Proceedings on the 15th CISL Winter Workshop · Kushu, Japan · February 2002

A survey on Reflective Memory Systems

Il Joo Baek

Control Information Systems Lab., School of Electrical Engr. and Computer Science, Seoul National University, Seoul, 151-742, Korea

Abstract: In this paper, several reflective memory systems (RMS) are surveyed. Differences between the reflective memory systems and the shared memory systems are described. A brief overview of architectures, advantages, disadvantages, history, and general features of each system is provided. The various RMS products are compared by complexity, scalability, and compatibility. Recent researches of each systems are also studied. This paper suggested methods to improve performance of RMS.

Keywords: Reflective Memory Systems; Distributed Shared Memory;

1 Introduction

Research and development to achieve high computing power with multiple computers over network have shown significant progress recently. However, an efficient interconnection among message-passing multicomputers is still hard to achieve for several reasons. First, the overhead of operating-system increases the communication latency. Second, the layered protocol software further consumes the processor time. Finally, predicting the communication latency is difficult. As the number of processors and the speed of network increases, these problems become more critical. The reflective memory system (RMS) is one of well known solutions to these problems. It is based on automatic updates of remote shared-memory copies. In the RMS, writes to memory are automatically distributed to other connected systems. Memory reads are executed on the local memory [1].

For the past 10 years, many RMS systems have been proposed, developed, and commercialized. Each of them was designed to satisfy specific requirements and purposes. Therefore, it is difficult for users to choose suitable and efficient RMS system for their purpose. Also, researchers have wasted their time to get the information of many RMS systems and to compare each other. At this point, the investigation of present RMS technologies and comparison of each system is very important to give helpful guideline for user decision-making. Also, this survey suggests proper direction of the RMSs to developers that propose better solutions and improve performance.

In this paper, a brief overview of architectures, advantages, disadvantages, history, and general features of following systems is provided. Encore RMS [2], reflective memory / memory channel (RMMC+) [1], mirror memory multiprocessor (MMM) [3], University of Tokyo's RMS [4], memory channel for peripheral component interconnection (MC for PCI) [5], network shared memory (NSM) [6], shared common RAM network (SCRAMNET) [7], VME Microsystems International corporation (VMIC)'s VMIC RM [8], finally scalable high performance really inexpensive multiprocessor (SHRIMP) [9]. Those RMSs are compared by complexity, scalability, and compatibility. Recent researches of each systems are also studied. Finally it suggested methods to improve performance of RMS.

This paper is organized as follow. Section 2 describes the conceptual overview of distributed shared memory (DSM) and RMS. It also provides a brief overview of architectures, general features, advantages, disadvantages, and history of each system. In section 3, RMSs

are compared by complexity, scalability, and compatibility. Section 4 discusses and predicts possible area for future work. Section 5 presents concluding remarks.

2 Reflective Memory System

2.1 Overview

Multiprocessors systems fall into two large classifications: shared-memory systems and distributed-memory systems. Shared-memory systems is a tightly coupled multiprocessor system, consisting of multiple CPUs and a single global physical memory. This memory systems offer a general and simple programming model. Users can readily emulate other programming models on these system. However a shared-memory multiprocessors typically suffer from increased contention and longer latencies in accessing the shared memory, which degrades peak performance and limits its scalability compared to distributed-memory multiprocessors. In contrast, a distributed-memory systems(often called a multi-computer) consist of multiple independent processing nodes with local memory module, connected by a general interconnection network. Consequently, it makes systems with very high computing power possible. Also, its hardware implementation is easer. However, the implement of software is more complex and it requires explicit use of send/recive primitives because communication between nodes involves a message-passing model. To recover these shortcomings, Distributed-Shared-Memory (DSM) is invented combining the advantages of the two approaches. A DSM system logically implements the shared-memory model on physically distributed-memory system [10][11][12][13][4].

RMS is a branch of distributed shared-memory (DSM). Therefore it has advantages of both shared-memory and distributed-memory. The RMS is defined as a distributed shared-memory system based on automatic updates using remote share-memory copies. In other words, when a shared data need to be reused, all processor's local memory should keep an exact copy of it. Therefore, the shared reads are always satisfied from the local memory. RMS is sometimes called mirror system, replicated memory system but reflective memory systems are being used more commonly [14].

The characteristic feature of RMS is that each computer physically has its own local memory, the results are the same as if all the computer were attached to a large common memory [2]. The reflective memory is composed of a dual-ported memory physically distributed and logically mapped into a global, shared address space.

