the EXTENSION

A Technical Supplement to Control Network

Using Trunking To Increase Backbone Performance

By George Thomas

When cascading switches, a reduction in performance can occur if the backbone connection is not properly designed. This issue addresses the concerns involved when cascading switches.

With repeating hubs, all stations on the network occupy the same collision domain and obey the rules for arbitrating access to the network. This is called shared Ethernet since all stations share the same media including the repeaters residing within the collision domain. No one station has precedence over another station. Even when repeating hubs are cascaded, there is no perceivable change in network performance since arbitration rules do not change.
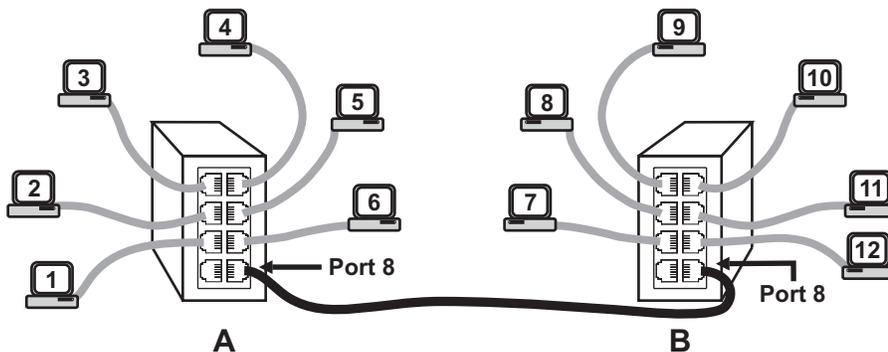

Figure 1. A possible bottleneck is created when two switches are cascaded.

However, the introduction of switch technology changes everything. Switch ports terminate collision domains allowing for increased distances over what can be achieved using repeating hubs. Traffic can be restricted to certain ports once the switch learns the location of station addresses. Switches have what is called a "switch fabric" that allows for the rapid transfer of data frames from port-to-port within the switch. A switch is called "non-blocking" or "wire-speed" if the switch fabric is fast enough, so that there is no noticeable degradation in throughput with the switch present or absent. For example, we have an eight-port switch and six connected stations all operating at 100 Mbps. It should be possible for all stations to communicate to one another as if the switch were not there. If that is the case, the switch is said to be non-blocking.

What happens if we want to add six more stations to the network? We would either need to replace the eight-port switch with a 16-port switch or we could simply add another eight-port switch using a switch-to-switch connection. Is there a difference in performance between the two approaches? Assume that both the 16-port and eight-port switches are non-blocking. This must mean that the 16-port switch fabric has higher performance over the eight-port switch fabric in order to accommodate twice as many ports in the same time frame. However, to the user there is no change in performance when twelve stations are each connected, each to a port on the 16-port switch.

What happens when these same twelve stations are split into two groups of six with six stations connected to one eight-port switch and another six connected to the other eight-port switch? (Figure 1) A single cable connects the two switches together for a net loss of two ports. Since both switches are non-blocking, there should be no change in performance. However, this is not the case. Assume port 8 on each switch is dedicated to the "backbone" connection linking the two switches together. Further, assume that stations 1 to 6 are on switch A and 7 to 12 are on switch B. For station 1 to send a message to station 12, the traffic must go through port 8. The same is true for any message originating from a station on switch B, attempting communication to a station on switch A. Therefore, port 8 theoretically handles half the traffic assuming equal distribution of messages. Port 8 becomes the bottleneck as frames are queued for transmission. The throughput is constrained by the data rate of port 8. This is not the case with a single non-blocking switch handling all the traffic since no one port has concentrated traffic. (Figure 2) The only exception would be if one of the ports was very popular by being connected to a centralized file server or to a master controller. Assuming equal distribution of traffic, a single switch arrangement is superior to a cascaded switch arrangement.

