2004
NASA FACULTY FELLOWSHIP PROGRAM

MARSHALL SPACE FLIGHT CENTER

UNIVERSITY OF ALABAMA
UNIVERSITY OF ALABAMA IN HUNTSVILLE
ALABAMA A&M UNIVERSITY

NASA Integrated Vehicle Health Management (NIVHM) A New Simulation Architecture: An Investigation PART I

Prepared by:	Gene Sheppard
Institution:	Georgia Perimeter College Math/Computer Science Dept.
NASA/MSFC Directorate:	ED-19
MSFC Colleague:	Leslie Murphy

Introduction

The overall objective of this research is to explore the development of a new architecture for simulating a vehicle health monitoring system in support of NASA's on-going Integrated Vehicle Health Monitoring (IVHM) initiative. As discussed in NASA MSFC's IVHM workshop on June 29-July 1, 2004, a large number of sensors will be required for a robust IVHM system. The current simulation architecture is incapable of simulating the large number of sensors required for IVHM. Processing the data from the sensors into a format that a human operator can understand and assimilate in a timely manner will require a paradigm shift. Data from a single sensor is, at best, suspect and in order to overcome this deficiency, redundancy will be required for tomorrow's sensors. The sensor technology of tomorrow will allow for the placement of thousands of sensors per square inch. The major obstacle to overcome will then be how we can mitigate the torrent of data from raw sensor data to useful information to computer assisted decisionmaking.

We can take our cue from the human sensory apparatus. We should consider how the human mind processes and assimilates the large sensor input. For example, consider the following [1, 2, 3, 4, 5]:

- The average human male has about 18,000 cm^2 of skin surface area.
- The number of pressure/pain sensors per cm^2 ranges from 50 on the back to 2,500 on the hand.
- Given an average of 1,000 sensors per cm^2, there are about 18,000,000 skin sensors constantly sending information to the brain 24/7.
- The human eye has about 125,000,000 photoreceptors constantly simulating the brain 24/7.
- The human nose has about 40,000,000 olfactory sensors sending input directly to the brain 24/7.
- Human nerve impulses travel about 0.1 to 100 meters per second.

For the sake of simplicity, we will presume each sensor sends a one or a zero. Roughly every 200 milliseconds about 183,000,000 bits (22,875,000 bytes) of information is processed. This is simply raw data being pushed around without regard for content or context. The wondrous part of all this is, the conscious mind is presented with only "pertinent" data and in a format easily assimilated. This suggests some intelligent processing occurs before alerting the conscious mind. Processing of sensor data just above the sensors is indicated by the reaction of a hand when fire is encountered. Before the fire-pain message reaches the brain, the hand is beginning to pull away from the fire. Mimicking the human model, IVHM will have the following hierarchy:

Sensors → low level processing → midlevel processing → intelligent decision processing

It is important to note the brain has no central processing unit ("CPU") indicating it is designed for distributed processing.

Beowulf is a distributed processing technology suited for the task of mimicking the human sensory and response apparatus. From a strictly technological perspective, a Beowulf system is ideal to deal with IVHM. Beowulf also promises tremendous potential for cost reduction associated with simulation of a vehicle health monitoring system. Beowulf technology is proposed as an ideal architecture for this computing problem because it can be easily used to simulate the above model and can be easily copied and used for IVHM simulation.

A brief history lesson is in order. The story of *Beowulf*, an old English poem originally composed in perhaps the early eighth century and put to paper around the year 1016, describes how Beowulf subdues a ferocious monster that has spent twelve years killing and eating people. The name Beowulf is applied to a computing technology intended to defeat of the monstrous cost of supercomputing.

Approach

Utilizing Beowulf techniques will allow design/development of a scalable/flexible platform which will easily accommodate future computer technology using heterogeneous computer systems and a communication infrastructure. Beowulf systems typically use 7 foot racks housing about 16 computing elements each having a 3x24x30 inch footprint. The computing nodes are interconnected via fast network hardware and protocol. The pictures on page 8 are examples of such systems.

In this context, flexible translates to the ability to add homogeneous and heterogeneous computing nodes using the same network hardware and software with minimal effort. Scalable means the ability of the software to adapt, incorporate new computing nodes, and redistribute of computing resources.

