



HPTC

What it is,
How to get it,
Why you need it

Russell Doty
HPTC Segment Manager
Compaq Computer Corporation

COMPAQ

Better answers

www.compaq.com



HPTC Definitions

- ◆ What is HPTC?
 - A set of computing technologies for very fast numerical simulation, modeling and data processing, enabling new insights to extremely complex problems
- ◆ Concepts:
 - Capability
 - Run a single huge job in acceptable turnaround time
 - Capacity
 - Run multiple jobs in high throughput mode



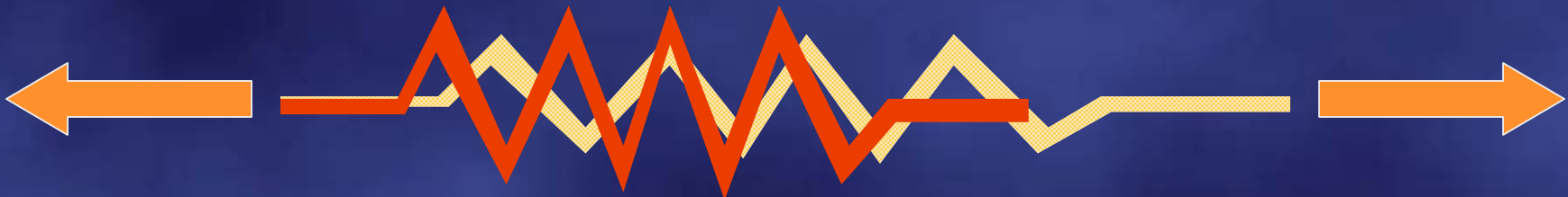
So? I 'll wait a little...

- ◆ You haven't got the time
 - You have to run as fast as you can to stay competitive
- ◆ You may not be young enough
 - Some simulations require multiple life spans...
- ◆ You need more detail and accuracy
 - To drive innovation and competitive products



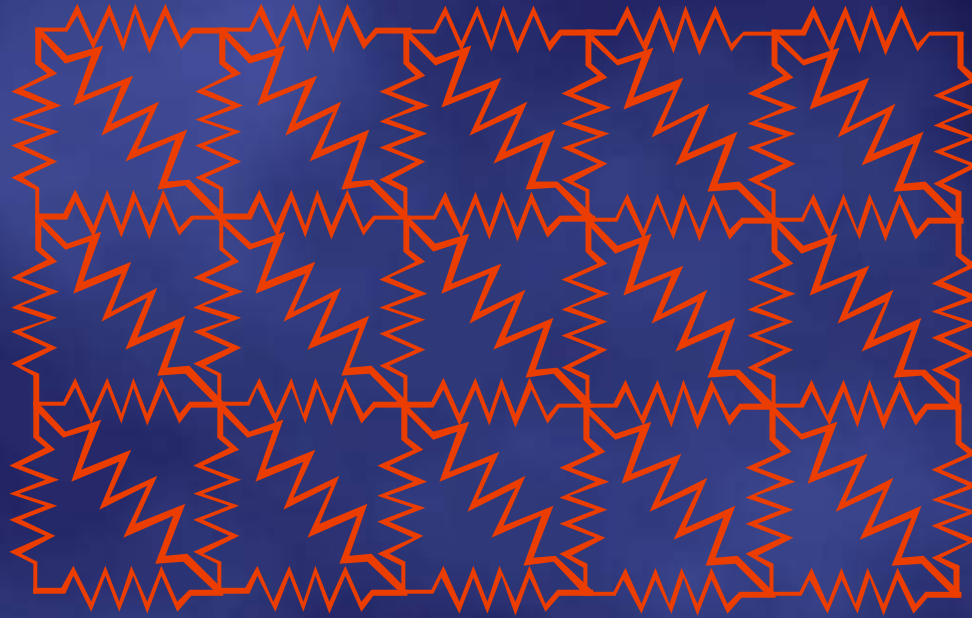
What *is* simulation?