Solutions to the Bottleneck Problem

Ideally, it would be nice to size the switch according to the number of stations on the network. If there is going to be 16 stations, select a 16-port switch. If
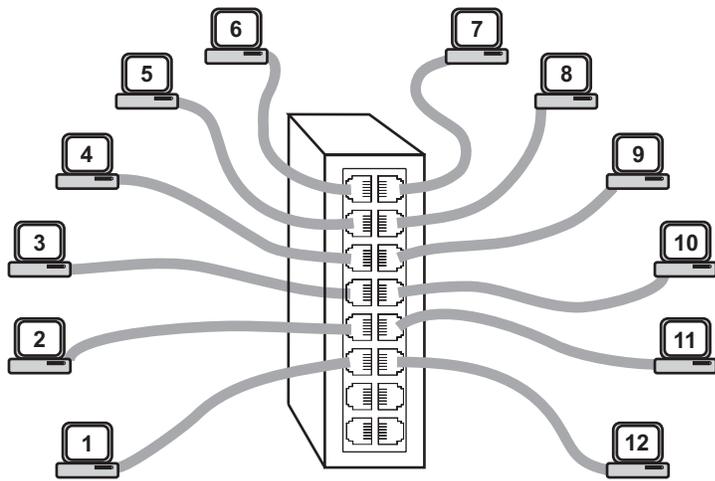
Figure 2. Wire-speed can be achieved if all stations are connected to a single switch.

there are going to be 20 stations, select a 24-port switch. Of course, there are limits to this approach. A network requiring 28-stations would require a less popular 32-port switch. Even if one is found, there are other issues. A single switch operating at 10, 100 or 1000 Mbps with twisted-pair ports, can only have segment lengths of 100 m each. This means that the network diameter is only 200 m. This distance limitation might be a constraint to the application. Fiber optics would help greatly in increasing network diameter; however, there may only be a limited number of fiber optic ports available on one switch.

The second approach is to have a higher-speed backbone port to directly address the port bottleneck issue. This port would have ten-times the data rate of the other ports dedicated for stations. However, to be fully effective, two backbone ports would be required on each switch. Two ports allow for the daisy-chain connection of more than two switches. One backbone port is fine for end of line applications, but two backbone ports are needed for mid-span applications. (Figure 3) The question then comes up: Is it better to standardize on a single two-port model or allow for two models—one-port for end of line and a two-port for mid-span? The cost of the two-port model must be weighed against the flexibility of having one model to fit all applications.

If all the stations were 10 Mbps devices, it would make sense to have 100 Mbps backbone ports. However, many of the devices today are capable of 100 Mbps operation and switch ports usually handle 10/100 Mbps selection through the auto-negotiation protocol. Even if a device connects at 100 Mbps, it does not mean it will be swamping the switch with traffic. The traffic from the device may be in spurts which would not burden the backbone ports. Therefore, the backbone ports could also be 100 Mbps. However, to ensure the greatest throughput, the station ports should be 100 Mbps and the backbone ports should be 1000 Mbps. This could be an expensive overkill in some applications. Because 1000 Mbps operation brings with it its own set of issues. The interface is complex, less robust and distance limited and for most control applications unnecessary.

There is still another approach to this problem and it is called link aggregation. With link aggregation there is a compromise between requiring a 10-times performance improvement in backbone port-speed versus port-usage. Plus, there are added benefits as we will see later.

Link Aggregation

The IEEE 802.3ad standard calls it Link Aggregation, but it is simpler to call the concept Trunking. We will use the words interchangeably. If one channel has the capability of sending data at 100 Mbps, why not add another channel to achieve 200 Mbps? If two channels can achieve 200 Mbps collectively, why not add two more channels to achieve 400 Mbps? This is the argument behind link aggregation or trunking. Standard ports on a switch would be configured as a trunk group in order to send data to a distant switch, also configured with a trunk group over a parallel path. The concept of sending data in parallel is not new and is the basis for the 1000BASE-T physical layer. With Gigabit Ethernet, symbols representing a data byte, are sent over four twisted-pairs in order to increase throughput without a significant increase in baud rate which would therefore limit segment length. The four pairs represent a parallel path, although to the user it appears that only one twisted-pair cable is being used. With trunking, there is no change in the physical layer interface. The ports used for trunking are no different, physically, from any other port on the switch. A separate cable is used for each trunk segment. Complete frames are alternately sent down
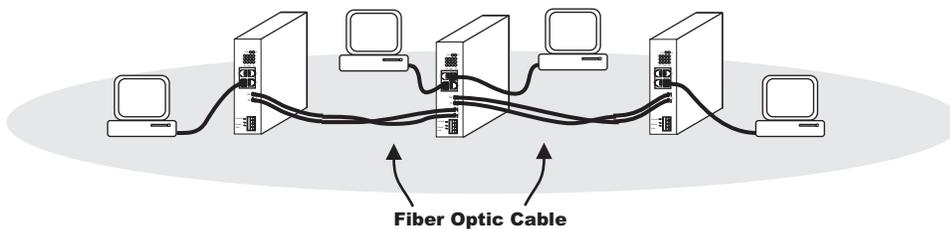

Fiber Optic Cable

Figure 3. Mid-span switches require two backbone ports while end of line switches only require one. In this example, fiber optics are used for the backbone.

each trunk segment and recombined at the other end. In the case of a 100 Mbps switch, the trunk ports are also 100 Mbps. It must be remembered that the switch used must have the capability for supporting trunking. A regular switch will not work. In fact, establishing a parallel path using standard switches will disrupt the network.

