

An Overview of Myrinet and Cluster Computing Message Libraries

Craig Ulmer

<http://www.ece.gatech.edu/~grimace>

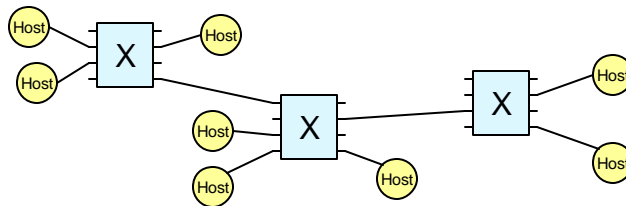


Outline

- **Myricom's Myrinet**
 - Hardware description
 - Utilization Techniques
- **Myrinet Communication Libraries**
 - AM, FM, BIP, VMMC, GM
- **GRIM**
 - Resource rich clusters
 - Low level mechanisms

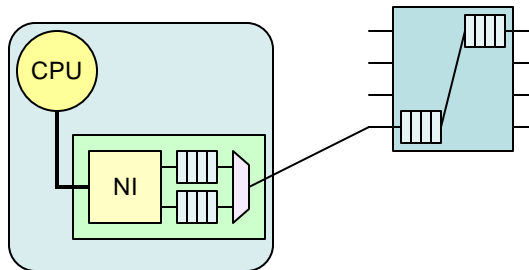
Myricom Myrinet

- Commercialized version of Cal Tech Mosaic multicomputer network hardware
 - Wormhole switches, source routing
 - High-speed, Ultra-reliable network
 - Configurable topology



Custom Network

- Simplify internals, move guts to hosts
 - Switch cut-through latency ~ 300 ns (16x16)
 - User writes NI firmware to add net functionality

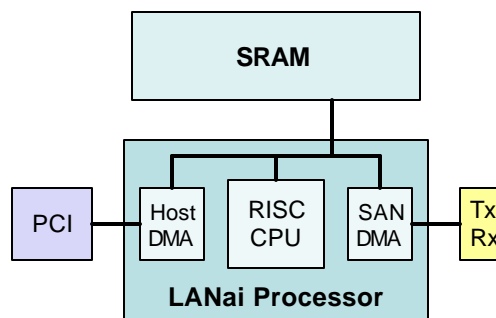


Myrinet NI Hardware

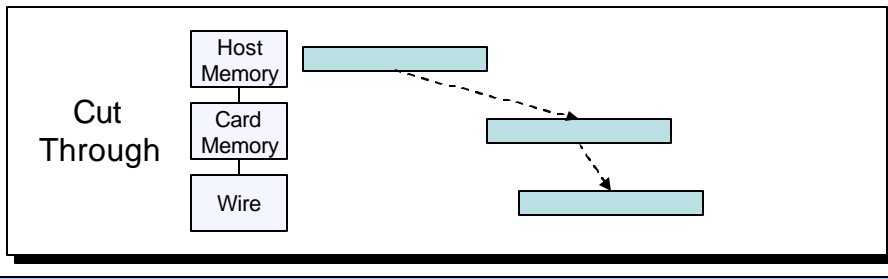
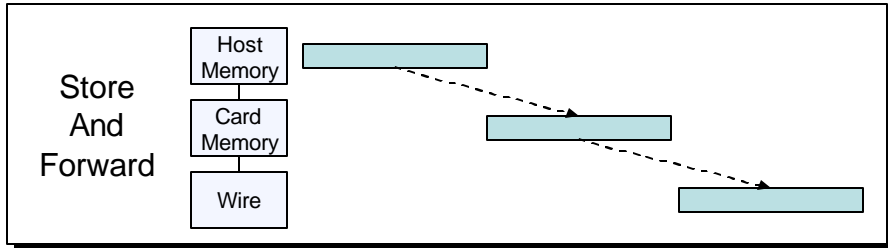
NI	Year	1-Way Link Speed	NI Speed	NI Memory	Host I/O
LANai 3	1994	640 Mbps	25 MHz ?	128KB/ 256KB	20 MHz S-Bus
LANai 4	1996	1.28 Gbps	33 MHz	1MB	32b/33 MHz PCI
LANai 9	2000	2.0 Gbps	133-200 MHz	2-8MB	64b/66 MHz PCI

Myrinet NI Architecture

- RISC CPU
 - Big endian
- SRAM
 - No CPU cache
- DMA Engines
 - PCI / SRAM
 - SRAM / Tx
 - Rx / SRAM

