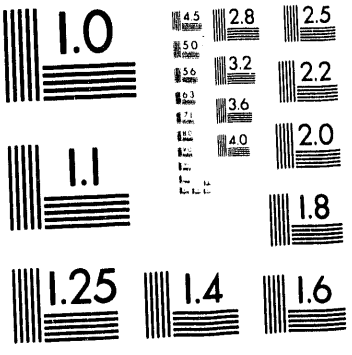

1.0
4.5
2.8
2.5
5.0
3.2
2.2
5.6
6.3
3.6
7.1
4.0
2.0
1.1
1.8
1.25
1.4
1.6

1 of 1

SAND93-8013
Unlimited Release
Printed October 1993

UC-405

EVALUATION OF DEC'S GIGASWITCH FOR DISTRIBUTED PARALLEL COMPUTING

Helen Chen, Jim Hutchins, and Jim Brandt
Networking & Communications Department
Sandia National Laboratories/California

Abstract

One of Sandia's research efforts is to reduce the end-to-end communication delay in a parallel-distributed computing environment. GIGAswitch is DEC's implementation of a gigabit local area network based on switched FDDI technology. Using the GIGAswitch, we intend to minimize the medium access latency suffered by shared-medium FDDI technology. Experimental results show that the GIGAswitch adds 16.5 microseconds of switching and bridging delay to an end-to-end communication. Although the added latency causes a 1.8% throughput degradation and a 5% line efficiency degradation, the availability of dedicated bandwidth is much more than what is available to a workstation on a shared medium. For example, ten directly connected workstations each would have a dedicated bandwidth of 95 Mbps, but if they were sharing the FDDI bandwidth, each would have 10% of the total bandwidth, i.e., less than 10 Mbps. In addition, we have found that when there is no output port contention, the switch's aggregate bandwidth will scale up to multiples of its port bandwidth. However, with output port contention, the throughput and latency performance suffered significantly. Our mathematical and simulation models indicate that the GIGAswitch line efficiency could be as low as 63% when there are nine input ports contending for the same output port. In a distributed parallel computing environment, output contention often involves the contention for a server's resources. Our data indicate that the delay introduced by contention at the server workstation is 50 times that introduced by the GIGAswitch. We conclude that the GIGAswitch meets the performance requirements of today's high-end workstations and that the switched FDDI technology provides an alternative that utilizes existing workstation interfaces while increasing the aggregate bandwidth. However, because the FDDI standard limits the available bandwidth to 100 Mbps and the speed of workstations is increasing by a factor of 2 every 1.5 years, the switched FDDI technology is only good as an interim solution.

MASTER

Acknowledgement

We would like to thank Don Myhra and Mike Wolinski from DEC for their help in making the GIGAswitch beta test a success. Special thanks are also due to Rich Gay and Tim Cody for designing and implementing the fiber optic infrastructure for the GIGAswitch testbed, which included 50 workstations.

Contents

Page

I. Introduction 7

II. GIGAswitch Characterization 7
- Switch Throughput 8
- Switch Latency 8
- Switch Line Efficiency 9

III. GIGAswitch Stress Test 10
- Baseline Workstation Performance 11
- GIGAswitch Aggregate Throughput and Line Efficiency 12
- GIGAswitch Head of Line Blocking Analysis 14
- Effect of GIGAswitch Traffic Load on End-to-End Latency in Distributed Parallel Computing 16
- End-to-End Latency Analysis Using DECmcc and Tekelec FDDI Analyzer 19

IV Conclusions 28

References 29

Acronyms and Abbreviations

DAS	Dual Attached Station
FDDI	Fiber Distributed Data Interface
Mbps	Megabits per Second
RTT	Round Trip Time
SAS	Singly Attached Station
TCP	Transmission Control Protocol

I. Introduction

One of Sandia's research efforts is to reduce the end-to-end communication delay in a distributed, parallel computing environment. To meet the challenge of achieving sub-millisecond latency for our distributed computing environment, we investigated the switched FDDI technology [1]. Switched FDDI provides a dedicated 100 Mbps to a single workstation, thereby eliminating the token access delay of a shared-medium network. Since a switched network has an aggregate bandwidth in the multi-gigabit range, the total number of stations that can be supported on the switched network can be significantly more than that of a shared-medium FDDI network. The DEC FDDI GIGAswitch [2] represents an early implementation of this technology and promises to provide increased performance on communication intensive applications. It provides an alternative that utilizes existing workstation interfaces while increasing the performance, all for the cost of a switch.

II. GIGAswitch Characterization

Figure 1 depicts the network configuration for characterizing the GIGAswitch's throughput, latency, and line efficiency. The GIGAswitch is initially configured with three FDDI line cards, each line having two dual attached station (DAS) FDDI ports. Four SGI R4000 Indigos are singly attached (SAS) to the four GIGAswitch FDDI ports. The remaining two GIGAswitch ports connect a DAS Fibronic FDDI-Ethernet bridge and a DAS cisco router. The Fibronic bridge provides a link to a DECmcc SNMP [3] station, and the cisco router connects the GIGAswitch testbed to our open production network. We disconnected the cisco router from the GIGAswitch during our test to prevent user traffic from interfering with our experiments.

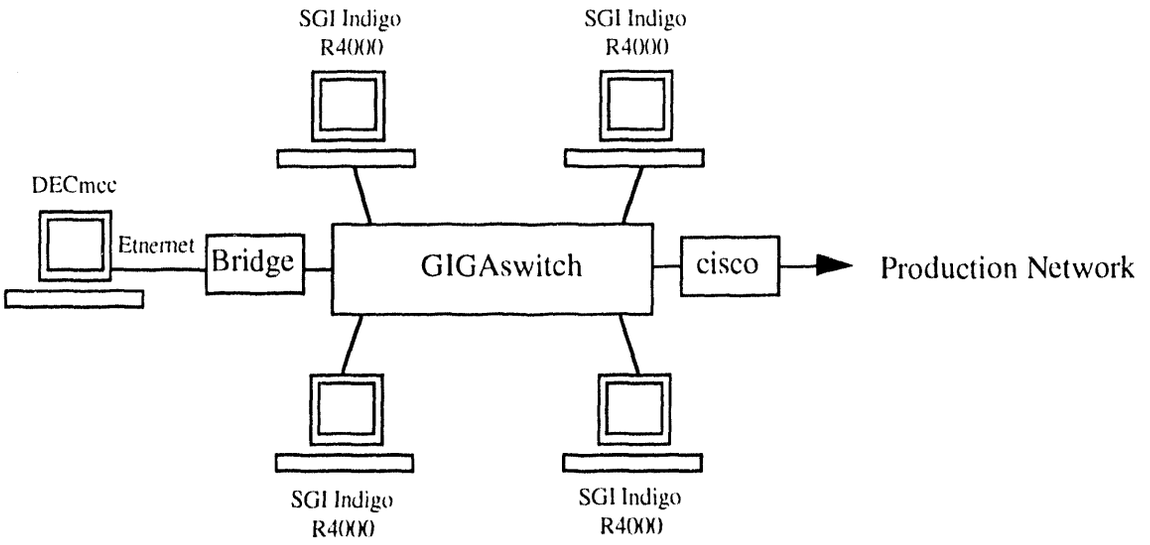


Figure 1. Phase I testbed configuration.

