intel®

Paragon™ XP/S

Paragon™ XP/S

Product Overview

The information in this document is subject to change without notice.

Executive Summary

The Intel Paragon™ XP/S supercomputer provides the highest levels of computational performance and capacity for Grand Challenge computing applications. The Paragon system delivers:

- A balanced and fully scalable parallel architecture.
- A range of standard configurations providing from 5 to 300 GFLOPS peak performance and 2.8 to 160 KMIPS.
- Up to 128 Gbytes of main memory.
- Aggregate system interconnect bandwidth of 12.8 Gbytes/sec for a 300 GFLOPS system.
- Individual node-to-node and node-to-I/O interconnect bandwidth of 200 Mbytes/sec.
- Up to a terabyte or more of internal secondary (disk) capacity.
- Scalable I/O and convenient access to tertiary storage.

The Paragon system also offers high-performance programming, with features that include:

- An integrated software development environment, with a Motif*-based graphical user interface, broad-based language support, and tools for parallelizing sequential applications.
- A versatile MIMD architecture supporting the full range of programming models: sequential, parallel, and vector.
- An integrated performance analysis system offering noninvasive performance instrumentation and low-overhead software monitoring.
- Intel's ProSolver™ matrix solver family and other application-specific libraries.
- Virtual memory and shared virtual memory support.
- Visualization capabilities via the X Window System*, the Distributed Graphics Library* (DGL), the Advanced Visualization System (AVS*), and remote HIPPI frame buffers.
- Full compatibility with Intel's iPSC®/860 supercomputers.

The Paragon system is designed for seamless integration into production supercomputing environments. Usability is assured through:

- A UNIX* operating system that offers high performance, multiuser access, direct logins, and scalable, parallel system services.
- Versatile scheduling and accounting services that support simultaneous interactive and batch processing, as well as client/server operation in a distributed computing environment.
- Broad use of industry connectivity standards, ranging from HIPPI and Ethernet* to NFS* and UniTree*.
- Convenient, economical operation as a result of air cooling, conventional AC power supplies, and extensive EMI controls.
- High-reliability engineering, including selective redundancy and a built-in subsystem- and component-level diagnostic system.

The Architecture

Developing Applications

Production Use

Introduction to the Paragon™ XP/S System

The Intel Paragon XP/S system offers the highest levels of performance, flexibility, usability, and programmability for the most demanding supercomputing applications. The Paragon system delivers scalable performance of up to 300 GFLOPS, backed with an aggressive architecture that will yield affordable, practical teraFLOPS by the middle of the decade. The system is designed for effective usage in distributed, production supercomputing environments, and includes a comprehensive suite of tools for developing and porting applications that realize high sustained performance.

Using the Paragon system's high performance and functionality, scientists and engineers can tackle grand challenge computational problems in more meaningful ways. They can investigate questions that were previously too big to tackle, analyze problems in greater detail, have greater confidence in the accuracy of their results, and obtain results in hours and minutes rather than months and weeks.

Maximum-Capacity Supercomputing

From computational nodes to I/O to visualization facilities, the Paragon system is designed for balanced, sustained performance.

Paragon XP/S configurations offer peak computational speeds ranging from 5 to 300 GFLOPS. This computational performance is matched by a high-bandwidth low-latency interconnection network that passes messages between any two nodes in the system at rates of 200 Mbytes/sec (full duplex).

Every aspect of the Paragon architecture is scalable. Main storage capacity scales to 128 Gbytes of dynamic RAM and more than a terabyte of high speed internal disk storage. Peripherals and network interfaces are scalable -- multiple channels for SCSI-2, VME, Ethernet, and High Performance Parallel Interface (HIPPI) to meet any I/O requirement. The bandwidth of the system's interconnection network rises with the number of nodes in the system.

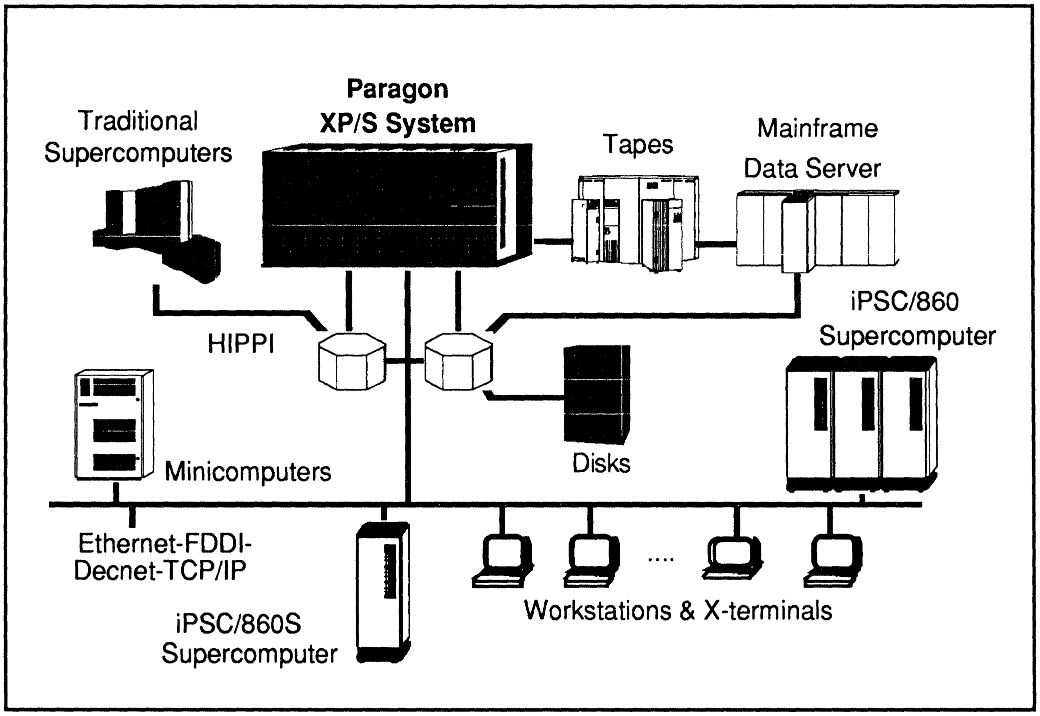
The Paragon operating system, a full implementation of UNIX, also scales, ensuring that delivered operating system services match hardware performance as system size increases. Once applications are running on the Paragon system, they too become scalable. For example, a computational fluid dynamics code could be developed and tested on a small number of nodes. When ready for production, the number of nodes used for the production runs can be based on the desired performance -- 400 nodes or 1,000 nodes could be employed without change of the application.

The system can scale to match the size of the data set without changing the application or the time it takes to get the answer. For instance, often the same algorithm can be used for a larger data set, allowing the user to model an entire structure rather than just a portion of the structure. Using the same application on a larger number of nodes allows the use of these larger data sets while retaining the quick turnaround time of the answer.

Paragon's scalability means that supercomputer centers can configure the system with the blend of computing power, memory, storage, connectivity, and software that precisely matches their application requirements — without resorting to custom engineering projects. It also provides native expandability through the ability to incrementally upgrade and dynamically reconfigure the system.

Designed for Production Supercomputing

Paragon's transparently distributed UNIX operating system — the first full UNIX implementation for a massively parallel supercomputer — ensures flexibility in managing the Paragon system, developing applications, and integrating the system into heterogeneous distributed computing environments. The operating system is the result of advanced developments in distributed operating system services and microkernel technology, and delivers full UNIX services to every node in the machine.



The Paragon XP/S system in the technical computing environment.

Adding to the system's ease of use and further integrating parallel supercomputing into the technical computing environment, the Paragon system supports client/server computing via scalable HIPPI, FDDI, and Ethernet networking. For flexible system access, users can submit jobs over a network or can log directly onto any Paragon node. The Paragon system can be used for batch and interactive jobs simultaneously, and can be reallocated dynamically to adjust the proportion of interactive and batch services. Comprehensive resource control utilities are available for allocating, tracking and controlling system resources.

A Rich Environment for Applications Development

Intel's proven, verastile MIMD architecture supports all programming styles and paradigms. Applications can be developed using the programmer's preferred programming model -- object-oriented, Single Program Multiple Data (SPMD), Single Instruction Multiple Data (SIMD), Multiple Instruction Multiple Data (MIMD), Shared Memory, or Vector Shared Memory.

Paragon parallel computer-aided software engineering (CASE) tools assist application developers in writing and porting applications to fully realize the system's performance potential. The development tool suite is integrated through a common database and a Motif*-based graphical user interface. Tools include optimizing compilers for FORTRAN, C, C++, Ada, and Data Parallel FORTRAN; the Interactive Parallel Debugger (IPD); parallelization tools such as FORGE* and CAST*; Intel's highly-tuned ProSolver library of equation solvers; Intel's Performance Visualization System (PVS™) and the Performance Analysis Tools (iPAT™).

Beyond Traditional Supercomputing

The Paragon system's massively parallel architecture provides performance and flexibility beyond the reach of traditional vector supercomputers:

- **Affordable performance**. Intel's expertise and leadership in microprocessor technology is approaching the performance of specialized supercomputing processors thousands of times more expensive. The high-volume, commodity nature of supercomputing microprocessors guarantees that this will continue and eventually surpass custom processor design.
- **Transparent software scalability.** Once an application is running on the Paragon XP/S system, it can run without change on any number of nodes. Users can move back and forth from a 5 GFLOP to a 300 GFLOP performance level with no reprogramming or system reconfiguration.
- **Simultaneous interactive and batch operation**. The Paragon system doesn't force the users to decide between operating in batch mode or interactive mode -- all modes of operation are provided simultaneously and all system resources can be dynamically reallocated as the system is running -- allowing data centers to support a mix of users and application types.
- **Native or networked development**. Developers are given the option to choose for themselves, using software tools directly on the system in native mode or on their own workstations, downloading their applications. This flexibility means the system fits the user's application, not the other way around.

The Road to TeraFLOPS

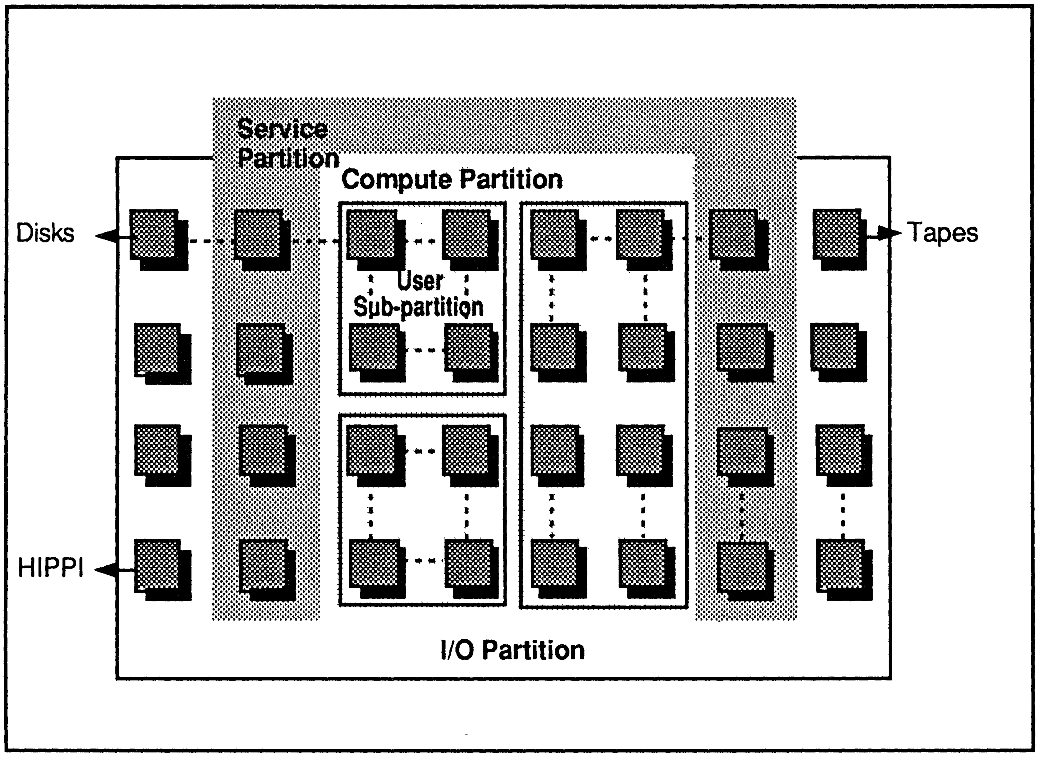
A Paragon XP/S system with TeraFLOP performance could be built today, but awaits a new generation of microprocessors, memories and other key VLSI components to make it truly affordable. Intel has driven microprocessor performance for more than twenty years since it first invented the microprocessor in 1971. Recognizing the challenge of boosting the performance of a supercomputer on a chip, Intel has redoubled its efforts and reorganized the company to increase the performance of the processor by an order of magnitude over the next five years. By mid-decade, advances in submicron semiconductor processes and multi-chip packaging technology will converge with advancements in system software and scalable applications, to enable Intel to deliver the first affordable supercomputer capable of teraFLOP performance.

