Mixed-Media Bridging

Background

Transparent bridges are found predominantly in Ethernet networks, and *source-route bridges* (SRBs) are found almost exclusively in Token Ring networks. Both transparent bridges and SRBs are popular, so it is reasonable to ask whether a method exists to directly bridge between them. Several solutions have evolved.

Translational bridging provides a relatively inexpensive solution to some of the many problems involved with bridging between transparent bridging and SRB domains. Translational bridging first appeared in the mid- to late-1980s but has not been championed by any standards organization. As a result, many aspects of translational bridging are left to the implementor.

In 1990, IBM addressed some of the weaknesses of translational bridging by introducing *source-route transparent* (SRT) bridging. SRT bridges can forward traffic from both transparent and source-route end nodes and form a common spanning tree with transparent bridges, thereby allowing end stations of each type to communicate with end stations of the same type in a network of arbitrary topology. SRT is specified in the IEEE 802.1d Appendix C.

Ultimately, the goal of connecting transparent bridging and SRB domains is to allow communication between transparent bridges and SRB end stations. This chapter describes the technical problems that must be addressed by algorithms attempting to do this and presents two possible solutions: translational bridging and SRT bridging.

Translation Challenges

Many challenges are associated with allowing end stations from the Ethernet/transparent bridging domain to communicate with end stations from the SRB/Token Ring domain:

- Incompatible bit ordering—Although both Ethernet and Token Ring support 48-bit Media Access Control (MAC) addresses, the internal hardware representation of these addresses differ. In a serial bit stream representing an address, Token Ring considers the first bit encountered to be the high-order bit of a byte. Ethernet, on the other hand, considers the first bit encountered to be the low-order bit.
- Embedded MAC addresses—In some cases, MAC addresses actually are carried in the data portion of a frame. The *Address Resolution Protocol* (ARP), a popular protocol in *Transmission Control Protocol/Internet Protocol* (TCP/IP) networks, for example, places hardware addresses in the data portion of a link-layer frame. Conversion of addresses that might or might not appear in the data portion of a frame is difficult because they must be handled on a case by case basis.

- Incompatible maximum transfer unit (MTU) sizes—Token Ring and Ethernet support different maximum frame sizes. Ethernet's MTU is approximately 1500 bytes, whereas Token Ring frames can be much larger. Because bridges are not capable of frame fragmentation and reassembly, packets that exceed the MTU of a given network must be dropped.
- Handling of *frame-status bit actions*—Token Ring frames include three frame-status bits: A, C, and E. The purpose of these bits is to tell the frame's source whether the destination saw the frame (A bit set), copied the frame (C bit set), or found errors in the frame (E bit set). Because Ethernet does not support these bits, the question of how to deal with them is left to the Ethernet-Token Ring bridge manufacturer.
- Handling of exclusive Token Ring functions—Certain Token Ring bits have no corollary in Ethernet. Ethernet, for example, has no priority mechanism, whereas Token Ring does. Other Token Ring bits that must be thrown out when a Token Ring frame is converted to an Ethernet frame include the token bit, the monitor bit, and the reservation bits.
- Handling of explorer frames—Transparent bridges do not inherently understand what to do with SRB explorer frames. Transparent bridges learn about the network's topology through analysis of the source address of incoming frames. They have no knowledge of the SRB route-discovery process.
- Handling of routing information field (RIF) information within Token Ring frames—The SRB algorithm places routing information in the RIF field. The transparent-bridging algorithm has no RIF equivalent, and the idea of placing routing information in a frame is foreign to transparent bridging.
- Incompatible spanning-tree algorithms—Transparent bridging and SRB both use the spanning-tree algorithm to try to avoid loops, but the particular algorithms employed by the two bridging methods are incompatible.
- Handling of frames without route information—SRBs expect all inter-LAN frames to contain route information. When a frame without an RIF field (including transparent bridging configuration and topology-change messages as well as MAC frames sent from the transparent-bridging domain) arrives at an SRB bridge, it is ignored.

Translational Bridging

Because there has been no real standardization in how communication between two media types should occur, no single translational bridging implementation can be called correct. The following describes several popular methods for implementing translational bridging.