Switch Throughput

We evaluated the GIGAswitch throughput using TTCP, which is a TCP-based [4] metric tool that measures the speed of data transfer between memories of two machines. Parameters such as TCP window size, user packet size, and the number of user packets of a session can affect the performance. These parameters can be specified as command line arguments. For our experiment, we selected a 63-kilobyte TCP window size, a 63-kilobyte user packet size, and 2048 packets for each run. Because the testbed is standalone, we believe that the real throughput can be reflected by the average of three measurements. We measured the FDDI throughput of two Indigos directly connected with the GIGAswitch and in between. The results are summarized in Table 1. As Table 1 shows, the TTCP throughput of two Indigos connected by the GIGAswitch is found to be 89.7087 Mbps. This value represents a 1.8% degradation of the throughput of two directly connected Indigos at 91.3749 Mbps. However, the 89.7087 Mbps is dedicated bandwidth and it is much more than the available bandwidth of a shared medium FDDI when there are more than a few machines on the network.

Table 1. Throughput Measurements for GIGAswitch

	Throughput (Mbps)			Average (Mbps)
Without Switch	91.5734	91.0189	91.5325	91.3749
With Switch	89.8410	89.3949	89.8903	89.7087

Switch Latency

A UDP-based [5] ECHO client tool was written to measure the round trip time (RTT) between a pair of the R4000 Indigos that were bridged by the GIGAswitch (Figure 1). The ECHO server, which is distributed by most UNIX operating systems, listens on UDP port 7 and responds to the echo requests. The UDP client can take command line arguments for packet size and for number of packets per run. We calculated the RTT by dividing the elapsed time by the number of packets sent. Our experiment sent 10,000 packets of a range of sizes—64, 128, 256, 512, and 1024 bytes—and each experiment was repeated three times. A similar experiment was run with the two workstations directly connected to each other; Table 2 summarizes the results of the two RTT experiments. Calculating the GIGAswitch latencies by taking one half of the differences of the RTT values, we found the average latency to be 16.5 microseconds. Figure 2 plots the RTT values versus the ECHO packet sizes. Larger packets require more serialization time and more memory bandwidth of the workstation, and therefore have a direct effect on the RTT values.

Table 2. Round Trip Time Measurements (Microseconds)

	Packet Size	RTT			Average
With/Without Switch	64	1354/1228	1234/1233	1241/1204	1276/1222
	128	1330/1297	1330/1304	1329/1292	1330/1298
	256	1375/1349	1362/1330	1362/1348	1366/1342
	512	1576/1659	1585/1558	1594/1537	1585/1585
	1024	1764/1841	1874/1728	1848/1750	1829/1773

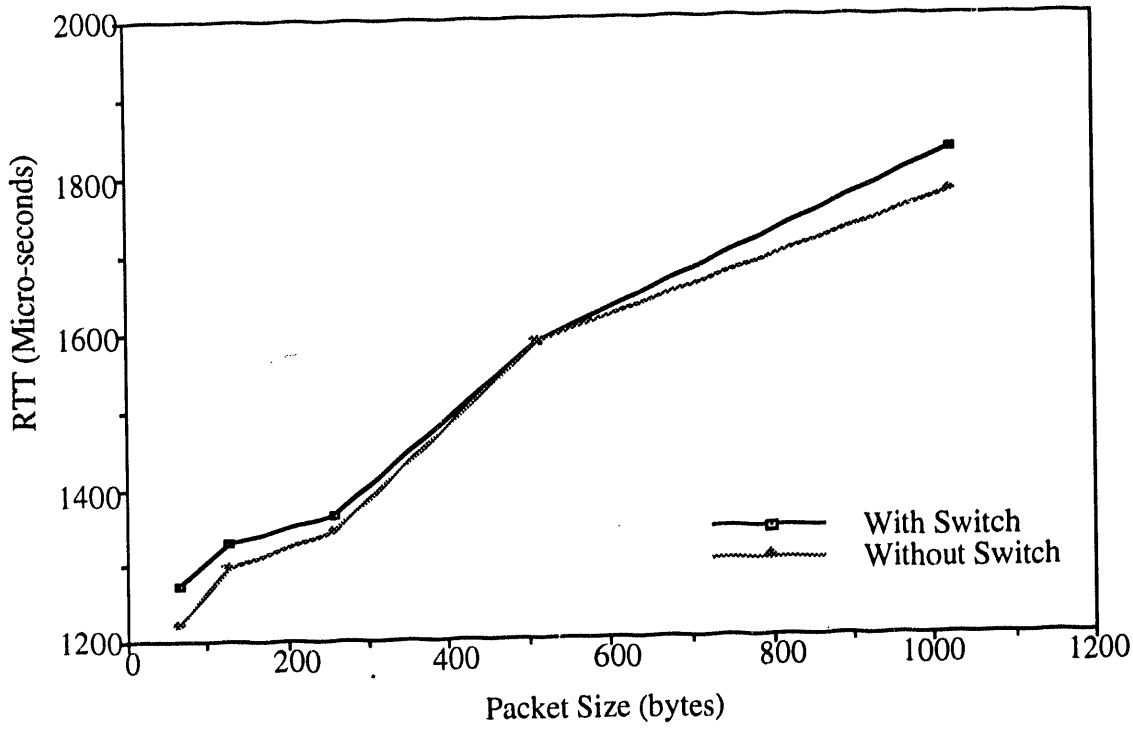


Figure 2. Graph of the end-to-end RTT vs. packet size.

Switch Line Efficiency

The bandwidth requirement of one TTCP session between a pair of Indigos bridged by the GIGAswitch is 81.1417 Mbps. Therefore, two concurrent TTCP sessions will saturate the available GIGAswitch port bandwidth of 100 Mbps. We conducted this experiment by running two concurrent TTCP sessions using the testbed configuration depicted in Figure 3. The two TTCP sessions were between workstations A and C and workstations B and D. The results are summarized in Table 3. The average GIGAswitch line efficiency, calculated by dividing the aggregate throughput of the two sessions by 100, is 94.5146% and is less than the line efficiency of a small FDDI network, 99.47% [6].

