CRAY
RESEARCH EUROPE

CRAY RESEARCH ANNOUNCES AVAILABILITY OF CRAY Y-MP EL COMPUTER SYSTEM

Customer Interest Strong, With 18 Orders

Cray Research has extended its leadership in the high-performance computing industry by unveiling its CRAY Y-MP EL system, an air-cooled, entry level system that is based completely on CRAY Y-MP architecture and is fully compatible with the company's entire product line. Deliveries of the system begin this quarter.

"This new entry-level system is the price/performance leader in its class and will bring the CRAY Y-MP computing environment within easy reach of hundreds of new customers and prospects," said John Rollwagen, Cray Research chairman and chief executive officer. "The CRAY Y-MP EL system is the most affordable high-performance computing option available today and is particularly well-suited for the first time supercomputer buyer, or the existing customer who requires a departmental system to complement its current Cray Research system."

The Cray Y-MP EL system is available with one to four central processing units (CPUs) and up to one gigabyte of central memory, at prices ranging from under $300,000 to about $1 million. This system succeeds the CRAY XMS system, which was based on technology acquired through the company's June 1990 purchase of Santa Clara, California-based Supertek Computers, Inc.

"This is a CRAY Y-MP system in every sense of the word," said Rollwagen. "The CRAY Y-MP EL system was designed and engineered at Cray Research and is based completely on CRAY Y-MP architecture. In developing the new system, we relied heavily on simulation, using almost 20,000 CRAY Y-MP CPU hours. This enabled us to bring this product to market faster, and is an excellent example of the competitive edge that our computer systems bring to customers."

- more -

• MEDIA INFORMATION •

Rollwagen added that the new system runs all CRAY Y-MP system and applications software, including approximately 600 of the most widely used applications programmes. Because the new machine is a CRAY Y-MP system, entry level customers can upgrade easily to more powerful Cray Research supercomputers, and/or run their software codes on larger Cray Research systems.

Customer Interest

The company is marketing the new system to existing and new customers for use as a stand-alone departmental system, as a component on a customer's heterogenous computer network, or as a high-speed file server when operating Cray Research's UNICOS Storage System software.

Dennis McFadden, general manager of the company's entry level systems (ELS) division, indicated that customer and prospect interest in the new product has exceeded expectations. Cray Research has received 18 orders for the CRAY Y-MP EL system from worldwide customers. Orders for the Cray Y-MP EL system in the UK have come from British Aerospace, who will install the system at its military aircraft division in Warton, Lancashire; the Admiralty, which requires a secure system for classified work; and Lotus Engineering, an automotive consultancy that is part of the Lotus group, the manufacturer of high-performance sports cars.

"The Cray Y-MP EL system will add the power of supercomputing to our consultancy operations," said Richard Jones, General Manager CAD CAE (computer aided design, computer aided engineering) at Lotus Engineering. "We will be using it to expand our high-level CAE capabilities, focusing on computer intensive tasks such as flow simulation and crash analysis. The accessible price means that the purchase will be a most cost-effective one for us."

Elsewhere in Europe orders have been received in France by Ecole Centrale de Paris (Laboratoire de Mécanique des Structures et des Sols), an engineering school, and the Université Paris-Sud Orsay (Central de Calcul P.S.I.).

- more -

Worldwide, orders for the CRAY Y-MP EL system have come from customers including Altair Engineering, an automotive engineering consulting company in the US; and Nikon, a camera and optics company in Japan. In addition, a comparable number of prospects have formally indicated their intent to acquire the new system.

"This high level interest, particularly from many first-time customers, is making our entry level system strategy come to life," said Rollwagen. "We set out to broaden our user base and grow our business with the CRAY Y-MP EL system. We see very solid sales volumes for the product in 1992. We also see these current EL users growing into our more powerful systems, contributing significantly to our future growth."

System Features

The new system can be installed in any air-conditioned office, operates on standard 200- to 240-volt, 50- to 60-hertz power, and can be easily upgraded in just a few hours at customer sites, all features which significantly reduce the cost of installing, operating, and maintaining a Cray Research system.

Additional features of the CRAY Y-MP EL system are:

- Balanced CRAY Y-MP architecture
- Highly reliable CMOS chip technology
- 30-nanosecond clock speed: four results per clock period
- Peak performance of 133 megaflops (one million floating point operations per second) per CPU when delivering four results per clock period
- 256 to 1024 megabytes of memory
- Over four gigabytes/second of total memory bandwidth (over one gigabyte/second per CPU)
- Operates Cray Research's powerful UNICOS (Unix-based) operating system; Cray Research's CF77 Fortran, Standard C, Ada and Pascal compiling systems; and more than 600 third party applications software codes for nearly all scientific and engineering disciplines

- more -

- Provides multi-CPU parallel processing, with automatic vectorized and Autotasking (TM) capabilities
- Up to four integral input/output (I/O) subsystems connected to each CPU, with data transfer via standard TCP/IP, HYPERchannel, Ethernet, or FDDI network interfaces. A HIPPI interface will be available by mid-1992
- Area: 11.0 square feet
- Power requirement, less than 6 kilowatts

Cray Research designs, manufactures, markets and supports high-performance computer systems for scientific and engineering applications.

- ends -

For further information, please contact:
Tara Jenkins/Tim Kaye
Shandwick PR
Tel: 071 835 1001

23 October 1991

Now, the competitive edge of supercomputing is within reach.

The CRAY Y-MP EL Supercomputer System

CRAY RESEARCH, INC.

Introducing the CRAY Y-MP EL Supercomputer System

As the maker of the world's most powerful and highest-quality computational tools, Cray Research has been giving scientists and engineers the competitive edge for over 15 years. To bring this capability to a broader range of users, Cray Research has developed the CRAY Y-MP EL system — the most affordable supercomputer we've ever offered.

Extending the range of supercomputing excellence

The CRAY Y-MP EL system delivers unmatched throughput performance in its price range by incorporating all the advantages of our powerful and balanced CRAY Y-MP architecture. With up to four CPUs working in parallel with up to 1024 Mbytes of central memory, the CRAY Y-MP EL provides the highest possible performance on a wide variety of applications. But its computational power doesn't stop there. As your problem-solving needs grow, the CRAY Y-MP EL provides a cost-effective pathway to the computational power of large-scale supercomputing.

High performance made accessible

Until now, the cost of supercomputing often exceeded the means of many who could benefit from this technology. The CRAY Y-MP EL computer system makes superior performance more affordable by significantly reducing the cost of acquiring, installing, operating, and maintaining a real Cray Research supercomputer.

Because it is air-cooled and uses less than 6 kW of power per cabinet, the CRAY Y-MP EL can be installed in an air conditioned office environment. It has a limited number of connections, making installation quick and easy.

A total supercomputing solution

At Cray Research, we offer a total supercomputing solution with outstanding performance and functionality. The CRAY Y-MP EL supercomputer works with Cray Research software, applications, and customer networks, allowing you to focus on science and engineering — not the system's requirements.

The CRAY Y-MP EL system is upwardly and downwardly compatible with other members of the CRAY Y-MP supercomputer family. All Cray Research systems run UNICOS, a powerful UNIX-based operating system optimized for maximum performance on production workloads. With outstanding functionality, performance, and ease of use, UNICOS is the most powerful and feature-rich operating system available for technical computing.

The CRAY Y-MP EL system offers performance-oriented software products that enhance its capabilities. From industry-leading compilers to powerful performance optimization tools, Cray Research software ensures that you will get the highest possible performance from your CRAY Y-MP EL system.

Cray Research systems are unsurpassed in their ability to connect to computer hardware from other vendors. The CRAY Y-MP EL conforms to industry standards and supports a variety of language extensions and tools from other vendors, so your existing network investments are protected. The result is an optimum computing environment with a wider range of resources that improves user productivity.

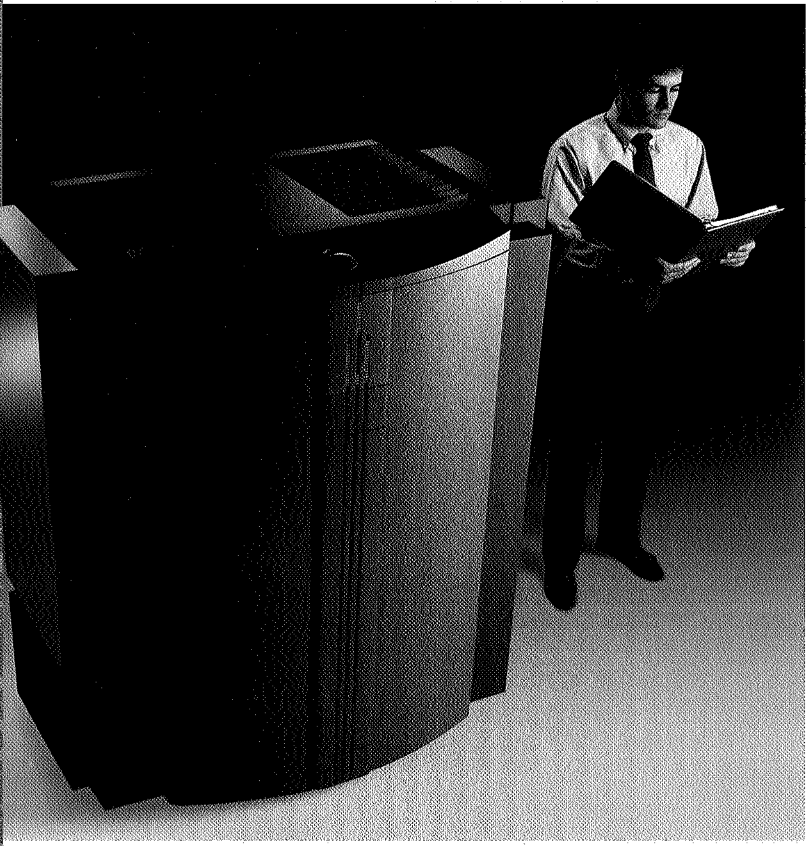
The CRAY Y-MP EL system

High-performance functionality

The CRAY Y-MP EL supercomputer features a powerful, balanced architecture that provides the highest possible performance in its class on scientific and engineering applications. In addition to departmental supercomputing, it also can be used in the following ways:

□ *As a complementary system for larger Cray Research systems.* The CRAY Y-MP EL is ideal for UNICOS application development. Because binaries from the CRAY Y-MP EL system will run on other CRAY Y-MP systems, work is easily scaled to larger Cray Research systems.

□ *As a secure system.* Because it is physically compact and offers removable storage media, the CRAY Y-MP EL is ideal for secure processing environments. UNICOS also provides multi-level security.



