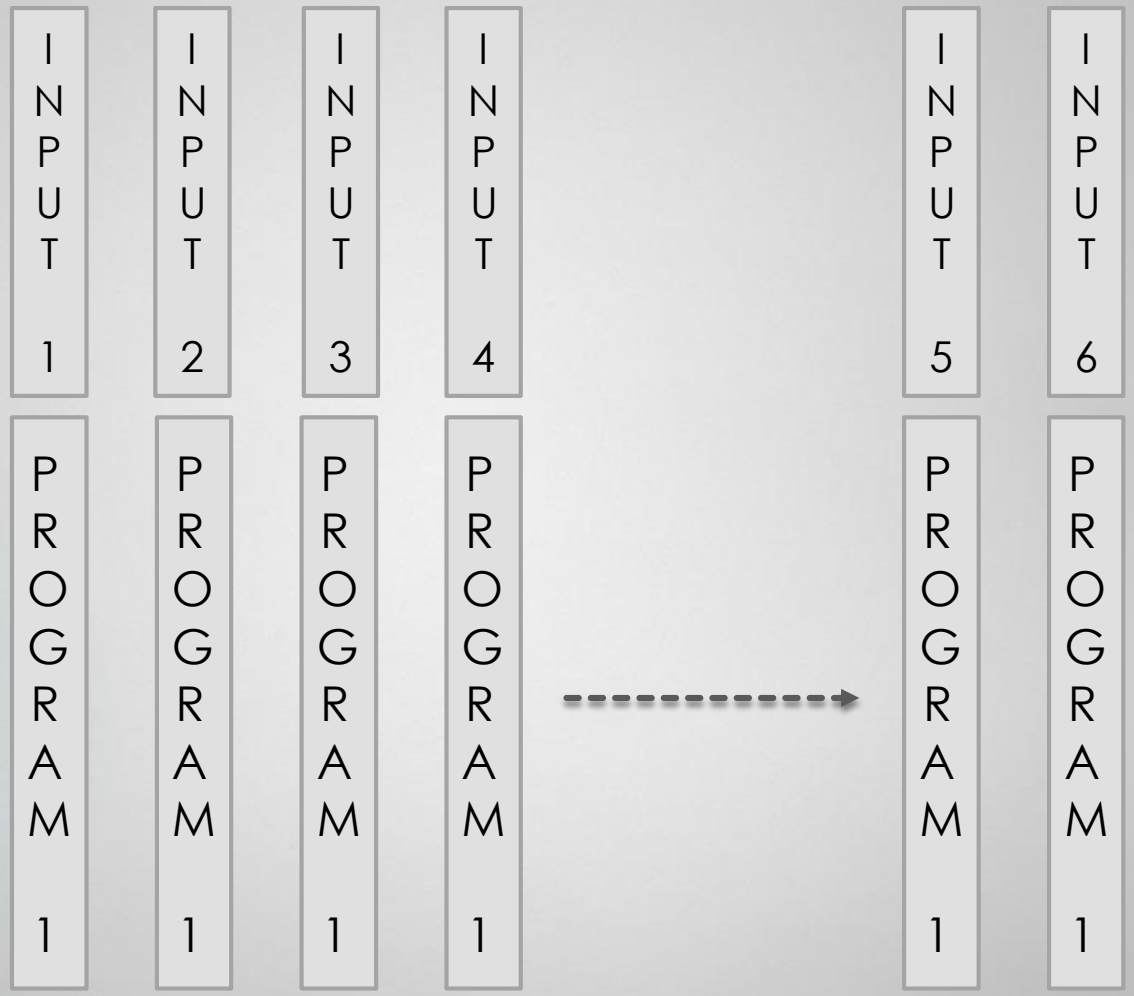
Une nouvelle approche pour la gestion de la mémoire avec CUDA

Une nouvelle approche pour la gestion de la mémoire avec CUDA

- Concepts de base
- Gérer la mémoire efficacement avec CUDA
- Qu'est ce que UVA et UVM ?
- Utiliser UVM dans une application