RM updates can occur over different types of interconnection networks: bus, bus hierarchy, ring, mesh, or crossbar. The form of sharing is various: page, word, segment or block. Also they can map shared-memory regions dynamically or statically. An RM's memory consistency model (MCM) can be strict, sequential, processor, release, or entry [15][16].

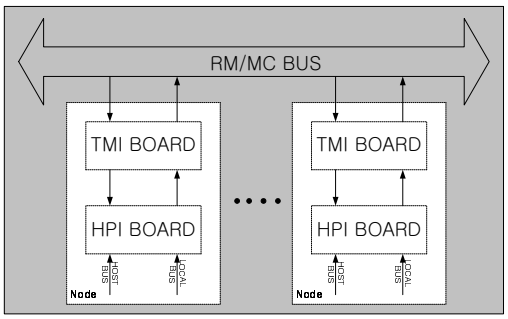

Figure 1: A Reflective Memory/Memory Channel Node

There are several advantages of RMS over other DSM systems. First, computation typically overlaps with communication. Second, memory access time is usually constant and deterministic. Third, RM supports a multiple-reader/ multiple-writer algorithm. Fourth, it has good fault tolerance. Finally, RM systems have been commercially implemented for decades now.

On the other hand, there are also disadvantages. First, RM system might produce unnecessary update traffic for application characterized with longer sequences of writes to the same word. Second, the interconnection medium might suffer from a bottleneck because of often data transfers. Third, one-to-all broadcast communication must be supported. Fourth, its access time is slightly longer than local memory because RM is typically implemented on different board.

2.2 RM/MC

The first RMS was designed and patented by Gould Electronics in 1985 [17]. After Encore Computer Corporation acquired Gould in 1989, RMS for real time application was implemented in 1990 [14] [1].

The Reflective Memory/Memory Channel system(RM/MC) [2] is an improved system compared to previous RMSs and was introduced in 1993. RM/MC was initially designed to satisfy the Online Transactions Processing application [18] needs.

RM/MC is bus-based and consists of up to eight processing nodes connected by the multiplexed, synchronous 64 bits RM/MC bus. RM/MC bus arbitration is centralized and uses a round-robin synchronous arbitration algorithm. Local memory pages are configured as reflective (shared) or private (nonshared) using translation windows on the transmit and receive

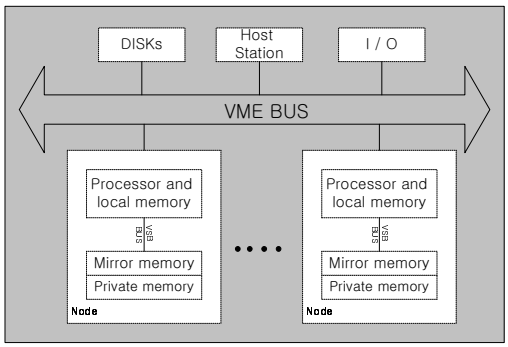

Figure 2: The Mirror Memory Multiprocessor

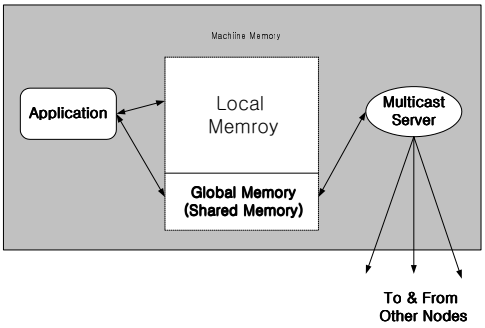

Figure 3: A Widely Distributed RMS

sides. The RM/MC provides receive and transmit FIFO buffers to make asynchronous transfers between the RM/MC bus and the host.

Advantages of RM/MC system are as follows. While previous RMS supports word updates only, RM/MC supports both word and block updates. RM/MC combines potentially high bandwidth with low latency time. The broadcast mechanism is easy to implement and update messages are small. Besides, propagation time for word update messages is small because the interconnection medium spends only two bus cycles to transfer an update messages to all nodes [14]. Finally, the RM/MC tolerates node failures without service disruption. Therefore RM/MC supports numerous real-time applications.

Disadvantages of RM/MC system are as follows. By a structural defect, RM/MC doesn't scale well and disable cache. Also it has high cable complexity. When short messages (word) and long messages (block) share the same FIFO buffers, short messages must wait until long ones are sent. Therefore nodes might suffer from starvation.