Assume two or more ports are assigned to a trunk group within a switch and the same is done to a second switch. The trunk group will act as one high performance port by sending frames to ports in the trunk group which are available. This usually results in frame transmissions being alternated between the ports within a group. This increases throughput since multiple channels are available for transmitting. On the receiving side, the frames received by the trunk group are treated as if they came from one port.

Advantages of Trunking

Trunking provides an incremental increase in backbone performance by simply assigning more ports to the trunk group. The alternative is to use one backbone port with a data rate that is ten-times faster than the station ports. That is a significant increase in performance that may not be needed. A two-times increase in speed may be all that is required which can be accomplished by simply assigning two 100 Mbps ports to a trunk group. If another increment in speed is desired, another port can be added to a trunk group. Therefore, backbone speed can be adjusted by the simple addition or removal of ports within a trunk group.

The other advantage is distance. By using regular ports for trunking, there are no changes to the cabling rules. Although a 100BASE-TX has the same 100 m segment length restrictions as a 1000BASE-T port, that is not the case for fiber optics. Both single-mode and multimode fiber optic maximum segment lengths are much less at Gigabit speed than they are at 100 Mbps.

While single-mode segment lengths of 15 km and greater are quite easy to achieve at 100 Mbps, lengths of only 5 km are possible at Gigabit speed. Segment length for multimode fiber is restricted to only 550 m at Gigabit speed using either longwave or shortwave devices. At 100 Mbps, 2 km can be achieved using longwave devices. Therefore, there is an advantage to using trunking when long distances are involved.

One might say that the cost of a trunk would exceed the cost of a single high-speed connection. Granted, more fiber pairs are required for a trunk, but fiber is generally pulled in bundles to ensure that spare fibers are available. The cost of pulling a fiber cable, regardless of the number of fibers in the bundle, will probably outweigh the cost of the fiber cable used.

There is another advantage to trunking and it is in regard to availability. With a trunk group, transmissions are sent to the first available port within the trunk group. What happens if the cable to one of those ports is defective? With many switches, the port that is no longer available is ignored and therefore bypassed. Traffic will be diverted to the remaining ports within the trunk group. Granted, throughput will be reduced but the network remains functional. With a single high-speed backbone, a lost of the single cable will sever the system.

I like to use the following analogy when discussing availability. Lake Michigan, which Chicago skirts, is a very large lake which many power boaters like to challenge. What is best to have—a boat with a single 150 HP inboard motor or a boat with two, 75 HP out-board motors. The inboard model is nice and sleek, but the outboard model may be more practical. A motor failure on the inboard leaves you "dead in the water;" however, a single motor failure on the dual-outboard model will still get you back to shore with reasonable swiftness. The availability of a second motor ensures continuing operation similar to the additional ports within a trunk group.
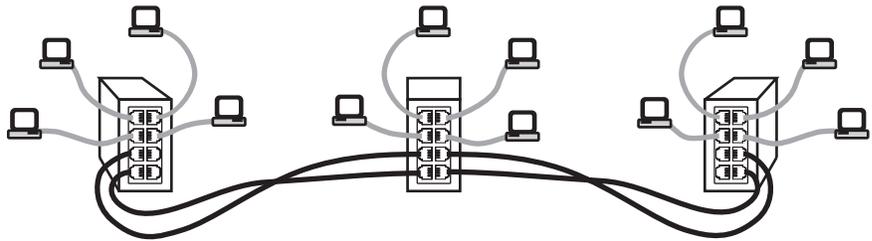

Figure 4. Trunking consumes ports especially in mid-span positions.

Disadvantages of Trunking

The first disadvantage of trunking is that trunk ports must first be configured. This means that the switch cannot be a simple plug-and-play switch. Either a configurable or managed switch must be used with a trunking option. This is not the case with having high-speed backbone ports. With high-speed backbone ports, it is possible to simply have a plug-and-play switch.

