REE: A COTS-BASED FAULT TOLERANT PARALLEL PROCESSING SUPERCOMPUTER

FOR

SPACECRAFT ONBOARD SCIENTIFIC DATA ANALYSIS

Raphael R. Some, REE Project Chief Engineer, Caltech JPL, (818) 354-3055

rsome@pop.jpl.nasa.gov

David C. Ngo, REE Project Technical Director, Sanders, a Lockheed Martin Co., (603) 885-9555

david.c.ngo@lmco.com

Abstract

NASA's future spaceborne science missions will require supercomputing capabilities for both near earth and deep space exploration. Limited downlink bandwidth and excessive round trip communication delays limit the capabilities and science value of missions which rely on terrestrial supercomputing resources. Projects such as the Gamma ray Large Area Space Telescope (GLAST), the Next Generation Space Telescope (NGST), and autonomous rovers being designed for Mars exploration in the next millennium will require onboard supercomputing capabilities to either enable or to greatly enhance their baseline missions. The difficulty encountered by these projects is that radiation-hardened components are both extremely expensive and lag several generations behind the commercial state of the art. The goal of the Remote Exploration and Experimentation (REE) project, part of NASA's HPCC program, is to migrate ground-based commercial supercomputing technology into space in a timely and cost-effective manner. Reaching this goal will enable new classes of science missions and make feasible the next major thrust in space exploration. The approach being taken on the REE project is to exploit a comprehensive architecture strategy to enable direct insertion of the prevailing generation of state of the art commercial (hardware/software) components in future space systems. The use of state of the art commercial hardware, coupled with a software-based fault tolerance strategy will allow high throughput computation even in the presence of relatively high rates of radiation-induced transient upsets as well as in the presence of permanent faults. Utilization of commercial state of the art software components allows the use of standard software development tools including compilers and debuggers and will simplify and speed the development and porting of application codes to the REE computer and their insertion into space-based systems. In this paper, we outline the overall project plan and status, and review the architecture of the First Generation Testbed, which is currently being fabricated by a team whose members include Sanders, Caltech JPL, The University of Illinois, and MPI Software Technologies.

0-7803-5749-3/99/$10.00 © 1999 IEEE

1. Introduction and Project Vision

Space exploration is moving from the relatively simple, ground-based data processing and human-based real time control paradigm of its early years to a more sophisticated paradigm characterized by onboard science-data processing and autonomous goal-directed operation. The lack of computational throughput and communication bandwidth in remote spacecraft and in earth-orbiting satellites are the major limitations in our ability to execute these next-generation missions. Thus, a key requirement of this paradigm shift will be the provision of an on-board "compute-server" to take on these higher-order, computationally intensive tasks. This space-based supercomputer will, at least initially, not be required to manage the spacecraft itself, and is currently being viewed as an adjunct computational element to the traditional, highly fault tolerant and radiation-hardened Spacecraft Control Computer (SCC).

The commercial computing industry is two orders of magnitude larger than the entire space and defense electronics industry, and each year this disparity grows larger. While great strides are being made in earth-based, commercial-off-the-shelf (COTS) computing and supercomputing technologies, there have been significant difficulties and delays in transitioning even simple general purpose computing technologies to space. Two of the most difficult aspects of computing in space are the susceptibility of hardware to space-radiation-induced upset, and the limited power levels available on spacecraft. Typical power allocations to the computing subsystems of a spacecraft are 100W or less. Radiation-induced transient fault rates in COTS computers can be 1-3 upsets per processor per hour or higher, in even the relatively benign environments of Deep Space (DS) or Low Earth Orbit (LEO). The radiation-hardened computer systems traditionally used in spacecraft, while highly tolerant or immune to radiation-induced damage and transient upset, are several generations behind the COTS technology curve, incur high costs, are high in power consumption, and are at least an order of magnitude lower in performance/power ratio compared to their COTS equivalents.

The Remote Exploration and Experimentation (REE) Project seeks to leverage the considerable investment of the ground-based computing industry to bring supercomputing technologies into space within the constraints imposed by that environment. Further, it seeks to accomplish this at a cost that is commensurate with a COTS solution in a time frame that is not significantly behind the COTS technology curve and which thus enables space-based computing systems to closely follow that curve. The availability of onboard supercomputing capability will enable a new way of doing science in space at significantly reduced overall cost. The vision of the REE Project, therefore, is:

To bring commercial supercomputing technology into space in a form that meets the demanding environmental requirements to enable a new class of science investigation and discovery.

2. Project Goals and Objectives

The REE Project has two principal goals:

1. Develop a process for rapidly transferring commercial high-performance computing technology into ultra-low power, fault-tolerant architectures for space.
2. Validate the notion that high-performance onboard processing capability enables a new class of science investigation and highly autonomous remote operation.

