プロセスとスレッドの適切な割り当て
ノードグループC編

この資料について

- 玄界の各計算ノードグループには複数のCPUやGPUやNICが搭載されている
- 最大性能を得るためには、それらの配置を考えたプログラムの実行が必要

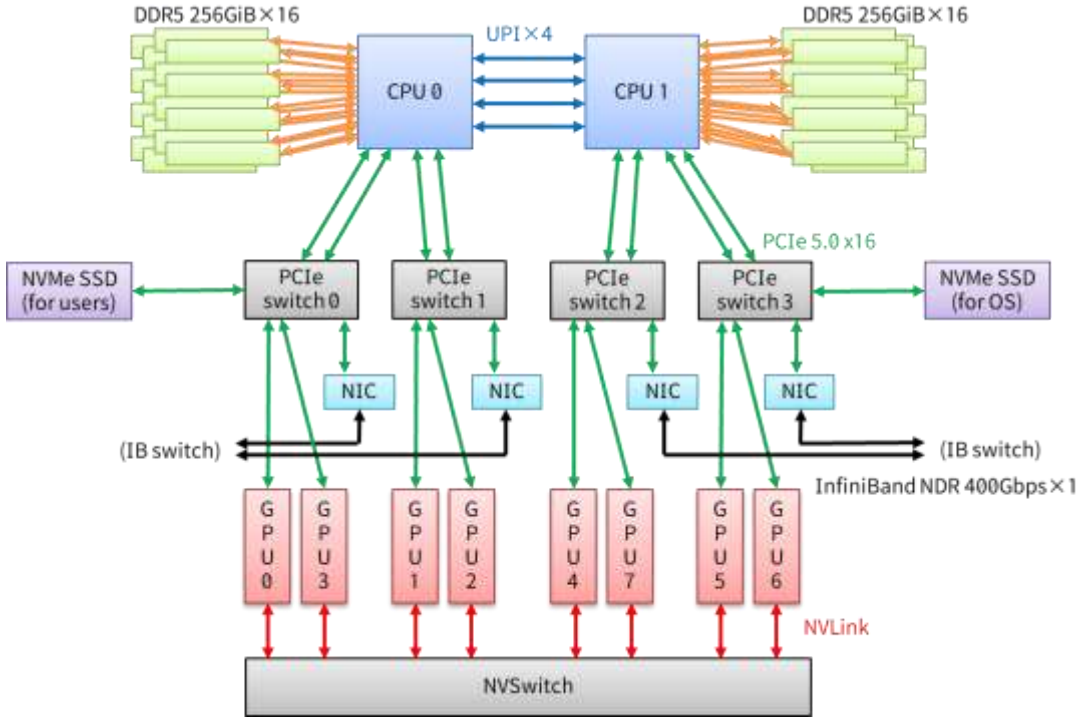
- この資料ではプロセスやスレッドの割り当てを最適化するためのヒントについて説明する

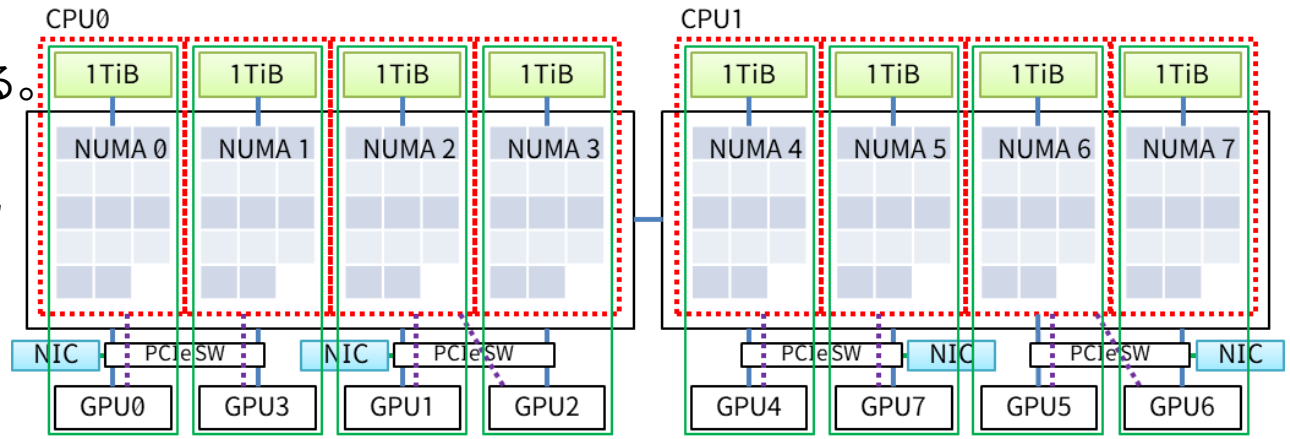
- ノードグループC編では、ノードグループB編を読んであることを前提として、ノードグループCに特有の内容のみを説明する