Architecture d’une carte graphique

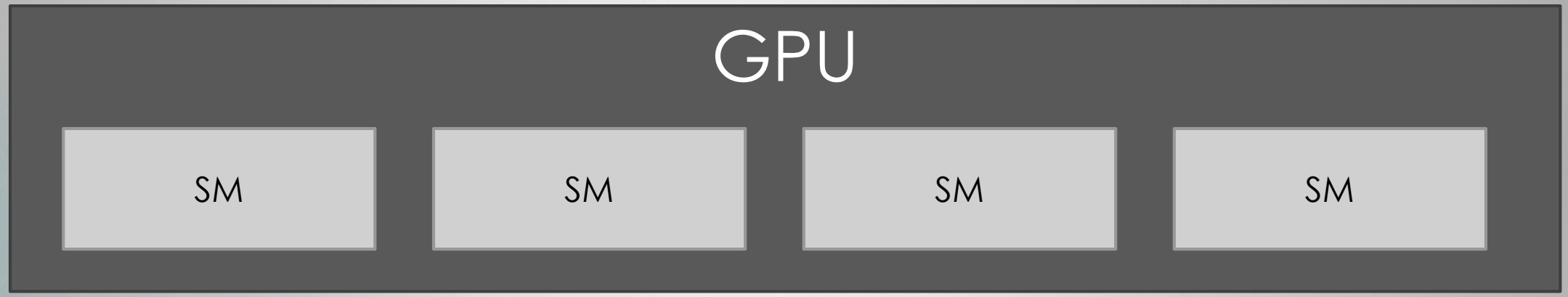
Architecture d'une carte graphique



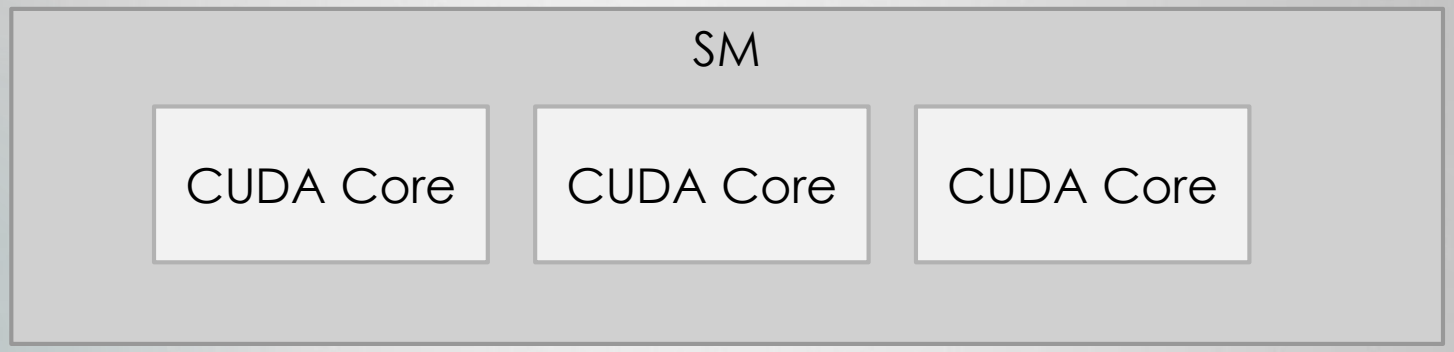
Architecture d'une carte graphique

```
MyCudaFunction<<<Block, Thread>>>(…);
```

