From Prototyping to Emulation: The StarT (*T) Era (1992-1999)

Derek Chiou

(Dataflow-StarT-Synthesis Era occupant)

The University of Texas at Austin

Machine Building In CSG 1992-1999

- *T
  - 88110MP-based
- StarT-NG (Next Generation)
  - PowerPC 620-based
- StarT-Voyager
  - PowerPC 604-based
- StarT-X, StarT-Jr
  - x86 PCI-based
- Moving forward: RAMP

Dataflow Machines Looked Impractical

- Monsoon worked well, but
  - IBM RS/6000 donated at the same time was about as fast as 8 node Monsoon machine
- Could we leverage commercial processors?

*T: Integrated Building Blocks for Parallel Computing

Greg Papadopoulos, Andy Boughton, Robert Greiner, Michael J. Beckerle

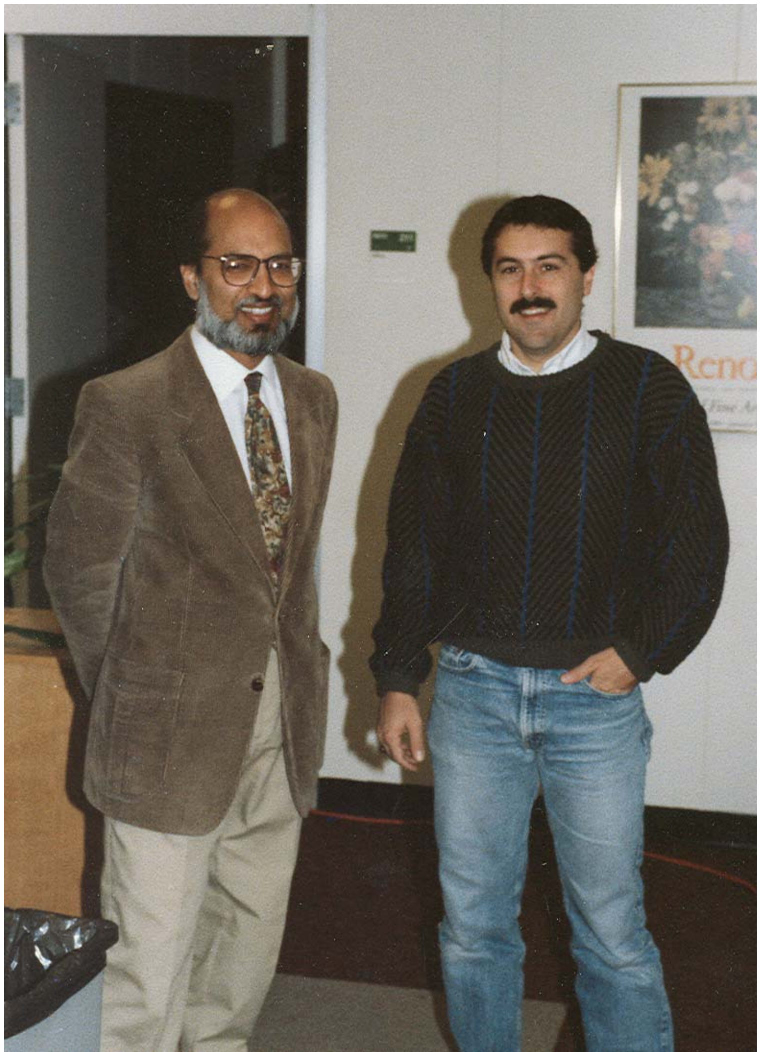
MIT and Motorola

*T: Motorola 88110MP

- Integrates NIU onto Motorola 88110 core
  - A functional unit
- Send/Receive instructions to access NIU
  - Use general-purpose registers
- Asymmetric message passing performance
  - Dual issue means 4 read ports, 2 write ports
- Motorola was doing the implementation
  - Many visits to Phoenix
- We grumbled
  - 6 cycles to send a message, 12 cycles to receive????
  - Monsoon was much better

And Then Arvind Has a Meeting

- And comes back with some news
- IBM/Motorola/Apple alliance
  - Out goes 88110
  - In comes PowerPC
- ***Re-*T?***
- PowerPC 620 selected as base processor
  - Not yet implemented, very aggressive 64b processor
- StarT-NG was born



StarT-NG: Delivering Seamless Parallel Computing

Derek Chiou, Boon S. Ang, Robert Greiner, Arvind,
James Hoe, Michael J. Beckerle, James E. Hicks,
and Andy Boughton