The legacy of the REE Project will not only be a new generation of scaleable onboard supercomputing in space, but the validation of a process that will keep spaceborne computing capabilities on the same technology track as the commercial computing industry.

From the Project Vision and Goals, REE has developed four specific objectives:

1. Demonstrate power efficiencies of 300–1000 MOPS per watt in an architecture that can be scaled up to 100 watts, depending on mission needs.
2. Demonstrate new spaceborne applications on embedded high-performance computing testbeds that return analysis results to the earth (in addition to whatever raw data can be transmitted within the available bandwidth and other communication constraints).
3. Develop fault-tolerant designs that will permit reliable operation for 10 years or more using commercially available or derived components.
4. Investigate ultra-low power onboard computer systems that will help open the entire Solar System to exploration without the need for nuclear technology.

3. Project Plan

Based on the results of an early Study Phase, the REE Project has developed a Technology and Applications Roadmap that leads to the attainment of the Project's goals and objectives and the accomplishment of the project's major milestone - a flight prototype system running space-based scaleable applications in the year 2003. This roadmap is shown in Figure 1. It consists of three parallel tracks: the Computing Testbeds Initiative, the System Software Initiative, and the Applications Initiative. These parallel tracks for the development of technology and applications are of equal importance. It is in the delivery of more science at lower cost that REE finds its ultimate *raison d'être*.

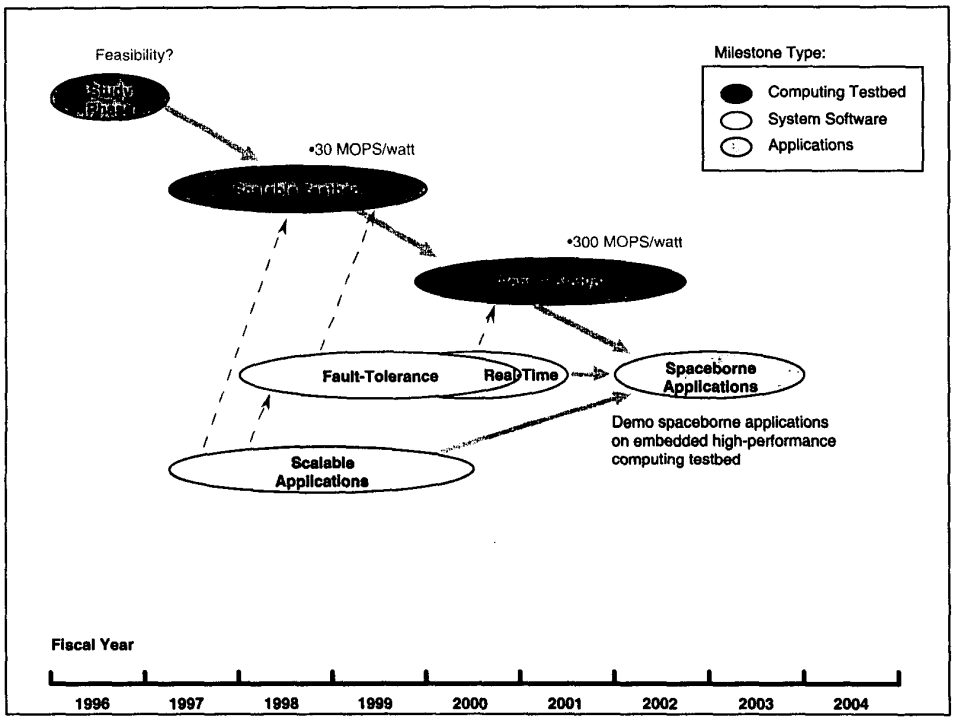

Figure 1. The REE Project Technology and Applications Roadmap calls for the parallel development of hardware testbeds, systems software, and applications.

3.1 *Computing Testbeds Initiative*

The Computing Testbeds Initiative will explore and develop a process for translating commercial high performance scaleable parallel computing architectures into low-power spaceborne implementations. This architecture must rely, to the maximum extent practical, on COTS technologies and must minimize or eliminate the use of radiation-hardened components. The process must be consistent with the rapid (18 months or less) transfer of new earth-based technologies to NASA space missions. Translated architectures must satisfy a number of additional criteria, including no single point of failure and graceful performance degradation in the event of hardware failure.

