This document is for the users that use high performance computing resources in UNIST Supercomputing Center. It starts on page 18 for the foreign researchers.

UNIST HPC User Guide

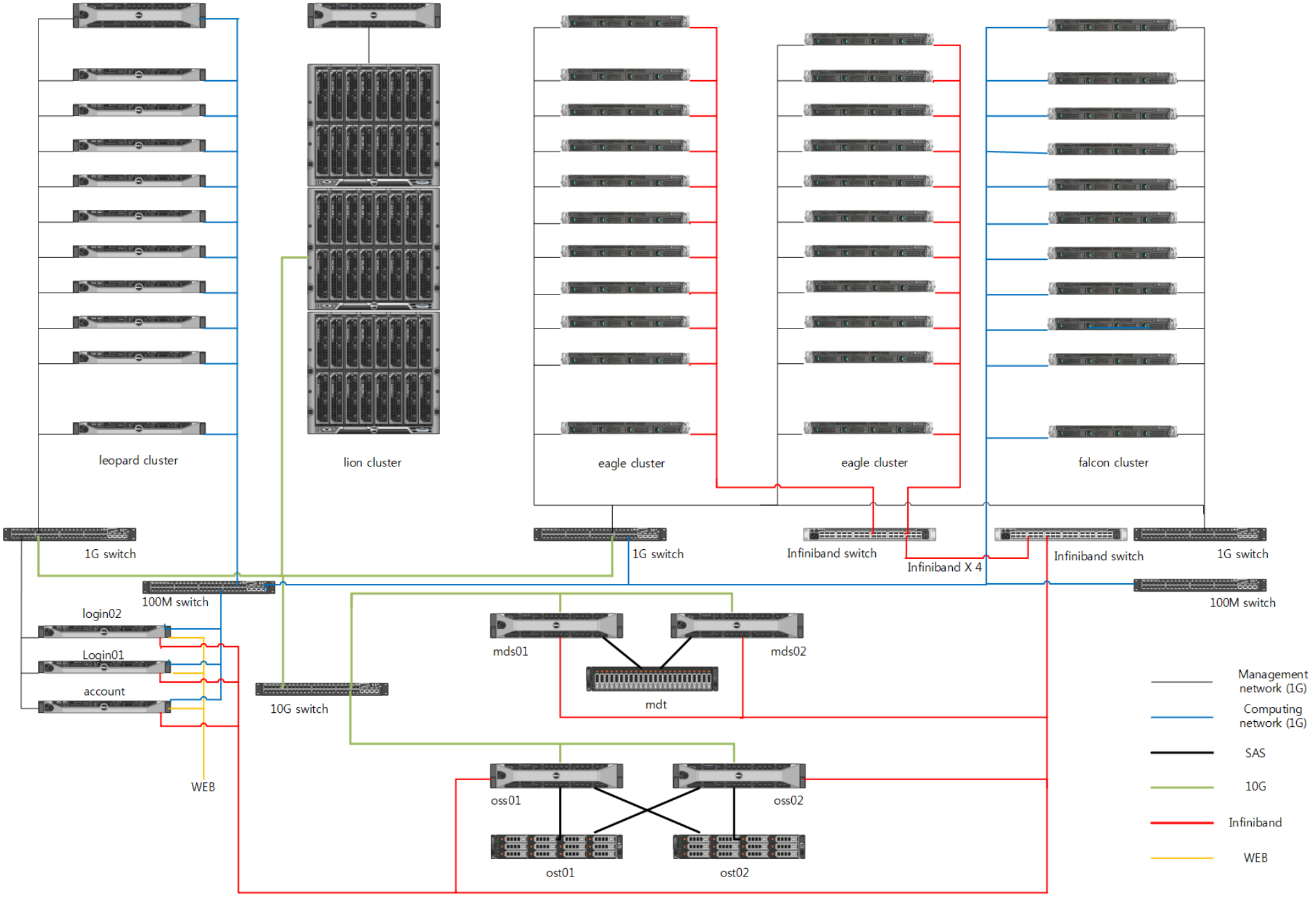
1. HPC Resources

A. Hardware

i. Configuration Overview

UNIST Supercomputing Center(이하 USC)에서는 다양한 연구 지원을 위해 High Performance Computing(이하 HPC) system 과 병렬 파일 시스템을 지원합니다. 저희 USC에서는 HPC x86_64 Linux 클러스터로 되어있는 Leopard, Lion, Falcon, Eagle이 있으며 병렬 파일 시스템과 높은 I/O, 계산을 지원하는 Lustre filesystem으로 구성되어 있습니다.

Leopard 시스템은 2011년에 설치 되었고 28개의 계산노드로 구성되어 있고, 각 node 당 8개의 코어를 장착하고 있습니다. Leopard 시스템의 CPU architecture는 Nehalem입니다. 이후, Lion 시스템이 도입되었고 Lion 시스템은 blade 형태의 장비로 41개의 계산노드, 각 node 당 12, 16, 20 core들로 구성되어 있습니다. 또한, Falcon 시스템과 Eagle 시스템이 설치되어 있습니다. Falcon은 40개의 계산노드에 20개, 24개, 28개 코어를 가진 노드로 구성되어 있고, Eagle은 31개의 계산노드로 구성되어 있습니다. HPC 시스템들의 총 core의 수는 약 1,800개 이고, 대부분의 시스템이 1G Gigabit Ethernet이나, eagle 시스템은 QDR(40Gbps) Infiniband로 연결되어 있습니다. 자세한 HPC system 구성은 다음 그림을 통해 확인할 수 있습니다.


leopard cluster
lion cluster
eagle cluster
eagle cluster
falcon cluster
1G switch
1G switch
Infiniband switch
Infiniband X 4
Infiniband switch
1G switch
100M switch
100M switch
login02
Login01
account
WEB
10G switch
mds01
mds02
mdt
oss01
oss02
ost01
ost02
Management network (1G)
Computing network (1G)
SAS
10G
Infiniband
WEB

ii. How to access HPC systems

1. All paths via LOGIN nodes

HPC시스템에는 login node가 2개 있습니다. 모든 사용자는 Login node를 통해서 HPC 시스템에 접근할 수 있습니다. 사용자 PC와 login node 간 연결은 SSH protocol(2022번 포트) 을 통해 이루어지고, 이것을 가능하게 해 주는 프로그램(PuTTY, Secure Shell Client, Xmanager 등)을 이용하시면 됩니다.