- ◆ Spring Equation: $d=f/k$ (or, $f=d*k$)
 - d = displacement
 - k = spring constant
 - f = applied force



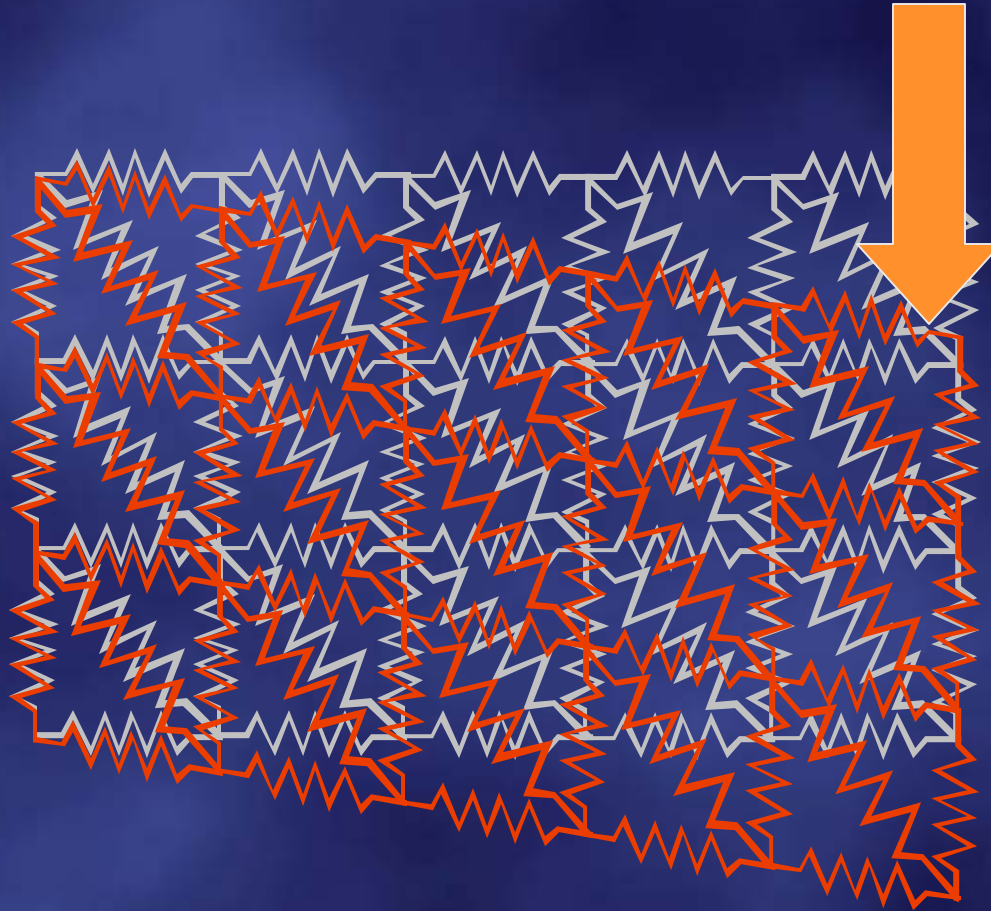


Model an Object as *Spring Network*





Apply spring calculations to each spring





Simulation

- ◆ Finite Element Analysis -- “points connected by springs”
- ◆ Other phenomena calculated in similar fashion
- ◆ Conceptual foundation of simulation

- ◆ Develop mathematical model
- ◆ Solve equations
- ◆ Typically floating point computation



Floating Point Calculation

- ◆ Multiply "A" times "B" and save as "C"

Load	"A"	R1
Load	"B"	R2
Multiply	R1, R2	R3
Store	R3	"C"



Vector Operation

- ◆ Vector: 1 dimensional list of values
- ◆ Multiply 2 vectors produces scalar

- ◆ Assume two vectors of length 3
 - $A = [A1, A2, A3], B = [B1, B2, B3]$
- ◆ Multiply vector A by vector B for result C:
 - $C = (A1*B1) + (A2*B2) + (A3*B3)$
 - 3 multiplications & 2 additions
 - 5 floating point operations



Matrix Operation

A00	A01	A02	A03
A10	A11	A12	A13
A20	A21	A22	A23
A30	A31	A32	A33



Matrix Size

- ◆ 1,000 x 1,000 x 1,000 matrix = 1 Billion elements
- ◆ CFD Example:
 - position = 24 bytes
 - direction = 24 bytes
 - velocity = 8 bytes
 - density = 8 bytes
 - pressure = 8 bytes
 - temperature = 8 bytes
 - 80 bytes per element
- ◆ 80 Gigabytes raw data!