Architecture d'une carte graphique

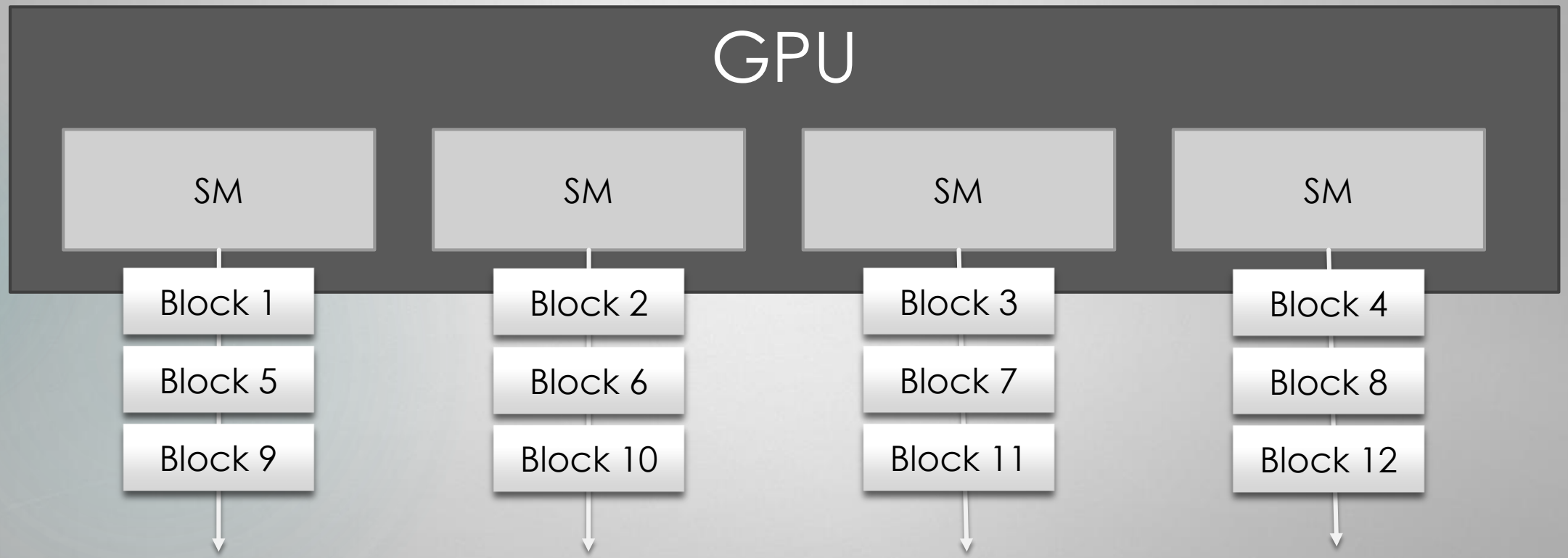


Architecture d'une carte graphique