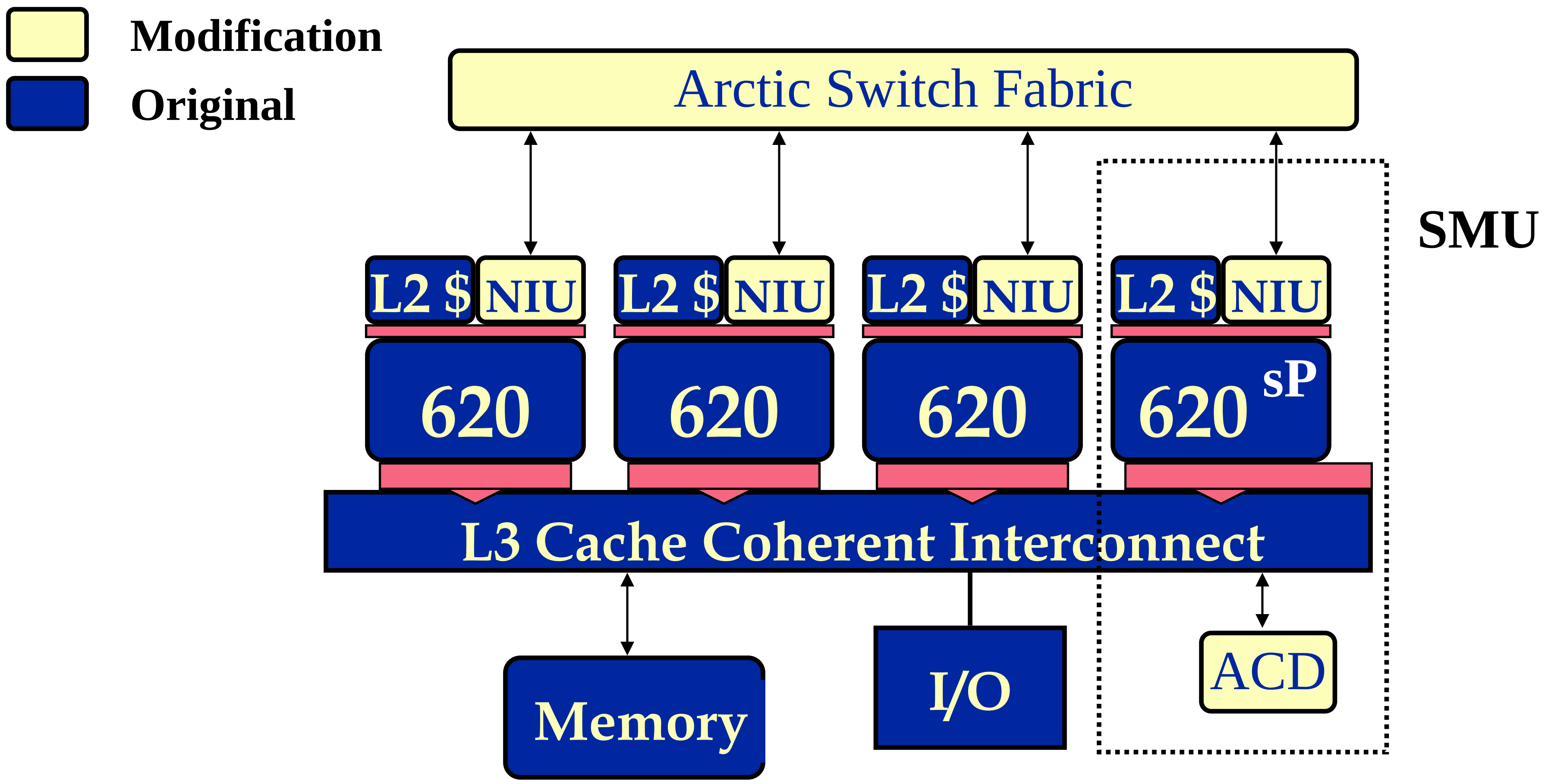
MIT and Motorola

http://www.csg.lcs.mit.edu:8001/StarT-NG

StarT-NG

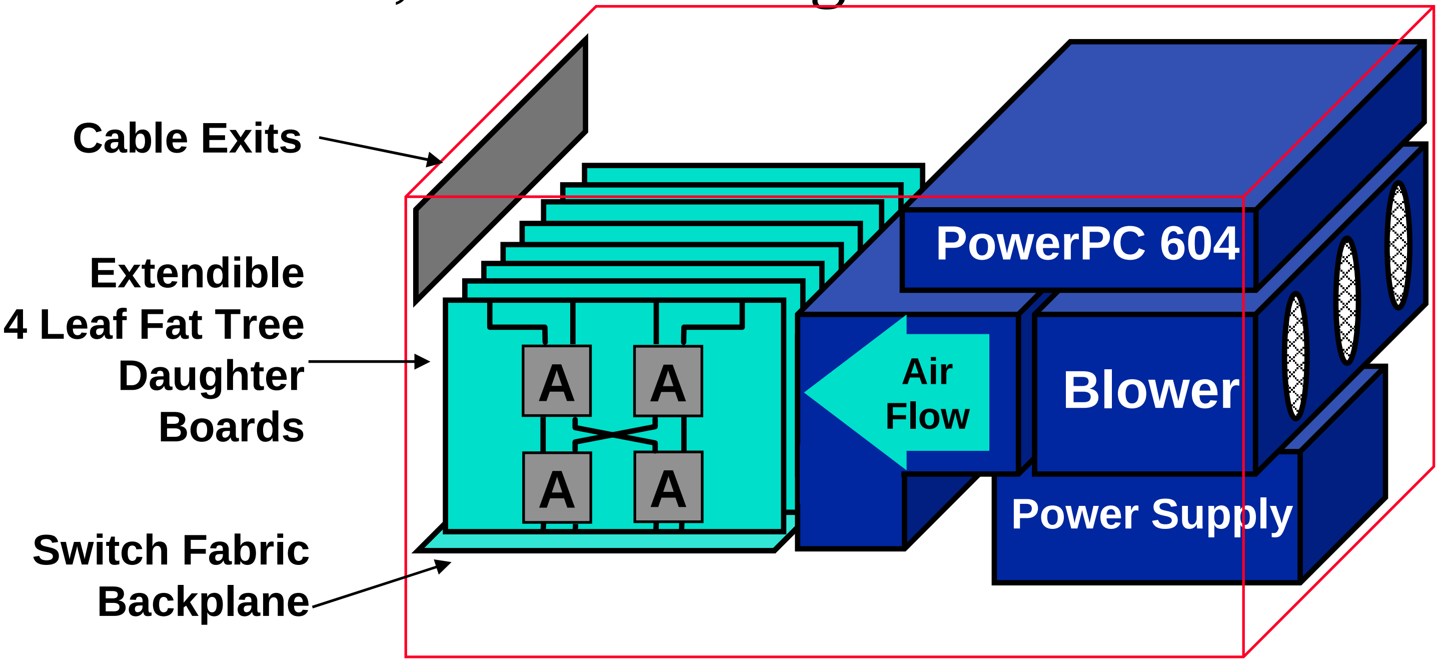
- A parallel machine providing
  - Low-latency, high-bandwidth message passing
    - Extremely low overhead
    - User-level
    - Time and space shared network
  - coherent shared memory test-bed
    - Software implemented, configurable
    - Extremely simple hardware
- Used aggressive, next-gen commercial systems
  - PowerPC 620-based SMPs
  - AIX 4.1

A StarT-NG Site

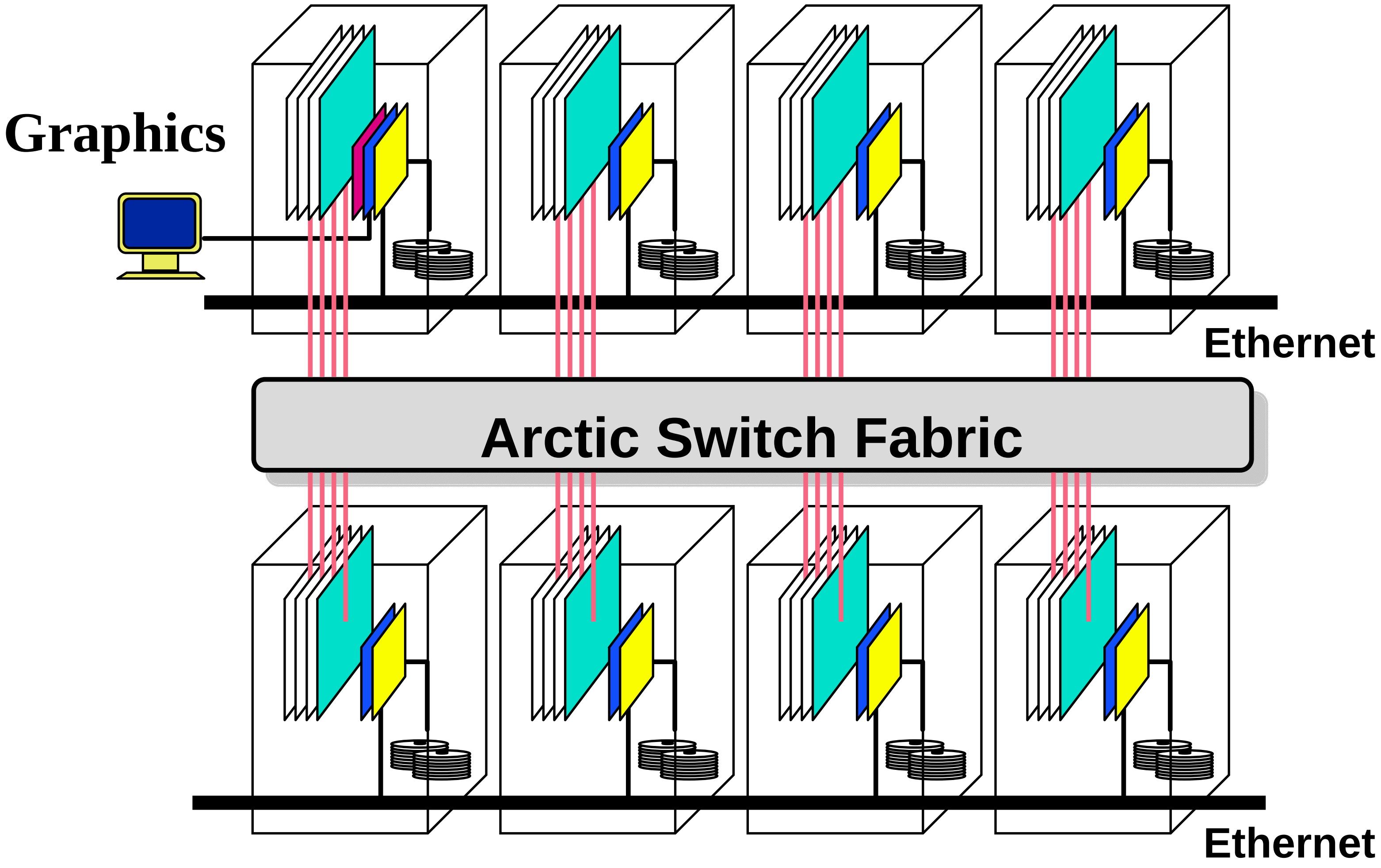


Arctic Switch Fabric

- 32-leaf full-bandwidth fat tree
  - 200MB/sec/direction
- Differential ECL links to endpoints
- Modular, scalable design