F/A-18 Hornet



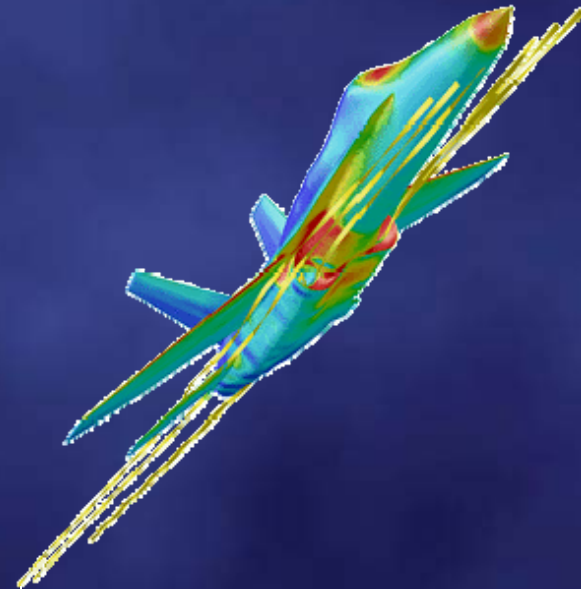
Length: 56'
Width: 38'
Height: 15'
Weight: 23,000 lb
Top Speed: Mach 1.8



F18 Aerodynamics Calculations

- ◆ Length: 56'
- ◆ Add space: 68' (816") working volume
- ◆ 1" grid = 816 x 576 x 300 (141M elements)
 - Current large models 2M-3M elements
- ◆ 100-1,000 time steps required

- ◆ Very large problem!
- ◆ Run times can be multiple weeks





Critical Response Time

- ◆ Grand Challenge
- ◆ Long job
- ◆ Weekend job
- ◆ Overnight job
- ◆ Single shift job
- ◆ Multiple runs per day
- ◆ Soft Interactive
- ◆ Real Time

Fundamental Insights

“CYA”

Useful

Impact design process

Important to design

Integral to design

Becomes design

Holy grail



Real World Technical Computing Definitions

- ◆ High Performance
 - *Expensive*
- ◆ High Performance Computer
 - *High performance memory system with a processor attached*
- ◆ Scalable High Performance System
 - *High performance interconnect with a lot of processors attached*
- ◆ MegaFLOP
 - *Marketing unit of measure; see MachoFLOP*



Running Big Jobs

- ◆ Multiprocessing
 - A bug, not a feature!
- ◆ The ideal computer
 - An infinitely fast uniprocessor
- ◆ Why a uniprocessor?
 - Easiest to program
 - Easiest to debug
 - Easiest to optimize
 - Easiest to manage
 - Easiest to understand



Why Multiprocessors?

- ◆ We can't build a fast enough processor!
 - Example: Astrophysics problem with estimated 5,000 year execution time on current processors.
- ◆ And we never will....
 - No matter how fast we make it, users will just tackle bigger problems.
 - We consider this a good thing...



Advantages of Multiprocessor

- ◆ More Work
- ◆ Less Time
- ◆ Use to:
 - Run bigger jobs (capability)
 - Run more jobs (capacity)



Multiprocessor Challenges

- ◆ Connect processors together
- ◆ Build multiprocessor system
 - How many processors?
- ◆ Manage multiprocessor system
 - Operating system
 - Resource management
 - System management
- ◆ Develop multiprocessor (parallel) applications



Scaling

- ◆ Add more resources to job
- ◆ Problem: 2 processors <2X of single processor
- ◆ If scaling is 90%:
 - 1 processor = 1.0X
 - 2 processors = 1.9X
 - 4 processors = 3.4X
 - 8 processors = 5.7X
 - 16 processors = 8.1X
 - 32 processors = 9.7X
- ◆ 32nd processor adds 0.04 of performance



System Design Configurations

- ◆ How many processors do you need?
- ◆ What are inter-processor communication requirements?
 - Determined by applications
- ◆ What is the target cost?
 - Hardware cost
 - Software (and software development) cost
 - System management cost



Communication Cost

$$\text{Cost} = (\text{Rate}) \times \frac{(\text{Bandwidth})}{(\text{Latency})}$$



