Ethernet Jumbo Frames

The Good, the Bad and the Ugly

A Chelsio Communications White Paper

Abstract

A decade has passed since jumbo Ethernet frames were first proposed, and an IEEE standard or significant deployment are yet to be seen. This paper synthesizes the various reasons behind the lack of acceptance of jumbo frames. First, performance considerations show that jumbo frames are only useful for bulk data transfer, and may adversely impact latency sensitive applications, thereby jeopardizing the Ethernet promise of a converged network. Second, the problems faced in integrating jumbo frames within the framework of other standards are discussed, including considerations related to the pervasive TCP, as well as the existing Ethernet infrastructure. The latter effectively preclude the standardization of large frames, and therefore perpetuate the issues due to the lack of a standard. Finally, the findings of experimental studies on actual Internet links show that practically none of the links tested supported jumbo frames, preventing their use outside of a controlled local area network environment.

Introduction

Many would argue that, today, Ethernet is on the verge of becoming the sole networking and inter-connect fabric, driving the convergence of data and multimedia (voice and video) communications, storage attachment and high performance computing. However, perceived performance shortcomings seem to be standing in the way of this long desired unification. A solution to these problems, in the form of jumbo frames, is claimed by various networking vendors and researchers.

Jumbo Ethernet frames are ones which are larger than the maximum standard frame size of 1,522 bytes (with VLAN tag), typically up to 9,180 bytes. The rationale behind increasing the frame size is clear when considering the high processing cost of network packets: larger frames reduce the number of packets to be processed per second. Note that this observation mainly applies to the end systems; network switches and routers typically are capable of operating at line rate with frames sizes that are much smaller than the maximum standard of 1522 bytes.

Jumbo frames are no new idea, having been around for more than 10 years. Indeed, the calls for using large frames in Ethernet systems grow loud each time the technology moves up in speed. The reason being that Ethernet's speed step is one order of magnitude at a time. Therefore, the network processing load tends to outpace CPU speed advances, leaving the new links only partially used for a period of time. This was the case when Gigabit Ethernet was rolled out in the mid 1990s and, today, with the recent deployment of 10 Gbps Ethernet networks, new calls for jumbo frame deployment are being heard. Admittedly, the CPU performance gap today looks particularly severe: even the best performing CPU on the market cannot fill half a 10 Gbps link when using standard frames.

In order to address this gap and leave some cycles for useful application work, large frames on the wire may appear to be a good idea at first sight, simple enough to implement: just increase the payload size! It turns out, however, that the use of jumbo frames introduces a plethora of issues which complicate the deployment process and could negate the pre-supposed benefits. In fact, the sheer number of potential problems and concerns have prevented this seemingly simple change from gaining popularity. Add to that higher network equipment cost associated with the adequate support of larger frames, and the reasons for the minimal levels of deployment become apparent.

However, it appears that the many known reasons for this failure seem to be lost to those who today again consider using jumbo frames. This paper presents a list of these known issues, which explain why jumbo frames are not a valid solution to today's Ethernet performance woes. It also suggests a demonstrably effective solution, in the form of performing the networking processing in hardware on the network adapter.

Jumbo Frames and Performance

Since the primary reason for deploying jumbo frames is the need for higher performance, we consider this aspect first.

There is no denying that the per-packet protocol processing costs associated with bulk data transfer are reduced when using larger packets. Back to back connected NIC benchmark tests indeed show that the reduced processing load allows today's systems to send and receive bulk data at 10Gbps when using 9,000 byte frames (see for example [FENG]). Using standard frame sizes, however, a high end system would barely achieve half of that, while fully utilizing the CPU and leaving no cycles for useful application processing.

Does this mean that all applications will benefit from enabling jumbo frames? Surprisingly, the answer, for most applications, turns out to be negative.

Knowing that 10Gbps is destined to become a unifying switching fabric, considering the bulk data transfer application only would be rather limiting. First, not all applications perform large transfers. It should be evident that applications which exchange small messages such as database applications get no benefits whatsoever from the provision of large frame sizes. This also applies to transaction-based applications.