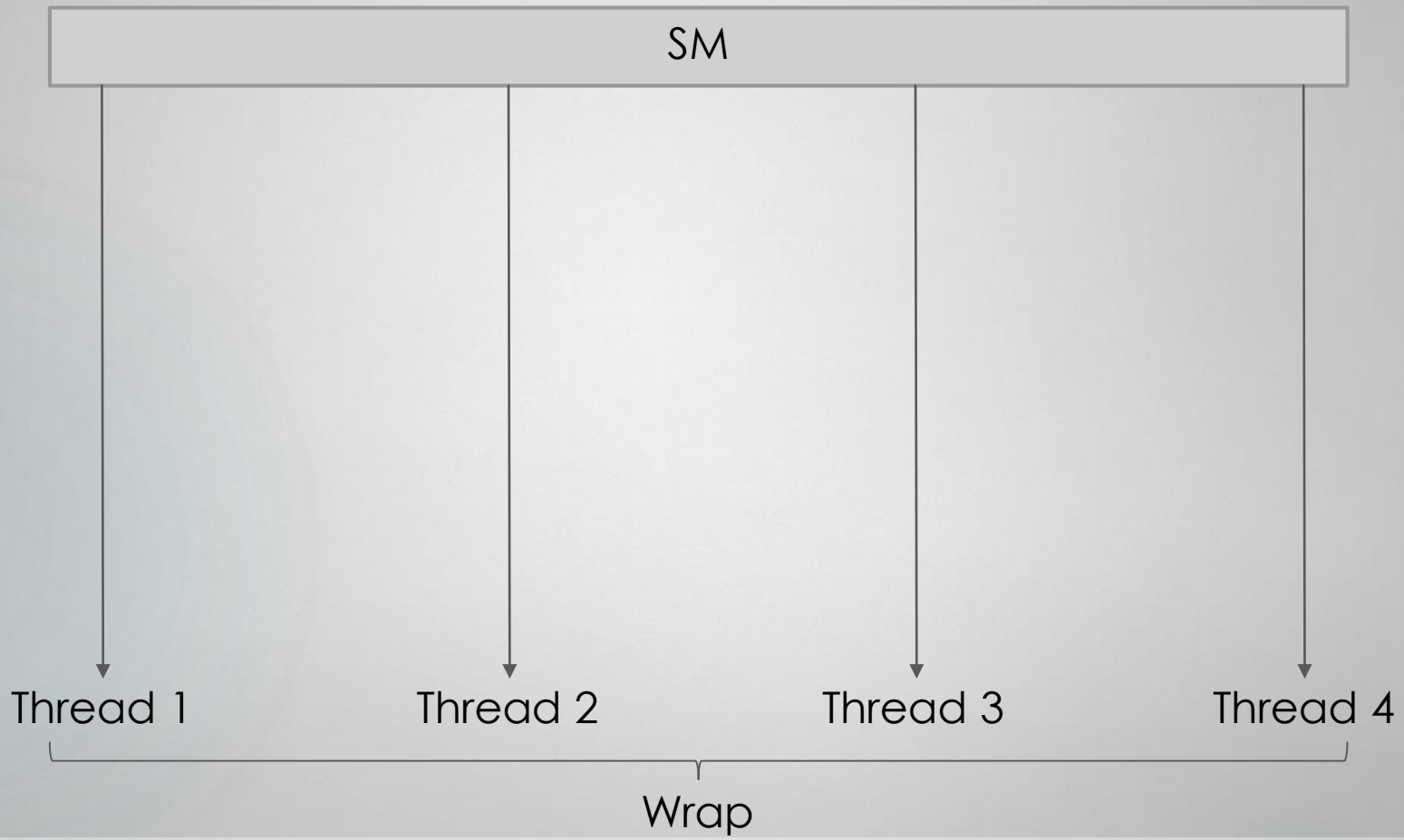


Architecture d'une carte graphique

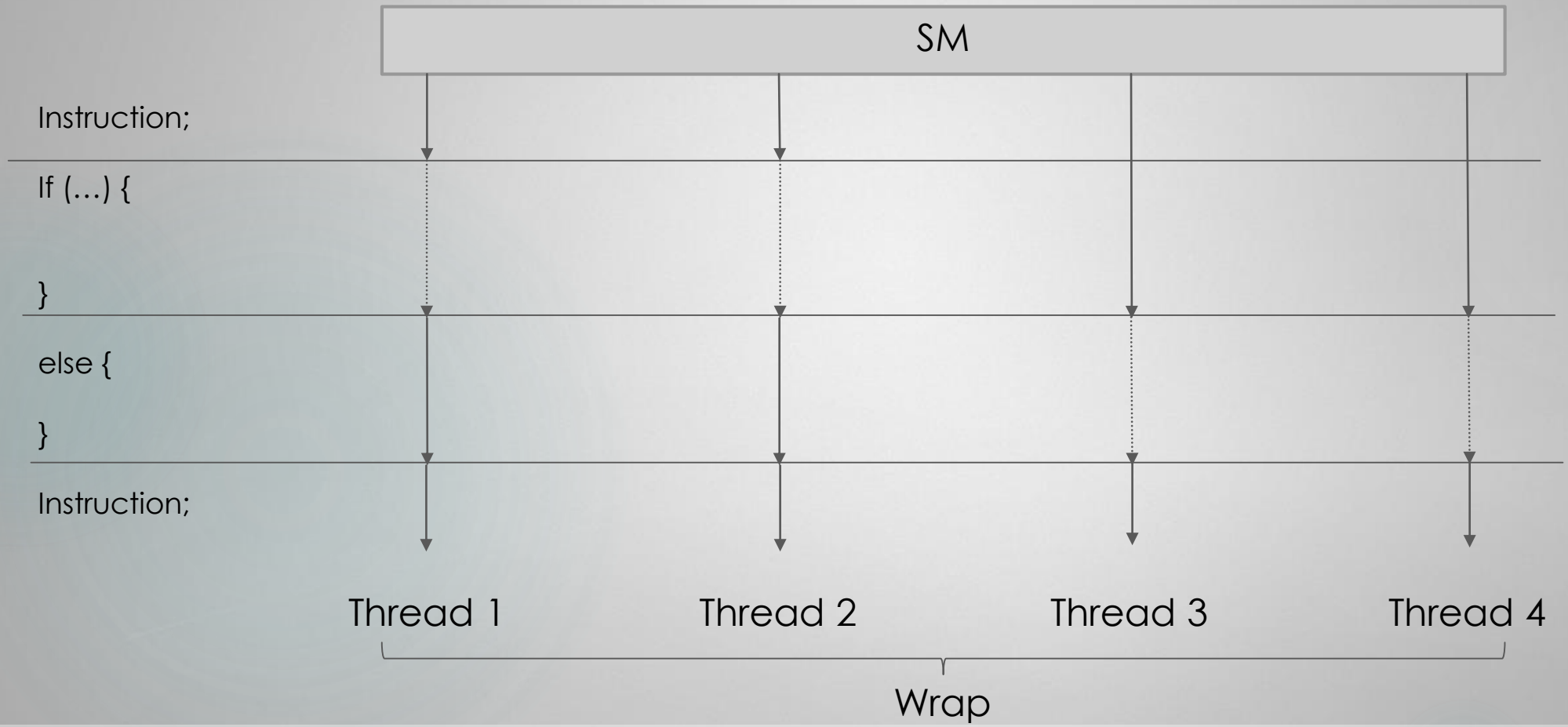
MyCudaFunction<<<12, …>>>(…);



