S-GES: Stacked Gigabit Ethernet as SAN

Zheng Zhang
Computer Systems and Technology Laboratory
HP Laboratories Palo Alto
HPL-2001-82
March 30th , 2001*

E-mail: zzhang@hpl.hp.com

Gigabit
Ethernet, SAN
(Storage Area
Network)

* Internal Accession Date Only Approved for External Publication

1. Introduction, Motivation, Excuses and Whatever..

The trend towards SAN-based, remote-attached I/O subsystem is occurring industrial wide, of which HP is an active participant. We can call this as "*operational paradigm shift*", since embodied in this trend is fundamental change of I/O semantics as well as new functionality requirements (which may or may not be shared by other companies). Yet the model is still computing centric, with I/O acting as peripheral to the multiple-hierarchies processor/memory complex. The time hasn't been ripe for I/O to reach a functionally higher level, it seems. These two steps of paradigm shift are shown in Figure 1.

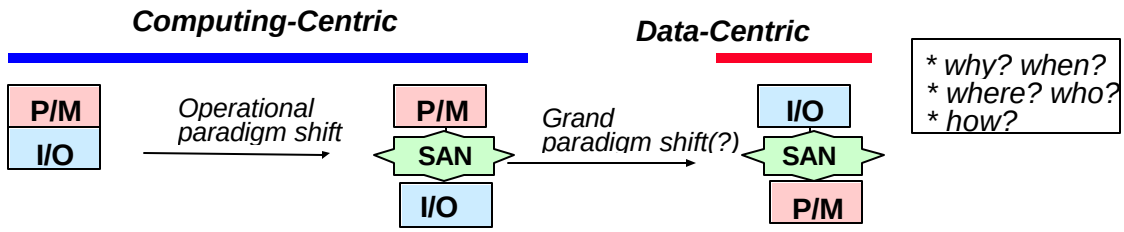

Figure 1: The Paradigm Shift

Steps involved to make this shift into reality are many. For example, define the new semantics and functionality, select a message-passing protocol and extend it to meet the needs, and integrate it with both the host and device ends. The other side of the coin is to find a suitable SAN which I/O messages will be riding on. And that is the focus of this report.

We had casual talks with Andrew Thomas from Bristol and Rajiv Gupta at CSL. The theme was whether is it possible to deploy commodity networking technologies, such as Gigabit Ethernet and/or ATM, and make them as SAN alternatives. The argument seems quite compelling, since as long as performance and reliability requirements are met, there is no reason why we shouldn't at least consider that – the large volume placed by commodity players will help us to enjoy a much better cost performance factor.

Stacked Gigabit Ethernet as SAN, or S-GES for short, is what we tried to look into here. We've ruled out ATM as one candidate, for technical reasons not to be discussed here. The S-GES proposal is *not* a "take it or leave it" one, it is simply hoped that even if itself can not be materialized, some of the ideas maybe helpful for other SAN deployment. True enough, at the writing of this report, many technical details have not even been thought about, so the plea is on to solicit your critiques. Likewise, we hope people won't dismiss it offhand for lack of technical merits, for things can always happen.

Now let's move on.

Section 2 discusses the basics of Gigabit Ethernet, it is recommended for those who are unfamiliar with the technology; it is even *more* recommended for those who knows Ethernet but not the *Gigabit Ethernet*, because there are fundamental changes from Ethernet to Gigabit Ethernet (although it is said to be *evolutional*, my personal take on this is that from a SAN perspective it is not). Section 3 outlines how we are going to use Gigabit Ethernet as a SAN alternative. Specifically, HP's deployment at a very crude level will be discussed. Some prior knowledge of message passing protocols such as ST (Scheduled Transfer) and VIA (Virtual Interface Architecture) will

be helpful. It also iterates over why system-wide reliability can be addressed with a sender-based protocol. Section 4 goes over some basic attributes that maybe offered by S-GES. In particular, it presents an algorithm for HA via redundant fragment and discusses QoS possibilities. Section 5 discuss what are still missing in the S-GES and other proposals that take the similar approach. Finally, Section 6 compares S-GES with other SAN alternatives, and Section 7 concludes the report.

