Intel Paragon XP/S Overview

- Distributed-memory MIMD multicomputer
- 2D array of nodes, performing both OS functionality as well as user computation
  - Main memory physically distributed among nodes (16-64 MB / node)
  - Each node contains two Intel i860 XP processors: application processor for user's program, message processor for inter-node communication
- Balanced design: speed and memory capacity matched to interconnection network, storage facilities, etc.
  - Interconnect bandwidth scales with number of nodes
  - Efficient even with thousands of processors

Intel MP Paragon XP/S 150 @ Oak Ridge National Labs

Paragon XP/S Nodes

- Network Interface Controller (NIC)
  - Connects node to its PMRC
  - Parity-checked, full-duplexed router with error checking
- Message processor
  - Intel i860 XP processor
  - Handles all details of sending / receiving a message between nodes, including protocols, packetization, etc.
  - Supports global operations including broadcast, synchronization, sum, min, and, or, etc.
- Application processor
  - Intel i860 XP processor (42 MIPS, 50 MHz clock) to execute user programs
- 16–64 MB of memory

Paragon XP/S Node Interconnection

- 2D mesh chosen after extensive analytical studies and simulation
- Paragon Mesh Routing Chip (PMRC) / iMRC routes traffic in the mesh
  - 0.75 µm, triple-metal CMOS
  - Routes traffic in four directions and to and from attached node at > 200 MB/s
    - 40 ns to make routing decisions and close appropriate switches
    - Transfers are parity checked, router is pipelined, routing is deadlock-free
  - Backplane is active backplane of router chips rather than mass of cables

Paragon XP/S Usage

- OS is based on UNIX, provides distributed system services and full UNIX to every node
  - System is divided into partitions, some for I/O, some for system services, rest for user applications
- Applications can run on arbitrary number of nodes without change
  - Run on larger number of nodes to process larger data sets or to achieve required performance
- Users have client/server access, can submit jobs over a network, or login directly to any node
  - Comprehensive resource control utilities for allocating, tracking, and controlling system resources

Paragon XP/S Programming

- MIMD architecture, but supports various programming models: SPMD, SIMD, MIMD, shared memory, vector shared memory
- CASE tools including:
  - Optimizing compilers for FORTRAN, C, C++, Ada, and Data Parallel FORTRAN
  - Interactive Debugger
  - Parallelization tools: FORGE, CAST
  - Intel's ProSolver library of equation solvers
  - Intel's Performance Visualization System (PVS)
  - Performance Analysis Tools (PAT)

