver selects a signal from one ring or the other.

BLSR and BPSR use two pair of fiber<sup>5</sup> between each node on the ring which provides a dedicated bidirectional OCn working link and equivalent protection capability. Using two pair of fiber increases the cost (both of the fiber and linecard ports) but provides both increased capacity and increased availability (and increased complexity).

### Revertive vs. Non-Revertive Switching

When a failed link is restored the APS mechanism must decide whether to *revert* to its original state or to maintain the status quo. Some techniques, such as 1+1 linear APS, frequently do not revert since the protection path is as good as the working path. Reverting provides an opportunity for another *glitch* in the network that might lose a frame or two so why do it if you don't have to? On the other hand, ULSR usually does revert since the protection path is much longer than the working path.

With SONET/SDH, when the APS is configured for revertive switching, there is a Wait To Restore (WTR) timer. When the failure is resolved, the timer is started and the reversion does not happen until it expires. If another failure occurs before the WTR time expires, the timer is reset. This behavior allows a bouncing link to settle out before the network is reconfigured.

### Ethernet Protection Mechanisms

We will look at four protection mechanisms for Ethernet:

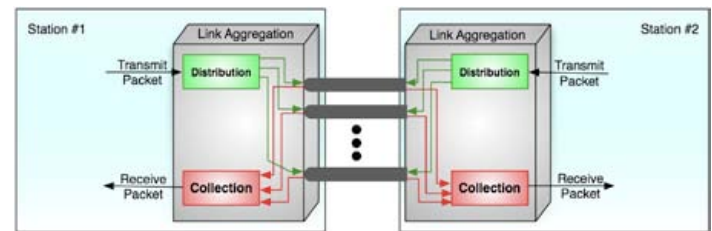
1. Link Aggregation
2. Spanning Tree
3. G.8032 Ring Protection Switching
4. G.8031 Ethernet Linear Protection Switching

### Link Aggregation

Link Aggregation is a mechanism specified in 802.3<sup>6</sup> both to increase the effective bandwidth between two Ethernet nodes and also to provide protection.

Link aggregation allows two nodes to be connected by multiple Ethernet links (the Link Aggregation Group or LAG) and provides rules on how these links are used to carry the Ethernet traffic between nodes. The links must all be the same speed (10Mbps, 100Mbps, 1000Mbps, etc.) and must all be full-duplex. You can think of link aggregation like this:

#### Ethernet Link Aggregation



One important aspect of link aggregation (and all other techniques that forward Ethernet packets) is that packets from a session cannot be re-ordered. The standard puts the responsibility for this on the *distribution* (transmit) function.

If the distribution function used a simple round robin technique and placed each packet on the next link in order (i.e., first packet on link 1, next packet on link 2, ...) then there is a likelihood of packets being received out of order. Consider a session that sends a 64 byte packet, then two 9600 byte packets then another 64 byte packet over a three link LAG. The first 64 byte packet would be put on link 1, the next packet (9600 bytes) would be placed on link 2, the next packet (9600 bytes) would be transmitted on link 3. The fourth packet (64 bytes) would be put on link 1. It is possible that the last packet would be received on the other side before either of the two 9600 byte packets.

In order to avoid this problem, the *distribution function* ensures that all packets from the same *session* are sent on the same link. The *session* can be determined in several ways. It can be based on the Ethernet destination address or source address or some combination of source and destination address or possibly based on the IP addresses within the packet.

What this means, is that a link aggregation group containing  $n$  links does not really provide  $n$  times as much bandwidth as a single link. In fact, if, during some period all of the traffic is from a single session then the bandwidth is just a single link. Over time link aggregation provides between 1 and  $n$  times the link bandwidth depending on the traffic mix.

<sup>5</sup> BLSR can be implemented with a single pair of fiber but this is not common.

<sup>6</sup> See IEEE 802.3-2005 section 43.

Whatever address or address combination is used, the value must be turned into a link number between 1 and  $n$ . The mechanism for doing this can be simple or complex. The reason for choosing a more complex technique is to increase the chances that two different inputs, such as Ethernet destination addresses, that are similar will *hash* to different links. This increases the effective bandwidth.

Since the *distribution function* is responsible for ensuring that packets are not reordered, the *collection function* can use any strategy it wants for taking packets from the LAG and passing them up to the switch. This fact is important for protection purposes.