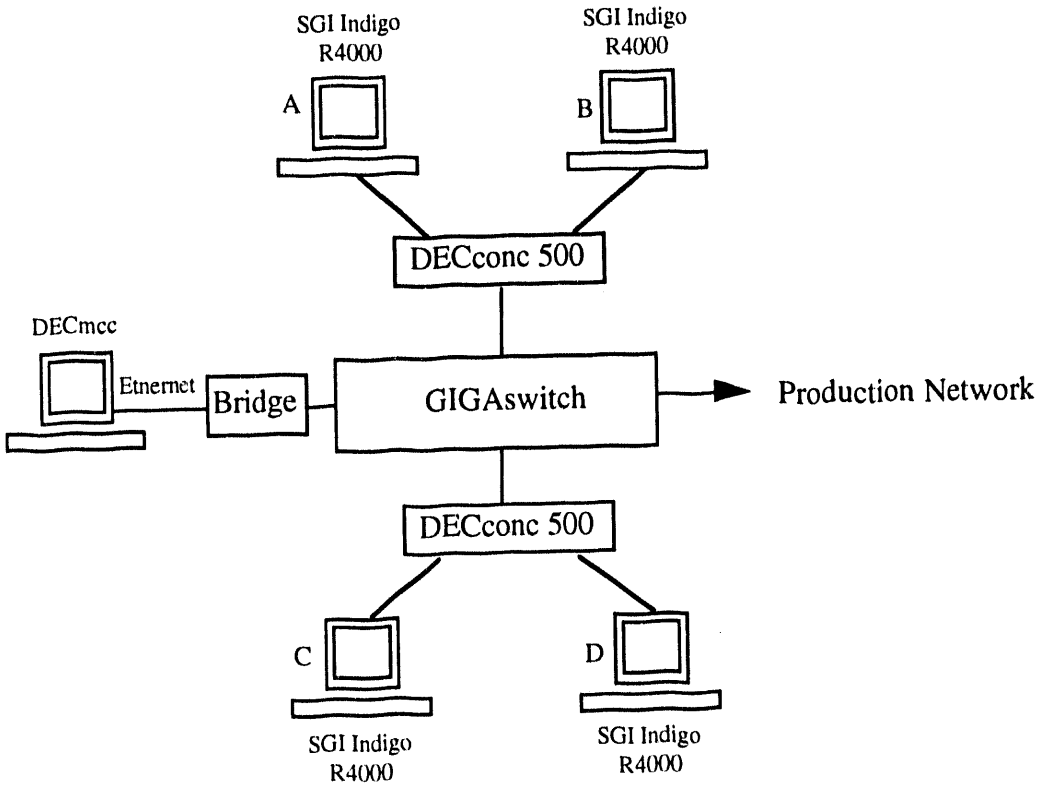


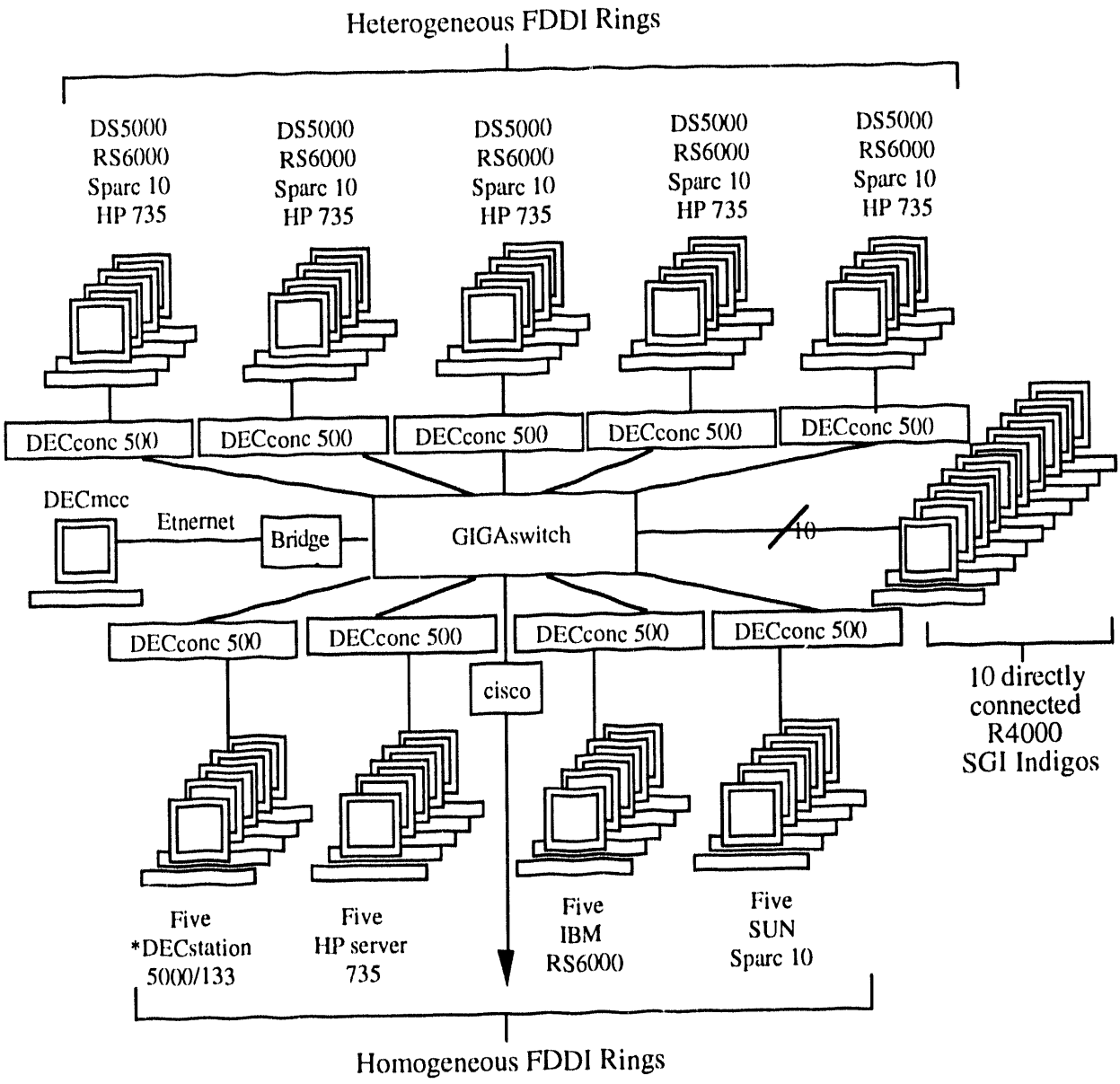
Figure 3. Testbed configuration for GIGAswitch line efficiency evaluation.