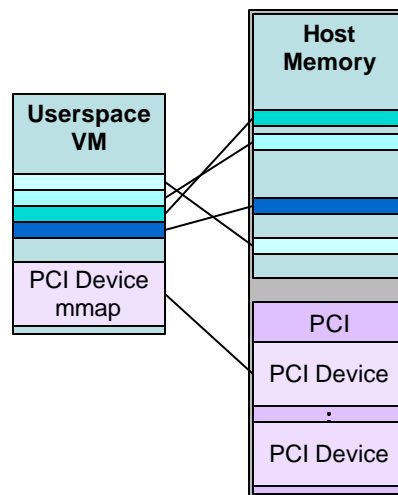


DMA Control Advantage



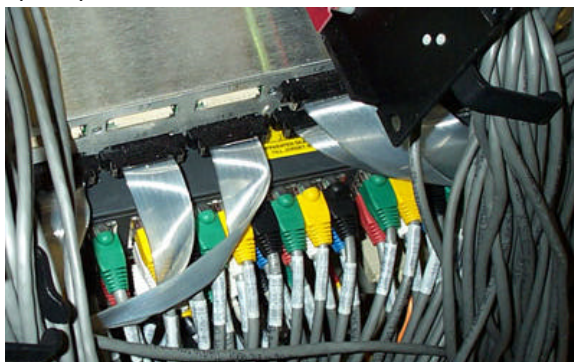
Virtual/Physical/Bus Memory

- **User data in VM**
 - Paged
 - Noncontiguous
- **Physical Memory**
 - Kernel manages
 - Pages to disk
- **Bus addresses**
 - PCI devices
 - X86: Physical=Bus



Myrinet Communication Libraries

- MyriAPI
- Active Messages (AM)
- Fast Messages (FM)
- BIP
- GM
- GRIM



Active Messages (UCB)

- TCP is too slow for cluster computing
 - Expose network performance to users
- Active Messages: process at arrival
 - Message has function *, four args, payload
 - Polling is explicit (and implicit)
 - Host-based flow control (fixed credits)

Fast Messages (UIUC)

- AM is too low level.. Fragmentation?
 - Better handlers, Better overlap
- Streaming message handlers
 - Handler extracts data as becomes available
 - Multiple simultaneous handlers
- NI Optimizations
 - NI directed flow control (Originally)
 - Optimistic: NI sends expecting buffer space available

BIP (UCBL)

- NI is slow, Handlers complicated
 - Want quick send/receive operations (MPI)
- Rendez-vous Messaging
 - Receiver posts tag in message layer (NI)
 - Incoming msg routed to VM based on tag
 - No tag, message dropped

VMMC (Princeton)

- **Message passing is hard**
 - Give us tools for shared memory
- **Host-to-host memory transfers**
 - RDMA-W Push page of VM to another host's VM
 - No need for buffer management
- **VM translation registers in NI**
 - Users request pages be managed by NI
 - VMMC firmware caches translations on-card

GM (Myricom)

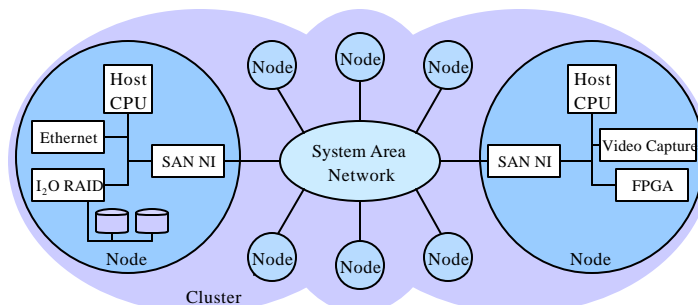
- **Do you trust academic message layers?**
 - Stable, universal, all-purpose message layer
- **Posted send/receives (MPI-ish)**
 - Split send: Sender has a callback
 - Multiple receives posted, poll to see completions
- **Reserved DMA memory for xfers**
 - User asks GM for DMAable memory
 - NI pushes/pulls only from DMAable memory
 - Advantage: NI has VM to Phys address translations

What's Next?