If the network is configured such that the traffic to be transferred will fit in less<sup>7</sup> than  $n-1$  of the Ethernets, then Link Aggregation can be used as a protection technique as well as a bandwidth enhancement technique. When link aggregation detects a failed link it can simply stop using the link. All it needs to do is change its distribution function to generate a number from 1 to  $n-1$  rather than from 1 to  $n$ . This is, effectively, a  $1:n-1$  linear protection mechanism. There is no designated protection link. All links are used during normal operation and when a failure occurs the traffic on the failed link is simply spread across the other links.

Since the topology is constrained and most of the reconfiguration can happen autonomously (i.e., without substantial protocol exchange) Link Aggregation reconfiguration can happen very fast.

Link Aggregation is usually revertive. When a failed link is restored it is usually just added back into the LAG.

## Spanning Tree Protocol

The original spanning tree protocol (STP) is about 20 years old. People have a negative reaction to the Spanning Tree protocol for a number of reasons. Early implementations of the protocols were, well, early. They had bugs. They had interoperability problems. As a result early bridged networks that used STP ended up with unresolved loops which bring networks down instantaneously (not a good way to win friends and influence people). In addition, the convergence time for the original spanning tree protocol was long—tens of seconds and sometimes many tens of seconds.

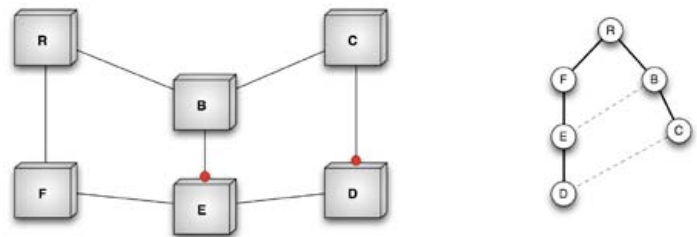
Ethernet, at that time, was only 10Mbps, and was used primarily as a way to connect PCs to file servers. Recovery from most faults took a while and, as noted earlier in this document, it was hard to detect failure on early Ethernets. Ethernet was not used for mission critical applications nor was it used by service providers.

That being said, STP is in many ways the most sophisticated protection mechanism being discussed in this white paper. Ethernet bridged networks were viewed as general packet-switched networks that were built with topologies that matched the necessary physical connectivity requirements. STP can detect a loop or multiple loops in any arbitrary topology of Ethernet switches<sup>8</sup> (not just point-to-point or a simple ring), break the loops, and bring a “protection” link into service in the event of a network failure. This puts it in a class similar to standard routing protocols such as OSPF and RIP—but whereas those protocols are willing to allow a (short duration) transient loop during a network reconfiguration, the spanning tree protocol is not. Putting loop avoidance as a primary design criterion has an impact on network restoration time.

Spanning Tree bridges send protocol messages (often referred to as BPDUs<sup>9</sup>) on attached Ethernets. On any single Ethernet segment (point-to-point or shared) only one bridge is sending messages. These messages are usually sent every 2 seconds. So the protocol uses very little bandwidth.

Using these messages, loops are found in the network, and one of the bridges in each loop puts one of its ports into a *blocking* state so it is not transmitting messages or processing received messages other than spanning tree messages. Consider the following diagram:

### Spanning Tree Network with Two Loops



In the diagram above (left picture) there are two loops, RBEF and BCDE (and actually there is a third loop composed from the other two, RBCDEF). Since there are two loops we need to block two ports and the spanning tree algorithm has chosen the top port on E and the top port on D (these choices are not arbitrary or random, the algorithm is completely deterministic). These ports will not transmit messages nor process received messages. They have, effectively, become the protection ports. The picture on the right depicts the tree that results (a tree is a network without loops).

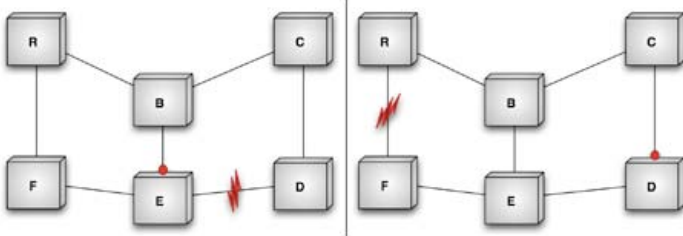
If there is a failure in the network, one or both of the blocking ports will be activated in order to heal the failure. The following picture shows two different failure scenarios. The left picture shows a failure on the ED link and the picture on the right shows a failure on RF.