Table 3. GIGAswitch Line Efficiency

	Experiment 1	Experiment 2	Experiment 3	Experiment 4
Session 1	47.8882	46.8257	47.6714	47.0451
Session 2	49.6707	47.7569	46.3794	47.4695
Percent Efficiency	97.5589	94.5826	94.0508	94.5146

III. GIGAswitch Stress Test

To stress test the GIGAswitch, traffic was generated by the fifty workstations in the Heterogeneous Environment and Testbed as shown in Figure 4. The GIGAswitch was reconfigured with eleven dual-port FDDI line cards or twenty-two FDDI ports. Because of its high performance requirements, each of the ten Indigos had a dedicated FDDI port. The other workstations (ten DS5000/133's, ten HP735's, ten RS6000's, and ten Sparc10's) shared the switch ports through nine DEC FDDI concentrators. Four concentrators were used to form four FDDI networks with homogeneous workstations, five DS5000/133's, five HP735's, five IBM RS6000's, and five Sun Sparc10's. There were also five DEC concentrators used to form five heterogeneous rings, each containing an HP735, a DS5000/133, an IBM RS6000, and a Sun Sparc10. Two switch ports connected to our production FDDI network and a DECmcc Network Management Station. This left a spare port, which was not used in this test.



* The DS5000/133s are in the process of being upgraded to Alphas.

Figure 4. Phase II testbed configuration.

Baseline Workstation Performance

The throughput and latency values of each kind of workstation were established using the TTCP and the ECHO RTT tool. These tests were performed between a pair of like workstations with no other traffic on the network. The client processes were initiated from the workstation on the homogeneous FDDI ring and the server processes were run on the workstation on the heterogeneous FDDI ring (see Figure 4). The results, summarized in Tables 4 and 5, will be used as the baseline values for evaluating the GIGAswitch performance under heavy traffic.

Table 4. Baseline Workstation Throughput (Mbps)

	R4000 Indigo	DS5000/133	HP 735	RS6000	Sparc10
Throughput	89.7087	14.96	34.20	42.69	23.56

Table 5. Baseline Workstation Round Trip Time (Microseconds)

Packet Size (bytes)	RTT				
	DS5000	HP735	Sparc10	RS6000	Indigo
64	2722	1544	1847	1994	1134
128	2836	1566	1935	1985	1267
256	3032	1639	1975	2223	1325
512	3542	1767	2001	2254	1421
1024	4009	2038	2171	2538	1626

GIGAswitch Aggregate Throughput and Line Efficiency

To evaluate the GIGAswitch performance under heavy load, a Client/Server based tool was used to trigger twenty-five concurrent TTCP sessions, each of which ran between a pair of like workstations. The twenty-five TTCP sessions are described as follows:

Session Description	No. of Concurrent Sessions
Indigo-to-Indigo	5
HP735 (homo)-to-HP735 (hetero)	5
DS5000 (homo)-to-DS5000 (hetero)	5
Sparc10 (homo)-to-Sparc10 (hetero)	5
RS6000 (homo)-to-RS6000 (hetero)	5

The five Indigo-to-Indigo TTCP sessions used dedicated FDDI input ports and dedicated FDDI output ports. All other workstations had their TTCP client sessions initiated from the workstations on the homogeneous FDDI rings and their TTCP server partners scattered among the five heterogeneous FDDI rings (see Figure 4). The results are tabulated in Table 6.

Table 6. GIGAswitch Aggregate Throughput (Mbps)

Sessions	Throughput					
	Session 1	Session 2	Session 3	Session 4	Session 5	Aggregate
Indigo	86.52	90.17	92.51	92.97	-*	N/A
HP735	19.03	19.13	19.53	19.47	-*	77.15
DS5000	14.89	15.00	14.91	15.20	-*	60.00
Sparc10	17.19	17.03	17.48	17.48	17.45	86.43
RS6000	14.34	14.11	14.84	14.47	14.80	72.56
				Average	Aggregate	73.93

* There are no results here due to workstation hardware failure.

Based on the baseline results in Table 4, the RS6000, the HP735, and the Sparc10 can saturate a GIGAswitch FDDI port with four concurrent TTCP sessions. However, the observed results shown in Table 6 do not support this prediction. The observed aggregate throughput suffers significant degradation when there are four input ports, one each from the Sparc10, HP735, DS5000, and RS6000 FDDI ring contending for the same output port containing the target Sparc10, HP735, DS5000, and RS6000. Additional experiments were performed to collect throughput values with one, two, and three input ports contending for the same output ports. The results are summarized in Table 7. The effect of output port contention on performance and on line efficiency is studied in more detail in the next section. Further study of the test results in Table 6 reveals that when there is no output contention as in the SGI Indigo sessions, the bandwidth of the GIGAswitch scales up to multiples of the port bandwidth.

Table 7. GIGAswitch Aggregate Throughput with Varying Output Port Contention

Contending Input Ports	Aggregate Throughput (Mbps)				
	RS6000	Sparc10	HP735	DS5000	Average
1	94.92	-	-	-	94.92
2	87.62	81.53	-	-	84.58
3	72.52	73.64	79.54	-	75.23
4	72.56	86.63	77.15	60.00	73.93

GIGAswitch Head of Line Blocking Analysis

The DEC GIGAswitch appears to suffer a significant performance degradation when multiple input ports attempt to send large amounts of data to one output port. After studying the problem to determine the likely cause, we surmise that the effect is probably due to head-of-line blocking. Head-of-line blocking occurs when an input port attempts to send data to an output port but is blocked because another input port is already sending to that output port. Any other data queued at the input port behind the blocked packet cannot be sent, even if its destination is unblocked, until the blocked packet has been sent. Thus, the entire queue blocks whenever the "head of the line" of the queue blocks.

An example of head-of-line blocking for a two-input-to-two-output case is shown in Figure 5. The figure shows the initial state, where each input has six packets to send to the output ports, and the state after six time periods. The gaps in the output streams represent periods of inactivity due to head-of-line blocking. For example, when input port 2 attempts to send packet 2 to output port B, it is blocked by packet 2 from input port 1 and must wait until the next time slot to send it, while output port A remains inactive.

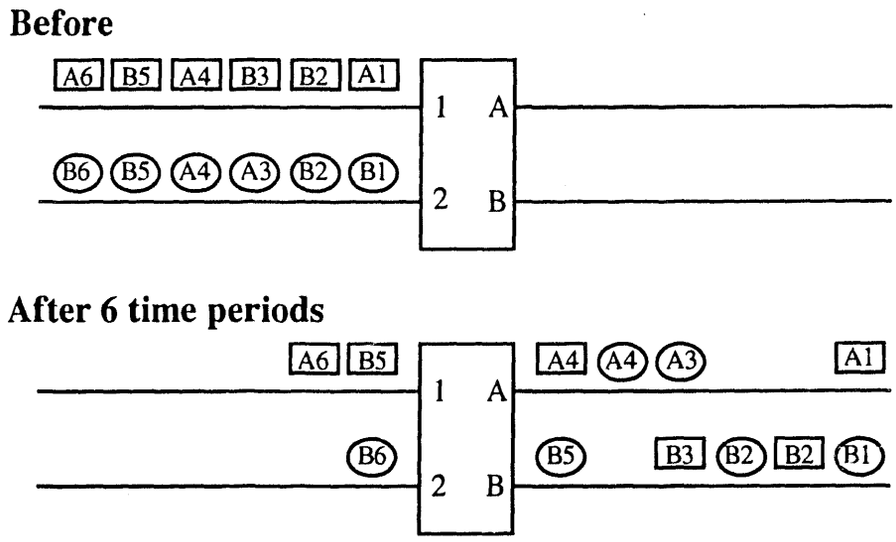


Figure 5. Example of Head-of-Line Blocking.