2. Gigabit Ethernet Basics

From the OSI model, Ethernet defines the bottom two layers: the physical link layer and the data link layer. Ethernet provides connectionless networking services and the transport unit is called *frame*. Each devices attached to Ethernet has a 48bits MAC address, and the frame that is put on the wire includes both the source and the destination MAC addresses. Traditionally Ethernet is deployed as a shared LAN, where each Ethernet NIC listens and delivers the frame to upper layer protocols if it is the destination of the transfer, and discard otherwise. The data payload can carry up to 1500 bytes of data. The Fast Ethernet is capable of 100Mb/s bandwidth. The MAC control of Ethernet is *Carrier Sense, Multiple Access with Collision Detection* (CSMA/CD in short). Each device, when transfer a frame, will try to detect collisions on the wire, and if there is one then set to transfer at a later time. Because of this MAC algorithm, there is a minimum requirement of frame size, so that the device can sense collision when it is still transmitting. This also puts a limit on the physical length of the wire.

Ethernet can be networked together by the use of bridges and switches, which transparently transfer frames across LAN segments. The primary delay for Ethernet are composed of the wire propagation delay, the switch delay, and the so-called interframe delay. All these add together maybe in the order of tens of microseconds (not exactly sure!). The loss of frames on Ethernet are collisions on the wires, which is especially true for *half-duplex* configuration where frames destined to a device can run into the frames transmitted from it. They can also be the results of insufficient buffer space at the switch.

Gigabit Ethernet is *evolutional* in the sense that it keeps the original frame semantics, and retains full compatibility to its predecessors from data link layer upwards. It is deployed upon the same physical medium used by fiber channel at the physical link level. By using a faster medium, Gigabit Ethernet has the maximum bandwidth of 1000Mb/s, closely matching that of today's PCI-1X. Furthermore, Gigabit Ethernet never uses shared LAN. For half duplex configuration, each end device will attach to a repeater hub, which broadcast frames to all ports (and thus all devices). More importantly, Gigabit Ethernet is seen mostly interesting for full-duplex and switching configuration. In this configuration, the MAC will be turned off, since there will be no collision on the wires whatsoever. The only loss of frames will therefore be the result of insufficient buffer space at the switches.

In conclusion, Gigabit Ethernet is of higher performance (by leveraging Fiber Channel's link level technology) and higher reliability (due to full-duplex switching configuration). It is more expensive, and since it is primarily targeted at the server backbone market segment, it will certainly not enjoy the rapid price decline as its predecessors. At the end, it has been projected that Gigabit Ethernet will be two or three times costly than Fast Ethernet and yet deliver 10 times better performance. Keep in mind that although the Gigabit Ethernet has a smaller target area, it is still a *commodity* parts.

3. Stacked Gigabit Ethernet as SAN

Strictly speaking, the SAN we talk here is the more for I/O than for clustering. Therefore, the latency maybe not as important an issue as bandwidth, and reliability, high-availability, manageability and interoperability can be of higher priority (the goal is to own the data center). (*What else that maybe different here?*)

3.1 Stretch Gigabit Ethernet to Meet the Challenge of SAN

For Gigabit Ethernet (and commodity networking technology in general) to be a viable alternative to SAN, it must be able to meet the performance and reliability requirements. Currently, HP's thinking of SAN is to ask for a link bandwidth around 1GB/s, that's about 8 Gigabit Ethernet links' bandwidth combined. Therefore, a natural approach is to aggregate the bandwidth of several Gigabit Ethernet links together.

Aggregate link bandwidth can have different flavors. At one extreme, links belong to the same switch can be ganged together to provide one logical "fat pipe". This is often known as *trunking*, which is what Cisco's Fast/Gigabit EtherChannel does. In particular, this is the Cisco's Catalyst 5000 9-port Gigabit EtherChannel Module, used as point-to-point switch connection to server backplane. This configuration is shown in Figure 2-(a). Obviously, while providing sufficient bandwidth, this approach suffers from limited connectivity as well as little flexibility to accommodate other topology for the purpose of SAN.