□ *As a high-performance file server.* Combined with the powerful data management features of the UNICOS operating system, the CRAY Y-MP EL system is an excellent file server platform. With support for standalone STK autoloading tape cartridge systems, the CRAY Y-MP EL file server can satisfy requests from multiple supercomputers over gigabit/second networks while providing service to smaller systems, workstations, and personal computers. When used as a file server, the CRAY Y-MP EL system may also simultaneously perform scientific processing.

Advantages of affordable supercomputing

The CRAY Y-MP EL system provides the following benefits without compromising performance:

□ *Unmatched price/performance.* The CRAY Y-MP EL offers more computing power for the money by offering the most throughput performance in its price range for multi-user technical computing.

□ *Extensive connectivity.* The CRAY Y-MP EL system will connect easily to your existing network; it offers extensive connectivity to a wide variety of mainframes, minicomputers, and workstations.

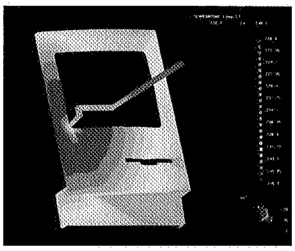
□ *Full functionality.* The CRAY Y-MP EL system can be a cost effective departmental supercomputer platform or a node in a heterogeneous networking environment.

□ *Upward compatibility.* The entire CRAY Y-MP product line is binary compatible, providing a seamless pathway from the CRAY Y-MP EL system to the world's most powerful supercomputer systems. This binary compatibility saves time and provides users and administrators with more control and consistency throughout the supercomputing environment.

□ *Easy access to high performance.* The CRAY Y-MP EL system runs the same powerful operating system, UNICOS, the same Autotasking/autovectorizing compilers, and the same library of software applications as other members of the CRAY Y-MP family.

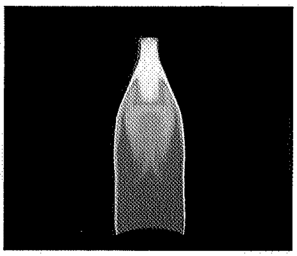
□ *Cost effectiveness.* The CRAY Y-MP EL supercomputer system is compact, easy to install, and inexpensive to operate. Its low power requirements, high reliability, and minimal service requirements reduce operating costs.

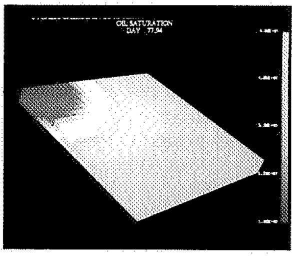
Supercomputers are used to simulate physical phenomena that would be difficult or impossible to create experimentally. These simulations provide the necessary insight to reduce design cycles and produce innovations.



Left, flow front distribution in an injection-molded plastic computer case modeled with MOLDFLOW. Molding is an Apple Macintosh Classic front bezel.

Right, fluid temperature pattern within a beer bottle as it is heated during pasteurization in order to sterilize beer, simulated with the FIDAP fluid dynamics program.





Left, simulated oil saturation in a quarter-five-spot model of a petroleum reservoir undergoing a chemical flood-enhanced recovery process. Image is displayed in three dimensions utilizing MPGS.

Proven, balanced architecture ensures high performance

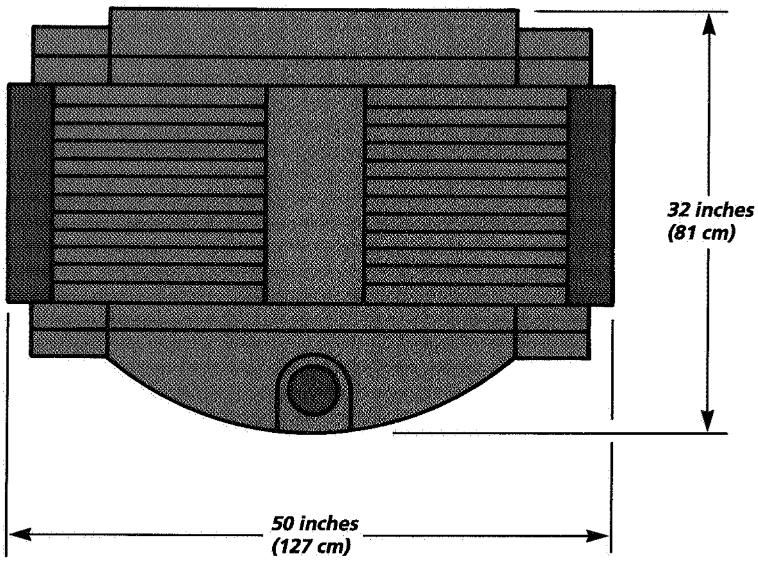
The CRAY Y-MP EL system combines a proven architecture with innovations that provide the highest level of sustained performance in its price range. Using the integrated vector CRAY Y-MP architecture, each CPU provides balanced scalar, vector, memory, and I/O performance. To enhance performance while preserving binary compatibility, the CRAY Y-MP EL includes an innovative multifunctional unit extension to this architecture that provides up to four results per clock period (instead of two). To enhance performance even further, the CRAY Y-MP EL CPU is designed to maximize the overlapping of vector, scalar, memory, and I/O operations.

CRAY Y-MP EL Product Specifications

CPU	
Technology	CMOS
Clock period	30 ns
Number of CPUs	1 - 4
Peak performance (per CPU)	133 MFLOPS
Memory	
Memory ports	**4 per CPU**
Technology	70 ns CMOS DRAM
Memory size	256 - 1024 Mbytes (32 - 128 Mwords)
Memory bandwidth per CPU	1.05 Gbytes/sec
Total memory bandwidth	4.2 Gbytes/sec
Memory banks	64, regardless of memory size
I/O	
Number of I/O Subsystems	1 - 4 per CPU
I/O bandwidth per CPU	264 Mbytes/sec
Total system I/O bandwidth	1.05 Gbytes/sec
Total VME bandwidth	640 Mbytes/sec
HIPPI	100 Mbytes/sec each
Physical characteristics (per cabinet)	
Weight	1400 lbs (635 kg)
Area	11 ft^2 (1 m^2)
Maximum power consumption per cabinet	6 kW (6 kVA)
Operating temperature	50° - 85° F (10° - 35° C)

Configurations to fit present needs, with room to grow

The CRAY Y-MP EL can be upgraded easily in the field. The standard CRAY Y-MP EL configuration consists of a single cabinet containing the CPU(s), memory, and one to four VME-based I/O Subsystems connected to disk subsystems or disk array subsystems, tape systems, and networking systems. Over 40 Gbytes of disk storage capacity can reside in the main cabinet. Up to three I/O peripheral cabinets may be added to provide up to 16 VME I/O Subsystems and over 200 Gbytes of disk storage capacity.



High-performance I/O

VME I/O subsystem

As with all CRAY Y-MP systems, the CRAY Y-MP EL offers the best I/O performance in its class. The IOS allows the central memory of the CRAY Y-MP EL system to communicate at high speeds with networks and peripherals such as disk storage units and tape units, while off-loading this activity from the CPU.

The CRAY Y-MP EL system uses industry-standard VME-based I/O subsystems that connect to a wide variety of peripherals and networks. The VME I/O subsystem is an integral part of the CRAY Y-MP EL design, acting as the mainframe's data distribution point.

The VME I/O technology is extremely versatile; it provides customers with a flexible computing platform that can grow with their I/O and peripheral needs. The standard configuration of the CRAY Y-MP EL system includes one VME I/O subsystem. Additional VME I/O subsystems can easily be configured at customer sites.

To increase the CRAY Y-MP EL production workload capacity, the CRAY Y-MP EL system has an aggregate VME I/O bandwidth of up to 640 Mbytes/sec to peripheral devices. This large bandwidth allows users to access more peripheral devices and perform more simultaneous activities.

Cray Research supports a wide range of peripheral and network devices to meet your performance, capacity, and budgetary needs. To provide high-speed access to data, the CRAY Y-MP EL supports disk drives with transfer rates of up to 18 Mbytes/sec as well as high-performance online tapes and stand-alone STK 4400 tape cartridge autoloaders.

Software

Performance-oriented, feature-rich software

The Cray Research application support environment is a complete body of performance-oriented system software that enables users to focus on their work, not the system's requirements. As part of a total system solution, the application support environment includes UNICOS, the world's first UNIX-based supercomputer operating system, as well as a set of powerful compilers, development tools, high-performance libraries, and data storage systems.

UNICOS operating system

UNICOS is the most powerful and feature-rich UNIX-based operating system available to supercomputer users. Based on the UNIX System V operating system with Berkeley extensions and performance enhancements, UNICOS is an interactive and batch operating system that offers a number of advantages including high performance, full functionality, portability, and connectivity.

UNICOS features hundreds of programmer years of optimizations that deliver very high performance on production workloads. Together with the powerful CRAY Y-MP EL computer hardware and Autotasking capabilities, this performance not only provides fast turnaround on individual jobs, but also high throughput for a varied workload through sophisticated job scheduling capabilities.

UNICOS combines all the inherent strengths of UNIX, such as a familiar user interface, with production-oriented features including high-performance I/O, optimal memory bandwidth utilization, multiprocessing support, ANSI/IBM tape support, resource control, sophisticated job scheduling, tunable accounting, and batch processing.

VME I/O subsystem highlights

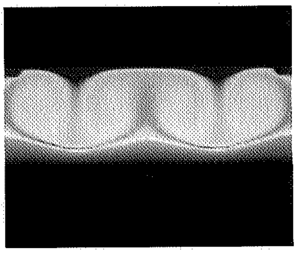
□ Up to 4 VME I/O subsystems per CPU
□ 40 Mbyte/sec bandwidth per VME
□ 160 Mbyte/sec VME I/O bandwidth per CPU
□ Aggregate VME I/O bandwidth of 640 Mbytes/sec

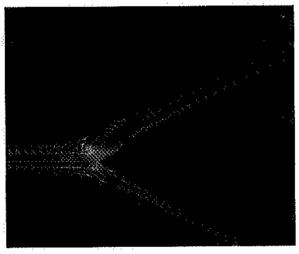
Peripheral highlights

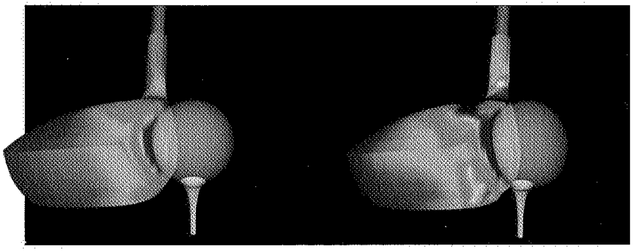
□ Individual disk speeds of 2.75, 9.3, and 18 Mbytes/sec
□ Main memory ldcache increases file system efficiency
□ Provides fast single file access disk striping
□ Tape drives supported include: 1/4 inch cartridge; 5 Gbytes, 8 mm; 125 ips, 9 track; IBM 3480-compatible

Right, crystal growth of silicon calculated using the code MHD2DTA.