Translational bridges reorder source and destination address bits when translating between Ethernet and Token Ring frame formats. The problem of embedded MAC addresses can be solved by programming the bridge to check for various types of MAC addresses, but this solution must be adapted with each new type of embedded MAC address. Some translational-bridging solutions simply check for the most popular embedded addresses. If translational-bridging software runs in a multiprotocol router, the router can successfully route these protocols and avoid the problem entirely.

The RIF field has a subfield that indicates the largest frame size that can be accepted by a particular SRB implementation. Translational bridges that send frames from the transparent-bridging domain to the SRB domain usually set the MTU size field to 1,500 bytes to limit the size of Token Ring frames entering the transparent-bridging domain. Some hosts cannot correctly process this field, in which case translational bridges are forced to drop those frames that exceed Ethernet's MTU size.

Bits representing Token Ring functions that have no Ethernet corollary typically are thrown out by translational bridges. Token Ring's priority, reservation, and monitor bits, for example, (contained in the access-control byte) are discarded. Token Ring's frame status bits (contained in the byte

following the ending delimiter, which follows the data field) are treated differently depending on the bridge manufacturer. Some bridge manufacturers simply ignore the bits. Others have the bridge set the C bit (to indicate that the frame has been copied) but not the A bit (which indicates that the destination station recognizes the address). In the former case, a Token Ring source node determines whether the frame it sent has become lost. Proponents of this approach suggest that reliability mechanisms, such as the tracking of lost frames, are better left for implementation in Layer 4 of the OSI model. Proponents of the "set the C bit approach" contend that this bit must be set to track lost frames but that the A bit cannot be set because the bridge is not the final destination.

Translational bridges can create a software gateway between the two domains. To the SRB end stations, the translational bridge has a ring number and bridge number associated with it, and so it looks like a standard SRB. The ring number, in this case, actually reflects the entire transparent-bridging domain. To the transparent-bridging domain, the translational bridge is simply another transparent bridge.

When bridging from the SRB domain to the transparent-bridging domain, SRB information is removed. RIFs usually are cached for use by subsequent return traffic. When bridging from the transparent bridging to the SRB domain, the translational bridge can check the frame to see if it has a unicast destination. If the frame has a multicast or broadcast destination, it is sent into the SRB domain as a spanning-tree explorer. If the frame has a unicast address, the translational bridge looks up the destination in the RIF cache. If a path is found, it is used, and the RIF information is added to the frame; otherwise, the frame is sent as a spanning-tree explorer. Because the two spanning-tree implementations are not compatible, multiple paths between the SRB and the transparent-bridging domains typically are not permitted. Figures 24-1 through 24-3 illustrate frame conversions that can take place in translational bridging.

Figure 24-1 illustrates the frame conversion between IEEE 802.3 and Token Ring. The destination and source addresses (DASA), service-access point (SAP), Logical-Link Control (LLC) information, and data are passed to the corresponding fields of the destination frame. The destination and source address bits are reordered. When bridging from IEEE 802.3 to Token Ring, the length field of the IEEE 802.3 frame is removed. When bridging from Token Ring to IEEE 802.3, the access-control byte and RIF are removed. The RIF can be cached in the translational bridge for use by return traffic.

Figure 24-1 Four fields remain the same in frame conversion between IEEE 802.3 and Token Ring.

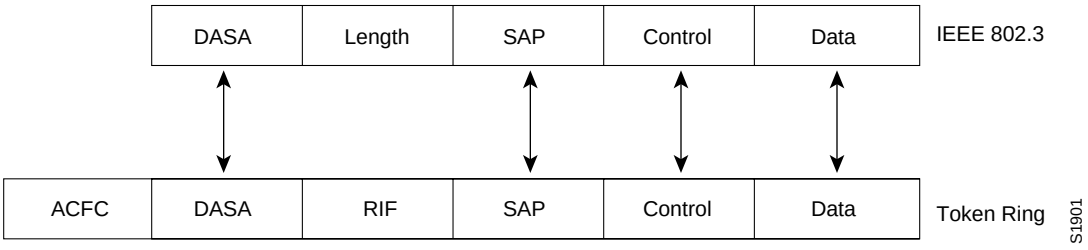


Figure 24-2 illustrates the frame conversion between Ethernet Type II and Token Ring Subnetwork Access Protocol (SNAP). (SNAP adds vendor and type codes to the data field of the Token Ring frame.) The destination and source addresses, type information, and data are passed to the corresponding fields of the destination frame, and the DASA bits are reordered. When bridging from Token Ring SNAP to Ethernet Type II, the RIF information, SAP, LLC information, and vendor code are removed. The RIF can be cached in the translational bridge for use by return traffic. When bridging from Ethernet Type II to Token Ring SNAP, no information is removed.