Architecture d'une carte graphique



Architecture d'une carte graphique



PTX: Parallel Thread Execution

```
setp.lt.s32 p|q, a, b
@p bra Else
instructionA
bra End
Else:
instructionB
End:
instructionC
```

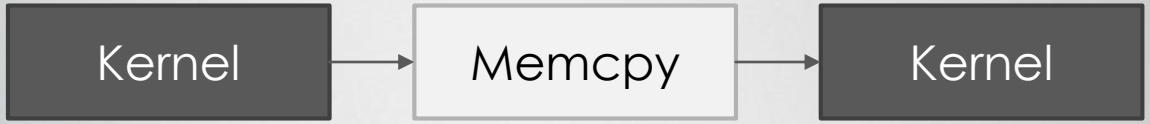
```
setp.lt.s32 p|q, a, b
@p InstructionA
@q InstructionB
InstructionC
```

CUDA Stream

```
MyCudaFunction<<<Block, Thread, Stream>>>(…);
```

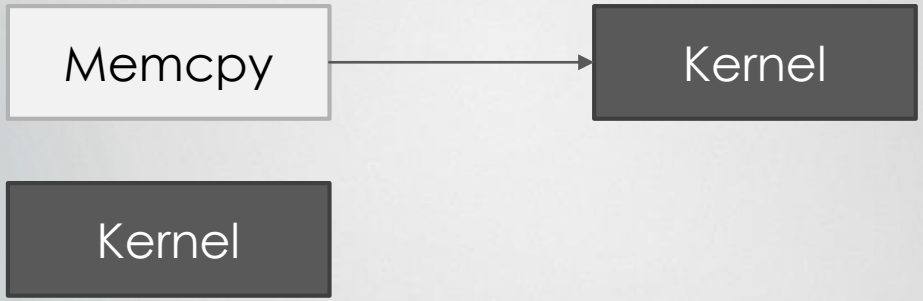
CUDA Stream

```
MyCudaFunction<<<Block, Thread>>>(…);
Cumemcpy(…)
MyCudaFunction<<<Block, Thread>>>(…);
```



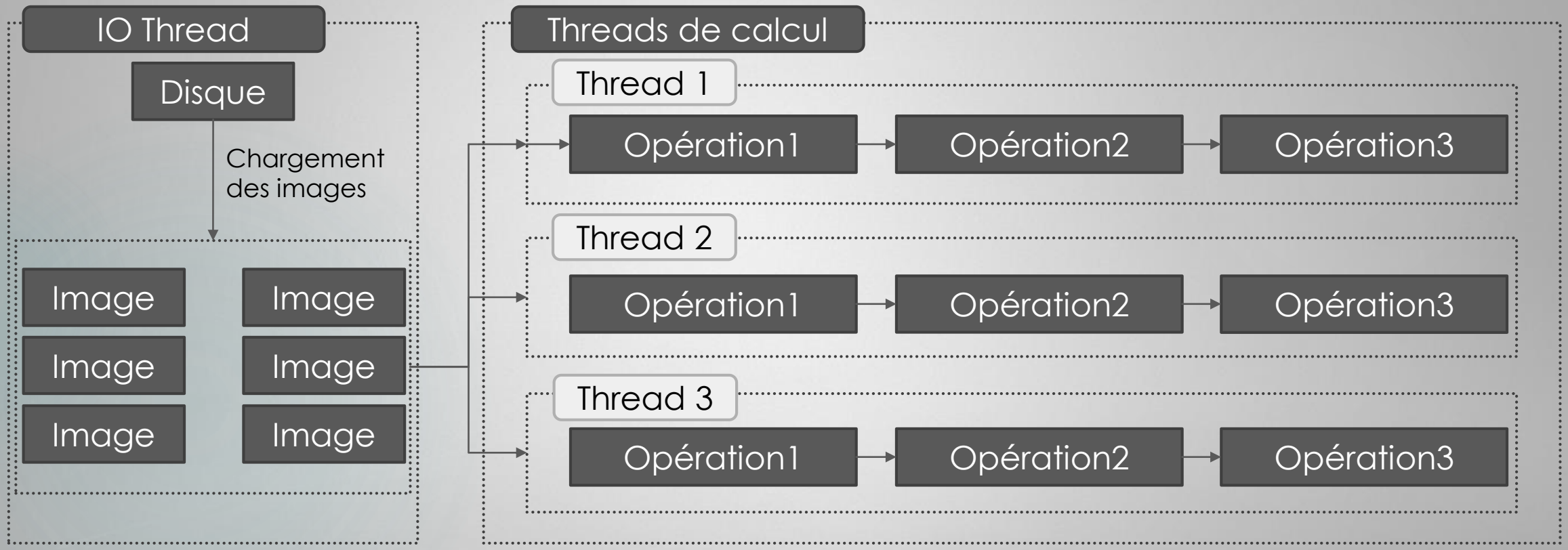
CUDA Stream

```
MyCudaFunction<<<Block, Thread, Stream1>>>(…);
Cumemcpy(…, Stream2)
MyCudaFunction<<<Block, Thread, Stream2>>>(…);
```