Far right, blood flow velocity magnitude vectors in a bifurcated vessel simulated with the FIDAP fluid dynamics program.







Left, stresses in a titanium metal driver at a particular instant as it strikes a golf ball at 100 miles per hour, simulated using the MSC/DYNA finite element program; data visualized with the MPGS postprocessing environment. Simulations such as this can be used to help design stronger, more lightweight clubs.

Compilers

Cray Research offers the most powerful compilers in the industry, including the CF77 Fortran compiling system, the Cray Standard C Compiler, Cray Ada, and Pascal.

The CF77 compiling system was the first Fortran compiler in the industry with the functionality required for automatic parallel processing, automatic vectorization, and scalar optimization. These compiling features typically require little or no code modification by the user, and full optimization is turned on by default.

The CF77 compiling system ensures portability with full compliance to ANSI standard 3.9-1978. The flexibility of CF77 allows it to accept many nonstandard constructs written for IBM, DEC, CDC, and other vendors' compilers.

The CF77 compiling system compiles Fortran77 programs into executable code modules that take full advantage of the CRAY Y-MP EL vector capabilities, while its Autotasking feature further enhances performance on multiprocessor systems.

For those codes that are not highly vectorizable, CF77 ensures the best possible execution time by providing scalar optimization for the CRAY Y-MP EL system.

Because supercomputing applications written in the C language are becoming increasingly popular, Cray Research offers the highest-performance ANSI standard C compiler in the industry. The Cray Standard C compiler can be used to create portable, highly optimized code with performance comparable to Fortran programs. Like CF77, the Cray Standard C compiler takes full advantage of the CRAY Y-MP EL performance capabilities with automatic vectorization, scalar optimization, and Autotasking.

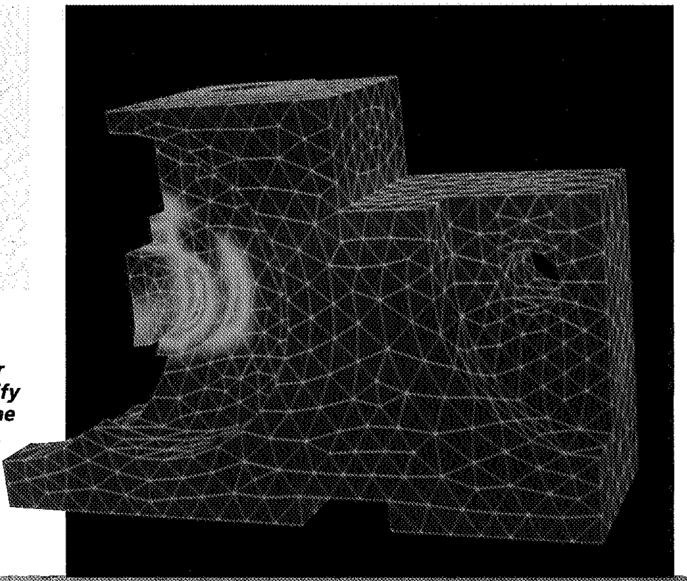
□ Removable disk drives
□ Support for high-speed and low-speed networks (Ethernet, HYPERchannel, and FDDI)
□ Support for STK 4400 tape autoloaders
□ Support for the ANSI standard HIPPI channel
□ Centronics-compatible printers
□ Plotters for seismic applications

UNICOS highlights

Full functionality
- Batch processing
- High-speed tape support
- Resource management
- Accounting
- Checkpoint/restart
- Networking
- Data Migration Facility (DMF)
- Online system diagnostics
- Multi-level security

High performance
- Autotasking and autovectorizing features
- Efficient I/O
- File system extensions

Ease of use
- Advanced program development tools
- X Window System support
- Performance analysis tools



Right, the ANSYS program was used to solve this linear elastic problem; colors signify the Von Mises stresses on the component — a tool holder.

Autotasking

The CF77 compiling system and the Cray Standard C Compiler include Autotasking features that can dramatically improve performance on multiprocessor CRAY Y-MP EL systems. The Autotasking feature divides a program into discrete tasks that can be performed concurrently on all processors in the CRAY Y-MP EL system. The Autotasking features also include a convenient, powerful set of directives that allow programmers to fine-tune their code for even better performance. In production environments, this feature can be used to improve individual jobs and overall system throughput.

UNICOS Storage System

The UNICOS Storage system is the world's first high-performance UNIX-based file server. With the UNICOS Storage System, the CRAY Y-MP EL system enables users to meet their computing needs while addressing the file storage needs of their network. The UNICOS Storage System provides transparent data access, file access capabilities, system administration, and automated storage management capabilities.

Applications

Cray Research offers leading-edge applications for nearly every scientific and engineering discipline, including the most widely used third-party application programs. These applications are used by diverse industries to accelerate product development, increase productivity, and solve basic research problems. Applications are available for industries such as aerospace, automotive, chemistry, energy, petroleum, and defense.

The power of visualization

Cray Research offers the following software packages to couple the power of visualization with its supercomputers:

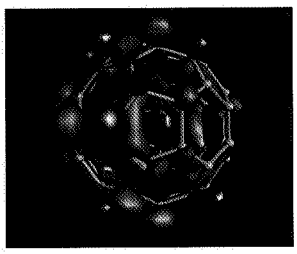
□ *Multipurpose Graphic System (MPGS),* an interactive menu-driven engineering visualization package for use on Cray Research computer systems. MPGS works with a wide variety of engineering applications.

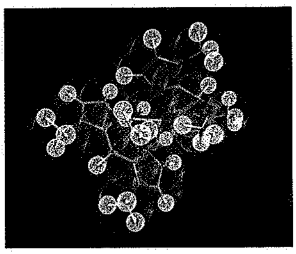
□ *UniChem,* Cray Research's easy-to-use supercomputing environment for computational chemistry simulation that enables researchers to explore complex chemical systems at a new level of detail from their desktops.

□ *The Cray Visualization Toolkit (CVT),* which enables users to run applications on Cray Research systems through their workstations. CVT allows users to generate graphics and user interfaces easily with the following tools:

- Release X11R4 of the X Window System
- Sun Microsystems' XView toolkit (OPEN LOOK)
- Open Software Foundation's (OSF) Motif 1.1 Toolkit
- Silicon Graphics, Inc. Distributed Graphics Library (DGL)

These tools allow most applications that run on Cray Research Systems to have the same "look and feel" as the most common workstation environments, making Cray Research systems even easier to use and making users more productive.





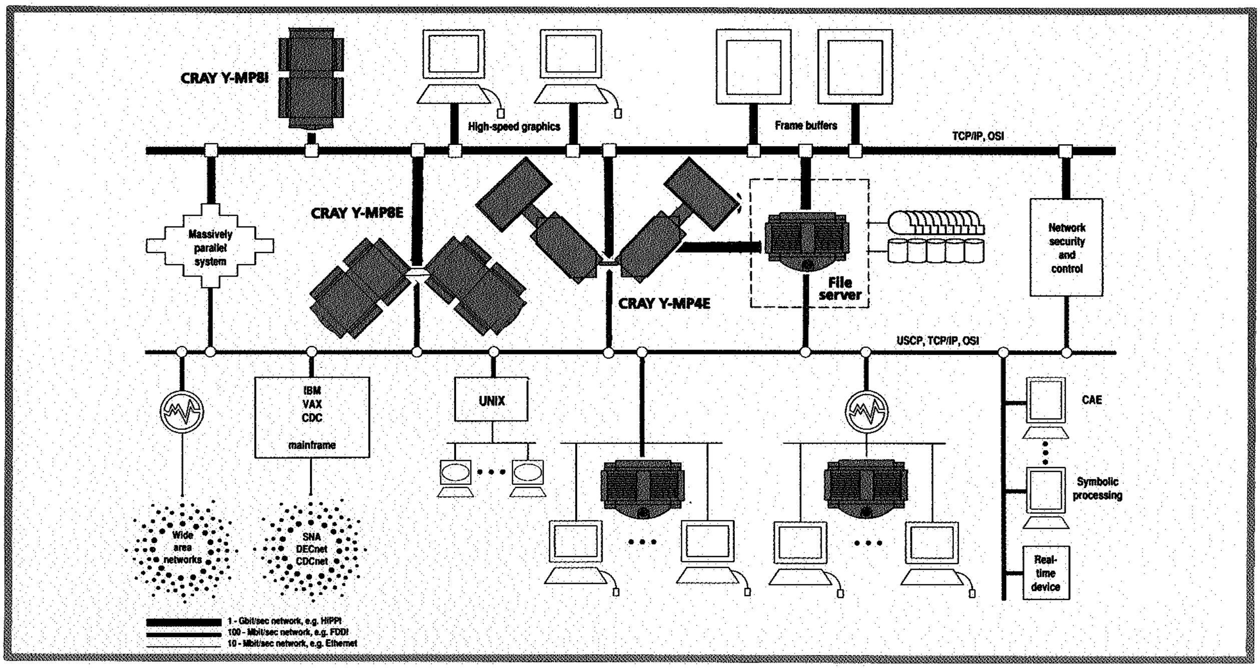
Cray Research's UniChem computational chemistry environment allows researchers to build, calculate, and visualize complex chemical systems. Far left, lowest unoccupied molecular orbital (LUMO) for the C-60 molecule (Buckminsterfullerene). Left, density functional model of a copper imizadole complex displayed with a Van der Waals surface.

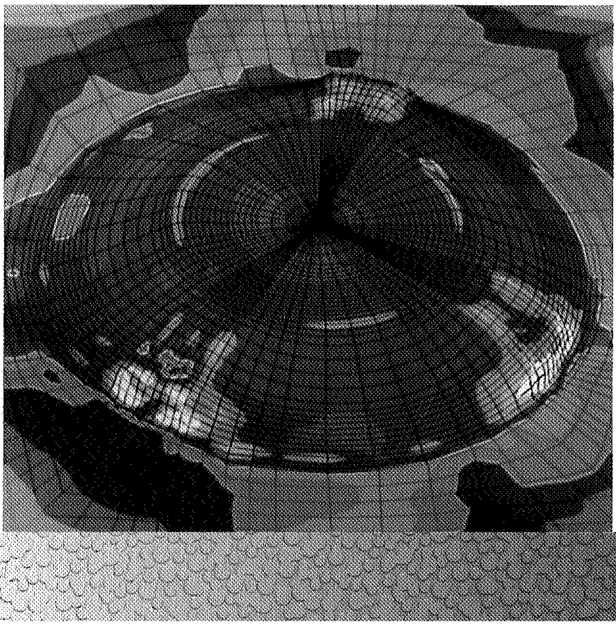
Delivering supercomputing power to your desktop

To bring the benefits of supercomputing to more users than ever before, Cray Research is dedicated to making its systems accessible through Network Supercomputing. Because Cray Research supercomputers support industry standards as well as a variety of language extensions and utilities from other vendors, they can be integrated easily into heterogeneous computing environments.