Hostname	login01	login02
DNS	**login01.usc.unist.ac.kr**	**login02.usc.unist.ac.kr**

※ UNIST supercomputing center에 설치된 시스템은 UNIST 외부에서 접속이 불가합니다. (특수한 경우, 허용된 IP에 의해서만 접근 허용)

2. After a successful connection to login nodes

Login node에 접속한 후 SSH(Secure Shell)를 사용하여 모든 시스템에 접속할 수 있습니다. 'ssh [hostname]'을 입력하면 해당 host로 접속할 수 있습니다.

iii. Disk quota

HPC는 병렬 file system인 Lustre로 만들어져 있고 모든 사용자의 home과 work directory는 고성능 네트워크로 구성된 parallel file system server에서 가져옵니다.

1. home directory

사용자의 home 디렉토리의 위치는 /uhome 아래에 있고, home dirsk quota는 50GB로 되어있습니다. 사용량을 확인하려면 아래의 command를 이용하시면 됩니다.

```
$ quota
```

2. work directory

Work directory의 경로는 /uwork/$USER로 되어 있습니다. work directory의 disk qouter는 1TB로 되어 있으나 해당 directory는 home directory와 다르게 data backup이 되지 않습니다.

7일이 경과된 미사용 data는 자동으로 삭제되게 정책이 되어있으니 중요한 data는 /uhome/$USER아래에 저장해두세요.

iv. HPCs

1. Leopard

A. System Overview

Hostname	Leopard
Number of node(# of core)	28 node (224 cores), 8cores/node
Processor	Intel Xeon E5540 2.53GHz
Memory	24GB per node
OS	Linux CentOS 6.5
Interconnection	1G Ethernet between computing nodes 10G network to parallel file system from leopard switch
Storage	/uhome, /uwork in Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs
leopard-short.q	48 hours	-	leopard-short
leopard-normal.q	168 hours	-	leopard-normal
leopard-long.q	504 hours	80	leopard-long
leopard-serial.q	no limitation	16	-

wall clock time이 길수록 낮은 우선순위를 가지며, 우선순위가 가장 높은 queue는 leopard-short.q입니다.

ii. Parallel Environment(PE)s

PE는 queue name과 동일하게 되어 있습니다. 사용자가 leopard-normal.q를 사용해서 job을 submission 할 경우, leopard-normal pe를 사용해야 합니다. 동일하게 leopard-long.q를 사용할 시 에는 leopard-long pe를 사용해야 합니다. Queue name과 pe가 일치하지 않으면 job은 계속 qw 상태에 머무르게 됩니다.

Also, very careful thing is here.

computing node에서는 physical core와 같은 core의 수를 사용합니다. leopard system은 node당 8core를 가지고 있으므로 사용자는 core수(8)의 배수를 사용해야 합니다. (ex. 8, 16, 32....)

iii. Resource quota

HPC의 resource독점사용을 막기 위해 resource 할당 정책이 설정되어 있습니다. Quota는 연구그룹별 quota로 주

어지며, 병렬계산 노드에서는 그룹별 80core까지, 순차코드 계산 노드에서는 12core까지만 '동시 사용'이 가능합니다. leopard27과 leopard28은 sequential job node입니다. Job status가 wait상태일 시 resource를 얼마나 사용하고 있는지 확인하시기 바랍니다.

2. Lion

A. System Overview

Hostname	Lion
Number of node(# of core)	41 nodes (584 cores), 12cores(25nodes), 16cores(9nodes), 20cores(7nodes)
Processor	Intel Xeon X5650 2.66GHz, Xeon E5-2670 2.60GHz, E5-2670v2 2.50GHz
Memory	24GB(25nodes) + 64GB(16nodes)
OS	Linux CentOS 6.5
Interconnection	1G Ethernet between computing nodes 10G network to parallel file system from lion switch
Storage	/uhome, /uwork in the Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs	notes
lion-short.q	48 hours	-	lion-short	

lion-normal.q	168 hours	-	lion-normal	
lion-long.q	504 hours	80	lion-long	
lion-octa.q	unlimited	-	mpi_1 ~ mpi_16	these queues are allowed to specific groups
lion-deca.q	unlimited	-	mpi_1 ~ mpi_20	
lion-serial.q	unlimited	12	-	

ii. Parallel Environment(PE)s

PE는 queue name과 같게 되어 있습니다. 사용자가 lion-normal.q를 사용해서 job submit 시 lion-normal pe를 사용해야 합니다. 동일하게 lion-long.q를 사용할 시 에는 lion-long pe를 사용해야 합니다.

Also, very careful thing is here.

computing node에서는 physical core와 같은 core의 수를 사용합니다. lion system중 lion01 ~ lion25는 node당 12core를 가지고 있으므로 사용자는 core수(12)의 배수를 사용해야 합니다.(ex. 12, 24, 36....) 추가로 lion26에서는 앞에서 말한 rule이 적용되지 않습니다.

C. Resource quota

HPC의 resource독점사용을 막기 위해 resource 할당 정책이 설정되어 있습니다. Quota는 연구그룹별 quota로 주어지며, 병렬계산 노드에서는 그룹별 96core까지, 순차코드 계산 노드에서는 18core까지만 '동시 사용'이 가능합니다. lion01과 lion02는 sequential job입니다. Job status가 wait상태일 시 resource를 얼마나 사용하고 있는지 확인하시기 바랍니다.

3. Falcon

A. System Overview

Hostname	falcon
Number of node(# of core)	16 nodes (320 cores)
Processor	Intel Xeon E5-2690v2 3.00GHz
Memory	128GB
OS	Linux CentOS 6.5
Interconnection	1G Ethernet between computing nodes 1G network to parallel file system from lion switch
Storage	/uhome, /uwork in the Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs
falcon-short.q	unlimited	-	mpi_1 ~ mpi_20, falcon-short
falcon-normal.q	unlimited	-	mpi_1 ~ mpi_20, falcon-normal
falcon-long.q	unlimited	-	mpi_1 ~ mpi_20, falcon-long

ii. Parallel Environment(PE)s

mpi1부터 mpi20까지와 falcon-short, falcon-normal, falcon-long은 PE job을 구동할 수 있습니다. falcon-short/normal/long은(mpi1~mpi20은 제외) allocation rules이 설정되어 있습니다 해당 queue를 사용 시 allocation rule 에 맞게 core를 입력하여야 합니다.(allocation 확인 시 $qconf –sp falcon-normal 또는 $qconf –sp falcon-short… allocation rule수치가 20일 때 입력 가능한 core는 20, 40, 60…)

mpi의 사용 syntax는 mpi_[digit1] [digit2]이며 [digit1]은 1부터 20까지 입력 가능합니다. [digit2]는 solving할 total core를 뜻하며 [digit1]은 하나의 node당 solving할 core의 수를 뜻합니다. [digit1]은 [digit2]보다 작아야 되며 [digit2]의 약수여야 합니다.(ex. mpi_15 15, 입력 시 하나의 계산노드에 15core를 사용하여 계산, mpi_16 32 입력 시 2개의 노드에서 16core씩 사용하여 계산, mpi_4 8, mpi_5 10…)

C. Notices

현재 Falcon cluster는 독점 시스템이므로 일반 사용자들은 추후에 사용 가능합니다.