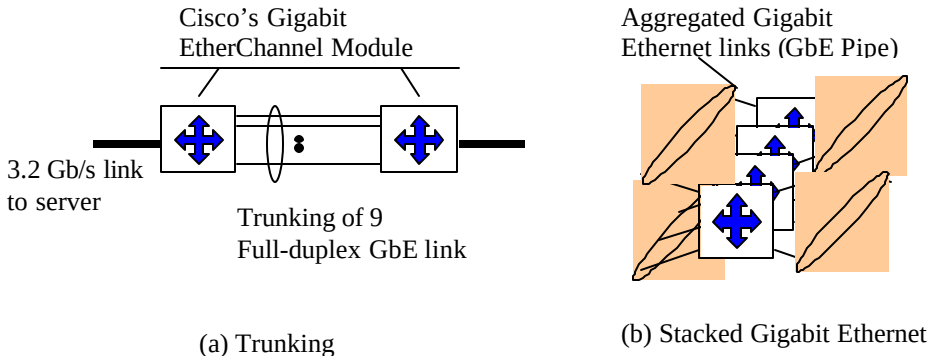

Figure 2: Trunking and Stacked Gigabit Ethernet

To enable higher connectivity, the aggregation must be done across switches. An example of this type of bundling is shown in Figure 2-(b), for simplicity only 4 switches each with 4 links are shown. This arrangement is called *Stacked Gigabit Ethernet*, and to differentiate with the EtherChannel approach, the aggregated links is termed as Gigabit Ethernet Pipe. Links of a GbE-Pipe can transmit totally independent frames, or a group of frames which are fragments of a large I/O messages. Comparing to a fat-pipe of the same bandwidth, then:

- Gigabit EtherChannel utilizes bandwidth more efficiently for small messages, but…
- Incurs fragmentation and reassemble overhead for large messages at transmit and receiving end.

The latency is contributed more from the upper layer stacks than the network itself. To reduce the latency of the network, we need to minimize end-to-end hops, which implies that the network should be of high dimension and use crossbars. For relatively small number of devices, say to support the connectivity of 16, we may be able to use multiple switch boxes directly with only one hop. For larger number of devices, we can elect to use topology such as MIN (Multi-stage Interconnect Network). These topologies are shown in Figure 3.

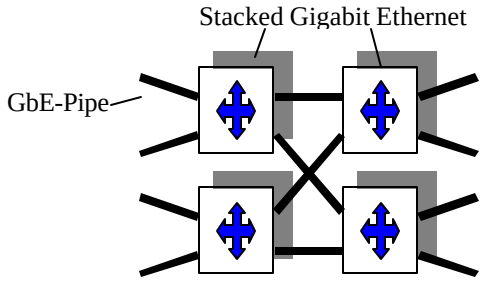

Figure 3: MIN topology of S-GES

Using Stacked Gigabit Ethernet in a full-duplex switch configuration as a SAN alternative, is what we call S-GES (***S****tacked* ***G****igabit* ***E****thernet as* ***S****AN*). The principle here is to use commodity parts to build SAN suitable for high and middle end servers' I/O.

3.2 HP's Deployment of GES

3.2.1 Hardware Attachment of S-GES

Figure 4-(a) shows the current I/O architecture for Yosemite type of systems. The REO links are full-duplex, capable of 1.2GB/s per direction. The REO chip can support up to 12 Ropes, each attach to the Elroy PCI bridge chip, each Elroy can in turn drive one 2X PCI bus. The Ropes can be aggregated together to drive 4X PCI. These PCI buses are attached to end I/O devices through, for example, Fiber Channel.

The current I/O infrastructure can be adopted to support SAN-attached I/O by directly attaching SANICs on the PCI buses, and thereby connecting to S-GES. This is shown in Figure 4. Whether each SANIC supports one Gigabit Ethernet link or several (and thereby connect to a GbE-Pipe directly from one card) is a design tradeoff. At one hand, it is conceivable that one link per SANIC may foster future opportunity to leverage commodity part further. Furthermore, this might have better high-availability characteristics. On the other hand, gang several links onto one SANIC would clearly help reducing the overhead of fragmenting and/or resembling large I/O messages, but it can be challenging to build such a card.