An array of communication products and protocols supported by Cray Research allows applications to be distributed within your network. Through the implementation of emerging and de facto networking standards, Cray Research provides connectivity to most UNIX-based mainframes, minicomputers, and workstations. These standards include the TCP/IP networking protocol and applications, the X Window System, the Network File System, the Open Systems Interconnect (OSI) of the International Standards Organization (ISO), the High Performance Parallel Interface (HIPPI), the Fiber Distributed Data Interface (FDDI), as well as other networking standards.

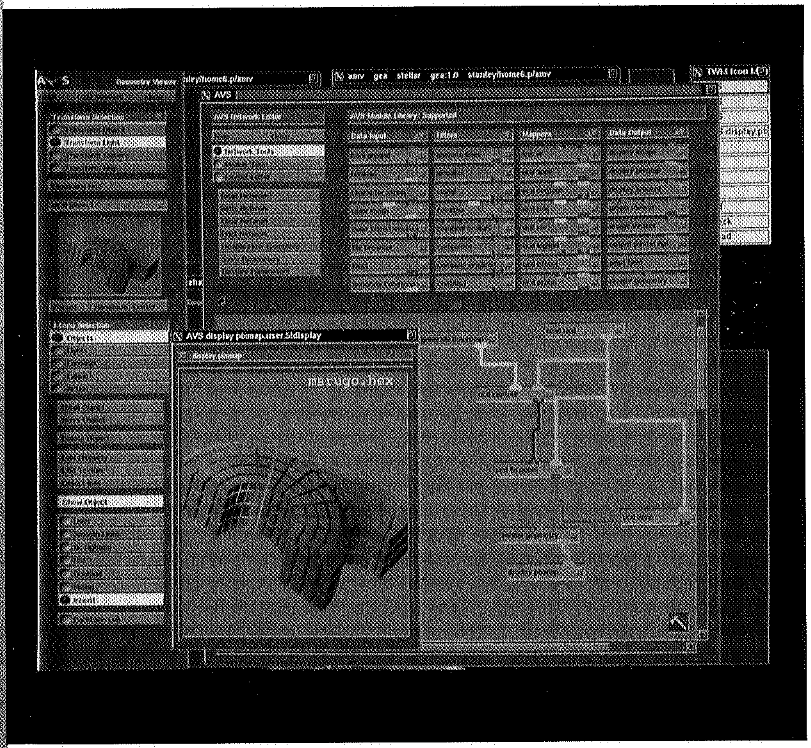
Network Supercomputing increases user productivity by allowing access to a wide range of computing platforms for optimal workload distribution. The result is a combination of flexibility and computing power unparalleled in the computer industry.





Left, equivalent plastic strain distribution during sheet metal forming, simulated with the MARC finite element analysis program.

Supercomputing excellence within reach



Above, scientific data enhancement using the AVS visualization system. An output from the ABAQUS finite element analysis program is integrated into an AVS network for analysis of the stress levels within each hexahedron cell of the structure. AVS provides a visual point-and-click interface to computing modules on both the workstation and the Cray Research system.

Maximized system availability

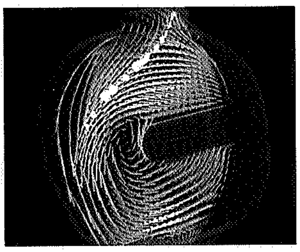
The CRAY Y-MP EL supercomputers provide high system reliability while maintaining high performance. System quality begins with a design process that integrates quality and reliability into every system component. Before shipment, your CRAY Y-MP EL computer system undergoes rigorous operational and reliability tests.

Cray Research offers a wide range of maintenance options for the CRAY Y-MP EL system to meet your needs. To assure high system availability, Cray Research has developed advanced system support tools including the new System Maintenance and Remote Test Environment (SMARTE), which provides continuous error detection and isolation.

The CRAY Y-MP EL system

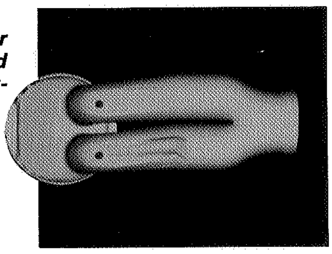
The CRAY Y-MP EL system brings supercomputing excellence within reach with outstanding performance and price/performance. As with all CRAY Y-MP systems, it excels in a wide range of applications. Backed by Cray Research's unmatched experience with total supercomputing solutions, the CRAY Y-MP EL system gives you the power to sharpen your competitive edge.

For more information on the CRAY Y-MP EL supercomputer, contact your local Cray Research representative.



Left, airflow velocity and fuel injection in an indirect injection diesel engine simulated with Cray Research's CRI/TurboKiva engineering program.

Right, a port fuel-injected, four-valve combustion chamber of a gasoline engine simulated with the CRI/TurboKiva combustion engineering program. Here, the air intake and injection of fuel are modeled.



CRAY
RESEARCH, INC.
655-A Lone Oak Drive
Eagan, MN 55121
(612) 683-3801

Image credits in the order they appear:

Computer case image courtesy of Apple Computer, Inc. Beer pasteurization image courtesy of Dr. Michael Engelman, Fluid Dynamics International. Simulation of oil saturation done using UTCHEM, developed by the Department of Petroleum Engineering at the University of Texas. Crystal growth image courtesy of Branko Kosovic, Penn State University. Blood vessel image courtesy of Dr. Clement Kleinstreuer, North Carolina State University. Titanium driver images from an animation produced by Cray Research, MacGregor Golf Company, and the MacNeal-Schwendler Corporation. Sheet metal forming image courtesy of MARC Analysis Research Corporation. AVS image courtesy of Stardent Computer, Inc., and ABAQUS data courtesy of Hibbitt, Karlsson, and Sorensen, Inc. Gasoline combustion chamber image courtesy of Nissan Motor Co., Ltd.

The product specifications contained in this brochure and the availability of the products are subject to change without notice. For the latest information, contact your Cray Research representative.

MCPB-102-0991

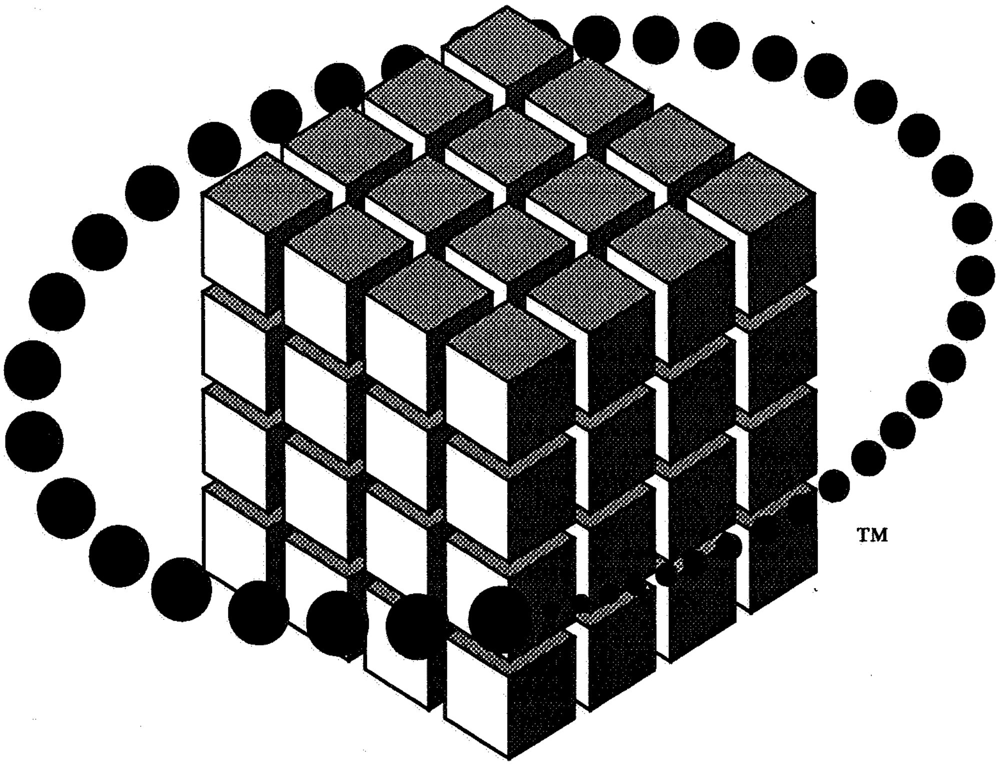
The Cray Research

Massively Parallel Processor System

CRAY T3D

Wilfried Oed
e-mail: wko@cray.com

Cray Research GmbH
Riesstraße 25
80922 München
Germany



November 15, 1993

Subject to change without notice

The special property of digital computers, that they can mimic any discrete machine, is described by saying that they are ***universal machines****. The existence of machines with that property has the important consequence that,* ***considerations of speed apart****, it is unnecessary to design various new machines* ***to do various computing processes****.*

Alan Turing 1950

1. The Potential of Massively Parallel Processor Systems

Alan Turing's quote has not lost any of its validity. Not long before, the mainframe unified all the necessary properties to serve as a "universal machine" for "various computing processes". This concept however is currently being challenged by server concepts, i.e. a set of more or less tightly coupled systems, each designed to fulfill a fairly special purpose.

The criterion "considerations of speed" early on was of importance, especially for scientific applications, and consequently led to the development of supercomputers based on the principle of vector processing. Up until now these systems are, thanks to their highly advanced software, the top of the line for a wide spectrum of applications. In a sense, they have become mainframes themselves.

1.1 Heterogeneous Systems

The availability of highly capable microprocessors, in addition to expanded memory sizes, enables the design and development of a diversity of machines, ranging from the basic requirements of file- and compute-serving to the highest performance. The latter requirement particularly seems promising by massively parallel processor systems (MPP). This will allow to tackle classes of applications that are intractable, or that can be addressed only inadequately, on even the fastest supercomputer systems available today.

On the other hand, the spectrum of scientific and technical applications is so diverse, in terms of both structure and performance demands, that MPP systems should be seen as complementing, i.e. extending, the capabilities of "classic" vector machines. Heterogeneous systems basically offer the potential for efficient and powerful performance, provided the individual components are employed at their appropriate spectrum. Consider for example an application having mixed types of embedded parallelism. Fig.1.1 depicts the units of time required, when being completely executed on a sequential or a vector machine in contrast to the individual portions being ported to a heterogeneous computing environment [KSP93]. This simple scenario takes a basic need for communication into consideration, the quantification however is of utmost importance to make this scenario feasable for real applications.