The Computing Testbeds Initiative will develop a series of hardware prototypes, leading to the demonstration of a capability of at least 300 MOPS[1]/watt. This represents an increase of two orders of magnitude over the power performance of the flight computer onboard the Mars Pathfinder spacecraft which landed on Mars in July 1997. At the present time, a hardware testbed is being developed to demonstrate that significant power performance (*30 MOPS/watt*) can be achieved in a scaleable embedded architecture using commercial technology. This testbed will also be the platform for conducting software-implemented fault tolerance experiments and for developing the system software needed to achieve the reliability goals. The next step will be the design and fabrication of a hardware prototype that will match the mass and form factor of a future flight model and will demonstrate scalability (*50 nodes*), reliability (*0.99 over 10 years*), and a power performance of at least *300 MOPS/watt.*

3.2 *System Software Initiative*

The System Software Initiative will provide a set of services that will enable applications to take full advantage of the computing capacity of the hardware architecture, while providing an easy-to-use development environment and ensuring reliable operation in space. By relying to the maximum extent practical on commercial software components, the system software layer will provide for the requisite performance capability and user interface. Since the hardware architecture will be based on commercially available components, radiation-induced faults will be common and hardware component failure will be a possibility. Hence the system software must provide mechanisms for detection and recovery from both permanent and transient faults. The major challenge to the System Software Initiative is to develop a fault detection and recovery scheme that ensures system reliability and availability without compromising the performance capability available to the applications.

The System Software Initiative will develop a middleware layer between a commercial operating system and the applications. This middleware layer will offer a suite of fault tolerance mechanisms from which the applications can make selections based on their reliability and efficiency requirements. The first version of the middleware layer will demonstrate reliability based on software implemented fault-tolerance (0.99 over 5 years), scaleability (50 nodes), and portability for all REE applications. Working with the COTS hardware and software layers, the middleware layer is projected to provide a system availability of 0.95 over the life of the mission and at least 0.90 within any 24-hour period. The overhead for this software layer is projected to be less than 10% of the average

[1] *MOPS: Millions of Operations Per Second. These may be a mixture of 32-bit integer and floating point arithmetic or logical operations. Although MIPS (millions of instructions per second) is a more traditional measure of processor capability, it does not quantify the actual amount of work accomplished on processors which have complex instruction sets. In many cases, however, MOPS and MIPS will be interchangeable*

system throughput during non-faulty operation. A later revision will add real-time and operate-through-fault capabilities.

3.3 Applications Initiative

The Applications Initiative will demonstrate that the unique high-performance, low-power computing capability developed by the project can enable new science investigation and discovery. Science Application Teams will demonstrate that substantial onboard computational capability will be a crucial ingredient in future science investigations. The Teams will ensure that architectures and system software produced by the project meet the needs of the spaceborne applications community. They will stimulate the development and implementation of new computational techniques that will transform the REE platforms from computers into tools of scientific discovery, on a par with the sensors and data collection systems with which they are integrated.

The Science Applications Teams will develop scaleable science and autonomy application algorithms. Software will be developed and installed on the hardware testbed. This software will be used to test, evaluate, and validate candidate architectures and system software using the REE testbed. A demonstration of scaleable applications on the hardware testbed will take place within months of its delivery. Subsequent generations of scaleable applications for installation on the REE flight computer prototype will build on the experience gained in the hardware testbed. These applications will be demonstrated on the flight hardware prototype. The project anticipates that there will be several flight opportunities available for this demonstration in the fiscal year 2002-2003 time frame.

In addition to generating the basic codes, the Applications Initiative will develop a suite of Applications Based Fault Tolerance (ABFT) mechanisms and guidelines. The effort will develop a taxonomy of application code types and a methodology for determining the applicable ABFT techniques based on the code's position in the taxonomy, and on application fault tolerance, timing, and efficiency requirements. In addition, a general theory of algorithmic and numerical fault tolerance techniques will be developed and a validated arithmetic library generated for use by the applications.

4. Project Status

Application Development Status:

Currently, there are five Science Application Teams working with the JPL REE engineering team:

Application	Principal Investigator	NASA Theme Addressed
Gamma Ray Large Area Space Telescope (GLAST)	Prof. Peter Michelson, Stanford University	Structure and Evolution of the Universe (SEU)
Mars Rover Science	Dr. Steven Saunders, Jet Propulsion Laboratory	Exploration of the Solar System (ESS)
Next Generation Space Telescope (NGST)	Dr. John Mather, Goddard Space Flight Center	Structure and Evolution of the Universe (SEU)
Orbiting Thermal Imaging Spectrometer	Prof. Alan Gillespie, University of Washington	Earth Science Enterprise
Solar Terrestrial Probe	Dr. Steven Curtis, Goddard Space Flight Center	Sun-Earth Connection (SEC)

Application codes from all five of the teams have been ported to a "Level 0 Testbed", a PC-based Beowolf cluster utilizing the Linux OS, and both Ethernet and Myrinet communications fabrics. Work continues on these applications to expand their capabilities and to improve their fault tolerance.