7 This needs to be fairly conservative here since it isn't clear how the traffic will be split over the remaining links.

8 The terms *Ethernet Bridge* and *Ethernet Switch* are used interchangeably.

9 Bridge Protocol Data Units. PDU is a term used in many standards to refer to a packet that carries a protocol message.

### Spanning Tree Networks with Failures



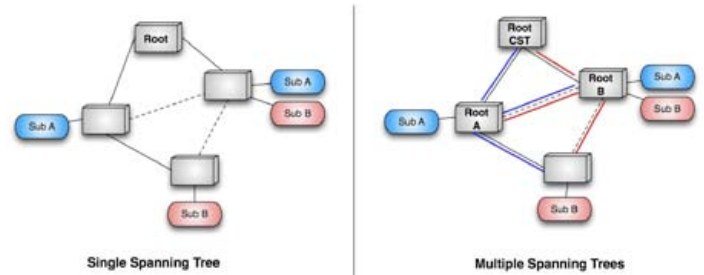
In the case of the ED failure, the blocking port on top of D becomes forwarding. Now D can be reached by way of C. With the RF failure on the right, the blocking port on the top of E that becomes forwarding. This failure (on the right) demonstrates that the spanning tree can reconfigure network nodes for failures that are distant to the node being reconfigured.

But what good is it if it converges in tens of seconds or minutes? The IEEE has addressed this issue to some extent. When the STP was first developed all Ethernets were shared. Other bridges could come and go on a network and create loops. One bridge really didn't know how many other bridges were on a particular LAN segment nor how the bridges were connected. As a result the protocol had to be very conservative and careful. Creating a loop on a LAN will take the LAN down instantaneously. Therefore the STP put loop avoidance above all other criteria. With the advent of point-to-point switched Ethernet many of these problems vanished. One bridge knows that there is exactly one other bridge on the other side of the link. This has allowed several enhancements to the protocol's convergence time. In addition a port can be configured as an *edge* port which has no bridges on the other side. These and several other enhancements make up the Rapid Spanning Tree Protocol (RSTP). RSTP can converge much faster than STP in many network configurations. Many configurations converge in under a second, and some configurations converge in milliseconds (as noted at the beginning of the paper, all of these times depend on being able to detect the failure and respond to it in an even shorter amount of time). RSTP can, in many configurations, achieve the 50ms restoration time expected by service providers. But not in all configurations and the restoration time is dependent on the network topology and where the failure occurs.

Note that RSTP nodes must also implement STP and if there is an STP node in the network the other nodes must "talk down" to it.

Another independent improvement in the spanning tree protocol has been the development of the Multiple Spanning Tree Protocol (MSTP). MSTP was developed to address the issue of sub-optimal forwarding in RSTP networks. Consider the following network:

### Multiple Spanning Trees



In this network there are two subscribers, the Sub A (blue) and the and Sub B (red). In a spanning tree network traffic must flow up towards the root of the tree until the first connection with the branch that contains the destination. In the left picture, for one Sub A to talk with its peer the traffic must go two hops (up to the root and down again). This is not too bad but given the direct connection between the two bridges it could be better. For Sub B to talk with its peer, however, requires 3 hops rather than one. There isn't any single placement of the root that would optimize both subscribers. MSTP addresses this problem by allowing the creation of multiple spanning trees. Each spanning tree has its own root. VLANs are assigned to spanning trees. The right side of the above diagram shows this capability. There are now multiple spanning tree *instances*, the blue one for Sub A (with the left bridge as the root) and the red one for Sub B (with the right bridge as the root). There is also still the original spanning tree — called the Common Spanning Tree — with its root still at the top. The Common Spanning Tree is used for all VLANs that are not assigned to specific MSTP instances. It is also used to communicate with bridges that do not implement MSTP.

Each MSTP instance runs RSTP. Since the Common Spanning Tree Instance runs RSTP and RSTP nodes must be able to talk down to STP nodes, MSTP bridges can live in the same network as both RSTP bridges and STP bridges.

## The Next Generation

Ethernet has been changing. Or rather, Ethernet's use has been changing. Ethernet has been migrating from an enterprise/data center technology to a carrier technology and from a relatively local to a wide-area network.