The Paragon™ XP/S System At A Glance

Capacity	5 to 300 GFLOPS peak 64-bit floating-point performance 2.8 to 160 KMIPS peak integer performance Node-to-node message routing at 200 Mbyte/sec (full duplex) 1-128 Gbytes main memory, up to 500 Gbyte/sec aggregate bandwidth 2 to 128 Mbytes processor cache, up to 4.0 Tbytes/sec aggregate bandwidth 6 Gbytes to 1 Tbyte internal disk storage, up to 6.4 Gbyte/sec aggregate I/O bandwidth
System Architecture	Scalable, distributed-memory multicomputer MIMD control model
Node Architecture	Nodes based on Intel's 50 MHz i860™ XP processor 75 double-precision MFLOPS, 42 VAX MIPS peak performance per processor 16-128 Mbytes DRAM per node 400 Mbytes/sec processor-to-memory bandwidth 800 Mbyte/sec processor-to-cache bandwidth
Interconnect Architecture	2D mesh topology Pipelined, hardware-based internode communications
Operating System	Full UNIX operating system Transparent, distributed scalable services Conformance with POSIX, System V.3, 4.3bsd Virtual memory
System Access	Simultaneous batch and interactive operation NQS, MACS utilities for resource management Client/server access, direct logins, remote host
Connectivity	Multiple HIPPI channels with 100 Mbytes/sec bandwidth each Multiple Ethernet channels Multiple VME* connections NFS, TCP/IP, DECnet* protocols, UniTree client support
Programming Environment	C, FORTRAN, Ada, C++, Data-parallel FORTRAN Integrated tool suite with a Motif-based GUI FORGE and CAST parallelization tools Intel ProSolver parallel equation solvers BLAS, NAG, SEGlib and other math libraries Interactive Parallel Debugger (IPD) Hardware-aided Performance Visualization System (PVS) Operating system support for shared virtual memory
Visualization Tools	X Window System PEX Distributed Graphics Library (DGL) client support AVS and Explorer interactive visualizers Connectivity to HIPPI frame buffers

The Paragon XP/S Architecture

The Paragon system employs a scalable, heterogeneous multicomputer architecture. Its various processing nodes populate a two-dimensional interconnection network. Nodes provide a variety of services including computation, input/output services, operating system services, and network connectivity. Main memory is physically distributed among the nodes.



The architecture of the Paragon system is reflected in the implementation: flexible, scalable, usable, and modular. All programming models are supported because the topology of the interconnect network allows the programmer to ignore the physcial location and function of the nodes. Systems are divided into partitions, which can consist of as little as one node or as much as all of the nodes. In this example, the system is configured with 8 nodes dedicated to I/O interfaces, 8 nodes to system services, and the remainder to the user applications.

The system makes extensive use of Intel microprocessor technology. Each employs at least two Intel i860 XP processors: one or more *application processors* dedicated to executing a user's application program, and a *message processor* dedicated to the task of internode communication.

The Paragon interconnection network establishes a new level of performance by providing low latency and high bandwidth between every node in the system. Two nodes, anywhere in a Paragon system, achieve process-to-process transfer latency of 25 microseconds — regardless of the size and configuration of the system. Messages are routed automatically, so the application programmer sees a system in which every node is connected to every node, with no difference in performance among nodes.

Balanced Design

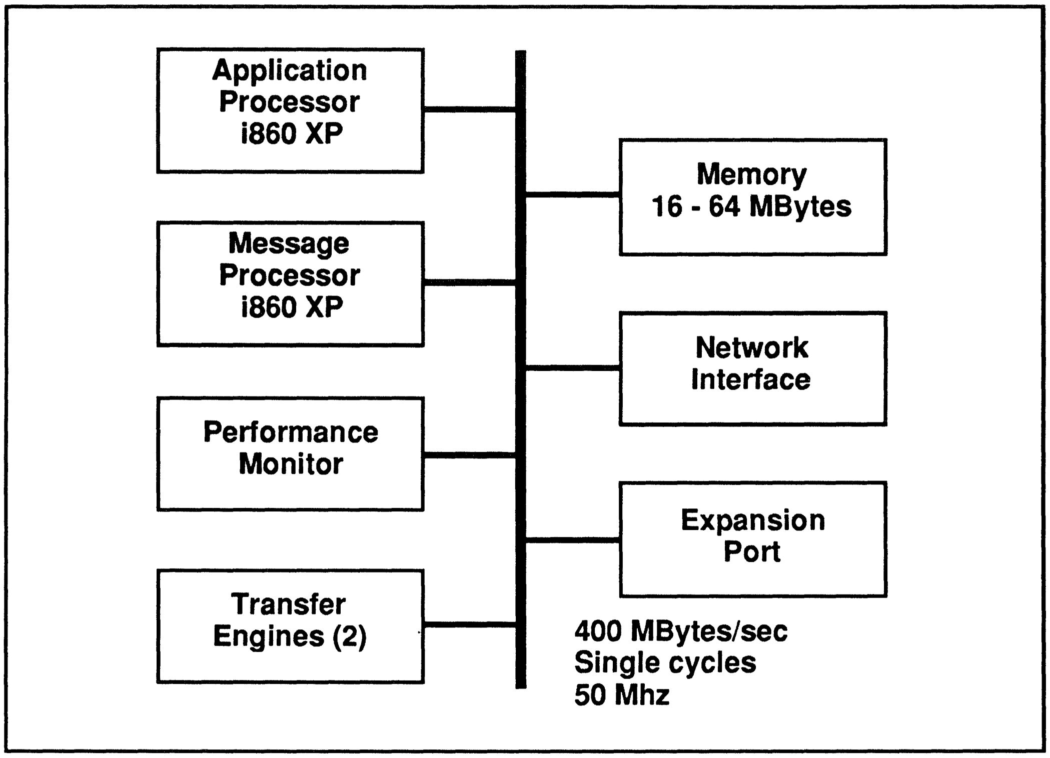
The key to high sustained performance in a multicomputer is balanced design: the speed and memory capacity of the individual nodes must be matched by the system's interconnection network, mass storage facilities, graphics devices, and network connections. The design of the Paragon XP/S system is a model of balance in all these areas.

Within each node, the microprocessor speed is matched by on-chip cache performance and high-bandwidth memory units. Aggregate interconnect bandwidth scales with the number of nodes and supports the efficient operation of systems with many thousands of processors. The bandwidth and latency of the node-to-network interfaces and node-to-node communication channels are carefully matched to the node memory bandwidth and execution speeds. Aggregate I/O performance, like computational performance, scales with the number of I/O nodes. The I/O interfaces are custom engineered to the needs of specific industry-standard bus and channel interfaces. With greater bandwidth than virtually any peripheral device, point-to-point I/O channel, or local area network, the interconnect sustains this outstanding I/O performance from or to any node in the system.

Special attention has been paid to low-latency, high-bandwidth internode communication. Dedicated programmable and fixed-function messaging hardware guarantees that applications achieve the maximum performance from the most time-critical sections of the operating system. Not only is sustained performance increased, but sophisticated messaging operations, such a broadcasting and other global operations, are reliably and efficiently implemented.

Node Architecture

Balance within each Paragon node is the result of engineering the memory unit, messaging facilities, and external I/O interfaces to match the performance demands of the Intel i860 XP microprocessor. Capable of 42 MIPS and 75 double-precision MFLOPS when operating at 50 MHz (20 ns clock cycle), the 2.55-million-transistor Intel i860 XP provides four-way set associative, on-chip, 16-Kbyte instruction and data caches. Floating-point-unit-to-cache bandwidth peaks at 800 Mbytes/second. Cache refills to local memory take place at 400 Mbytes/sec and are fully supported by an interleaved, dual-bank, single-cycle, 64-bit memory unit design. Local memory is based on 60 ns DRAM arrays with full single-bit error correction and double-bit error detection.



The Paragon node is modular in architecture and implementation, with the network interconnect boosted with a dedicated Intel i860 XP separate from the i860 XP processor used for user applications. The design also benefits from a performance monitor with ties to all the major components and a connection to its own (optional) data collection network.

The basic Paragon node design is used within the system in several ways: computation, input/output, and service. Individual compute and service nodes may be configured with various memory options from 16 Mbytes to 128 Mbytes. Input/output nodes may be configured with from 16 Mbytes to 64 Mbytes of memory and one of several I/O interfaces, as described below.

MP Technology

A second Paragon node design, currently in development, incorporates multiple processors per node. This MP node architecture dramatically increases both the volumetric and the peak performance capacity of the Paragon XP/S system. Innovative packaging and cache memory technology enables operating speeds well above the nominal 50 MHz clock of the Paragon XP node design. Wide-word memory units maintain the necessary bandwidth to satisfy cache refills and pipelined processor loads and stores.

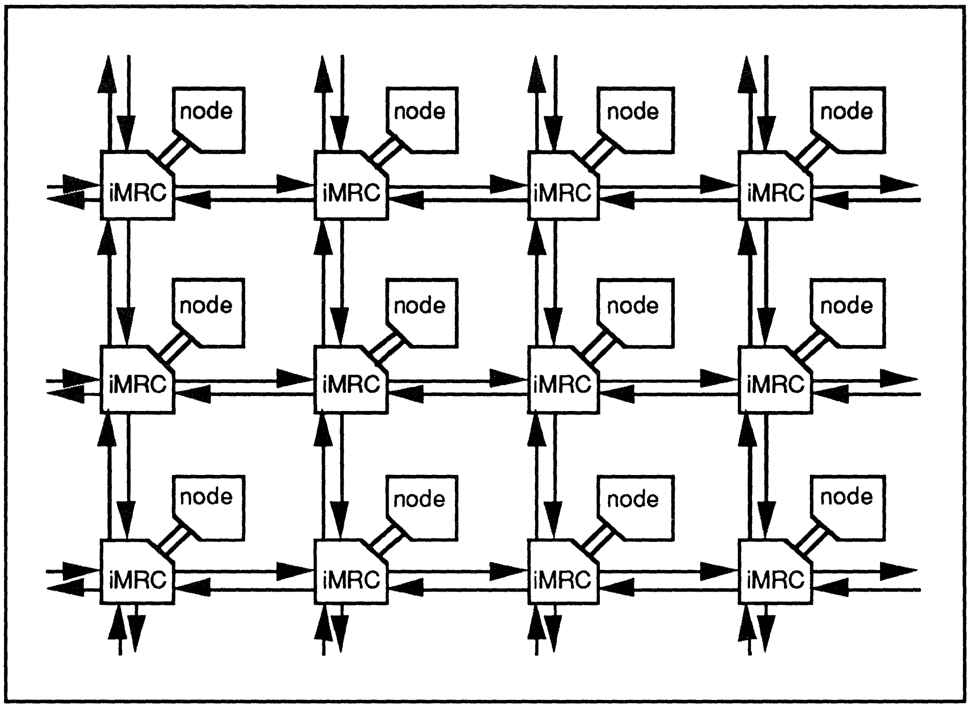
Interconnection Network

The Paragon interconnection network is the culmination of a decade of research and development in multicomputer architecture. Beginning at Caltech with the leadership of Professor Charles Seitz, expanding at MIT under Professor William Dally, and continuing at Intel as part of the joint Intel-DARPA Touchstone project, this line of research has given the Paragon XP/S system the most advanced multicomputer interconnection network ever designed.

