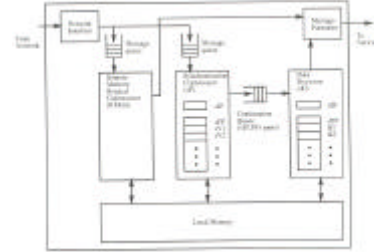


General Purpose Node-to-Network Interface in Scalable Multiprocessors

CS 258, Spring 99
David E. Culler
Computer Science Division
U.C. Berkeley

*T: Network Co-Processor

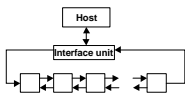


3/12/99

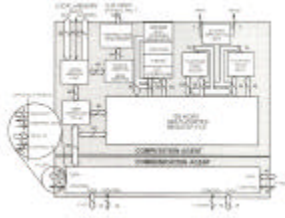
CS258 S99

2

iWARP: Systolic Computation



- Nodes integrate communication with computation on systolic basis
- Msg data direct to register
- Stream into memo

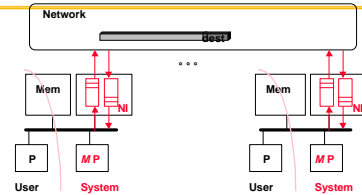


3/12/99

CS258 S99

3

Dedicated Message Processor



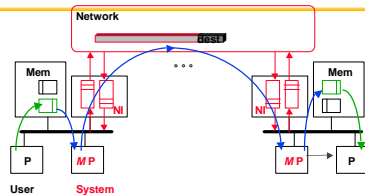
- General Purpose processor performs arbitrary output processing (at system level)
- General Purpose processor interprets incoming network transactions (at system level)
- User Processor <-> Msg Processor share memory
- Msg Processor <-> Msg Processor via system network transaction

3/12/99

CS258 S99

4

Levels of Network Transaction



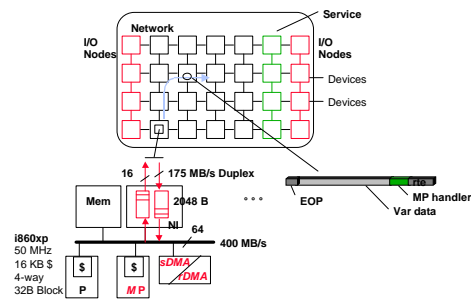
- User Processor stores cmd / msg / data into shared output queue
 - must still check for output queue full (or make elastic)
- Communication assists make transaction happen
 - checking, translation, scheduling, transport, interpretation
- Effect observed on destination address space and/or events
- Protocol divided between two layers

3/12/99

CS258 S99

5

Example: Intel Paragon

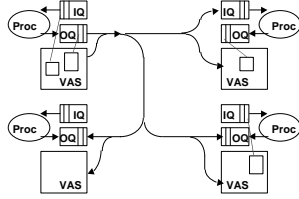


3/12/99

CS258 S99

6

User Level Abstraction (Lok Liu)



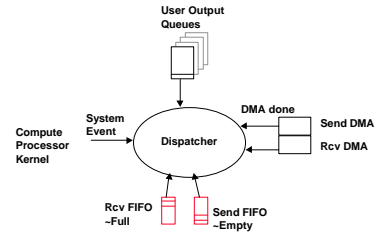
- Any user process can post a transaction for any other in protection domain
 - communication layer moves $OQ_{src} \rightarrow IQ_{dest}$
 - may involve indirection: $VAS_{src} \rightarrow VAS_{dest}$