We began our research by examining the flexibility component and transfer speeds/latencies of the current network in the MAST lab. We addressed the questions: Can heterogeneous computing nodes be seamlessly integrated (as much as possible)? What transfer latencies are incurred?

Open-source software and network solutions were examined for cost containment purposes. The open-source operating system of choice is Linux. The open-source network software Message Passing Interface (MPI) is supported the best. Both Linux and MPI are also available from commercial resources. The following software variations of MPI were examined: MPICH-1 (implements MPI version 1 - freeware), MPICH2 (implements MPI version 2 - freeware), MPI/RT (RT means real time – freeware and commercial versions), MPI Globus (commercial), Pallas MPI (commercial), LogP-MPI (commercial), and LAM-MPI (Local Area Multiprocessing - freeware).

Of the versions of MPI we examined, the only version we were able to configure and run properly was LAM-MPI. During our tests, the following systems were successfully communicating over an Ethernet network using LAM-MPI:

- 5 SGI single-CPU Octane systems running IRIX 6.5
- 1 SGI four-CPU system running Slackware Linux 2.2
- 1 SGI 10-CPU Origin system running IRIX 6.5 operating system
- 1 Concurrent with two Intel CPUs (4 with hyper-threading) running Redhat Linux

- 1 Dell with a single Intel CPU running Slackware Linux 2.4

It must be noted the SGI implementation of MPI was easily configured and brought up; however, because of SGI's array-services daemon, the systems did not play well with dissimilar computing nodes.

As part of our research, we examined publications, write-ups, and white papers on optical-electrical hardware implementing the reflective memory that would be required in order to significantly lower data transfer latencies between computing nodes. The optical-electrical products we examined were Infiniband, ScramNet, Myrinet, and Memory Channel. Each of the products either has or is planning to implement a version of MPI on its hardware.

Performance tests we conducted proved to be of little value. Ethernet times and functionality were easily overshadowed by optical-electrical reported speeds. For example, moving one byte via Ethernet requires about 500 microseconds and optical-electrical requires less-than or equal to 1 microsecond. Also, reflective memory is not an option on Ethernet. Ethernet will not be a viable option for this environment until speeds matching the optical-electrical systems can be reached implementing reflective memory.

Recommendations

In order to provide NASA with a flexible/scalable simulation system, I recommend utilizing Beowulf technology. My vision for IVHM and future simulation development is pictorially described in Figures 1 and 2. Each node in Figure 1 will have one or more CPUs and collaborate via MPI over an optical-electrical intranet. The nodes can be configured in sub-clusters within a rack or across racks. The sub-clusters can collaborate on a single simulation or execute different simulations. If more computing power is needed, another rack of nodes can be added. I believe the system as depicted in Figure 1 can be expanded to between 2 and 50 computing nodes without extra effort.

Figure 2 shows a Beowulf cluster modeling the human sensory system or IVHM. Each cluster utilizes the same architecture as in Figure 1. This design offers a very flexible/scalable system. As sensor technology advances, computing nodes with the appropriate sensor interface can be added and plugged into the IVHM simulation. Examining and evaluating sensor data as close to the sensor as possible is a paradigm easily incorporated. This is the architecture I highly recommend for IVHM and future simulation.

There are several issues of concern regarding open-source resources and off-the-shelf commodities that must be considered in the planning and design of a simulation system. Those issues include:

1) Use of open source technology. Open source is a good place to begin, but reliable support is found internally only. I have noticed as an open-source technology maturates, volunteer support begins to diminish and industry will commercialize it provided there is a perceived market. Therefore, I highly recommend adoption of both of the following:
    a. Development of an in-house knowledge base for supporting the open-source.

   b. Purchase of commercial adaptations of open-source tools allowing for technical support for as long as warranted.

2) Off-the-shelf computing nodes may not be acceptable because of their large footprint. I recommend using semi-customized, off-the-shelf computing nodes in order to save space. As indicate in Figure 2, space is a resource requiring constant attention.
3) A small 5 to 10 node Beowulf proof-of-concept system should be evaluated for both a simulation and IVHM.
4) Ask reflective memory vendors to show their wares on the above system.
5) Complete simulation/test of architectures; verify simulation accuracy of systems.
6) Verify simulation test-bed against representative workloads.
7) Use simulation test bed to study advanced application workloads and to set requirements for IVHM and future simulation.