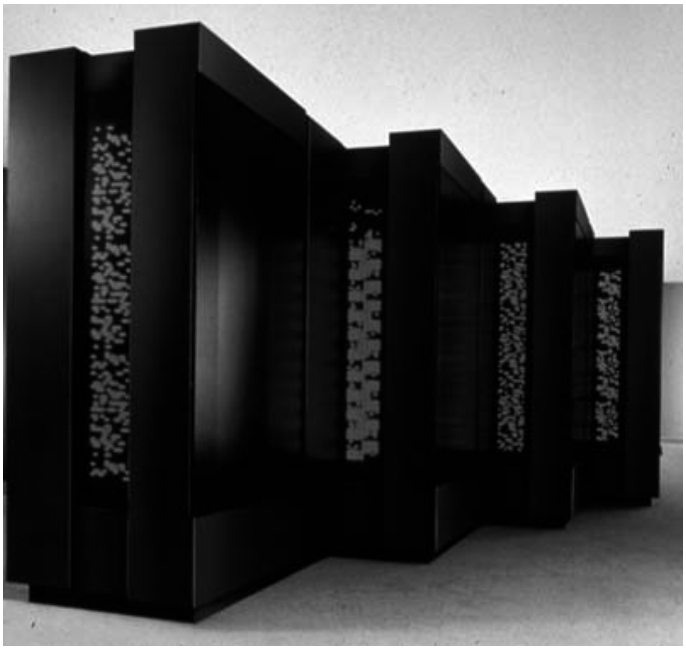
Thinking Machines CM-5 Overview

- Distributed-memory MIMD multicomputer
  - SIMD or MIMD operation
- Processing nodes are supervised by a control processor, which runs UNIX
  - Control processor broadcasts blocks of instructions to the processing nodes, and initiates execution
  - SIMD operation: nodes are closely synchronized, blocks broadcast as needed
  - MIMD operation: nodes fetch instructions independently and synchronize only as required by the algorithm
- Nodes may be divided into partitions
  - One control processor, called the partition manager, per partition
  - Partitions may exchange data

Thinking Machines CM-5 Overview (cont.)

- Other control processors, called I/O Control Processors, manage the system's I/O devices
  - Scale to achieve necessary I/O capacity
  - DataVaults to provide storage
- Control processors in general
  - Scheduling user tasks, allocating resources, servicing I/O requests, accounting, security, etc.
  - May execute some code
  - No arithmetic accelerators, but additional I/O connections
  - In small system, one control processor may play a number of roles
  - In large system, control processors are often dedicated to particular tasks (partition manager, I/O cont. proc., etc.)

Thinking Machines CM-5@ NCSA



CM-5 Nodes

- Processing nodes
  - SPARC CPU
    - 22 MIPS
  - 8-32 MB of memory
  - Network interface
  - (Optional) 4 pipelined vector processing units
    - Each can perform up to 32 million double-precision floating-point operations per second
    - Including divide and square root
- Fully configured CM-5 would have
  - 16,384 processing nodes
  - 512 GB of memory
  - Theoretical peak performance of 2 teraflops

CM-5 Networks

- Control Network
  - Tightly coupled communication services
  - Optimized for fast response, low latency
  - Functions: synchronizing processing nodes, broadcasts, reductions, parallel prefix operations
- Data Network
  - 4-ary hypertree, optimized for high bandwidth
  - Functions: point-to-point commn. for tens of thousands of items simultaneously
  - Responsible for eventual delivery of messages accepted
  - Network Interface connects nodes or control processors to the Control or Data Network (memory-mapped control unit)

Tree Networks (Reference Material)

- Binary Tree
  - 2^k–1 nodes arranged into complete binary tree of depth k–1
  - Diameter is 2(k–1)
  - Bisection width is 1
- Hypertree
  - Low diameter of a binary tree plus improved bisection width
  - Hypertree of degree k and depth d
    - From “front”, looks like k-ary tree of height d
    - From “side”, looks like upside-down binary tree of height d
    - Join both views to get complete network
  - 4-ary hypertree of depth d
    - 4^d leaves and $2^d(2^{d+1}–1)$ nodes
    - Diameter is 2d
    - Bisection width is 2^{d+1}

CM-5 Usage

- Runs Cmost, enhanced vers. of SunOS
- User task sees a control processor acting as a Partition Manager (PM), a set of processing nodes, and inter-processor communication facilities
  - User task is a standard UNIX process running on the PM, and one on each of the processing nodes
  - The CPU scheduler schedules the user task on all processors simultaneously
- User tasks can read and write directly to the Control Network and Data Network
  - Control Network has hardware for broadcast, reduction, parallel prefix operations, barrier synchronization
  - Data Network provides reliable, deadlock-free point-to-point communication

IBM SP2 Overview

- Distributed-memory MIMD multicomputer
- Scalable POWERparallel 1 (SP1)
  - Development started February 1992, delivered to users in April 1993
- Scalable POWERparallel 2 (SP2)
  - 120-node systems delivered 1994
    - 4–128 nodes: RS/6000 workstation with POWER2 processor, 66.7 MHz, 267 MFLOPS
    - POWER2 used in RS 6000 workstations, gives compatibility with existing software
  - 1997 version (NUMA):
    - High Node (SMP node, 16 nodes max): 2–8 PowerPC 604, 112 MHz, 224 MFLOPS, 64MB–2GB memory
    - Wide Node (128 nodes max): 1 P2SC (POWER2 Super Chip, 8 chips on one chip), 135 MHz, 640 MFLOPS, 64MB–2GB memory

IBM SP2 @ Oak Ridge National Labs

IBM SP2 Overview (cont.)