To validate this hypothesis, a mathematical model of the head-of-line blocking effect was created to compare against the experimentally derived data. Since the experiment consisted of machines sending large amounts of data over TCP connections at their maximum rate, the model assumes that all packets are of the same size. Further, since when a port blocks it must wait until the other sender finishes, all the senders will tend to become synchronized on when they send. By ignoring the interdependency between adjacent time periods, the model simplifies to M input ports randomly trying to send to N output ports at the same time. The expected throughput from the standpoint of the input ports is the percentage of the input packets attempted that get through. Using probability theory, the expected input throughput efficiency, E, is

$$E = [\, 1 + (1 - \frac{1}{N}) + (1 - \frac{1}{N})^{2} + \ldots + (1 - \frac{1}{N})^{M-1} \,] / M$$

which reduces to

$$E = \frac{N}{M}\left[1 - \left(1 - \frac{1}{N}\right)^{M} \right]$$

In our experiment, we varied the number of sending ports while leaving the number of receiving ports constant at five. Table 8 shows the calculated efficiency for one through five input ports sending to five output ports.

Table 8. Calculated Percent Efficiency for One Through Five Ports Contending for Each of the Five Output Ports

No Input Ports	1	2	3	4	5
% Efficiency	100	90.0	81.3	73.8	67.2

The maximum throughput we were able to achieve with no ports contending and hence no head-of-line blocking was 95 Mbps. Assuming this is the maximum speed for this implementation of FDDI, we use the percent efficiency as the percentage of this rate. Table 9 contains the calculated and observed rates for one through five input ports. The agreement of the numbers within the simplifying assumptions appears good. A mathematical model which takes into account the interdependency between adjacent time periods was derived by Joseph Hui and Edward Arthurs [7] and yields efficiencies slightly lower than our model, which brings the agreement even closer.

Table 9. Comparison of Simulation Rates to the Observed Rates

No. Input Ports	Expected Rate	Observed Rate
1	95.0	94.92
2	85.5	84.58
3	77.2	75.23
4	70.1	73.93 *
5	63.9	N/A

*Because of hardware problems, the test with four input ports only had four machines available on two of the input ports, so a lower port collision rate and hence higher throughput rate would be expected.

Figure 6 plots the projected maximum efficiency for the number of output ports versus the number of input ports based on the model for a larger number of inputs and outputs. This plot can be used as a quick reference to estimate maximum efficiency of line utilization for various configurations and traffic patterns.

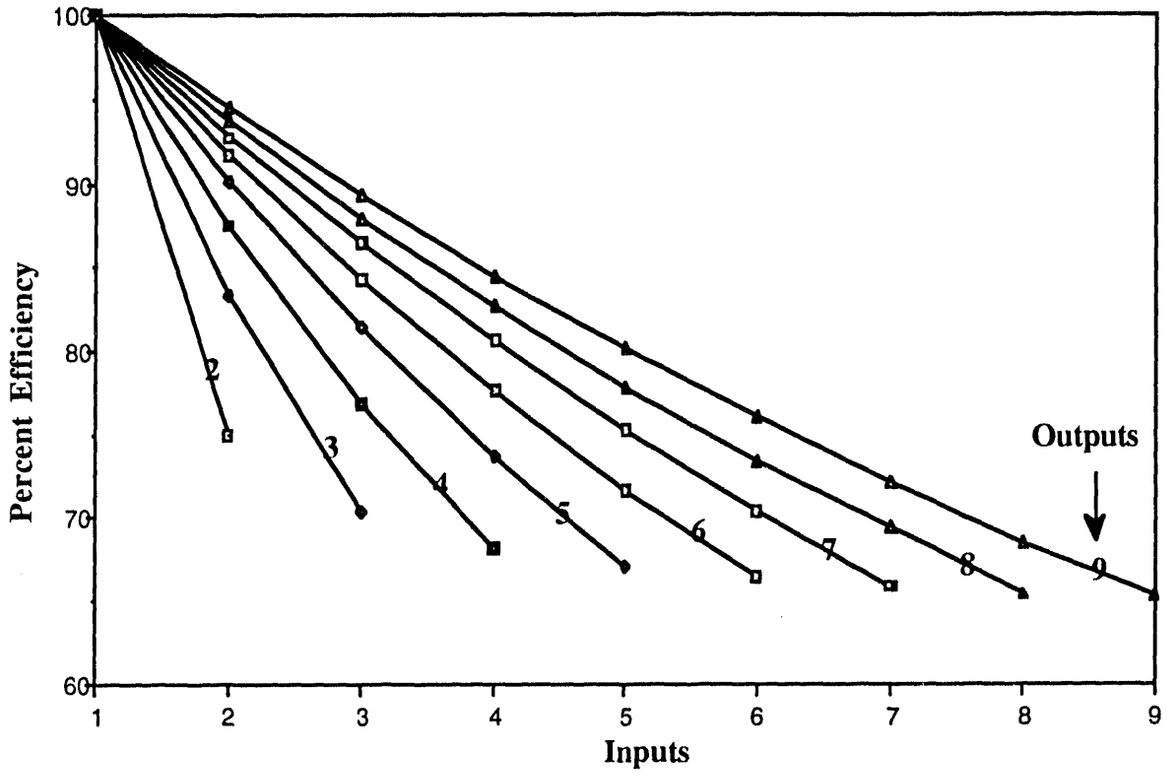


Figure 6. Maximum efficiency for number of output ports versus number of input ports.

Effect of GIGAswitch Traffic Load on End-to-End Latency in Distributed Parallel Computing

This experiment evaluated the effect of the GIGAswitch's traffic load on the end-to-end latency in a distributed parallel computing environment. Again we chose packet sizes ranging from 64 to 1024 bytes because of the message-passing nature of the distributed parallel computing. Using the ECHO RTT tool, the round-trip-times were measured as follows:

(1) One-to Many: seven RTT ECHO clients from one Indigo to seven RTT ECHO servers on seven different Indigos; and

(2) Many-to-One: seven RTT ECHO client sessions from seven separate Indigos to the same RTT ECHO server on one Indigo.