System Development Status:

- Initial radiation and system fault models have been generated for current COTS processing systems in a variety of earth orbits as well as deep space. The models have been extrapolated for future technologies out to the 2002 time frame.
- A software fault injector has been developed and ported to the Level 0 Testbed, and initial experimentation performed on the native fault tolerance of several application kernels.
- An initial set of Software Fault Tolerance (SIFT) techniques was implemented as a middleware layer and ported to the Level 0 Testbed.
- An initial set of ABFT routines were written into the application library and linked to the applications kernels.
- A series of fault injection experiments were run on the Level 0 Testbed with the SIFT Middleware and ABFT'd application kernels. Significant improvement was seen in fault detection and handling. Faults were inserted into the application code and data segments and into processor registers. In all cases, the errors were detected and appropriately handled. While these early experiments are not conclusive, they are encouraging and work continues in expanding the fault injection capabilities into other areas of the system.
- A series of technology and system design studies were performed and, based on these, a contractor (Sanders, a Lockheed Martin Co.) was chosen to fabricate a First Generation Testbed. The goals for this testbed are to:

1. Implement a COTS-based, 20-node parallel multicomputer, expandable to 50 nodes.
2. Demonstrate 30 MOPS/watt while running scientific parallel-processing applications.
3. Provide typical COTS fault detection/handling features such as EDAC on memories and standard processor exception handling.
4. Provide a boot-up sequence that guarantees that the system will properly initialize in the presence of permanent faults.
5. Have no single point of failure and degraded mode operability.
6. Provide a standard COTS (Posix compliant) node-level operating system.
7. Provide a fault injection system capable of injecting faults into any hardware/software component of the system.

The system implementation is currently underway with delivery scheduled for November 1999. When it becomes available, the application codes, a SIFT middleware layer, and a global system executive will be ported onto the system and a campaign of fault injection experiments run to determine the coverage of the system and to validate the approach. Additional development of the SIFT and System Executive as well as ABFT will proceed on this testbed. Reliability and Performability models will be generated and validated using the First Generation Testbed, and the performance:power metrics will be measured.

From this work, JPL will generate the requirements and system concept for the Flight Prototype Testbed hardware and software. JPL will then generate new models for the Flight Prototype system and will validate, through testing and modeling activities, the radiation and system fault models. Final validation testing on the Flight Prototype will certify its utility for space-based supercomputing applications and a suitable flight experiment will be found for insertion.

The final stage of the project will fabricate a next-generation REE system from then state-of-the-art COTS and fabricate a flight-ready system in 18 months.

5. REE First Generation Testbed System Architecture Overview

The First Generation Testbed (TB) System, to be built by Sanders, implements a state of the art instantiation of an REE system fully capable of being targeted for space flight application. The architecture is based on COTS technology and industry standards adapted for NASA space missions requiring low power, scaleable performance, and fault tolerance that are critical in meeting the REE project goals. The TB will be based on the Two-Level Architecture (Figure 2) and the Myrinet network technology developed from recent DARPA research. The architecture consists of distributed nodes that are interconnected by a fault-resilient system network that supports graceful degradation and has no single point of failure. Within the node, industry standard hardware and software interfaces are employed to isolate the application processor environment from the complex system control and communications environment in the architecture. This separation enables a stable system control and communications infrastructure that will endure over time, providing future systems with an established reliable platform that is independent of the fast changing COTS processor technology. The direct benefit is rapid insertion (18 months or less) of latest COTS technology to NASA missions, enabling at least an order of magnitude improvement in performance/power over traditional space radiation hardened computer solutions. The separation of the application from the control and communications layers also effectively isolates direct fault effects introduced in the application environment from the system environment.

5.1 Hardware Architecture

The First Generation Testbed hardware architecture is shown in Figure 3. The Testbed consists of 20 homogeneous nodes and a reconfigurable network housed in a standard VME chassis. The Myrinet network interconnects the nodes and external Sun workstations to complete the Testbed facility. The nodes and the Myrinet network operate as a complete embedded system. The Sun workstations provide:

1. External program storage for system startup support and for emulation of embedded mass-storage to applications running on the Testbed.
2. Control of the supplied fault injection facility
3. Emulation of Space Craft Computer (SCC) functions and interfaces
4. System reset

The node architecture consists of the Processing Element (PE), Node Controller (NC), and interface to the system network, thus fully reflecting the two-level architectural approach (Figure 4). Details of the hardware architecture components are provided in the sections that follow.