4. Eagle

A. System Overview

Hostname	**eagle**
Number of node(# of core)	31 nodes (620 cores)
Processor	Intel Xeon E5-2690v2 3.00GHz
Memory	128GB
OS	Linux CentOS 6.5
Interconnection	Infiniband(40G) network between computing nodes Infiniband network to the parallel file system
Storage	/uhome, /uwork in the Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs
eagle-short.q	24 hours	Up to group quota	mpi_2 ~ mpi_20
eagle-normal.q	48 hours	Up to group quota	mpi_2 ~ mpi_20

ii. Parallel Environment(PE)s

mpi_2부터 mpi_20까지 PE job을 수행할 수 있습니다. Eagle 클러스터에서는 single core를 이용하는 serial job을 실행할 수 없습니다.

mpi의 사용 syntax는 mpi_[digit1] [digit2]이며 [digit1]은 2부터 20까지 입력 가능합니다. [digit2]는 solving할 total core를 뜻하며 [digit1]은 계산에 사용할 노드 당 core의 수를 뜻합니다. [digit1]은 [digit2]보다 작아야 되며 [digit2]의 약수여야 합니다. (ex. mpi_5 5, 입력 시 하나의 계산 노드에 5core를 사용하여 계산, mpi_6 12 입력 시 2개의 노드에서 6core씩 이용 총 12코어를 사용하여 계산)

C. Quota

HPC의 resource독점사용을 막기 위해 resource 할당 정책이 설정되어 있습니다. Quota는 연구그룹별 quota로 주어지며, 병렬계산 노드에서는 그룹별 160core까지입니다.

D. Notice

Eagle system은 node간 통신 속도가 40GB/s인 Infiniband network로 구성되어있습니다. 이론적으로는 Ethernet network보다 40배 빠릅니다.

2. Software

A. Compilers

Compilers	Installed system
GNU 4.4.7	leopard, lion, falcon, eagle
Intel composer xe 2013	
PGI 14.3	lion

B. Libraries

Classification	Libraries	Installed systems
Parallel libraries	OpenMPI 1.6.5	leopard, lion, falcon, eagle
	OpenMPI 1.8.1	
	MPICH-1.2.7p1	
	MPICH2-1.4.1p1	
	MPICH-3.1.2	lion, falcon
Mathematical libraries	FFTW 2.1.5	leopard, lion, falcon, eagle
	FFTW 3.3.4	
	Intel MKL	

C. Computational Packages

Classification	Applications	Installed systems
Commercial	ABAQUS 6.14-2	leopard, lion, falcon, eagle
	ANSYS CFD, Fluent	
	COMSOL Multiphysics 5.0	
	VASP 5.3	
	Gaussian 09	
Open Source	GROMACS	leopard, lion, falcon, eagle
	LAMMPS	
	Quantum Espresso	
	R Source	

※ NOTICE : 사용자가 연구를 위해 상용 package 사용을 원하는 경우 불법 license사용 방지를 위해 사용 전 연락 주시기 바랍니다.

3. Environment Settings

A. Overview

USC는 설치된 모든 software에 대해서 module을 만들어 놨습니다. USC has implemented module environment to manage users' environment for all software installed in our HPC systems./ 사용자들은 프로그램 동작을 위해 HPC에서 환경설정 자주 사용하는데 기존의 방법대로 설정하자면 실행파일, 경로, 라이브러리 경로 등을 추가해줘야 하는데 UNIX를 자주 쓰던 사용자는 익숙할 수 있지만 처음 사용하는 사용자에게는 번거로운 작업입니다. 때문에 저희 HPC system에서는 이와 같은 수고스러움을 방지하고자 module system을 사용하고 있습니다. Module file을 생성해 놓아서 필요한 module을 "load"하거나 "unload"하면 됩니다. 추가로 사용자가 필요로 할 시 ".file"을 수정하여 개별 module file을 변경할 수 있습니다.

module 을 사용하게 되면 경로, 라이브러리 경로, MAN 경로 등을 업데이트 할 때 쉽게 변경할 수 있습니다

HPC system 에서 계정이 생성되었을 때 ".file"을 사용하여 user environment 를 구성 및 변경해야 되지만(default 는 bash shell), module 을 사용하면 좀더 쉽게 user environment 를 설정할 수 있습니다.

B. Module commands

아래는 module에 대한 command입니다.

```
$ module help
```

"help" 옵션은 아래와 같이 출력됩니다. (help를 입력하지 않아도 동일하게 출력됩니다.)

```
Available SubCommands and Args:
 + add|load            modulefile [modulefile ...]
 + rm|unload           modulefile [modulefile ...]
 + switch|swap          [modulefile1] modulefile2
 + display|show        modulefile [modulefile ...]
 + avail              [modulefile [modulefile ...]]
 + purge
 + list
```

$ module list

현재 modulefile 을 출력합니다

$ module avail

사용할 수 있는 모든 modulefiled 을 출력합니다.

$ module purge

현재 load 된 modulefile 을 모두 unload 합니다.

$ module load *modulefile*

"*modulefile*"을 load 합니다.

$ module unload *modulefile*

"*modulefile*"을 unload 합니다.

$ module switch modulefile_old modulefile_new

"*modulefile_old*"을 "*modulefile_new*"로 변경 합니다.

$ module show *modulefile*

"*modulefile*"을 PATH, LD_LIBRARY_PATH, MANPATH, etc 로 변경하는 방법을 출력합니다.

4. Job Submission

A. Overview

USC에서는 사용자가 code를 실행하기 위해서 queue system을 사용해야 합니다. 저희 USC에서는 사용자의 효과적인 사용 및 편의를 위해 SGE(Sun Grid Engine)를 사용하여 job을 관리 합니다.