Interconnect performance is fundamentally determined by the number of wires that cross from one half of the network (the bisection) to the other. Networks with more wires, assuming they are equally short, should be faster than networks with fewer wires. In practice, the number of wires crossing the bisection is limited by electrical factors, such as power dissipation, and mechanical factors, such as trace, connector, and cable density. Designing an optimal network is principally an effort to make the most efficient use of the available wires, given that the wire limit is essentially the same at a given moment in time for all practical designs.

Extensive analytical studies, thousands of hours of simulation, and several prototype systems (most notably the Touchstone DELTA and SIGMA systems) have demonstrated that low-dimensional mesh networks make the most efficient use of available wires. That means, given an equal number of wires at the bisection of the network, a two-dimensional mesh will outperform any toroidal, hypercube, or tree-structured network for uniformly distributed communication traffic in systems containing up to several thousand nodes. The mesh advantage actually *grows* as the communication traffic becomes more localized.

To deliver on the promise of the mesh topology, the switching elements and electrical design of the Paragon XP/S interconnection network have been carefully considered. The individual switches, located at each vertex in the network, are fabricated in the same 0.75 micron, triple-metal, CMOS technology used to build the Intel i860 XP. Each Paragon Mesh Routing Chip (PMRC) can simultaneously route traffic moving in the four network directions (±X and ±Y), and to and from its attached node, at speeds in excess of 200 Mbytes/sec. It takes the PMRC less than 40 ns to make an individual routing decision and close the necessary switches. Internal operation of the router is fully parallel, and all transfers are parity checked. To maximize router bandwidth, it is fully pipelined both internally and externally. Message traffic moving from one router to another is pipelined along the wires so that speed becomes independent of distance for all practical purposes. The routing algorithm is provably deadlock-free.



The Paragon architecture utilizes a two-dimensional mesh network topology, originally developed under the Intel-DARPA Touchstone project. Every node, regardless of its function in the system, utilizes the interconnection network, permitting every system service and function to scale with the system as it grows to more than a 1000 nodes.

Instead of a passive backplane, as used in most conventional systems, the Paragon system uses an active backplane built from a rectangular arrangement of router chips. Each backplane section connects to its horizontal and vertical neighbors by flexible printed circuits and ultra-reliable connectors. Open the back door of a Paragon system and the backplanes are completely clean. Gone are the fragile cables and wiring mat that have become the trademarks of conventional vector supercomputers and are still found in many highly parallel machines. Also missing are the critically measured and cut clock control lines, as the Paragon interconnect is completely self-timed and clock-free. Only the Paragon mesh routers are visible as they quietly go about the business of moving information between the nodes at speeds limited only by their underlying VLSI technology and the speed of light. For systems with as few as 256 nodes, the bisection bandwidth is in excess of 6 Gbytes per second. A Paragon system with 1,000 nodes has a bisection bandwidth of over 12 Gbytes per second.

Progress in interconnect architecture and router design continues at a very rapid rate. Anticipating future developments, the Paragon interconnect is open to future technology insertion. Developments are already underway that will increase the bandwidth of the network to over 25 Gbytes per second for 1,000-node systems and provide greater routing efficiency for messages travelling long distances.

Node-Network Interface

Studies for the Intel-DARPA Touchstone prototypes revealed that fast nodes and a fast interconnect do not guarantee balanced internode communication. An often-overlooked design element, the *node-network interface,* is key to unlocking overall system performance. It is here that Paragon has moved further than any extant parallel system design.

A semi-custom VLSI chip, the Network Interface Controller (NIC), provides a full-bandwidth, pipelined electrical interface between a node and its PMRC. Included in this chip is a parity-checked, full-duplex router port and end-to-end error checking for each message transfer. Internal parallel operation permits simultaneous inbound and outbound communication at 200 Mbytes/sec. Since the node processor cannot sustain this speed for long messages, two block transfer engines are designed into the memory interface. They can move up to 4,096 bytes of data at a time at an aggregate speed just short of the full node memory bandwidth.

To complete the node-network interface, a second full-speed, full function Intel i860 XP *message processor* is also incorporated into the basic node design. Its purpose is to free the *application processor* from the details of internode communication and to provide a variety of advanced communication services within the Paragon system. Operating in parallel, the message processor handles all communication traffic to and from the node. It autonomously receives messages and prepares them for use by the application processor. Similarly, it handles all the details of sending a message, including protocol and packetization, if required.

The message processor has many subtle benefits that become obvious only after understanding the low-level issues of MIMD multicomputer design. By freeing the application processor of messaging details, the message processor reduces cache turbulence in both processors. The application code and data stay in the caches on board the application processor to sustain performance. The messaging software of the operating system is small enough to remain resident in the message processor's code and data caches. This guarantees the lowest possible latency, because library code doesn't first have to fill the processor caches before it can run at full speed. The dual-processor design also avoids a costly context switch in the application processor during messaging operations and while handling interrupts from the network interface. This, in turn, helps avoid draining the floating-point and memory pipelines in the application processor.

The message processor at each node also enables the Paragon system to offer a full range of versatile and efficient global operations that are performance-critical in both MIMD and SPMD applications. The global operations that are autonomously implemented by cooperating message processors include broadcasting to and synchronizing a group of compute nodes, as well as mathematical operations, such as global sum and global minimum, for both integer and floating-point operands. Global operations are also available for logical operands, such as global AND and OR, and for aggregates, such as global string concatenation.

It is important to note that while the message processor's computational resources are utilized by the Paragon system, the performance rating of the message processors are *not* included in the aggregate peak performance ratings of the Paragon XP/S system.

Performance Instrumentation

Supercomputers are fundamentally performance machines. Realizing their performance potential requires balanced hardware design, efficient system software, and good tools for measuring and analyzing application behavior. Recognizing that performance tools are only as good as the data they receive, the Paragon system incorporates a dedicated hardware instrumentation system to capture performance data on the fly.

The Paragon Performance Visualization System (PVS™) is the most comprehensive facility of its kind. Each node is equipped with a VLSI data-capture chip designed by computer-performance experts at the U.S. National Institute of Standards and Technology (NIST). This chip detects and timestamps a variety of hardware events that occur within a node with *zero* overhead. It also captures and timestamps software-defined events via a memory-mapped interface with the overhead of a single memory cycle. When the event rates are moderate and trace intervals short, each node periodically reads out the MultiKron and either stores the data or ships it elsewhere in the system. There, it can be merged with data from other nodes, filtered, and visualized.

For high rate events and extended trace periods, the MultiKron is also connected to a separate performance instrumentation network. When attached to this network, a specialized hardware performance analyzer can be used to conduct extremely detailed studies of system and application behavior. High-bandwidth performance information flows across this dedicated network without disturbing the communication behavior of the application under study.

Mass Storage and I/O

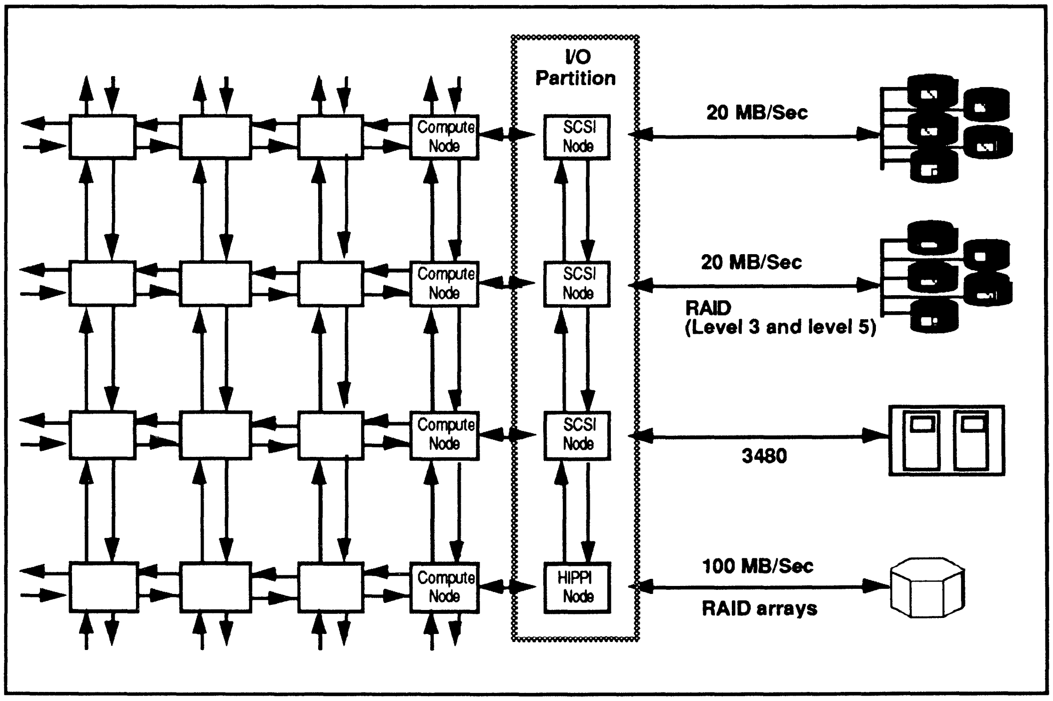
Grand challenge-class applications are distinguished not only by their high requirements for numeric processing speed but by their large data sets, with storage requirements reaching into the range of thousands of terabytes.

The Paragon™ XP/S system offers up to a terabyte of scalable internal mass storage accessible at an aggregate bandwidth of over 6.4 Gbytes/sec. The scalable design also incorporates high-bandwidth links to external disk arrays, storage servers, and wide area networks. The number of interfaces can increase as system size increases — or as applications require additional I/O facilities — and disk access bandwidth to external devices scales as I/O capacity rises.

Scalable I/O Interfaces

To meet diverse I/O requirements, a number of interfaces to I/O devices are provided, with type and number of the interfaces determined by users' applications:

- A fast, 16-bit wide SCSI-2 interface for 20 Mbyte/sec connection to disk and tape devices.
- A HIPPI interface for 100 Mbyte/sec full-duplex connection to stand-alone disk arrays, HIPPI switches, HIPPI frame buffers, or other computer systems.
- A VME interface that supports custom or third-party VME-based peripheral controllers and network interfaces, such as Ethernet.



In this more detailed view of the architecture, the allocation of nodes to a variety of I/O interfaces can be seen. Two RAID disk arrays internal to the Paragon XP/S are shown at the top right, with 5 disks totaling 6 GBytes formatted storage connected to each SCSI interface node. The other interfaces are a 3480 format tape system (SCSI) and an external RAID disk array (HIPPI).