The results are tabulated in Tables 10 and 11, and a graph of the RTT values against message packet sizes is depicted in Figure 7. As shown in Figure 8, the round-trip-time is significantly increased when there is contention at the destination port due to queuing. Furthermore, when packet sizes are 512 bytes or larger, all of the RTT ECHO sessions aborted due to programmed time-outs. In an attempt to understand the symptom, we studied the effect further using the SNMP-based DECmcc and using the traffic traces collected by a Tekelec FDDI analyzer. The results are analyzed in the next section.

Table 10. Effect GIGAswitch's Traffic Load to RTT (microseconds): One-to-Many (Indigos). * These failures are due to programmed time-outs.

Packet Size (bytes)	RTT							
	Session 1	Session 2	Session 3	Session 4	Session 5	Session 6	Session 7	Average
64	2692	2540	2563	2600	2630	2547	2650	2603
128	2889	3016	2816	2959	3044	3032	3458	3030
256	2642	2772	2864	2778	2870	2907	2843	2811
512	3256	3306	3278	3300	3347	3323	failed*	3347
1024	3330	3425	3346	3381	3475	3411	failed*	3395

Table 11. Effect of GIGAswitch's Traffic Load to RTT (Microseconds): Many-to-One.

Packet Size (Bytes)	RTT							
	Session 1	Session 2	Session 3	Session 4	Session 5	Session 6	Session 7	Average
64	2393	2383	2389	2381	2387	2356	2389	2383
128	2498	2483	2487	2488	2484	2468	2457	2481
256	2602	2608	2614	2605	2611	2614	2616	2610
512*	failed	failed	failed	failed	failed	failed	failed	-
1024*	failed	failed	failed	failed	failed	failed	failed	-

*When the UDP packages are 512 bytes or larger, all the RTT ECHO sessions are aborted due to time-outs.

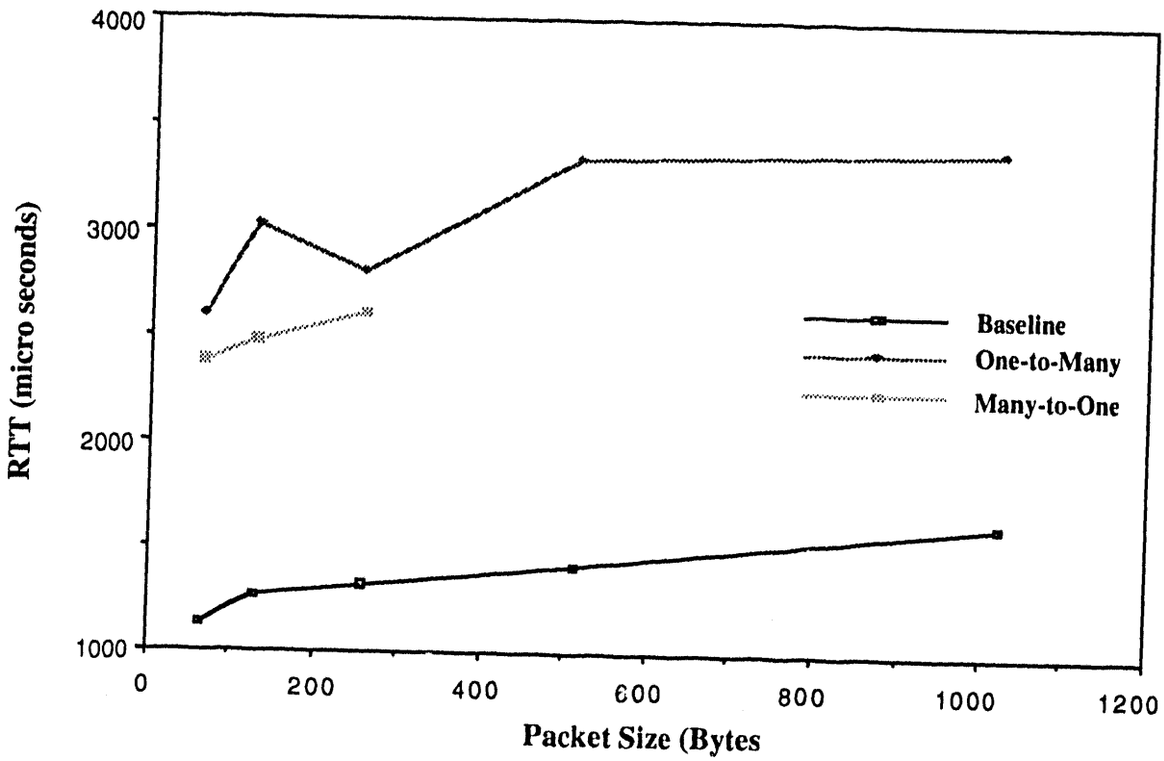


Figure 7. *Effect of GIGAswitch's traffic load on end-to-end delay.*

End-to-End Latency Analysis Using DECmcc and Tekelec FDDI Analyzer.

A Tekelec FDDI analyzer and a DECmcc station were used to monitor the statistics of the GIGAswitch for the "One-to-Many" and the "Many-to-One" RTT ECHO experiments. Because the GIGAswitch counters did not show any packet drops during the experiments, we suspected that the aborted experiments might have been caused by a time-out in the RTT program, which was set at 10 ms. In order to ensure that the previous RTT ECHO sessions would not abort due to time-out, the time-out value was extended from 10 to 30 milliseconds, and the same experiments were repeated with seven R4000 Indigos.

There were six concurrent RTT ECHO clients from six Indigos sending echo requests to one RTT ECHO server. Two identical runs were traced for message size at 64, 128, 256, 400, 500, 512, and 1024 bytes. The first run had the Tekelec FDDI analyzer placed between the GIGAswitch and an ECHO client Indigo, as shown in Figure 8, and the second run had the Tekelec placed between the GIGAswitch and the ECHO server Indigo. For each configuration, the traffic was collected between the same two Indigos.

