Measure performance of memory and networks

How to choose the nodes hardware ?

- Bandwidth between CPU - Cache – RAM
- Bandwidth of I/O channels
- Bandwidth and latency of the networks

Memory performance evaluation

Some benchmarks:

- **stream** (http://www.cs.virginia.edu/stream)
- **memperf** (http://www.cs.inf.ethz.ch/CoPs/ECT)

Stream

Measures memory (not cache) bandwidth performing basic instructions over a long stream of data:

- Copy: c[i]=a[i]
- Scale: c[i]=scalar*a[j]
- Add: c[i]=a[i]+b[i]
- Triad: c[i]=a[i]+scalar*b[i]

Running "stream"

- Download /home/school/perf/stream.tgz from server 140.105.19.181
- Untar and compile it:

```
tar -xvf stream.tgz
cd stream
 gcc -O3 -ostream *.c
```

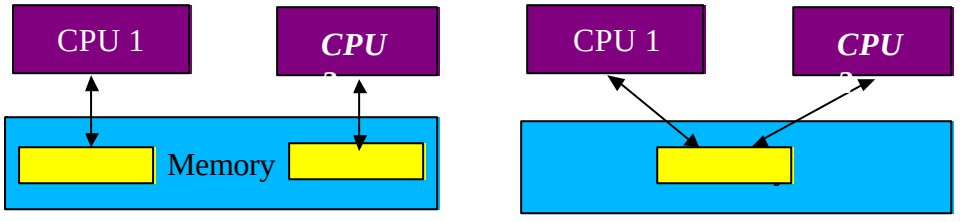
- ./stream

Memperf

- ***Measures memory bandwidth in a 2 dimensional way***
  - ***with different the block size***
    (explores the different cache levels)
  - ***with different access patterns***
    (from contiguous to different strided access)
- ***4 test provided:***
  - ***Load***
  - ***Store***
  - ***Load-copy (strided reads / contiguous writes)***
  - ***Copy-store (contiguous reads / strided writes)***

Memperf

- A number of processes can be specified to test simultaneous use of memory bus in SMP systems
- In this case it's possible running the test over a shared memory area (build with "make shmem")



Running "memperf"

download, compile and run:

140.105.19.181:/home/school/perf/memperf_v0.9e.tgz

cd memperf

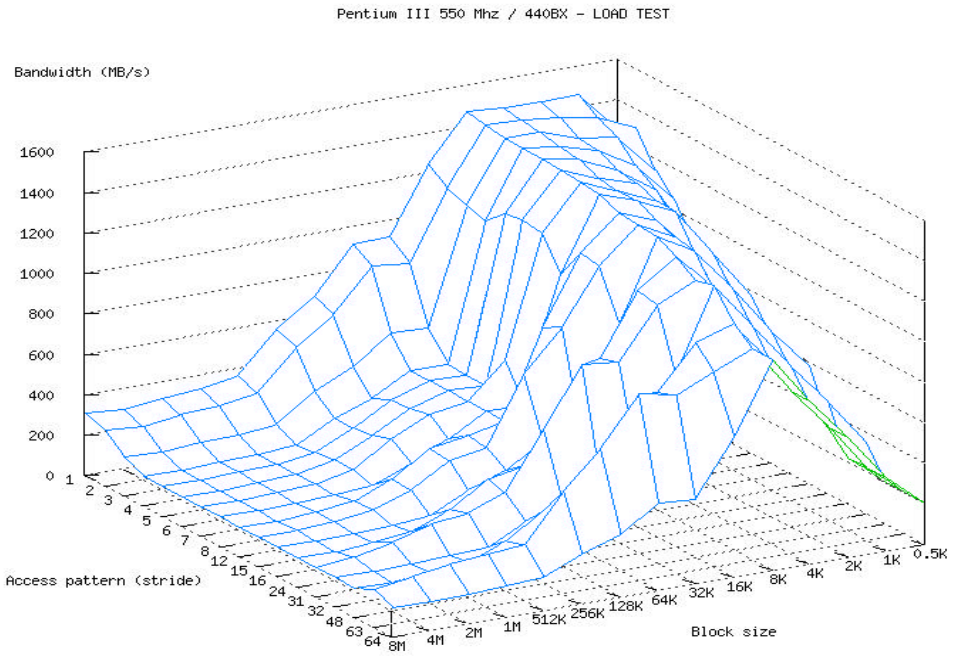
make

./memperf -m 0 (load test)

createplot3D chart.m0.p1.out

createplot2D chart.m0.p1.out chart.m1.p1.out

Memperf



Network performance

- Bandwidth versus size of block to transfer
- Latency (usually estimated as RTT/2)

Lots of benchmarks for TCP/IP:

- Ttcp
- Netperf (www.netperf.org/netperf/NetperfPage.html)
- Netpipe (http://www.scl.ameslab.gov/netpipe)
- .....

TCP/IP

- Nagle Algorithm
- Delayed acknowledge
- Slow start
- Congestion avoidance
- Socket buffer size (tunable)

Changing del default read/write socket buffer size

For example in Linux 2.2 to set the max and default buffer size:

echo 262144 > /proc/sys/net/core/wmem_max

echo 262144 > /proc/sys/net/core/rmem_max

echo 65536 > /proc/sys/net/core/rmem_default

echo 65536 > /proc/sys/net/core/wmem_default

In Linux 2.4:

echo "4096 87380 262144" > /proc/sys/net/ipv4/tcp_rmem

echo "4096 65536 262144" > /proc/sys/net/ipv4/tcp_wmem

echo 262144 > /proc/sys/net/core/wmem_max

echo 262144 > /proc/sys/net/core/rmem_max

MPICH socket buffer size option

Using mpich p4 over high speed TCP networks

You can specify a static socket buffer size

defining the enviroment variable:

P4_SOCKBUFSIZE=.....

Netpipe

- Test the latency and the bandwidth in a point to point transfer with increasing block sizes
- Can be used over TCP/IP (NPtcp) and over MPI (NPmpi)
- produces a file that contains the transfer time, throughput, block size, and transfer time variance for each data size **(you can use the CreateNetpipePlot" to render this file)**

Running NetPipe over TCP

- ***Download and untar /home/school/perf/netpipe-2-4.tgz from the server***
- ***cd netpipe-2.4***
- ***make***
- ***Open 2 shells on two different nodes of the cluster:***
- ***On node1 run: ./Nptcp -r***
- ***On node2 run: ./Nptcp -t -h node1 -P***
- ***Try to change options.....***

Running netpipe over MPI

- make MPI

run test using:

- mpirun -np 3 Npmpi -P

or better using a pbs job script

(for example /home/school/job.mpich)

Netpipe options

-A: specify buffers alignment e.g.: <-A 1024>

-b: specify send and receive buffer sizes e.g. <-b 32768>

-h: specify hostname <-h host>

-i: specify increment step size e.g. <-i 64>

-l: lower bound start value e.g. <-i 1>

-O: specify buffer offset e.g. <-O 127>

-o: specify output filename <-o fn>

-P: print on screen

-p: specify port e.g. <-p 5150>

-r: receiver

-s: stream option

-t: transmitter

-u: upper bound stop value e.g. <-u 1048576>

Pallas MPI Benchmarks(PMB)

http://www.pallas.com/pages/pmb.htm

Provide a concise set of benchmarks targeted at measuring the most important MPI functions.

- **See other information sheet**