Internal Disk Storage

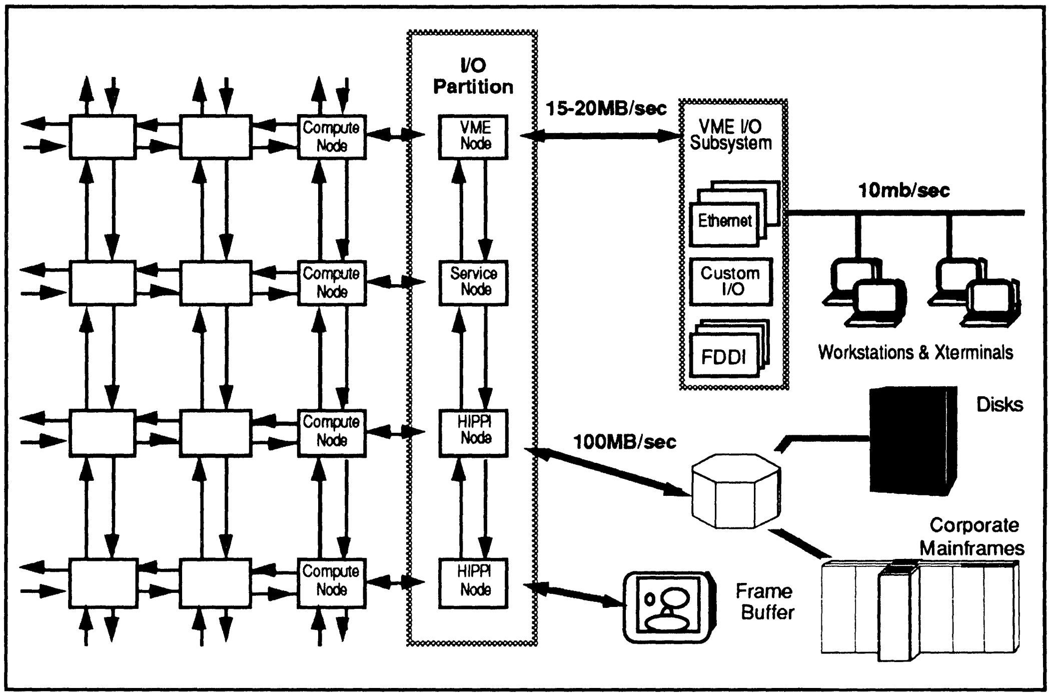
Internal disk storage is supplied via arrays of commodity 5.25-in Winchester devices each with 1.5 Gbytes formatted storage and sustained disk head transfer rates of 3.5 Mbytes/sec. Each disk supports an on-device read-ahead cache, to enable applications to realize more of the potential read bandwidth within the I/O subsystem. Typical system configurations will connect an I/O node with five of these disks via a SCSI-2 interface.

For reliability and cost-effectiveness, each Paragon disk subsystem is protected by hardware RAID technology (Redundant Arrays of Inexpensive Disks). RAID controllers on the Paragon I/O nodes manage multiple disk drives, distributing application data over four data drives and one parity drive, using automated routines to rebuild data in event of a drive failure. Replacement of a failed disk drive and recovery of the data can be done as the system is running, or off-line. The file system remains intact, and does not need rebuilding. The Paragon system offers Levels 0, 3 and 5 RAID functionality.

There is effectively no limit on the number of disk arrays in a Paragon configuration, since the number of I/O nodes can scale with the size of the system.

Internal Tape Storage

Although most data processing centers are expected to utilize the ParagonHIPPI connections to access existing backup facilities, a large number of tape devices can be configured in the system as are needed. The industry-standard Exabyte 8500 8mm devices, each of which holds up to 5 Gbytes of data and transfers data at sustained rates of 500 Kbytes/sec can be included in the configuration. The 8mm tape devices are well-suited to loading new system software, creating "personal" backup files, and backing up small file systems.



This example shows the VME intercase used to connect a variety of peripherals in an external I/O subsystem, as well as Ethernet network connection. A number of tape connections would be found in a typical configuration. Two different uses of the HIPPI connection are also shown — 100 MByte/sec in full duplex is supported.

External Mass Storage

Third-party peripherals and storage systems can be connected to Paragon I/O nodes via a HIPPI interface or SCSI-2 bus interface, to support external disk arrays and IBM 3480-compatible tape devices, respectively. The peak transfer rate for the HIPPI connection is 100 Mbytes/sec (full-duplex), and the SCSI-2 bus transfers data from a string of tape devices at up to 20 Mbytes/sec.

A single HIPPI controller manages a set of third-party RAID storage systems, adding another dimension of scalability to Paragon's mass storage subsystem. A fully-populated array can provide 345 Gbytes of formatted disk storage.

The system supports IBM 3480-compatible tape formats, in order to interchange data with systems that read and write IBM-labeled tapes. For highly secured environments, the system also supports removable cartridge disks. Further system integration facilities are available from third parties to meet additional security needs.

Remote Storage Subsystems

The Paragon system includes mechanisms that aid applications in reading and writing data stored on remote devices. Paragon supports the Network File System (NFS) and File Transfer Protocol (FTP), as well as a UniTree client interface that provides access to UniTree services.

High-Speed Graphics Devices

The Paragon system offers several approaches for connecting to high-speed graphics devices. Nodes can write output to graphics workstations and X terminals connected via Ethernet. For high-speed rendering and highly interactive visualization, users can also connect graphics frame buffers through the system's HIPPI channels.

Paragon™ XP/S UNIX Operating System

The UNIX operating system is the standard for high-performance computing. However, it was originally designed for single-processor machines and its native architecture is ill-suited to the performance needs of massively parallel applications.

The Paragon XP/S system is the first to provide all the familiar features of UNIX to every node of a massively parallel system. The Paragon system's transparently distributed implementation and single "system image" makes every UNIX file, every process, and every network service available to every Paragon application — despite the fact that the operating system is running in parallel on a distributed system of hundreds or thousands of nodes.

Because of the system's full UNIX operating system, users gain increased applications portability, ease in using the system, and ease in integrating the Paragon system into the technical computing environment.

In keeping with the needs of performance-driven supercomputing applications, the architecture of the Paragon operating system delivers efficient message-passing performance, by minimizing the involvement of the operating system and taking full advantage of the interface to the interconnection network. This also means that the performance and the capacity of the operating system increases with the size of the Paragon system.

Standards

The Paragon operating system is derived from two well-known operating systems technologies: the Mach system from Carnegie Mellon University and the Open Software Foundation's OSF/1AD distributed system for multicomputers. This complete UNIX implementation gives users all the popular UNIX features, including virtual memory; utilities, commands, and shells; I/O services; and networking support for *ftp*, *rpc*, and NFS.

The Paragon operating system complies with current and emerging standards for System V Interface Definition Issue 2 (base and kernel extensions), OSF/1*, XPG3, and POSIX specifications and the SVr4 IPC interface.

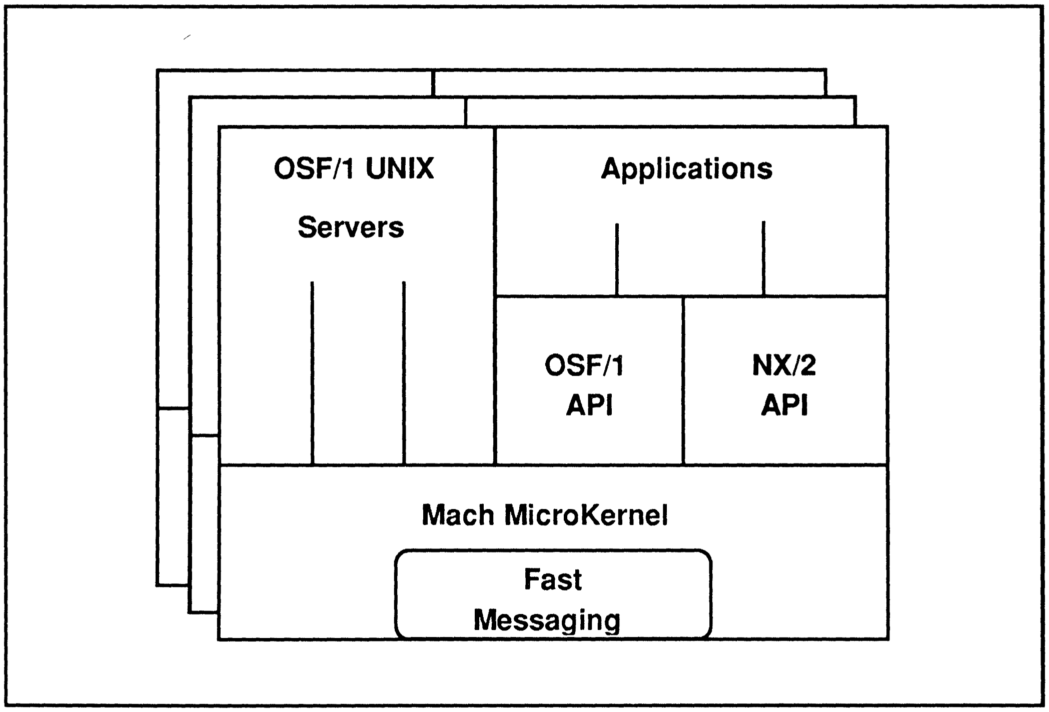
Compatiblity

Source compatibility is supplied by two application program interfaces (APIs) for OSF/1 applications and for iPSC/860 applications authored under NX/2. The Paragon system is upward compatible with the iPSC/860 system from Intel, which can be used as a development environment for the Paragon system.

Distributed Operating System Architecture

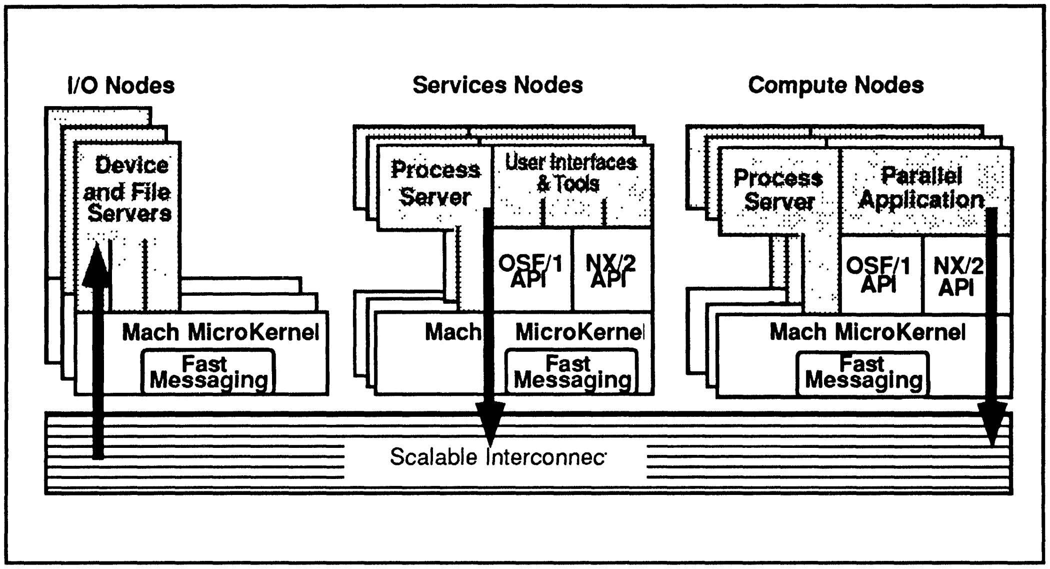
The Paragon OS architecture utilizes a small, performance-optimized operating system kernel, the *microkernel*, in every node of the system. Operating system services, such as the file system *server*, are physically located on service nodes. Each microkernel provides a transparent interface to these system servers, and in this way the full implementation of the UNIX operating system is distributed across the system. Every node sees a single "system," minimizing complexity and ensuring a familiar and industry-standard environment.

The Paragon system's modular operating system design also provides performance advantages. Because the system is modular rather than monolithic, user applications can access the Paragon interconnection network without traversing multiple layers of operating system software. The microkernel hands off message handling to the message processor — an arrangement that decreases message latency, and boosts application performance by minimizing the involvement of the operating system and the application processor in message passing. And the compactness of the microkernel means that more node memory is available to the application.



The UNIX operating system ensures familiar services and standards compliance; the use of microkernel technology delivers optimal message-passing performance without sacrificing the benefits of the UNIX environment.

As the number of nodes in a Paragon system increases, more nodes can be allocated for operating system servers. There are no inherent limits on the number of servers or nodes allocated for their use, and the allocation can be changed as the system is running, so operating system services can scale with the size of the system.