- RS/6000 as system console
- SP2 runs various combinations of serial, parallel, interactive, and batch jobs
  - Partition between types can be changed
  - High nodes — interactive nodes for code development and job submission
  - Thin nodes — compute nodes
  - Wide nodes — configured as servers, with extra memory, storage devices, etc.
- A system “frame” contains 16 thin processor or 8 wide processor nodes
  - Includes redundant power supplies, nodes are hot swappable within frame
  - Includes a high-performance switch for low-latency, high-bandwidth communication

IBM SP2 Processors

- POWER2 processor
  - Various versions from 20 to 62.5 MHz
  - RISC processor, load-store architecture
    - Floating point multiple & add instruction with latency of 2 cycles, pipelined for initiation of new one each cycle
    - Conditional branch to decrement and test a "count register" (without fixed-point unit involvement), good for loop closings
- POWER 2 processor chip set
  - 8 semi-custom chips: Instruction Cache Unit, four Data Cache Units, Fixed-Point Unit (FXU), Floating-Point Unit (FPU), and Storage Control Unit
    - 2 execution units per FXU and FPU
    - Can execute 6 instructions per cycle: 2 FXU, 2 FPU, branch, condition register
    - Options: 4-word memory bus with 128 KB data cache, or 8-word with 256 KB

IBM SP2 Interconnection Network

- General
  - Multistage High Performance Switch (HPS) network, with extra stages added to keep bw to each processor constant
  - Message delivery
    - PIO for short messages with low latency and minimal message overhead
    - DMA for long messages
  - Multi-user support — hardware protection between partitions and users, guaranteed fairness of message delivery
- Routing
  - Packet switched = each packet may take a different route
  - Cut-through = if output is free, starts sending without buffering first
  - Wormhole routing = buffer on subpacket basis if buffering is necessary

IBM SP2 AIX Parallel Environment

- Parallel Operating Environment — based on AIX, includes Desktop interface
  - Partition Manager to allocate nodes, copy tasks to nodes, invoke tasks, etc.
  - Program Marker Array — (online) squares graphically represent program tasks
  - System Status Array — (offline) squares show percent of CPU utilization
- Parallel Message Passing Library
- Visualization Tool — view online and offline performance
  - Group of displays for communications characteristics or performance (connectivity graph, inter-processor communication, message status, etc.)
- Parallel Debugger

nCUBE Overview

- Distributed-memory MIMD multicomputer (with hardware to make it look like shared-memory multiprocessor)
- History
  - nCUBE 1 — 1985
  - nCUBE 2 — 1989
    - 34 GFLOPS, scalable
    - ?–8192 processors
  - nCUBE 3 — 1995
    - 1–6.5 TFLOPS, 65 TB memory, 24 TB/s hypercube interconnect, 1024 3 GB/s I/O channels, scalable
    - 8–65,536 processors
- Operation
  - Can be partitioned into "subcubes"
  - Programming paradigms: SPMD, inter-subcube processing, client/server

nCUBE 3 Processor

- 0.6 µm, 3-layer CMOS, 2.7 million transistors, 50 MHz, 16 KB data cache, 16 KB instruction cache, 100 MFLOPS
  - Argument against off-the-shelf processor: shared memory, vector floating-point units, aggressive caches are necessary in workstation market but superfluous here
- ALU, FPU, virtual memory management unit, caches, SDRAM controller, 18-port message router, and 16 DMA channels
  - ALU for integer operations, FPU for floating point operations, both 64 bit
    - Most integer operations execute in one 20ns clock cycle
    - FPU can complete two single- or double-precision operations in one clock cycle
  - Virtual memory pages can be marked as "non-resident", the system will generate messages to transfer page to local node