- 2025年3月
  - ノードグループCのNUMA構成に誤りがあったため修正

基本的な構成の理解

- 上図：ノードグループCのCPU・GPU・メモリ・NICの物理的な構成
  - 1基のCPUに2つのPCIe switchがつながっており、そこから1基のNICと2基のGPUに接続
  - GPU同士はNVSwitchで双方向に高速接続
- 下図：CPUの内部はHW的には4つのグループ（NUMAノード）に分かれている
  - 1NUMAノードと1GPUがペアになっていると思えば良いが、PCI-Expressの構成上の都合で偏りがある点に注意。
    - GPUとNUMAの順番が異なっており、一部のNUMAには複数のGPUが接続されている。ノード共有ジョブ実行時には細い緑枠のメモリ・CPU・GPUが1単位で割り当たる。
  - GPUによる計算とGPU間直接通信（NVSwitchによりどのCPU間も同等に高速）を中心に考えるなら、プロセスの配置などはあまり考えなくて良いかもしれない。
- 実行時には `numactl -H`, `numactl -s`, `lstopo`, `nvidia-smi topo -m` などで構成を確認可能





CPUのコア数が他ノードグループと異なり56である点にも注意（NUMAノードあたりCPUコア数は14）

ユーザからはどう見える？（numactl、1ノード占有ジョブ実行時）

- CPU2ソケットで合計8つのNUMAノードが確認できる

```
[ku40000105@c0002 ~]$ numactl -H
available: 8 nodes (0-7)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13
node 0 size: 1031298 MB
node 0 free: 1029564 MB
node 1 cpus: 14 15 16 17 18 19 20 21 22 23 24 25 26 27
node 1 size: 1032187 MB
node 1 free: 1030962 MB
node 2 cpus: 28 29 30 31 32 33 34 35 36 37 38 39 40 41
node 2 size: 1032187 MB
node 2 free: 1030639 MB
node 3 cpus: 42 43 44 45 46 47 48 49 50 51 52 53 54 55
node 3 size: 1032187 MB
node 3 free: 1031094 MB
node 4 cpus: 56 57 58 59 60 61 62 63 64 65 66 67 68 69
node 4 size: 1032187 MB
node 4 free: 1030317 MB
node 5 cpus: 70 71 72 73 74 75 76 77 78 79 80 81 82 83
node 5 size: 1032187 MB
node 5 free: 1029920 MB
node 6 cpus: 84 85 86 87 88 89 90 91 92 93 94 95 96 97
node 6 size: 1032187 MB
node 6 free: 1030638 MB
node 7 cpus: 98 99 100 101 102 103 104 105 106 107 108 109 110 111
node 7 size: 1032141 MB
node 7 free: 1031167 MB
右上へ続く
```

0から3がCPU0、4から7がCPU1

```
node distances:
node   0   1   2   3   4   5   6   7
  0:  10  12  12  12  21  21  21  21
  1:  12  10  12  12  21  21  21  21
  2:  12  12  10  12  21  21  21  21
  3:  12  12  12  10  21  21  21  21
  4:  21  21  21  21  10  12  12  12
  5:  21  21  21  21  12  10  12  12
  6:  21  21  21  21  12  12  10  12
  7:  21  21  21  21  12  12  12  10

[ku40000105@c0002 ~]$ numactl -s
policy: default
preferred node: current
physcpubind: 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
cpubind: 0 1 2 3 4 5 6 7
nodebind: 0 1 2 3 4 5 6 7
membind: 0 1 2 3 4 5 6 7
```