Each Paragon node has a lightweight microkernel, regardless of the role of the node in the system. Each type of node may then be configured with the appropriate Application Program Interface or servers, but retains the same microkernel and message-passing performance. In this way there is no inherent limit to the number of nodes that can be added for each of these functions as the system grows in size.

Virtual Memory

Paragon applications run in virtual memory, allowing each process to access more memory than physically exists on a single node. This simplifies the porting of applications, since no special consideration is needed when bringing large applications onto a single node. It also means that small Paragon configurations can run large applications that would otherwise be constrained by the amount of physical memory.

The Paragon VM implementation provides these benefits at little performance cost, because it avoids unnecessary page faults when an application fits within a node's physical memory.

Shared Virtual Memory

While the message-passing programming model delivers the greatest application performance, *Shared Virtual Memory* gives programmers the option of ignoring the message passing programming model and instead viewing the system with a single memory address space. Shared Virtual Memory is an advanced virtual memory management implementation that offers the advantages of a shared memory programming model.

The system automatically maps memory references, a page at a time, between the nodes so that the single memory address space is visible to all node programs. In this way, the programmer can share variables across node boundaries without explicit message passing of the variable.

The system permits all types of data references, while ensuring data consistency across all of the nodes in the system.

Dynamic System Configuration

The operating system provides the flexibility of dynamic system configurability. System administrators can add or remove subsystems, such as physical and pseudo device drivers, network protocols, file systems, and STREAMS modules and drivers, within the running kernel.

File Systems

The Paragon UNIX file system provides a path to several file system types through its Virtual File System (VFS) interface. The Paragon UNIX implementation of the VFS interface is derived from the 4.4bsd (Berkeley Software Distribution) operating system. The VFS interface enables multiple types of file systems to be used transparently.

For compatibility, Paragon UNIX applications can reach through the VFS interface to the UNIX file system, which is compatible with the 4.3BSD Tahoe release; to a System V file system; and to a Network File System-compatible distributed file system.

Paragon's UniTree client libraries provide access to UniTree file servers providing file management services, including access to remotely connected disk and tape devices and automatic migration of files through a hierarchy of I/O devices. UniTree features include transparent archival, automatic backup and restore, large file management, and shared peripherals.

Networking Support

The Paragon operating system fully supports a broad range of connectivity options. The system realizes high performance via parallel STREAMS and sockets frameworks. The industry-standard Network File System (NFS) is supported, along with optimized TCP/IP protocols and Ethernet interfaces.

The operating system also provides the X/Open Transport Interface (XTI). XTI is an emerging industry-wide application programming interface for network applications. Using XTI, developers can write network applications that are independent of the underlying transport mechanism. The XTI implementation in the system allows a path to either the Berkeley or STREAMS frameworks for compatibility with protocol families using either framework.

International Use

The internationalization and localization capabilities of the Paragon UNIX system enable developers to meet the linguistic and cultural needs of many countries without altering applications for each country. The system provides OSF/1 provisions for non-English characters, along with international date, time, and monetary formatting, numeric conventions, and the display of error messages.

Programming the Paragon™ XP/S System

The proven, verastile MIMD architecture used by the Paragon system supports all programming styles and paradigms. The application developer can make their own choice of programming model: object-oriented, Single Program Multiple Data (SPMD), Single Instruction Multiple Data (SIMD), Multiple Instruction Multiple Data (MIMD), Shared Memory, or Vector Shared Memory.

Using the MIMD programming model over the past decade has seen two distinct advantages: first, applications can easily scale to much larger systems for more performance or larger data sets (or both); and second, the programming model of the vector supercomputer isn't necessarily rejected, as it can be used at each node -- and then replicated through the system.

Several features of the Paragon system's development environment support alternative programming models, such as Shared Virtual Memory. For the MIMD programming model, the Paragon environment offers several tools specifically designed to minimize the details and overhead of parallel application development -- tools that help manage complexity and minutae of a distributed application. The Paragon tools are based on industry standards, ensuring compatibility for applications that use system libraries and a familiar environment for programmers.

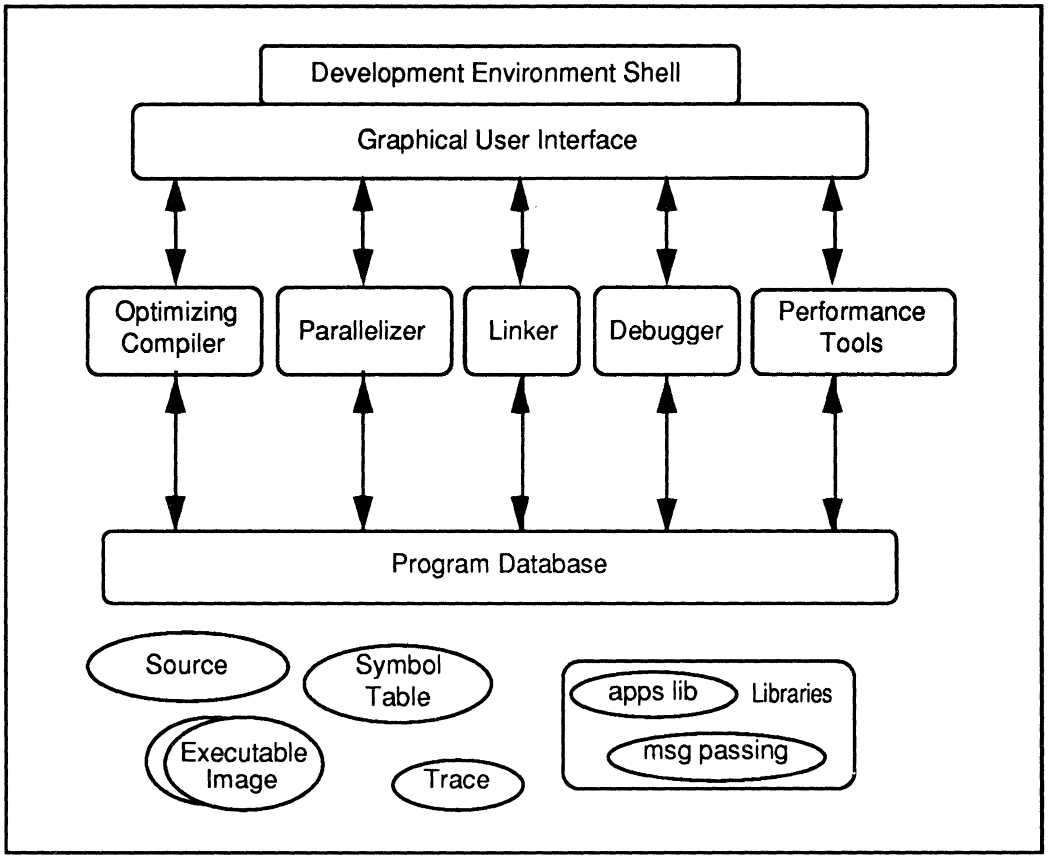
Support Throughout the Process

Moving an application from a vector supercomputer or mainframe environment to the Paragon system's parallel environment consists of porting the source code to a single Paragon node, recompiling and running the application on one node, moving the application to multiple nodes, and tuning for performance.

The Paragon system's resource-rich node architecture, in which each node is itself a powerful computer, gives developers a head start on this process. Support for virtual memory also contributes to easy porting by allowing even the largest programs to execute on a single node. An extensive set of languages, all fully conforming to U.S. and international standards, further enhances portability, as does the rich UNIX programming environment.

For the parallelization and performance tuning stages, the system's parallel CASE environment simplifies the remaining work of parallelization and fosters optimum application performance. The environment encompasses the FORGE and CAST parallelization tools, the Interactive Parallel Debugger, and both hardware- and software-based performance analysis capabilities.

The Paragon system's MIMD architecture offers the power of the MIMD programming model, as well as the flexibility of supporting virtually any other programming style, from shared vector memory through SPMD. A global messaging library provides straightforward message-passing facilities and allow programmers to ignore issues such as node placement. An application can exchange messages with other applications running on the system, as well as within the nodes of the application itself.



The integrated tool suite features a shared database of information about the application program, as well as a consistent Graphical User Interface to ensure a consistent "look and feel" among all the tools.

An Integrated, Open Environment

Paragon's integrated tool suite supports programmer productivity by providing a common "look and feel" for all development tools and eliminating the need to learn a different interface for each tool. The Paragon graphical user interface (GUI) is based on the Open Software Foundation's OSF/Motif, an industry-standard GUI that will already be familiar to many developers. The GUI gives the system a consistent interface whether the user is directly connected to the physical system or accessing it from across the country via a long-haul network.

Paragon users can choose a native or non-native development environment. Most Paragon tools can be used on the supercomputer itself or as cross-development tools on a Silicon Graphics* or Sun* workstation.

Paragon development tools are integrated via a shared database of program modules and shared data formats and definitions, to allow information transfer among different tools. The environment is open and expandable, so new tools can be integrated on an ongoing basis. A GUI style guide and a library of Motif "widgets," used for implementing the guide, are available for tools developers.

Optimizing Compilers

The Paragon system offers a broad set of program development languages:

- ANSI FORTRAN (FORTRAN-77)
- ANSI C
- C++
- Certified Ada*
- Data-parallel FORTRAN

The range of available languages allows for a full complement of programming paradigms. FORTRAN, C and Ada let programs take full advantage of both MIMD and SPMD programming models, while data-parallel FORTRAN provides a simple model for SPMD programming. And, object-oriented languages such as C++ allow for easy specification of MIMD constructs.

The system's optimizing compilers enable applications to fully exploit the superscalar architecture and large caches of the i860 XP processor and provide as much as a tenfold performance improvement over earlier versions. Supported optimizations include vectorization, procedure inlining, software pipelining, cache utilization strategies, dual instruction and dual operation modes, and global and local loop optimizations.

The Paragon FORTRAN, C and C++ languages are interoperable, allowing compiler output to be linked without regard to the source language and freeing developers to use the source language that best befits the task at hand.

Parallel CASE Tools

For the parallelization step, Paragon's FORGE and CAST parallel CASE tools offer a powerful set of interactive tools for parallel programming and significantly simplify the task of creating efficient parallel programs in FORTRAN. The tools aid in converting sequential FORTRAN programs into efficient parallel programs and in designing and implementing new parallel algorithms.

FORGE is an analytical tool for understanding and managing the content and structure of sequential FORTRAN programs. Using FORGE, programmers can "tour," then modify large FORTRAN codes. For example, programmers can anticipate the effect of modifying global data structures before actually editing and recompiling. FORGE also offers a program maintenance facility.

CAST, the Concurrent Applications Structuring Tool, builds upon FORGE and offers interactive program restructuring and parallelization. CAST uses the comprehensive analytical data generated by FORGE to recommend opportunities for MIMD-style parallelism. And, while automatic parallelizing compilers tend to extract low-level parallelism from a sequential code, CAST helps restructure code at a higher level, to produce more efficient algorithms. By providing restructuring tools and representing a program's data flow graphically, CAST makes it easy to manipulate code blocks and create more efficient structures. CAST also uses its analytical data about the new structures to ensure the application's integrity and stability.

Debugging

The Interactive Parallel Debugger (IPD) takes advantage of memory monitoring and breakpoint features of Paragon's i860 XP processors, to provide non-invasive debugging.

A source-level debug tool, IPD implements all the capabilities associated with traditional symbolic debugging, plus a variety of additions for observing, controlling, and repairing complex, multi-node application programs. Supported features include "context debugging," which allows programmers to access and control groups of processes spread across multiple nodes, and data reduction, which filters information. Breakpoints can be spread across any or all of an application's nodes, and an optimized single-step feature provides quick response during program execution. Programmers can customize IPD by command aliasing, command abbreviations, and personalized startup files.

In addition, Paragon's Ada environment includes an embedded, Ada-specific symbolic debugger.