Furthermore, most applications of interest in the early deployments of 10 Gbps Ethernet are latency sensitive rather than throughput heavy. When these applications are considered, such as distributed grid and cluster computing or transaction oriented storage over iSCSI, it is often the case that transfer sizes are relatively small. Therefore, the node-to-node store and forward delays make up a large part of the total transfer time, directly affecting execution time. In this context, the use of jumbo frames results in increased node-to-node delay due to limited pipelining (i.e., the limited overlap of transmissions over successive links). Figure 1 illustrates this simple idea, where the time it takes to transfer a jumbo frame from a source to a destination station, with no intermediate switch, is compared to the time it would take to transfer the equivalent amount of data split into standard frames.

In fact, it is often claimed that the transfer time of a 9,000 byte frame on a 10 Gbps link (i.e., 7.2usec) is short enough to be of no concern. Let's take another look at this claim. The transfer time is actually incurred several times between the source and destination, even when the two stations are directly attached: the frame needs to be transmitted over the PCI bus which typically is slower than 10 Gbps (e.g., 8Gbps for PCI-X at 133MHz), then transmitted over the link, and transmitted again over the PCI bus. The combined delay is now about 25 microseconds, not including any other delay components. For a message size of 16KB, the total delay is more than double the intrinsic transmission time of the message. On the other hand, using standard frames, which allow better pipelining, the total delay would be only slightly larger than the actual transmission time of the message (i.e. 9.6usec in this case). The extra delay in the jumbo frame case potentially represents a significant drop in compute power.

It is a straightforward observation to make that the numbers for jumbo frames get worse as the number of store and forwards increases, such as by going through intermediate switches. In addition, considering the same scenario in a network containing Gigabit links, let alone 100 Mbps or 10 Mbps, makes it clear that application performance may not benefit from jumbo frames as it was thought to do. In fact, it may very well deteriorate.

