
SKL 224 node Documentation

CINES

Nov 27, 2018

Contents

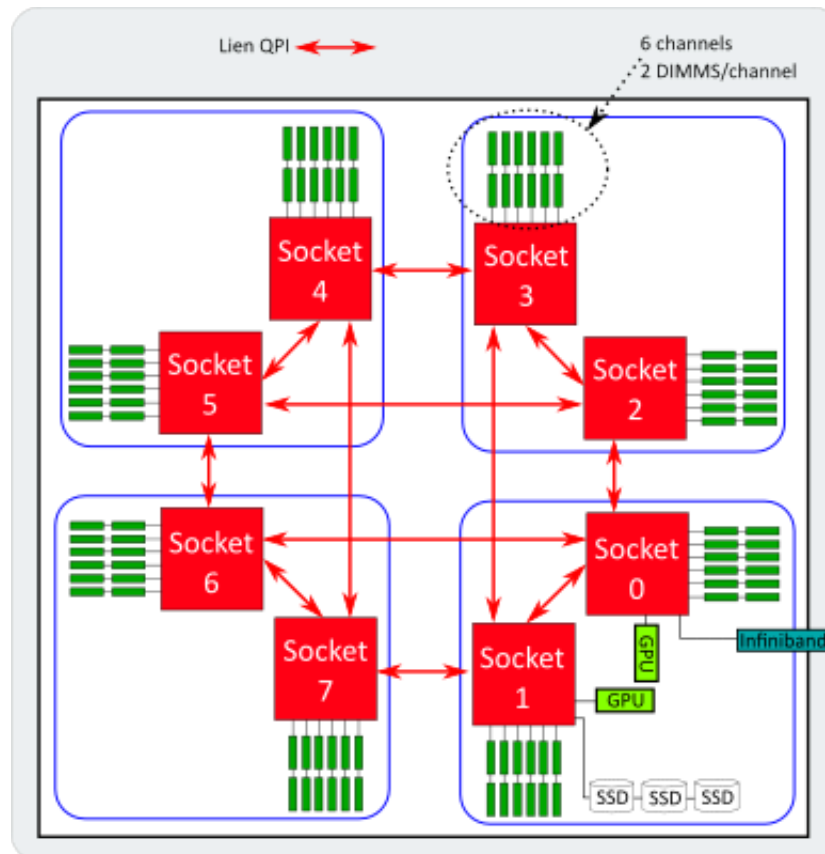
1	About the SKL224 node	2
2	Get the access	2
3	File systems	3
4	Module	3
5	Usage modes	3
5.1	MPI	3
5.2	OpenMP	4
5.3	Hybrid	5

1 About the SKL224 node

Since November 2018, occigen has been integrating a Bull x808 node with:

- 3TB of RAM accessible in NUMA (glueless interconnect) by 8 sockets Intel® Xeon® Platinum 8176 (224 cores total).
- 2 Nvidia GPU Tesla P100 (12GB).

Here is the simplified representation of this node:



2 Get the access

To access the node, please [contact us](#), defining the need that motivates you to use this resource.

Warning: the 224 cores are invoiced because the full node is dedicated to your job.

Here are some examples of how to use such a machine:

- large memory or non-optimal jobs on thin nodes
- production / processing of images, meshes and data analysis outside interactive visualization
- Proof of Concept: GPU software testing not used in a production mode

Note that in the same batch job, you cannot request this SKYLAKE node together with other HASWELL, BROADWELL, or visualization nodes.

3 File systems

This node can access 4 disk spaces:

- \$HOME
- \$SCRATCHDIR
- \$STORDIR
- /tmp

A full description of the three first file systems can be found [here](#)

/tmp is local and take advantage of SSD disks. This volatile workspace can make you benefit of a high performance 1.2TB storage.

Warning: this workspace is not defined to store data on a long term. If you want to keep your data between jobs, please use secured filesystems as \$HOME or \$STOREDIR.