This migration has a number of important ramifications. As carriers have migrated to Ethernet they have built Ethernet topologies that mirror their existing SONET/SDH topologies — i.e., linear connections and rings (and cascaded rings). More importantly, carriers have been using Ethernet as a *service-based network* rather than a peer-to-peer packet-switched network. In some ways the connectionless Ethernet is being asked to mimic the connection-oriented capabilities of its predecessors.

Obviously traditional Ethernet technologies don't address these changes. Link Aggregation and Spanning Tree are powerful solutions — but for the wrong problems.

Over the last few years several vendors have developed various protocols and techniques for Ethernet protection to address some of the deficiencies in spanning tree (e.g., Cisco has developed spanning tree improvements such as *portfast*, *uplinkfast*, and *backbonefast*) and to improve performance in simple rings. Cisco has developed Resilient Ethernet Protocol (REP), Foundry has developed Metro Ring Protocol (MRP), Extreme has developed Ethernet Automatic Protection Switching (EAPS). Each of these addresses the issues in various ways. But the problem is that they are vendor-specific. This limits interoperability and there are very few other vendors that have implemented these protocols.

To address this problem, the International Telecommunications Union (ITU) has been developing protocols for fast restoration in service-oriented Ethernet networks. These protocols are, in a sense, moving Ethernet closer to SONET/SDH since they focus on specific topologies (rings and linear, sound familiar?) rather than the general/arbitrary topology handled by the spanning tree protocol. Since the topologies are constrained the restoration time is faster. Also these protocols provide restoration more like *path* restoration rather than *line* restoration since they are done at the VLAN level. The two protocols are *G.8031 – Ethernet Protection Switching* and *G.8032 – Ethernet Ring Protection Switching*.

These new protocols depend on another ITU standard, *Y.1731 – OAM functions and mechanisms for Ethernet based networks*. Y.1731 provides several maintenance mechanisms for Ethernet physical and logical connections including continuity checking, loopback, link trace, and performance monitoring. The format of the Y.1731 message used for ring protection is as follows:

### Y.1731/G.8032 Message Format

MEL	Version	OpCode (R-APS = 40)	Flags (0)	TLV Offset (32)
R-APS Specific Information (32 octets)				
[optional TLV starts here; otherwise End TLV]				
				End TLV (0)

The second byte of the message is the OpCode. This field indicates the type of operation:

- OpCode 1 is Continuity Check Message
- OpCode 2 is Loopback Message
- OpCode 3 is Loopback Response
- OpCode 4 is Link Trace
- etc.

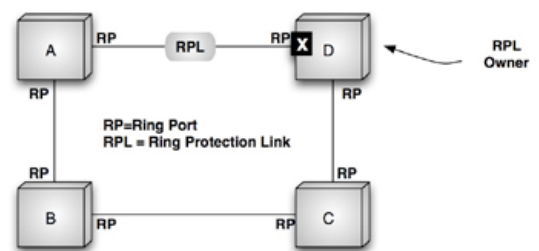
The protection switching techniques make use of the Y.1731 continuity checking for detecting network faults. The APS messages used in G.8031 and G.8032 employ the message structure defined in Y.1731 (OpCodes 39 and 40, respectively).

## G.8032 - Ring Protection Switching

G.8032 Ring Protection Switching is a fairly simple mechanism and its approach has a lot in common with STP.

For a set of Ethernet bridges in a ring, each bridge on the ring is explicitly configured with its *ring ports*. One of the links around the ring is declared the *Ring Protection Link (RPL)* and one of the nodes that connects to the RPL is declared the *RPL owner*. This is all done through configuration<sup>10</sup>. The RPL owner blocks its port connected to the RPL to break the loop (this is the same sense as blocking in the spanning tree, i.e., no transmission or reception of data packets). The resulting network looks like this:

### G.8032 Ring in Initial State

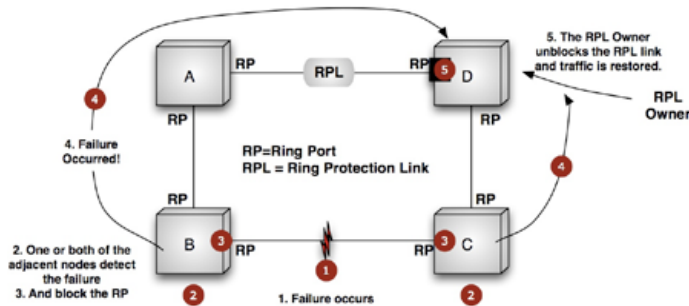


When a failure occurs on the ring, the node(s) that detect the failure first block their ring ports on the failing link (to prevent a loop from happening if the failure magically disappears) and then send a message around the ring indicating that