B. SGE commands

Commands	Example	Description
qsub	qsub job_script_file_name	submit a job.
qstat	qstat	show job status oneself
	qstat -u '*'	show job status for all users
qhost	qhost	show computing node status
qdel	qdel job_ID qdel -u user_ID (user's all job cancelled)	cancel a job
qconf	qconf -sql	show all queue list
	qconf -spl	show all pe list
	qconf –sq queuename	show about "queuename" detailed
	qconf -sul	show all user list
	qconf -srqs	show resource quota policy
	qconf -shgrpl	show host group list
	qconf -sc	show complex attributes

C. 시스템별 작업 스크립트 예제

i. Leopard HPC

1. Serial Job (코어 1개 이용)

Leopard HPC에서 serial job은 27번, 28번 전용 노드에서 계산됩니다.

```
#!/bin/bash
#$ -V                                  # job submit node의 shell 환경변수를 computing node에도 적용(default)
#$ -cwd                                # 현재 directory를 작업 directory로 사용
#$ -N serial_job                       # Job Name. 명시하지 않으면 job_script 이름을 가져옵니다.
#$ -q leopard-serial.q                 # queue name
#$ -S /bin/bash                        # shell selection
#$ -wd /uhome/<user01>/serialtest      # 작업할 directory를 설정합니다. 현재 directory(pwd)가
                                       # /uwork/<user01>이 아닐 경우 사용.
                                       # 그렇지 않다면 cwd만 써도 무방
#$ -l h_rt=01:00:00                    # 작업 경과시간(hh:mm:ss)(wall time clock).
                                       # 미 기입 시 작업이 강제 종료 됩니다.
                                       # job이 wall time clock에 도달 시 자동으로 중지됩니다.
. /etc/profile.d/modules.sh
module load intel/mkl-10.1.3
./execution_file_name
```

2. Parallel Job (1개 이상 코어 이용)

```
#!/bin/bash
#$ -V                          # job submit node의 shell 환경변수를 computing node에도 적용(default)
#$ -pe leopard-short 8         # parallel environment(pe)로 설정, 사용자가 원하는 core수 만큼 입력
```

```
#$ -N parallel_job          # Job name. 명시하지 않으면 job_script이름을 가져옵니다.
#$ -q leopard-short.q       # 사용할 queue name
#$ -S /bin/bash             # shell selection users want to use
#$ -cwd                     # 현재 directory를 작업 directory로 사용
#$ -l h_rt=24:00:00         # 작업 경과시간(hh:mm:ss)(wall time clock).
                            # 미 기입 시 작업이 강제 종료 됩니다.
                            # job이 wall time clock에 도달 시 자동으로 중지됩니다.

. /etc/profile.d/modules.sh
module load mpi/intel-11.1/openmpi-1.4.4     # ex) in case of OpenMPI-1.4.4
mpirun -machinefile $TMPDIR/machines -np $NSLOTS ./execution_file
```

ii. Lion HPC

1. Serial job (1개 코어 사용)

```
#!/bin/bash
#$ -V                                # job submit node의 shell 환경변수를 computing node에도 적용(default)
#$ -cwd                              # 현재 directory를 작업 directory로 사용
#$ -N serial_job                     # Job Name. 명시하지 않으면 job_script 이름을 가져옵니다.
#$ -q lion-serial.q                  # queue name
#$ -S /bin/bash                      # shell selection
#$ -wd /uhome/<user01>/serialtest    # 작업할 directory를 설정합니다. 현재 directory(pwd)가
                                     # /uwork/<user01>이 아닐 경우 사용.
                                     # 그렇지 않다면 cwd만 써도 무방
#$ -l h_rt=01:00:00                  # 작업 경과시간(hh:mm:ss)(wall time clock).
                                     # 미 기입 시 작업이 강제 종료 됩니다.

. /etc/profile.d/modules.sh
```

```
module load intel/mkl-10.1.3
./execution_file_name
```

2. Parallel job (2개 이상 코어 사용)

```
#!/bin/bash
#$ -V                          # job submit node의 shell 환경변수를 computing node에도 적용(default)
#$ -pe lion-short 12           # parallel environment(pe)로 설정, 사용자가 원하는 core수 만큼 입력
#$ -N parallel_job             # Job name. 명시하지 않으면 job_script이름을 가져옵니다.
#$ -q lion-short.q             # 사용할 queue name
#$ -S /bin/bash                # shell selection users want to use
#$ -cwd                        # 현재 directory를 작업 directory로 사용
#$ -l h_rt=24:00:00            # 작업 경과시간(hh:mm:ss)(wall time clock).
                               # 미 기입 시 작업이 강제 종료 됩니다.
                               # job이 wall time clock에 도달 시 자동으로 중지됩니다.

. /etc/profile.d/modules.sh
module load mpi/intel-11.1/openmpi-1.4.4     # ex) in case of OpenMPI-1.4.4
mpirun -machinefile $TMPDIR/machines -np $NSLOTS ./execution_file
```

iii. Eagle HPC

1. Serial job – Eagle HPC에서는 serial job(코어 1개 사용)은 허용되지 않습니다.

2. Parallel Job

```
#!/bin/bash
#$ -V                          # job submit node의 shell 환경변수를 computing node에도 적용(default)
#$ -pe mpi_4 4                 # parallel environment(pe)로 설정, 사용자가 원하는 core수 만큼 입력
#$ -N parallel_job             # Job name. 명시하지 않으면 job_script이름을 가져옵니다.
```

```
#$ -q eagle-short.q        # 사용할 queue name
#$ -S /bin/bash            # shell selection users want to use
#$ -cwd                    # 현재 directory를 작업 directory로 사용
#$ -l h_rt=24:00:00        # 작업 경과시간(hh:mm:ss)(wall time clock).
                           # 미 기입 시 작업이 강제 종료 됩니다.
                           # job이 wall time clock에 도달 시 자동으로 중지됩니다.

. /etc/profile.d/modules.sh
module load mpi/intel-11.1/openmpi-1.4.4     # ex) in case of OpenMPI-1.4.4
mpirun -machinefile $TMPDIR/machines -np $NSLOTS ./execution_file
```

D. 어플리케이션(프로그램)별 작업 스크립트 예제

i. ANSYS CFD (FLUENT)