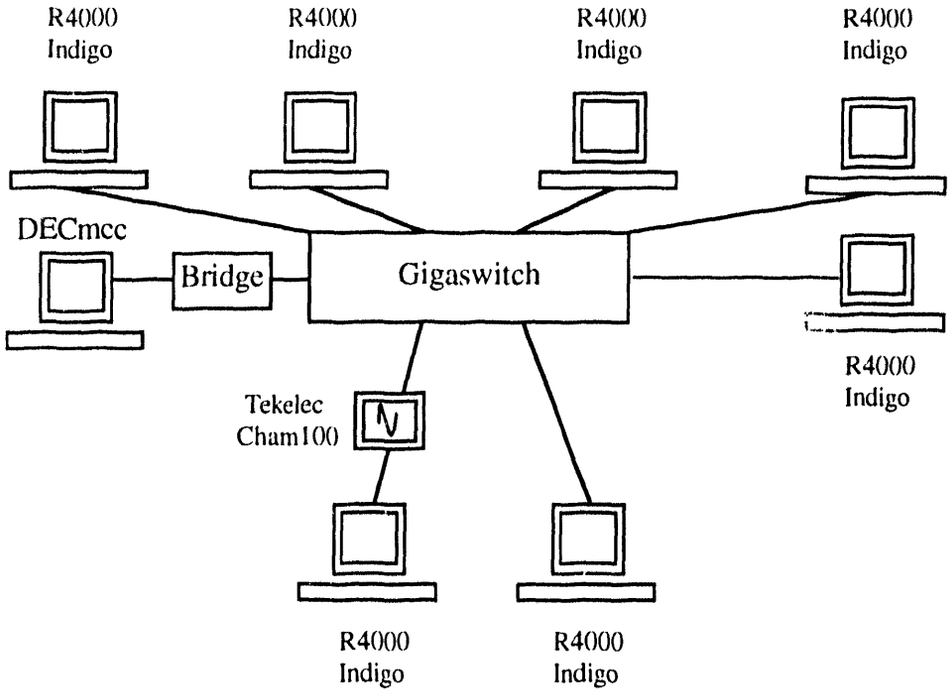


Figure 8. *Testbed configuration for latency analysis.*

Figure 9 depicts the time-line for ECHO packets. Elapsed time "A" is the round trip time measured by the Tekelec on the client side. It is the combined latency of the GIGAswitch and the server workstation. Elapsed time "B" is the round trip time measured by the Tekelec on the server side. Elapsed time "C" is the client workstation latency. Elapsed time "D" is the combined latency of the client workstation plus the GIGAswitch. We ignored propagation delays in our calculation because they were less than 40 nanoseconds. Ten values were selected from the middle of the two traces for each test category, and the results are tabulated in Tables 12 through 18. Tests are categorized by the RTT ECHO packets sizes.

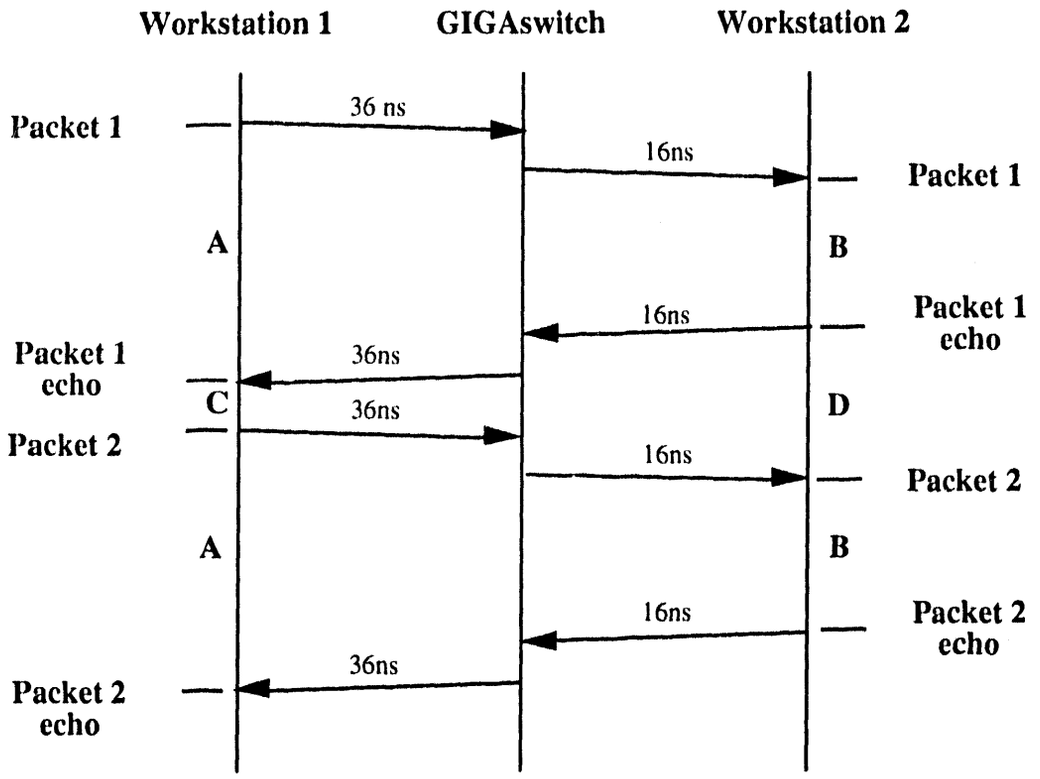


Figure 9. Time line for RTT ECHO packets.

Table 12. Latency Breakdown - (64 Byte)

Packet Number	RTT(send) microseconds A	Interpacket(send) microseconds C	RTT (recv) microseconds B	Interpacket(recv) microseconds D
1	1828	481	1522	517
2	1736	484	1566	515
3	1548	481	1425	509
4	1481	481	1699	515
5	1603	481	1576	512
6	1607	489	1464	507
7	2010	481	1573	512
8	1602	481	1563	514
9	1512	481	1555	512
10	1621	618	1522	512
Average	1655	496	1547	513

Table 13. Latency Breakdown (128 Byte)

Packet Number	RTT (send) (microseconds) A	Interpacket (send) (microseconds) C	RTT (recv) (microseconds) B	Interpacket(recv) (microseconds) D
1	1530	1029*	1595	519
2	1404	496	1668	529
3	1561	503	1766	525
4	1787	492	1786	520
5	1642	496	1494	520
6	1614	491	1657	525
7	1900	492	1595	528
8	1627	497	1668	520
9	1730	489	1587	530
10	1705	494	1766	650
Average	1650	494	1658	537

*This value is dropped, because the CPU is probably processing non-related tasks.

Table 14. Latency Breakdown (256 Byte)

Packet Number	RTT (send) (microseconds) A	Interpacket (send) (microseconds) C	RTT (recv) (microseconds) B	Interpacket (recv) (microseconds) D
1	1805	515	1629	543
2	1568	504	1713	539
3	1664	510	1713	535
4	2008	643	1735	542
5	1494	505	1717	537
6	1802	507	1704	537
7	1870	507	1765	540
8	1732	510	1758	535
9	1636	506	1619	537
10	1821	510	1781	558
Average	1740	522	1713	540

Table 15. Latency Breakdown (400 Byte)

Packet Number	RTT (send) (microseconds) A	Interpacket (send) (microseconds) C	RTT (recv) (microseconds) B	Interpacket (recv) (microseconds) D
1	1846	535	1866	561
2	2018	936	1813	695
3	1890	669	1795	561
4	1893	532	1736	564
5	1777	679	1813	1764*
6	1907	535	1954	562
7	1719	529	1733	561
8	1849	531	1751	579
9	1867	529	1823	567
10	1882	532	1925	567
Average	1865	601	1821	580

*This value is dropped, because the CPU is probably processing non-related tasks.

