An Invited Presentation to Workshop on HPC Programming Languages:

Summary: What is the Future of Architecture

Thomas Sterling
Department of Computer Science
December 12, 2006

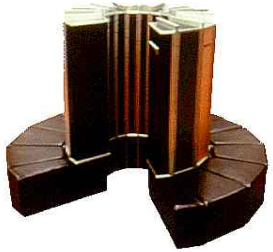
Center for Computation & Technology

AT LOUISIANA STATE UNIVERSITY

Notes

- Multicore – Niagra, cell, and everything else
  - Cell: gather-compute-scatter
- Transactional memory
- Shared memory: yes or no
- Reconfiguration – mixed media
- Global asynchrony
- Accelerators – did we forget Amdahl?
- Technology matters – power, bandwidth, wasted area
  - Off chip memory controls performance
  - IPC/core more sensitive to latency than bandwidth
  - Flat off chip physical latency implies relative latency grow with clock
- PIM
  - PNM
- More $$ for watts than hardware

A Growth-Factor of a Billion in Performance in a Single Lifetime



1959
IBM 7094

1976
Cray 1

1991
Intel Delta

1996
T3E

2003
Cray X1

1949
Edsac

1 | 10^3 | 10^6 | 10^9 | 10^{12} | 10^{15}

One OPS | KiloOPS | MegaOPS | GigaOPS | TeraOPS | PetaOPS

1823
Babbage Difference Engine

1943
Harvard Mark 1

1951
Univac 1

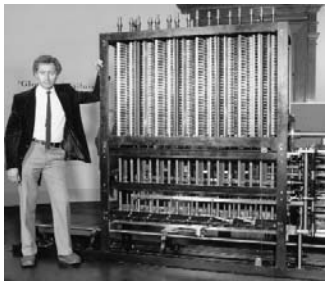
1964
CDC 6600

1982
Cray XMP

1988
Cray YMP

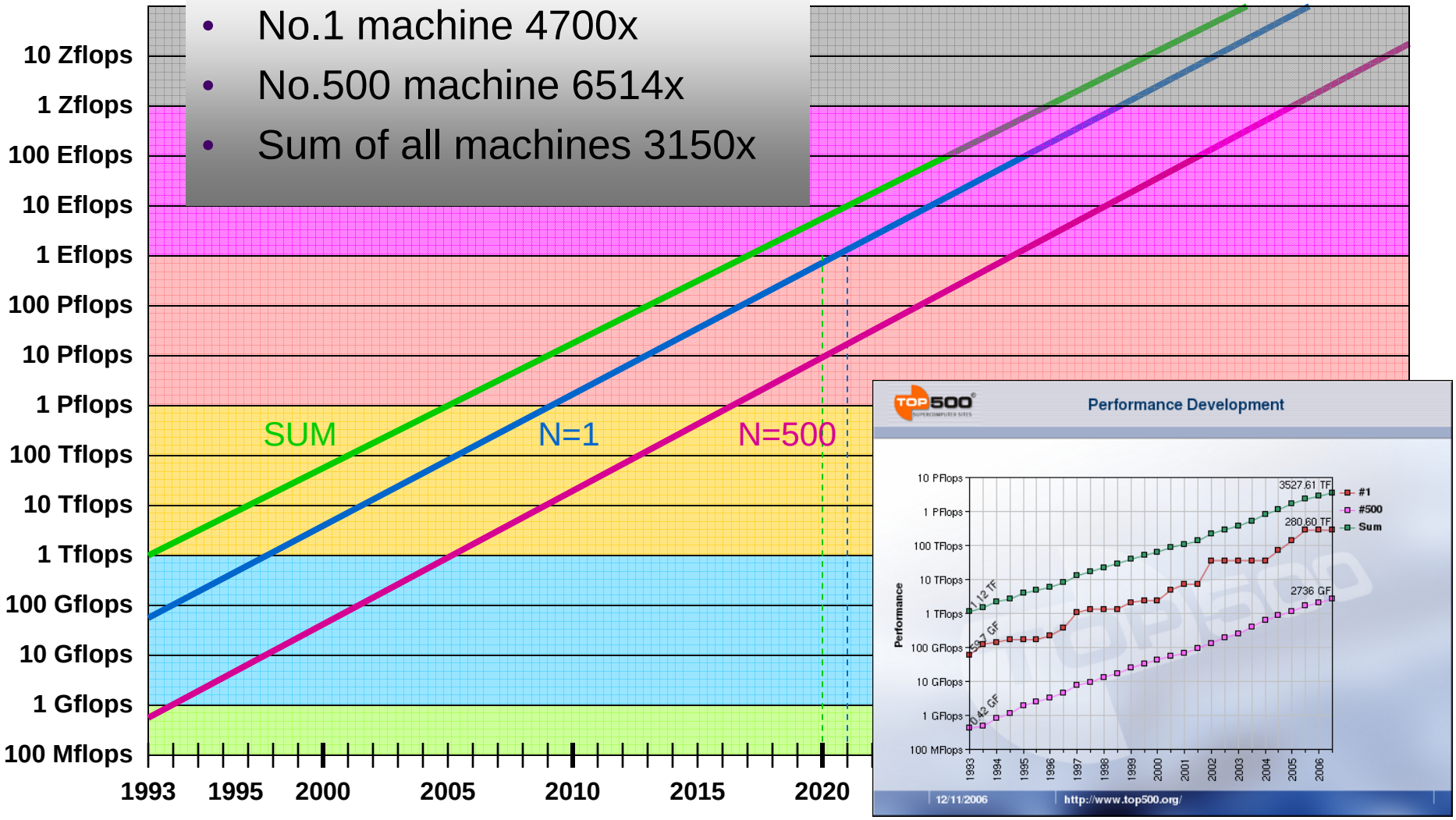
1997
ASCI Red

2001
Earth Simulator



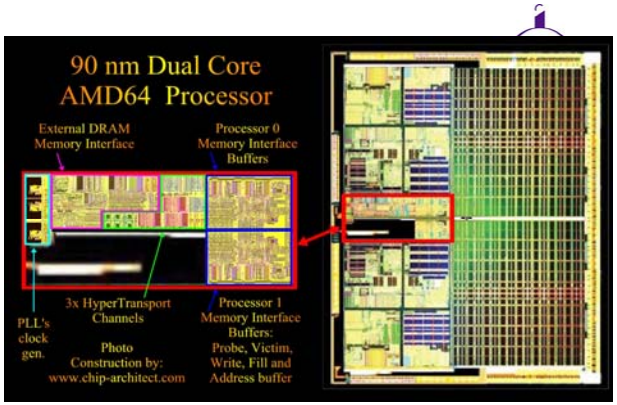
Linpack 1 Exaflops in 2020

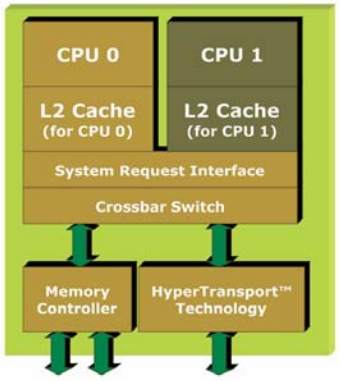
- Spans 13.5 years
- No.1 machine 4700x
- No.500 machine 6514x
- Sum of all machines 3150x