4 Module

Software environment: module

CINES enables the use of multiple softwares on its clusters, with multiple versions.

To avoid conflicts between different versions of the same software, we usually need to define a specific environment for a given version.

Available softwares can be handled using the following commands:

module avail	show the list of available softwares
module load	load a software to your environment
module list	list the loaded modules
module purge	remove all loaded modules
module show	show information of the module

Si vous ne trouvez pas le logiciel dont vous avez besoin dans la liste, [contactez nous](#).

5 Usage modes

The node can be used in three different ways:

- MPI only
- OpenMP only
- Hybrid MPI+OpenMP

Example scripts are available in the following sections.

5.1 MPI

```

#!/bin/bash
#SBATCH -J skl224_mpi
#SBATCH --nodes=1
#SBATCH --ntasks=224
#SBATCH --ntasks-per-node=224
#SBATCH --time=0:40:00
#SBATCH -C SKL224
#SBATCH --exclusive
#SBATCH --output skl224_mpi.output.slurm

set -e

#####Intelmpi placement auto
# module load intel/18.1 intelmpi/2018.1.163
# export I_MPI_DOMAIN=auto
# export I_MPI_PIN_RESPECT_CPUSET=0
# ulimit -s unlimited
# srun ../../../../bin/hello_mpi

#####Intelmpi avec placement pour mpirun
# module load intel/18.1 intelmpi/2018.1.163
# export SLURM_CPU_BIND=NONE
# export I_MPI_PIN=1
# export I_MPI_PIN_PROCESSOR_LIST=0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,
↪20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,
↪48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,
↪76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,
↪103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,
↪124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,
↪145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,
↪166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,
↪187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,
↪208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223
# ulimit -s unlimited
# mpirun ../../../../bin/hello_mpi

#####Openmpi placement auto
module load intel/18.1 openmpi/intel/2.0.2
ulimit -s unlimited
srun ../../../../bin/hello_mpi

```

5.2 OpenMP

```

#!/bin/bash
#SBATCH -J skl224_omp
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --ntasks-per-node=1
#SBATCH --cpus-per-task=224
#SBATCH --time=0:40:00
#SBATCH -C SKL224
#SBATCH --exclusive
#SBATCH --output skl224_omp.output.slurm

```

(continues on next page)

(continued from previous page)

```
set -e

#Make sure that OMP_NUM_THREADS = cpus-per-task * KMP_HW_SUBSET
export KMP_HW_SUBSET=1T
export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK
export KMP_AFFINITY=verbose,compact,1,0,granularity=fine

module load intel

ulimit -s unlimited

rm -f *.out

srun ../../../../bin/hello_omp
```

5.3 Hybrid

```
#!/bin/bash
#SBATCH -J skl224_hybrid
#SBATCH --nodes=1
#SBATCH --ntasks=8
#SBATCH --ntasks-per-node=8
#SBATCH --cpus-per-task=28
#SBATCH --time=0:40:00
#SBATCH -C SKL224
#SBATCH --exclusive
#SBATCH --mem=50GB
#SBATCH --output skl_hybrid.output.slurm

set -e

#####Intelmpi
# module load intel intelmpi
# export I_MPI_DOMAIN=auto
# export I_MPI_PIN_RESPECT_CPuset=0
# #Make sure that OMP_NUM_THREADS = cpus-per-task * KMP_HW_SUBSET
# export KMP_HW_SUBSET=1T
# export OMP_NUM_THREADS=12
# export KMP_AFFINITY=verbose,compact,1,0,granularity=fine
# ulimit -s unlimited
# srun ../../../../bin/hello_hybrid

#####Openmpi
module load intel/18.1 openmpi/intel/2.0.2
#Make sure that OMP_NUM_THREADS = cpus-per-task * KMP_HW_SUBSET
export KMP_HW_SUBSET=1T
export OMP_NUM_THREADS=28
export KMP_AFFINITY=verbose,compact,1,0,granularity=fine
ulimit -s unlimited
srun ../../../../bin/hello_hybrid
```