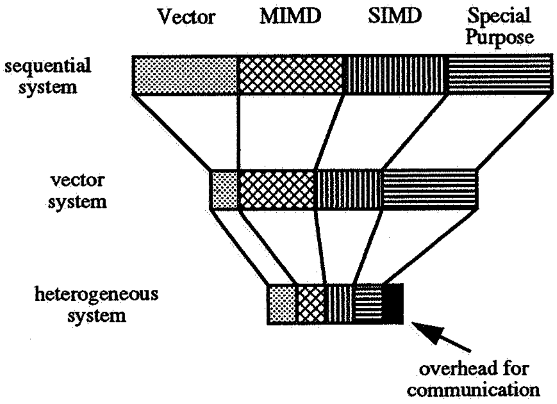


Fig.1.1 Execution of an example code on scalar, vector, and heterogeneous systems; the potential can only be exploited for minimal communication overhead

A qualitative illustration of this situation (fig.1.2) would show a large number of applications with small-to-moderate performance demands typically being done on workstations, a transition to vector systems and their exploitation of moderate parallelism as performance demands increase, and applications with highest performance demands being handled on massively parallel systems.

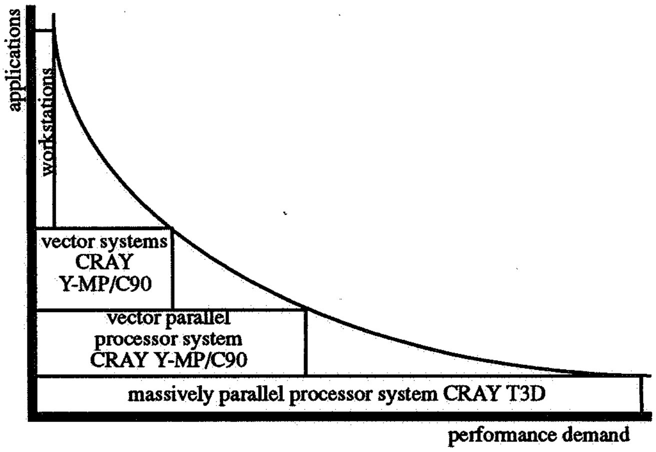


Fig.1.2 Spectrum of applications and performance demand

1.2 Scalability

Amdahl's Law formulates a principle hurdle to be overcome for successfully employing massively parallel systems, i.e. the desired speedup can only be achieved for completely parallelized programs [Amd67].

A solution to this dilamma is comprised under the rather ambiguous term "scalability" [Hil90], which may be characterized by several constraints [SHG93]:

- Constant problem size, i.e. the goal is solving the same sized problem within a shorter amount of time. This case generally will be limited by the scalar part of the code as formulated in Amdahl's Law.
- Memory-constrained scaling, i.e. the size of the problem will be increased linearly with the number of processors. However, this can lead to an unacceptable increase in execution time, since the number of operations required for the solution of a problem size O(n) may be of the order $O(n^2)$ or $O(n^3)$.
- Time-constrained scaling, i.e. the problem size will be increased along with the number of processors, such that the larger problem is solved within the same amount of time as the smaller one on less processors [Gus88]. The success of this approach however requires the scalar part as well the overhead induced by parallelization to remain constant.

In addition to the algorithmic scalability described above, the architecture of the parallel system must be scalable as well [NuA91]. A processor system may only be regarded as scalable, when all components grow adequately. For example, duplicating the number of processors must coincide with a duplication of the memory sizes as well as the bandwidths, since any required communication and synchronization within the system must remain constant.

1.3 Cray Research Strategy

Cray Research has established the goal of delivering a massively parallel processing system capable of solving a broad spectrum of scientific and technical applications at sustained speeds in the TFLOPS range. This project was formally initiated in mid-1989, and was preceded by an approximate two-year period of extensive analysis and research on distributed memory architectures.

Through the use of standard components, such as CMOS DRAM memory chips and RISC microprocessors, it is possible for development to keep pace with the current state of these technologies. Critical for the actual performance of a massively parallel processing system are, aside from the processor speed itself, efficient communication among the processor elements (PEs), i.e. efficient data exchange among their associated memories, as well as support from a flexible programming model. The Cray Research MPP system meets these requirements using a MIMD concept with globally addressable memory, a high bandwidth for data exchange, and extremely fast synchronization mechanisms, which enable the system to efficiently support SIMD style programming as well. The chief factor here is the interconnect of the respective local memories, rather than simply the connections among the processors themselves.

This development needs to be supported by a well balanced combination of hardware and software. Cray Research brings to this task 20 years of experience as the market leader for supercomputer systems. In close collaboration with customers, users, and third-party software vendors, software and applications are implemented in parallel to the hardware development.

The design of the first-generation system, dubbed CRAY T3D, has been completed. By the end of August '93 a prototype 32 PE system has been installed at the Pittsburgh Supercomputer Center, successfully been tested, and now is running in production mode. This system will be upgraded to 512 PEs within the next couple of months.

New RISC microprocessor technologies will enable Cray Research to produce a massively parallel processing system with a peak performance of 1 TFLOPS in the 1995/96 timeframe. Further expected technological advances will lead to a system in the 1997/98 timeframe that will have a peak performance of several TFLOPS and be capable to deliver sustained TFLOPS performance on real applications.

The development of a stable macroarchitecture, i.e. the interface seen by the user, will play a key role in this phased plan. Programming and optimization concepts will scale with each new generation. The underlying microarchitecture will be adapted to incorporate the current state-of-the-art technologies.

2 CRAY T3D System Architecture

The CRAY T3D massively parallel processing system is a scalable MIMD (Multiple Instruction Multiple Data) system with a physically distributed, globally addressable memory. Figure 2.1 shows the relationship between the physical memory organization and address space.

This design allows users to benefit from the advantages of a shared memory parallel system that is suitable for a broad range of applications as well as for parallelizing constructs with low granularity.

2.1 CRAY T3D Compute Nodes

The CRAY T3D compute nodes are comprised of two PEs, each with a CPU, local memory, and a memory control unit, which are coupled by an extremely fast interconnection network, with a transfer rate of 300MB/s per data channel (fig.2.2). The data channels are bidirectional and independent in each of the three dimensions x, y, and z.

The CRAY T3D system employs the DEC Alpha processor, which operates at the same clock speed as the network which is 150 MHz (6.6 ns). This superscalar, single chip processor can initiate a floating-point operation, a load or store, or an integer operation in one cycle. Thus, a nominal peak performance of 150 MFLOPS is achieved, with floating-point operation being performed in 64Bit IEEE format [Sit92].

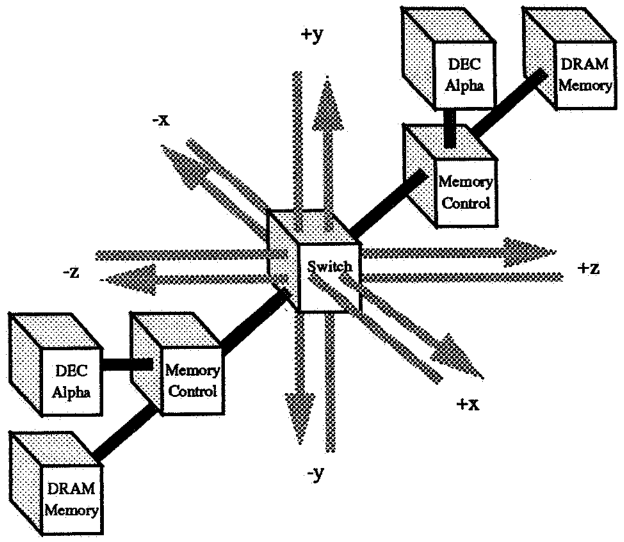
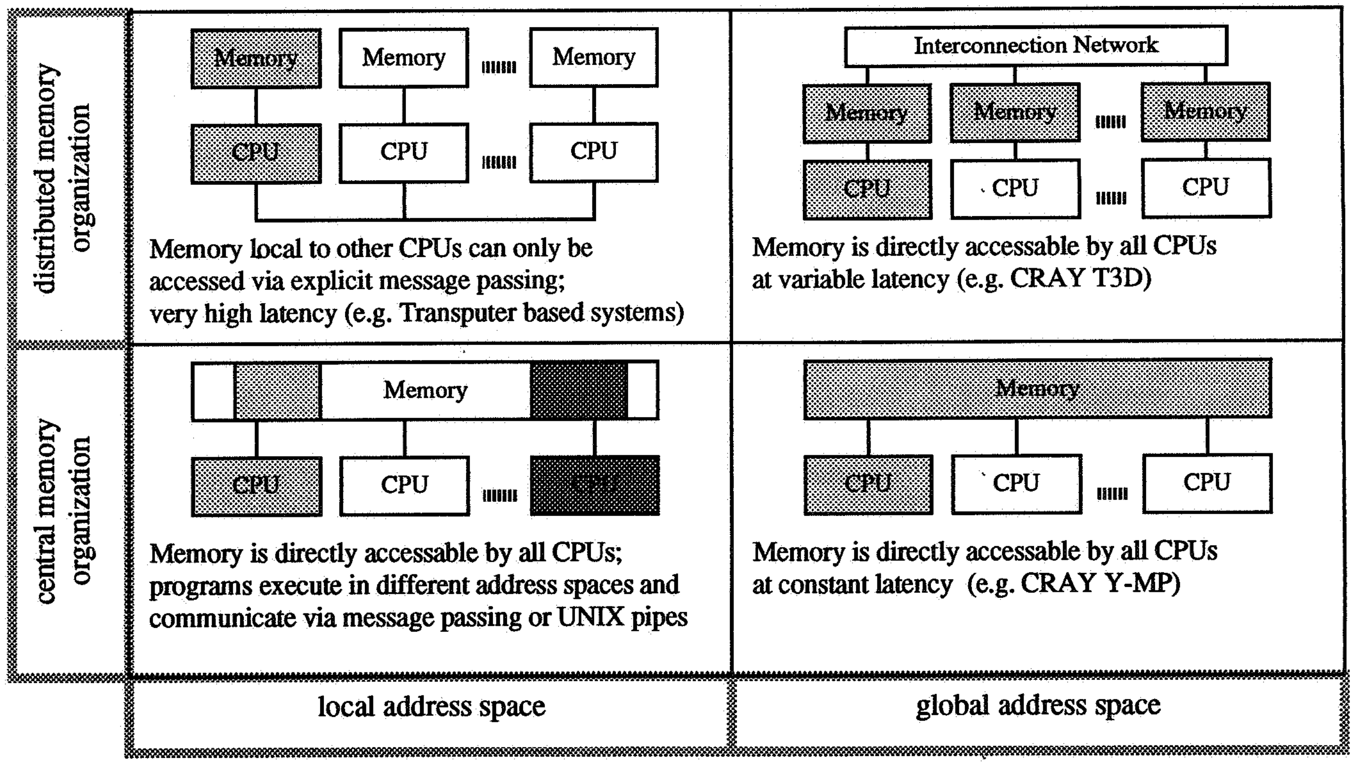


Fig.2.2 CRAY T3D compute node

Local memory either is implemented with 4MBit or 16MBit DRAM chips and has a capacity of 16MBytes respectively 64MBytes.