Table 16. Latency Breakdown (500 Byte)

Packet Number	RTT (send) (microseconds) A	Interpacket (send) (microseconds) C	RTT (recv) (microseconds) B	Interpacket (recv) (microseconds) D
1	1920	555	1808	588
2	1884	554	1800	585
3	1635	555	1846	580
4	1751	552	1844	578
5	1856	556	1851	580
6	1767	548	1859	580
7	1816	551	1818	578
8	1883	547	1861	721
9	1865	546	1920	583
10	1950	546	1800	585
Average	1833	551	1841	596

Table 17. Latency Breakdown (512 Byte)

Packet Number	RTT (send) (microseconds) A	Interpacket (send) (microseconds) C	RTT (recv) (microseconds) B	Interpacket (recv) (microseconds) D
1	24559	683	18072	725
2	17983	1235*	17839	717
3	17800	685	17895	716
4	18117	683	17957	721
5	18095	685	18205	729
6	17845	693	17882	721
7	17872	688	24526	715
8	18015	955*	18115	717
9	17589	550	17928	717
10	17964	683	18122	717
Average	18584	669	18654	720

*This value is dropped because the CPU is probably processing non-related tasks.

Table 18. Latency Breakdown (1024 Byte)

Packet Number	RTT (send) (microseconds) A	Interpacket (send) (microseconds) C	RTT (recv) (microseconds) B	Interpacket (recv) (microseconds) D
1	18088	780	17476	807
2	18175	774	24781	797
3	18144	775	18189	799
4	18271	769	18073	804
5	19045	770	18755	806
6	18274	768	18086	798
7	18133	775	18133	804
8	18095	778	18082	804
9	25349	778	18201	809
10	18361	781	18090	1495*
Average	18904	775	18787	803

*This value is dropped because the CPU is probably processing non-related tasks.

The GIGAswitch RTT in the forward direction was calculated by subtracting elapsed time "B" from "A" (see Figure 9), and the GIGAswitch RTT in the echo direction was calculated by subtracting elapsed time "C" from "D". The workstation RTTs are the interpacket times on the ECHO client and server, and they are the protocol processing overhead of the client and server workstation. Table 19 summarizes the GIGAswitch and workstation RTTs for packets sizes of 64, 128, 256, 400, 500, 512, and 1024 bytes. Our result indicates that multiple clients contend for not only the output port bandwidth of the GIGAswitch but also for the CPU cycle, the I/O bus bandwidth, and the memory bandwidth of the server workstation. As Table 19 shows, the performance bottleneck at the workstation is 50 times more severe than it is at the GIGAswitch.

Table 19. Summary Latency Analysis (microseconds)

Packet Size	GIGAswitch RTT (forward)* A - B	GIGAswitch RTT (echo)* D - C	The ECHO Client Latency C	The ECHO Server Latency B
64	108	17	481	1547
128	-8	43	494	1658
256	27	18	522	1713
400	44	-21	601	1824
500	-8	45	551	1841
512	70	51	669	18654**
1024	207	28	775	18787**

* The GIGAswitch RTT values were calculated from two separate Tekelec traces. Because of the dynamic nature in network traffic, some of the calculated results are negative. Therefore, the results are to be used as indications and not absolute values.

** The sharp degradation in RTT performance is specific to the RTT ECHO server workstations and they are not included in our final analysis.

IV. Conclusions

One of Sandia's research efforts is to reduce the end-to-end communication delay in a parallel distributed computing environment. GIGAswitch is DEC's implementation of a gigabit local area network based on switched FDDI technology. Using the GIGAswitch, we intend to minimize the medium access latency suffered by shared-medium FDDI technology. Experimental results show that the GIGAswitch adds 16.5 microseconds of switching and bridging delay to an end-to-end communication. Although the added latency causes a 1.8% throughput degradation and a 5% line-efficiency degradation, the availability of the dedicated bandwidth is much more than what is available to a workstation on a shared medium. For example, ten directly connected workstations each would have a dedicated bandwidth of 95 Mbps, but if they were sharing the FDDI bandwidth each would have 10% of the total bandwidth, i.e., less than 10 Mbps. In addition, we have found that when there is no output port contention, the switch's aggregate bandwidth will scale up to multiples of its port bandwidth. However, with output port contention, the throughput and latency performance suffer significantly. Our mathematical and simulation models indicate that the GIGAswitch line efficiency could be as low as 63% when there are nine input ports contending for the same output port. In a distributed parallel computing environment, output contention often involves the contention for a server's resources . Our data indicate that the delay introduced by contention at the server workstation is 50 times that introduced by the GIGAswitch.

We conclude that the GIGAswitch meets the performance requirements of today's high end workstations and that the switched FDDI technology provides an alternative that utilizes existing workstation interfaces while increasing the aggregate bandwidth. However, because the FDDI standard limits the available bandwidth to 100 Mbps and the speed of the workstation is increasing by a factor of 2 every 1.5 years, the switched FDDI technology is only good as an interim solution.

References

[1] Stallings, W., *Data and Computer Communications*, Macmilla Publishing Company, p409-p413 (1991).

[2] GIGAswitch System Manager's Guide, Part No. EK-GGMGA-MG.A01 (July 1993).

[3] RFC 1157, Case, J. D., Fedor, M., Schoffstall, M. L., Davin, C., *Simple Network Management Protocol (SNMP)*. (May 1990).

[4] RFC 793, Postel, J. B. *Transmission Control Protocol* (September 1981).

[5] RFC 768, Postel, J. B. *User Datagram Protocol* (August 1980).

[6] Digital Technical Journal, Vol. 3 No. 3 (Summer 1991).

[7] Hui, J. Y., Arthurs, E., *A.*, *Broadband Packet Switch for Integrated Transport*, IEEE Journal on Selected Areas in Communications (October 1987).

UNLIMITED RELEASE

INITIAL DISTRIBUTION:

1900	D. L. Crawford
1901	R. E. Palmer
1902	N. R. Morse
1932	C. D. Brown
1951	P. W. Dean
1951	F. T. Bielecki
1951	J. M. Brandt
1951	H. Y. Chen (10)
1951	C. Fang
1951	J. A. Hutchins
1951-1	D. H. Ching
1952	R. E. Cline
1954	M. O. Vahle
1954	S. A. Gossage
1954	J. H. Naegle
1954	L. G. Pierson
1954	T. J. Pratt
1954	N. Testi
1954	B. Whittet
1957	W. D. Swartz
8000	J. C. Crawford
	Attn: E. E. Ives, 5200
	J. B. Wright, 5300
	M. E. John, 8100
	R. J. Detry, 8200
	W. J. McLean, 8300
	L. A. Hiles, 8400
	P. E. Brewer, 8500
	L. A. West, 8600
	R. C. Wayne, 8700
8535	Publications for OSTI (10)
8535	Publications/Technical Library Processes, 7141
7141	Technical Library Processes Department (3)
8523-2	Central Technical Files (3)

DATE
FILMED
11/17/93

END