5.1.1 Node Controller

Within the node architecture (Figure 4), the Node Controller (NC) is the "first-level" computer in every node, providing consistent control and communication functions for the system. The NC consists of a generic programmable controller and a Network Transport Engine (NTE). The NTE provides the network processing functions and interfaces to the Processing Element (PE) and to the system network. The NTE is implemented in hardware using field programmable gate array (FPGA) technology to significantly improve the performance:power ratio associated with network communications. The NTE-based Node Controller also provides efficient fault-tolerant support for reliable control and communications for all supported protocols. The programmable flexibility of the NC also allows support of other standard network protocols for maximum application flexibility.

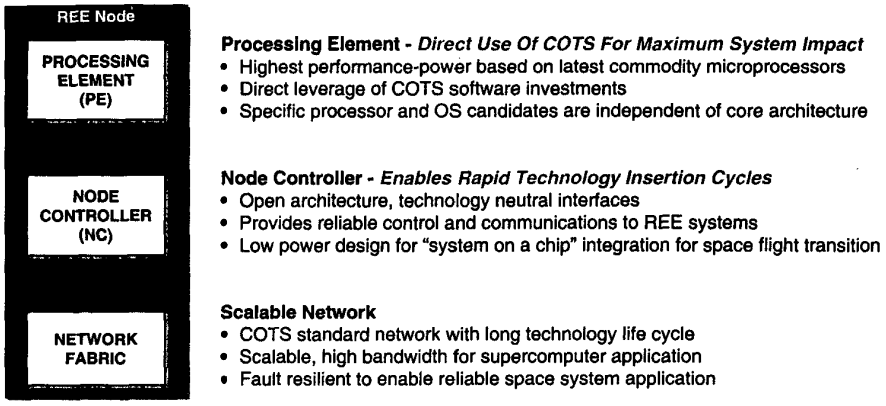

Figure 2. REE TB Two-Level Architecture

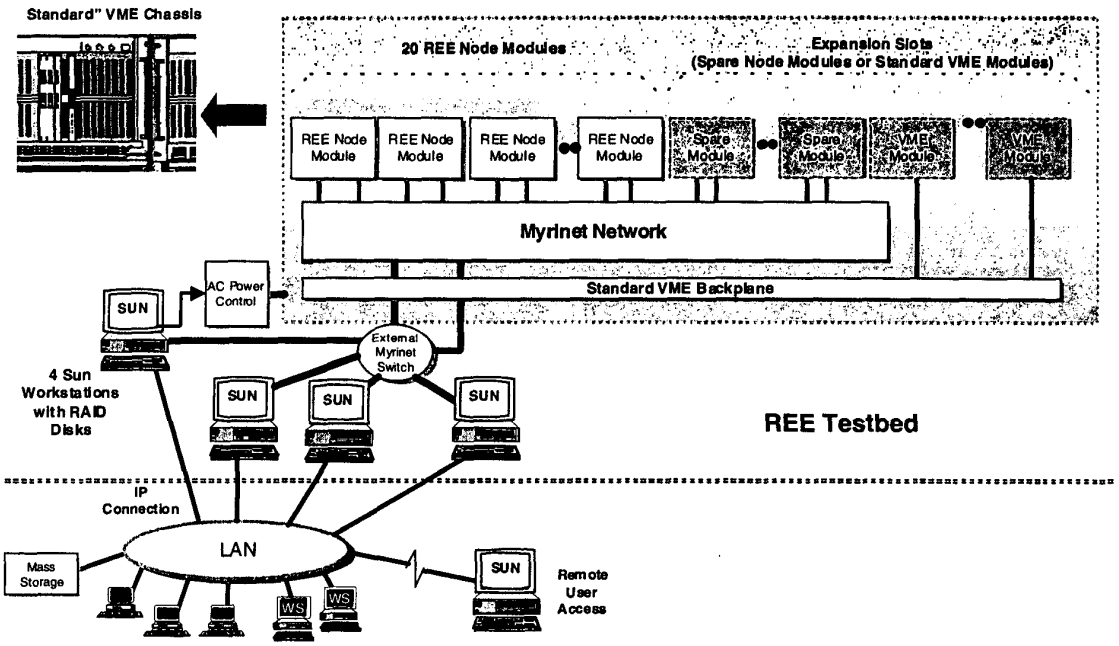

Figure 3. REE First Generation Testbed

The NTE is connected to the PE within the node through the industry standard PCI. The PCI implementation is 32 bits wide at 33 MHz and is fully compliant to the PCI Local Bus Specification, Revision 2.1. The NTE is also connected to the Myrinet network fabric through an external Myricom FI device for full compatibility with other devices on the Myrinet (e.g., Sun workstations with Myrinet interfaces). A redundant Myrinet FI interface is provided for fault-tolerance. For the REE First Generation Testbed, the NC will be packaged in a daughter card assembly that is compatible with the industry PCI Mezzanine Card (PMC) standard. The NC assembly can be mated with COTS boards with standard PMC support to form a complete node. This packaging approach will allow easy integration of the NC with new PE technology to continuously leverage COTS for performance:power improvements.