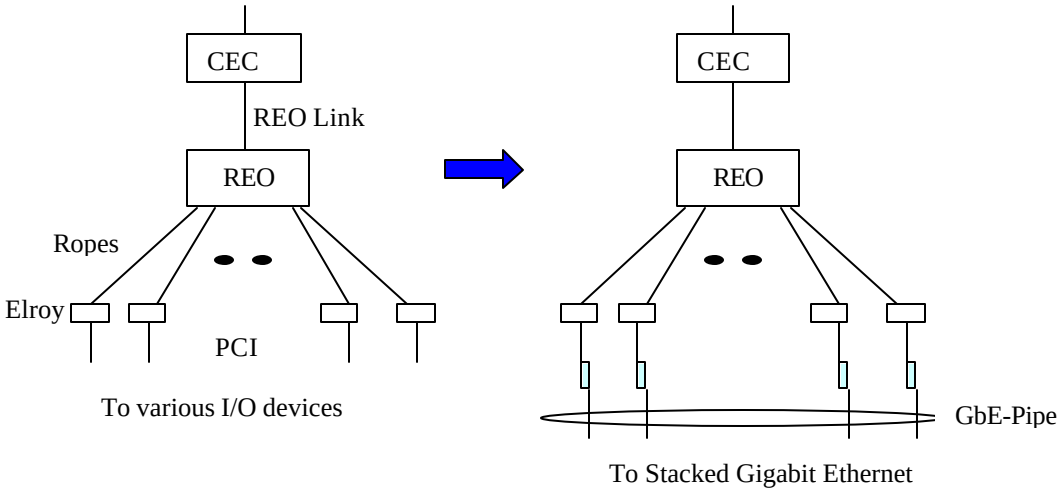

Figure 4: Hardware Attachment to S-GES

With this architecture, we can retain HP's current I/O investment (REO link, REO chip, Elroy etc.) and at the same time leverage the commodity parts via S-GES.

3.2.2 Hardware-Software Interface of S-GES

While other components of S-GES are commodity (Gigabit Ethernet links, switches), and the hardware attachment completely retains HP's current investment, the SANIC can be a new piece to be developed. Inline with the operational paradigm shift, the SANIC must support some message passing protocol extended to/tailored with I/O needs. The example protocol we select here is VIA. However, ST can be equally deployed. Intel has demonstrated VIA through a 100Mb/s Fast Ethernet NIC. Given that Gigabit Ethernet retains the semantic of the data link layer completely, a VIA-compliant Gigabit Ethernet NIC might not be that difficult/expensive to achieve. The host side of the VIA needs to be changed to assume the duty of fragmenting and reassembling for GbE-Pipe, it will also need to administrate multiple SANIC. In fact, this is precisely HP's control point of this technology.

Of many attributes that VIA embodies, the two most relevant are:

1. The ability to establish many end-to-end Virtual Interface connections. We will use the VI connections as vehicles to map GbE-pipe connections.

2. The ability of the SANIC to directly access to and from the host memory in a protected fashion. This is going to facilitate the fragmentation and reassembling for large I/O messages.

With the first attribute of VIA, we are already able to establish end-to-end logical connections. The major difference here is that, while VIA as today dictates that one connection must run within a single pair of SANICs, we must enable it to run on a set of SANICs. For QoS and HA purposes, we should also be able to dynamically change the set of SANICs allocated to a particular VI connection.

Fragmentation should be easy: we simply chop large message into fragments and dispatch them to dedicated SANICs by ringing the associated doorbells. Reassembly comes with two steps, the first is to upload each fragment into the remote-end's memory. All fragments must be tagged with

the relative offset so that they can be directed to the right portion of the receiving buffer. The second step, which is to notify the complete arrival of the whole message, is a little bit tricky. This is because we no longer have the luxury of having one central control point at the remote end where we can easily know the arrivals of all the fragments. The algorithm employed in Hamlyn to do message reassembly would therefore be deployed elsewhere, for example in the host memory, with a read-write-compare sequence.