<sup>10</sup> In theory the RPL and RPL owner could be assigned automatically but in the current standard they are manually configured.

a failure has occurred. When the RPL owner receives this message it unblocks the RPL port and ring connectivity is restored. The sequence of events looks something like this:

#### Failure Recovery in G.8032



At this point the topology is stable and traffic can reach all nodes.

When the failed link returns to operation the network can either revert to normal operation, or not. This is another difference between G.8032 and spanning tree which always reverts immediately. If revertive operation is selected then the first step is to wait for the Wait To Restore (WTR) time to expire. When WTR expires, the nodes adjacent to the (previously) failing link send a message around the ring noting the re-establishment of the link. When this message is received by the RPL owner it blocks its RPL port and sends a message around the ring indicating that the RPL is blocked. When nodes B and C receive this message they unblock their ring ports and the ring is back to normal operation. Note that the order of all of these operations is carefully designed to avoid the formation of a loop.

Since there are no timers involved in the restoration and reversion (traffic is flowing while waiting for WTR to expire) these operations can happen very fast. They just depend on the propagation of messages around the ring. The G.8032 standard sets the following performance goal<sup>11</sup>:

In an Ethernet ring, without congestion, with all nodes in the idle state (i.e., no detected failures, ...), with less than 1200 km of ring fibre circumference, and fewer than 16 nodes, the switch completion time (transfer time) for a failure on a ring link shall be less than 50ms. On rings under all other conditions, the switch completion time may exceed 50ms (the specific interval is under study), to allow time to negotiate and accommodate coexisting APS requests.

G.8032 as currently defined by the ITU supports single rings as shown in the above diagrams. Work is ongoing to extend it to cascaded rings that are connected at single points and multiple points.

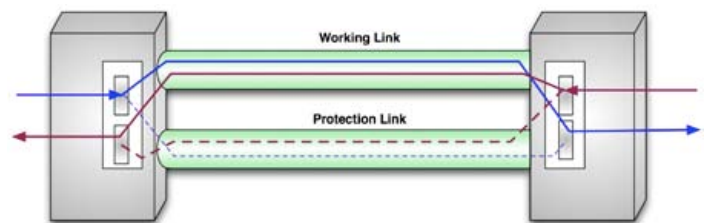
## G.8031 - Ethernet (Linear) Protection Switching

G.8031 provides a linear protection mechanism that is very similar to SONET/SDH linear APS. However it is more general than SONET linear APS. G.8031 can protect a link between two adjacent nodes using a protection link and it can protect a point-to-point VLAN between two nodes in a network using a protection VLAN. It requires a working entity and a protection entity and supports three modes of operation:

- 1+1 Unidirectional Protection Switching
- 1+1 Bidirectional Protection Switching
- 1:1 (Bidirectional) Protection Switching

The basic G.8031 configuration looks like this:

#### Basic G.8031 Configuration



Each bridge has an APS process that determines where each transmit packet should be sent and selects which link to use to receive packets. The APS process also sends and processes APS messages between the nodes.

**1+1 Unidirectional Protection Switching** is closest to SONET 1+1 APS. Each side sends all packets over both links, the working link and the protection link, i.e., *head-end bridge*. At the receive side a selection is done based on some set of criteria, i.e., *tail-end select*. This is a *unidirectional* approach because each side makes its selection decision independently. It is therefore possible (e.g. if the failure is in only one direction) that one side will be receiving from the working link and one side will be receiving from the protection link. This is the only one of the three modes that does not require any signaling (protocol messages) between the nodes in order to effect a protection switch.

**With 1+1 Bidirectional Protection Switching** the head-end bridge is done the same way as described in the previous paragraph. With this method, however, both tail-ends always select the same link. They are usually configured to select the working link initially, but if there is a failure on the working link both sides will switch to the protection link. This obviously requires signaling between the two nodes. But clearly this can happen very quickly.