112コア、8NUMAノード

ユーザからはどう見える？（numactl、1GPUジョブ実行時）

- 確認方法によって見え方が違う

```
[ku40000105@c0002 ~]$ numactl -H
available: 8 nodes (0-7)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13
node 0 size: 1031283 MB
node 0 free: 997835 MB
node 1 cpus: 14 15 16 17 18 19 20 21 22 23 24 25 26 27
node 1 size: 1032187 MB
node 1 free: 1029434 MB
node 2 cpus: 28 29 30 31 32 33 34 35 36 37 38 39 40 41
node 2 size: 1032187 MB
node 2 free: 998962 MB
node 3 cpus: 42 43 44 45 46 47 48 49 50 51 52 53 54 55
node 3 size: 1032187 MB
node 3 free: 955597 MB
node 4 cpus: 56 57 58 59 60 61 62 63 64 65 66 67 68 69
node 4 size: 1032187 MB
node 4 free: 1030204 MB
node 5 cpus: 70 71 72 73 74 75 76 77 78 79 80 81 82 83
node 5 size: 1032187 MB
node 5 free: 1031236 MB
node 6 cpus: 84 85 86 87 88 89 90 91 92 93 94 95 96 97
node 6 size: 1032187 MB
node 6 free: 1029895 MB
node 7 cpus: 98 99 100 101 102 103 104 105 106 107 108 109 110 111
node 7 size: 1032141 MB
node 7 free: 1031059 MB
node distances:
node   0   1   2   3   4   5   6   7
  0:  10  12  12  12  21  21  21  21
  1:  12  10  12  12  21  21  21  21
  2:  12  12  10  12  21  21  21  21
  3:  12  12  12  10  21  21  21  21
  4:  21  21  21  21  10  12  12  12
  5:  21  21  21  21  12  10  12  12
  6:  21  21  21  21  12  12  10  12
  7:  21  21  21  21  12  12  12  10
```

0から3がCPU0、4から7がCPU1

```
[ku40000105@c0002 ~]$ numactl -s
policy: default
preferred node: current
physcpubind: 56 57 58 59 60 61 62 63 64 65 66 67 68 69
cpubind: 4
nodebind: 4
membind: 0 1 2 3 4 5 6 7
```

numactl -s では使える資源のみが見える（14コア見える）
メモリノードだけはノード全体が見えてしまう（仕様）

numactl -Hではハードウェア情報が見えてしまうためノード占有ジョブと変化なし
（メモリ容量の表示に多少の違いがあるが気にしなくて良い）

nvidia-smi topo -m

GPU1とGPU2が同じNUMA2に接続、
GPU5とGPU6が同じNUMA6に接続、となっている。
CPU-GPU間のデータ転送に影響するはずである。

```
[ku40000105@c0002 ~]$ nvidia-smi topo -m
        GPU0    GPU1    GPU2    GPU3    GPU4    GPU5    GPU6    GPU7    NIC0    NIC1    NIC2    NIC3    CPU Affinity    NUMA Affinity   GPU NUMA ID
GPU0     X      NV18    NV18    NV18    NV18    NV18    NV18    NV18    PIX     SYS     SYS     SYS     0-13    0               N/A
GPU1    NV18     X      NV18    NV18    NV18    NV18    NV18    NV18    SYS     SYS     SYS     SYS     28-41   2               N/A
GPU2    NV18    NV18     X      NV18    NV18    NV18    NV18    NV18    SYS     PIX     SYS     SYS     28-41   2               N/A
GPU3    NV18    NV18    NV18     X      NV18    NV18    NV18    NV18    SYS     SYS     SYS     SYS     14-27   1               N/A
GPU4    NV18    NV18    NV18    NV18     X      NV18    NV18    NV18    SYS     SYS     PIX     SYS     56-69   4               N/A
GPU5    NV18    NV18    NV18    NV18    NV18     X      NV18    NV18    SYS     SYS     SYS     SYS     84-97   6               N/A
GPU6    NV18    NV18    NV18    NV18    NV18    NV18     X      NV18    SYS     SYS     SYS     PIX     84-97   6               N/A
GPU7    NV18    NV18    NV18    NV18    NV18    NV18    NV18     X      SYS     SYS     SYS     SYS     70-83   5               N/A
NIC0    PIX     SYS     SYS     SYS     SYS     SYS     SYS     SYS      X      SYS     SYS     SYS
NIC1    SYS     SYS     PIX     SYS     SYS     SYS     SYS     SYS     SYS      X      SYS     SYS
NIC2    SYS     SYS     SYS     SYS     PIX     SYS     SYS     SYS     SYS     SYS      X      SYS
NIC3    SYS     SYS     SYS     SYS     SYS     SYS     PIX     SYS     SYS     SYS     SYS      X

Legend:
省略

NIC Legend:

  NIC0: mlx5_0
  NIC1: mlx5_1
  NIC2: mlx5_2
  NIC3: mlx5_3
```