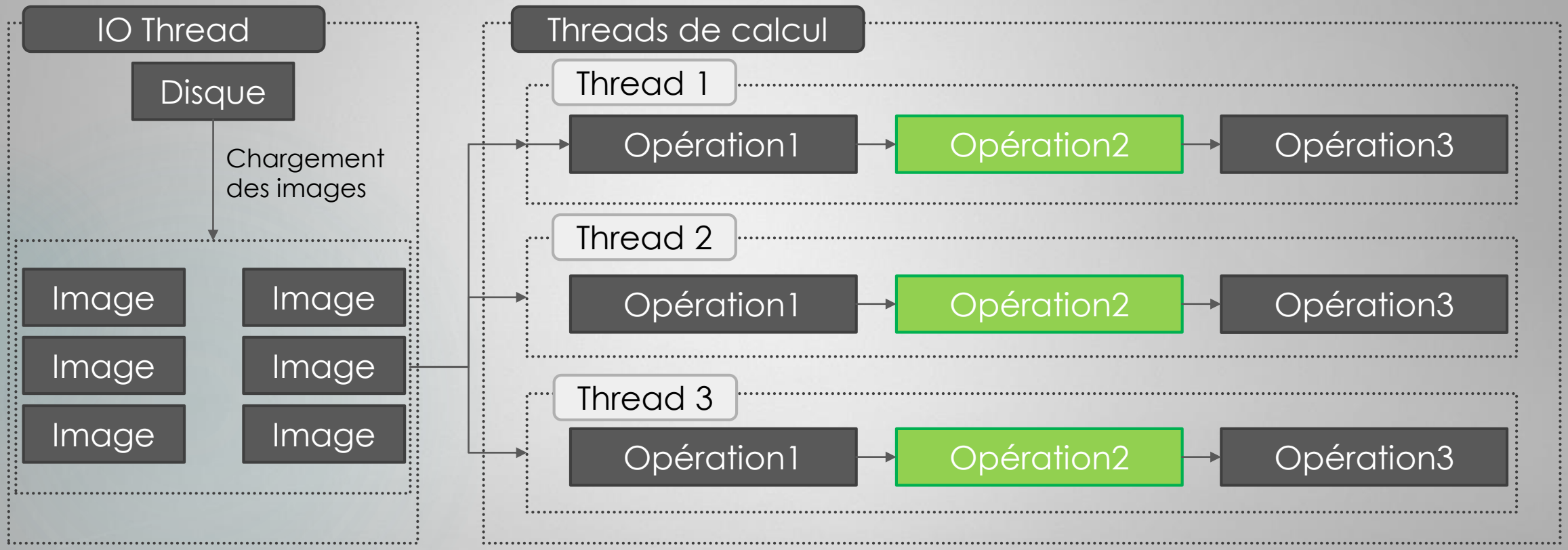


Gérer la mémoire efficacement avec CUDA

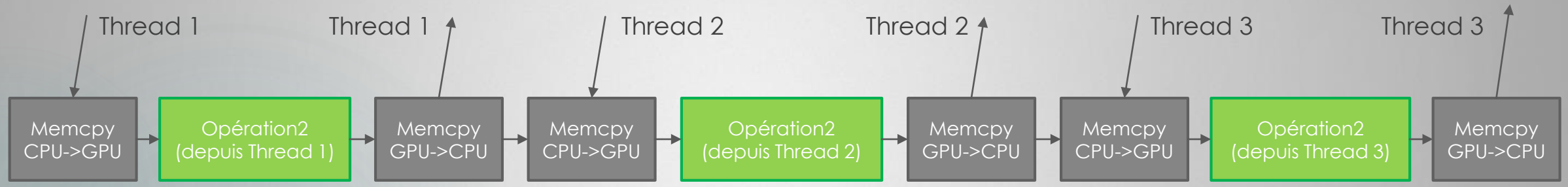
Gérer la mémoire efficacement avec CUDA



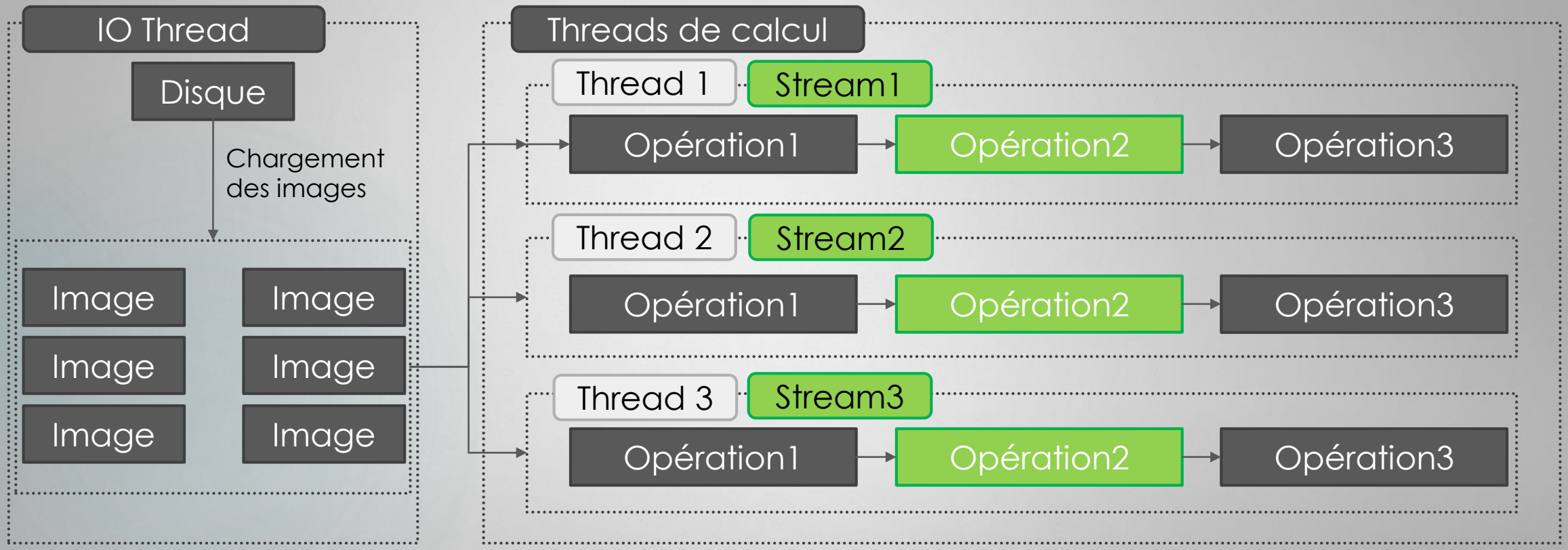
Gérer la mémoire efficacement avec CUDA



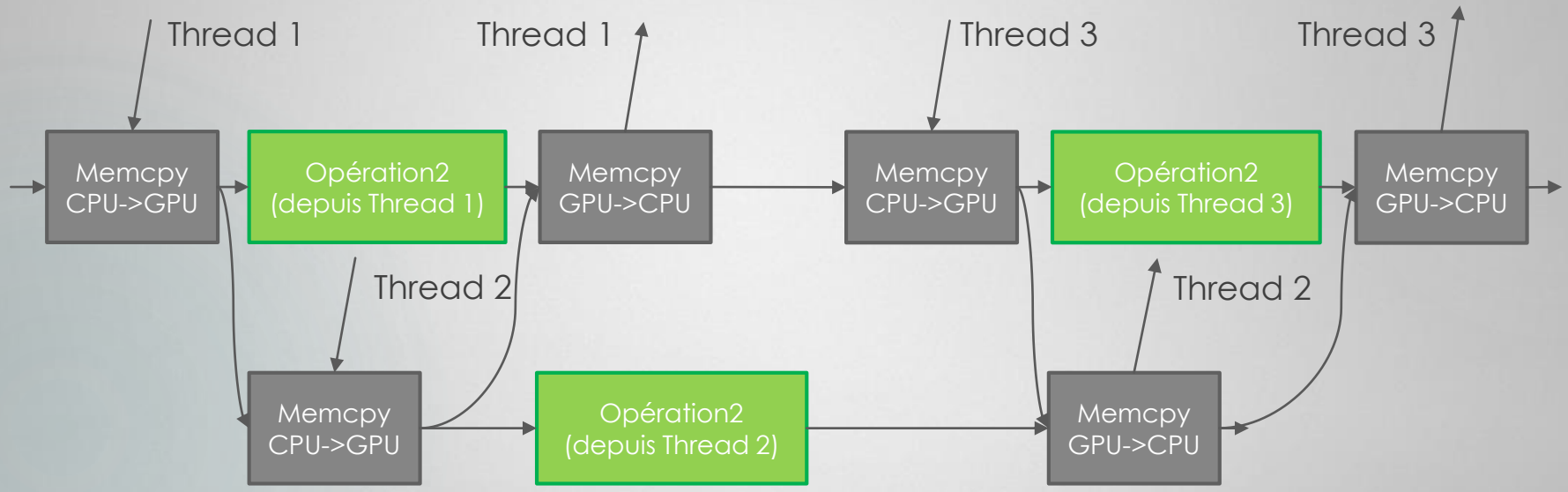
Gérer la mémoire efficacement avec CUDA



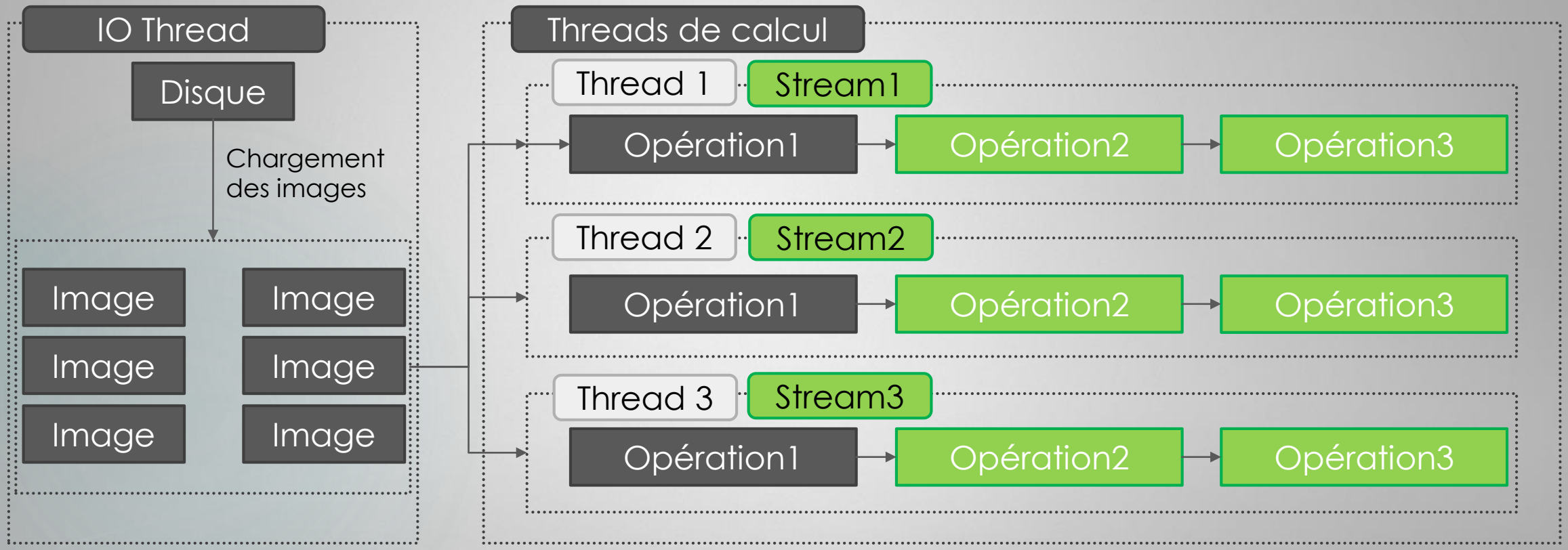
