WP2

GridRPC/GridMPI:
The NAREGI Grid Programming Environment

Satoshi Sekiguchi, Yoshio Tanaka, Motohiko Matsuda, Yutaka Ishikawa
Programming Environment

- Grid allows a programmer to concentrate on developing software.
  - Separation of control flow, data flow, and program

- Future computing fabric – cluster and grid
  - Utilization of both concept.

- Grid RPC and Grid MPI cover the most of programming model
  - Grid RPC for “Master-Worker”, simple but useful
  - Grid MPI for general parallel model, powerful
Parallel Programming Models for LSSC

MPI

- flexiblity
- programming difficulty

RPC

- Simple
- Seamless programming

100000 CPU

100-500 CPU
NAREGI Software Stack

- Grid-Enabled Nano-Applications
- Grid Visualization
- Grid PSE
- Grid Workflow
- Super Scheduler
- Distributed Information Service
- Grid VM
- Grid Programing
  - Grid RPC
  - Grid MPI

(Globus, Condor, UNICORE ➔ OGSA)

Super Scheduler

High-Performance & Secure Grid Networking

SuperSINET

Computing Resources

- NII
- IMS
- Research Organizations
- etc
Climate Simulation using Ninf-G

Live Demonstration
**Application: Climate Simulation**

- **Goal**
  - Long term, global climate simulation
    - Winding of Jet-Stream
    - Blocking phenomenon of high atmospheric pressure

- **Barotropic S-Model**
  - Climate simulation model proposed by Prof. Tanaka
  - Simple and precise
    - Modeling complicated 3D turbulence as a horizontal one
    - Keep high precision over long periods
      - Taking a statistical ensemble mean
        - ~ several 100 simulations
      - Introducing perturbation at every time step
  - Typical parameter survey
Ninfty the original (seq.) climate simulation

Dividing a program into two parts as a client-server system

- **Client:**
  - Pre-processing: reading input data
  - Post-processing: averaging results of ensembles

- **Server**
  - Climate simulation, visualize

Client:
- Pre-processing: reading input data
- Post-processing: averaging results of ensembles

Server:
- Climate simulation, visualize

Web browser

S-model Program

Reading data
Solving Equations
Averaging results
Visualize

Ninf-g
Climate Simulation

**Servers**
- NAREGI Cluster
  - 160 cpu / 80 nodes
- AIST Super Cluster F32
  - 256 cpu / 128 nodes

**Grid PSE**
- Globus-job-run
- Gass-url-copy

**Sim. Server**

**Vis. Server**

**grpc_call**
Preliminary Evaluation

**Testbed: 500 CPU**
- TeraGrid: 225 CPU (NCSA)
- ApGrid: 275 CPU (AIST, TITECH, KISTI)

**Ran 1000 Simulations**
- 1 simulation = 12 seconds
- 1000 simulation = 12000 seconds = 3 hours 20 min (if runs on a single PC)

**Results**
- 150 seconds = 2.5 min

**Insights**
- Ninf-G2 efficiently works on large-scale cluster of cluster
- Ninf-G2 provides good performance for fine grain task-parallel applications on large-scale Grid.
GridMPI
Performance Demonstration using
WAN Emulated Environment
GridMPI Software Architecture

**RPIM (Remote Process Invocation Mechanism)**
- Providing the remote process invocation mechanism

**LACT (Latency-aware Communication Topology)**
- Collective Communications are implemented based on the network latency and bandwidth characteristics

**Request Layer**
- MPI Basic Communication Request Handling

**Point-to-Point Communication**
- Point to Point Communication Handling

---

<table>
<thead>
<tr>
<th>MPI Core</th>
<th>Latency-aware Communication Topology</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPIM Interface</td>
<td>IMPI</td>
</tr>
<tr>
<td>Vendor MPI</td>
<td>Request Layer</td>
</tr>
<tr>
<td>ssh</td>
<td>rsh</td>
</tr>
<tr>
<td>TCP/IP</td>
<td>PMv2</td>
</tr>
</tbody>
</table>
GridMPI Status

**Features**
- MPI-1.2 compatible
- IMPI 0.0 full implementation
- YAMPIII TCP/IP and SCore implementation

**Stability**
- The following test programs have been passed
  - MPICH Test suite
  - Intel Test suite

**On going**
- MPI-2
- Vendor MPI
- Incremental Checkpoint
- LACT

---

<table>
<thead>
<tr>
<th>MPI Core</th>
<th>Latency-aware Communication Topology</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPIM Interface</td>
<td>Vendor MPI</td>
</tr>
<tr>
<td>ssh</td>
<td>rsh</td>
</tr>
<tr>
<td>IMP</td>
<td>IMP</td>
</tr>
</tbody>
</table>

**Request Interface**

**Request Layer**

**P2P Interface**

Latency-aware Communication Topology

Topology
NaReGI WAN Emulated Environment

**Cluster Specification**
- Two clusters connected by emulated WAN
- Intel Xeon 2.8GHz
- 16 Procs for each cluster

**GNET-1 Specification**
- WAN Emulation at wire-speed
- Four GigaBit Ethernet ports
- FPGA + SRAM (144Mbits/port)
  - WAN Emulation (delay over 100ms)
  - Precise traffic measurement (1usec precision using GPS)
  - Traffic shaping, Smart bits
Demonstration

- **NAS Parallel Benchmark**
  Programs running on the
  WAN Emulated Environment

  - NAS Parallel Benchmarks
    - Data set size: CLASS=A
    - Proc size: NPROCS=32
  - Latency varied
    - 0 to 100ms (one-way)
  - Bandwidth varied
    - 10Mbps to 1Gbps
  - Bandwidth display
    - Measured by GNET-1

- Using IMPI protocol
  - Using YAMPIII p2p protocol
  - Using YAMPIII p2p protocol