↑ノード占有ジョブ

1GPUジョブ→

```
[ku40000105@c0002 ~]$ nvidia-smi topo -m
        GPU0    NIC0    NIC1    NIC2    NIC3    CPU Affinity    NUMA Affinity   GPU NUMA
ID
GPU0     X      SYS     SYS     PIX     SYS     56-69   4               N/A
NIC0    SYS      X      SYS     SYS     SYS
NIC1    SYS     SYS      X      SYS     SYS
NIC2    PIX     SYS     SYS      X      SYS
NIC3    SYS     SYS     SYS     SYS      X
```

GPUはどう見える？：ノード占有ジョブの場合

```
[ku40000105@c0001 ~]$ nvidia-smi
+---------------------------------------------------------------------------------------+
| NVIDIA-SMI 535.154.05             Driver Version: 535.154.05   CUDA Version: 12.2     |
|-----------------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|=========================================+======================+======================|
|   0  NVIDIA H100 80GB HBM3          On  | 00000000:18:00.0 Off |                    0 |
| N/A   20C    P0              70W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+
|   1  NVIDIA H100 80GB HBM3          On  | 00000000:2A:00.0 Off |                    0 |
| N/A   19C    P0              71W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+
|   2  NVIDIA H100 80GB HBM3          On  | 00000000:3A:00.0 Off |                    0 |
| N/A   20C    P0              68W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+
|   3  NVIDIA H100 80GB HBM3          On  | 00000000:5D:00.0 Off |                    0 |
| N/A   21C    P0              70W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+
|   4  NVIDIA H100 80GB HBM3          On  | 00000000:84:00.0 Off |                    0 |
| N/A   21C    P0              72W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+
|   5  NVIDIA H100 80GB HBM3          On  | 00000000:8B:00.0 Off |                    0 |
| N/A   18C    P0              69W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+
|   6  NVIDIA H100 80GB HBM3          On  | 00000000:91:00.0 Off |                    0 |
| N/A   19C    P0              70W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+
|   7  NVIDIA H100 80GB HBM3          On  | 00000000:E4:00.0 Off |                    0 |
| N/A   19C    P0              70W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+

+---------------------------------------------------------------------------------------+
| Processes:                                                                            |
|  GPU   GI   CI        PID   Type   Process name                            GPU Memory |
|        ID   ID                                                             Usage      |
|=======================================================================================|
|  No running processes found                                                           |
+---------------------------------------------------------------------------------------+
```

```
[ku40000105@c0001 ~]$ nvidia-smi -L
GPU 0: NVIDIA H100 80GB HBM3 (UUID: GPU-18f89107-82c8-c810-9ac0-417f7bde688a)
GPU 1: NVIDIA H100 80GB HBM3 (UUID: GPU-17699e23-e834-5fe0-94ab-c21db1d4be48)
GPU 2: NVIDIA H100 80GB HBM3 (UUID: GPU-62e3611d-9d98-fe05-cf5e-aa89893d4040)
GPU 3: NVIDIA H100 80GB HBM3 (UUID: GPU-2ba8796f-07fe-bbc0-832d-cb96d94595c7)
GPU 4: NVIDIA H100 80GB HBM3 (UUID: GPU-78cdd0fd-6071-bd89-4b21-9c1c2cafab56)
GPU 5: NVIDIA H100 80GB HBM3 (UUID: GPU-b1becf31-c2ce-40c3-daa2-a800e8b13f8c)
GPU 6: NVIDIA H100 80GB HBM3 (UUID: GPU-0876220f-71ac-c4f4-f709-2d69ebd3414a)
GPU 7: NVIDIA H100 80GB HBM3 (UUID: GPU-f377ef5a-d441-8918-95c6-2a0fe733d284)
```