System Complexity

- ◆ Complexity = (# processors) X (communication)
- ◆ Large # processors and high communication requirements produces an expensive system
- ◆ Reducing # processors or reducing communication requirements greatly reduces system cost
 - Clusters are cost effective because they combine high processor counts with low communication costs

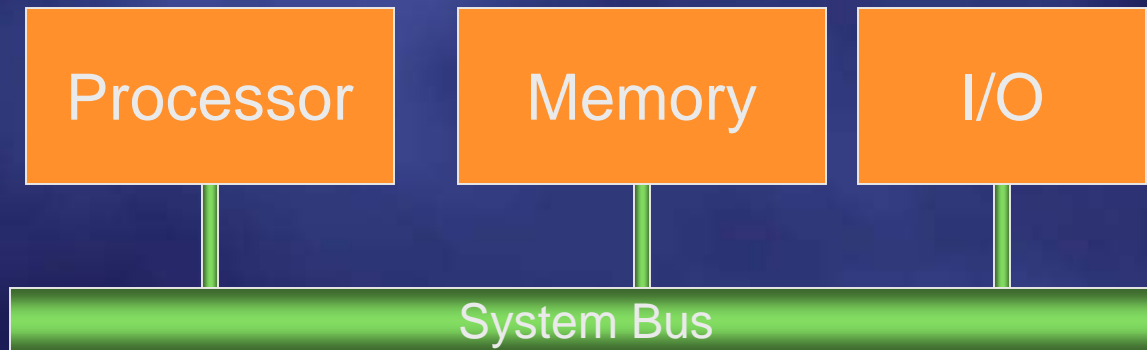


Building *Effective* Technical Computing Systems

- ◆ Start with most powerful processor possible
- ◆ Provide fast memory subsystem
 - Plus large, fast cache
- ◆ Connect processors together
 - High bandwidth
 - Low latency
 - Low Overhead
- ◆ Make the system usable and manageable