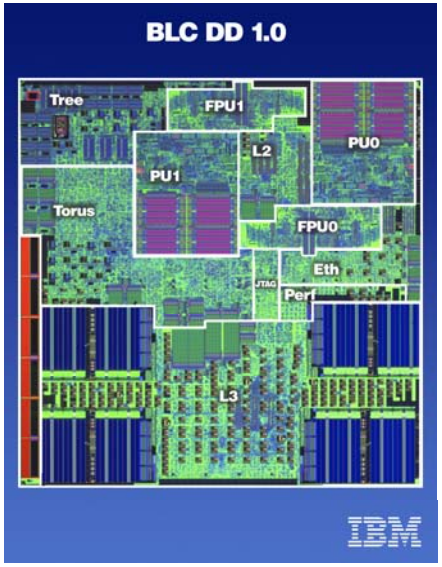
Multi-Core

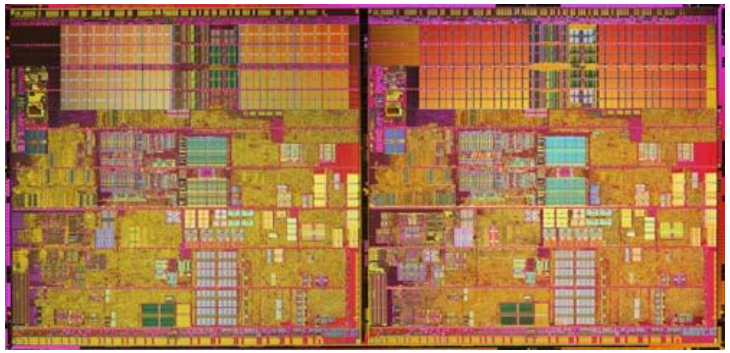
- Motivation for Multi-Core
  - Exploits increased feature-size and density
  - Increases functional units per chip (spatial efficiency)
  - Limits energy consumption per operation
  - Constrains growth in processor complexity
- Challenges resulting from multi-core
  - Relies on effective exploitation of multiple-thread parallelism
    - Need for parallel computing model and parallel programming model
  - Aggravates memory wall
    - Memory bandwidth
      - Way to get data out of memory banks
      - Way to get data into multi-core processor array
    - Memory latency
    - Fragments L3 cache
  - Pins become strangle point
    - Rate of pin growth projected to slow and flatten
    - Rate of bandwidth per pin (pair) projected to grow slowly
  - Requires mechanisms for efficient inter-processor coordination
    - Synchronization
    - Mutual exclusion
    - Context switching





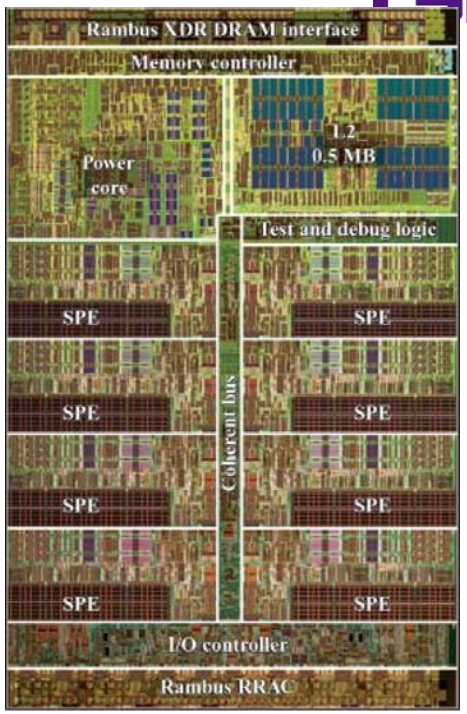
AMD Athlon™ 64 X2
Dual-Core Processor Design





Heterogeneous Architecture

- Combines different types of processors
  - Each optimized for a different operational modality
    - Performance > nX better than other n processor types
  - Synthesis favors superior performance
    - For complex computation exhibiting distinct modalities
- Conventional co-processors
  - Graphical processing units (GPU)
  - Network controllers (NIC)
  - Efforts underway to apply existing special purpose components to general applications
- Purpose-designed accelerators
  - Integrated to significantly speedup some critical aspect of one or more important classes of computation
  - IBM Cell architecture
  - ClearSpeed SIMD attached array processor



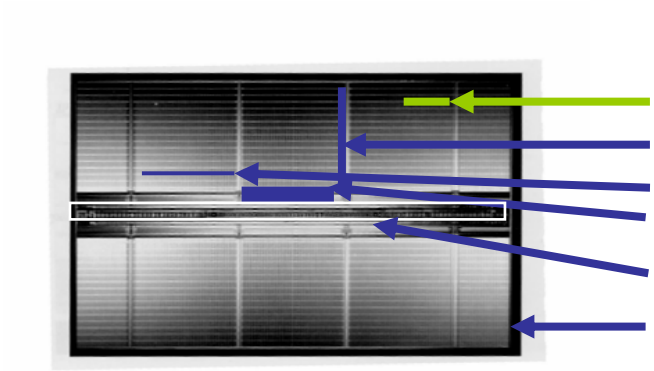
Strategic Forces Driving Direction

- Expectation and wish
  - Searching under the lamp post
  - Some do not believe in new directions
  - Others insist upon it
- Perception of Market
  - Legacy (looking forward thu the rearview mirror)
  - Status quo (only better)
  - Clock rates (Intel) and pretty boxes (Apple)
  - Size (it does matter)
  - Gullibility (bottled water)
- Money and all that
  - What does it really cost
  - How much margin will the market tolerate
    - Competition pushes back
  - How much NRE investment required (biases towards reuse and incrementalism)
- Real application opportunities
  - Games, sex, and myspace.com
  - Mobile, embedded, and user interfaces
  - Important specialized applications (where Federal funding plays)
  - Benchmarks
- What the technology will permit/enable and where it is going
  - Device
  - Architecture
  - Programming models