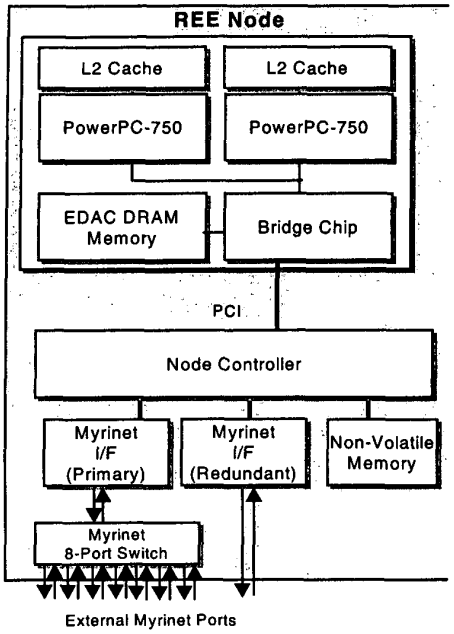

Figure 4. Node Architecture

5.1.2 Processing Element

The Processing Element (PE) is the "second level" computer in the TB node architecture and is the primary application processing resource. The PowerPC 750 was selected as the leading generation of general purpose COTS microprocessors based on commercial performance:power trends. The PowerPC 750 is a low-power, 32-bit implementation of the PowerPC Reduced Instruction Set Computer (RISC) architecture with a strong legacy of software support. Based on a superscalar architecture, it is capable of issuing three instructions per clock cycle into six independent execution units: two integer units; floating-point unit; branch processing unit; load/store unit; and system register unit. Cache supports include separate L1 instruction and data caches at 32 KB each, and backside L2 cache support of up to 1 MB. The latest version of the PowerPC 750 product is based on a 0.2-micron, six-layer metal CMOS process, 2V internal core voltage, with a typical power dissipation of 5.5 watts at the maximum clock rate of 466 MHz.

The PE architecture for the REE First Generation Testbed is based on a dual parallel PowerPC 750 configuration that will maximize performance:power at the node level. Each PowerPC runs at the nominal clock rate of 366 MHz and is supported by a dedicated 1 MB of backside L2 cache, allowing both PowerPC microprocessors to operate in parallel with minimum power overheads. Each PowerPC 750 can deliver 17.2 SPECint95 and 11.1 SPECfp95 in performance. A COTS bridge chip provides the interface between the two PowerPCs, DRAM memory, and PCI interface that connects the PE to the Node Controller. There are 128 MB of DRAM memory provided on each REE node with expansion support for up to 256 MB. The DRAM memory is supported by error correction code (ECC) that corrects on single-bit errors and detects on double-bit errors.

The First Generation Testbed PE will be a direct application of a prevailing commercial computer board product based on dual-PowerPC 750s, representing the latest instantiation of leading edge COTS technology. Based on preliminary characterization results of ~1.98 operations attained per clock cycle for each PowerPC750 operating at 366 MHz, the node is expected to achieve ~ 45 MOPS/watt based on an estimated typical-max power consumption of 32 W. Continuous insertion of future generations of COTS products based on the PowerPC technology roadmap would achieve frequent technology refresh cycles of less then 18 months (e.g., next generation, and G4 insertion).

5.1.3 Reconfigurable Network

The TB nodes are interconnected by a reconfigurable high-speed system network based on Myrinet (ANSI/VITA 26-1998). Myrinet is a COTS switch-based network supporting scaleable system configuration of up to 64,000 nodes using the currently available 4-, 8-, or 16-port switches. A Myrinet interface consists of a pair of byte-wide links for concurrent input and output transfers (a link per direction), achieving a bi-section peak

bandwidth of 1.28 Gbit/s per link for each direction, and using low-voltage signaling technology. Figure 5 shows the flexible network connectivity offered by Myrinet.

In the REE testbed, each node has a separate primary and secondary (redundant) Myrinet interface that is physically connected to a single Myrinet system network through different switches. The redundant interface is normally power off to conserve power. The system network implementation will provide at least two independent routes to each node's primary and redundant interfaces using multiple 8-port Myrinet switches to enhance bandwidth performance and fault tolerance. The switch-based network and LAN-like characteristics provided by Myrinet provide a scaleable, flexible, high performance network that is also highly resilient to faults.