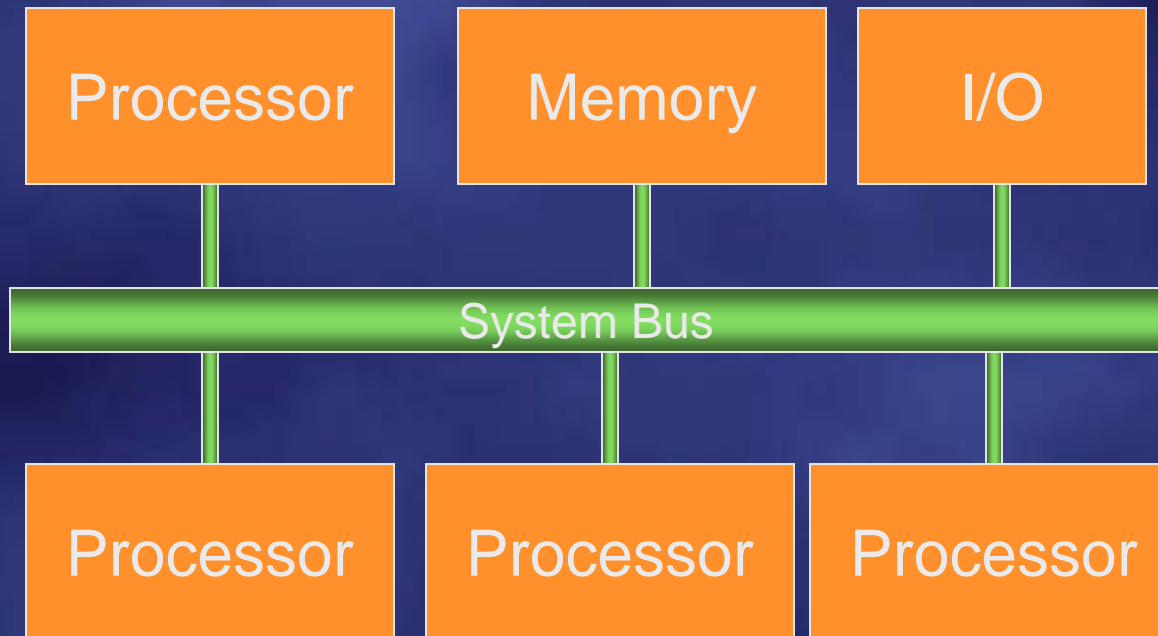


System Components



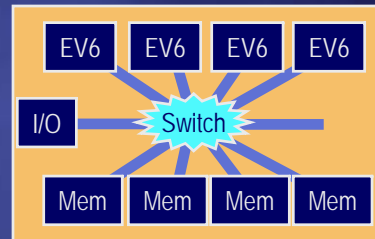


Unbalanced system





Shared Resources -- Crossbar Switch

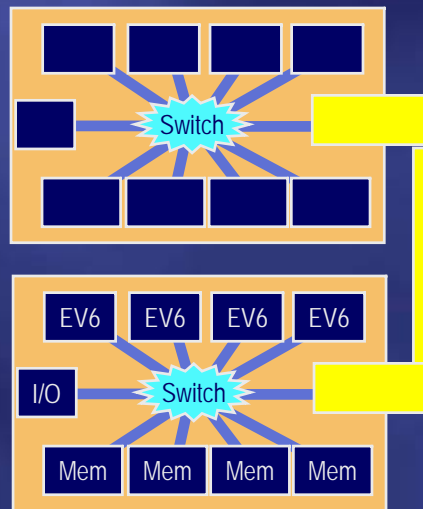


4 Processors

- ◆ Direct point to point connection between resources
- ◆ Bandwidth scales with more resources
- ◆ Difficult to build *high performance* crossbar switch with more than 8-12 ports



