An Application-Oriented Communication System for Clusters of Workstations

Prof. Dr. Antônio Augusto Fröhlich
Thiago Robert dos Santos

snow@lisha.ufsc.br
http://snow.lisha.ufsc.br/
Outline

- Motivation
- The SNOW Project
- Application-Oriented System Design
- EPOS
  - Overview
  - Communication System
  - Performance
- Conclusions
Motivation

- Parallel computing performance revisited in 2000
  - a cluster of commodity workstations
  - running on commodity and custom run-time systems
Commodity Hardware, Custom Run-time System

- Commodity X Custom Hardware
  - Convergence

- Commodity X Custom Run-time Systems
  - Commodity
    - multi-{user, tasks,...}, interactive, web-aware
    - more distributed than parallel
  - Custom
    - high performance and low latency
    - specially designed to support parallel computing

- Clusters do need dedicated RTS in order to be as efficient as traditional supercomputers
SNOW Goals

- Developing an application-oriented parallel programming environment for clusters of workstations
  - run-time support system (EPOS)
  - programming language (DPC++)
  - management tools (CODINE)
- Bringing cluster performance closer to traditional supercomputers
- Validated by selected parallel applications
  - computational biology
  - complex industrial processes
Overview of a SNOW Cluster

- Server
  - fs
  - EPOS
  - Linux
  - ix86 FE

- Work-node
  - Application
  - MPICH DPC++
  - EPOS
  - Linux
  - ix86 Myrinet FE

- Work-node
  - Application
  - MPI DPC++
  - EPOS
  - ix86 Myrinet FE

Service network
High-speed network
Application-Oriented System Design
EPOS: SNOW's RTS

- application program
- inflated interfaces
- framework

- analyzer
- configurator
- generator

- info
- aspects
- adapters
- components

- system instance
Myrinet NIC

- LANai processor
  - Programmable NIC
- Parallel DMA engines
  - Host
  - Network send
  - Network receive
  - Two memory access per clock cycle
- Doorbell mechanism
  - Hardware support for VIA
Communication System (paged address space)

Host (GNU/Linux) -> Data Packets -> Application Address Space

1. Host (GNU/Linux)
2. NIC
3. Copy block (Non-swappable)
4. NIC
5. Host (GNU/Linux)
Communication System (flat address space)
Pipelined Communication System

Host (Epos)

Messages

Frames

Physical Memory
Non-swappable
Flat address space

OS

NIC

Send Ring

Receive Ring

1

2

3

4

Unsolicited Ring

Tx DMA Requests

Tx FIFO Queue

NIC

Rx FIFO Queue

Rx DMA Requests

Host (Epos)
Light-Weight Communication Protocols

- Infrastructure
  - broadcast/multicast
  - reliability
  - flow control
  - QoS
  - protection schemes
  - ...

- Parallel
  - immediate communication
  - barriers
  - collective operations
  - ...

Scalable Network of Workstations

June 2004
http://snow.lisha.ufsc.br
Communication System Performance

![Graph showing latency comparison between GM and Baseline]

- **Latency (microseconds)**
- **Frame size (bytes)**

The graph compares the latency of the GM network to the Baseline network for different frame sizes in bytes.
Conclusions

- SNOW is operational as regards
  - Run-time support system (EPOS)
  - MPI on EPOS
  - DPC++

- SNOW is able to support high-performance applications
  - RTS < 10 KB
  - Thread scheduling < 375 ns
  - Dynamic memory allocation < 100 ns
  - Round-trip latency < 5 us

- But real applications must confirm it!
  - Molecular biology at LabBioInfo