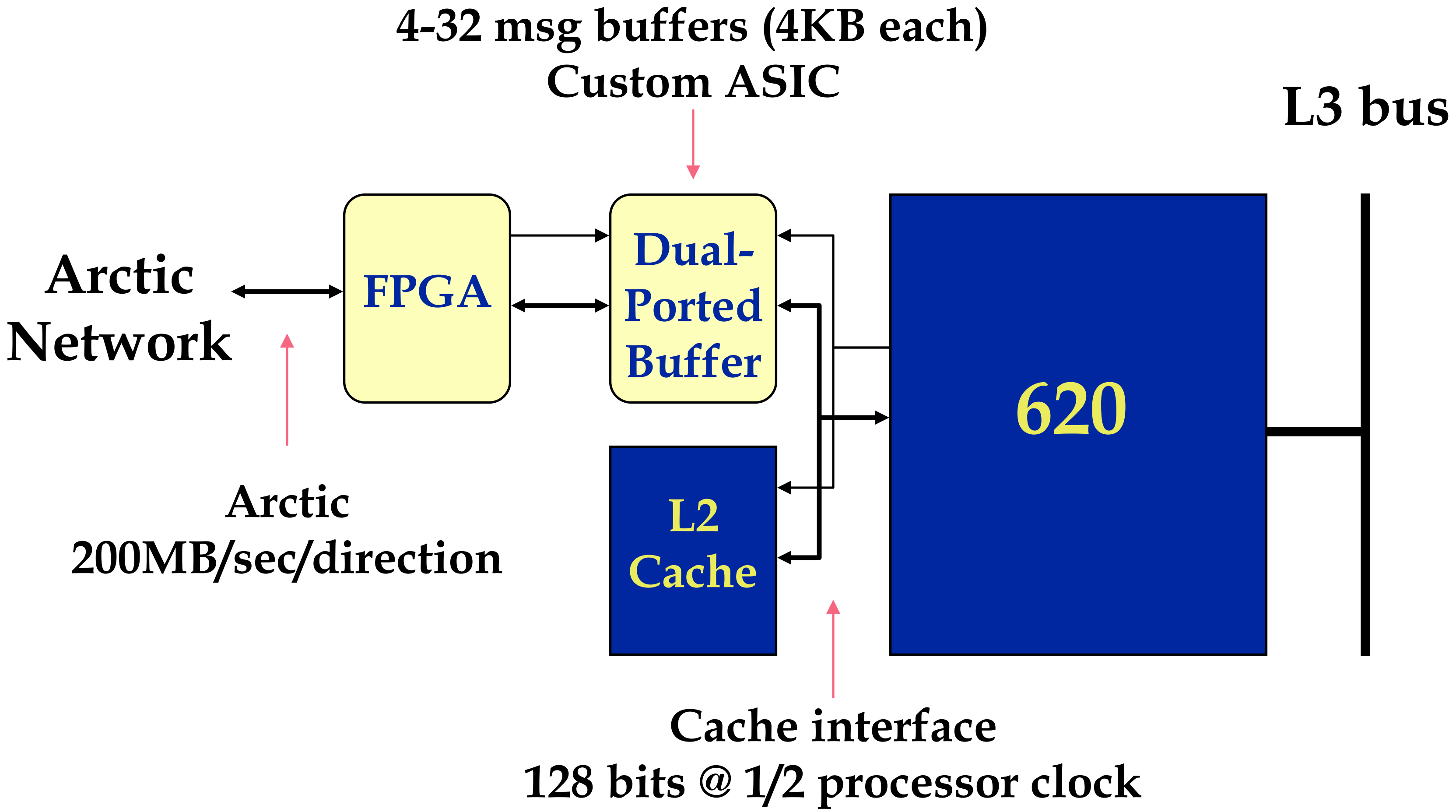
8-Site StarT-NG



Network Interface Unit (NIU)

- 620 provides a *coprocessor interface* to L2
  - accesses to specific region of memory go directly to L2 coprocessor
    - » bypass L2 cache interface
  - still cacheable within L1, if desired
- NIU attached to L2 coprocessor interface

NIU Implementation



Attempted Full Performance

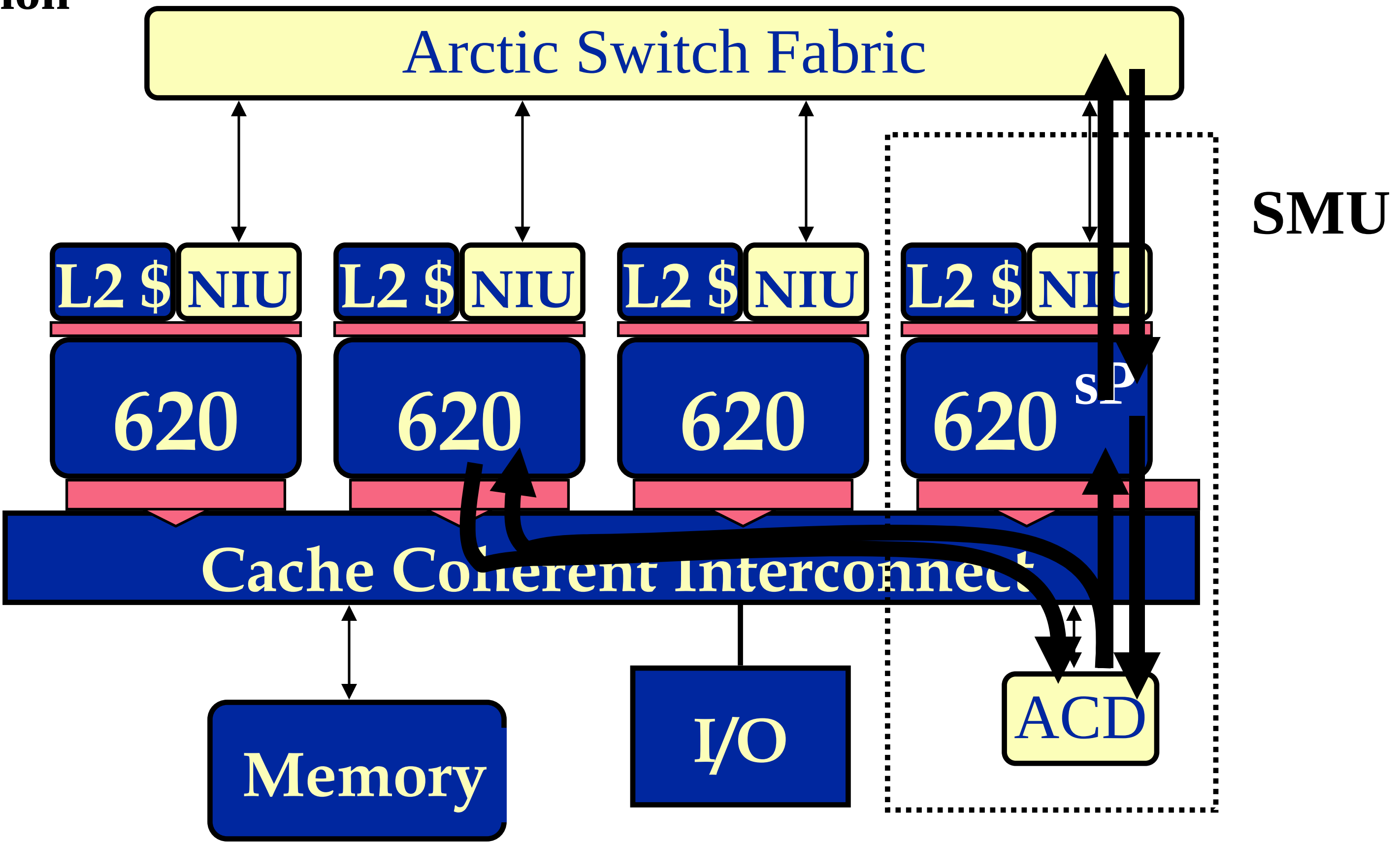
Address Capture Device (ACD)

- Allows an SMP 620 (**sP**) to service bus ops
  - Support shared memory
- ACD is simple hardware on L3 bus
  - “captures” global memory bus transactions
- sP communicates with ACD over L3 bus
  - Reads captured accesses to global address
  - Services requests using message passing
  - Writes back returned cache-lines to ACD
  - depends on out-of-order 620 bus
- If not needed, sP becomes an aP

ACD Example

🟨 Modification

🟦 Original



Status *(from EuroPar 95 talk)*

- Hardware & Software design completed
  - implementations in progress
- Hardware will be available soon after the 620 SMP is available