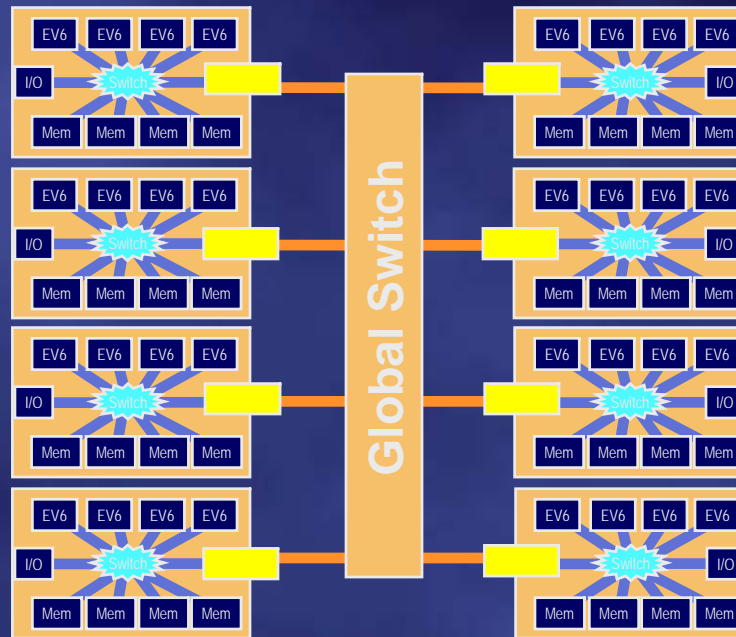
Connected Crossbar Switches



8 Processors



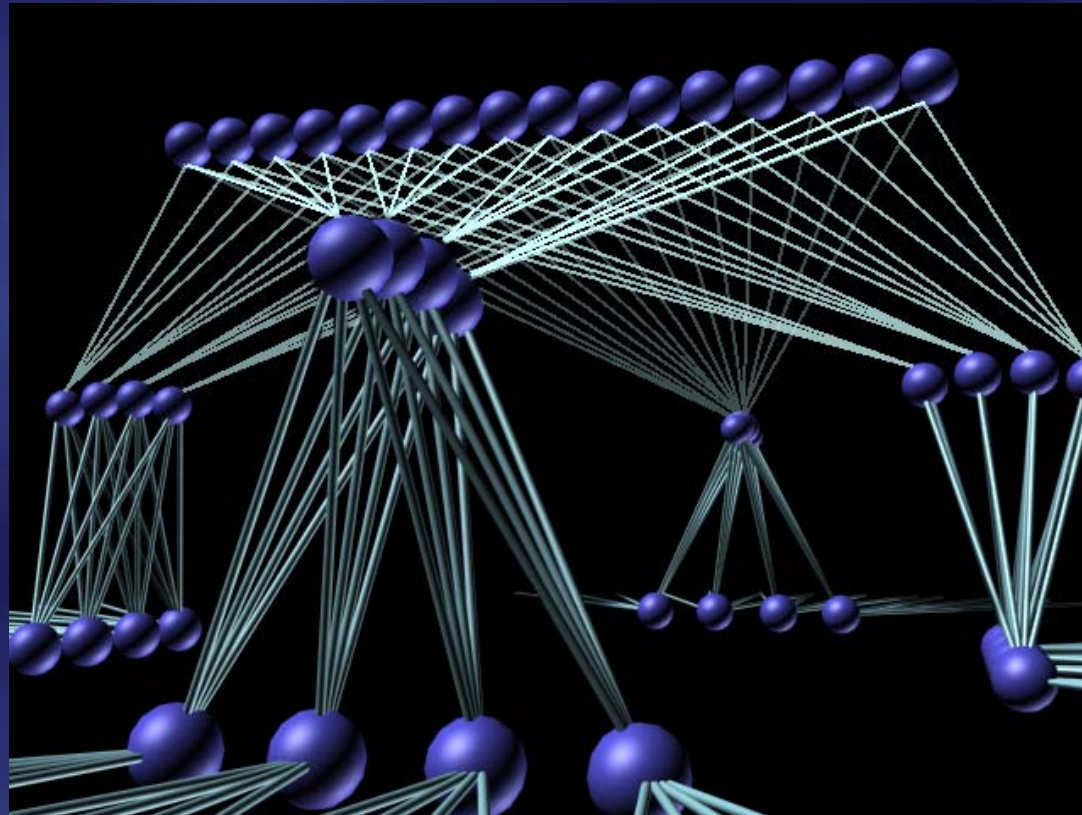
Hierarchical Crossbar Switch



32 Processors



Out of Box Experience.... Switching System Area Interconnect



128+ Nodes



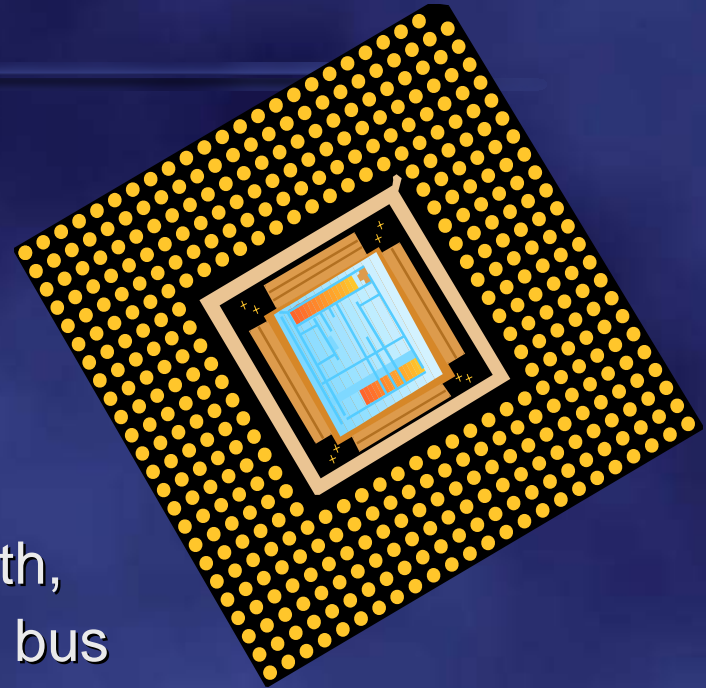
Compaq HPTC strategy

- ◆ Fastest possible processor
- ◆ Combine processors into SMP nodes
 - System Bus
 - Crossbar Switch
 - Hierarchical Crossbar Switch (NUMA)
- ◆ Combine nodes
 - Network interconnect
 - Memory Channel interconnect
 - QSW interconnect



Alpha Processor

- ◆ 700+ MHz
 - 1.4+ GFLOP
- ◆ 6 Execution units (4 integer, 2 floating point)
 - 2.8+ billion instructions/sec
- ◆ L1 cache: 11 GB/sec bandwidth,
- ◆ 2MB-8MB L2 cache, backside bus
- ◆ 333 MHz, 2.6 GB/sec front side bus
- ◆ Aggressive out-of-order execution
 - 80 instructions in-flight
 - 30 memory operations in flight