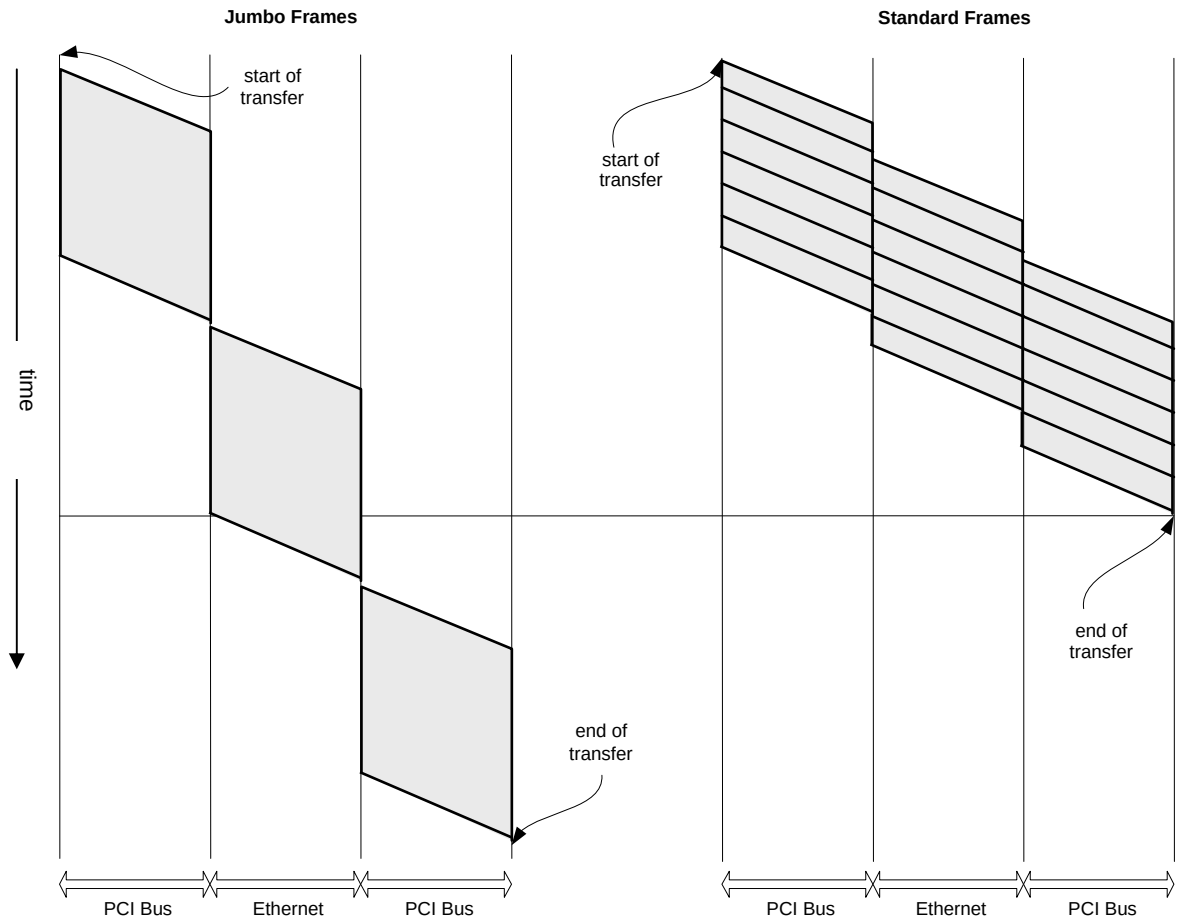

Figure 1 Weak pipelining increases end-to-end delay for jumbo frames

Finally, in a converged network, latency sensitive applications such as voice, will share the infrastructure with storage, bulk transfers and other applications. In order to satisfy the low latency requirements in the presence of jumbo frames it becomes necessary to upgrade all links to be Gigabit or faster, and to add Quality of Service functionality in switches to give priority to such applications with pre-emption of low priority traffic. This hinders the ability of Ethernet to drive the convergence of the network, which capitalizes on the extension of the installed base rather than a complete overhaul of the infrastructure.

Another aspect worth mentioning is that due to their larger size and weak pipelining, jumbo frames are more likely to be dropped due to limited buffering resources in switches compared to standard size frames. The increased drop rates would translate into much reduced performance for TCP traffic, which is particularly sensitive to packet loss at high speeds.

Jumbo Frames and Standards

One of the main impediments to the deployment of jumbo frames has been the lack of standardization. This in turn is due to concerns about their compatibility with the existing and ubiquitous Ethernet standard. An obvious issue is that the Type/Length field which distinguishes Ethernet II and IEEE802.3 formats does not support sizes in excess of 1,536 bytes, thereby limiting their use to pure 802.3 network types. This is not the only restriction, as we will see below.

First, it turns out that the standard frame size has been a significant aspect in the design of Ethernet, quoting the chair of the IEEE 802.3 (Ethernet) working group (as quoted in [THOMPSON]):

> The expectation of no more than 15-1600 bytes between frames and an interpacket gap before the next frame is deeply ingrained throughout the design and implementation of standardized Ethernet/802.3 hardware. This shows up in buffer allocation schemes, clock skew and tolerance compensation and fifo design.

This basically places a big question mark on the inter-operability with deployed networking gear. There are more issues to be addressed when considering the standards aspect, again quoting [THOMPSON]:

> For some Ethernet/802.3 hardware (repeaters are one specific example) it is not possible to design compliant equipment which meets all of the requirements and will still pass extra long frames. Further, since clock frequency may vary with time and temperature, equipment may successfully pass long frames at times and corrupt them at other times. Therefore, attempts to verify the ability to send long frames over a path may produce inaccurate results.