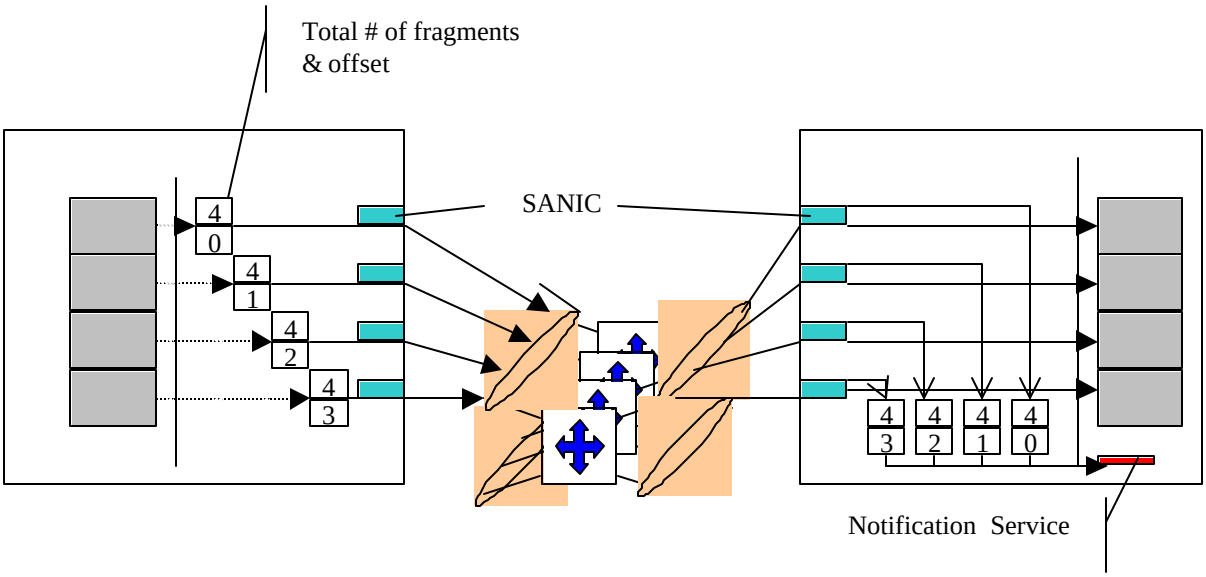

Figure 5: Fragmentation and Reassembly of Large Message

The above diagram illustrates an example of reassmbly with Hamlyn's algorithm running at the host memory. The read-modify-compare sequence performed by SANIC at the time of delivering a fragment thus runs as follows:

1. Read out the number of fragments received
2. Increment the number by one
3. If the result equals the total number of fragments, notify the host that a complete message has arrived.

3.2.3 Improve the Reliability of S-GES

One primary concern of translating any commoditiness into high/mid-end servers' play field is the issue of reliability. For I/O, this isn't universally true. For example, higher stacks of inter-networking are designed to deal with WAN where loss of packets are simply routine cases. For storage, however, drivers are assuming a robust infrastructure underneath, losing storage related I/O messages is unthinkable.

In order for the S-GES proposal to be practical, reliability must be addressed. Fortunately, HP's partner in the switching business, Cisco, has reportedly achieved end-to-end QoS as well as the true-robustness: no loss of packets. At the writing of this report, it is unknown to the author how Cisco implemented that. Most likely, MAC level credit-based flow control is employed.

As mentioned earlier, the main loss of frames in a full-duplex Gigabit Ethernet configuration is in the limited buffer space in the switch. But *memory is cheap*, so we can equip those commodity switches with large amount of memory and thus improve the reliability of S-GES.

But how would we know how much memory is needed in the switch?