A University of Belgrade group and Encore introduced upgraded RMS for personal-computer (PC) network. RMS for PC has better scalability than RM/MC. It reduces RM bus traffic and alleviates the memory-contention problem.[19]

RM/MC++ is another project in cooperation with Encore and a University of Belgrade group[19]. It was proposed to enhance RM/MC system performance. The main idea of RM/MC+ is to minimize the delays that occurs when short messages wait until long ones are sent in the same FIFO buffers.

This system supports numerous real-time applications including vehicle simulation, telemetry, instrumentation, and nuclear power plant simulation and control. Especially, RM/MC is designed to satisfy the online transactions processing applications (OTPL) needs.

2.3 MMM

MMM stands for the mirror memory multiprocessor. This system was proposed by MODCOMP in 1997 [3]. It is designed for time-critical applications which require both the high computational performance and real-time performance.

MMM consists of a host computer that controls the over all system operation and up to eight nodes. Each node consists of a single-board computer and a memory board. These boards are connected to two buses, the VERSA module eurocard bus (VME) and VME subsystem bus (VSB). This system uses data and interrupt broadcast mechanisms by using VME bus slave function and location monitors. The bus slave function is created by logic in a dual port memory and it supports broadcast mechanism. The location monitors is contained in memory and generates interrupts across the VSB bus to the target computer.

This system is strong for hard and soft real-time applications; all tasks complete within their deadlines. In addition, it supports a variety of high-performance industry standard I/O buses, interfaces, and protocols.

While, MMM does not scale well compared to other bus-based multiprocessors. Architectural weakness is VME bus's dual-fuction; VME bus serves as a medium for maintaining the RM coherence and also as the system bus.

MMM supports numerous hard and soft real-time applications. For example, factory automation, process control, Supervisory control and data acquisition, data communication, and simulation and trainer application.

2.4 A Widely Distributed RMS

A University of Tokyo group has proposed widely distributed replicated shared memory [4] in 1994. It is designed for a small number of nodes distributed over a wide area.

Each node maintains a copy of the global shared-memory space on local memory. A multi-cast server broadcasts write accesses to all nodes.

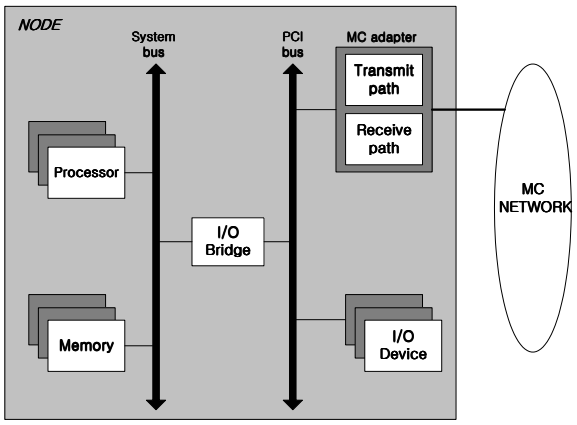

Figure 4: The Memory Channel Network for PCI architecture

This RSM's Memory Consistency Model (MCM) is looser than that in typical RM system.

In this system, multicast server is completely realized in software, it will create a software overhead problem.

It is especially suitable for applications that impose real-time operations in a widely distributed environment. This is because other latency hiding techniques such as context switching or prefetching are not always effective for real time operation.

2.5 Memory Channel Network for PCI

Digital Equipment Corporation designed a Memory Channel for peripheral component interconnect (PCI) in 1996 [5]. It was designed to enhance a cluster's parallel performance and high availability.

The basic network primitive is a memory-mapped circuit that provides a write-only connection between a page of virtual-address space on a transmitting node and a page of physical memory on a receiving node [20]. Also, it uses crossbar interconnection network and supports page-level connection granularity.

MC supports several connection models, including point-to-point, multicast, and broadcast. In MC, a sever can transmit data directly to the requesting nodes without affecting its local memory. It also lets receiver nodes send acknowledgments and implements an innovative remote-read primitive as two write transfers without software intervention.

MC was designed with homogeneous clusters, it dose not support heterogeneous computing, in spite of the fact that all computers incorporating the PCI bus can be connected using this approach. Also cross bar interconnection network limits the number of system nodes.