Eagle HPC 예제 (Leopard, Lion HPC도 거의 비슷합니다.)

```
#!/bin/bash
#$ -V
#$ -pe mpi_12 12
#$ -N MXTTA
#$ -q eagle-short.q
#$ -S /bin/bash
#$ -cwd
#$ -l h_rt=24:00:00

. /etc/profile.d/modules.sh
module load ansys/17.0
module load mpi/intel-14.0/openmpi-1.8.1

INPUT_FILE=MyFluentJobInputFileName
```

```
Fluent 3ddp -t$NSLOTS -g -cnf=$TMPDIR/machines -sge -pinfiniband -mpi=openmpi -i $INPUT_FILE
```

ii. COMSOL Multiphysics

```
#!/bin/bash
#$ -V
#$ -pe mpi_8 16
#$ -N CMSXMPL
#$ -q eagle-short.q
#$ -S /bin/bash
#$ -cwd
#$ -l h_rt=24:00:00

. /etc/profile.d/modules.sh
module load comsol/53

NUM_CORE=$(echo $PE | cut -d '_' -f2)   ## Number of cores to be used by each compute node
NUM_NODE=`expr $NSLOTS / $NUM_CORE`     ## Total number of compute nodes
NUM_PROC=$(nproc)                       ## Total number of processors per compute node

comsol batch -nn $NUM_NODE -np $NUM_CORE -f $TMPDIR/machines -inputfile MyInputFileName -
outputfile MyOutputFileName -batchlog MyLogFileName -tmpdir /uwork/p0xxxxx/
```

iii. ABAQUS

```
#!/bin/bash
#$ -V
#$ -pe mpi_10 10
#$ -N CMSXMPL
#$ -q eagle-short.q
#$ -S /bin/bash
#$ -cwd
```

```
#$ -l h_rt=24:00:00

. /etc/profile.d/modules.sh
module load abaqus/2017

# Define particulars of this run:
INPUT_FILENAME=AQS.inp
JOBNAME=${JOB_NAME}
ABAQUS_ARGS=
SCRATCH_DIR=PathForTheTempraryFile
#
# To manage abaqus jobs, you need to catch signals
# and use "abaqus terminate" to stop the job
#
exit_gracefully () {
abaqus terminate job=$JOBNAME
echo Abaqus job $JOBNAME terminated
exit
}

# invoke abaqus in the background on the compute node:
trap exit_gracefully SIGUSR2

abaqus cpus=$NSLOTS mp_mode=mpi job=$JOBNAME input=$INPUT_FILENAME scratch=$SCRATCH_DIR
$ABAQUS_ARGS

# Report some useful info
/bin/uname \-a

#
# now sleep until lock file disappears
#
sleep 60
while [ -f ${JOBNAME}.lck ]; do
  sleep 5
done
```

last updated October 19th, 2017 by Sangmin Park

UNIST HPC User Guide

1. HPC Resources

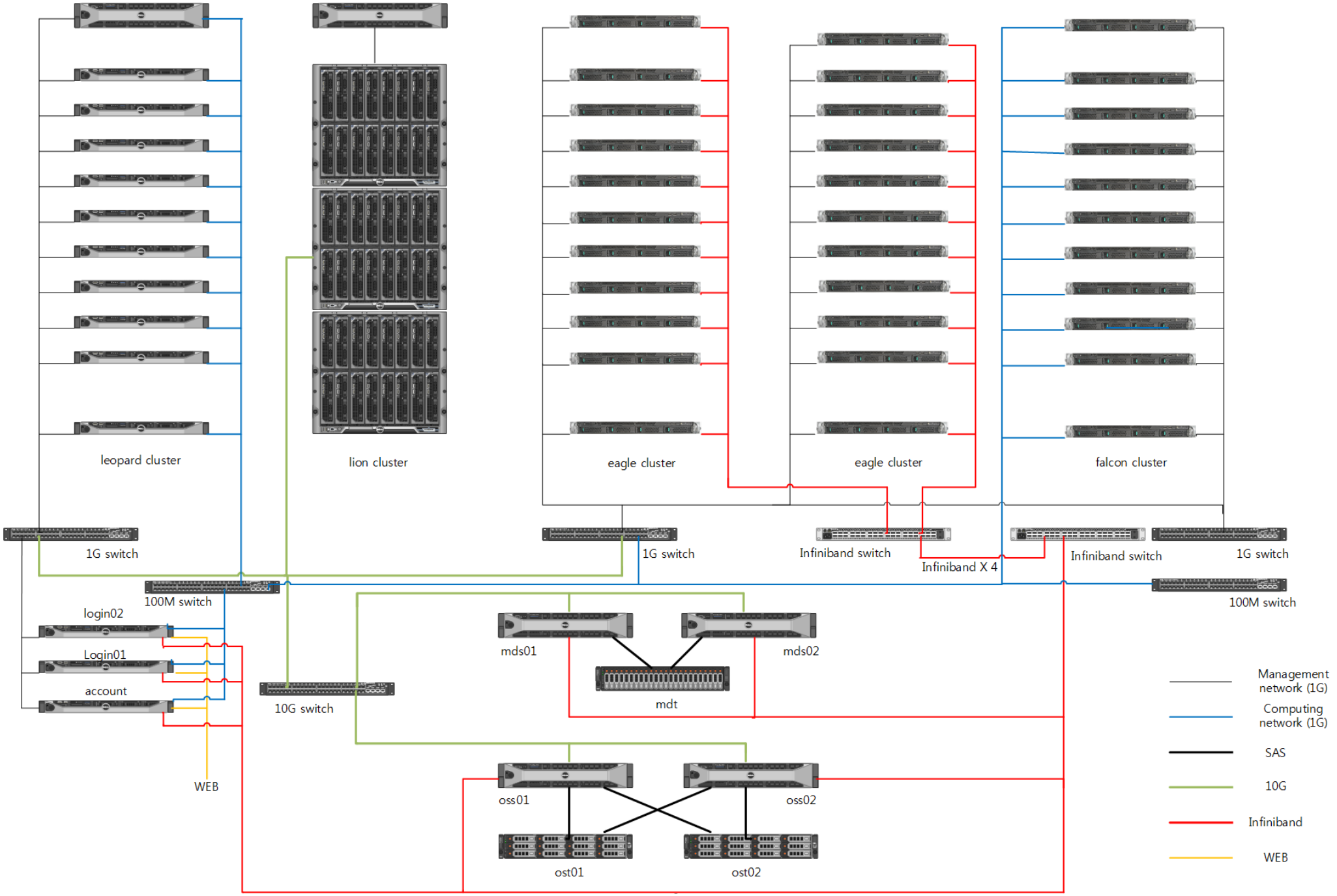
A. Hardware

i. Configuration Overview

UNIST Supercomputing Center(USC) has several High Performance Computing(HPC) systems and parallel file system to support a variety of researches that find the secret in the universe. For those, we have four HPC x86 Linux clusters named it leopard, lion, falcon, and eagle and Lustre parallel file system to be able to support high I/O throughput calculations.