Interestingly, for sender-based message passing protocols, the answer is right at hand. Protocols such as ST/Lowfat basically prevent loss of message at the receiving end by letting sender and receiver negotiate available buffer space up front, this enables sender to capture the precise image of memory usage at the remote end, and thereby forbid the sender of running over.

An upbound of switch memory can therefore be reasoned as follows. For a given pair of connection *A->B*, let *M(A,B)* be the buffer space allocated at the end of *B*, since there are maximum *M(A,B)* messages that will be inflow, we need at the worst case *M(A,B)* at the switch. The total sum of memory at the switch is thus the summation over *M(X,Y)* for any *X->Y* connections that will route through this switch. This equation can be worked backwards, i.e., given the switch memory attached to a switch output, find out how much memory the ST/Lowfat protocol should allocate at the remote end. Philosophically, this is similar to finding the path-MTU in the networking world.

This upbound is likely to be excessive, whether it is truly excessive depends on the memory required by end devices and hosts (as shown in the equation), times the relative expense of – memory. Better algorithms that use the memory more efficiently might be developed later.

4. S-GES Attributes

4.1 High-Availability

Use large amount of thinner links naturally aids high-availability in a cost-efficient way. As long as the connectivity is still on, even one link should be able to get the system up running. Cisco's EtherChannel employs one redundant link out of nine, and if one link goes bad, then a transparent recovery is realized in 1 second. This is not sufficient, not only because trunking is not a preferred solution at the first place (see 3.1), but also due to the fact that it does not account for switch failure. Furthermore, the recovery time may be too high.

S-GES makes fanciful solutions, such as RAIDed networking which is rendered too expensive in the context of main processor/memory interconnect, possible again. For example, when transmitting fragments of large message, we can always compute a parity fragment and send it across a redundant plane of the S-GES. In this case, recovery is instantaneous and the incremental cost is only 1/N. The problem, of course, is the overhead incurred computing the parity fragment as well as reconstructing the missing fragment. This is not a problem if the computation is processed in hardware.

Figure 6 describes how this would work for 5 links, one of which is the redundant SANIC (SANIC-5). For each normal fragment sent out, a copy is sent to SANIC-5, which XORs over the normal fragments into the parity fragment. When SANIC-5 receives all the fragments, the parity fragment has been completely processed and it is sent out. The counter part of SANIC-5 at the remote end performs recovery in much the same way: it will get a copy of normal fragments from other SANICs as well as the parity copy itself received from the network; it then performs XORs

on these fragments. If one normal fragment is missing but all of the rest, including the parity fragment are received, the missing fragment is reconstructed automatically. Note that in order for this algorithm to work, peer-to-peer support must be provided among the SANICs. This solution would certainly beat the 1 second Cisco's recovery time fair and square. Note that with dynamic allocation of links to VIA, incremental HA is possible: if one normal SANIC or channel goes bad, we just drop it; if the redundant one goes down, one normal channel can be reallocated to be the redundant one again.

This algorithm does introduce an expected problem: the violation of exactly-once delivery, since it's possible that a missing fragment that has been re-constructed by the parity fragments simply emerged again from the network again. Some provisions must deal with this unwanted redundancy.

Figure 6: Instantaneous HA solution for S-GES

This is a *N+1* solution, for large *N*, the probability of multiple faults can be higher than desired. In that case, solutions such as *M(N/M+1)* can be used. That is to say, the normal channels are grouped into *N/M* groups, each has a redundant channel.

4.2 QoS

The VIA-compliant SANIC does not provide an adequate solution for QoS, since the card is required to multiplex requests fairly. As a matter of fact, QoS isn't an easy bite. Cisco's QoS solution in Gigabit EtherChannel, if indeed lives up to its promise *and* is adequate to our needs, should be leveraged.

Some crude-level QoS can be realized via GbE-pipe. Different applications can simply use different set of the links. A QoS service should be derived to control the allocation of channels/links of the GbE-pipe among applications dynamically.

Detailed algorithm should be worked out. Since remote devices are shared, local QoS which only deals with allocating QoS metrics among the local applications, probably isn't enough.