Performance Tuning

The Performance Visualization System (iPVS™) allows programmers to accurately characterize the run-time efficiency of large-scale parallel applications and tune their applications for optimum performance. iPVS uses dedicated logic on each computational node as well as a special network to non-invasively collect data on run-time hardware performance and to provide low-overhead collection of software events.

The Paragon system offers extensive flexibility in gathering and analyzing data. Users can control data collection through compiler switches, which automatically profile and trace application performance; through environment variables, which interactively turn data gathering on and off; and through a program interface, which makes it possible to manually insert performance monitoring system calls as needed. Users can direct the performance analysis tools to selectively collect information on:

- CPU activity
- Communications activity
- I/O usage
- Procedure and loop profiling
- Message tracing
- Event tracing for both OS and user-defined events

Raw analysis data is filtered, manipulated and summarized, and results can be presented as tables, graphs, or bar charts, to enhance comprehension and support the code-tuning process.

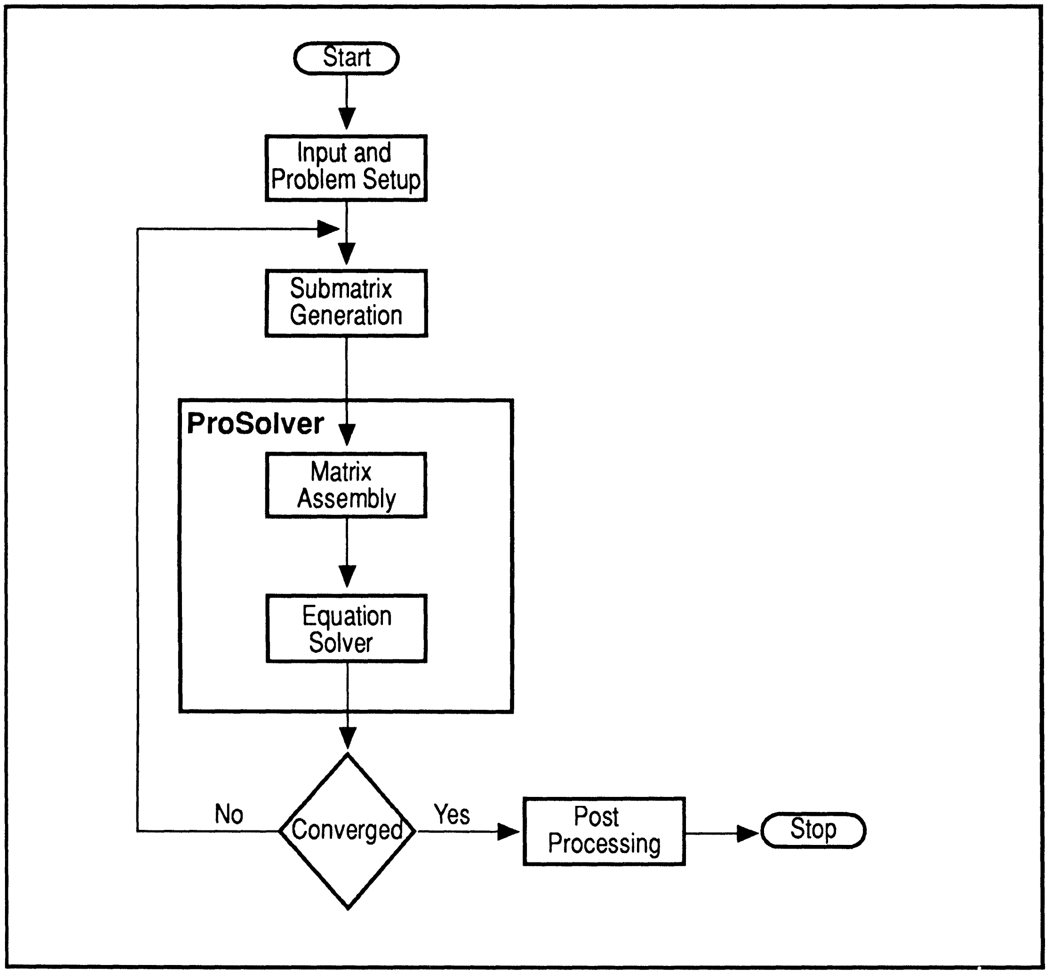
Intel's Performance Analysis Tools (iPAT) software provides an additional path to identifying computation and communication hot spots and optimizing application performance. Using iPAT, developers can monitor the time spent in individual routines, assess time spent in communication and I/O, and analyze the interactions among processors. iPAT also includes powerful, event-driven profiling.

iPSC®/860 Compatibility

Intel's 7.6 GFLOPS iPSC/860 supercomputer shares with the Paragon system the same development toolset, and the UNIX operating system is also available for the iPSC/860. Applications developed on the iPSC/860 move to the higher performance of the Paragon with no code modification, making the iPSC an ideal development platform for the Paragon system.

Paragon™ XP/S Application Tools

From electromagnetic modeling and radar signature analysis to structural mechanics and fluid dynamics, supercomputing applications have always benefitted from performance-optimized libraries. The Paragon XP/S system delivers the traditional libraries, as well as parallel versions of popular solvers -- already optimized for the MIMD architecture. In addition, applications with very large data sets are supported with a parallel file system that supports use of large files and includes run-time support for asynchronous I/O. The Intel-developed tools offer tuned performance for critical applications, while minimizing time and effort in porting sequential model applications.



Existing sequential applications, especially those of "dusty deck" Fortran, can be moved quickly to the parallel environment by utilizing the ProSolver in the heart of the application, while preserving most of the application unchanged. Applications also benefit from the performance optimizations already tested.

ProSolver™ Linear Equation Solvers

Intel's ProSolver library addresses the user's need to solve large systems of equation. It provides a comprehensive set of easy-to-use, parallel solvers for the most frequently encountered types of matrix problems. Developed and optimized expressly for Intel supercomputers, the ProSolver toolkits apply efficient solution techniques to solving dense and sparse systems of equations. The libraries handle both matrix assembly and equation solution.

Intel's ProSolver library includes:

- ProSolver-DES, for direct solution of dense matrix equations. ProSolver-DES efficiently solves in-core and out-of-core systems of equations for both real and complex data. It uses hand-tuned computational kernels, communications, and I/O routines to achieve maximum performance.
- ProSolver-SES, for direct solution of sparse systems of equations. ProSolver-SES uses skyline storage and an efficient dot-product kernel optimized for the Paragon system.
- ProSolver-IES, for iterative solution of sparse systems of equations. ProSolver-IES is a fully-sparse solver for both symmetric and asymmetric matrix problems. ProSolver-IES offers a fast, parallel version of the popular PCG-PAK iterative in-core solver. The solver includes a common interface to the matrix assembly routines used by ProSolver-SES, so programs can switch between iterative and direct techniques as needed.

Parallel Fast Fourier Transform Library

For the wide range of applications utilizing transformation of time or space domains to frequency domains (or the inverse), the Paragon system provides Fast Fourier Transform (FFT) routines. Both two-dimensional and three-dimensional versions of the algorithm are optimized to perform FFTs on a matrix of data distributed among the nodes of the Paragon system.

Node Mathematical Toolkits

Paragon also supports a number of node-level libraries, including the widely used BLAS, NAG, and SEGlib libraries.

BLAS (Basic Linear Algebra Subprograms) Routines

The industry standard for linear algebra routines, BLAS, is fully supported and optimized for the Paragon system. All three levels are supported:

- Level 1: Basic vector operations (dot product, daxpy, etc)
- Level 2: Matrix-vector products, rank-one and rank-two updates, solutions of triangular equations
- Level 3: Matrix-matrix products, rank-k updates of symmetric matrices, multiplying a matrix by a triangular matrix, solving triangular equations with multiple right-hand sides.

These routines are optimized for the i860 XP processors in the Paragon system, and are supported for all data types: single-precision, double-precision, complex, and double-precision complex.

NAG FORTRAN Library

The full Numerical Algorithm Group (NAG) FORTRAN Library routines are available for the Paragon system. These routines guarantee a consistent level of accuracy and are optimized for the i860 XP processor. Routines are supported for the double-precision and double-precison complex data types.

The Paragon system also supports all NAG FORTRAN Library routines for Mark 14. These include:

Ordinary Differential Equations
Partial Differential Equations
Numerical Differentiation
Integral Equations
Interpolation
Eigenvalue and Eigenvectors
Correlation and Regression Analysis
Multivariate Methods
Analysis of Variance
Random Number Generators
Operations Research
Sorting
Zeros of Polynomials
Roots of Transcendental Equations

SEGLib Routines

SEGLib, developed by the Society of Exploration Geophysicists, is a standard library of single-precision (real and complex) signal processing routines. The library is fully supported, including histograms, convolutions, FFTs and filter operations.

Parallel File System

The Paragon operating system provides a complete implementation of the UNIX file system, requiring no modifications to existing applications to access files. In addition, for applications requiring disk files of 500 Mbytes or larger, the Paragon system's parallel file system allows users to create and access files limited in size only by the amount of physical disk space available in the system — more than a terabyte in the largest configurations.

Some applications benefit from high-bandwidth reads and writes from the file system, and these applications can also take advantage of the Paragon system's parallel read and write requests: a single I/O request in an application can generate multiple I/O paths for reads and writes.

Scientific Visualization

Scientific visualization, which blends traditional computer graphics with sophisticated techniques of image processing and rendering, is an important element of grand challenge-level computing. By representing computational results in 2D and 3D moving images, visualization fosters increased comprehension and enables technical users to analyze a greater portion of their computational results.

Network-Transparent Visualization

Paragon provides multiple levels of support for network-transparent scientific visualization:

- Graphical representation of applications using general-purpose workstations on a network.
- Sophisticated graphical input and output capabilities using visualization workstations on a network.
- Use of the Paragon system as a visualization computing device and displaying the resulting images at high speeds via HIPPI-connected frame buffers.

Client/Server Graphics

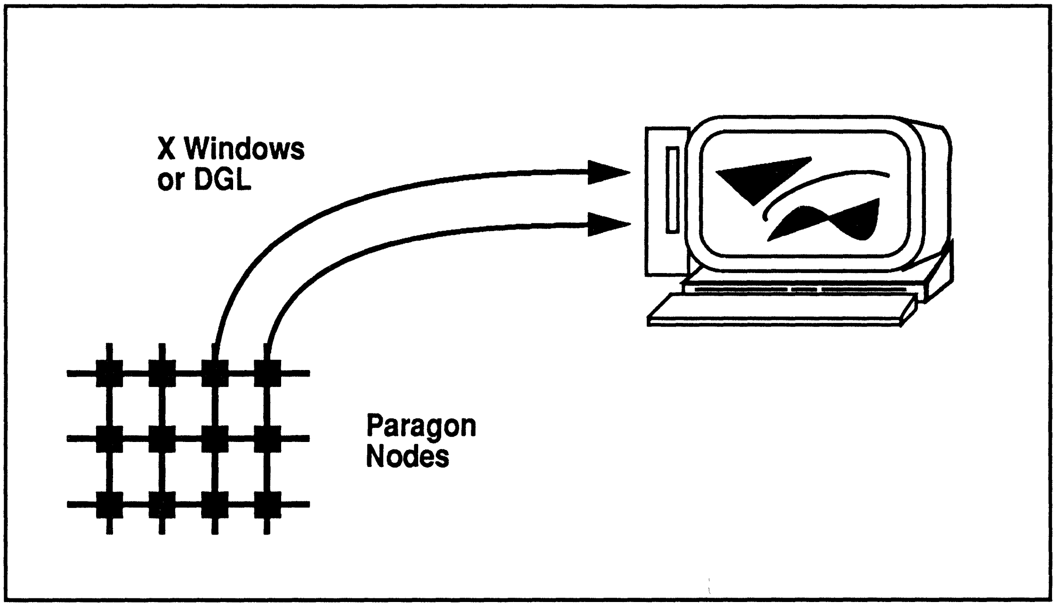
Paragon offers two standard software libraries that ensure a high degree of graphics portability and allow users to produce graphical renderings from their parallel applications: the industry-standard X Window System and the Distributed Graphics Library (DGL) developed by Silicon Graphics. By providing client libraries for X and DGL, the Paragon system enables client applications to run on the supercomputer and direct their output to a graphics workstation server.