At first, the leopard system was installed 2011. It consists of 28 computing nodes and each node has 8 cores. The microarchitecture of leopard system is Nehalem. After leopard, we launched lion cluster. The lion is blade type. It has 41 computing nodes. The lion system consists of 12, 16, and 20 cores computing nodes. The falcon and the eagle system were installed one after another. Those systems have newest specification. The falcon consists of 16 computing nodes and the eagle cluster has 31 computing nodes. Each computing node in those systems has 20 cores that are Intel Ivy-bridge microarchitecture.

Total number of cores in all HPC cluster are approximately 1,800. The HPC systems vary in the interconnection. Those are connected from the 1G gigabit to the 40G infiniband connection. Detailed HPC system configuration in USC could be shown by the following figure.


leopard cluster
lion cluster
eagle cluster
eagle cluster
falcon cluster
1G switch
1G switch
Infiniband switch
Infiniband X 4
Infiniband switch
1G switch
100M switch
100M switch
login02
Login01
account
WEB
10G switch
mds01
mds02
mdt
oss01
oss02
ost01
ost02
Management network (1G)
Computing network (1G)
SAS
10G
Infiniband
WEB

ii. How to access HPC systems

1. All paths via LOGIN nodes

We have two login nodes to access HPC systems. Login nodes are the main gateway to access to HPC systems. So, users need to go through the login nodes to access HPC systems. (2022 port)

hostname	login01	login02
DNS	**login01.usc.unist.ac.kr**	**login02.usc.unist.ac.kr**

2. After a successful connection to login nodes

Users can access to all of HPC systems using Secure Shell(SSH) with passwordless. Just type 'ssh hostname'. That's all.

iii. Disk quota

USC has parallel file system that is made up by Lustre. So, all user's home and work directory locates in the parallel file system server. It is attached to the high performance network.

1. home directory

User's home directory is located under /uhome. And all users have up to 50GB home disk quota and can check how much a user takes disk space use this command.

```
lfs quota -u [username] -h [user's home directory path]
ex) lfs quota -u smpark -h /uhome/smpark
```

All users' home directories are backed up for data safety.

2. work directory

The work directory has no limitation for disk usage. However, it is not backed up unlike home directory. All users have work directory like home directory. Comman path of work directory is /uwork/$USER.

iv. HPCs

1. Leopard

A. System Overview

Hostname	leopard
Number of node(# of core)	28 node (224 cores), 8cores/node
Processor	Intel Xeon E5540 2.53GHz
Memory	24GB per node
OS	Linux CentOS 6.5
Interconnection	1G Ethernet between computing nodes 10G network to parallel file system from leopard switch
Storage	/uhome, /uwork in Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs
leopard-short.q	48 hours	-	leopard-short
leopard-normal.q	168 hours	-	leopard-normal
leopard-long.q	504 hours	80	leopard-long
leopard-serial.q	no limitation	16	-

leopard-short.q has the highest priority. The longer wall clock time is, the lower priority is.

ii. Parallel Environment(PE)s

PEs are high coupled with queue name. A user has to use leopard-normal pe when submitting a job using leopard-normal.q. In the case of leopard-long.q , should be leopard-long pe.

Also, very careful thing is here.

Users have to use the number of cores same as physical cores in the computing nodes. The leopard system has 8 cores per node. So, users have to use multiple of the number of cores, 8.

iii. Resource quota

For the fair usage of HPCs, there is a resource quota policy. Users can not take over 40 cores simultaneously in the parallel job nodes and 12 cores in sequential job nodes. The leopard27 and leopard28 are for only sequential job. It's the policy for the restriction of resource exclusive use.

2. Lion

A. System Overview

Hostname	lion

Number of node(# of core)	41 nodes (584 cores), 12cores(25nodes), 16cores(9nodes), 20cores(7nodes)
Processor	Intel Xeon X5650 2.66GHz, Xeon E5-2670 2.60GHz, E5-2670v2 2.50GHz
Memory	24GB(25nodes) + 64GB(16nodes)
OS	Linux CentOS 6.5
Interconnection	1G Ethernet between computing nodes 10G network to parallel file system from lion switch
Storage	/uhome, /uwork in the Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs	notes
lion-short.q	48 hours	-	lion-short	
lion-normal.q	168 hours	-	lion-normal	
lion-long.q	504 hours	80	lion-long	
lion-octa.q	unlimited	-	mpi_1 ~ mpi_16	these queues are allowed to specific groups
lion-deca.q	unlimited	-	mpi_1 ~ mpi_20	
lion-serial.q	unlimited	12	-	

ii. Parallel Environment(PE)s

PEs are high coupled with queue name. A user has to use lion-normal pe when submitting a job using lion-normal.q.

In the case of lion-long.q , should be lion-long pe. In case of lion-octa.q and lion-deca.q does not follow this policy.

Also, very careful thing is here.

Users have to use the number of cores same as physical cores in the computing nodes. The lion system has 12 cores per node from lion01~lion25. So, users have to use multiple of the number of cores, 12 in these nodes. From lion26, this rule does not be applied.

iii. Resource quota

For the fair usage of HPCs, there is a resource quota policy. Users can not take over 48 cores simultaneously in the parallel job nodes and 18 cores in sequential job nodes. The lion01 and lion02 are for sequential job only. It's the policy for the restriction of resource exclusive use.

3. Falcon

A. System Overview

Hostname	**falcon**
Number of node(# of core)	16 nodes (320 cores)
Processor	Intel Xeon E5-2690v2 3.00GHz
Memory	128GB
OS	Linux CentOS 6.5
Interconnection	1G Ethernet between computing nodes 1G network to parallel file system from lion switch

Storage	/uhome, /uwork in the Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs
falcon-short.q	unlimited	-	mpi_1 ~ mpi_20, falcon-short
falcon-normal.q	unlimited	-	mpi_1 ~ mpi_20, falcon-normal
falcon-long.q	unlimited	-	mpi_1 ~ mpi_20, falcon-long

ii. Parallel Environment(PE)s

Users can take parallel environment from mpi_1 to mpi_20 and falcon-short/normal/long. falcon-short/normal/long PEs have core allocation rules, the number of core using calculation should be same as physical cores. 20, 40, 60 …

iii. Notices

Falcon cluster is for exclusive groups. It will take some time to open for general users.