4.3 Interoperability

If owning the data center is the goal, selecting the most popular network medium is the strategy to go. This is arguably one of the strength of S-GES.

4.4 Cost / Performance

This should really be derived in contrast with other solutions, i.e., in Section 6. But stacking Gigabit Ethernet has its interesting cost structure: the fewer layers you stack, the less bandwidth you get, and the incremental is in the order of 1Gb/s (one direction per connection).

5. Unsolved Mysteries

Using commodity parts to build a SAN adequate for high/mid-end share a few problems:

1. **Manageability**
   Managing more components is always harder. We should think hard on how to leverage the management tools developed by Cisco.

2. **Wiring**
   Back to its glory days of last decade, Center for Supercomputer Research and Development in University of Illinois at Urbana-Champaign built a 32-way multiprocessor called *Cedar*. One of my early assignments as a Ph.D student was to study its network performance. We were surprised to learn that the middle stage of its MIN-topology network has only 16 links, so our study has to take a 16-way configuration to setup a balanced system. What was the problem? Well, it turns out that the designers of Cedar made an early mistake of not leaving enough cabinet space for wiring. The same package problem might come to hunt us once we decide to take the aggregation approach – early caution is very much needed.

3. **Bandwidth (*Still!)***
   Because of the first two issues (and possibly many others), there are simply physical and mental limits that an aggregation approach can go as far as the bandwidth is concerned. Technical applications, albeit not the dominant market segment, demands lots of bandwidth, possibly in the order of 2GB/s per processor. There are two possible solutions, both are speculations at this stage and not very satisfactory. The rational, however, is the same: if aggregation approach satisfy majority of applications crucial to HP's business, then only changes pertain to technical applications should be made.

   The first is to see if a hybrid approach is possible. If the traffic pattern is such that there is significant disparity between bandwidth requirements to and from the hosts, then aggregation can be used to target the lightly loaded one, and subsequently adopt a fat pipe for the other direction.

   The second is to ask the fundamental question as *whether we are fight the battle at the right hill*? HP-Lab's MADE group had made the argument in the past that query primitives can be easily offloaded into the I/O subsystem, by which a 4X performance gain can be delivered, not to say the *significant* bandwidth reduction. I believe the same is true for some Web applications, such as proxy caching. None effort has been made to repeat the same qualitative and quantitative estimation for technical applications, as well as the associated feasibility. It may be time to re-visit the topic again.

4. **Others?**

6. Firehose versus Bunches of Drinking Straws: SAN technology compared

At a higher level, we can classify SAN alternatives into two classes: those with an aggregation approach such as S-GES and those using a fat-pipe. It is important to understand that S-GES is more an option than a solution. To be a true option, any proposal has to be a legitimate solution first, however we want to emphasize that our mindset should be open rather than closed. As such, we are waiting it to be challenged! Quality only comes with competition. What other options are available?

- Myrinet
- SCI
- Fiber Channel/Switch
- HIPPI-6400/GSN
- *(please help to fill more*)

The structure of comparison should be as follows. For each option, a deployment plan should be similarly developed first. This would include elements such as topology, attachment, hardware-software interface. Once this step is completed, then we would like to compare them along a few metrics: cost/performance, reliability, HA, QoS, manageability, packaging etc.

7. Conclusions

The major takeaways from the report are as follows:

1. It will be beneficial to leverage commodity parts to play the high/mid-end game.
2. It will be advantageous to retain HP's current I/O hardware infrastructure.
3. Bandwidth of commodity part is *not* a problem: they can be aggregated; Latency isn't much an issue here either: hardware is lean and fast, software stack is the one to blame.
4. The aggregation needs algorithms to do fragmentation and reassemby, which isn't too difficult.
5. Sender-based message passing protocols can help to perfect the reliability of Gigabit Ethernet (or anything else for that matter).
6. HA can be provided through redundant parity fragment…
7. SAN alternatives using an aggregation approach such as S-GES faces a few major challenges, such as manageability and packaging/wiring. Because of these problems, there is also a limit it can go to supply bandwidth.