Parallel Algorithms

Jesper Larsson Träff
traff@par.tuwien.ac.at
Technical University of Vienna
Institute of Information Systems
Parallel Computing, 184-5
Favoritenstrasse 16, 3. Stock

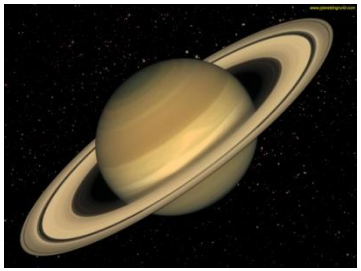
Sprechstunde: Monday, 13-15

TU Wien parallel computing shared-memory node

4xAMD „magny cours" 12-core Opteron 6168 processors
128GByte main memory, 1.9GHz, total number of cores 48

- Per core L1 cache: 128KB
- Per core L2 cache 512KB
- Shared L3 cache 12288KB

Name:
saturn.par.tuwien.ac.at



Is ready!

Intended use:

1. Develop/debug in lab/own PC
2. Transfer to saturn
3. Run, test, debug, run, benchmark, …
4. Transfer results back, write-up at own PC, submit…

Available software: emacs, vi, gcc, gdb – standard Linux tools

Experiment with optimization levels

Pthreads:

```
gcc -Wall -o myprogpt myprogpt.c -O3 -pthread
```

OpenMP:

```
gcc -Wall -fopenmp -o myprogomp -O3 myprogomp.c
```

BE CAREFUL:

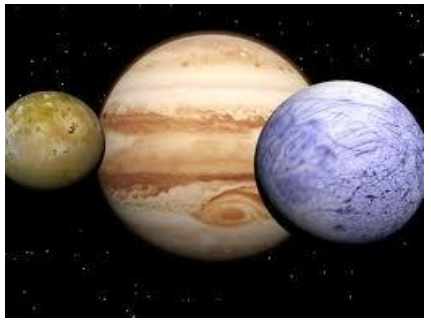
- Do not allocate too much memory(!!) – node may crash/hang
- Make sure processes are killed before exiting

TU Wien parallel computing distributed-memory system

36x2xAMD „magny cours" 8-core Opteron 6134 processors
32GByte main memory per node, 2.3GHz, total number of cores 576, total memory 1152GBytes.

Infiniband QDR switch MT4036

Name:
jupiter.par.tuwien.ac.at



Intended use:

1. Delevop/debug on own PC/lab
2. Transfer to jupiter (scp)
3. Run/benchmark
4. Transfer results back, write report

Currently installed: NEC mpi

bash

```
PATH=$PATH":"/opt/NECmpi/gcc/inst/bin64
```

tcsh

```
setenv PATH $PATH":"/opt/NECmpi/gcc/inst/bin64
```

Compile with

```
mpicc –o mpiprog mpiprog.c <all standard gcc
options>

mpicc -help
```

Execute with

```
mpirun –np <number of procs on node> mpiprog

mpirun –host jupiter1 –np 5 –host jupiter2 –np 5 …
mpiprog
```

Helpful (tcsh):

```
setenv MPILX_NODELIST 'jupiter0,jupiter1,…'

mpirun –node 1-33 –nnp 4 mpiprog
```

bash

```
MPILX_NODELIST=`jupiter1,jupiter2,jupiter3,…`
export MPILX_NODELIST
```

BE CAREFUL:

- Start on single node, jupiter0, up to 16 mpi processes
- Gradually extend number of nodes (!!!)
- NOTE: currently only 34 nodes (2 will be replaced)

- NOTE: installation for multi-node execution needs fix (see www), currently only single node (jupiter0)

- WORKAROUND: jupiter0 is special. It is possible to compile and run multi-node jobs from any other node, but use only nodes jupiter1, jupiter2, …, jupiter33