4. Eagle

A. System Overview

Hostname	eagle
Number of node(# of core)	31 nodes (620 cores)

Processor	Intel Xeon E5-2690v2 3.00GHz
Memory	128GB
OS	Linux CentOS 6.5
Interconnection	Infiniband(40G) network between computing nodes Infiniband network to the parallel file system
Storage	/uhome, /uwork in the Lustre parallel file system

B. Job Queues and PEs

i. Queues

queue name	wall clock time	available cores	available PEs
eagle-short.q	24 hours	-	mpi_2 ~ mpi_20
eagle-normal.q	48 hours	-	mpi_2 ~ mpi_20

ii. Parallel Environment(PE)s

mpi_2 ~ mpi_20 pe are available to run a job. You cannot run a serial job using a single core in an Eagle cluster. The usage syntax of mpi is mpi_ [digit1] [digit2], and [digit1] can be input from 2 to 20. [Digit2] refers to the total core that are used during the computation and [digit1] refers to the number of cores per node to be used in the calculation. [Digit1] must be less than [digit2] and must be an aliquot part of [digit2]. (Ex: mpi_5 5, calculated by using 5 cores for the calculation in a single node, and 6 cores for two nodes if you use mpi_6 12)

C. Quota

The resource allocation policy is set to prevent exclusive use of HPC resources. The quota is given by research group quota, and for parallel computational nodes up to 160 cores per group.

D. Notice

Eagle cluster has Infiniband network that is able to communicate between computing nodes very fast, 40GB/s. It is 40x faster than Ethernet network theoretically. So, user does not need to match up the number of cores with the physical number of cores like leopard and lion cluster.

2. Software

A. Compilers

Compilers	Installed system
GNU 4.4.7	leopard, lion, falcon, eagle
Intel composer xe 2013	
PGI 14.3	lion

B. Libraries

Classification	Libraries	Installed systems

Parallel libraries	OpenMPI 1.6.5	leopard, lion, falcon, eagle
	OpenMPI 1.8.1	
	MPICH-1.2.7p1	
	MPICH2-1.4.1p1	
	MPICH-3.1.2	lion, falcon
Mathematical libraries	FFTW 2.1.5	leopard, lion, falcon, eagle
	FFTW 3.3.4	
	Intel MKL	

C. Computational Packages

Classification	Applications	Installed systems
Commercial	ABAQUS 6.14-2	leopard, lion, falcon, eagle
	ANSYS CFD, Fluent	
	COMSOL Multiphysics 5.0	
	VASP 5.3	
	Gaussian 09	
Open Source	GROMACS	leopard, lion, falcon, eagle
	LAMMPS	
	Quantum Espresso	
	R Source	

※ NOTICE : When you want to use a commercial package for your research, please contact us first, due to prevent the illegal usage of license.

3. Environment Settings

A. Overview

USC has implemented module environment to manage users' environment for all software installed in our HPC systems. Users are required very often to define some variables that affect the environment on each HPCs. And the operation of the program is executed under this environment. These environment variables are used to inform the HPC system the location of execution files, documentation, or related libraries. This would make new user or users unfamiliar with UNIX-like system very annoying. Through the module approach, however, users are no longer painful with configuring user's environment. The scripts(modulefiles) are made by the staff and users simply "load" and "unload" modules to configure the environment, also user can make individual module files.

Environment module approach updates users' environment very easily, especially **PATH**,**LD_LIBRARY_PATH**, **MANPATH** variables, etc.

When your account is made in HPC systems, "dot-files" should be set up for user's environment(default – bash shell). You can modify this "dot-files" to configure your environment firsthand. But, there is a way to set your environment up more conveniently.

B. Module commands

To know how to use environment module approach, type the following:

```
$ module help
```

"help" option displays as below. (not full result, select useful options)

```
Available SubCommands and Args:
 + add|load            modulefile [modulefile ...]
 + rm|unload           modulefile [modulefile ...]
```

```
+ switch|swap      [modulefile1] modulefile2
+ display|show     modulefile [modulefile ...]
+ avail           [modulefile [modulefile ...]]
+ purge
+ list
```

$ module list

prints a list of the currently loaded modulefile.

$ module avail

lists all the modulefiles which are available to be loaded.

$ module purge

unloads all loaded modulefiles currently.

$ module load *modulefile*

loads *the modulefile.*

$ module unload *modulefile*

unload *the modulefile.*

$ module switch modulefile_old modulefile_new

switches *the modulefile_old* with *modulefile_new*.

$ module show *modulefile*

displays how the *modulefile* changes the environment such as PATH, LD_LIBRARY_PATH, MANPATH, etc.

4. Job Submission

A. Overview

USC uses the queuing system to run users' code. We are using SGE(Sun Grid Engine) to handle user's job. This makes HPCs more effective and gives users convenience within HPCs use.

B. SGE commands

Commands	Example	Description
qsub	qsub job_script_file_name	submit a job.
qstat	qstat	show job status oneself
	qstat -u '*'	show job status for all users
qhost	qhost	show computing node status
qdel	qdel job_ID qdel -u user_ID (user's all job cancelled)	cancel a job
qconf	qconf -sql	show all queue list
	qconf -spl	show all pe list
	qconf -sq short.q	show about short.q detailed
	qconf -sul	show all user list
	qconf -srqs	show resource quota policy
	qconf -shgrpl	show host group list
	qconf -sc	show complex attributes

C. Job Script Examples

i. Leopard HPC

1. Serial Job (using single core)

Serial job in Leopard HPC is allowed to the compute node of leopard27 and leopard28.

```
#!/bin/bash
#$ -V                                  # exporting environment of master node to the compute nodes .
#$ -cwd                                # current working directory .
#$ -N serial_job                       # Job Name. if you don't set this option,
                                       # default value is job_script name.
#$ -q leopard-serial.q                 # queue name
#$ -S /bin/bash                        # shell selection
#$ -wd /uhome/<user01>/serialtest      # set working directory.
                                       # if you are not in /uwork/<user01> directory then
                                       # set option -wd /uwork/<user01>
                                       # or else it's okay to set option -cwd (current working directory).
#$ -l h_rt=01:00:00                    # resource time for job. (hh:mm:ss)(wall time clock).
                                       # if you don't set this option, Job won't work.

. /etc/profile.d/modules.sh
module load intel/mkl-10.1.3
./execution_file_name
```

