# Yellowstone



# Yellowstone Tutorial

## Computing at NCAR

- Mesa Lab decommissioning schedule
  - Bluefire decomissioned January 31, 2013
    - `?/ptmp` read-only as of January 14
  - Mirage & Storm (Analysis machines) decommissioned February 28
  - `/glade` file systems decommissioned March 31
- Yellowstone & NCAR Wyoming Supercomputing Center
  - Hardware
    - 72,288 cores, 16 cores per node, 4518 nodes
    - Intel processors
    - low-latency high bandwidth interconnect (Mellanox InfiniBand)
  - Graph of Peak FLOPS at NCAR
  - Construction

## Using Yellowstone

- Yellowstone, Geyser & Caldera
- Account Management: https://sam.ucar.edu
- Software Modules
  - compilers: we recommend Intel
  - python: coming soon
- File systems
  - /glade/u/home/username: 10 GB, backed up
  - /glade/scratch/username: 10 TB, temporary file space, regularly purged
  - /glade/p/work/username: 512 GB
- Compute queues
  - Account Key: Get from sam.ucar.edu or your sponsor
  - small, regular & economy

## LTR

- About
  - LFM-MIX, CMIT, LFM-RCM
  - Version 2.2.0
    - InterComm-2.0
    - Job scripts
    - MakeItSo
- Download
  - Tarball: http://cism.hao.ucar.edu/files/LTR/LTR-2_2_0.tar.gz (password required)
  - Subversion

```
svn checkout https://proxy.subversion.ucar.edu/cism_MODELS/tags/LTR-para/LTR-2_2_0
```

- Setup
  - All prerequisites are already built. The code is shipped with the appropriate machine configuration (see env/Make.yellowstone). To build the code, you need to set up some modules & environment variables:
  - Modules: It's probably a good idea to just set your environment to load all the appropriate modules by default. Here's how you do that:

```
module purge
module load intel/12.1.5 ncarenv ncarcompilers python all-python-libs
module setdefault
```

- Environment Variables: Set the following at login/startup:

```
export MACHINE=yellowstone

export LTRROOT=??????
export PATH=${LTRROOT}/misc/python:${PATH}
export PATH=${LTRROOT}/misc/pyLTR/scripts:${PATH}
export PYTHONPATH=${LTRROOT}/misc/pyLTR:${PYTHONPATH}
export TGCMDATA=/glade/u/home/schmitt/tgcmdata
```

- Compile

```
gmake LFM-MIX RESOLUTION=single
```

- Execution
  - MakeItSo
    - updated for 2.2.0; scripts are not necessarily backwards-compatible
    - MPMD job scripts
  - Solar Wind Files & Solar Wind Processor
  - Typical Performance on Yellowstone:

| Resolution | Name | Core count | Performance |
|---|---|---|---|
| 53-24-32 | single | 8 | 1/6 faster than real time |
| 53-48-64 | double | 24 | 2/3 faster than realtime |
| 106-96-128 | quad | 144 | 4x slower than realtime |

- Postprocessing
  - Interactive job on Geyser

```
bsub -XF -Ip -q geyser -W 1:00 -n 1 -P P28100045 /bin/bash
```

  - pyLTR
  - ParaView
  - CISM_DX

## Yellowstone Performance

Here are some rough performance numbers to anticipate. Note that these are rough estimates for standard solar wind input. The LFM uses a variable timestep and your results may vary, especially for high speed flows.

| Resolution | Grid | Core count | Performance |
|---|---|---|---|
| Single Resolution | 53x24x32 | 8 | 1.33 core hours per simulated hour |
| Double Resolution | 53x48x64 | 24 | 16 core hours per simulated hour |
| Quad Resolution | 106x96x128 | 144 | 576 core hours per simulated hour |

## How do I debug my code with TotalView?

There are a few simple steps to run your code with the TotalView debugger:

1. Load debugging modules

```
module load debug totalview
```

2. Compile your code with debugging flags enabled. For the LFM, edit `env/Make.yellowstone` and set

```
OPTLVL = -g -traceback -debug full
TRAP  =  -fp-stack-check -fstack-security-check -ftrapuv
```

3. Edit job run script, adding the following three lines to the LSF/BSUB settings near the top:

```
#BSUB -XF   # X11 forwarding
#BSUB -Ip   # interactive job
#BSUB -a tv # select the tv elim
```

4. Now submit your job script via bsub

Here's a complete sample job script to run one binary (LFM) with TotalView:

```
#!/bin/sh
#BSUB -J totalview
#BSUB -o totalview.%j.output
#BSUB -e totalview.%j.error
#BSUB -XF   # X11 forwarding
#BSUB -Ip   # interactive job
#BSUB -a tv # select the tv elim
#BSUB -n 24
#BSUB -R "span[ptile=16]"
#BSUB -W 01:00
#BSUB -q small
#BSUB -P xxxxxxxxx
#BSUB -R "select[scratch_ok > 0]"

# Setup
#source /glade/u/home/schmitt/opt-intel-12.1.4/InterComm-2.0/lib/build.env
#export LD_LIBRARY_PATH=/glade/u/home/schmitt/opt-intel-12.1.4/overture/lib:${LD_LIBRARY_PATH}
ln -sf INPUT1-001.xml INPUT1.xml

# Executable to run TotalView with
mpirun.lsf ./LFM  < /dev/null > totalview.out 2>&1
```

### Troubleshooting

If you get an error like

```
Job <876021> is submitted to queue <small>.
<<ssh X11 forwarding job>>
<<Waiting for dispatch ...>>
Warning: Permanently added '10.12.2.17' (RSA) to the list of known hosts.
lyon@10.12.2.17's password:
```

then you need to reset your SSH keys. CISL wrote a script to reset your keys, execute this:

```
/glade/u/home/siliu/bin/ssh-auth.bash
```

# Resources

## Presentation slides

| File | Modified |
|---|---|
| Microsoft Powerpoint Presentation IntroductionToYellowstone.pptx | Jan 15, 2013 by Peter Schmitt |
| PDF File IntroductionToYellowstone.pdf | Jan 15, 2013 by Peter Schmitt |

Download All

## Useful Links