3/12/99

CS258 S99

7

Msg Processor Events

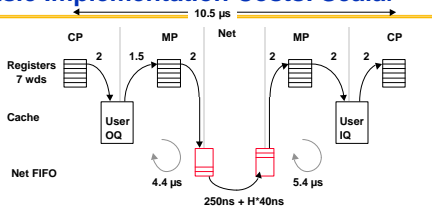


3/12/99

CS258 S99

8

Basic Implementation Costs: Scalar



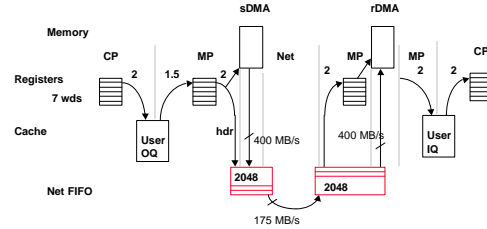
- Cache-to-cache transfer (two 32B lines, quad word ops)
 - producer: read(miss,S), chk, write(S,WT), write(L,WT),write(S,WT)
 - consumer: read(miss,S), chk, read(H), read(miss,S), read(H),write(S,WT)
- to NI FIFO: read status, chk, write, . . .
- from NI FIFO: read status, chk, dispatch, read, read, . . .

3/12/99

CS258 S99

9

Virtual DMA -> Virtual DMA



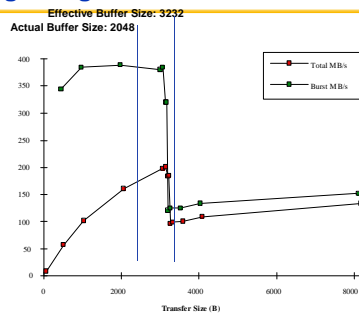
- Send MP segments into 8K pages and does VA -> PA
- Rcv MP reassembles, does dispatch and VA -> PA per page

3/12/99

CS258 S99

10

Single Page Transfer Rate

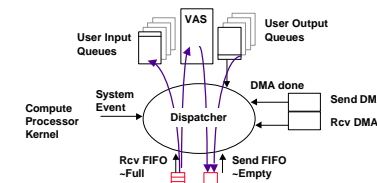


3/12/99

CS258 S99

11

Msg Processor Assessment



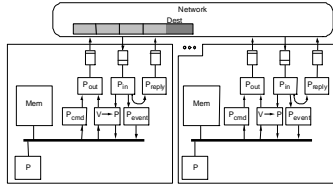
- Concurrency Intensive
 - Need to keep inbound flows moving while outbound flows stalled
 - Large transfers segmented
- Reduces overhead but adds latency

3/12/99

CS258 S99

12

Case Study: Meiko CS2 Concept



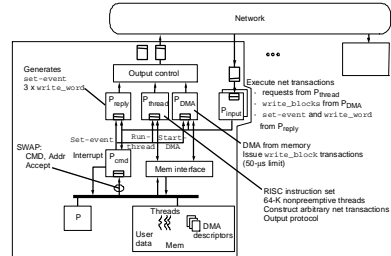
- **Circuit-switched Network Transaction**
 - source-dest circuit held open for request response
 - limited cmd set executed directly on NI
- **Dedicated communication processor for each step in flow**

3/12/99

CS258 S99

13

Case Study: Meiko CS2 Organization

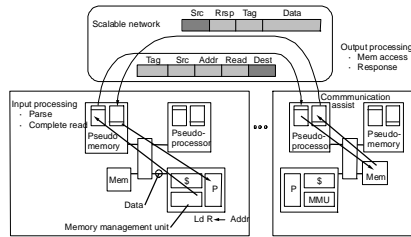


3/12/99

CS258 S99

14

Shared Physical Address Space

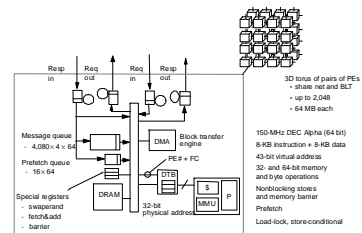


- **NI emulates memory controller at source**
- **NI emulates processor at dest**

3/12/99 - must be deadlock free CS258 S99

15

Case Study: Cray T3D



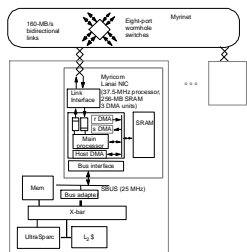
- **Build up info in 'shell'**
- **Remote memory operations encoded in address**