Future Goals

My future goals for this research include:

- Continue developing proof-of-concept system for three-dimensional memory model using Beowulf
- Use LAM-MPI in three-dimensional memory model development
- Examine MPI operations with optical-electrical networks if money is available
- Examine MPI coupled with OpenMP, an open source software for shared memory management/processing
- Create a prototype system
- Complete simulation/test of architectures; verify simulation accuracy
- Continue to work with NASA MSFC on IVHM

Conclusion

For the purposes of developing a three-dimensional memory model, the time spent at NASA MSFC has provided me with the practical knowledge to begin my research. This opportunity has enabled me to build upon what had been an abstract vision. To start, I will use the Slackware version of Linux and LAM-MPI to lay the communication foundation and then examine OpenMP for the shared memory (between intra-node CPUs) part of the project.

It is interesting to note my research goal will provide benefits for both NASA's IVHM and simulation projects and myself. I believe IVHM will necessarily have an intelligent component and the results of my research will provide the fundamentals for an intelligent IVHM. I look forward to continuing to support NASA's IVHM project after returning to school and far into the future, hopefully even next summer.

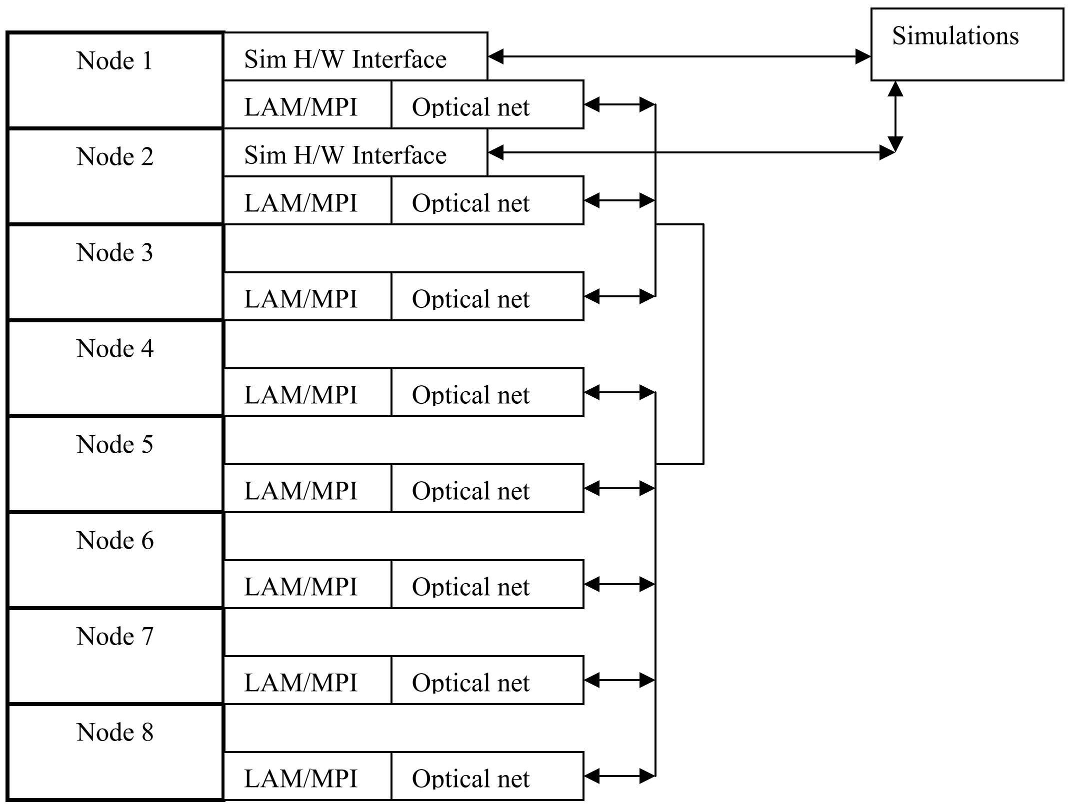

Simulations
Node 1
Sim H/W Interface
LAM/MPI
Optical net
Node 2
Sim H/W Interface
LAM/MPI
Optical net
Node 3
LAM/MPI
Optical net
Node 4
LAM/MPI
Optical net
Node 5
LAM/MPI
Optical net
Node 6
LAM/MPI
Optical net
Node 7
LAM/MPI
Optical net
Node 8
LAM/MPI
Optical net