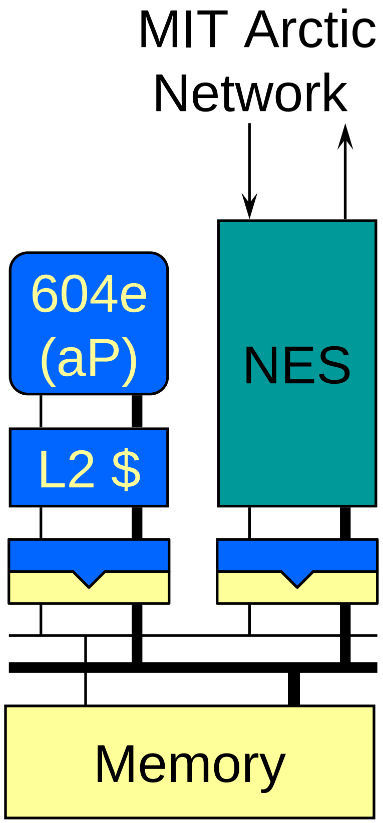
Then, in 1996, Arvind has a meeting

- PowerPC 620 indefinitely delayed
  - Look for another processor
- Lesson to current grad students
  - Don’t let Arvind go to meetings
- PowerPC 604e chosen
  - Available off the shelf

The StarT-Voyager Parallel System

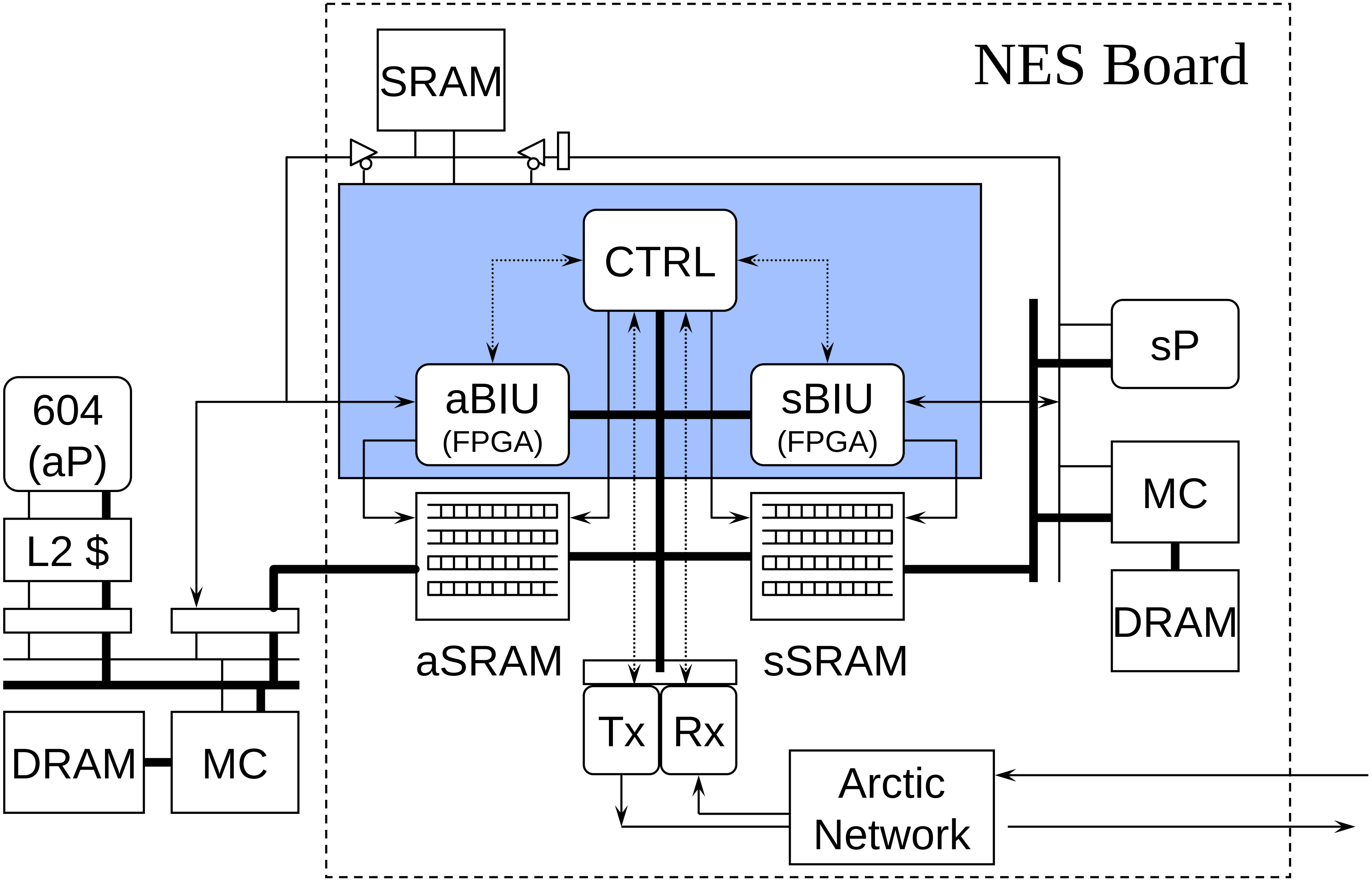
Derek Chiou, Boon S. Ang, Dan Rosenband,
Mike Ehrlich, Larry Rudolph, Arvind,
MIT Laboratory for Computer Science

StarT-Voyager

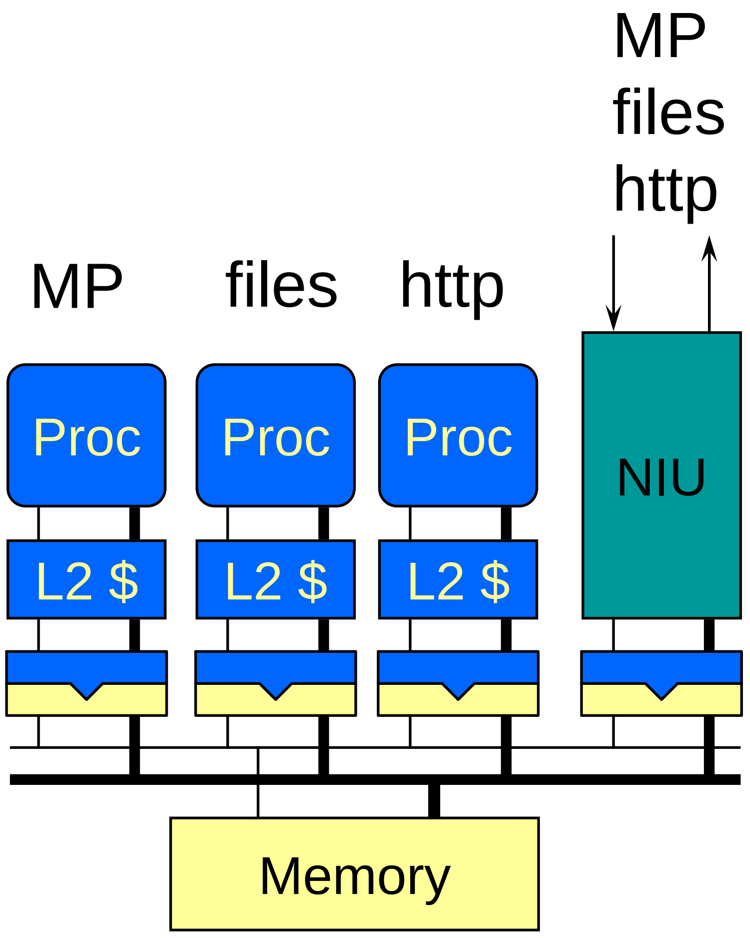


- Scalable SMP cluster
  - IBM 604e-based SMP building blocks
  - Custom Network Endpoint Subsystem (NES) connects SMP to network via memory bus
- Intended Research
  - network sharing
  - communication mechanisms
  - architecture
  - system and application software