Memory local to other CPUs can only be accessed via explicit message passing; very high latency (e.g. Transputer based systems)

Memory is directly accessable by all CPUs at variable latency (e.g. CRAY T3D)

Memory is directly accessable by all CPUs; programs execute in different address spaces and communicate via message passing or UNIX pipes

Memory is directly accessable by all CPUs at constant latency (e.g. CRAY Y-MP)

Fig.2.1 Relationship between memory organization and address space

2.2 CRAY T3D Interconnection Network

An efficient, low latency interconnection network is key to a powerful MPP system. There are many cases where only fine granular parallelism can be exploited and consequently a high degree of communication is required.

A class of interconnection networks are the *k*-ary *n*-cubes, i.e. networks with dimension *n* and *k* nodes in each dimension [Dal90]. The higher the dimension, the more nodes are directly linked to each other. However in high dimensional networks, like a hypercube, which is a binary *n*-cube, actual bandwidths between neighboring nodes are usually very low, because of wire constraints. In addition, many physical links in a hypercube network are not being used if the installed system is smaller than the design limit. Therefore, grid topologies have become the preferred implementation.

The CRAY T3D network topology is a three-dimensional torus, i.e. a three-dimensional grid with wrap-around connections (fig.2.3). This topology ensures short connection paths as well as a high bisection bandwidth (up to 76.8GB/s).

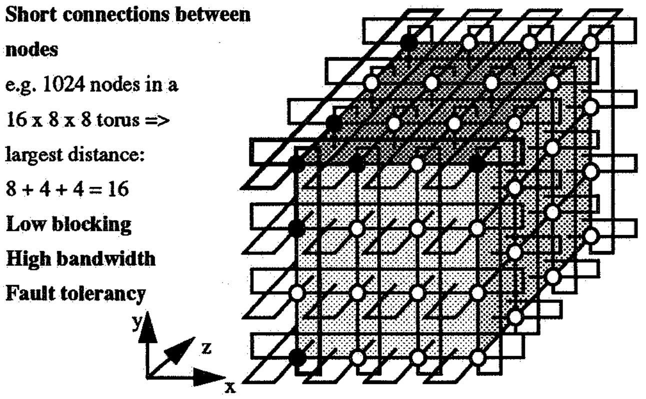
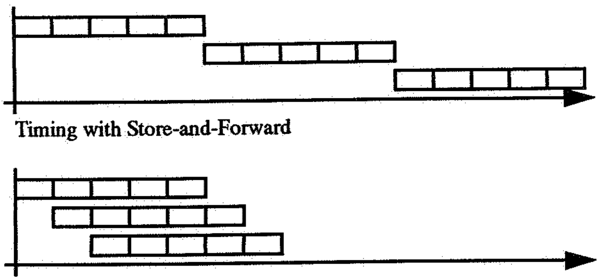


Fig.2.3 3D-Torus network topology

Each PE processes data residing in its local memory independently from the network. Simultaneously, data and messages may be transferred through the node independently over separate communication paths in each of the x, y, z dimensions.

Since the communication paths are bidirectional, locality of communication may be exploited. If node A sends a message to node B, there is a high probability of B sending back a message to A. In a unidirectional network, e.g. a ring topology, a roundtrip will always involve completely circling the machine in at least one dimension [Dal90].

The routing scheme is a combination of "Virtual-Cut-Through" and "Wormhole-Routing" [KeK79, NiM93]. Virtual-Cut-Through is a method like Store-and-Forward, however at the arrival of the first unit of information, a so-called "PHIT" (physical transfer unit), it is directly forwarded to the next node if the path is available. Only in case of conflicts will the packet be stored (fig.2.4).



Timing with Store-and-Forward

Timing with Virtual-Cut-Through (no conflicts)

Fig.2.4 Comparison between Store-and-Forward and Virtual-Cut-Through when transmitting three packets, each consisting of five PHITs

The size of a buffer in each switch is sufficiently large to completely store small packets, in case of larger packets several buffers may be needed, thus becoming wormhole routing. However, the paths are never completely switched nor being blocked for large data transfers. This scheme, in conjunction with virtual channels enable this network to sustain a high throughput.

The routing itself is dimension-ordered, i.e. first the x-direction is completely traversed, then the y-direction, and finally the z-direction. The bidirectional data paths enable the routing in negative directions as well as in positive ones. Which direction will be pursued depends on the routing information locally stored in tables. This scheme in combination with virtual channels guarantees this network to be deadlock free [Dal92]. The routing tables are loaded by software, which on one hand allows the flexible partitioning of the system as well as the isolation of bad nodes or connections. In case of a failing node, a redundant node will be configured in its place. The number of redundant nodes is dependent upon system size.

Furthermore, each switch node contains a so-called Block-Transfer-Engine. This mechanism allows to transfer large blocks of data between local and remote memories or vice versa independently and asynchronously to the PEs. This functionality is of great benefit for example to redistribute the storage order of a matrix from column order to row order.

2.3 CRAY T3D Synchronization Mechanisms

The PEs of the CRAY T3D contain efficient mechanisms for synchronization on various levels of granularity and support control parallelism as well as data parallelism:

- Barrier/Eureka Synchronization
- Fetch-and-Increment Register
- Atomic Swap
- Messaging Facility

An important function for MIMD systems is the barrier synchronization, i.e. all processors wait at a synchronization point until the last one has arrived. This functionality is implemented in the CRAY T3D system by a spearate, tree-like network (fig.2.5). It performs the logical AND (barrier) as well as the logical OR function. The latter function, being called the Eureka mode, is especially well suited when performing parallel search. As soon as a PE has found the object being searched for, it may signal this event to all other PEs, which then immediately may discontiune their part of the search.

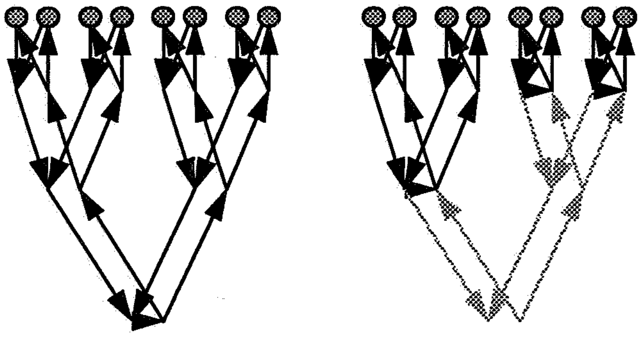


Fig.2.5 Functionality of the independent barrier network; left 8 PEs; right a partitioned network with 4, 2 und 2 PEs each

Each PE has a special Fetch-and-Increment register, which automatically is incremented when being read. This register is not private to a PE, i.e. it may be accessed across the network, which also resolves any potential access conflicts. This concept allows to use this functionality in a central as well as distributed fashion. The latter fashion is especially designed to avoid hot-spot contention. This atomic function is for instance very well suited to dynamically distribute the iterations of parallel loops.

The Atomic-Swap operation exchanges the contents of a special local register with the contents of a memory location, which also may be a remote location. This functionality may be used for "full/empty" or the implementation of locks. Full/empty is checking whether a memory cell contains a valid value (full) or not (empty).

An independently operating Messaging-Facility allows to send a message to a processor, then possibly interrupt and request an answer or not interrupt the PE and let it check for a message at a convenient time. Each PE has such a mechanism and supports efficient message passing, since messages can be sent asynchronously into predefined queues.

These functions, together with the 3D torus network guarantee latencies to only be marginally higher when accessing remote data compared to accessing local data. In addition, because of the independently operating data transfer and synchronization hardware, many operations may be overlapped and latencies mostly be hidden.

2.4 CRAY T3D Configurations

The CRAY T3D system is tightly coupled to the moderately parallel CRAY Y-MP/C90 vector systems, thus forming a seamless supercomputer environment. Two or more I/O-Clusters (IOC) are used to connect peripheral equipment and for network access. These components are connected among each other via CRAY Y-MP high speed channels (HISP). Data are transferred via separate I/O-Interface- Nodes (I/F), which are directly connected to the I/O- Clusters by 400 MB/s high speed channels (fig.2.6).

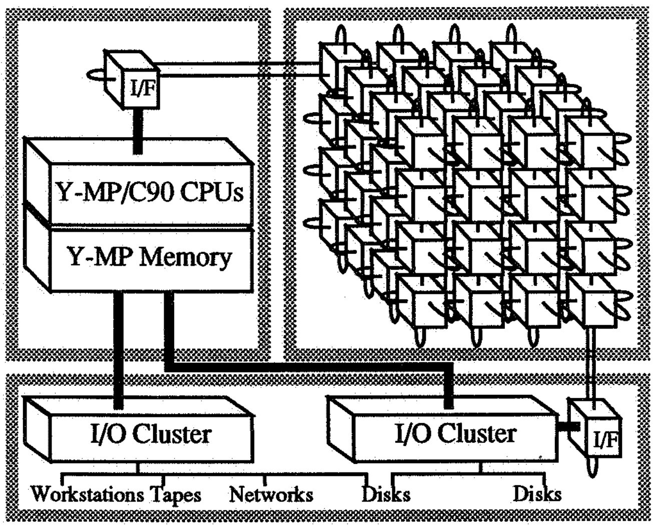


Fig.2.6 CRAY T3D system as part of a supercomputer environment

The interconnection network as well as the synchronization hardware is implemented in the proven CRAY Y-MP and C90 technology, thus enabling the complete system to be uniformly clocked at 150 MHz. A modul houses 8 PEs, their respective local memories and switching components.

CRAY T3D configurations will scale up to 2048 PEs, which translates to a peak performance of 300 GFLOPS. "Single-Chassis" configurations contain all components in one chassis, "Multi-Chassis" configurations will connect to existing CRAY Y-MP Model E or CRAY C90 systems.

3 CRAY T3D Software

The usefulness of massively parallel processing systems in production environments primarily depends on appropriate software and capable programming models. The software environment for the CRAY T3D system features many of the mature software characteristics and capabilities of the Cray Research vector supercomputers, which are themselves moderately parallel systems (CRAY C90 up to 16 CPUs [Oed92]). To support the massively parallel components, there are enhancements in the area of languages, libraries, and analysis tools.

Development efforts are focusing on a spectrum of supercomputer applications that promise superior performance on the CRAY T3D system. A key objective is to maintain compatibility of the environment through successive generations of more powerful MPP systems.

Cray Research provides for its massively parallel system message passing via PVM (Parallel Virtual Machine) for Fortran, C, and C++ as well as a special Fortran programming model. The essential features of this programming model are presented in section 3.1 followed by a comparison to the emerging High Performance Fortran (HPF) specification in section 3.2.