FIGURE 1

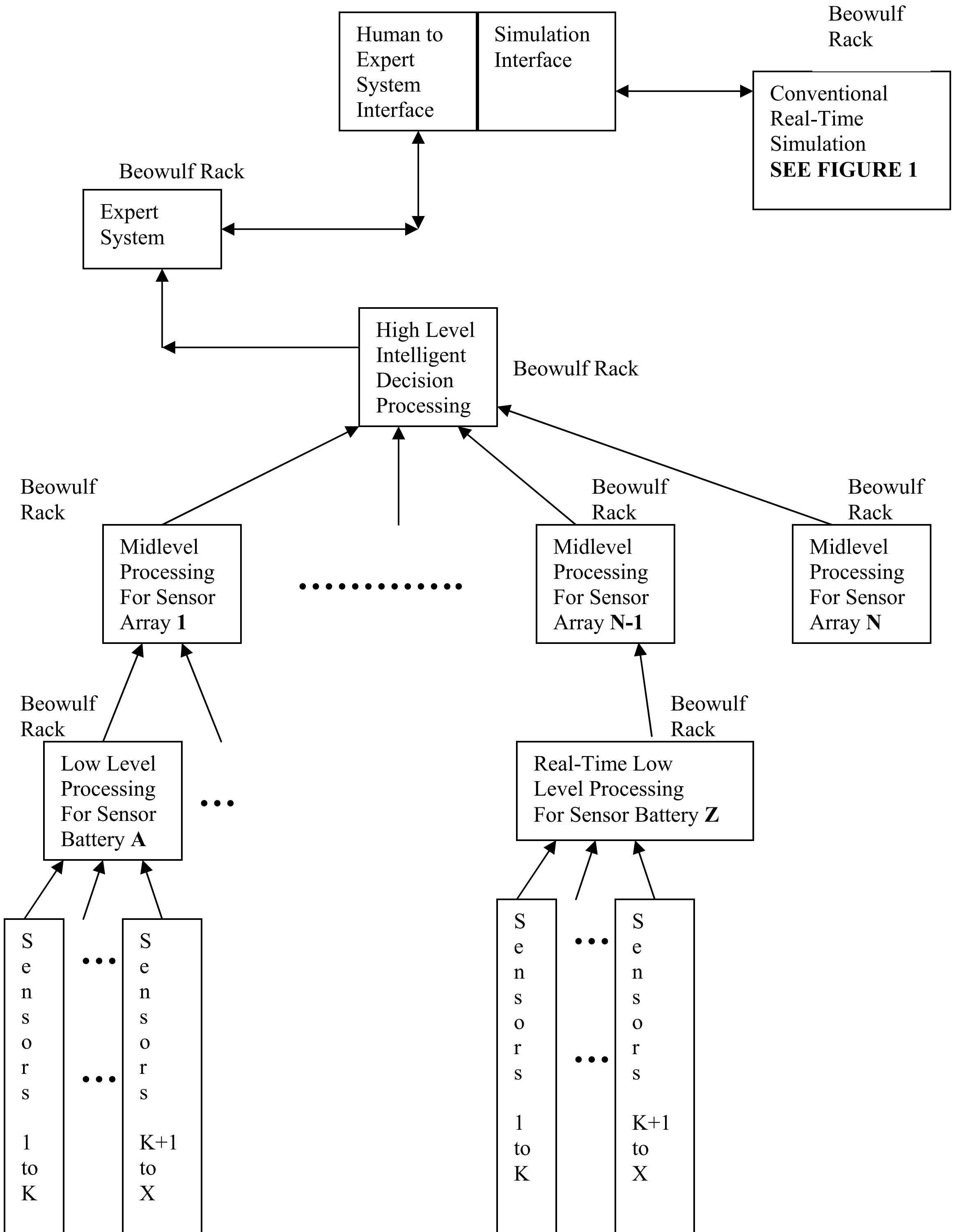
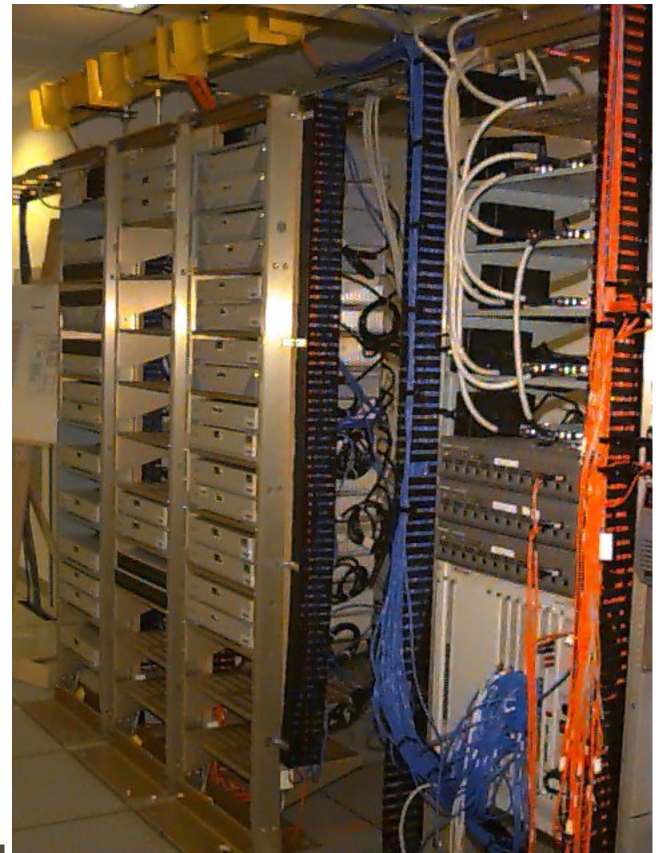

