GPUs en TeideHPC

GPU es el acrónimo de *Graphics Processing Unit* y representa precisamente el corazón de una tarjeta gráfica al igual que la CPU lo hace en un PC. Aparte del corazón, también es su cerebro, ya que es la encargada de realizar todos los cálculos complejos que permiten que algunos programas pueden ejecutare mucho más rápido que en una CPU.

Entre las principales usos de las GPUs están los siguientes:

- Edición de vídeo
- Renderización de gráficos 3D
- Aprendizaje automático
- Aplicaciones científicas
- etc...

El clúster TeideHPC ofrece 2 modelos diferentes de GPU para usar con sus trabajos. Recomendamos echar un vistazo a la descripción del clúster para tener una idea de cómo se ve.

Modelos de GPU disponibles

Estas son las GPU disponibles actualmente en TeideHPC:

modelos	**# nodos**	**# GPU/ nodo**	**tipo slurm**	**Cores CPU/ nodo**	**Memoria CPU/ nodo**	**Capacidad de cómputo (*)**	**Memoria GPU (GiB)**
Nvidia A100	16	4	a100	64	256GB	80	40 GB
Nvidia A100	1	8	a100	64	512GB	80	40 GB
Nvidia Tesla T4	4	1	t4	32	256GB	75	16 GB

(*) *Capacidad de cómputo o Compute Capability* es un término técnico creado por NVIDIA como una forma compacta de describir qué funciones de hardware están disponibles en algunos modelos de GPU y no en otros. No es una medida de rendimiento y solo es relevante si está compilando sus propios programas de GPU. Consulte la página sobre programación CUDA para obtener más información.

¿Qué es MIG? *(NVIDIA Multi-Instance GPU*

Multi-Instance GPU (MIG) o GPU de múltiples instancias es una tecnología de NVIDIA que permite **dividir la GPU** en hasta siete instancias completamente aisladas con su propia memoria de gran ancho de banda, caché y núcleos de procesamiento.

¿Cuál es la motivación para usar MIG?.

Sin MIG, diferentes trabajos que se ejecutan en la misma GPU, como pueden ser diferentes solicitudes de inferencia de IA, compiten por los mismos recursos. Un trabajo que consume un mayor ancho de banda de memoria priva a otros de él, lo que hace que varios trabajos no alcancen sus objetivos de latencia.

Con MIG, los trabajos se ejecutan simultáneamente en diferentes instancias, cada una con recursos dedicados para el uso de cómputo, memoria y ancho de banda de memoria, lo que da como resultado un rendimiento predecible con QoS y máxima utilización de la GPU.

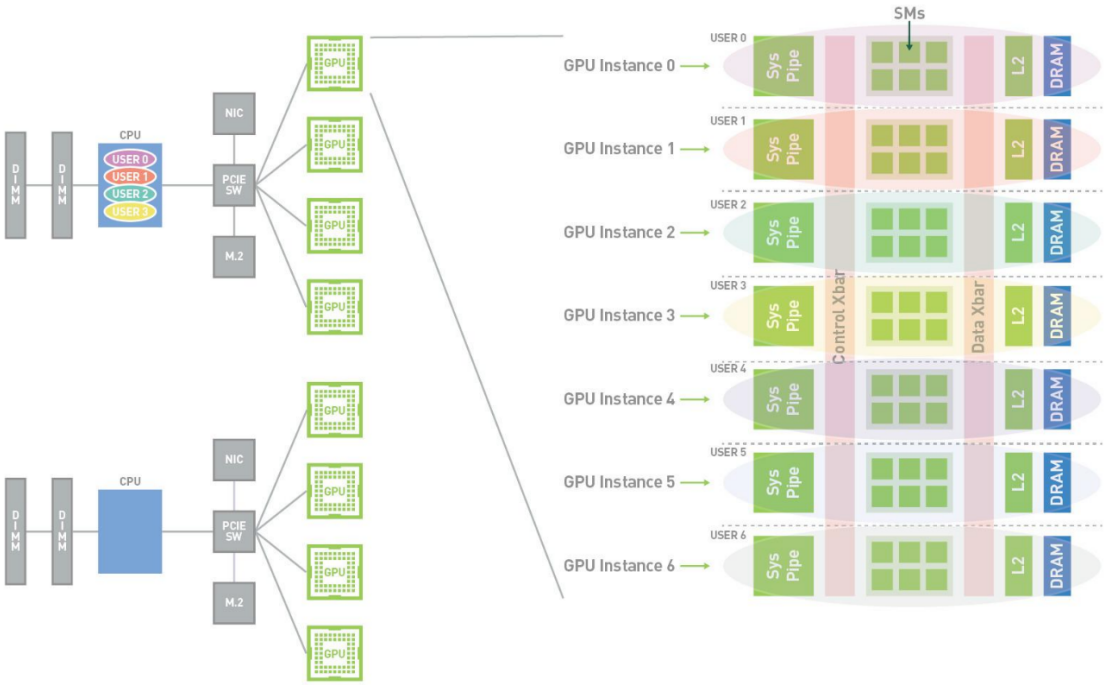
En resumen:

