Workshop: High-performance computing for economists

Lars Vilhuber[1] John M. Abowd[1] Richard Mansfield[1]
Kevin L. McKinney

[1]Cornell University, Economics Department,

August 20-22, 2013: Day 3

HP resources

Data management

The gist of it all: QSUB

- Basic submission
- Understanding queue parameters
- Customizing

What to watch out for

- Too many jobs
- Within-job scheduling
- Threading of your software (SAS, Stata, Matlab)
- Storage
  - Size
  - Speed

Wihin-job scheduling

- Simple
- Pmon [locked]

Simple within-job scheduler

```
01_01_readBLS.R
02_01_readCensus.R
02_02_prepareCensus.R
03_01_create_analysis_data.R
04_01_runOLS.R
README.txt
```

Simple within-job scheduler

```
1  #!/bin/bash
2  #PBS -l ncpus=2
3  cd /to/mydir
4  set -- $(ls 0*R)
5  while [[ ! -z $1 ]]
6  do
7      R --vanilla < $1 &
8      shift
9      R --vanilla < $1
10     shift
11     wait
12 done
```

Threading

SAS

- Default on ECCO set to 1 thread.
- Override with ”sas -cpucount 2”
- ON ECCO, override with ”qsas prog.sas [chunks]”

qsas

```
> qsas
    /usr/local/bin/qsas prog[.sas] chunks

    will launch SAS under PBS-like systems, requesting [chunks] chunks
    and adjusting SAS memsize, sortsize, and sumsize appropriately.

    If not specifying [chunks], uses 1 CPU and 8GB of RAM.
    Chunks are defined in units of 2CPUS:16GB of RAM

    If these limits are insufficient, you may need to run a
    custom qsub job with '#PBS -l mem=XXXXmb' as one of the PBS options.

    (expert usage)
    To add additional PBS options, set a environment variable PBSEXTRA
    with the full set of options. It will be appended to the qsub
    command line
```

Example job

See QWI Macro examples

- code
- data

Storage speed differences

```
> iqsub 2
-sh-3.1$ echo "libname here '.';
                data here.one;
                  do i = 1 to 100000000;
                    output;
                  end;
                run;
                proc datasets library=here;
                  delete one;
                quit;
                " | sas -stdio 2>&1 | grep "real time" | tail -1

      real time                6.30 seconds

-sh-3.1$ cd /dev/shm
-sh-3.1$ echo "libname here '.'; data here.one;do i = 1 to
100000000;output;end;run; proc datasets library=here; delete one;quit;" | sas
-stdio 2>&1 | grep "real time" | tail -1

      real time                3.38 seconds

-sh-3.1$ cd /temporary/
-sh-3.1$ echo "libname here '.'; data here.one;do i = 1 to
100000000;output;end;run; proc datasets library=here; delete one;quit;" | sas
-stdio 2>&1 | grep "real time" | tail -1

      real time                4.27 seconds
```

Wrap-up