Human to Expert System Interface
Simulation Interface
Beowulf Rack
Conventional Real-Time Simulation
SEE FIGURE 1
Beowulf Rack
Expert System
High Level Intelligent Decision Processing
Beowulf Rack
Beowulf Rack
Midlevel Processing For Sensor Array 1
Beowulf Rack
Midlevel Processing For Sensor Array N-1
Beowulf Rack
Midlevel Processing For Sensor Array N
Beowulf Rack
Low Level Processing For Sensor Battery A
Beowulf Rack
Real-Time Low Level Processing For Sensor Battery Z
Sensors 1 to K
Sensors K+1 to X
Sensors 1 to K
Sensors K+1 to X

Figure 2



BIBLIOGRAPHY

[1] Author unknown. (2004, July 16). Skin: a Living Envelope--Protecting, Sensing, Repairing. Skin Science [On-line]. Available: http://www.skin-science.com/_en/_ww/loreal-skin-science/proteger/organe.aspx?

[2] Nave, Rod. (2004, July 16). Rods and Cones. Hyper Physics Sponsored by Georgia State University [On-line]. Available: http://hyperphysics.phy-astr.gsu.edu/hbase/vision/rodcone.html

[3] Elert, Glenn (Editor). (2004, July 16). Surface Area of Human Skin. The Physics Factbook [On-line]. Available: http://hypertextbook.com/facts/2001/IgorFridman.shtml

[4] Author unknown. (2004, July 16). Nervous System. Biologymad.com [On-line]. Available: http://www.biologymad.com/master.html?http://www.biologymad.com/NervousSystem/NervousSystem.htm

[5] Chudler, Eric H. (2004, July 16). Brain Facts and Figures. Available: http://www.hallym.ac.kr/~neuro/kns/tutor/facts.html

[6] Sterling, T. (Editor). (2002). Beowulf Cluster Computing with Linux. Cambridge, MA: The MIT Press

[7] Pacheco, P.S. (1997). Parallel Programming with MPI. San Francisco, CA: Morgan Kaufmann Publishers

[8] Cheng, Albert M. K. (2002). Real-Time Systems Scheduling, Analysis, and Verification. Hoboken, NJ. Wiley-Interscience, John Wiley & Sons, Inc.

[9] Author unknown. (2004, July) LAM/MPI Parallel Computing [On-line]. Available: http://www.lam-mpi.org/