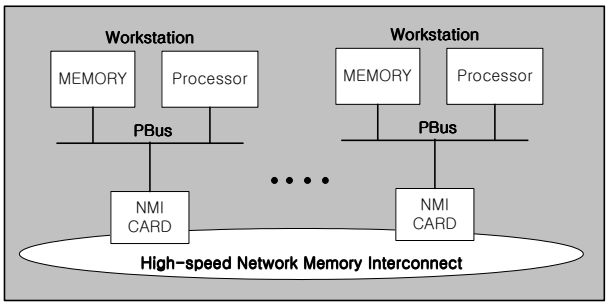

Figure 5: Network Shared Memory

2.6 Network Shared Memory

Network shared-memory (NSM) was proposed by architecture technology corporation (ATC) in 1994 [6]. It is a low-cost approach for clustering workstations into a single, shared memory mid-range parallel computer.

Nodes are connected in a unidirectional slotted ring by high speed optical links. Network Memory Interface (NMI) is used to interconnect the processing elements of the workstations into a parallel computer. Memory update size is a word and its MCM is sequential consistency. The system is successful up to 60 processors.

The main advantage of this system is to provide hardware support for synchronization by implementing a separate synchronization ring. However, NSM does not support overlapping computation with communication to ensure memory consistency. It is being used in CAD, weather data processing, and parallel databases.

2.7 SCRAMNET+

Shared common RAM network (SCRAMNET)[21] is developed by Systran Corporation in 1989 and is advanced into SCRAMNET+ [7].

It is ring based RM products for realtime applications. Up to 256 nodes can be connected via a single fiber optic ring. Each node keeps its own copy of the entire RM space. Data can be transmitted up to 3500 meters over fiber optic cable and 30 meters over coaxial cable, respectively [22].

This system has various advantages. To improve the effective network throughput, SCRAMNET+ provides variable length packets. Also the network board is designed to be medium independent. Another advantage is data filtering in which only data-value changed writes broadcast to other nodes.

While it has many advantages, there are a few disadvantages. Its reflective memory is limited to 8Mbytes because its standard specifies a fixed number of address lines. Moreover each node keeps a copy of the entire RM space.

SCRAMNET+ is being applied for many realtime

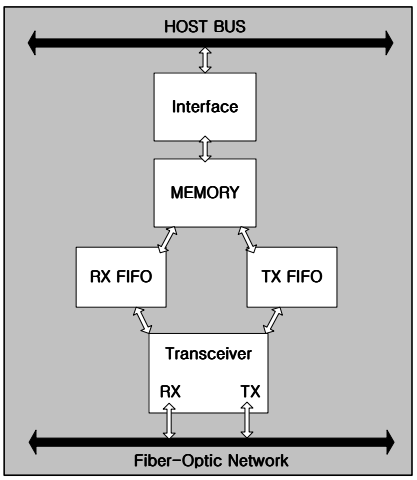

Figure 6: A VMIC RM Network

application including aircraft, land vehicle, missile simulation, robotics, data acquisition, and virtual reality.

2.8 VMIC RM Network

VME microsystems international corporation (VMIC) has produced VMIC RM network family since 1995 [8]. The VMIC network consists of five product families - the 5550 family, the 5560 family, the 5570 family, the 5580 family, and 5590 family.

Each RM board is configured with on-board SDRAM. Writes are stored in local SDRAM and broadcast over a high-speed fiber-optic data path to other RM nodes. It allows data to be shared between up to 256 independent nodes at rates up to 174 Mbyte/s.

The VMIC RM network supports a number of popular system buses. Programmable Logic Controller (PLC) can networked in this system. It also uses static random access memory to implement onboard RM. Therefore, it can provide fast read access time to stored data.

There are a few disadvantages with this system. First, each node keeps its own copy of the entire RM space. Second, the VMIC RM network does not provide data filtering. Third, the system interrupt to read just-received data increases software overhead.

This RM network is designed for many realtime applications including aircraft, ship, submarine, power plant simulators, and data acquisition.

2.9 SESAME

MERLIN stands for memory routed, logic interconnection network which is developed by Sandia National Laboratories and the State University of New York, Stony Brook. To improve performance of MERLIN system, scalable eagerly shared memory (SESAME) is proposed in 1991 [14].

SESAME's type of interconnection is not fixed. However, the prototype implements a fiber-optic 2D mesh. The path of an update is not carried with data, but rather determined by the global virtual address. An update size is a word. The system permits multiple writes within a multicast group. Also, sequence numbers are assigned to packets to enforce arrival in the original write order [23].

This system is designed for heterogeneous networks. Only 10 percents of the system's node interface is processor dependent. Therefore, it can easily be ported to numerous different processors. To gain higher bandwidth, the system can merge single word packets with consecutive addresses. SESAME determines routes for update messages statically. However, based on a compile analysis, it can dynamically disable sharing for temporary data changes.