nCUBE 3 Interconnect

- Hypercube interconnect
  - Added hypercube dimension allows for double the processors, but processors can be added in increments of 8
  - Wormhole routing + adaptive routing around blocked or faulty nodes
- ParaChannel I/O array
  - Separate network of nCUBE processors for load distribution and I/O sharing
  - 8 computational nodes (nCUBE processors plus local memory) connect directly to one ParaChannel node, and can also communicate with those nodes via the regular hypercube network
  - ParaChannel nodes can connect to RAID mass storage, SCSI disks, etc.
    - One I/O array can be connected to more than 400 disks

nCUBE 3 Software

- Parallel Software Environment
  - nCX microkernel OS — runs on all compute nodes and I/O nodes
  - UNIX functionality
  - Programming languages including FORTRAN 90, C, C++, as well as HPF, Parallel Prolog, and Data Parallel C

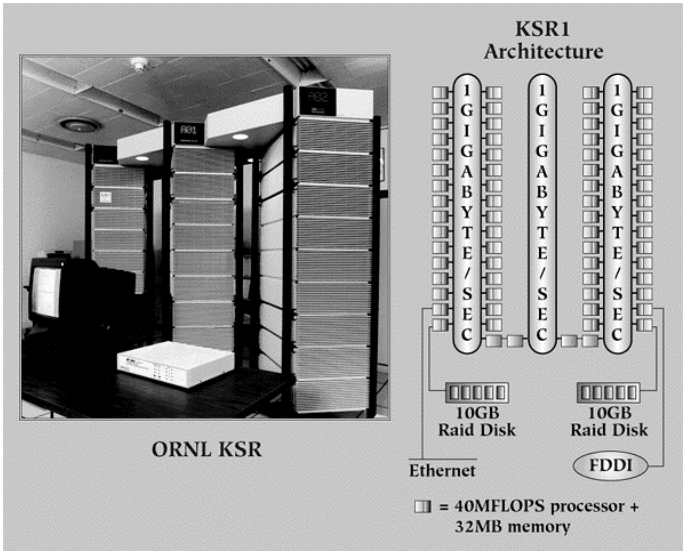
MediaCUBE Overview

- Emphasized on their web page; for delivery of interactive video to client devices over a network (from LAN-based training to video-on-demand to homes)
  - MediaCUBE 30 = 270 1.5 Mbps data streams, 750 hours of content
  - MediaCUBE 3000 = 20,000 & 55,000

Kendall Square Research KSR1 Overview

- COMA distributed-memory MIMD multicomputer (with hardware to make it look like shared-memory multiprocessor)
- 6 years in development, 36 variations in 1992 (8 cells for $500k, 1088 for $30m)
  - 8 cells: 320 MFLOPS, 256 MB memory, 210 GB disk, 210 MB/s I/O
  - 1088 cells: 43 GFLOPS, 34 GB memory, 15 TB disk, 15 GB/s I/O
- Each system includes:
  - Processing Modules, each containing up to 32 APRD Cells including 1GB of ALLCACHE memory
  - Disk Modules, each containing 10 GB
  - I/O adapters
  - Power Modules, with battery backup

KSR1 @ Oak Ridge National Labs



Kendall Square Research KSR1 Processor Cells

- Each APRD (ALLCACHE Processor, Router, and Directory) Cell contains:
  - 64-bit Floating Point Unit, 64-bit Integer Processing Unit
  - Cell Execution Unit for address gen.
  - 4 Cell Interconnection Units, External I/O Unit
  - 4 Cache Control Units
  - 32 MB of Local Cache, 512 KB of subcache
- Custom 64-bit processor: 1.2 µm, each up to 450,000 transistors, packaged in 8x13x1 printed circuit board
  - 20 MHz clock
  - Can execute 2 instructions per cycle