Gérer la mémoire efficacement avec CUDA



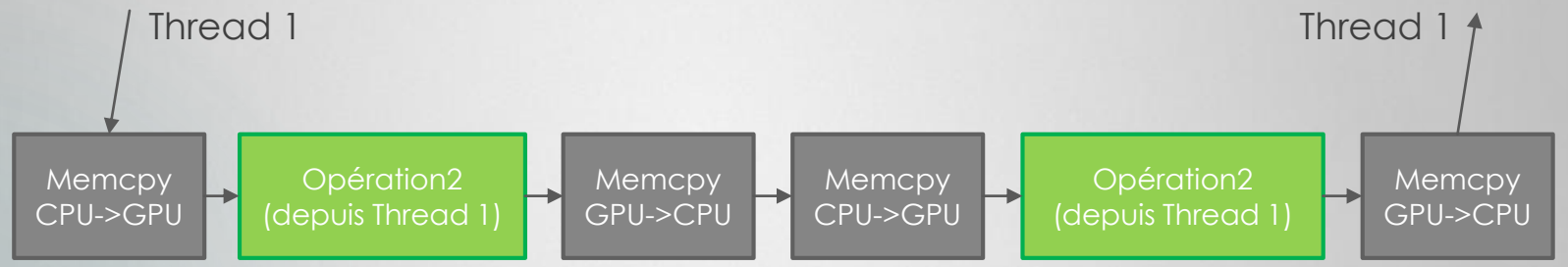
Gérer la mémoire efficacement avec CUDA



Gérer la mémoire efficacement avec CUDA



Gérer la mémoire efficacement avec CUDA

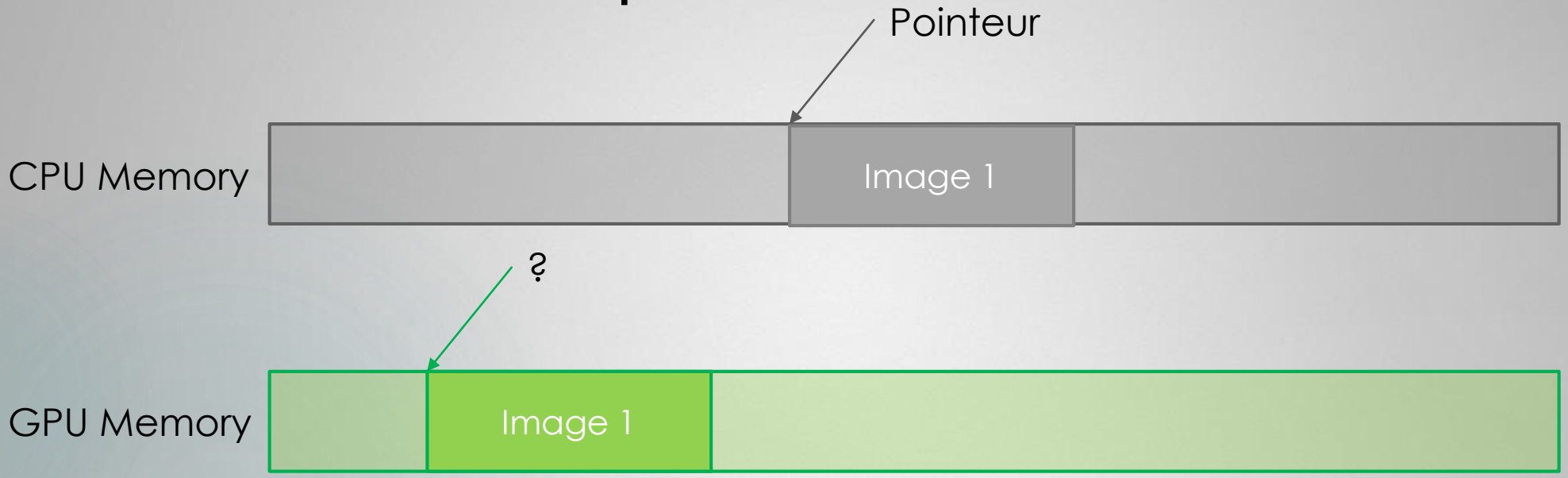


Gérer la mémoire efficacement avec CUDA

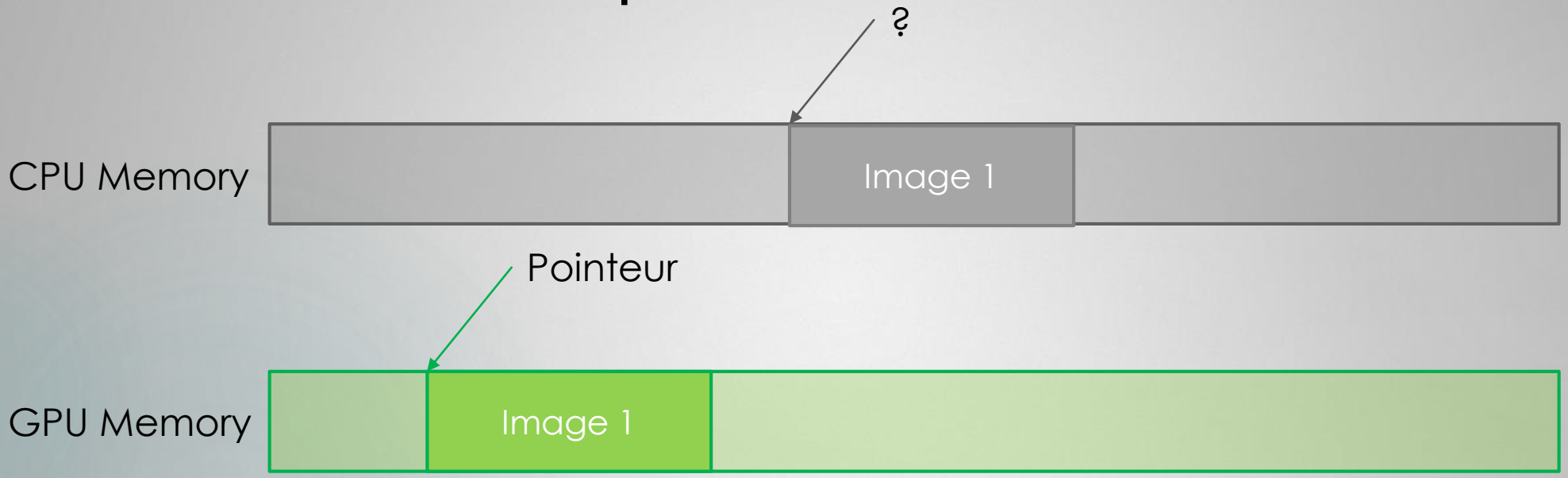