The X Window System, originally developed at the Massachusetts Institute of Technology and now managed by an industry consortium, is the standard window management system for UNIX computing, particularly in networked environments. The Paragon system's X client implementation allows applications to display output on any number of X-compliant graphics devices, including Sun, Hewlett-Packard and DEC* workstations.

The system supports X11 Release 5, as well as PHIGS 2D and 3D graphics capability via the PEX extension to X. In addition, the industry-standard OSF/Motif GUI is on hand for developing user/application interfaces.

For higher levels of visualization performance, DGL enables user applications to take advantage of the high- performance graphics displays of Silicon Graphics and IBM* RS/6000* workstations.

The Paragon system can be configured with graphics frame buffers connected by HIPPI channels. This configuration allows the development of scientific visualization applications directly from programs executing on Paragon's computational nodes.



By utilizing the X and DGL client libraries on the compute nodes of the Paragon system, users can use the computational power of the Intel i860 XP to directly generate graphics for industry-standard display servers.

Interactive Visualizers

The software packages AVS* and Explorer* are supported for the Paragon system to allow users to display and interact with their computational results on a graphics workstation. Users can create AVS or Explorer models that run on the Paragon compute nodes, which can then be linked to the AVS or Explorer interactive visualizer on their own workstation. In this way, users can combine the benefits of visualization with the computational power of the Paragon system, without resorting to detailed or custom programming of the graphics software.

HIPPI Frame Buffers

The most complex visualization images will exceed the capability of networks such as Ethernet to carry the images from the Paragon system to graphics devices. For these highly complex visualization applications, the Paragon system can be configured with a graphics frame buffer connected by a high-bandwidth (100 Mbytes/sec) HIPPI channel. Applications running on the Paragon nodes use Paragon rendering libraries to create graphical output, which can be displayed on the frame buffer at full animation speeds of 60 frames per second.

In keeping with the inherent scalability of the Paragon design, systems can be configured with multiple I/O nodes and multiple HIPPI channels, each connecting, at the full bandwidth of the HIPPI channel, to multiple frame buffers. And, via the HIPPI channels, the frame buffers can be located either adjacent to the Paragon system, or in remote locations.

Using the Paragon™ XP/S System

Supercomputer resources are ideally shared among a mix of users, but users' computational requirements are often diverse and conflicting. The Paragon XP/S system offers an alternative to statically controlled and rigidly allocated systems, providing simultaneous batch and interactive use, with dynamic allocation of system resources.

The Paragon system supports all of the typical modes of operation:

- interactive use while developing applications
- server tasks controlled by client applications
- large applications submitted as batch jobs

All of these modes are supported interchangably with simple controls while the system is running -- there is no requirement to bring the system down for tuning or reallocation.

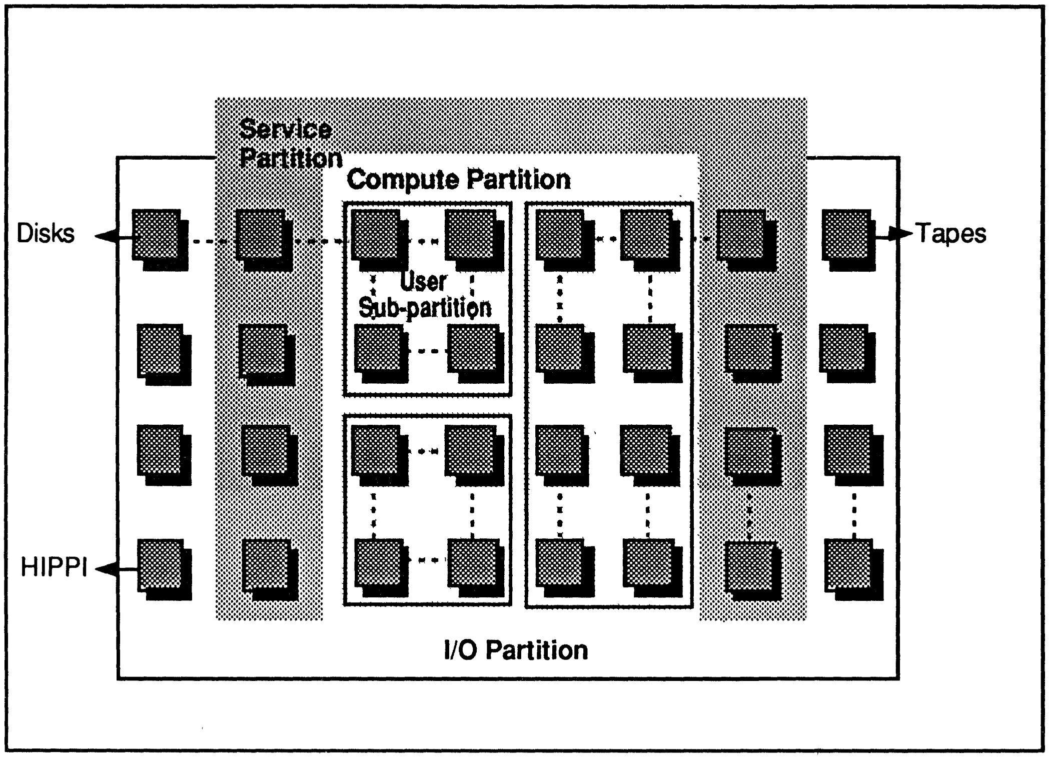
In the development mode, users can choose to remain on their own workstation for all development tasks (including compilation) and download their application to the Paragon system when they are ready to run the application. Alternatively, they can login directly into the Paragon system and develop their applications in native mode. As the system grows, all of these capabilities grow to keep the system in balance and ensure effective use of all of the system resources such as disks and network connections.

Partitions

The Paragon XP/S system supports a broad range of supercomputing applications. The nodes in the system are allocated into one of three *partitions*: the *compute partition*, where the user's parallel application is executed; the *service partition*, where operating system utilities such as shells, editors, and compilers are supported; and the *I/O partition*, where disk, tape, and network connections are supported with I/O nodes. A partition can have as few as one node allocated, or as many as all of the nodes in the Paragon system.

Compute Partition

The majority of the nodes in the system are allocated to the Compute Partition, where the user's application executes. The Compute Partion itself can be divided into subpartitions. These subpartitions can be used for batch or interactive processing -- typically several of each will be allocated. For instance, one subpartition may be established for long-running, large parallel applications. Another dedicated to smaller batch jobs with shorter time limits, and several subpartitions dedicated to interactive jobs as users develop and test applications. The size and shape of any of these subpartitions can be changed at any time to match the needs of the users.



The resources of each Paragon system are allocated differently, depending upon the user community needs and requirements. Although this example appears to be static, the resources can be reallocated throughout the computational day to meet the changing needs of researchers.

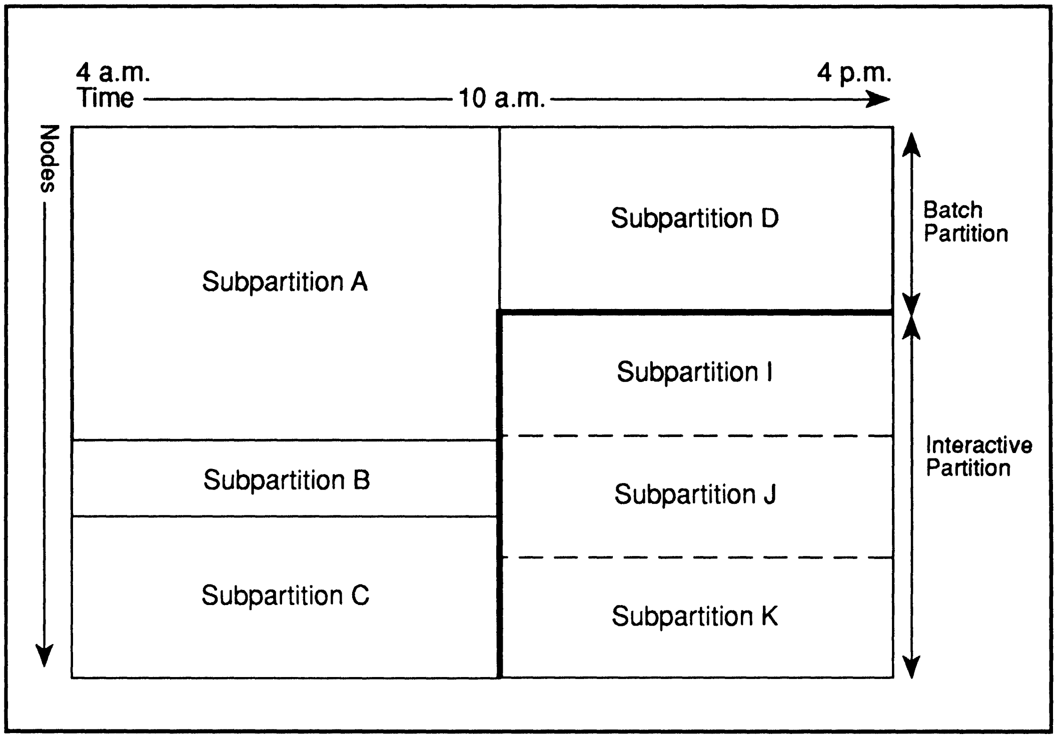
Service Partition

The Service Partition supports general user services such as editors, compilers, and UNIX shells. This partition can grow or shrink as the system is running, depending upon the number of users and the requirements for operating system utilities.

Since the physical nodes used in the Compute Partition and the Service Partition are identical, the user load can determine the allocation of nodes from the Compute Partition to the Service Partion, or in the other direction. For instance, in the middle of the workday, the system can be configured to support more operating system services and utilities with a larger Service Partition, while in the middle of the night the Compute Partition can use all of the nodes in the system, leaving the Service Partition empty.

I/O Partition

The I/O partition contains all of the nodes providing I/O capabilities. Examples of I/O nodes include SCSI nodes for service to disks and tapes, VME nodes for specialized I/O devices, and HIPPI nodes for connections to large disk arrays and graphics frame buffers. Although I/O nodes may also provide service node functionality, they are never allocated to the user's application code -- ensuring efficient I/O response and throughput. The I/O capacity and bandwidth of the system grows to meet application needs with the addition of more I/O nodes.



An example of a system job mix in the Compute Partition with its subpartions. The Compute Partition is dividing during the late night hours into three subpartitions (A,B,C), each with a different number of nodes. At 10:00am, the partition has been divided into four subpartitions, one dedicated to batch use (D) and another three subpartitions assigned to interactive use(I,J,K). The reallocation at 10:00am was made as the system was running.

Interactive Use

Although most users of the Paragon system will utilize the system as a server for large batch jobs, users can login directly and remote applications can *rpc* to the Service Partition. There is no concept of "front ends" or "system managers." Users log onto the Paragon system from their workstations or other host computer using standard UNIX tools such as *xterm, telnet,* or *rlogin.* The user runs a UNIX shell of their choice on a service node. The actual locations of these shell processes in the Service Partition are transparent to the users.

User processes, such as a compile or link, are run on available nodes within the Service Partition by the operating system. The system balances the load among the nodes in the Service Partition by dynamically migrating user processes from heavily used nodes to lightly loaded nodes.