This essentially means that existing 10/100 Mbps networks do not support jumbo frames, limiting their use to Gigabit and above speeds. The main issue becomes the lack of assurance for inter-operability, and its effect on a basic value of Ethernet, which is uniformity and standardization:

> The huge value of Ethernet/802.3 systems in the data networking universe is their standardization and the resulting assurance that systems will all interoperate. No such assurance can be provided for oversize frames with both the current broadly accepted standard and the large installed base of standards based equipment. In summary, with regard to greatly longer frames for Ethernet, much of the gear produced today would be intolerant of greatly longer frames. There is no way proposed to distinguish between frame types in the network as they arrive from the media. Bridges might and repeaters would drop or truncate frames (and cause errors doing so) right and left for uncharacterized reasons. It would be a mess. What might seem okay for small carefully characterized networks would be enormously difficult or impossible to do across the Standard.

Therefore, the essential requirement of backwards compatibility will most likely preclude the violation of the existing standards in the interest of the claimed performance improvements provided by jumbo frames. This is especially true given that these are yet to be proven beyond a specific application and a particular network setup.

Another serious consideration associated with using jumbo frames is the fact that the "error checking mechanism embodied in the 4 byte [...] CRC is known to degrade at greater frame lengths [THOMPSON]. For instance, [JAIN] shows that the CRC loses its ability to catch all 3 burst errors for frame sizes exceeding 1,553 bytes.

The lack of a standard does not mean that jumbo frame capable equipment is not available. Rather, it means that the different equipment may handle different maximum frame sizes (i.e. equipment vendors may validly claim support of jumbo frames, referring to a maximum of 2450 bytes [SAUVER]). This proves to have a critical impact on the effective use of jumbo frames for TCP transfers over different networks, as discussed below.

The vast majority (95%) of Internet traffic is carried by the Transmission Control Protocol. In order to use large datagrams (i.e. larger than 576 bytes including IP and TCP headers), the

maximum segment size (MSS) needs to be negotiated between the two end points of a connection at connection establishment time. Note that the MSS is equal to the underlying network maximum transfer unit (MTU) minus the IP and TCP header sizes (typically 40 bytes). In today's network, when two endpoints desire to utilize the standard Ethernet 1500 byte MTU, it is usually sufficient that their respective networks support it. This benefit follows from the fact that most Internet gear supports this value. Otherwise, end-to-end path MTU discovery would be needed. Since jumbo frames are non standard, this process is required, and turns out to be a major obstacle. To understand the reasons for this difficulty, recall that path MTU relies on ICMP error messages, which are returned by intermediate routers as the endpoints probe the path with large packets, decreasing the size until they find one which is supported all the way to the destination. First, in the Internet, this process is required in both directions since routing is not guaranteed to be symmetric. Second, a number of denial of service attacks have involved sending ICMP packets, and these are therefore filtered at the boundary of many networks. Finally, the problematic equipment may be a layer 2 switch, which does not respond to ICMP packets [MATHIS, RUTHERFORD, SAUVER]. This renders the debugging of connectivity problems very difficult.

However, the problems of using jumbo frames with TCP are not limited to connectivity. Using large MSS values results in more aggressive TCP behavior since the minimum window size and each window increase step in Slow Start become equivalent to 6 standard sized frames. While this may translate into better single stream performance over a dedicated path, in the more realistic case of multiple connections sharing network links, the resulting burstiness may lead to increased congestion and packet loss. Again, packet loss translates into disproportionately low TCP performance, increased by the need to retransmit larger amount of data for each lost segment.

Finally, large MSS values mean that the likelihood of Nagle's algorithm holding the transmission of successive writes because they are smaller than 1 MSS increases. As a result, a known interaction of Nagle with delayed ACK may cause severe performance penalty for non-bulk transfers (e.g., request-response) applications when using jumbo frames. A related requirement is the need for socket buffers which are significantly larger than today's defaults, in order to avoid similar dynamics in the interaction between the receive window sizes and the MTU [FARRELL], and other effects due to the Silly Window Syndrome Avoidance schemes.