3/12/99

CS258 S99

16

Case Study: NOW



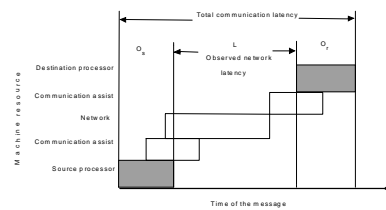
- **General purpose processor embedded in NIC**

3/12/99

CS258 S99

17

Message Time Breakdown



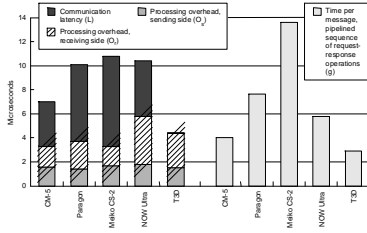
- **Communication pipeline**

3/12/99

CS258 S99

18

Message Time Comparison

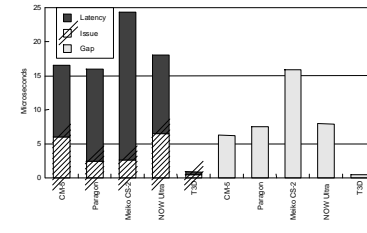


3/12/99

CS258 S99

19

SAS Time Comparison

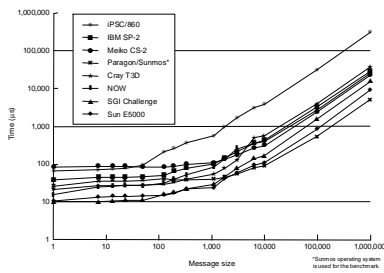


3/12/99

CS258 S99

20

Message-Passing Time vs Size

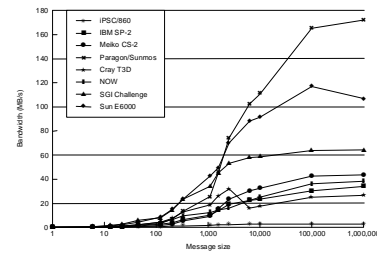


3/12/99

CS258 S99

21

Message-Passing Bandwidth vs Size

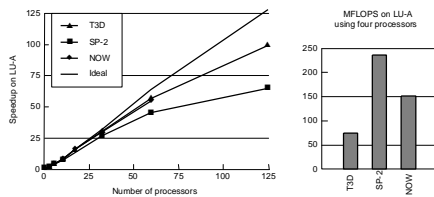


3/12/99

CS258 S99

22

Application Performance on LU

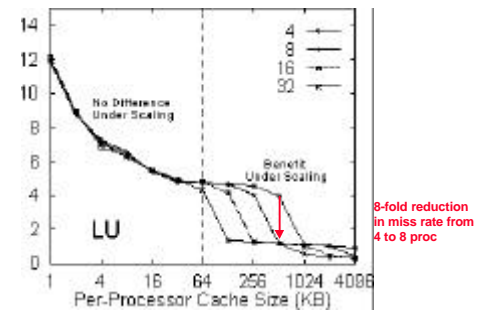


3/12/99

CS258 S99

23

Working Sets Change with P

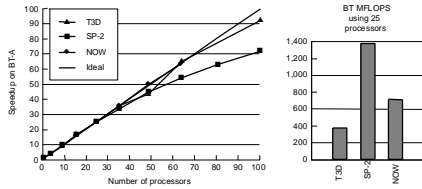


3/12/99

CS258 S99

24

Application Performance on BT

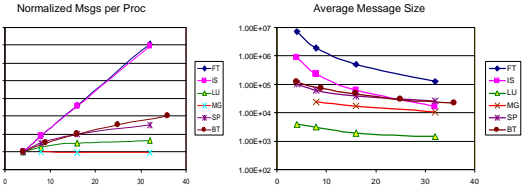


3/12/99

CS258 S99

25

NAS Communication Scaling

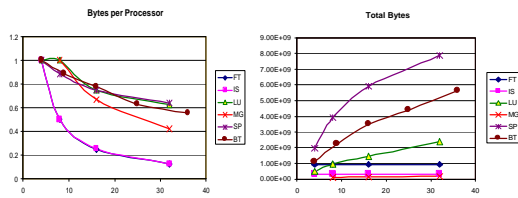


3/12/99

CS258 S99

26

NAS Communication Scaling: Volume



3/12/99

CS258 S99

27

Communication Characteristics: BT

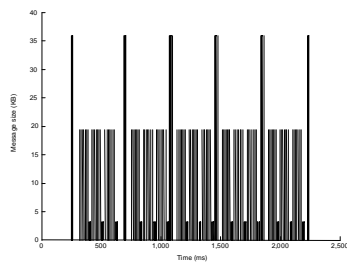
	4 Processors		16 Processors		32 Processors		64 Processors					
	Msg Size (KB)	Data Xfer (KB)	Msg Size (KB)	Data Xfer (KB)	Msg Size (KB)	Data Xfer (KB)	Msg Size (KB)	Data Xfer (KB)				
	43.5	12	513	114	1,652	4.5	540	2,505	3	1,344	4,266	
	81.5	24	1,916	61	96	5,742	29	540	15,425	19	1,344	25,266
	261	12	3,062	69	144	9,738	45	216	9,545	35.5	384	13,406
Comm Vol (KB)			5,491		17,132		27,475					42,538
Sec per iter			5.43		1.46		0.67					0.38
Ave BW (MB/s)			1		11.5		40					110.3