Network Endpoint Subsystem

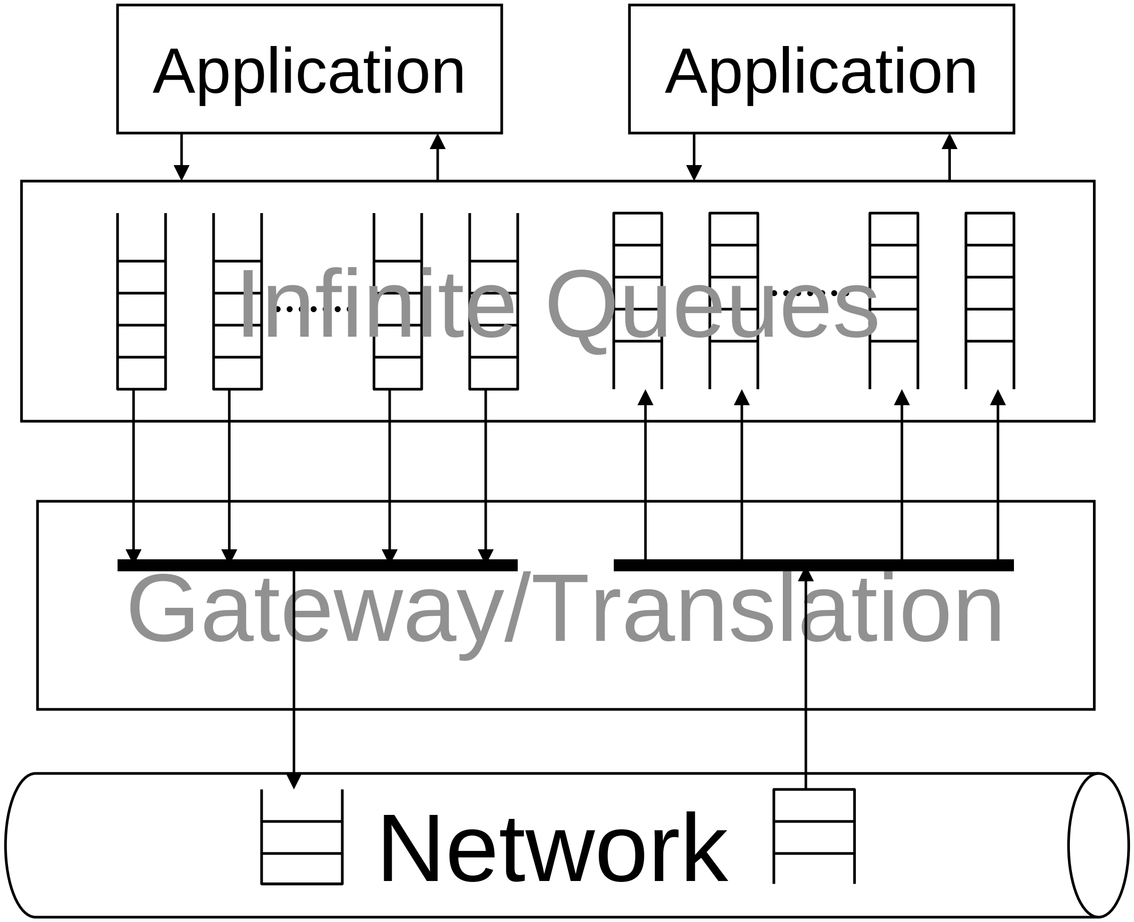


Why Share Network?

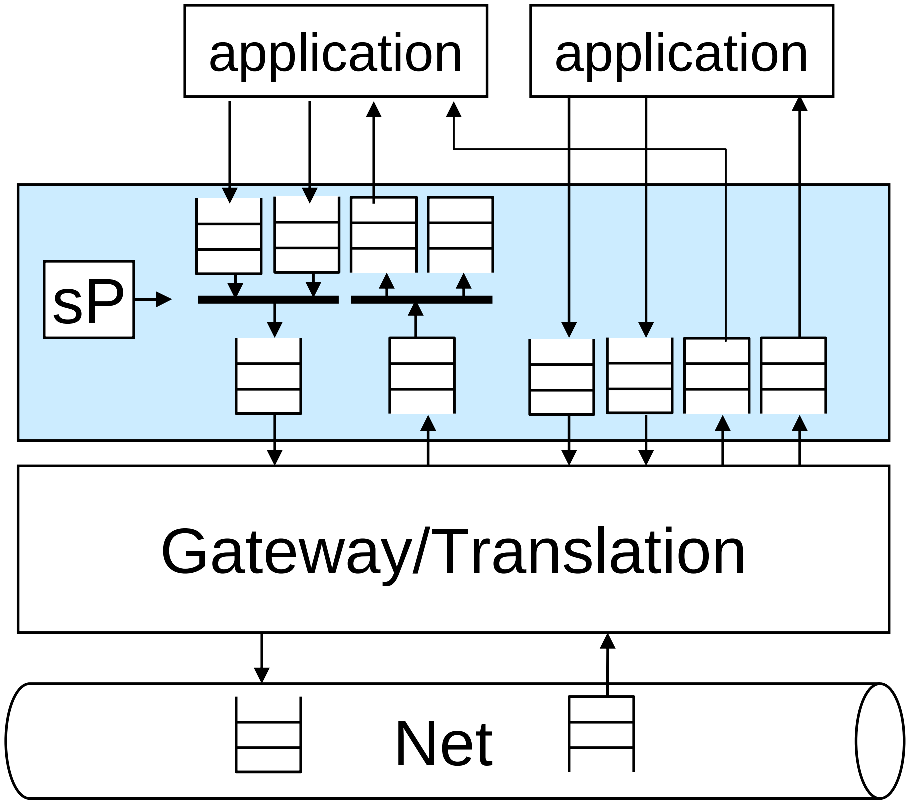


- Single network
- Different Services
  - – message passing (MP)
  - – coherence protocol
  - – file system….
- Multiple processors/node
  - – multiple network jobs
  - – multiple services/processor