SESAME does not support real block transfers generated by direct memory access (DMA) unit. Also it cannot correct transmission errors successfully because data is transformed into bit-serial streams for fiber optic channel.

2.10 SHRIMP

In 1994, Princeton University introduced scalable high performance really inexpensive multiprocessor (SHRIMP) [9]. SHRIMP is based on a custom-designed virtual-memory-mapping network interface. The network interface maps the virtual memory of a sending process to the virtual memory of a receiving process. A network-interface page table holds information about the mapping [24][25].

The system supports both automatic and deliberate updates. Consecutive writes buffers in the transmit FIFO are merged in the same automatically updated block packet. Also caching is supported in the system. The main disadvantage to SHRIMP is that does not support broadcast and multicast communication models [26][27].

3 Comparison

Complexity of system means how hard it is to implement system in both hardware and software. This factor is very important for commercializing because it is directly involved in cost. Kinds of interconnection network, shape of cable connection, and the number of necessary chips on the board were considered to decide complexity of each system. Scalability influences on the limit of possible application area both directly and indirectly. In this paper, low scalability means the

Table 1: Complexity, Scalability and Compatibility of RMSs

System	Complexity	Scalability	Compatibility
RM/MC++	high	low	high
MMM	low	low	low
RSM	low	low	med
MC	low	low	low
NSM	high	med	high
ScramNet+	low	high	high
VMIC	low	high	high
Shrimp	med	high	high

system can use less than 8 processors, medium scalability means more than 32 processors can be attached to network, and over 256 processors in system represents high scalability. Compatibility is also critical factor when users adapt the system to their environment. It can reduce extra cost for adapting other system to each different purpose and system environment. Table 1 compares complexity, scalability, and compatibility of each RMS.

4 Discussion

Table 2 shows each RMS's interconnection network and main application area. Also, it identifies what company or university has involved proposing and developing the RMSs. As surveyed so far, various RMSs have proposed and commercialized for decade now. However there are still topics deserve further investigation. Data filtering will reduce update traffic and memory contention increasing interconnection network utilization. Node prioritization can lower transient overload on both transmit side and receive side, unnecessary waiting. Therefore it is able to increase system performance and usability. Hierarchy of RM buses increases system scalability. Dynamic RM mapping can get rid of the need for a copy of the entire RM space and will bring better utilization of available memory. Caching RM region might significantly decrease memory access time. The hardware support for heterogeneous computing improves system usability. Hardware broadcast mechanism is able to improve system performance and alleviate MCM implementation. Separating system bus for RM updates only might be able to reduce propagation time of update message.

5 Conclusion

This paper surveyed history, architecture, general features, advantage, disadvantage, and application area of

Table 2: Reflective Memory Systems

System	Network	Application	Company
RM/MC++	Bus	*OLTP*[1]	Encore
MMM	Bus	Real time	Modcomp
RSM	Bus	Real time	Tokyo
MC	Crossbar	Client-server	DEC
NSM	Ring	*S&E*[2]	ATC
ScramNet+	Ring	Real time	Systran
VMIC	Ring	Real time	VMIC
Shrimp	Mesh	Client-server	Princeton

[1] *OLTP* online transaction processing
[2] *S&E* scientific and engineering

recent RMS technologies. Also, complexity, scalability, and compatibility of each RMS was compared. The result of this paper could be used by users who need set up the RMS for their purpose and by researchers who study about the RMS. Since the RMS designed for industrial environment had short history, this paper did not provide comparison of each RMS's performance.

The survey on RMS would be used to propose improved RM model for the industrial environment.

References

[1] "Reflective Memory Specifics," *ENCORE www.encore.com/products/hardware/reflecive.*

[2] I. Lucci, S.; Gertner, "Reflective-memory multiprocessor," *System Sciences. Vol. II., Proceedings of the Twenty-Eighth Hawaii International Conference on*, vol. 1, pp. 85 –94, 1995.

[3] B. Furht, "Architecture and Performance Evalution of the MMM," *IEEE IC Computer Architecture Newsletter, Mar.*, vol. 1, pp. 66–75, 1997.

[4] H.; Oguchi, M.; Aida, "A proposal for a DSM architecture suitable for a widely distributed environment and its evaluation," *High Performance Distributed Computing, 1995., Proceedings of the Fourth IEEE International Symposium on*, vol. 7, pp. 32 –39, 1995.