**With 1:1 Protection Switching** there is no head-end bridge. Traffic is sent down only one link (initially the working link) and it is switched to the other link on failure. This obviously requires signaling ("I have an LOS on the link please send on the other link") and therefore only supports bidirectional switching (if you have to exchange messages anyway why not coordinate the receive links).

11 See ITU-T G.8032 section 7.3.

G.8031 also supports both revertive (with WTR timer) and non-revertive operation. The standard covers this topic clearly<sup>12</sup>:

1+1 protection is often provisioned as non-revertive, as the protection is fully dedicated in any case, and this avoids a second “glitch” to the normal traffic signal. There may, however, be reasons to provision this to be revertive (e.g., so that the normal traffic signal uses the “short” path except during failure conditions. Certain operator policies also dictate revertive operation even for 1+1).

1:1 protection is usually revertive. Although it is possible to define the protocol in a way that would permit non-revertive operation for 1:1 protection, however, since the working transport entity is typically more optimized (i.e., from a delay and resourcing perspective) than the protection transport entity, it is better to revert and glitch the normal traffic signal when the working transport entity is repaired.

In general, the choice of revertive/non-revertive will be the same at both ends of the protection group. However, a mismatch of this parameter does not prevent interworking; it just would be peculiar. . .

## G.8031/8032 Carrier Characteristics

In addition to providing support for common carrier topologies and fast fault recovery these protocols also provide a number of configuration options that are similar to SONET/SDH. Controlled reversion is one these options discussed above.

In addition there are controls on the APS itself. Both protocols support the ability to manually perform a protection switch (e.g., so that the primary facility can have maintenance done) or inhibit a protection switch completely.

None of these capabilities are natively available with link aggregation or spanning tree.

## Summary

Ethernet is becoming a carrier technology. Ethernet is simpler and less expensive than existing synchronous carrier technologies such as SONET/SDH, but won't really be able to replace these networks unless it can provide many of the same characteristics. Availability and service recovery time after a failure are important characteristics.

Over the life of Ethernet various protocols and techniques have been developed to address protection and restoration, the existing standards solutions — spanning tree and link aggregation — don't provide adequate solutions.

Link Aggregation can recover quickly from network failures but is not deterministic. Since it is difficult to know how traffic is split across the links in a link aggregation group, ensuring that there is sufficient bandwidth to provide the necessary services is difficult, at best. Spanning tree, on the other hand, is a powerful and deterministic protection mechanism, but because it was designed to operate in an arbitrary topology its convergence time is frequently too long to meet the demands of today's service providers.

Other approaches that come closer to solving the problem are vendor-specific and not widely implemented.

The long-term solution is reflected in the efforts of the ITU and several other organizations<sup>13</sup> to define a set of concepts and protocols which are standardized and available industry-wide. These approaches have been designed to operate in topologies that are commonly deployed in carrier networks and meet the stringent availability requirements of today's services.

## References and Sources

- Goralski, Walter (2002). *SONET/SDH, 3rd Edition*. McGraw-Hill/Osborne.
- Gorshe, Steve (2005). *A Tutorial on SONET/SDH Automatic Protection Switching (APS)*. PMC-Sierra, Inc.
- IEEE Computer Society. *IEEE Std 802.1D™-2004, Media Access Control (MAC) Bridges*.
- IEEE Computer Society. *IEEE Std 802.1Q™-2005, Virtual Bridged Local Area Networks*.
- IEEE Computer Society. *IEEE Std 802.3™-2005, Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*.
- International Telecommunications Union (2003). *ITU-T G.709/Y.1331, Interfaces for the Optical Transport Network OTN*.
- International Telecommunications Union (2006). *ITU-T G.8031/Y.1342, Ethernet Protection Switching*.
- International Telecommunications Union (2008). *ITU-T G.8032/Y.1342, Ethernet Ring Protection Switching*. (Prepublished recommendation)
- International Telecommunications Union (2008). *ITU-T Y.1731, OAM functions and mechanisms for Ethernet based networks*.
- International Telecommunications Union (2007). *Quality of Experience Requirements for IPTV*.

<sup>12</sup> See ITU-T standard G.8031 section 10.3.

<sup>13</sup> Other organizations such as the Metro Ethernet Forum (MEF) are providing underpinnings for the ITU solutions by formalizing many concepts associated with making Ethernet look more like a connection-oriented network.