Alpha Systems: ES40



- ◆ 4 Alpha processors
- ◆ 40+ GB/sec L1 cache bandwidth
- ◆ 5.2 GB/sec memory bandwidth
 - 1.3 GB/sec per processor
- ◆ 533 MB/sec I/O bandwidth (PCI)
- ◆ 16 GB memory



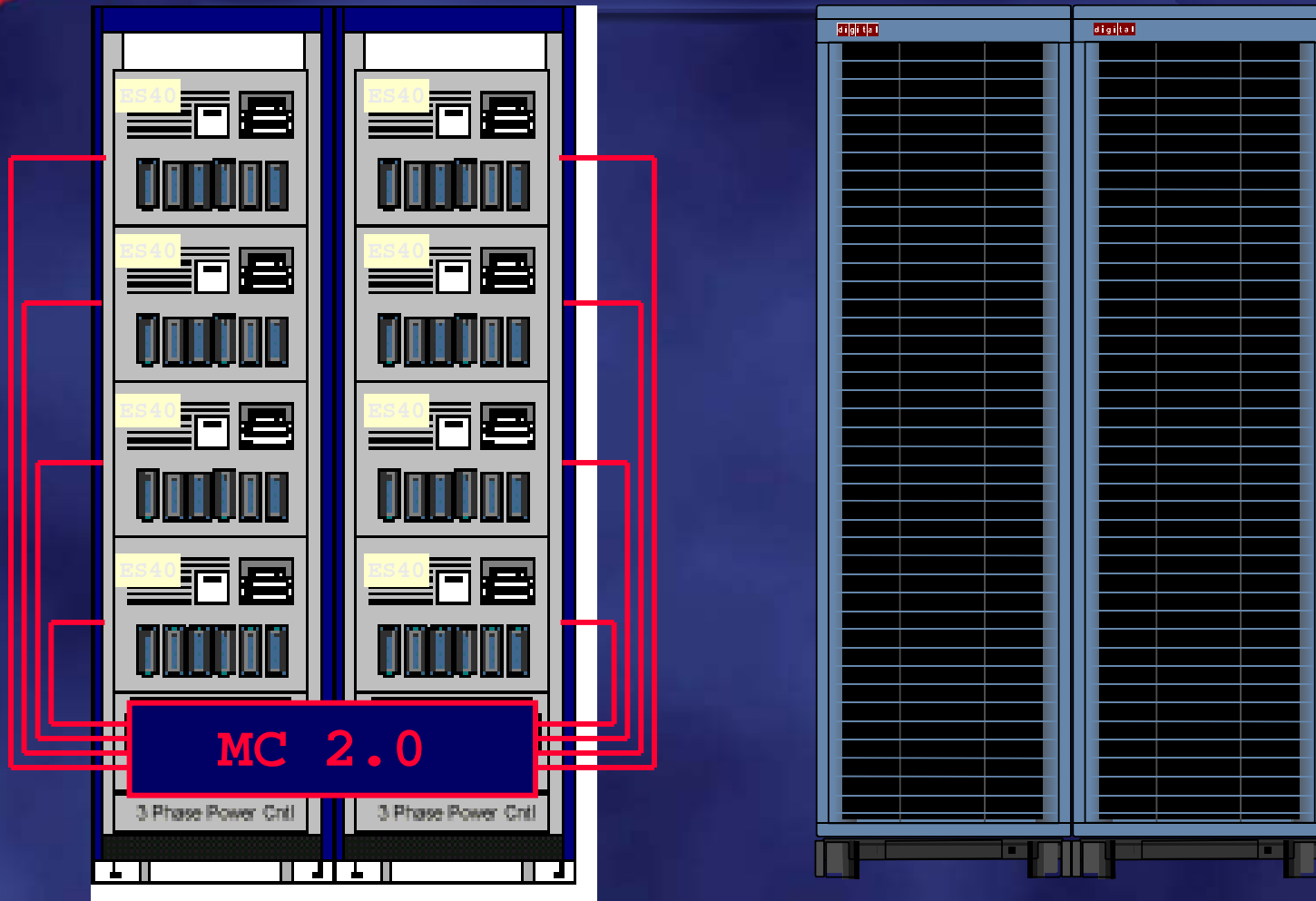
Future Alpha Systems

- ◆ 32+ processors per node
 - Hierarchical crossbar switch architecture
 - Near Uniform Memory Access
- ◆ 256+ node systems
 - 500-4,000-10,000+ processor systems