3.1 The Cray Research MPP Fortran Programming Model

For scientific and technical applications Fortran remains the most important programming language, especially since previously missing elements have been added in the Fortran 90 standard. Even this standard, however, contains no explicit constructs for parallel processing, other than the array syntax supporting compilers on SIMD systems.

To express MIMD parallelism in a Fortran program, additional mechanisms are needed. These range from explicit message passing and special language constructs to control by directives. Standardized extensions would be desirable, however the concepts have to account for the key features of the target architecture in the interest of optimal performance. The two most important features to consider are:

- the basic paradigm being employed (SIMD vs. MIMD)

- the physical memory organization (shared vs. distributed) and its logical addressability (local vs. global) as shown in figure 2.1.

Moderately parallel systems, like the CRAY Y-MP systems with up to 16 extremely powerful vector processors employ a shared memory organization with global addressing. They therefore are typically not being programmed using message passing. There is no need for explicitely exchanging data among the processors, since all data can directly be accessed with uniform latency.

Cray Research has ten years of experience with programming models on shared memory systems. Macro-, Micro-, and above all Autotasking [Nag90] have been applied with great success to complex production programs. These models support MIMD and SPMD parallelization at the subroutine level, as well as data parallelism at the loop level, the latter for the most part automatic with Autotasking.

In principle, these models could directly be carried over to the Cray Research massively parallel processor system, since it too employs a global addressing scheme. However, the large number of processors prohibits the implementation of a shared memory organization with uniform access latency. Consequently, a programming model targeted for superior performance has to provide means for controlling the data distribution and its alignment to individual local memories, since latencies differ considerably between local and remote access.

The Cray Research MPP Fortran programming model [PMM93] represents a directives based extension to Fortran 77, including the array syntax defined in the Fortran 90 standard. It is a work-sharing model, i.e. each PE executes its own copy of the code. In that, parallel work can be distributed on virtually all levels, from subroutine (MIMD-style), to individual loops (SPMD-style), or array syntax (SIMD-style), while sequential work may either be executed redundantly (for instance initializing private arrays), or PEs will wait on barriers for others to complete. The data distribution functionality was partially taken from Fortran-D [FHK91] and Vienna Fortran [CMZ91], the control mechanisms rely on concepts from Cray Auto-, Micro-, as well as Macrotasking. The essential concepts can be summarized as follows:

- All variables are private by default, i.e. each PE has a private copy in its local memory, and other PEs have no direct access. Global variables are explicitely declared via directives (SHARED), and exist in only one instance. The compiler may generate copies for optimization purposes, similar to a compiler keeping often referenced variables in registers. In addition to being declared as shared, a distribution may be specified for arrays. There is an implicit alignment, since the first element of a distribution always resides on (the virtual) PE 0. Fig. 3.1 shows for 4 PEs a block-wise distribution of array A, and a column-wise distribution for array B.

```
      DIMENSION A(8,8)
CDIR$ SHARED A(:BLOCK,:BLOCK)

 1,1 1,2 1,3 1,4|1,5 1,6 1,7 1,8
 2,1 2,2 2,3 2,4|2,5 2,6 2,7 2,8
 3,1 3,2 3,3 3,4|3,5 3,6 3,7 3,8
 4,1 4,2 4,3 4,4|4,5 4,6 4,7 4,8
----------------+----------------
 5,1 5,2 5,3 5,4|5,5 5,6 5,7 5,8
 6,1 6,2 6,3 6,4|6,5 6,6 6,7 6,8
 7,1 7,2 7,3 7,4|7,5 7,6 7,7 7,8
 8,1 8,2 8,3 8,4|8,5 8,6 8,7 8,8

      DIMENSION B(8,8)
CDIR$ SHARED B(:,:BLOCK)

 1,1 1,2|1,3 1,4|1,5 1,6|1,7 1,8
 2,1 2,2|2,3 2,4|2,5 2,6|2,7 2,8
 3,1 3,2|3,3 3,4|3,5 3,6|3,7 3,8
 4,1 4,2|4,3 4,4|4,5 4,6|4,7 4,8
 5,1 5,2|5,3 5,4|5,5 5,6|5,7 5,8
 6,1 6,2|6,3 6,4|6,5 6,6|6,7 6,8
 7,1 7,2|7,3 7,4|7,5 7,6|7,7 7,8
 8,1 8,2|8,3 8,4|8,5 8,6|8,7 8,8
```

Fig.3.1 Example of a SHARED declaration and distribution

- Parallel execution is an integral part of the model, i.e. all PEs concurrently and independently of each other execute the same code. Purely sequential code segments, which may only be executed once, have to be marked as such by the directives pair MASTER (enter a sequential region), and END MASTER (leave a sequential region). A sequential region is executed by PE0, others wait for completion to assure synchronization.

- Parallelization at the loop level is invoked by the directive DO SHARED. This enables parallelization on a low level of granularity (SIMD-style), or the easy distribution of work to subroutines executable in parallel (SPMD-style). For purposes of optimization, it can be controlled, which indices of a loop are to be executed on which PE. For instance the directive DO SHARED (I,J) ON A(I,J) invokes parallel execution of a (tightly) nested loop over index I as well as J, where the particular index pairs I,J will be executed on that PE in whose local memory the array elements A(I,J) reside. For purposes of dynamic load balancing, adaptive scheduling of parallel loops can be invoked by GUIDED or RANDOM.

- Synchronisation is implicit at the end of a parallel region, e.g. a parallel loop, or at the end of a sequential region. In addition synchronization may be specified explicitely by the directive BARRIER, marking a point, where all PEs have to arrive before execution continues. An essential feature for a shared memory programming model is the capability to serialize code sequences, i.e. all PEs will eventually pass through here, but only one PE at a time. Such critical regions are marked by the directives pair CRITICAL and END CRITICAL. In cases where only the orderly access to shared variables has to be assured, a more efficient method is provided through the directive ATOMIC UPDATE. In that case, parallel execution basically continues, and only the conflicting modification of a shared variable will be guarded, and cache coherency between processor(s) and memory will be provided.

```
      subroutine sc_prod (a, b, n, s)
      real a(n), b(n), s, s_priv
cdir$ shared a(:block),b(:block),s,n
cdir$ master
      s = 0.0
cdir$ end master
      s_priv = 0.0
cdir$ do shared (i) on a(i)
      do i = 1,n
        s_priv = s_priv + a(i)*b(i)
      end do
cdir$ atomic update
      s = s + s_priv
      return
      end
```

global variable s may only be initialized once

parallel execution of index i on that PE where a(i) is local; all data access is guaranteed to be from/to local memory in this example, since b has the same distribution as a

final summation of s must be serialized

Fig.3.2 Programming example for a dot product

- Special intrinsic functions provide basic operations, like reduction (e.g. SUM, PRODUCT, MINVAL), parallel prefix (maintaining intermediate results), parallel scan (maskable parallel prefix operations), and informative functions about data layout (e.g. BLKKCT, LOWIDX, HIIDX, IS_SHARED) or processor space (e.g. MY_PE).

- I/O by default is performed sequentially, i.e. a file is opened once and the access from the various PEs is synchronized. Parallel I/O is invoked by the directive SHARED_IO. In case private data are read, they will be broadcast to every PE, global data will be routed to their destinations as defined by the distribution.

For the purpose of demonstrating some of the directives, the example of a dot product is provided (fig.3.2). Normally, this functionality will not be programmed explicitly, rather the appropriate intrinsic function be used.

A full description of the Cray MPP Programming Model may be obtained via anonymous ftp from: ftp.cray.com

3.2 High Performance Fortran (HPF)

The High Performance Fortran (HPF) Forum consists, under the direction of K.Kennedy (Rice University), of members from over 40 organizations from industry, universities, and research labs. Its goal is to enhance the Fortran 90 standard, in order to have programs execute with best possible performance on MIMD and SIMD parallel processing systems with memory organizations of non-uniform access latency.

Cray Research is an active participant in the HPF Forum and presently is evaluating when to implement this specification. A first draft has been voted on and been released, which however is not a standard yet [Ric92, Lov93]. The basic advantage of HPF is portability, as soon as a standard has been established and also been accepted by the user community. It currently is not clear whether HPF will be capable to achieve high performance on MIMD systems, since it is lacking basic concepts like private variables and explicit synchronization mechanisms. A well founded assessment is premature. A comparison of the basic features of the Cray Research MPP programming model and the proposed HPF standard is shown in table 3.1.

3.3 Message Passing

Message passing is implemented on the basis of PVM 3.0. Since, in principal PVM may be used in heterogeneous environments, extensive modifications were made in the interest of optimal performance, while retaining the standard interface. Thus, existing programs that have been parallelized with PVM are easily ported onto the CRAY T3D system.

Furthermore, the message passing modell PARMACS, which is widely used in Europe, is currently being ported in cooperation with PALLAS GmbH in Germany.

3.4 Operating System Environment

MPP programs execute completely on the CRAY T3D system, distributed applications between the CRAY T3D and CRAY Y-MP systems are supported as well.

A program is initiated under UNICOS on the CRAY Y-MP system via a so-called MPP-agent, which then communicates between UNICOS and UNICOS-MAX (Massively Parallel UNIX), the CRAY T3D microkernel. Each PE holds a copy of the small and efficient, MACH-based microkernel. I/O is under the control of UNICOS with direct data transfer from the CRAY T3D to the connected I/O-Clusters (fig.2.6).

The MPP program itself may issue UNICOS system calls, which are then communicated by the microkernel to the MPP-agent. Thus, the user always sees the powerful and well proven UNICOS environment.

The system may flexibly be partitioned (space-sharing). The system is split into so-called resource pools, for instance according to criteria like batch or interactive. Subpartitions may then dynamically be allocated within a resource pool.

The NQS batch subsystem has been enhanced for UNICOS to administer the CRAY T3D according to the number of PEs and the time limits requested.

3.5 Analysis Tools

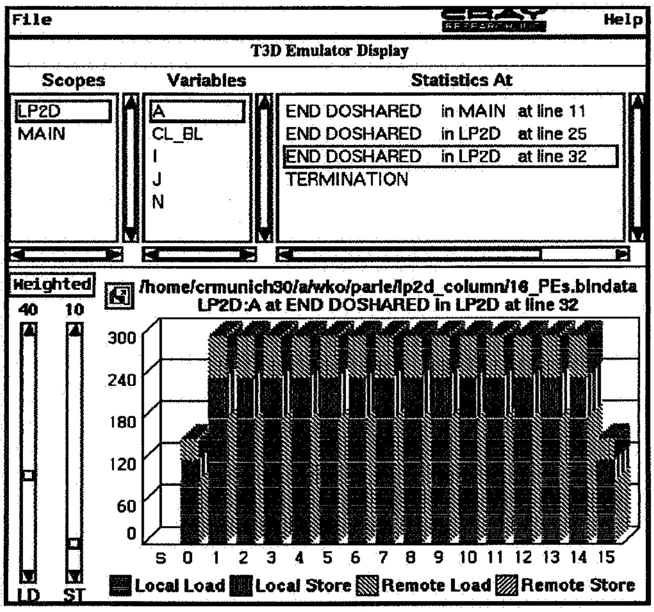
For debugging, *Cray TotalView*$_{\text{TM}}$ allows to analyze Fortran as well as C programs directly on each PE. *TotalView* originally has been developed by Bolt, Beranek, and Newman (BBN) and been customized and extended for the CRAY T3D system.