↑

このUUID情報（ GPU-18f89107-82c8-c810-9ac0-417f7bde688a など）をCUDA_VISIBLE_DEVICESに与えてGPUの指定に使うことができる（単純にデバイス番号を与えても良い）

もちろん複数ノードの場合はノードごとに情報が見える

GPUはどう見える？：ノード非占有ジョブの場合

- GPU数を指定したジョブの場合は指定した数分だけのGPUが見える（これは1GPUの例）

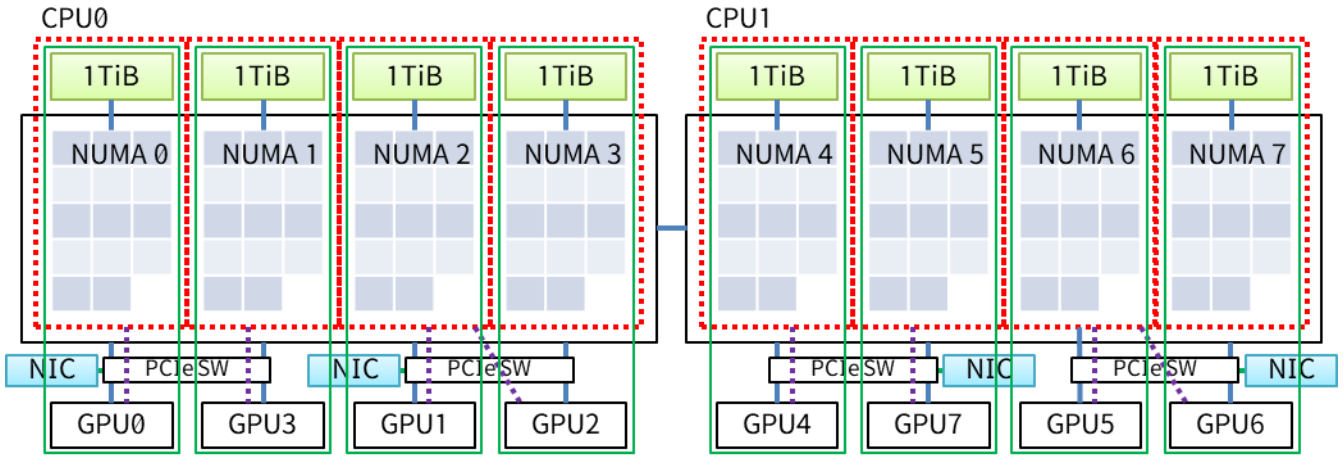
```
[ku40000105@c0001 ~]$ nvidia-smi
+---------------------------------------------------------------------------------------+
| NVIDIA-SMI 535.154.05             Driver Version: 535.154.05   CUDA Version: 12.2     |
|-----------------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|=========================================+======================+======================|
|   0  NVIDIA H100 80GB HBM3          On  | 00000000:84:00.0 Off |                    0 |
| N/A   20C    P0              71W / 700W |      0MiB / 81559MiB |      0%      Default |
|                                         |                      |             Disabled |
+-----------------------------------------+----------------------+----------------------+

+---------------------------------------------------------------------------------------+
| Processes:                                                                            |
|  GPU   GI   CI        PID   Type   Process name                            GPU Memory |
|        ID   ID                                                             Usage      |
|=======================================================================================|
|  No running processes found                                                           |
+---------------------------------------------------------------------------------------+

[ku40000105@c0001 ~]$ nvidia-smi -L
GPU 0: NVIDIA H100 80GB HBM3 (UUID: GPU-f05187ad-3328-eca7-5e38-5494aa6ce7fd)
```