StarT-Voyager Network Sharing

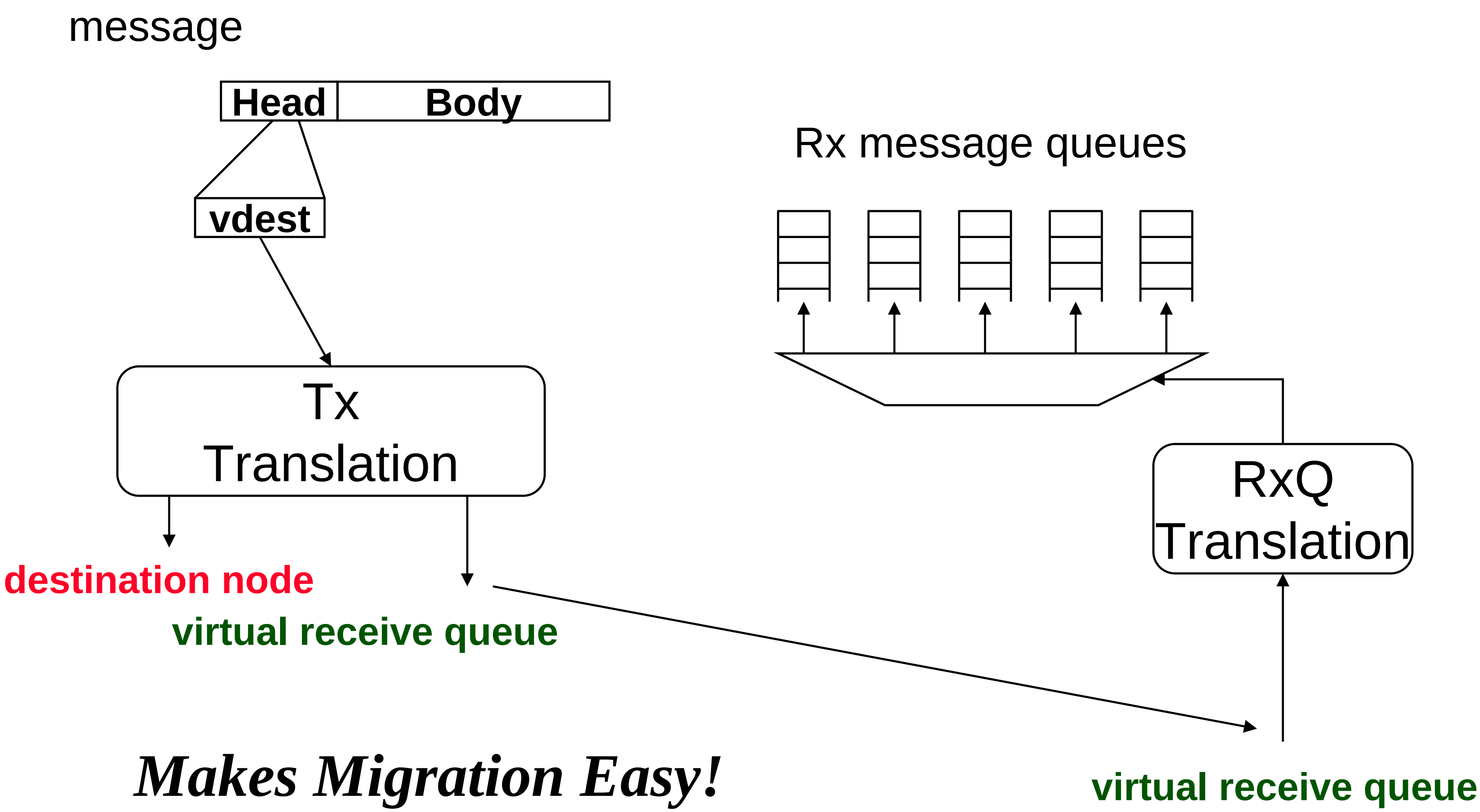


Multiple Queues



- Fixed number hdw queues
- Service Processor (sP) emulates infinite queues
  - sP controls/uses NES
- Critical queues use hdw queues (resident), others emulated by sP (non-resident)
- Application oblivious
  - switch queues without app knowledge or support (VM)
- Synchronization
- Flow control

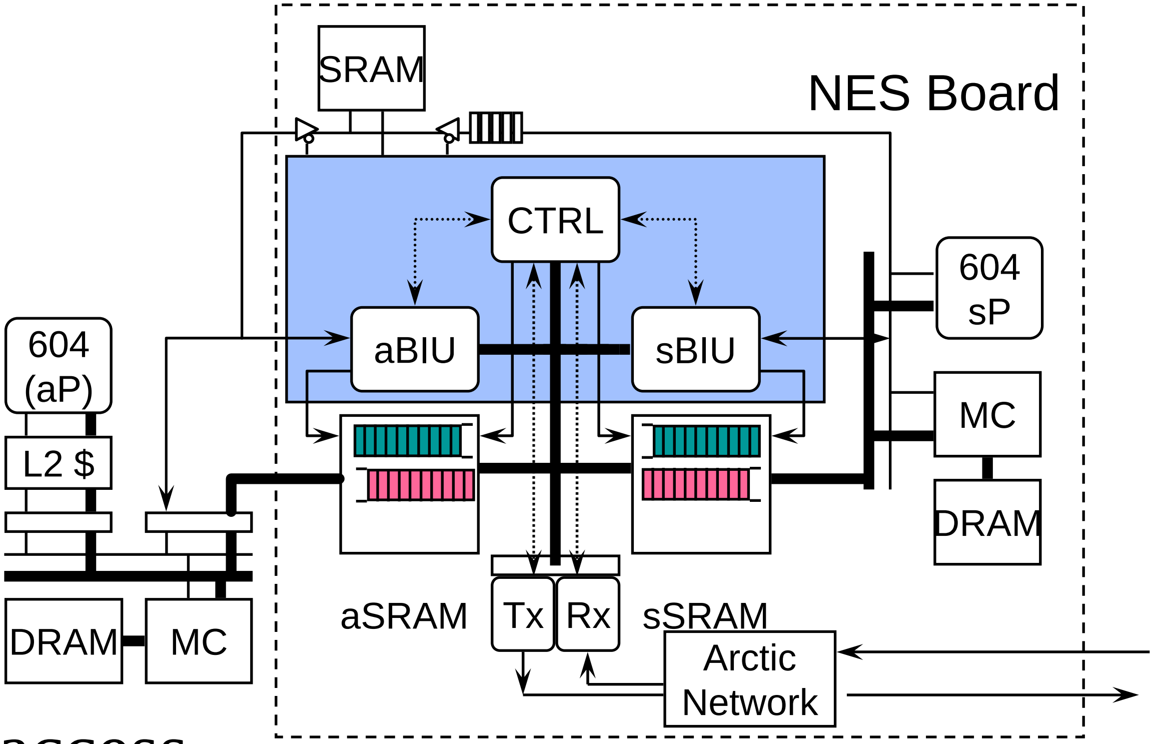
Virtualized Destination



Makes Migration Easy!

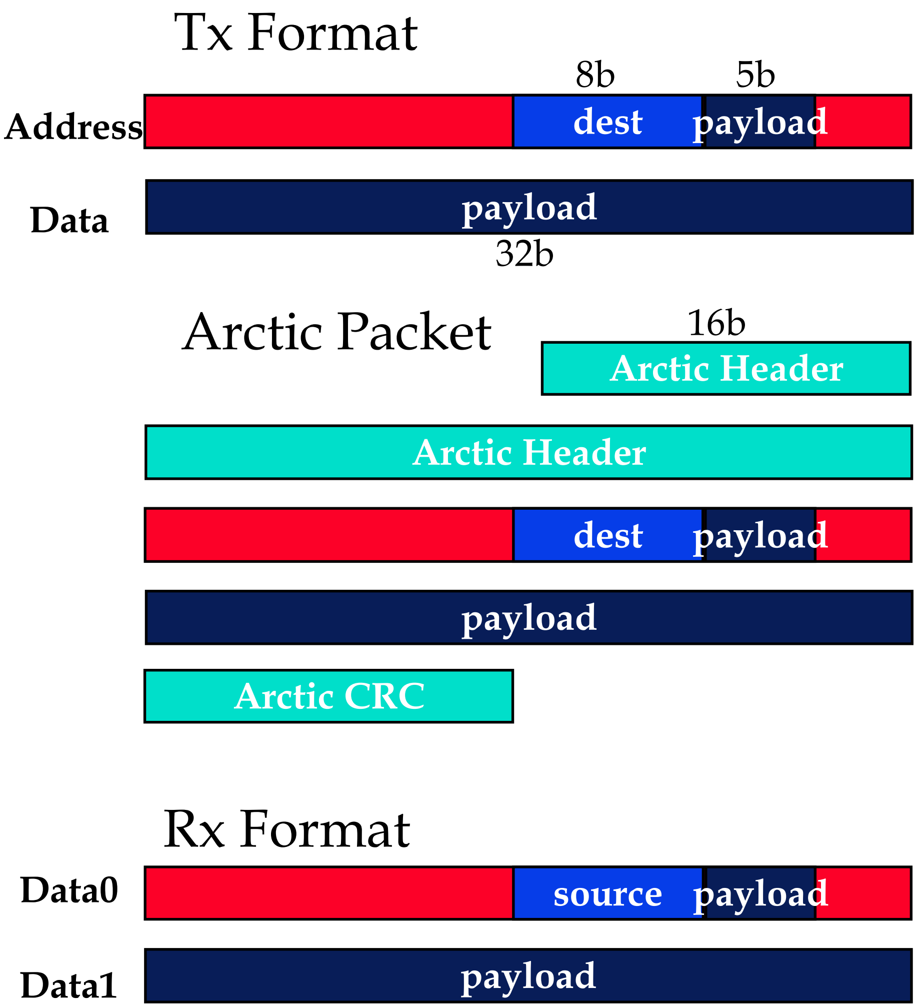
Memory with Weird Semantics: Message Passing Mechanisms

- Four mechanisms
  - *Basic Message*
  - *Express Message*
  - *Tag-on Message*
  - *DMA*
- 512 msg queues
  - *16 resident*
- Protected user-level access
  - *Multi-tasking (space / time)*
  - *No strict gang scheduling required*



Express Messages

- For small messages, e.g. Acks:
  - Payload: 32 + 5 bits
- Uncached access to message queues
- Advantages:
  - Avoid weak memory model's SYNC
  - No coherence maintenance for msg queue space



