

HPCx Hardware



- Introduction
- Power5 CPUs
- Memory
- Interconnect

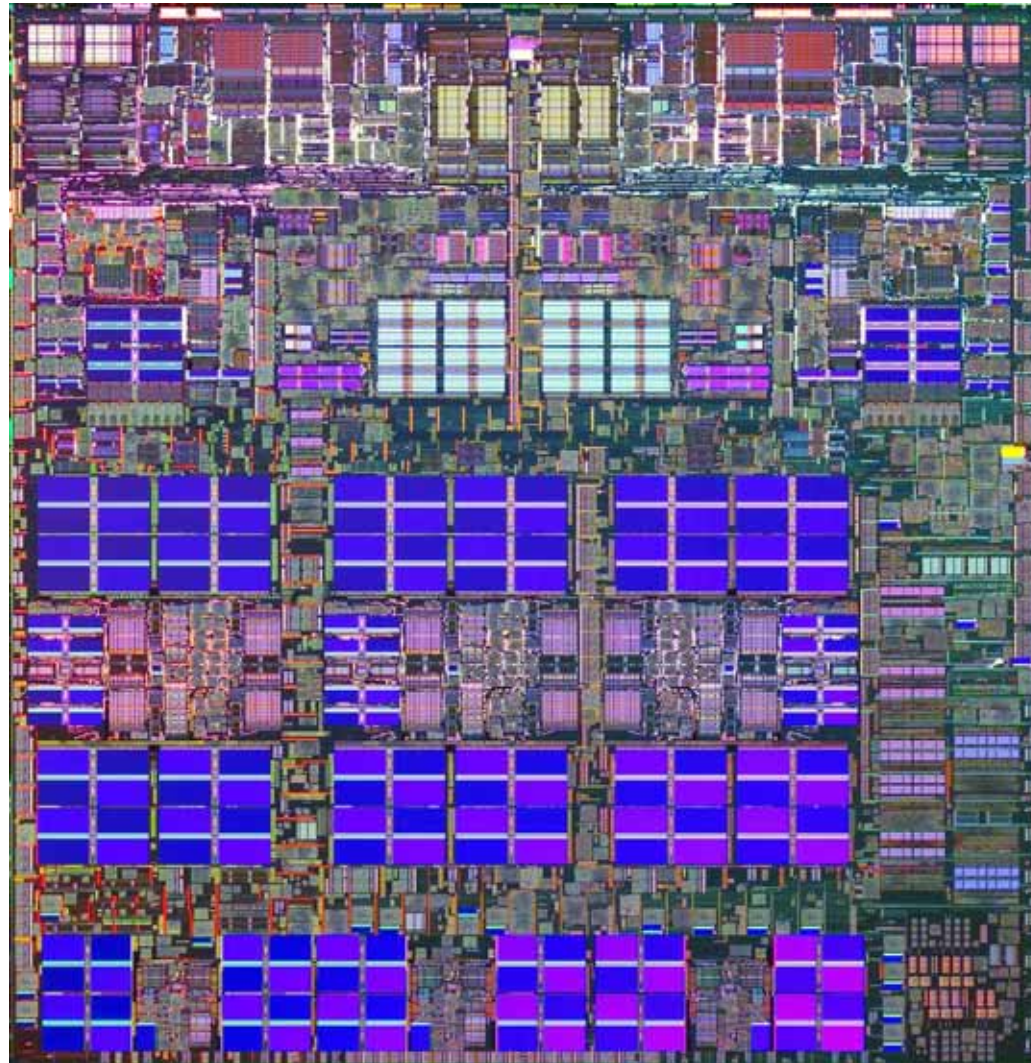
- **EPSRC's objectives for follow-on service to CSAR**
 - aim "to deliver the optimum service resulting in world-leading science"
 - to address "the problems involved in scaling existing codes to the capability levels required"
- **Capability Computing**
 - use significant fraction of resource, eg 512+ CPUs
- **HPCx Service won the competitive procurement**
 - six-year contract worth £53M
 - technical support provided by EPCC & Daresbury Lab
 - three-phase hardware roadmap supplied by IBM



- Previously two approaches for high-end HPC
- Massively Parallel Processing
 - many single-CPU nodes each with their own memory
 - communicate using a high-speed network
 - eg Cray T3D at EPCC, Cray T3E's at EPCC and CSAR
- Shared Memory Systems
 - multiple CPUs all sharing the same memory space
 - communicate via reads and writes to shared memory
 - eg SGI Origin 3000, SGI Altix at CSAR
- Modern systems from all vendors combine both
 - HPCx systems will all be shared-memory clusters