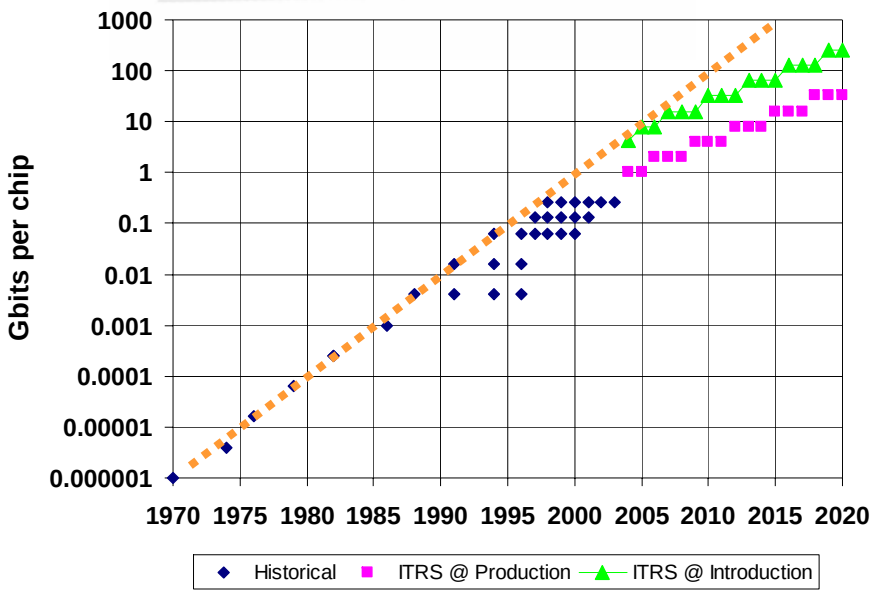
Device Technology Implications

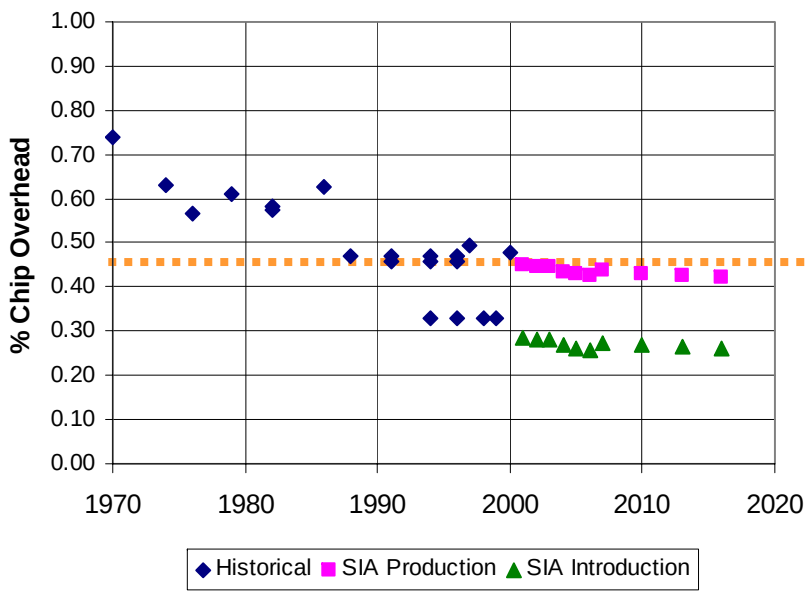
- Power, cost, size
  - We finally really care
- ALUs take relatively little real estate
  - byte/flops area ration 2 to 3 orders of magnitude
  - Clock rates uncertain
- Memory capacity defines processor size
  - If 1:1 design point forced, would dominate cost
- Memory external bandwidth small percentage of innate internal bandwidth (primary sense amps)
  - Processor chip (multicore) demand greatly exceeds the memory bandwidth (hence caches)
- Communication costs would dominate for flat interconnect
  - Low percentage versus ops
  - On chip communication growing factor on chip design and real estate overhead

Classical DRAM



- Memory mats: ~ 1 Mbit each
- Row Decoders
- Primary Sense Amps
- Secondary sense amps & “page” multiplexing
- Timing, BIST, Interface
- Kerf