The other disadvantage is loss of ports. With a high-speed backbone port, only one port is lost for this function. If it is a mid-span device, two ports are lost. (Figure 4) With trunking, a minimum of two ports are lost for backbone use and with mid-span devices, four ports are lost. If the trunk group is larger than two ports, even more ports are lost. Lost ports mean fewer ports are available on the switch for actual stations.

Take the example of an eight-port switch. If two ports are dedicated for trunking, only six ports are available for stations. If this same switch is located in a mid-span position with other switches connected to its left and right, four-ports would be dedicated

for trunking and only four ports remain for stations. If the trunk group were increased to four, no ports would be available for stations. Trunking is best applied to switches that have sixteen or more ports.

Poor Man's Redundancy

As mentioned before, trunking improves availability by providing a redundant path in case there is failure of one of the cables or ports within the trunk. However, this feature does not provide true cable redundancy since throughput is lower when one of the connections fail. Still, for many applications, this may not be a problem if the application can remain functional even at the reduced throughput. Trunking offers cable redundancy without the complexity of other redundancy schemes.

One of the more popular redundancy schemes is the fiber optic ring. This is a proprietary scheme where a ring topology is created by the backbone connection of compatible switches. All switches that function in the ring have two backbone ports and all are used because the left-most and right-most switches are connected through a redundant connection to form a ring. This cannot be done with ordinary switches. With the fiber ring, endless communication around the ring is prevented by the use of a ring manager (positioned as one of the switches in the ring) which also verifies that the ring is intact. If the ring is broken, communication is re-routed around the break. There is a reconfiguration time that must be observed during recovery and the address table in each of the switches within the ring must be cleared, requiring all switches within the ring to re-learn the new topology. With all switch memory cleared, the switches will function basically as repeating hubs and therefore, throughput will be impacted until all switches re-learn the location of stations. Another problem with this approach is that of the ring topology itself. Plant layouts may not be conducive to ring topology. If the left-most switch is at one end of the plant and the right-most at the other far end, a redundant link must be established between these two switches. This run may be beyond the segment limit of the switches so care must be exercised when locating switches within the plant.

A simplified redundancy scheme can be established using trunking without the need of creating a ring topology. Let's assume a two-port trunk group in each switch and multiple switches. The end switches do not require a redundant path that would create a ring. (Figure 4) Instead, cable redundancy exists between any pair of switches. If a cable fails, communication continues. Of course, it is best to run the redundant cables over different paths to guard against an accident like a severed cable tray or conduit. Do not put both cables in the same tray or conduit.

Cable Fault Annunciation

The expectation of a redundant system is as follows. With a single failure, the system continues to operate (although possibly at reduced performance) while identifying the source of failure. With a second failure, the system will fail to work. What must be done with trunking is to identify the source of failure which is usually done by observing the link integrity at each port.

One nice feature of Ethernet link segments (twisted-pair and fiber optics) is that each port on a hub, switch or station supports the link integrity function. This is true for both 10 and 100 Mbps ports. A functioning link is continuously checked by circuitry, observing a link pulse sent by each transmitter on a functioning link. Loss of link indication could mean a cable fault or port fault. By observing the link status of each port within a trunk group, it is possible to determine if one of the connections has failed. Automatic acknowledgement can occur if the switch has a programmable relay output or supports the SNMP protocol. With the SNMP protocol, a trap can be set in the switch that is tripped when loss of link is detected on one of the ports in the trunk group. Some configurable or managed switches can provide this functionality. An alarm can be programmed to occur upon lost of link although communication continues. Plug-and-play switches cannot provide this functionality.

SUMMARY

Trunking provides an incremental improvement in backbone speed. While a switch with a high-speed backbone can provide a ten-fold increase in speed, many applications do not require this level of speed improvement. Trunking utilizes standard ports on the switch which in some cases can provide longer segment distances than high-speed backbone ports. A simple cable redundancy scheme can be implemented with trunk groups which is less complex than the fiber optic ring scheme while providing adequate protection against a single cable fault.

REFERENCES

The Switch Book, Rich Seifert, 2000, Wiley Computer Publishing

International Standard ISO/IEC 8802-3 ANSI/IEEE Std 802.3, 2000, The Institute of Electrical and Electronic Engineers, Inc.

CONTEMPORARY CONTROLS®
www.ccontrols.com

Past issues of the Extension are available. If you would like a copy, please send your request to **info@ccontrols.com**