MB/s per Proc			0.25		0.72		1.11					1.72

3/12/99

CS258 S99

28

Beware Average BW analysis

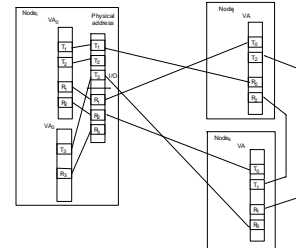


3/12/99

CS258 S99

29

Reflective Memory



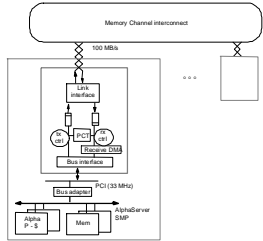
- Writes to local region reflected to remote

3/12/99

CS258 S99

30

Case Study: DEC Memory Channel



- See also Shrimp

3/12/99

CS258 S99

31

Scalable Synchronization Operations

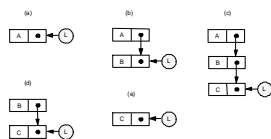
- Messages: point-to-point synchronization
- Build all-to-all as trees
- Recall: sophisticated locks reduced contention by spinning on separate locations
 - caching brought them local
 - test&test&set, ticket-lock, array lock
 - » O(p) space
- Problem: with array lock location determined by arrival order => not likely to be local
- Solution: queue-lock
 - build distributed linked-list, each spins on local node

3/12/99

CS258 S99

32

Queue Locks



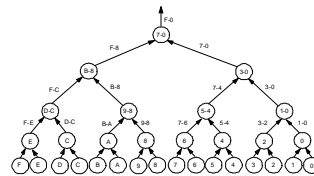
- Head holds lock; Each points to next waiter
- Shared pointer to tail
- Acquire
 - swap (fetch&store) tail with node address, chain in prev
- Release
 - signal next
 - compare&swap plus check to reset tail

3/12/99

CS258 S99

33

Parallel Prefix: Upward Sweep



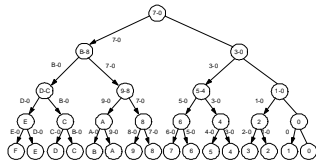
- generalization of barrier (reduce-broadcast)
- compute $S_i = X_i + X_{i-1} + \dots + X_0$, for $i = 0, 1, \dots$
- combine children, store least significant

3/12/99

CS258 S99

34

Downward Sweep of parallel Prefix



- Least branch send to most sig child
- when receive from above
 - send to least significant
 - combine with stored and send result to most sign

3/12/99

CS258 S99

35