Kendall Square Research KSR1 ALLCACHE System

- The ALLCACHE system moves an address set requested by a processor to the Local Cache on that processor
  - Provides the illusion of a single sequentially-consistent shared memory
- Memory space consists of all the 32 KB local caches
  - No permanent location for an “address”
  - Addresses are distributed and based on processor need and usage patterns
  - Each processor is attached to a Search Engine, which finds addresses and their contents and moves them to the local cache, while maintaining cache coherence throughout the system
    - 2 levels of search groups for scalability

Kendall Square Research KSR1 Programming Environment

- KSR OS = enhanced OSF/1 UNIX
  - Scalable, supports multiple computing modes including batch, interactive, OLTP, and database management and inquiry
- Programming languages
  - FORTRAN with automatic parallelization
  - C
  - PRESTO parallel runtime system that dynamically adjusts to number of available processors and size of the current problem

Cray T3D Overview

- NUMA shared-memory MIMD multiprocessor
- DEC Alpha 21064 processors arranged into a virtual 3D torus (hence the name)
  - 32–2048 processors, 512MB–128GB of memory, 4.2–300 GFLOPS
  - Parallel vector processor (Cray Y-MP or C90) acts as a host computer, and runs the scalar / vector parts of the program
  - Each processor has a local memory, but the memory is globally addressable
  - 3D torus is virtual — the physical layout includes redundant nodes that can be mapped into the torus if a node fails
- System contains:
  - Processing element nodes, Interconnect network, I/O gateways

Cray T3D

T3D Processing Element Nodes

- Node contains 2 PEs; each PE contains:
  - DEC Alpha 21064 microprocessor
    - 150 MHz, 64 bits, double issue, 8/8 KB L1 caches
    - Support for L2 cache, eliminated in favor of improving latency to main memory
  - 16–64 MB of local DRAM
    - Access local memory: latency 87–253ns
    - Access remote memory: 1–2μs (~8x)
  - Alpha has 43 bits of virtual address space, only 32 bits for physical address space — external registers in node provide 5 more bits for 37 bit phys. addr.
- Node also contains:
  - Network interface
  - Block transfer engine (BLT) — asynchronously distributes data between PE memories and rest of system (overlap computation and communication)

T3D Interconnection Network and I/O Gateways

- Interconnection network
  - Between PE nodes and I/O gateways
  - 3D torus between routers, each router connecting to a PE node or I/O gateway
  - Dimension-order routing: when a message leaves a node, it first travels in the X dimension, then Y, then Z
- I/O gateways
  - Between host and T3D, or between T3D and an I/O cluster (workstations, tape drives, disks, and / or networks)
  - Hide latency:
    - Alpha has a FETCH instruction that can initiate a memory prefetch
    - Remote stores are buffered — 4 words accumulate before store is performed
    - BLT redistributes data asynchronously

Cray T3D Usage

- Processors can be divided into partitions
  - System administrator can define a set of processors as a pool, specifying batch use, interactive use, or both
  - User can request a specific number of processors from a pool for an application, MAX selects that number of processors and organizes them into a partition
- A Cray Y-MP C90 multiprocessor is used as a UNIX server together with the T3D
  - OS is MAX, Massively Parallel UNIX
  - All I/O is attached to the Y-MP, and is available to T3D
  - Shared file system between Y-MP & T3D
  - Some applications run on the Y-MP, others on the T3D, some on both

Cray T3D Programming

- Programming models
  - Message passing based on PVM
  - High-Performance FORTRAN — implicit remote accessing
  - MPP FORTRAN — implicit and explicit communication
- OS is UNICOS MAX, a superset of UNIX
  - Distributed OS with functionality divided between host and PE nodes (microkernels only on PE nodes)