- Las GPU NVIDIA A100 son actualmente las GPU de la más potentes que se puede comprar.
- Su memoria va desde los 40 GB hasta los 80 GB por tarjeta.
- **No muchas aplicaciones pueden aprovechar toda la potencia de estas tarjetas.**
- Lamentablemente, **una vez que SLURM asigna una GPU, ningún otro trabajo puede utilizar la GPU**.

Los trabajos son exclusivos de GPU

Aunque existen otros enfoques, como NVIDIA MPS, pero desafortunadamente SLURM solo puede usar una GPU con MPS por nodo).

- Básicamente nos permite particionar las GPU para que más de un trabajo pueda hacer uso de una GPU.



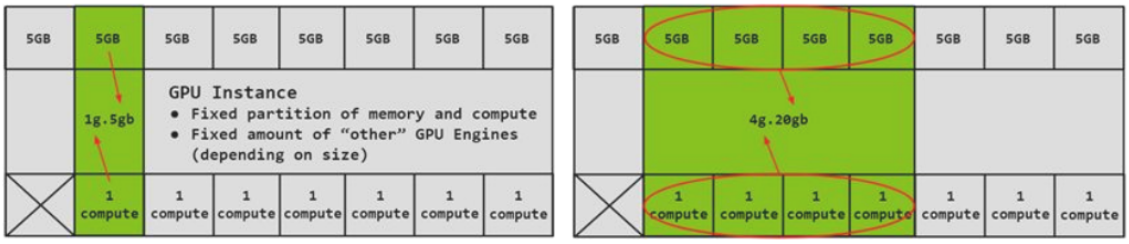
Partición GPU para un A100 de 40 GB.

Una GPU se puede dividir en particiones de instancias MIG de diferentes tamaños. Por ejemplo, en un NVIDIA A100 de 40 GB, un administrador podría crear dos instancias con 20 gigabytes (GB) de memoria cada una, tres instancias con 10 GB o siete instancias con 5 GB. O una combinación de estas.

No todas las particiones son posibles. Hay restricciones. Estas son las posibilidades:

Config	GPC Slice #0	GPC Slice #1	GPC Slice #2	GPC Slice #3	GPC Slice #4	GPC Slice #5	GPC Slice #6
1	7						
2	4				2		1
3	4				1	1	1
4	3			3			
5	3			2		1	
6	3			1	1	1	
7	2		2		3		
8	2		1	1	3		
9	1	1	2		3		
10	1	1	1	1	3		
11	2		2		2		1
12	2		1	1	2		1
13	1	1	2		2		1
14	2		1	1	1	1	1
15	1	1	2		1	1	1
16	1	1	1	1	2		1
17	1	1	1	1	1	2	
18	1	1	1	1	1	1	1

Aquí puedes ver 2 ejemplos de particionado para una GPU A100 de 40 GB.



Para obtener la lista de funciones y recursos de cada nodo y partición mig, puede usar este comando.

```
sinfo -o "%40N  %10c  %10m  %50f  %10G "
```

```
NODELIST                                 CPUS       MEMORY     AVAIL_FEATURES
GRES
node18109-1                              64         257214     ilk,gpu,a100                  gpu:a100:8
node2204-[3-4]                           20         31906      ivy                           (null)
node17109-1,node17110-1,node18110-1,node 64         257214     ilk,viz,t4
gpu:t4:1
node0303-2,node0304-[1-4],node1301-[1-4] 16         30000+     sandy
(null)
node17101-1,node17103-1,node17104-1,node 64         257214
ilk,gpu,a100                  gpu:a100:4(S:0-1)
node17102-1                              64         257214     ilk,gpu,a100,3g.20gb,2g.10gb,1g.5gb
gpu:3g.20gb:1(S:0),gpu:2g.10gb
```