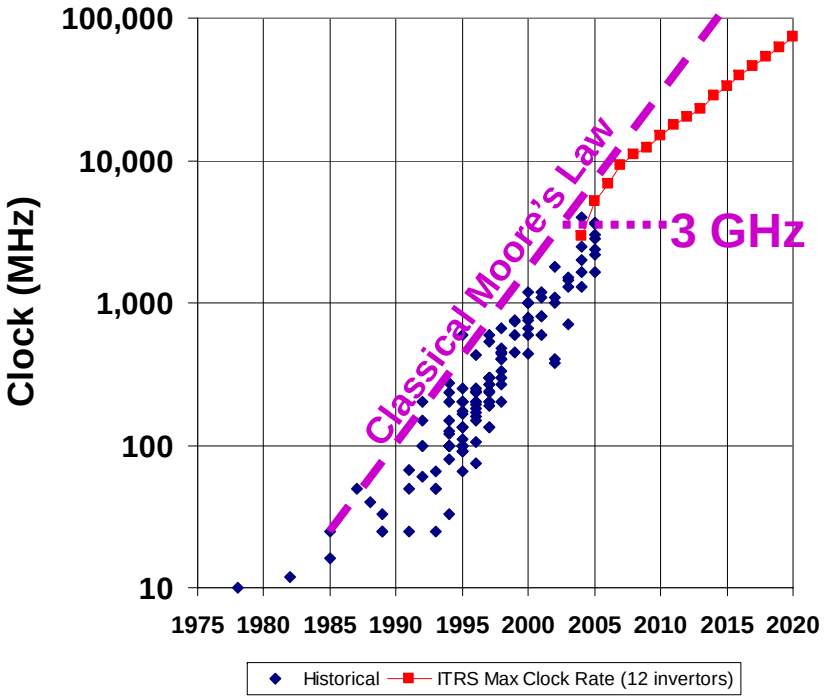


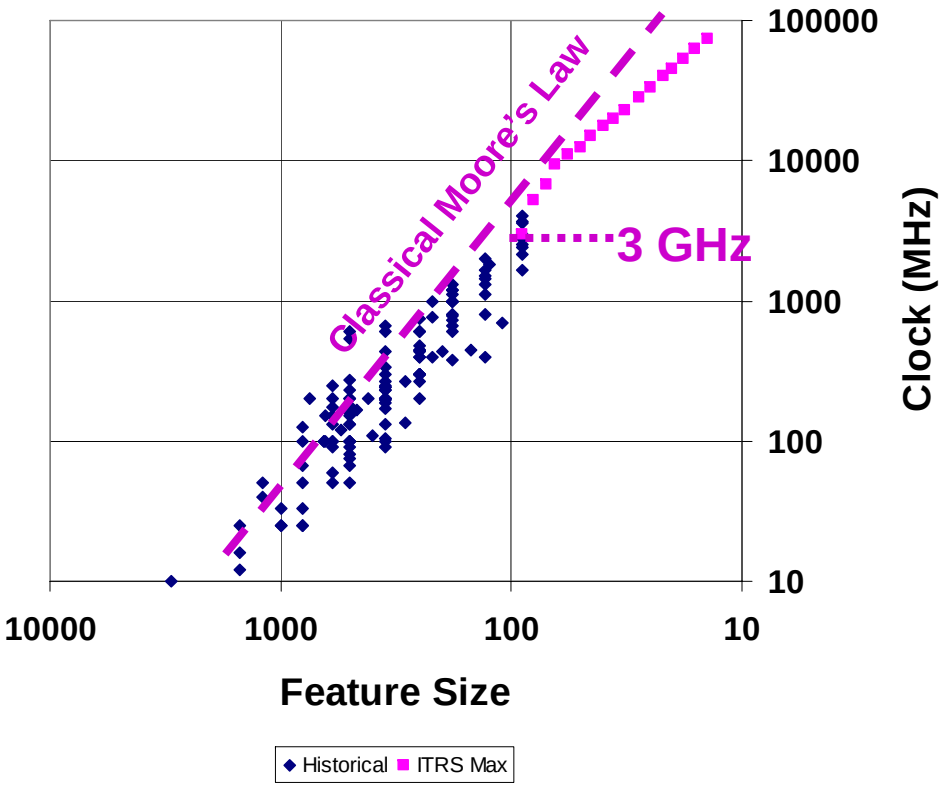


Density/Chip has dropped below 4X/3yrs

And 45% of Die is *Non-Memory*

Peak Logic Clock Rates

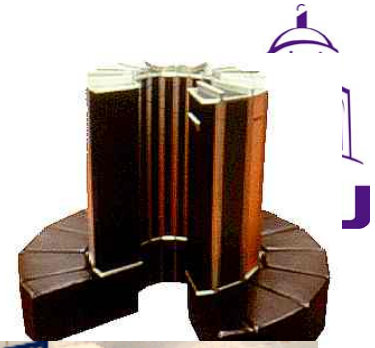


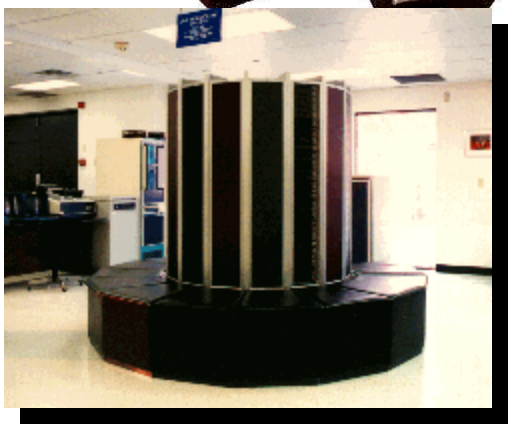


2005 projection was for 5.2 GHz – and we didn't make it in production. Further, we're still stuck at 3+GHz in production.

Golden Age of Parallel Architecture

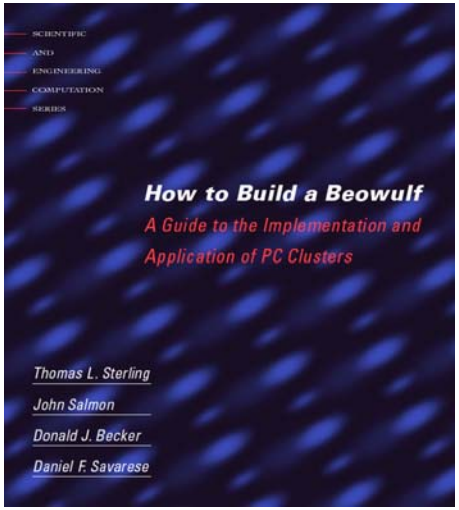
- 1975 – 1992
- Vector
  - Cray-1&2, NEC SX, Fujitsu VPP
- SIMD
  - Maspar, CM-2
- Systolic
  - Warp
- Dataflow
  - Manchester, Sigma, Monsoon
- Multithreaded
  - HEP, MTA
- Actor-based
  - J-Machine

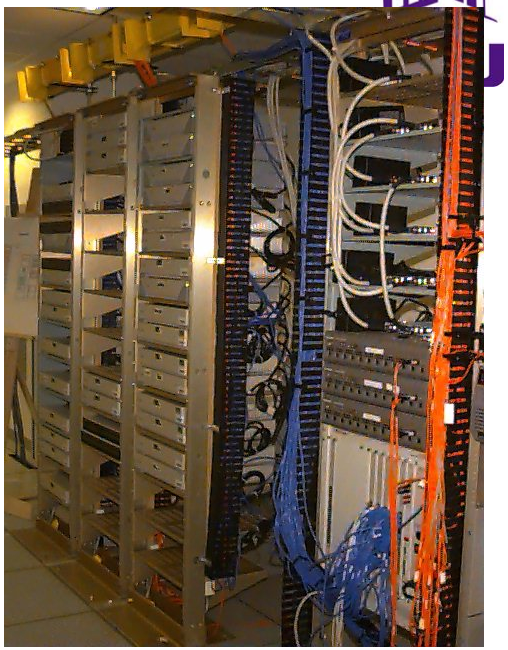




Dark Ages of Parallel Computing
Technology drivers

- 1992 to present
- Killer Micro and mass market PCs
- High density DRAM
- High cost of fab lines
- CSP
  - Message passing
- Economy of scale S-curve
- MPP
- Weak scaling
  - Gustafson et al
- Beowulf, NOW Clusters
- MPI
- Ethernet, Myrinet
- Linux





Architecture evolution or revolution

- Is HPC community simply victim of vendor whims?
  - Yes: somebody has to build the stuff we use
  - No: HPC enhancements can help general purpose processing
- Will architecture stay wedged in 1950s tradeoff balance
  - Yes: as long as possible, it keeps us in the same species
  - No: there will be a singularity point
    - punctuated equilibrium
    - When the gap between opportunity and delivery is too large
- Towards a new balance
  - Ubiquitous ALUs: high availability, not utilization
    - Increased bandwidth
    - Reduced latency
  - Bandwidth precious resource
    - Memory bandwidth most important
    - Global bandwidth
  - Memory capacity defined system size
- Threaded parallelism emergent
  - Multicore
  - Multithreaded
  - Remote invocation
  - Lightweight synchronization
- Towards simpler cores

Challenges for Future Architectures

- Performance degradation
  - Latency (idle time due to round trip delays)
  - Overhead (critical path support mechanisms)
  - Contention (inadequate bandwidth)
  - Starvation (sufficient parallelism and load balancing)
- Power consumption
  - Just too much!
  - Dominating practical growth in mission critical domains
- Reliability
  - Single point failure modes cannot be tolerated
  - Reduced feature size and increased component count
- Changing application workload characteristics
  - Data (meta-data) intensive for sparse numerics and symbolics
- Programmability & ease of use
  - System complexity, scale and dynamics defy optimization by hand