In summary, the interaction of the many TCP mechanisms with large frames beyond the bulk transfer application is not well understood, and there are reasons for concerns regarding potential negative impact on performance.

Jumbo Frames and Real Life

In this section, we turn to the more practical question of how easy is it to enable jumbo frames. While it may seem that setting the MTU for a network adapter is sufficient to use jumbo frames (assuming the operating system supports it), the reality is a lot more complex.

In a controlled network environment, purchasing equipment with identical jumbo frame support may allow the use of large frames. However, such equipment is typically much more expensive than standard-only equipment [SAUVER]. The real problems surface as soon as one considers the Internet at large. Several studies of jumbo frame support in the Internet came back with the same conclusion: most paths do not support them [RUTHERFORD, SAUVER]. The fact that some paths and equipment support one maximum jumbo frame size or the other only serves in increasing the confusion.

Conclusion

The Ethernet situation today may seem critical. Jumbo frames have failed to gain sufficient traction to make them a universally usable approach. CPU speed increases appear to be, for

the first time, lagging severely behind the network speed. While hacks such as LSO and LRO may appear to offer relief, they are effectively contained in a limited sphere of applicability: a particular setup (e.g., directly connected hosts) and a particular application (e.g., bulk transfer). What is needed is a solution which scales across number of connections, transfer sizes and application types (both latency sensitive and throughput demanding). This solution is in hardware offload of network processing, as implemented in Chelsio's TCP Offload Engine (TOE).

Chelsio's TCP offload engine architecture provides cut-through low latency processing for latency sensitive applications, in addition to allowing high throughput applications to send payload in large chunks to the hardware, which uses standard frames on the wire. This achieves the goal of reducing host CPU utilization while preserving the compatibility with standard Ethernet gear. The offload engine was demonstrated to provide superior performance to normal network adapters using jumbo frames by shattering and holding the Ethernet speed record on one hand, and pose serious competition to a specialized inter-connect fabric in cluster computing application on the other [FENG, PANDA]. In summary, *the standards-based TOE obviates the need for non-standard jumbo frames for throughput demanding applications, while providing the required low latency for delay sensitive ones.*

For more information about Chelsio Communications and the Terminator architecture, visit the Chelsio web site at www.chelsio.com or send an e-mail to info@chelsio.com.

References

[THOMPSON] Email from Geoff Thompson, chair IEEE 802.3, to Scott Bradner, director Transport Area of the IETF, cited in J. Kaplan http://www.ietf.org/proceedings/01aug/I-D/draft-ietf-isis-ext-eth-01.txt

[JAIN] R. Jain, "Error Characteristics of Fiber Distributed Data Interface (FDDI)," IEEE Transactions on Communications, Vol. 38, No. 8, August 1990, pp. 1244-1252

[MATHIS] M. Mathis, "Arguments About MTU", Web page, http://www.psc.edu/~mathis/MTU/arguments.html

[FENG] W. Feng et al., "Performance Characterization of a 10-Gigabit Ethernet TOE", in Proceedings of Hot Interconnects, August 2005.

[BALAJI] P. Balaji et al., "Head-to-TOE Evaluation of High-Performance Sockets over Protocol Offload Engines", Technical Report, Los Alamos National Laboratory (LA-UR-05-4148) Ohio State University (OSU-CISRC-5/05-TR35)

[RUTHERFORD] Rutherford Research, "Core network 9000 byte (9k) maximum transmission unit (MTU) research", http://www.rutherford-research.ca/rrx/network/internet2MtuProject.php

[SAUVER] J. St Sauver, "Jumbo Frames Presentation", http://darkwing.uoregon.edu/~joe/jumbos/jumbo-frames.pdf

[FARELL] P. A. Farrell, H. Ong, "Communication Performance over a Gigabit Network", 19th IEEE International Performance, Computing, and Communications Conference -- IPCCC 2000.