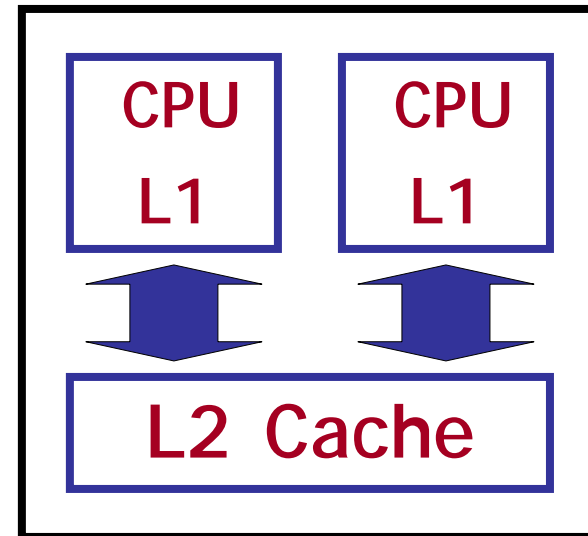
- **Building block of the HPCx system**
 - 1.5 GHz clock speed
 - two independent floating point units per CPU
 - single instruction for floating point multiply-add (FMA)
 - peak is therefore 6.0 GFlops per CPU!
- **Very deep pipeline**
 - floating point instructions take 6 cycles to complete
 - peak performance requires 12 independent FMAs ...
 - and ability to access data from memory at that rate
- **Compilers are aware of the architecture**
 - even so, don't expect more than 15-20% of peak for real codes

IBM Power5 Chip

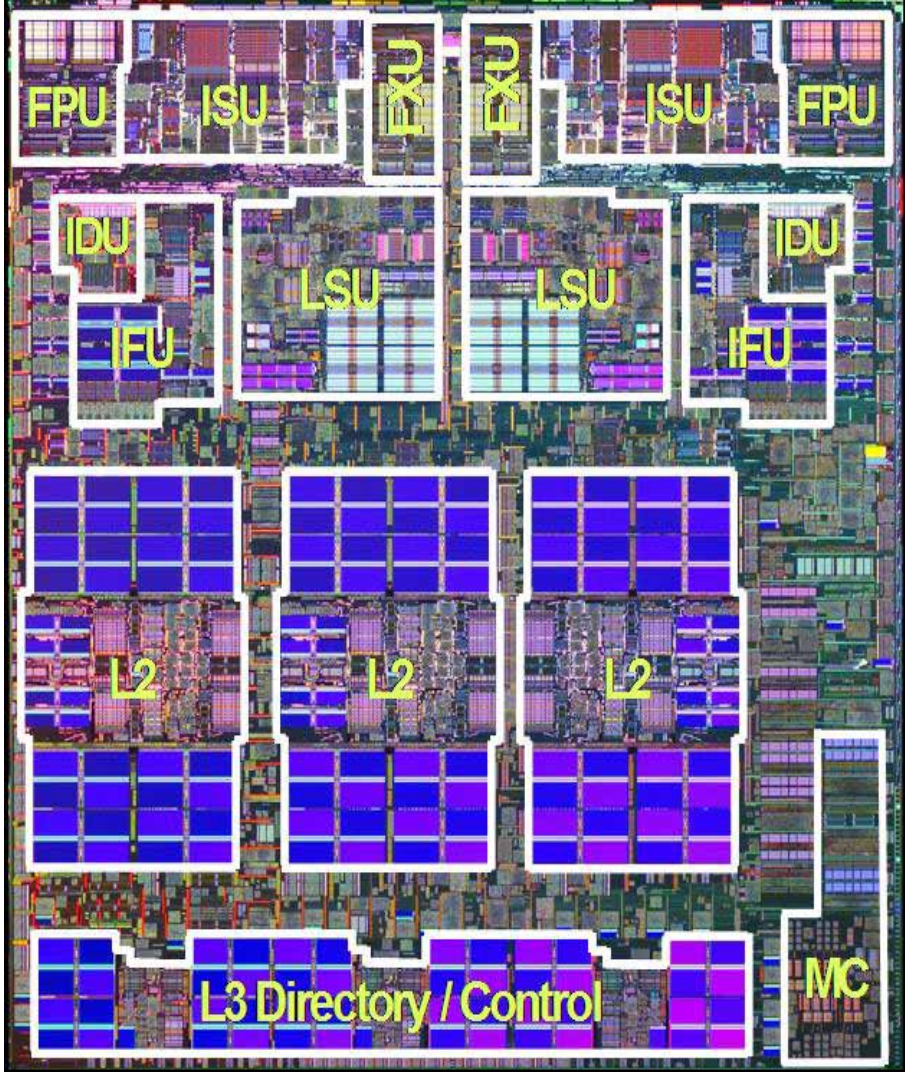


- Everything is based around 64-bit arithmetic
 - ie Fortran **DOUBLE PRECISION** or C **double**
- Each CPU has
 - 120 FP registers actually accessible by the CPU
 - Compiler sees 32 virtual FP registers
- Level 1 caches based on 128-byte lines
 - 64K direct-mapped first-level instruction cache
 - 32K 2-way associative first-level data cache
- Write through policy in L1
 - data written to L1 cache only if already allocated
 - always written to L2 cache