Programming Models

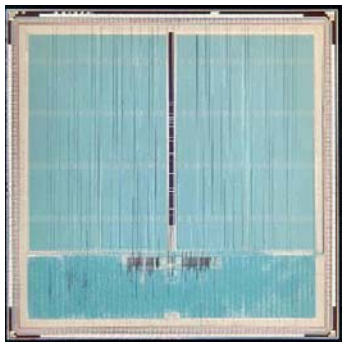
- How to coordinate application driver with architecture
- Purpose of this workshop
  - Multiple target architecture classes
- Issues
  - Exposing parallelism
  - Coordinating concurrency
  - Ease of representation of algorithm constructs and data object relationships
- Missing is the next model of computation
  - Ad hoc interrelationships among architecture elements
  - Governs architecture design
  - Guides programming language
- Who will tell us what's next?
  - Bill, Rusty, Marc
  - DARPA and its anointed vendors

ParalleX Semantics

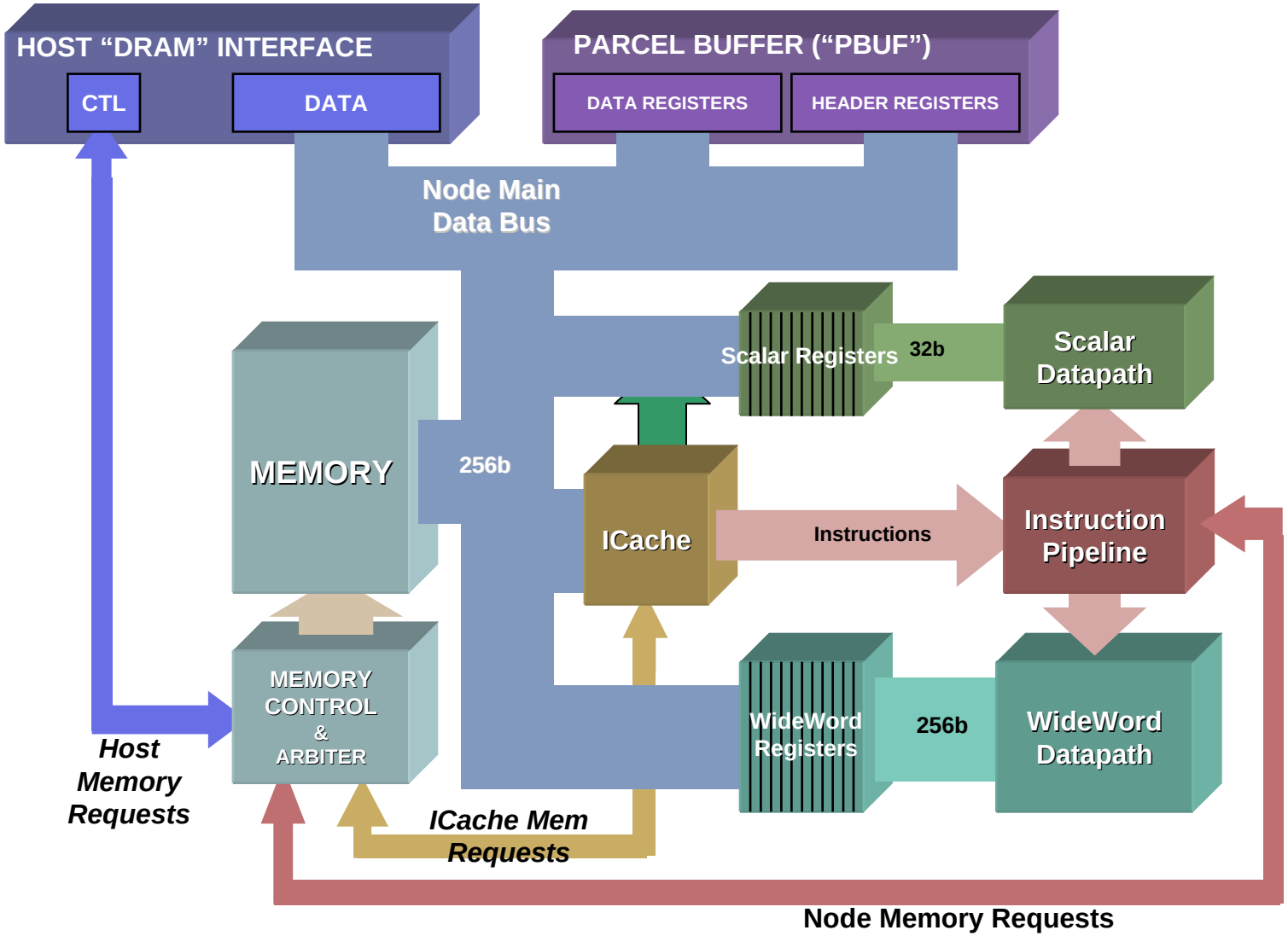
- Locality domains
  - Intra-locality: Controlled synchronous
  - Inter-locality: Asynchronous between localities
- Split-phase transactions
  - Work queue model
    - Only do work on local state
    - No blocking or idle time for remote access
- Message-driven computation
  - Parcels carry data, target address, action, continuation
- Multi-threaded
  - First class objects
  - Dataflow on transient values
- Local control objects
  - Futures
  - Dataflow
  - Data-directed parallel control
- Meta-data embedded address translation
- Failure-oriented with micro-checkpointing

The *Gilgamesh-2* System Strategy

- Cost imperatives:
  - High availability ALUs
  - High utilization of memory bandwidth
  - Percolation: cheap threads manage global parallel flow control
- New Model of Parallel computation (*ParalleX*)
  - Intrinsic latency hiding
  - Message-driven split-phase transaction processing
  - Near fine-grain in-memory synchronization local control objects
- Heterogeneous architecture for disparate temporal-locality modalities
  - High temporal locality: high ALU-density dataflow controlled
  - Low temporal locality: in memory threads
- Global name space
  - Address translation in meta-data
  - Copy semantics for targeted value-sets



DIVA USC ISI

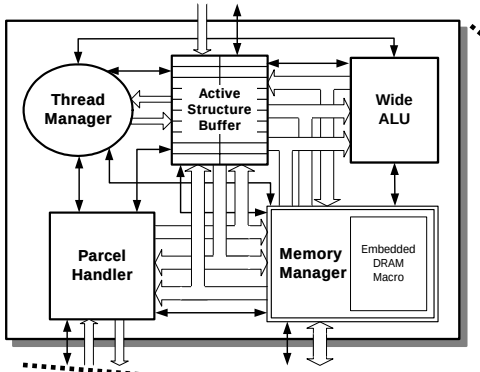

HOST "DRAM" INTERFACE
CTL
DATA
PARCEL BUFFER ("PBUF")
DATA REGISTERS
HEADER REGISTERS
Node Main Data Bus
MEMORY
256b
Scalar Registers
32b
Scalar Datapath
ICache
Instructions
Instruction Pipeline
MEMORY CONTROL & ARBITER
WideWord Registers
256b
WideWord Datapath
Host Memory Requests
ICache Mem Requests
Node Memory Requests