5.2 Baseline Software Architecture

The TB will be delivered to JPL with a baseline set of software including the node level operating system, communication and network control software, system boot up software, and a system fault injector for use by JPL in developing global system executive, SIFT middleware, ABFT, and applications software.

The TB Baseline Software Architecture (Figure 6) is based on industry standards in order to preserve system and application software investments. The Testbed implementation employs mainstream COTS software products that include operating system, development tools, and libraries to fully leverage commercial investments. The application programming environment employs industry standard APIs to ensure seamless code portability between workstation, testbed, and future generation flight hardware. The software architecture takes advantage of the TB two-level architecture with communication services provided by the node controller to maximize system performance:power. Enhanced software services include a reliable communications layer to enhance system fault tolerance; system services for fault injection, power control and monitoring; system configuration; and logging. Together, these software services provide a robust environment to support reliable system operations.

5.2.1 TB Application Programming Model

The TB Programming Model (Figure 7) is based on a distributed system concept employing explicit message passing paradigms for interprocessor communications. Each node is a computer function with an independent operating system per processor, supporting multiple processes per processor at both user level and supervisory level. Supervisory level processes share a common address space while user process have separate, protected address spaces. Processes can control other processes. Typical applications may have one or more processes that communicate among themselves.

5.2.2 Portable Application Programming Interfaces

The TB supports industry standard portable application programming interfaces (API) that include MPI 1.2 subset, Socket, and NFS (see Figure 8). The MPI 1.2 subset standard implements the most commonly used functions by the real-time community with full support for point-to-point and collective communications. Parallel applications developed under widely supported workstation platform can be directly ported to the TB to preserve application software investments. The TB implementation of the MPI 1.2 subset is designed for performance and portability by fully exploiting the NTE features in the Node Controller, providing packet acceleration to maximize performance:power with standard support for C, C++, and Fortran bindings. The implementation will allow remaining MPI API functions to be added as required based on future user requirements. Support for TCP/IP are also provided for standard Socket and NFS services to further maximize application programming flexibility and portability.