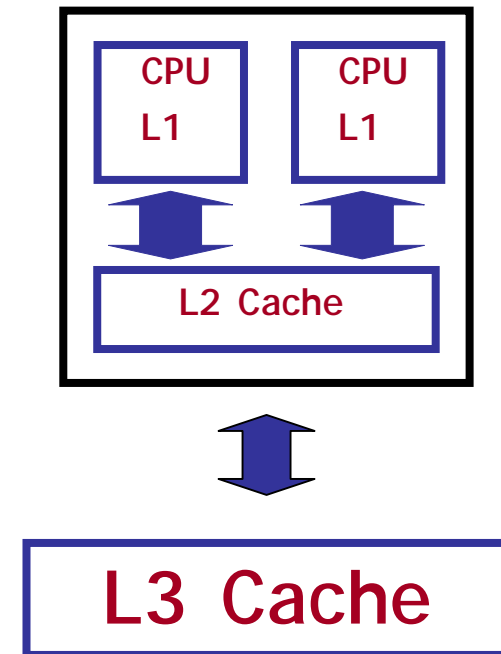
- A single chip comprises
 - 2 independent CPUs
 - shared L2 cache
 - peak of 12.0 Gflops
- L2 cache characteristics
 - 128-byte lines
 - 10-way associative
 - holds both instructions and data
 - 1.9Mb capacity combined data and instruction cache
 - write-back policy



Power5 Chip



- 2 chips are packaged together with a shared level 3 cache into a dual-core module (DCM)
- L3 cache characteristics
 - 36 Mb total capacity
 - Shared between 2 processors
 - 256-byte lines
 - 12-way associative
 - instructions and data



- Each eServer 575 node contains 8 DCM's
 - 16 CPUs
 - L3 cache local to each DCM
 - 32 GB shared memory per node
 - each node runs a separate copy of the operating system
- 8 nodes make up a frame/cabinet
 - 128 CPUs
 - peak of 768 Gflops





- Full system has 12 frames
 - 1536 CPUs and 3072 Gb main memory
 - 9.2 Tflops peak (~7.4 Tflops Linpack)
- Naming convention
 - $lmfnn(n)$ is logical partition no. m in frame no. nn or nnn
 - e.g. l1f82

- **Hardware is quite complicated**
 - details perhaps an issue for ultimate performance
 - not really important for porting and early runs
- **Don't panic!**
 - users see 96 16-way shared-memory nodes
 - think of each CPU having 2 Gb main memory
 - actually about 27Gb of memory per node is available to user:
other 5Gb is reserved for the OS and switch
- **Other hardware**
 - additional nodes for logging in, I O etc
 - 100 Tb of storage on disk and tape

- Nodes communicate using the IBM High Performance Switch (HPS)
 - also known as Federation
 - packet based, omega-like network
 - each node has two switch connections
- Performance as measured from user MPI code
 - 6.0-6.5 microsecond latency
 - 3 GB/s bandwidth between two tasks on different nodes
 - 4 GB/s aggregate bandwidth between two nodes

- Expect most users to do MPI message-passing
 - using either Fortran or C
- Each CPU is programmed independently
 - shared-memory nature of nodes not an issue ...
 - ... except in terms of performance
- MPI programmers see HPCx as
 - 1536 independent processors
 - each with ~2Gb of memory

- **Additional nodes for I/O**
 - 36Tb of disk storage
 - 64Tb of tape storage
 - Manual archiving from disk to tape.
- **Compute nodes have some local disk, but this is reserved for the OS.**
 - All user I/O goes to the I/O nodes over the switch
 - I/O bandwidth from a single node effectively limited by switch bandwidth.
 - Aggregate bandwidth of ~1Gb/s from multiple nodes to same file is achievable.

- Constellation system (cluster of SMPs)
- Cached architecture
- CPUs share memory bandwidth
 - L2 and L3 cache shared as well as main memory
- CPUs share communications bandwidth
 - two switch connections among 16 processors
- Machine comms/calc balance
 - significant change from T3E
 - much faster CPUs (7x to 14x) but similar communications (1.0x to 3x)

- December 2002: 3.4 Tflops Linpack
 - 40x32-way p690 (Regatta H) frames + Colony Switch
- May 2004: 6.188 Tflops Linpack
 - 50x32-way Regatta H+ frames + Federation switch
- November 2005: 7.395 Tflops Linpack
 - 96x16-way eServer 575 LPARs + Federation switch
- November 2006: 12 Tflops Linpack
 - possibly 192x16-way eServer 575 LPARs + Federation switch

- HPCx one of the world's most powerful machines
 - One of the largest academic supercomputers in Europe
 - upgrades should continue to keep it competitive
- Architecture similar to almost all modern systems
- Some challenges for programmers
 - eg scaling to many 100's of CPUs
 - but do not expect any major porting problems
- Focus is on Capability Computing