Gilgamesh-2 System Elements

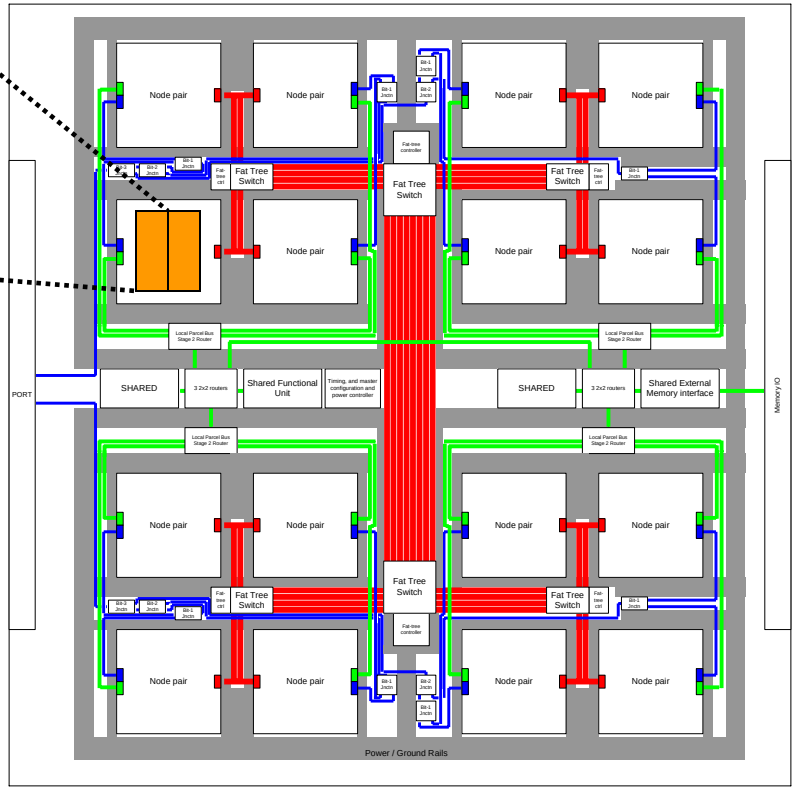
- Executable Memory
  - Supports low-temporal (e.g. touch once) locality global data operations
  - Threads in memory with wide ALUs
- Dataflow Accelerator
  - Supports high-temporal locality operations
  - Very high throughput low latency processing
  - Low power per operation
- Data Vortex optical network
  - Innovative topology
  - Low latency, low logic
  - Graceful degradation of injection rate with traffic density
  - High degree switches
- Penultimate store
  - Fast backing store for core computing
  - Exploits highest density semiconductor memory
  - Reconfigurable for fault tolerance