2. Parallel Job (more than 2 cores)

```
#!/bin/bash
#$ -V                          # exporting environment of master node to the compute nodes (default).
#$ -pe leopard-short 8         # set parallel environment(pe), set number of cores you want.
#$ -N parallel_job             # Job Name. if you don't set this option,
                               # default value is job_script name.
#$ -q leopard-short.q          # queue name
#$ -S /bin/bash                # shell selection users want to use
#$ -cwd                        # use current directory as working directory.
#$ -l h_rt=24:00:00            # resource time for job. (hh:mm:ss)(wall time clock).
                               # if you don't set this option, Job won't work.
                               # Job will be automatically stopped when it reaches the wall time clock.

. /etc/profile.d/modules.sh
module load mpi/intel-11.1/openmpi-1.4.4     # ex) in case of OpenMPI-1.4.4
mpirun -machinefile $TMPDIR/machines -np $NSLOTS ./execution_file
```

ii. Lion HPC

1. Serial job (use 1 core)

```
#!/bin/bash
#$ -V                          # exporting environment of master node to the compute nodes (default).
#$ -N serial_job               # Job Name. if you don't set this option,
                               # default value is job_script name.
#$ -q lion-serial.q            # queue name
#$ -S /bin/bash                # shell selection
#$ -wd /uhome/<user01>         # set working directory.
```

```
                                    # if you are not in /uwork/<user01> directory then
                                    # set option -wd /uwork/<user01>
                                    # or else it's okay to set option -cwd (current working directory).
#$ -l h_rt=01:00:00                 # resource time for job. (hh:mm:ss)(wall time clock).
                                    # if you don't set this option, Job won't work.

. /etc/profile.d/modules.sh
module load intel/mkl-10.1.3
./execution_file_name
```

2. Parallel job (more than 2 cores)

```
#!/bin/bash
#$ -V                       # exporting environment of master node to the compute nodes (default).
#$ -pe lion-short 12        # set parallel environment(pe), set number of cores you want.
#$ -N parallel_job          # Job Name. if you don't set this option,
                            # default value is job_script name.
#$ -q lion-short.q          # queue name
#$ -S /bin/bash             # shell selection users want to use
#$ -cwd                     # use current directory as working directory.
#$ -l h_rt=24:00:00         # resource time for job. (hh:mm:ss)(wall time clock).
                            # if you don't set this option, Job won't work.
                            # Job will be automatically stopped when it reaches the wall time clock.

. /etc/profile.d/modules.sh
module load mpi/intel-11.1/openmpi-1.4.4     # ex) in case of OpenMPI-1.4.4
mpirun -machinefile $TMPDIR/machines -np $NSLOTS ./execution_file
```

iii. Eagle HPC

1. Serial job – Eagle HPC doesn't allow serial job.

2. Parallel Job

```
#!/bin/bash
#$ -V                        # exporting environment of master node to the compute nodes (default).
#$ -pe mpi_4 4               # set parallel environment(pe), set number of cores you want.
#$ -N parallel_job           # Job Name. if you don't set this option,
                             # default value is job_script name.
#$ -q eagle-short.q          # queue name
#$ -S /bin/bash              # shell selection users want to use
#$ -cwd                      # use current directory as working directory.
#$ -l h_rt=24:00:00          # resource time for job. (hh:mm:ss)(wall time clock).
                             # if you don't set this option, Job won't work.
                             # Job will be automatically stopped when it reaches the wall time clock.

. /etc/profile.d/modules.sh
module load mpi/intel-11.1/openmpi-1.4.4     # ex) in case of OpenMPI-1.4.4
mpirun -machinefile $TMPDIR/machines -np $NSLOTS ./execution_file
```

D. Example for each applications job script

i. ANSYS CFD (FLUENT)

Example for Eagle HPC (similar with Leopard, Lion HPC)

```
#!/bin/bash
#$ -V
#$ -pe mpi_12 12
```

```
#$ -N MXTTA
#$ -q eagle-short.q
#$ -S /bin/bash
#$ -cwd
#$ -l h_rt=24:00:00


. /etc/profile.d/modules.sh
module load ansys/17.0
module load mpi/intel-14.0/openmpi-1.8.1

INPUT_FILE=MyFluentJobInputFileName

Fluent 3ddp -t$NSLOTS -g -cnf=$TMPDIR/machines -sge -pinfiniband -mpi=openmpi -i $INPUT_FILE
```

ii. COMSOL Multiphysics

```
#!/bin/bash
#$ -V
#$ -pe mpi_8 16
#$ -N CMSXMPL
#$ -q eagle-short.q
#$ -S /bin/bash
#$ -cwd
#$ -l h_rt=24:00:00


. /etc/profile.d/modules.sh
module load comsol/53

NUM_CORE=$(echo $PE | cut -d '_' -f2)   ## Number of cores to be used by each compute node
NUM_NODE=`expr $NSLOTS / $NUM_CORE`     ## Total number of compute nodes
NUM_PROC=$(nproc)                       ## Total number of processors per compute node

comsol batch -nn $NUM_NODE -np $NUM_CORE -f $TMPDIR/machines -inputfile MyInputFileName -
outputfile MyOutputFileName -batchlog MyLogFileName -tmpdir /uwork/p0xxxxx/
```

iii. ABAQUS

```
#!/bin/bash
#$ -V
#$ -pe mpi_10 10
#$ -N CMSXMPL
#$ -q eagle-short.q
#$ -S /bin/bash
#$ -cwd
#$ -l h_rt=24:00:00

. /etc/profile.d/modules.sh
module load abaqus/2017

# Define particulars of this run:
INPUT_FILENAME=AQS.inp
JOBNAME=${JOB_NAME}
ABAQUS_ARGS=
SCRATCH_DIR=PathForTheTempraryFile
#
# To manage abaqus jobs, you need to catch signals
# and use "abaqus terminate" to stop the job
#
exit_gracefully () {
abaqus terminate job=$JOBNAME
echo Abaqus job $JOBNAME terminated
exit
}

# invoke abaqus in the background on the compute node:
trap exit_gracefully SIGUSR2

abaqus cpus=$NSLOTS mp_mode=mpi job=$JOBNAME input=$INPUT_FILENAME scratch=$SCRATCH_DIR
$ABAQUS_ARGS
```

```
# Report some useful info
/bin/uname \-a

#
# now sleep until lock file disappears
#
sleep 60
while [ -f ${JOBNAME}.lck ]; do
  sleep 5
done
```

last updated October 19th, 2017 by Sangmin Park