[5] Gillett R.; Collins M.; Pimm D., "Overview of memory channel network for PCI," *Compcon '96. 'Technologies for the Information Superhighway'*, pp. 244–249, 1996.

[6] Ramanjuan R.S.; Bonney J.C.; Thurber K.J., "Network shared memory: a new approach for

clustering workstations for parallel processing," *High Performance Distributed Computing, 1995., Proceedings of the Fourth IEEE International Symposium*, pp. 48–56, 1995.

[7] Systran Corporation, "SCRAMNet+ Shared Memory – Speed, Determinism, Reliability, and Flexibility for Distributed Real-Time Systems," *www.systran.com.*

[8] VME Microsystems Int'l Corp., "VMIC's Reflective Memory Network," *www.vmic.com.*

[9] et al. M. A. Blumrich, "Virtual memory mapped network interface for the SHRIMP multicomputer," in *Proc. of the 21th Annual Int'l Symp. on Computer Architecture*, 1994, pp. 142–153.

[10] V. Nitzberg, B.; Lo, "Distributed shared memory: a survey of issues and algorithms," *Computer*, vol. 24, pp. 52 –60, 1991.

[11] M. Protic, J.; Tomasevic, "Distributed shared memory: concepts and systems," *IEEE Parallel and Distributed Technology: Systems and Applications*, vol. 4, pp. 63 –71, 1996.

[12] Brendan Tangney Alan Judge, Paddy Nixon, Vinny Cahill and Stefan Weber, "Overview of distributed shared memory," Tech. Rep., 1998.

[13] C. Amza, A. L. Cox, S. Dwarkadas, P. Keleher, H. Lu, R. Rajamony, W. Yu, and W. Zwaenepoel, "Treadmarks: Shared memory computing on networks of workstations," *IEEE Computer*, vol. 29, no. 2, pp. 18–28, 1996.

[14] V. Jovanovic, M.; Milutinovic, "An overview of reflective memory systems," *IEEE Concurrency*, vol. 7, pp. 56 –64, 1999.

[15] V. Protic, J.; Milutinovic, "Reflective Memory System Based on a Grid of Buses," *21st international conference on microeledtronics Sep. 1997*, vol. 2, 1997.

[16] Kai Li and Paul Hudak, "Memory coherence in shared virtual memory systems," in *Proceedings of the 5th ACM Symposium on Principles of Distributed Computing (PODC)*, New York, NY, 1986, pp. 229–239, ACM Press.

[17] Christopher Wilks, "SCI-Clone/32 - A Distributed Real Time Simulation System," *Computting in High Energy Physics, edited by Hertzberger and Hoogland, North Holland Press*, vol. 1, pp. 416–422, 1986.

[18] C. Leff, A.; Pu, "A classification of transaction processing systems," *Computer*, vol. 24, pp. 63 –76, 1991.

[19] Jovanovic M.; Tomasevic M.; Milutinovic V., "A simulation-based comparison of two reflective memory approaches," *System Sciences. Vol. II., Proceedings of the Twenty-Eighth Hawaii International Conference*, vol. 1, pp. 140 –149, 1995.

[20] Gillett R.B., "Memory channel network for PCI," *IEEE Micro*, vol. 1, pp. 12–18, 1996.

[21] Systran Corporation, "SShared-Memory Networking Architectures – Simplicity and Elegance," *www.systran.com.*

[22] Systran Corporation, "SCRAMNet+ Overview ," *www.systran.com.*

[23] C. Maples, "A high-performance memory-based interconnection system for multicomputer environments," 1992.

[24] M. A. Blumrich and R. D. Albert, "Design choices in the SHRIMP system: An empirical study," in *Proc. of the 25th Annual Int'l Symp. on Computer Architecture*, 1998.

[25] Angelos Bilas and Edward W. Felten, "Fast RPC on the SHRIMP virtual memory mapped network interface," *Journal of Parallel and Distributed Computing*, vol. 40, no. 1, pp. 138–146, 1997.

[26] E. W. Felten, R. D. Alpert, A. Bilas, M. A. Blumrich, D. W. Clark, S. M. Damianakis, C. Dubnicki, L. Iftode, and K. Li, "Early experience with message-passing on the SHRIMP multicomputer," in *Proc. of the 23rd Annual Int'l Symp. on Computer Architecture (ISCA'96)*, 1996, pp. 296–307.

[27] Stefanos N. Damianakis, Cezary Dubnicki, and Edward W. Felten, "Stream sockets on SHRIMP," in *Communication, Architecture, and Applications for Network-Based Parallel Computing*, 1997, pp. 16–30.