Cray MPP Fortran Programming Model	High Performance Fortran
Language Elements	**Language Elements**
- Fortran 77 and Fortran 90	- Fortran 90 is prerequisite
+ Automatic arrays	+ FORALL
+ Array assignment, WHERE	+ Some intrinsics
+ Subset of Fortran 90 intrinsics	+ Library routines
+ Additional Intrinsics	
Support for Parallel Execution	**Support for Parallel Execution**
- Array syntax	- Array syntax
- Fortran 90 array intrinsics	- Fortran 90 array intrinsics
	- FORALL
- CDIR$ DO SHARED	- !HPF$ INDEPENDENT
- Library routines	- Library routines
- Sequential execution by CDIR$ MASTER	- Sequential execution is default
- Synchronization via barriers, locks, events, critical regions	- No explicit synchronization mechanisms!
Execution Model	**Execution Model**
- Parallel execution is part of the model	- Execution according to sequential semantic
- Compiler directives influence the semantics	- Compiler directives do not influence the semantics
- Private data is the default	- No private data!
- Explicit declaration of shared data via the SHARED directive	- All data are shared
- All PEs execute the code; possibly redundantly	- All PEs execute a statement
- Message passing may directly be included	- Message passing only via EXTRINSIC functions, in which all data are private (leaving the standard!)
Data Distribution Mechanisms	**Data Distribution Mechanisms**
- Relative positioning implicitely, since the first element resides on PE 0	- Relative positioning statically via ALIGN and dynamically via REALIGN directives
- Distribution on a subroutine level	- Distribution on a subroutine level and within via REDISTRIBUTE directive
- w:BLOCK(m)	- BLOCK
- BLOCK(1)	- CYCLIC
- Collpased	- Collapsed
	- Replicated
- Private data may be broadcast when leaving a sequential region via the directive CDIR$ END MASTER COPY $var_1,var_2,...,var_n$	- n/a
- General templates via GEOMETRY directive	- General templates via TEMPLATE directive
- Redistribution, if necessary upon entering/leaving a subroutine	- Redistribution, if necessary upon entering/leaving a subroutine (no REDISTRIBUTE possible)
Processor Topology	**Processor Topology**
- Predefined (Hardware is a 3D torus)	- Logical model may be defined independent from the actual hardware implementation
I/O	**I/O**
- CDIR$ SHARED_IO	- No parallel I/O (3 proposals were made)

Tab.3.1 Comparison between Cray MPP Programming Model and HPF

Performance analysis is supported by *MPP Apprentice*$_{TM}$, a tool similar to the Autotasking *Atexpert*. Apprentice is targeted to be a performance tool that will truly scale to thousands of processors. It collects statistics on individual PEs, like floating-point arithmetic, cache hits, memory accesses, etc. and uses the power/speed of the MPP to sum the statistics across the PEs. It instruments optimized code to gather its statistics. It relates the statistics it gathers back to the user's original source code via the browser.

Early program development and/or porting may be undertaken by the use of the MPP emulator, which is available with release 6.0 of the cf77 compiling system. The emulator executes on CRAY Y-MP and CRAY C90 systems and allows to verify programs that have been parallelized for the CRAY T3D system (both with PVM and the MPP Fortran programming model) as well as perform some tuning, since statistics of data access (local or remote) are provided (fig.3.3).



LP2D
MAIN

A
CL_BL
I
J
N

```
END DOSHARED   in MAIN   at line 11
END DOSHARED   in LP2D   at line 25
END DOSHARED   in LP2D   at line 32
TERMINATION
```

/home/crmunich90/a/wko/parle/lp2d_column/16_PEs.bindata
LP2D:A at END DOSHARED in LP2D at line 32

Fig.3.3 Example output of an emulator run

4. CRAY T3D Performance

Since the CRAY T3D is a new system, benchmarking still is an ongoing process. Basic communication performance and results of the NAS Parallel Benchmarks shall be reported here.

The currently published results for the NAS Parallel Benchmark comprises the complete suite for 64 PEs and 128 PEs (tab.4.1). The report [BBD93] also shows that the CRAY T3D system is superior to any other MPP system on a comparable number of PEs.

Code	64 PEs [sec]	Y-MP / 64 PEs	128 PEs [sec]	Y-MP / 128 PEs
EP	9.70	12.97	4.80	26.24
MG	3.03	7.33	1.57	14.15
CG	6.14	1.94	3.74	3.19
FT	4.53	6.35	2.33	12.35
IS	3.42	3.35	1.75	6.54
LU	102.00	3.27	55.40	6.02
SP	159.90	2.95	82.80	5.70
BT	153.70	5.16	78.90	10.04

Tab.4.1 NAS Parallel Benchmark results

A key performance factor of an MPP system is a highly efficient communication. The communication performance as seen by the user the depends on the chosen communication mechanism, ranging from native PVM with the lowest performance, because of all the software overhead incurred, over fast PVM implementations ("PVM channels") to the direct use of the global addressing hardware. The latter one is used internally by the MPP Fortran Programming model, in addition can be used explicitely in a message passing style by calling low level routines (`get` for loading remote data, `put` for storing to a remote location).

Latencies that include the respective software overhead have been measured from well below 1µs for the low level communication routines, 2-10µs for the PVM channel routines, and 30-40µs for the native PVM implementation.

Basic communication tests and their parameterization have been described by Hockney [Hoc91]. Two types of ping-pong tests have been developed measuring the startup time, the peak transfer rate (r_∞), and the size of packets for which half of the peak transfer rate is achieved ($n_{1/2}$). "Comms1" sends data from PE to another which then returns it, while "Comms2" has two PEs sending data to each other. Table 4.2 shows the extremely short startup times, high sustainable bandwidth, and the very short data packets required for achieving the half performance mark on the CRAY T3D system. These results show the communication performance of the CRAY T3D system to be typically two orders of magnitude higher than that of other MPP systems.

Test	r_∞(MBytes/s)	$n_{1/2}$(Bytes)	Startup(µs)
Comms1	106	161	1.5
Comms2	188	250	1.3

Tab.4.2 Comms Tests from Genesis Benchmark

5. References

[Amd67] Amdahl G.M. *Validity of the single-processor approach to achieving large scale computing capabilities* Proc. AFIPS 1967, 483-485

[AnS91] Anderson R.J., Snyder L. *A Comparison of Shared and Nonshared Memory Models of Parallel Computation* Proc. IEEE, Vol.79, No.4, April 1991, 480-487

[BBD93] Bailey D.H., Barszcz E., Dagum L., Simon H. *NAS Parallel Benchmark Results* RNR Technical Report RNR-93-016, October 27, 1993

[BDG93] Beguelin A., Dongarra J., Geist A., Sunderam V. *Visualization and Debugging in a Heterogeneuos Environment* IEEE Computer June 1993, 88-95

[CMZ91] Chapman B.M., Mehrotra P., Zima H.P.. *Vienna Fortran - A Fortran Language Extension for Distributed Memory Multiprocessors* Report 91-72, ICASE, Sept.1991

[Dal90] Dally W.J. *Performance Analysis of k-ary n-cube Interconnection Networks* IEEE Trans. on Computers June 1990, 775-785

[Dal92] Dally W.J. *Virtual-Channel Flow Control* IEEE Trans. on Parallel and Distributed Systems March 1992, 194-205

[EsO91] Escaig Y., Oed W. *Analysis Tools for Micro- and Autotasking programs on CRAY multipro- cessor systems* Parallel Computing 17 (1991) 1425-1433

[FGP91] Felperin S.A., Gravano L., Pifarre G.D., Sanz J. *Routing Techniques for Massively Parallel Communication* Proc. IEEE, Vol.79, No.4, April 1991, 488-503

[Fen81] Feng T. *A Survey of Interconnection Networks* IEEE Computer Dec. 1981, 12-27

[Fox88] Fox G. *Solving Problems on Concurrent Processors* Prentice Hall 1988

[FHK91] Fox G., Hiranandani S., Kennedy K., Koelbel C., Kremer U., Tseng C., Wu M. *FORTRAN D Language Specification* Rice Univ. April 1991

[Gus88] Gustafson J.L. *Reevaluating Amdahl's Law* Comm. ACM May 1988

[Hil90] Hill M.D. *What is Scalability?* Computer Architecture News Dec.1990, 18-21

[Hoc91] Hockney R. *Performance parameters and benchmarking of supercomuters* Parallel Computing 17 (1991), 1111-1130

[KeK79] Kermani P., Kleinrock L. *Virtual cut-through: A new computer communication switching technique* Computer Networks, vol.3, 1979, 267-286

[KPS93] Khokhar A.A., Prasanna V.K., Shaaban M.E. *Heterogeneous Computing: Challenges and Opportunities* IEEE Computer June 1993, 18-27

[Lov93] Loveman D.B. *High Performance Fortran* IEEE Parallel & Distributed Technology, Feb 1993, 25-42

[Nag90] Nagel W.E. *Exploiting Autotasking on a CRAY Y-MP: An improved software interface to multitasking* Parallel Computing 13 (1990), 225-234

[NiM93] Ni L.M., McKinley P.K. *A Survey of Wormhole Routing Techniques in Direct Networks* IEEE Computer Feb. 1993, 62-76

[NuA91] Nussbaum D., Agarwal A. *Scalability of Parallel Machines* Comm. ACM March 1991, 57-61

[Oed92] Oed W. *CRAY Y-MP C90: System features and early benchmark results* Parallel Computing 18 (1992) 947-954

[PMM93] Pase D.M., MacDonald T., Meltzer A. *MPP Fortran Programming Model* Cray Research Feb. 1993

[Ric93] Rice University *High Performance Fortran Language Specification* Mai 1993

[Sie90] Siegel H.J. *Interconnection Networks for Large -Scale Parallel Processing* McGraw Hill 1990

[SHG93] Singh J.P., Hennessy J.L., Gupta A. *Scaling Parallel Programs for Multiprocessors: Methodology and Examples* IEEE Computer July 1993, 42-50

[Sit92] Sites R.L. *Alpha Architecture Reference Manual* Digital Press 1992