HPC320

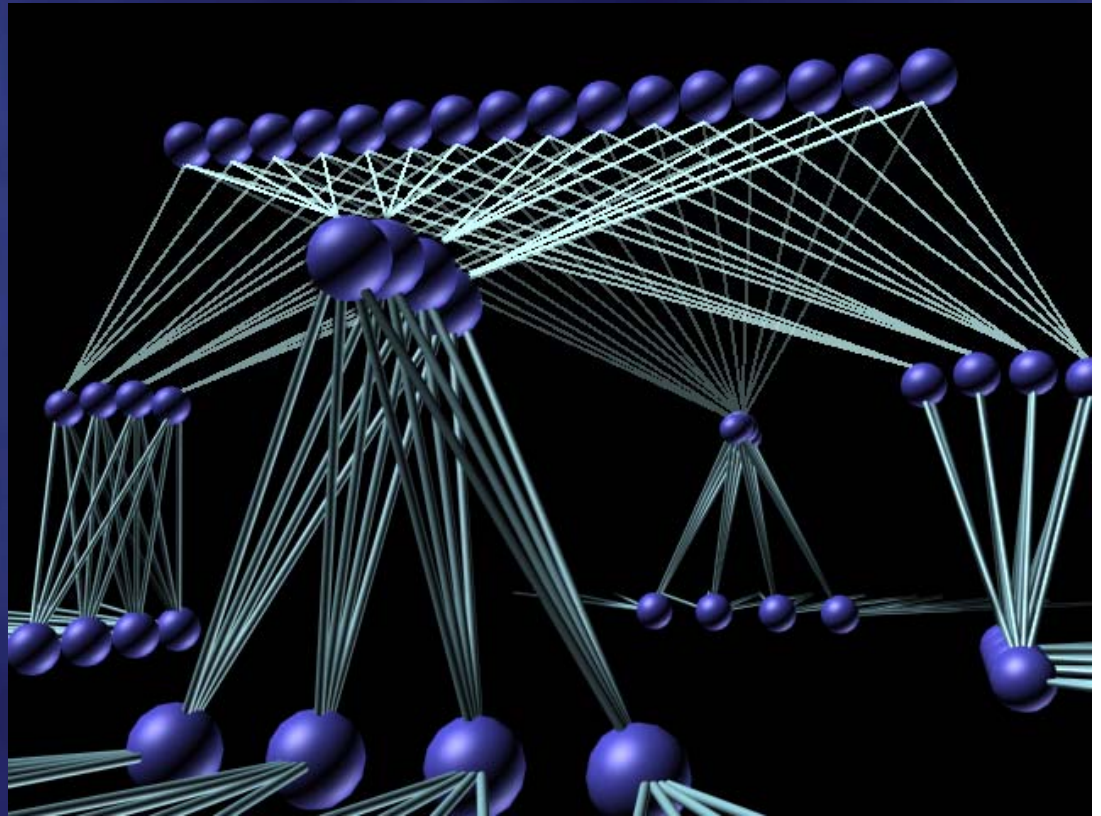
1.2m





Out of Box Experience.... Switching System Area Interconnect

- Elan-3 PCI adapter
 - DMA driven
 - Get and put
 - 200-800 MB/s/rail bi-directional
- Elite “fat tree” switch
 - 8-way x-bar chips
 - 16 or 128 port package
 - Up to 20m cables
 - 0.035 μ s switch latency
- Multiple virtual circuits and load Balancing
- Latency: 3 μ s DMA/shmem, 6 μ s MPI



A 3D perspective rendering of a large-scale server system. The system consists of several long, dark-colored server racks arranged in a grid-like pattern on a dark floor with a light grid pattern. Each rack is filled with numerous server units, which are depicted as small, light-colored rectangular blocks with red and blue indicators. The overall scene is illuminated with a cool blue light, creating a futuristic and high-tech atmosphere. The text "10,000+ Processor System" is overlaid in the center of the image in a bold, orange font.

10,000+ Processor System



HPTC Summary

- ◆ Processor Power is important
 - ◆ Bandwidth is good
 - ◆ Latency is bad
 - ◆ Overhead hurts
 - ◆ Software is hard
 - ◆ System management happens
- ◆ ***Application performance is the reason you care***

COMPAQ

Better answers

www.compaq.com