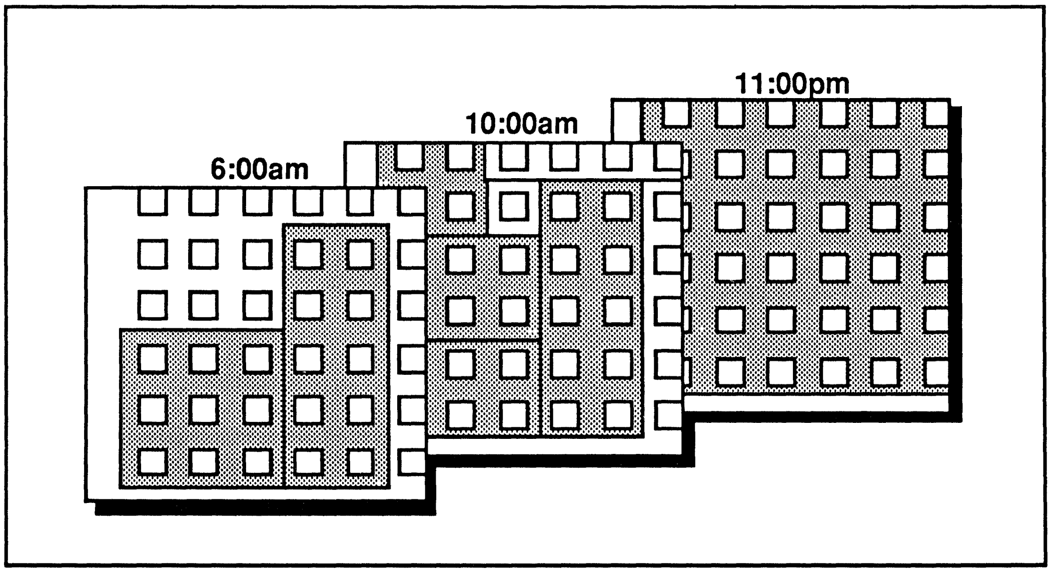
The user or application sees the system as a single entity. For example, UNIX process IDs are unique across all nodes and a user can signal a process without regard to its location. In addition, the file system is accessible from any process on any node, and a file's pathname is independent of the node that is accessing the file. Users themselves are known throughout the system — the UNIX *who* command shows all users logged into the Paragon system.

System Resource Control

Paragon utilizes the MACS (Multi-user Accounting, Control, and Scheduling) system developed by the San Diego Supercomputer Center to manage the many resources in the system for a diverse group of users. The NQS (Network Queuing System) software, originally developed at NASA Ames, is used to manage batch jobs, providing an effective means of controlling and monitoring system resources. This combination offers:

- Flexible, automated job scheduling schemes.
- Priority-based allocation and preemption.
- Reports on resource utilization and user activity.

The scheduling schemes support a wide variety of usage models, including open, "first-come, first-served" scheduling; "tennis court" style scheduling with time slots reserved for specified users; and priority-based batch queues built upon NQS services. By offering both automated process scheduling and simultaneous real-time scheduling, MACS allows the supercomputer to accommodate heavy interactive work loads, large batch jobs, and open allocation periods as needed throughout the computing day.



In this example, MACS has scheduled two jobs in the Compute Partition at 6:00am. Due to the hour, few of the nodes are in use. At 10:00am, MACS uses all of the available nodes in the Compute Partition for a number of previously scheduled jobs. At 11:00pm the same evening, MACS consumes all of the nodes in the Compute Partition with a single large job that was previously scheduled.

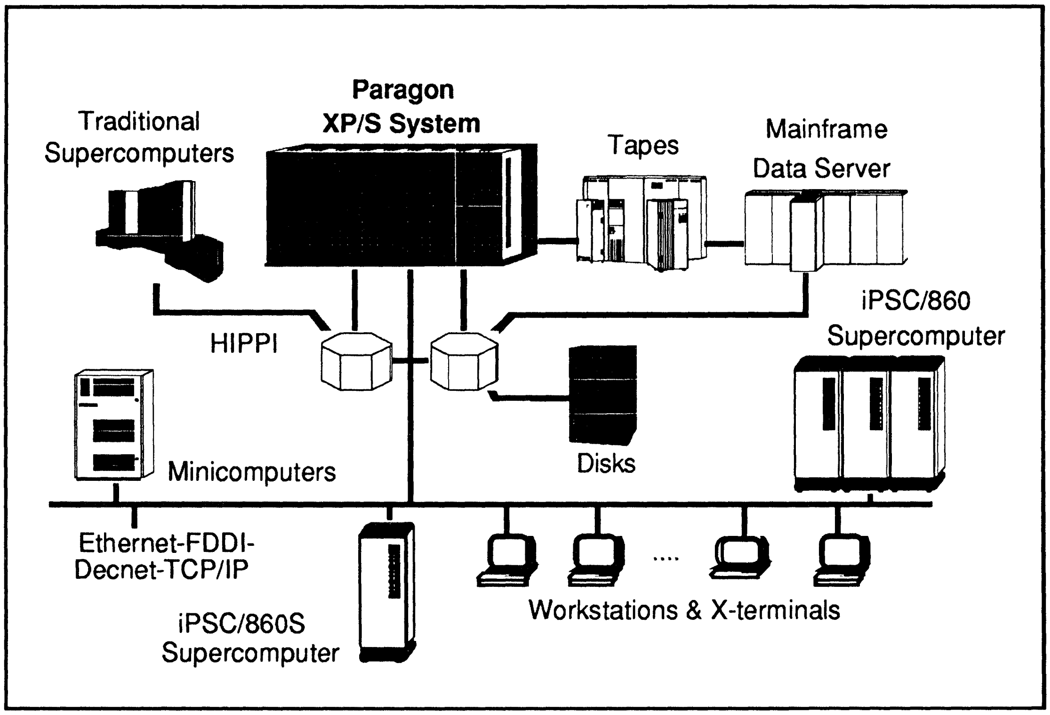
Batch Scheduling

The scheduler allows jobs to be executed directly from NQS queues. It also allows specification of number of nodes to be employed, queue priorities and sizes, and the number of jobs permitted to wait in each queue. To aid system administrators in planning system use by a diverse mix of users, MACS implements a "shape-based" scheduling scheme using a hierarchy of criteria.

Regardless of how a job is scheduled, MACS can monitor and account for user node allocation, reporting the number and duration of nodes held for each user in the form of concise reports or an interactive monitor. This gives the data processing center a way to assess resource use with quantitative data.

Network Connectivity

Whether in the industrial production environment or the national research laboratory, the network increasingly is the backbone of the computing environment. The Paragon system anticipates its role as a server among nation-wide networks of clients by supporting client/server access to the system's parallel resources. The system provides both the hardware and software needed for reliable, high-bandwidth integration into a wide range of networked computing environments, and, as with other elements of the system, network connectivity is scalable.



The system requires no host or other "system manager", and combined with the connectivity to Ethernet, FDDI, and HIPPI (at the full bandwidth of each), the Paragon system is a full-fledged member of the network.

Scalable Connectivity

To facilitate integration into diverse computing environments, the Paragon system offers industry-standard HIPPI and Ethernet networking hardware and protocols, as well as providing DECnet connectivity to Digital Equipment Corporation computers via the Ethernet. There is no inherent limit to the number of networking connections, which enables the system's networking resources to increase along with the number of nodes in the system or the number of users accessing the system.

HIPPI provides fast point-to-point communications with other supercomputers, workstations, mainframes and peripherals. The Paragon HIPPI connection matches the industry standard of 100 Mbytes/sec and ties into the Paragon interconnection network, which itself has a bandwidth of 200 Mbytes/sec.

Ethernet is the most common LAN for sub-nets within a facility or laboratory. The Paragon system supports the full IEEE 802.1 protocol, and multiple Ethernet interfaces can be used to increase the Ethernet bandwidth for remote logins and interactive user sessions.

The system's FDDI networking supports the ANSI X3T9.5 protocol standard. Both Ethernet and FDDI are implemented via the VME interface.

Network Protocols

For Ethernet networking, the system supports the TCP/IP protocols; the UNIX sockets mechanism; basic networking facilities such as FTP, telnet and remote login; and application-level protocols such as the Network File System (NFS) and the UNIX facilities for *rsh* and *rpc*. The system can also support the DECnet protocol for integration into DEC computing environments.

System Reliability

In the demanding environment of the supercomputer center, system availability is as essential as system performance. The Paragon™ XP/S system has a wide range of features that ensure maximum system availability and reliability.

Reliable Hardware

Hardware reliability starts with the system's extensive use of VLSI, including Intel's proven i860 family of RISC processors, which have been chosen for more than 100 commercial designs. The use of an industry-standard CPU ensures that the reliability of the VLSI has been proven in a variety of applications.

Error-correcting memory on each node helps identify and correct memory errors if they occur. In addition, the communications mesh is designed with its own error-detection facilities, and the PMRC mesh router chips can continue to pass messages even in the unlikely event of a node failure. Critical aspects of the system hardware, including power supplies and disk subcontrollers, have redundant components. Disk rebuilding can be performed on-line or off-line, further contributing to high system availability.

Paragon systems require no special cooling or other protection beyond that provided by a standard computer-room environment. Eliminating the need for special cooling equipment also eliminates any of the potential problems of complex cooling apparatus traditionally required of supercomputer systems. It also ensures that the Paragon system can be installed quickly and immediately after shipment.

In addition, disk failures can be resolved without shutting down the system or the I/O system, by simply removing and replacing the faulty disk while the system is running.

Reliable Software

As with the commercially available VLSI components, Paragon operating system software starts from a robust, commercially proven foundation — the Open Software Foundation's OSF/1 UNIX implementation. The system also includes software reliability features such as "fire walls," which separate user code from system code and prevent inadvertent corruption of system software.

Many of the system's development tools, libraries, and applications originated on workstations and on previous supercomputers, and reflect the reliability and improvements of several years of intensive use.

Software is rigorously tested prior to release, including full regression testing, comparison against previously reported software bugs, and user testing at the hands of Intel's most experienced and demanding users.

Reliable Operation

To keep the system functioning smoothly, an internal diagnostic network monitors important system components and operation, and easy-to-use diagnostic utilities drive the diagnostic subsystem to quickly isolate the failure to a field-replaceable unit. Units can be replaced quickly, and the operating system is designed for a quick restart.

For example, the system monitors itself for high temperatures and automatically (and gracefully) shuts down the system if they are detected. In addition to enabling users to protect their data, this capability minimizes the number of components harmed by excessive temperatures and contributes to a quick restoration to full capacity once the problem is diagnosed and corrected.

Development With Intel

To ensure a smooth transition from traditional supercomputers to the Paragon series of supercomputers, Intel offers comprehensive technical support. Among the services Intel offers are:

- Applications consulting with the scientists and mathematicians of Intel's computational sciences group.
- Systems specification and systems integration support, to assist in configuring the Paragon system to meet specific site requirements and integrating it into a given computing environment.
- Porting services, including full-time, on-site personnel to assist with applications development and porting.
- Benchmarking assistance, to help evaluate Paragon's performance and see how it applies to targeted applications.
- Training courses for programmers and system administrators, covering topics such as parallel programming, performance analysis, system software and hardware design concepts, and fault isolation.

Worldwide Users

Paragon XP/S users also benefit from the growing number of third-party applications developed for Intel supercomputers, and from membership in a large and growing user community. With more than 300 installations worldwide, Intel has the largest installed base of massively parallel processor systems.

User Group

Customers also gain from the resources of the Intel Supercomputer Users' Group. With more than 1,000 participants, the Users' Group maintains an active calendar of user conferences, technical sessions, special interest group meetings, publications, and other events.

intel®

Intel Corporation
Supercomputer Systems Division
15201 N.W. Greenbrier Parkway
Beaverton, OR 97006
(503) 629-7600

Intel Corporation (UK) Ltd.
Supercomputer Systems Division
Pipers Way
Swindon SN3 1RJ
England
(+44) -793-696578

Intel Corporation K.K.
Supercomputer Systems Division
5-6 Tokodai, Tsukuba City
Ibaraki-Ken 300-26
Japan
(+81) -29847-8511

Intel Semiconductor GmbH
Dornacher Strasse 1
8016 Feldkirchen bel Muenchen
Germany
(+49) -89-90-992-415

Intel Corporation
1 Rue Edison-BP303
78054 St. Quentin-en-Yvelines Cedex
France
(+331) -30-57-7182