- para un nodo específico con particiones mig:

```
scontrol show node node17102-1
```

```
NodeName=node17102-1 Arch=x86_64 CoresPerSocket=32
   CPUAlloc=8 CPUEfctv=64 CPUTot=64 CPULoad=0.00
   AvailableFeatures=ilk,gpu,a100,3g.20gb,2g.10gb,1g.5gb
   ActiveFeatures=ilk,gpu,a100,3g.20gb,2g.10gb,1g.5gb
   Gres=gpu:3g.20gb:1(S:0),gpu:2g.10gb:1(S:0),gpu:a100:3(S:0-1),gpu:1g.5gb:2(S:0)
   NodeAddr=node17102-1 NodeHostName=node17102-1 Version=22.05.8
   OS=Linux 4.18.0-372.9.1.el8.x86_64 #1 SMP Tue May 10 14:48:47 UTC 2022
   RealMemory=257214 AllocMem=0 FreeMem=142408 Sockets=2 Boards=1
   State=MIXED ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=main,batch,express,long
   BootTime=2023-07-24T08:38:22 SlurmdStartTime=2023-07-24T09:20:41
   LastBusyTime=2023-08-03T09:07:43
   CfgTRES=cpu=64,mem=257214M,billing=64,gres/gpu=7,gres/gpu:1g.5gb=2,gres/gpu:
2g.10gb=1,gres/gpu:3g.20gb=1,gres/gpu:a100=3
   AllocTRES=cpu=8,gres/gpu=2,gres/gpu:a100=2
   CapWatts=n/a
   CurrentWatts=0 AveWatts=0
   ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
```

- Un nodo sin partiones mig:

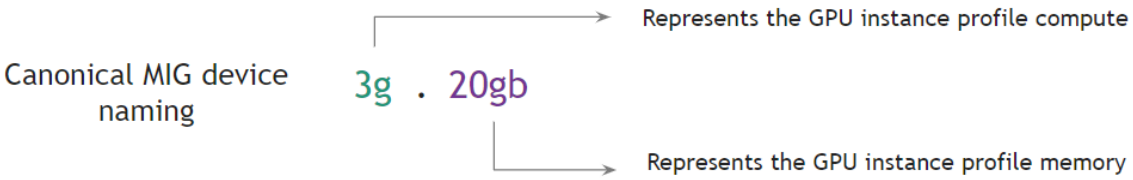
```
scontrol show node node17101-1
```

```
NodeName=node17101-1 Arch=x86_64 CoresPerSocket=32
   CPUAlloc=8 CPUEfctv=64 CPUTot=64 CPULoad=0.00
   AvailableFeatures=ilk,gpu,a100
   ActiveFeatures=ilk,gpu,a100
   Gres=gpu:a100:4(S:0-1)
   NodeAddr=node17101-1 NodeHostName=node17101-1 Version=22.05.8
   OS=Linux 4.18.0-372.9.1.el8.x86_64 #1 SMP Tue May 10 14:48:47 UTC 2022
   RealMemory=257214 AllocMem=0 FreeMem=119559 Sockets=2 Boards=1
   State=MIXED ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=main,batch,express,long
   BootTime=2023-07-17T13:29:12 SlurmdStartTime=2023-07-21T12:42:08
   LastBusyTime=2023-08-03T11:06:24
   CfgTRES=cpu=64,mem=257214M,billing=64,gres/gpu=4,gres/gpu:a100=4
   AllocTRES=cpu=8,gres/gpu=4,gres/gpu:a100=4
   CapWatts=n/a
   CurrentWatts=0 AveWatts=0
   ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
```

MIG Device Names

De manera predeterminada, un dispositivo MIG consta de una sola *“GPU Instance”* y una sola *“Compute Instance”*. La siguiente tabla destaca una convención de nomenclatura para hacer referencia a un dispositivo MIG por el número de segmentos de cómputo de su instancia de GPU y su memoria total en GB (en lugar de solo el número de segmentos de memoria).

Cuando se crea un único CI (que consume toda la capacidad de cómputo de la GI), el tamaño del CI está implícito en el nombre del dispositivo.(Editado) Recuperar original



La siguiente descripción muestra los nombres de perfil en A100-SXM4-40GB. Estos son el nombre del dispositivo cuando se usa un sólo CI.

Memory	**20gb**	**10gb**	**5gb**
GPU Instance	3g	2g	1g
Compute Instance	3c	2c	1c
MIG Device	3g.20gb	2g.10gb	1g.5gb
	GPCGPCGPC	GPCGPC	GPC

En la siguiente sección puede ver cómo se solicita un nodo de GPU para ejecutar con slurm. También recomendamos leer la sección solicitud de recursos de GPU y cómputo