Tx Format

8b 5b

Address | dest | payload

Data | payload

32b

Arctic Packet

16b

Arctic Header

Arctic Header

dest | payload

payload

Arctic CRC

Rx Format

Data0 | source | payload

Data1 | payload

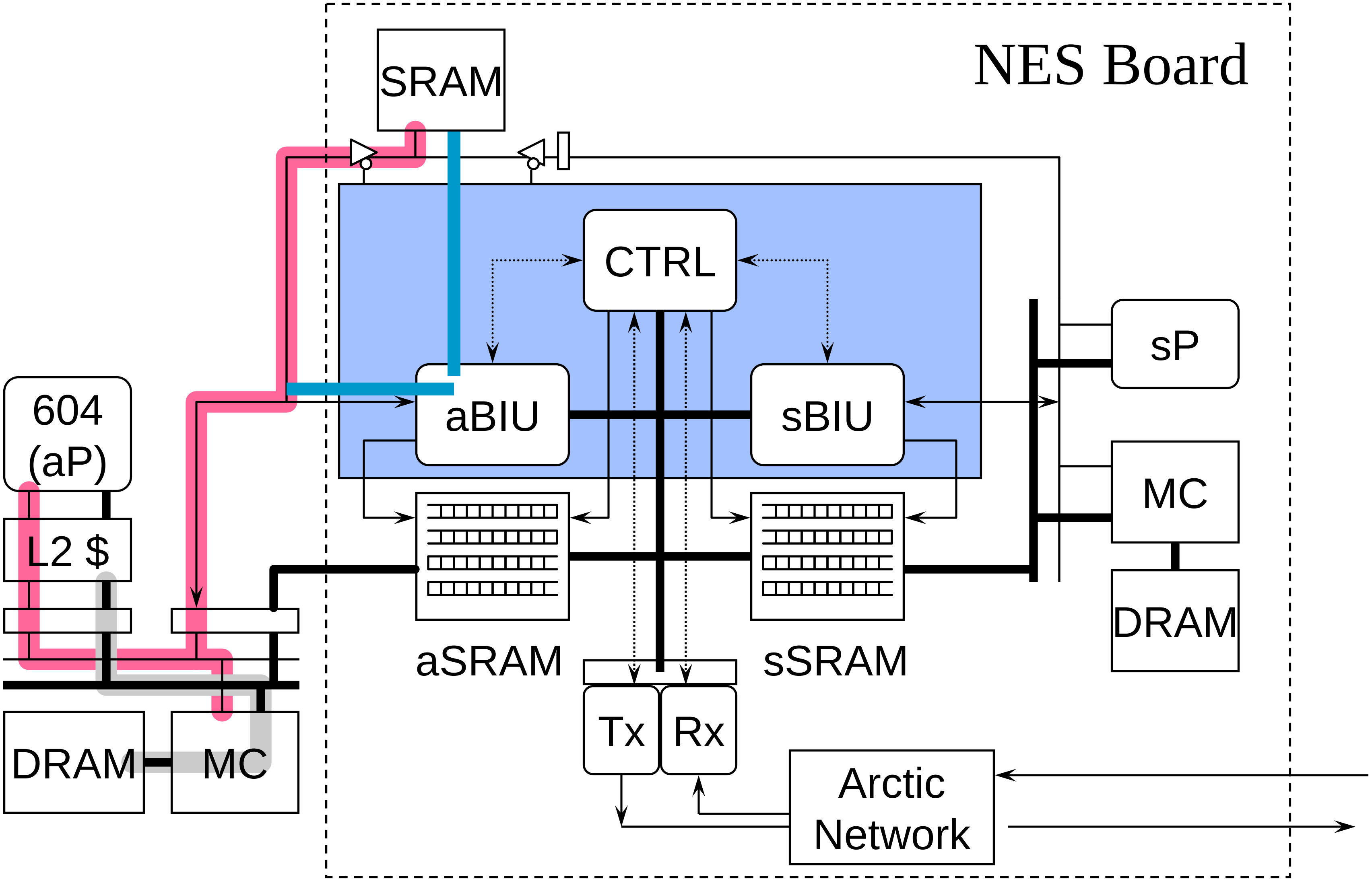
S-COMA Shared Memory

- Global mem mapped to local physical mem
  - Page granularity allocation
  - cache-line granularity protection
- Accesses to global mem snooped by NES
  - legal access completes against local RAM
  - illegal access passed to sP for servicing
    - » aP bus operation retried until sP fixes

S-COMA Hardware Support

- NES hdw snoops part of physical memory
- F(Bus Operation, HAL State) -> Action
  - – Proceed
  - – Proceed & Forward to sP
  - – Retry & Forward to sP
- sP only entity that can modify HAL state
  - – simplicity at slight restriction on functionality

Accessing S-COMA Memory



Implementation

- It worked!

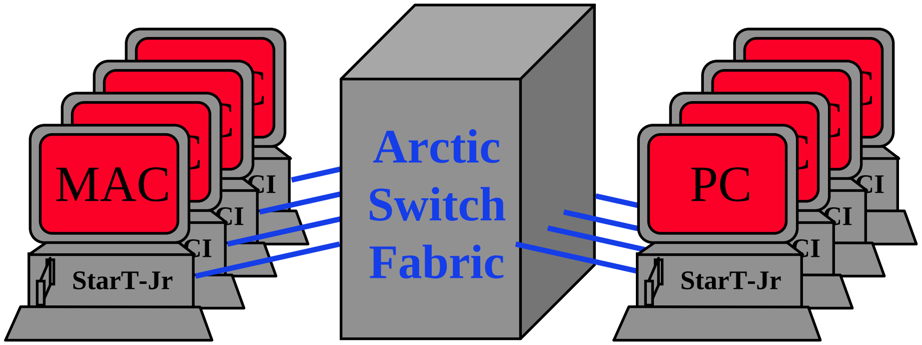
- NESChip implemented in Chip Express technology
  - laser-cut gate array prototyping (1 week)
- TxU/RxU implemented in FPGA’s
- Buffers implemented by dual-ported SRAM’s and FIFO’s
- Implemented by students and staff

StarT-X/StarT-Jr

James Hoe, Mike Ehrlich

James Hoe et al, SC99

StarT-X: A Real Success



Heterogeneous Network of Workstations

StarT-X PCI-Arctic network interface

Integrated network processor

James Hoe et al, SC99

StarT-Hyades Cluster

- Our system
  - 16 2-way Pentium-II SMPs running Linux
  - Fast Ethernet (LAN)
  - Even faster system area network (StarT-X)
  - Owned by a single research group

- Application: MITgcmUV
  - Coupled atmosphere and ocean simulation for climate research
  - Traditionally relied on shared Big Irons

James Hoe et al, SC99

Application Performance

Processor Count	Machine	Sustained Performance (Gflop/s)	Normalized Performance 1-proc C90
1	Hyades	0.054	<0.1
16	Hyades	0.9	1.5
32	Hyades	1.8	3.0
1	Cray C90	0.6	1.0
4	Cray C90	2.2	3.7
1	NEC-SX4	0.7	1.2
4	NEC-SX4	2.7	4.5