- Message layers are for host CPUs
 - What if we have powerful I/O devices?
- Distributed digital libraries
 - Intelligent storage
 - Hundreds/Thousands of IP ports
- Multimedia processing clusters
 - Video capture
 - Streaming computations (FPGAs, DSPs)

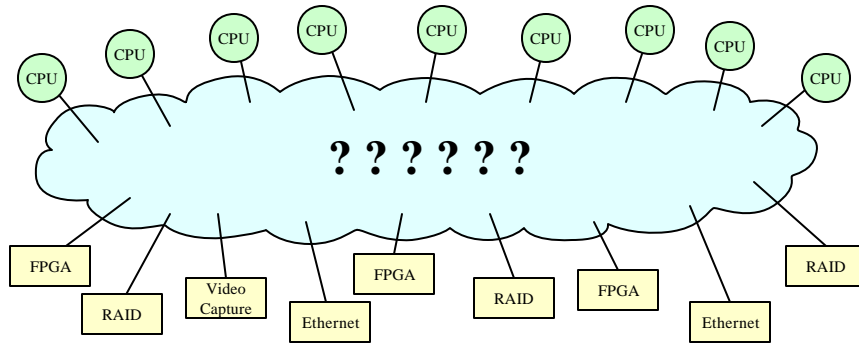
Resource Rich Clusters

- Inclusion of diverse peripheral devices
 - Ethernet s/a cards, multimedia capture devices, smart disks, hardware accelerators
- Processing in CPUs and peripherals



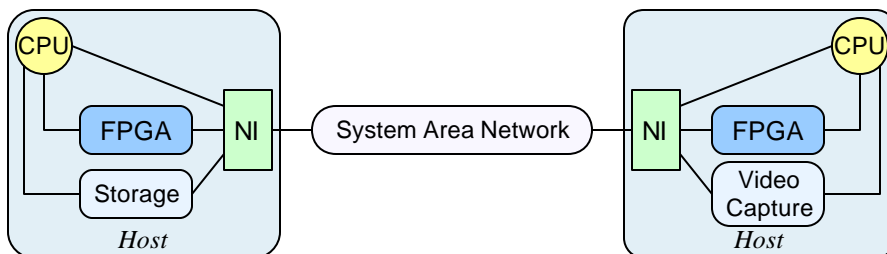
Message Layer Tasks

- Flat communication space
- Distributed processing mechanisms



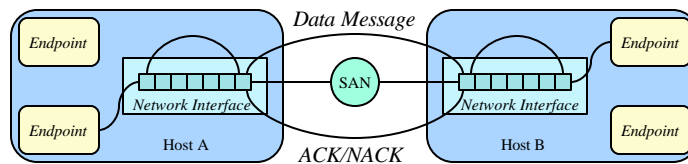
Communication Library: GRIM

- **General-purpose Reliable In-order Messages**
 - Supports inter-/intra- host communication
 - NI-based FC, Logical Channels, Active Messages



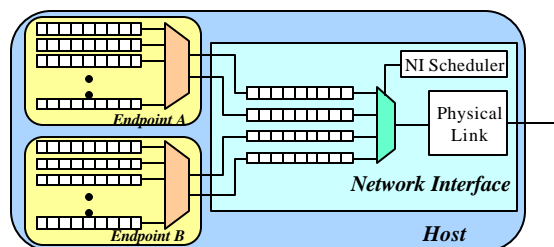
NI Directed Flow Control

- NI dynamically manages buffer FC
- Simplifies endpoint implementation
- Optimistic approach:
 - Transmit expecting success
 - Errors (receiver full): “go back n”



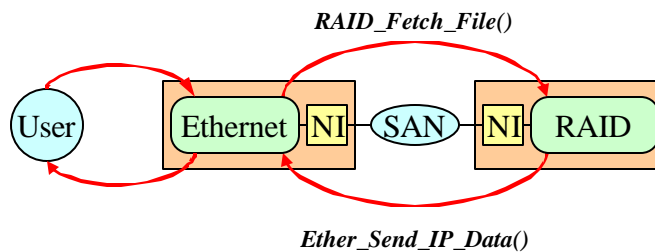
NI Logical Channels

- Messages have logical channel IDs
 - NI implements small set of channels (queues)
 - NIs synchronize FC on NI logical channels
- Endpoints assigned set of NI queues
 - Simplifies injection: no synchronization needed