ノードグループC（全体）はどう考えて使えば良い？

- NUMAノードとGPUが同数あり、NUMAノード*n*とGPU*n*は距離が近い
- GPU間はNVSwitchにより相互に高速接続

  ↓
- GPUによる処理がメインであるマルチGPUプログラムの最適化を行う場合、1NUMAノードに1MPIランク（プロセス）を配置し、そのプロセスが近い1GPUを担当するようプロセスの配置とCUDA_VISIBLE_DEVICESの指定を行えば十分

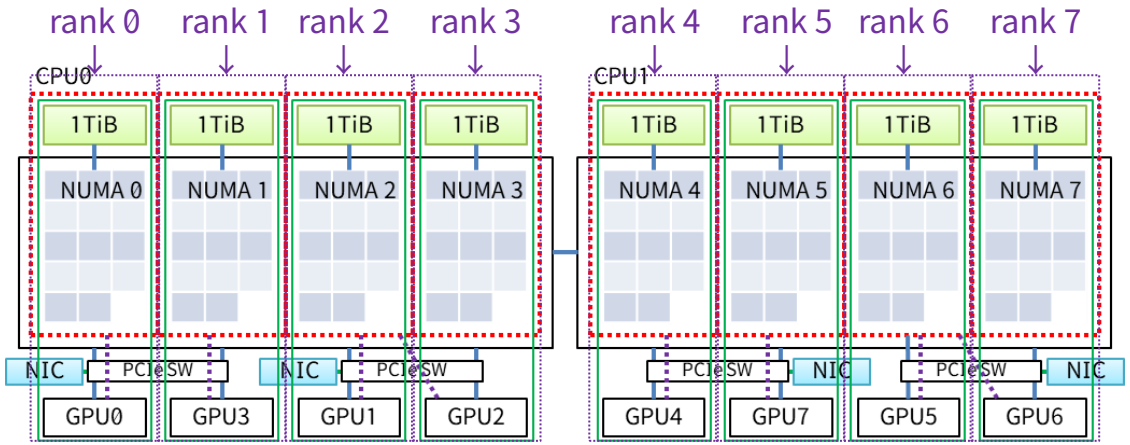
- ユーザ用NVMe SSDは1つしかないためストレージ性能まで最適化したい場合は注意が必要
  - 本資料では扱っていない
  - （後日追記する可能性あり）



具体的な例

- 割り当ての基本的な技法はノードグループB向けの資料を参照
- ノードあたり8NUMAノード・8GPUのため、8MPIプロセスが1NUMAノード・1GPUずつを使うように割り当てるのが妥当。NUMAノード番号とGPU番号の対応がずれている点に注意が必要。

```
#!/bin/bash
#PJM -L rscgrp=c-batch
#PJM -L node=1
#PJM -L elapse=10:00
module load gcc ompi ※必要に応じてCUDA関係のmoduleなども追加
export OMP_NUM_THREADS=任意スレッド数
mpirun -display-map -display-devel-map -n 8 --map-by numa ￥
--bind-to numa --rank-by numa ./runC.sh ./a.out
```



runC.sh

```
#!/bin/bash
case ${OMPI_COMM_WORLD_LOCAL_RANK} in
[0])
 export CUDA_VISIBLE_DEVICES=0;;
[1])
 export CUDA_VISIBLE_DEVICES=3;;
[2])
 export CUDA_VISIBLE_DEVICES=1;;
[3])
 export CUDA_VISIBLE_DEVICES=2;;
[4])
 export CUDA_VISIBLE_DEVICES=4;;
[5])
 export CUDA_VISIBLE_DEVICES=7;;
[6])
 export CUDA_VISIBLE_DEVICES=5;;
[7])
 export CUDA_VISIBLE_DEVICES=6;;
esac
numactl -l $@
```

-display-devel-map 指定時に得られる出力の例