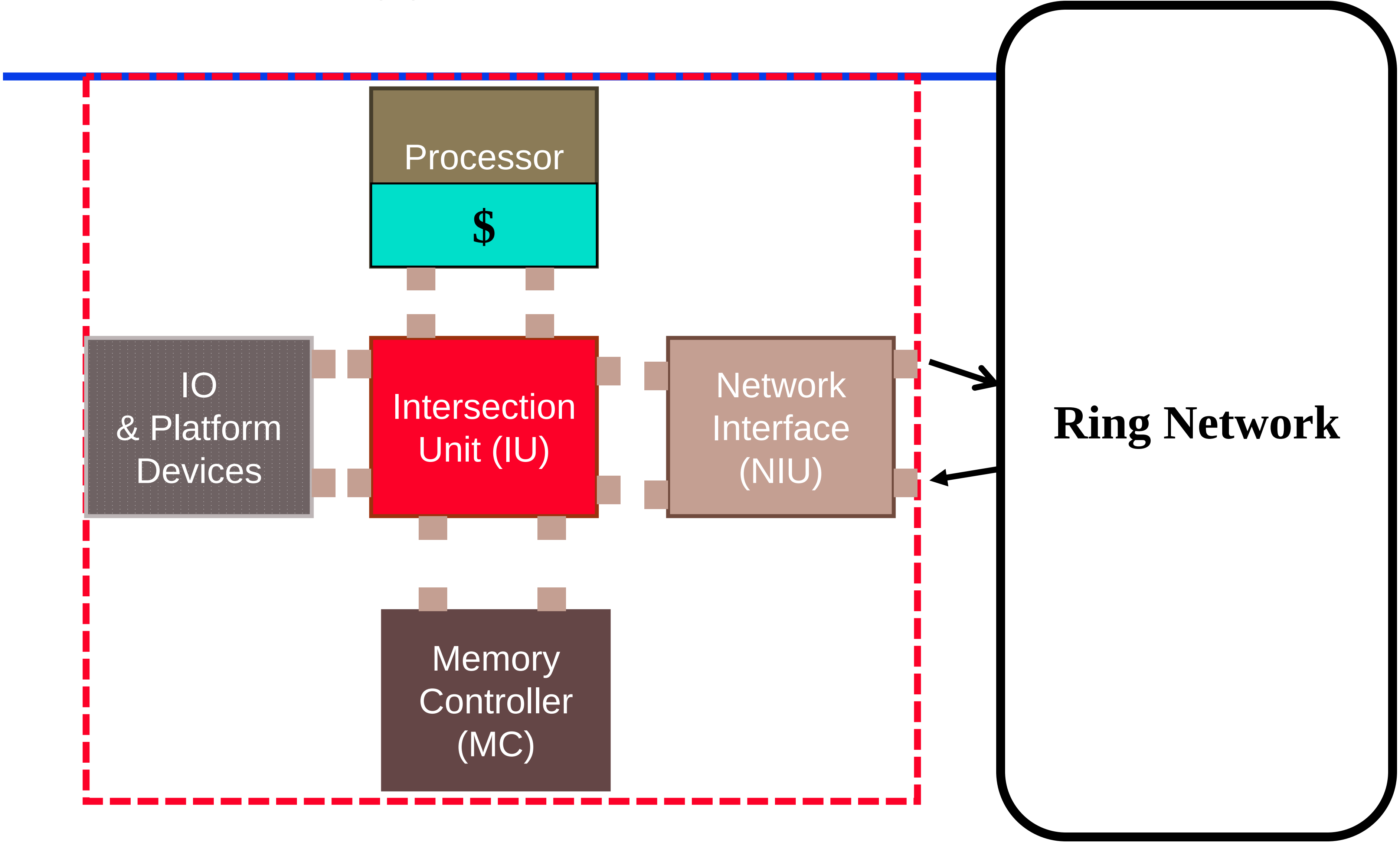
Modern Day: RAMP: MPP on FPGAs

- Goal 1000-CPU system for $100K early next year
  - *Not intended to be prototype*
- ≈ 16 CPUs will fit in Field Programmable Gate Array (FPGA)
  - Need about 64 FPGAs
  - ≈ 8 32-bit simple “soft core” RISC at 100MHz in 2004 (Virtex-II)
- HW research community shares logic design (“gate shareware”) to create out-of-the-box, MPP
  - Use off-the-shelf processor IP (simple processors, ~150MHz)
  - RAMPants: **Arvind (MIT)**, Krste Asanovíc (MIT), Derek Chiou (Texas), James Hoe (CMU), Christos Kozyrakis (Stanford), Shih-Lien Lu (Intel), Mark Oskin (Washington), David Patterson (Berkeley, Co-PI), Jan Rabaey (Berkeley), and John Wawrzynek (Berkeley, PI)
- “Research Accelerator for Multiple Processors”

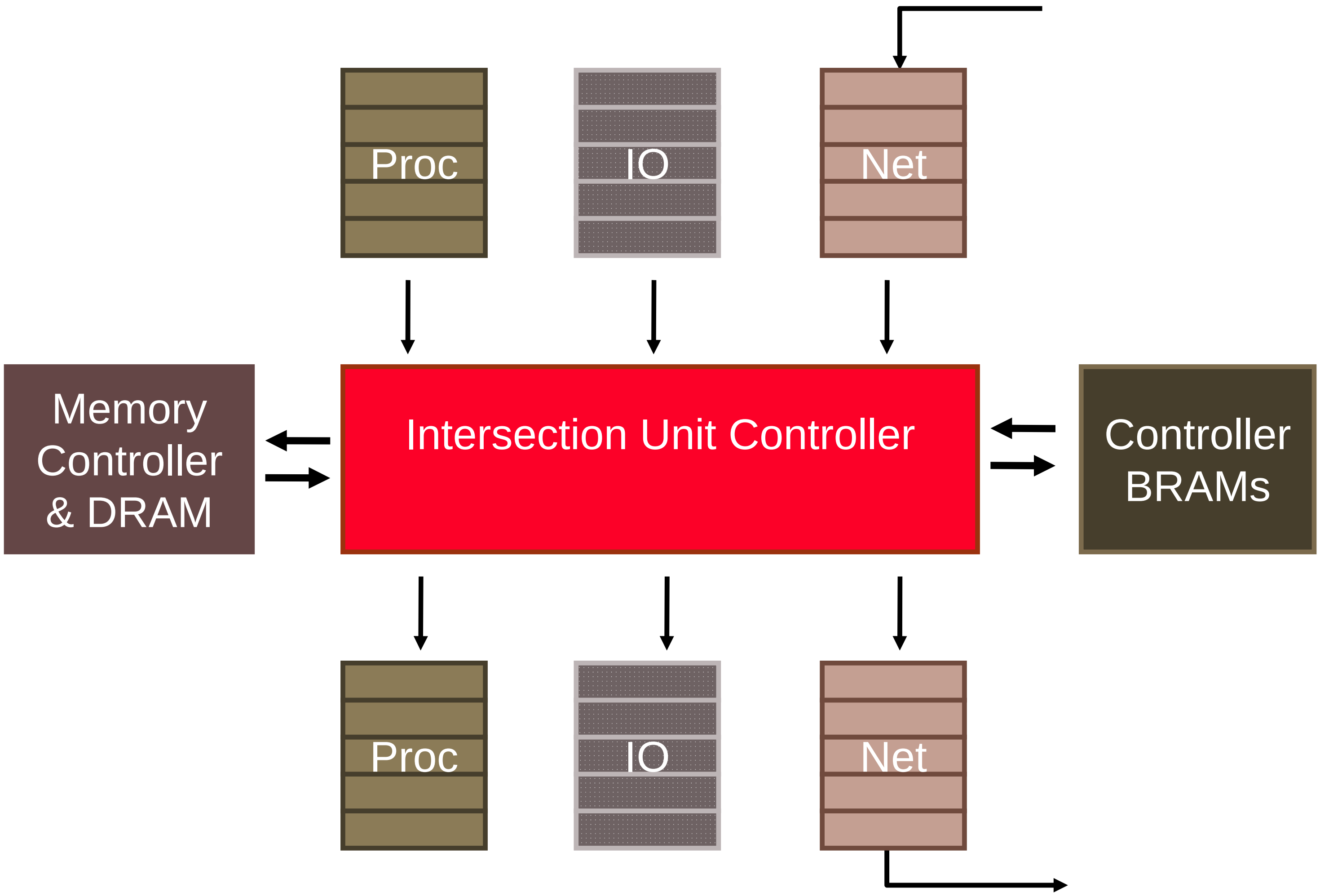
RAMP-White Reference Platform

- Very flexible shared memory platform
  - Different components/policies/parameters
- Uses StarT-Voyager-like bus retry
- 3 Phase Approach:
  - Phase 1: Incoherent global shared memory
    - All accesses to main memory
    - No caches
  - Phase 2: Snoopy-based coherency over a ring
    - Adds coherent cache
  - Phase 3: Directory-based coherency over network
    - Adds directory

RAMP-White Phases



Intersection Unit (in Bluespec)



Conclusions

- Ideas recycle
  - RAMP-White ≈ StarT-Voyager
- Don’t be too implementation-ambitious
  - Matching industry is impossible
  - Balance between implementation effort and accuracy
- Delicate balance between rolling your own and depending on others
  - Reuse whatever you can (Arctic)
- Thanks Arvind!
  - Using what I learned in grad school daily
  - Bluespec