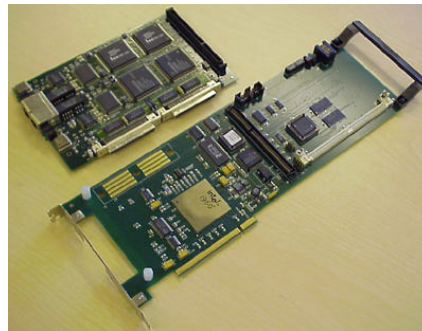
Active Message Processing

- Messages have associate function handler
 - Handlers registered with server
- Useful for peripheral devices
 - Encapsulate device capabilities



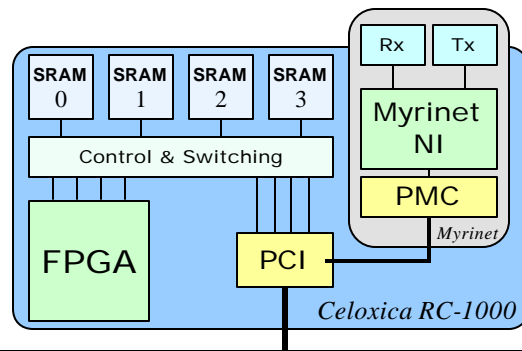
Cyclone Systems Server Adaptor

- “High-end” server card
 - 66MHz i960, 4 MB
 - Dual 100 Mbps Ethernet
 - Dual Ultra-wide SCSI
 - VxWorks
- GRIM Port
 - PCI Transactions
 - Polls like regular endpoint



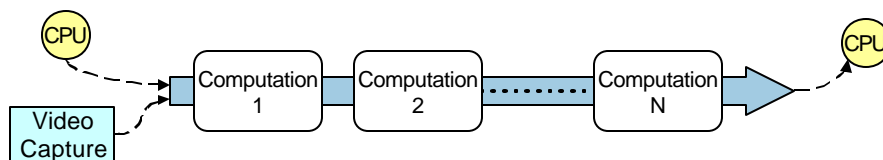
Celoxica RC-1000 FPGA Card

- **Celoxica RC-1000**
 - Xilinx Virtex-1000 FPGA
 - 8 MB SRAM
- **Myricom Myrinet NI**
 - LANai 4.3 / 33MHz / 1MB
 - PMC form factor



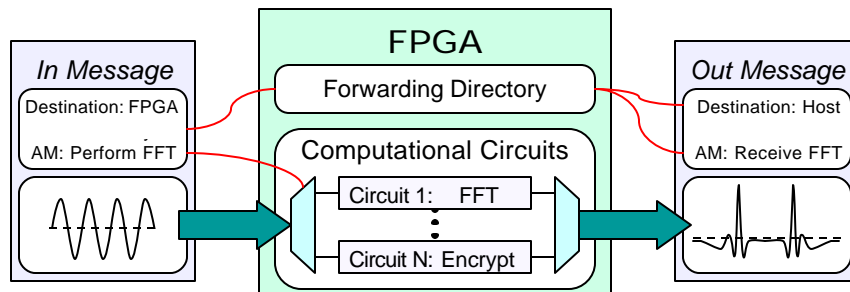
FPGA Programming Model

- Streaming computations
 - Load FPGAs with computational circuits
 - Stream data through FPGA and NI
- Extension: Route through multiple FPGAs

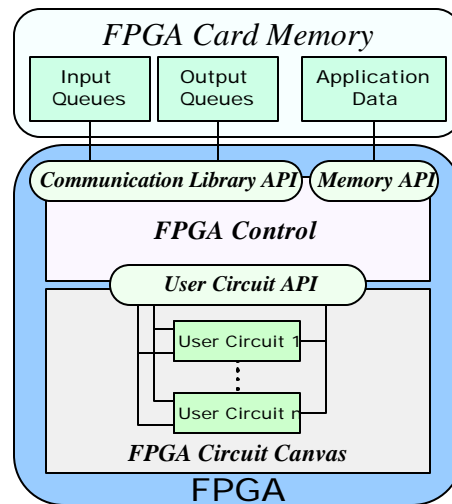


Streaming Fundamentals

- **Computation: How is computation specified?**
 - Active message approach
- **Forwarding: Where are results transmitted?**
 - Programmable *forwarding directory*



FPGA Interfaces



Concluding Remarks

- **Myrinet flexible cluster network fabric**
 - Network control by firmware
 - Minimal NI hardware, but high performance
- **Several message layers**
 - Expose network performance (cost?)
 - VM, removing extra copies a major issue
- **GRIM**
 - Different goals change design of ML
 - The doughnuts were poisoned