Figure 24-2 Three fields remain the same in frame conversion between Ethernet Type II and Token Ring SNAP.

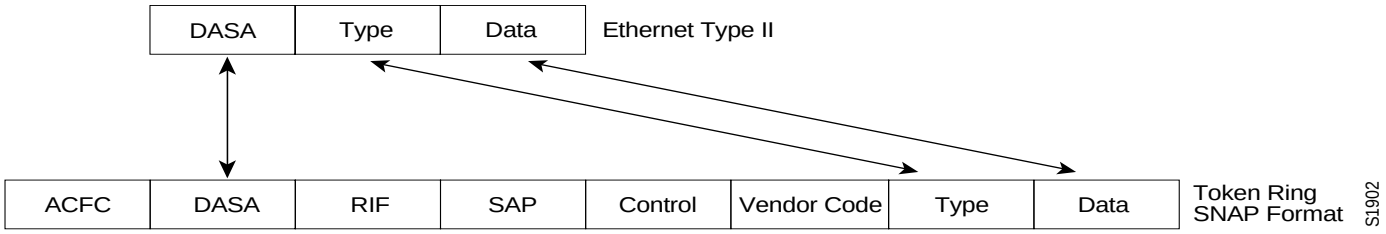
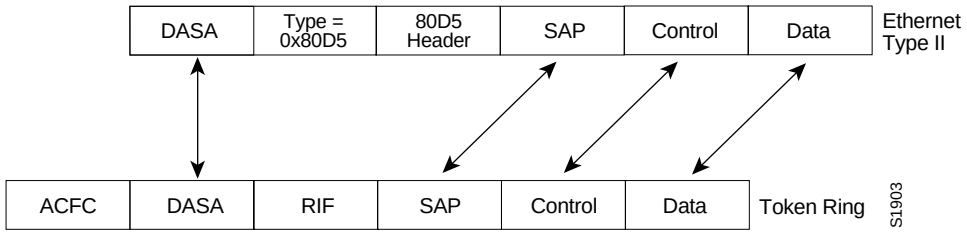


Figure 24-3 illustrates the frame conversion between Ethernet Type II "0x80D5" format and Token Ring. (Ethernet Type II "0x80D5" carries IBM SNA data in Ethernet frames.) The DASA, SAP, LLC information, and data are passed to the corresponding fields of the destination frame, and the destination and source address bits are reordered. When bridging from Ethernet Type II "0x80D5" to Token Ring, the type and 80D5 header fields are removed. When bridging from Token Ring to Ethernet Type II "0x80D5," the RIF is removed. The RIF can be cached in the translational bridge for use by return traffic.

Figure 24-3 Four fields remain the same in frame conversion between Ethernet Type II "0x80D5" format and Token Ring.



Source-Route Transparent Bridging

SRT bridges combine implementations of the transparent-bridging and SRB algorithms. SRT bridges use the *routing information indicator* (RII) bit to distinguish between frames employing SRB and frames employing transparent bridging. If the RII bit is 1, a RIF is present in the frame, and the bridge uses the SRB algorithm. If the RII bit is 0, a RIF is not present, and the bridge uses transparent bridging.

As with translational bridges, SRT bridges are not perfect solutions to the problems of mixed-media bridging. SRT bridges still must deal with the Ethernet/Token Ring incompatibilities described earlier. SRT bridging is likely to require hardware upgrades to SRBs to allow them to handle the increased burden of analyzing every packet. Software upgrades to SRBs also might be required. Furthermore, in environments of mixed SRT bridges, transparent bridges, and SRBs, source routes chosen must traverse whatever SRT bridges and SRBs are available. The resulting paths potentially can be substantially inferior to spanning-tree paths created by transparent bridges. Finally, mixed SRB/SRT bridging networks lose the benefits of SRT bridging, so users feel compelled to execute a complete cutover to SRT bridging at considerable expense. Still, SRT bridging permits the coexistence of two incompatible environments and allows communication between SRB and transparent-bridging end nodes.