Cray T3E Overview

- T3D = 1993,
  T3E = 1995 successor (300 MHz, $1M),
  T3E-900 = 1996 model (450 MHz, $.5M)
- T3E system = 6–2048 processors,
  3.6–1228 GFLOPS,1–4096 GB memory
  - PE = DEC Alpha 21164 processor (300 MHz, 600 MFLOPS, quad issue), local memory, control chip, router chip
    - L2 cache is on-chip so can't be eliminated, but off-chip L3 can and is
    - 512 external registers per process
  - GigaRing Channel attached to each node and to I/O devices and other networks
  - T3E-900 = same w/ faster processors, up to 1843 GFLOPS
- Ohio Supercomputer Center (OSC) has a T3E with 128 PEs (300 MHz), 76.8 GFLOPS, 128 MB memory / PE

Convex Exemplar SPP-1000 Overview

- ccNUMA shared-memory MIMD
  - 4–128 HP PA 7100 RISC processors, up to 25 GFLOPS
  - 256 MB – 32 GB memory
  - Logical view: processors – crossbar – shared memory – crossbar – peripherals
- System made up of "hypernodes", each of which contains 8 processors and 4 cache memories (each 64–512MB) connected by a crossbar switch
  - Hypernodes connected in a ring via Coherent Toroidal Interconnect, an implementation of IEEE 1596-1992, Scalable Coherency Interface
    - Hardware support for remote memory access
    - Keeps the caches at each procesor consistent with each other

Cray Exemplar SPP-1000

Convex Exemplar SPP-1000 Processors

- HP PA7100 RISC processor
  - 555,000 transistors, 0.8µ
- 64 bit wide, external 1MB data cache & 1MB instruction cache
  - Reads take 1 cycle, writes take 2
- Can execute one integer and one floating-point instruction per cycle
- Floating Point Unit can multiply, divide / square root, etc. as well as multiply-add and multiply-subtract
  - Most fp operations take two cycles, divide and square root take 8 for single precision and 15 for double precision
- Supports multiple threads, and hardware semaphore & synchronization operations

Silicon Graphics POWER CHALLENGEarray Overview

- ccNUMA shared-memory MIMD
- "Small" supercomputers
  - POWER CHALLENGE — up to 144 MIPS R8000 processors or 288 MISP R1000 processors, with up to 109 GFLOPS, 128 GB memory, and 28 TB of disk
  - POWERnode system — shared-memory multiprocessor of up to 18 MIPS R8000 processors or 36 MIPS R1000 processors, with up to 16 GB of memory
- POWER CHALLENGEarray consists of up to 8 POWER CHALLENGE or POWERnode systems
  - Programs that fit within a POWERnode can use the shared-memory model
  - Larger program can span POWERnodes

Silicon Graphics POWER CHALLENGEarray Programming

- Fine- to medium-grained parallelism
  - Shared-memory techniques within a POWERnode, using parallelizing FORTRAN and C compilers
- Medium- to coarse-grained parallelism
  - Shared-memory within a POWERnode or message-passing between POWERnode
  - Applications based on message-passing will run within a POWERnode, and libraries such as MPI or PVM will use the shared-memory instead
- Large applications
  - Hierarchical programming, using a combination of the two techniques

Silicon Graphics Origin 2000 Overview

- ccNUMA shared-memory MIMD
- Various models, 2–128 MIPS R10000 processors, 16 GB – 1 TB memory, 6–200 GB/s peak I/O bandwidth
  - Largest model also supports 96 PCI cards and 160 Ultra SCSI devices
  - Processing node board contains two R10000 processors, part of the shared memory, directory for cache coherence, node interface, and I/O interface
- ccNUMA (SGI says they supply 95% of ccNUMA systems worldwide)
  - Crossbar switches that scale upwards
- Packaged solutions for business (file serving, data mining), Internet (media serving), & high-performance computing

SGI Origin 2000 @ Boston University