MIND elements

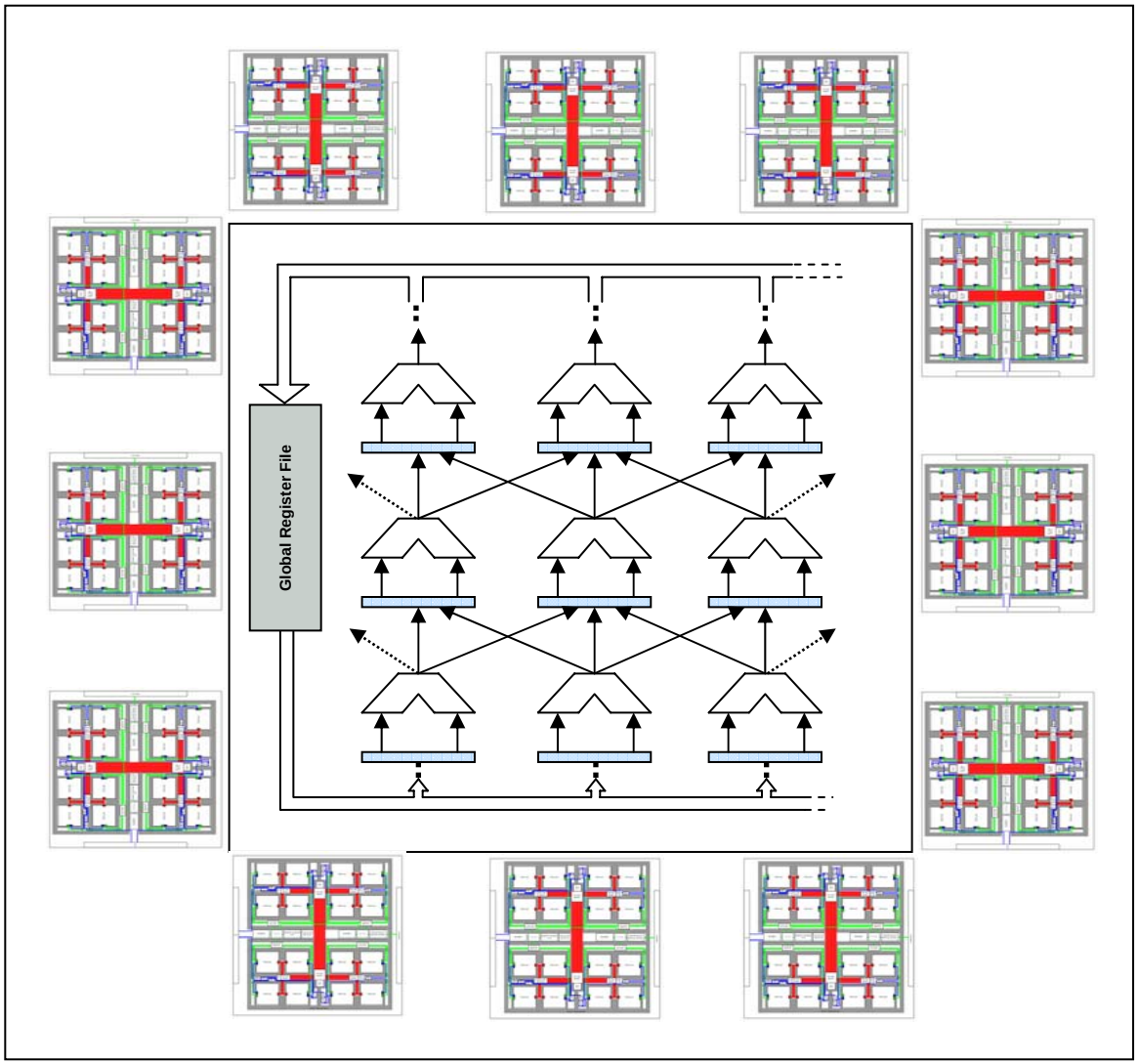
MIND memory accelerator



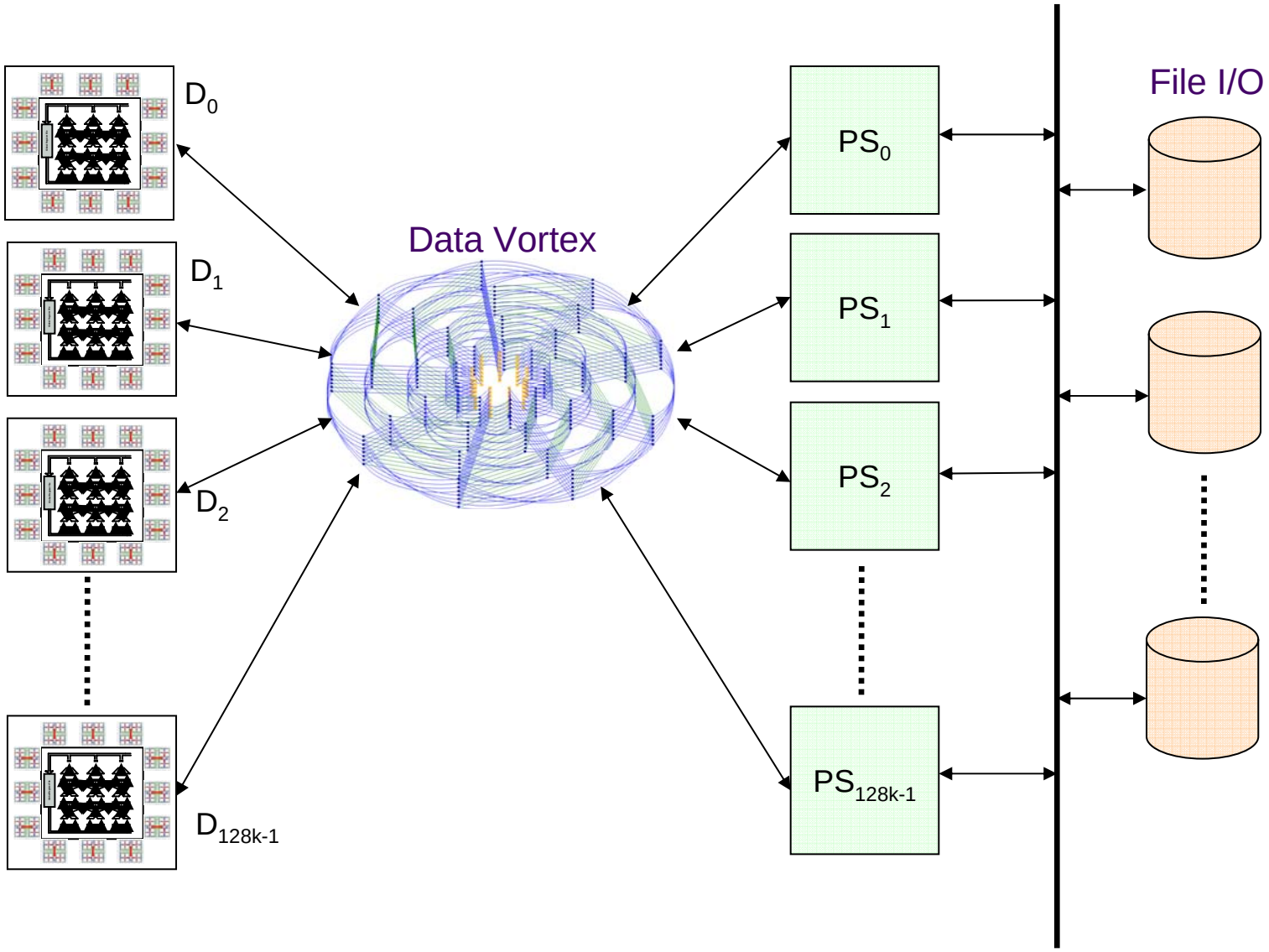
Gilgamesh



Dodecatron



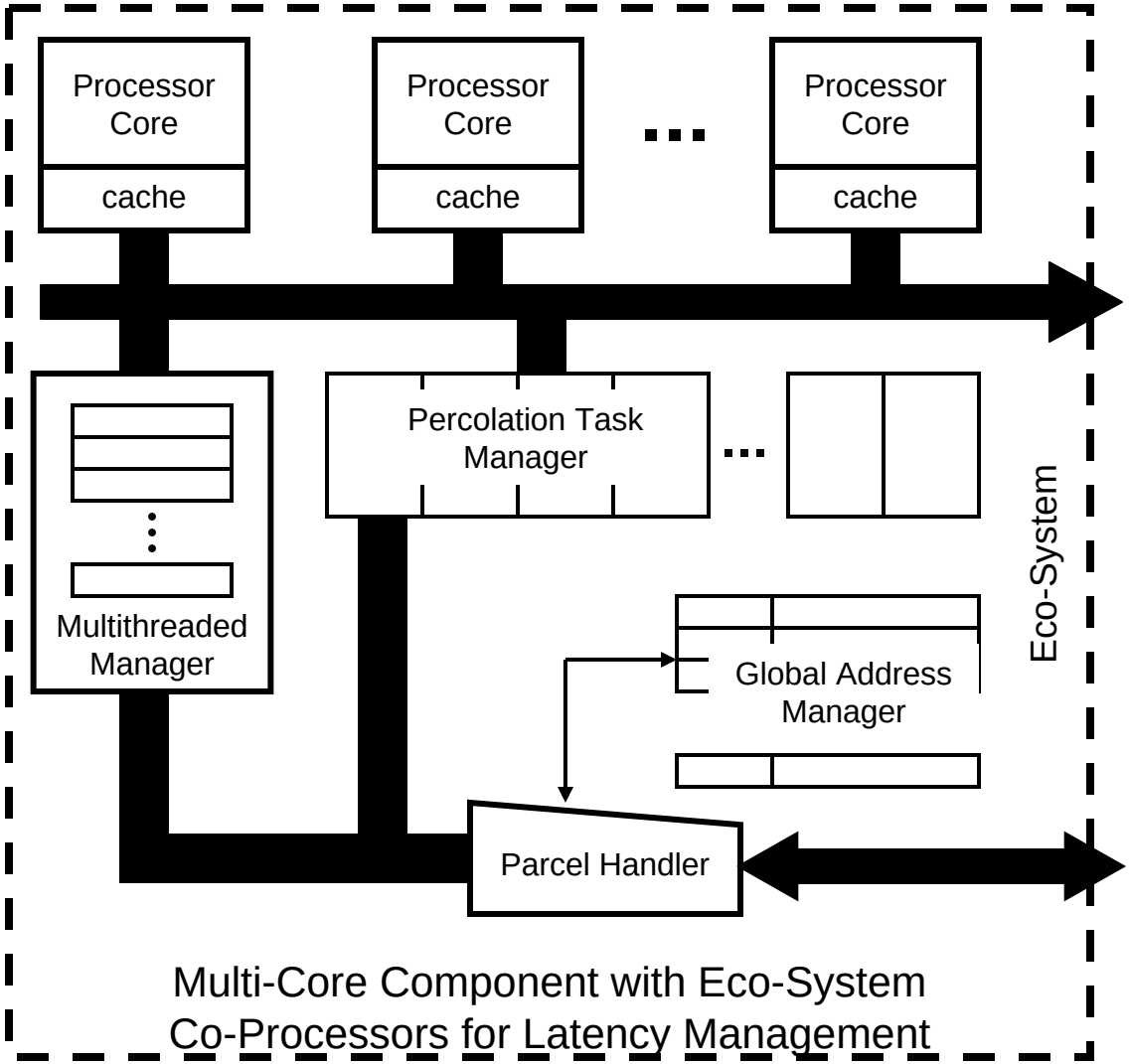
Vision2020 System



Summary : Vision2020 Characteristics

Parameter	Value
Gilgamesh component	
Clock frequency	16 GHz
MIND accelerators	32
FP operations per cycle (1 ALU)	8
Peak performance	4 TFLOPS
Memory capacity	512 MB
Dataflow component	
Clock frequency	32 GHz
FP operations per cycle (1 ALU)	1
Number of ALUs	256
Peak performance	8 TFLOPS
Dodecatron (single chip)	
Gilgamesh components	12
Dataflow components	1
Peak performance	56 TFLOPS
Memory capacity	6 GB
System	
Number of chips	128 K
Peak performance	> 7 EXAFLOPS
Dodecatron memory capacity	768 TB
Penultimate storage capacity	128 PB

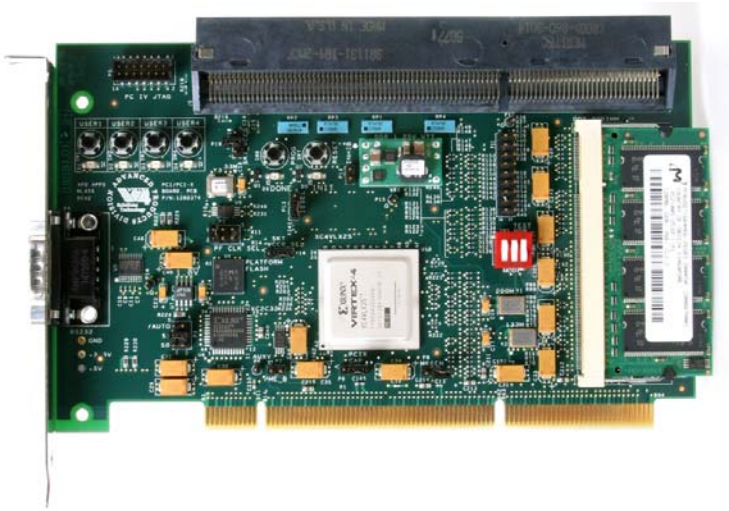
Can’t Change the Architecture but if I could:

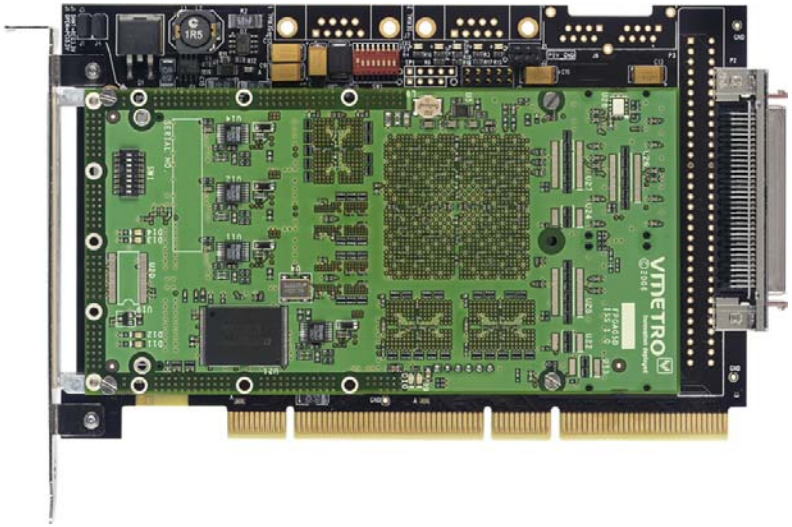


FPGA ParalleX Accelerator

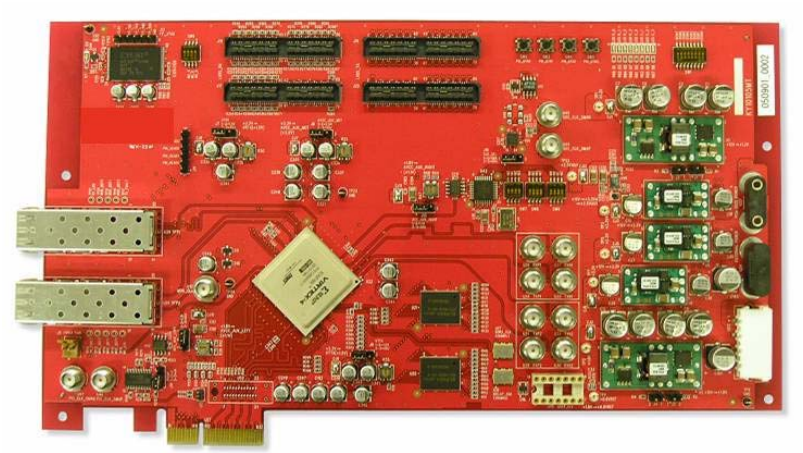
- Based on prior work performed on MIND architecture as part of Caltech/JPL Gilgamesh project
- Goal: enhance scalability and efficiency
  - Hide system wide latency
  - Reduce parallelism control overhead
- Design FPGA-based hardware drivers and co-processors to support ParalleX model
  - Parcel message-driven computation handler
  - Medium grained multithread execution scheduler
  - Global address translation support
  - Percolation pre-staging task manager
  - (possibly) local control object synchronization acceleration

FPGA attached boards: a new opportunity for advanced execution models





And a couple more:



Significant Changes to architecture over the next decade and beyond

- Realignment from processor oriented to memory oriented architectures
- Development and adoption of a new model of computation with abundant parallelism, intrinsic latency hiding, built-in reliability management, low overhead control mechanisms, and automatic load balancing
- Dominance of symbolic processing over numeric processing
- Replacement of conventional core processors with very fine grain elements
- Merger of logic and memory
- *Fixless* computers where maintenance is never performed and system automatically reconfigures for graceful degradation
- Multi-billion to trillion-way parallelism
- Self programming computers through goal directed
- New technologies beyond conventional semiconductors