

# LAB

## NUMA Control for Hybrid Applications



Hang Liu

September 13, 2011

# What you will learn

- Using numactl in execution of serial, MPI and a 4x# (4 tasks each with # threads) hybrid code

# Hybrid Computing -- Ranger

- Untar the file numahybrid.tar
  - cd (Start in your home directory.)
  - tar xvf ~train00/numahybrid.tar (extract files)
  - cd numahybrid

# numactl\_serial– Ranger

The memory intensive daxpy code is run on four different sockets using local, interleave and off-socket-memory policies. The commands below make the daxpy executable and run it with numa control commands. See the job script and the table on the next page for the numa options. Run the job and report the times and relative performance.

- Execute:

```
cd numactl_serial (change directory to numactl_serial)
```

```
make
```

```
qsub job (submits job)
```

# numactl\_serial– Ranger

- From the job output fill in the table.

| Command             | Time (secs) |
|---------------------|-------------|
| numactl -l -C 0     |             |
| numactl -l -C 1     |             |
| numactl -l -C 2     |             |
| numactl -l -C 3     |             |
| numactl -i all -C 0 |             |
| numactl -i all -C 1 |             |
| numactl -i all -C 2 |             |
| numactl -i all -C 3 |             |
| Numactl -m 3 -C 0   |             |

Rank the performance of local, interleave, and off-Socket-memory policies.

1.)

2.)

3.)

best to poorest performance.

# Numactl\_4x1, Numactl\_4x4 – Ranger

The daxpy code is run as 4 tasks in a node (4x1) and 4 tasks with 4 threads in a node. Cd down to directories numactl\_4x1 and numactl\_4x4, respectively, and follow the instructions below.

- Execute:  
cd numactl\_4x1 or numactl\_4x4  
make  
qsub job (submits job)

# Numactl\_4x1, Numactl\_4x4 – Ranger

- From the job output fill in the table.

| Command (4x1)              | Time (secs) |
|----------------------------|-------------|
| <no numactl><br>numactl -l |             |
| numactl -i all             |             |
| numactl tacc_affinity      |             |

  

| Command (4x4)              | Time (secs) |
|----------------------------|-------------|
| <no numactl><br>numactl -l |             |
| numactl -i all             |             |
| numactl tacc_affinity      |             |

Rank the 4x1 performance of.

- 1.)
- 2.)
- 3.)

From best to poorest performance.

Rank the 4x4 performance of.

- 1.)
- 2.)
- 3.)

From best to poorest performance