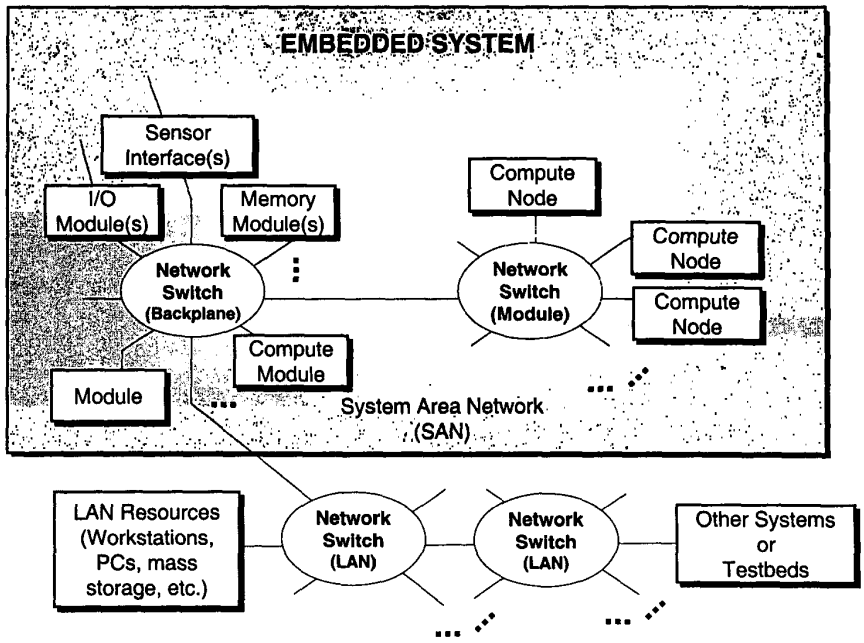

Figure 5. Myrinet Network

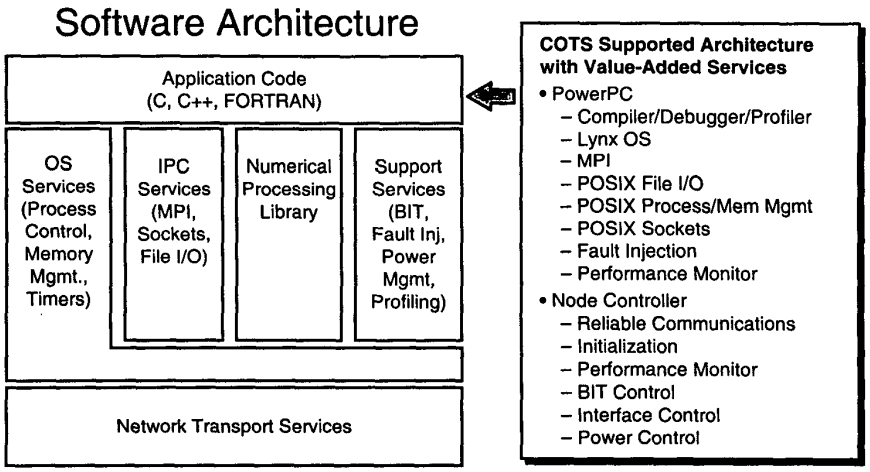

Figure 6. TB Baseline Software Architecture

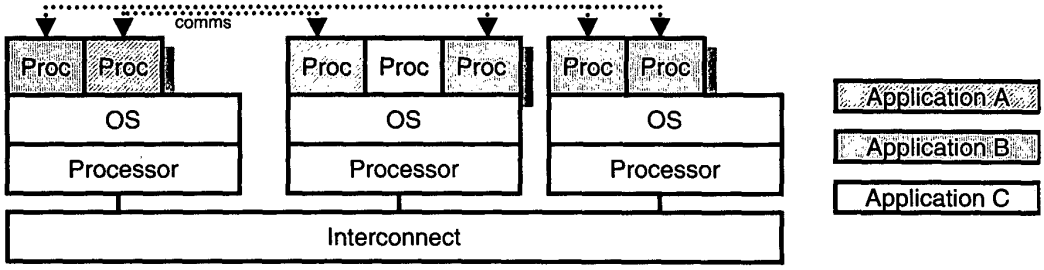

Figure 7. TB Programming Model

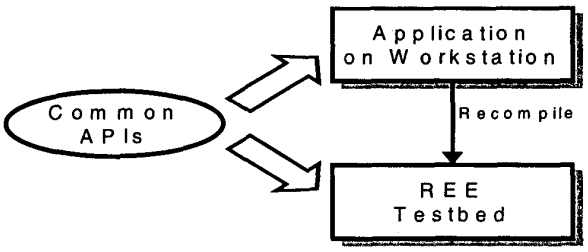

Figure 8. Portable Application Programming Interfaces

5.2.3 COTS Operating System

The software architecture is independent of the specific operating system that runs on the PE (e.g., PowerPC 750). However, the First Generation Testbed will be running Lynx as the COTS-embedded OS in its implementation. This selection was based on the existing Lynx support for separate user and supervisory processes, process protection from other processes, and memory protection features that, together, provide the Testbed with a robust OS platform to support SIFT development. Lynx supports industry standard APIs that include standard Unix file I/O, threads (Posix 1003.1c), sockets (Posix 1003.1g), NFS and RPC, process and memory management (Posix 1003.1), Posix 1003.1b real-time extensions, and third party numerical processing library.

5.2.4 Support Software and Tools

The TB is supported with a comprehensive board support package (BSP) for the PowerPC 750 PE that includes the Lynx OS, standard development environment (C, C++, Fortran), and TotalView parallel source level debugger. At the node level, the BSP software also includes support for node power-up, boot-up, built-in test (BIT), and FLASH memory loader functions. At the system level, the Testbed will be supported by an autonomous system startup capability even in the presence of permanent faults.

6. Fault Injector

The Testbed will be supported by a fault injector to support system fault analysis as an alternative to traditional approaches requiring exhaustive failure effects and modes analysis. The fault injector approach is software-centric as it is constrained by the use of COTS components. The fault injector emulates transient and semi-permanent faults that are induced by single event upsets (SEUs). The fault injector will be used by JPL to experimentally measure system fault behavior, to verify system design assumptions, and as a debugging tool for SIFT systems.

7. Conclusion

Insertion of supercomputing facilities into NASA space exploration missions has been shown to be critical to the continuation of the space program. NASA's REE project has been initiated to address this need through the use of COTS hardware and software components. The use of COTS components within a scaleable system architecture will provide rapid and low-cost development and insertion of the state of both systems and application codes for space-based systems. A first instantiation of an REE system, in the form of the First Generation REE Testbed, will be delivered during the fourth quarter CY99. The Testbed will be used to develop and validate the project and system development concepts. The knowledge gained through the development of the Testbed and its subsequent use as a test and experimentation vehicle will provide the basis for the design of the Flight Prototype System, which is expected to achieve at least 300 MOPS/watt in a space-qualified system ready for fielding in the 2003-2004 time frame.

Acknowledgments

This work is performed at the Jet Propulsion Laboratory, California Institute of Technology and Sanders, A Lockheed Martin Company, under a contract with the National Aeronautics and Space Administration. This project is part of NASA's High Performance Computing and Communications Program, and is funded through the NASA Office of Space Sciences.