Que faire si les opérations sont dans des bibliothèques ?

Que faire si les opérations sont dans des bibliothèques ?

Que faire si les opérations sont dans des bibliothèques ?



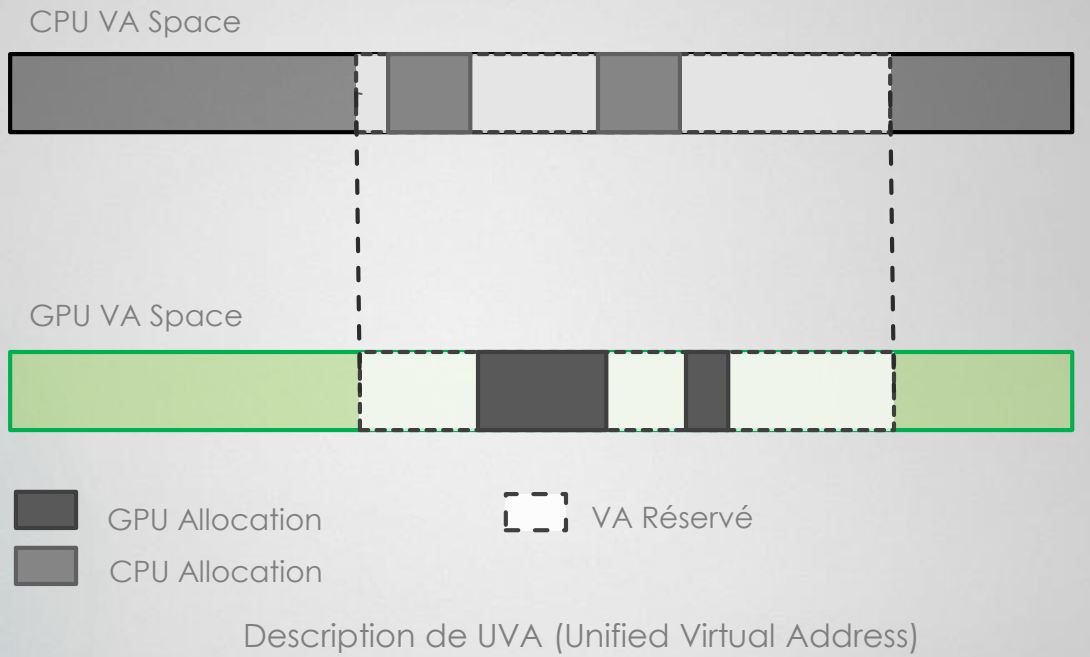
Que faire si les opérations sont dans des bibliothèques ?



Que faire si les opérations sont dans des bibliothèques ?

Qu’est ce que UVA et UVM ?

Qu'est ce que UVA?

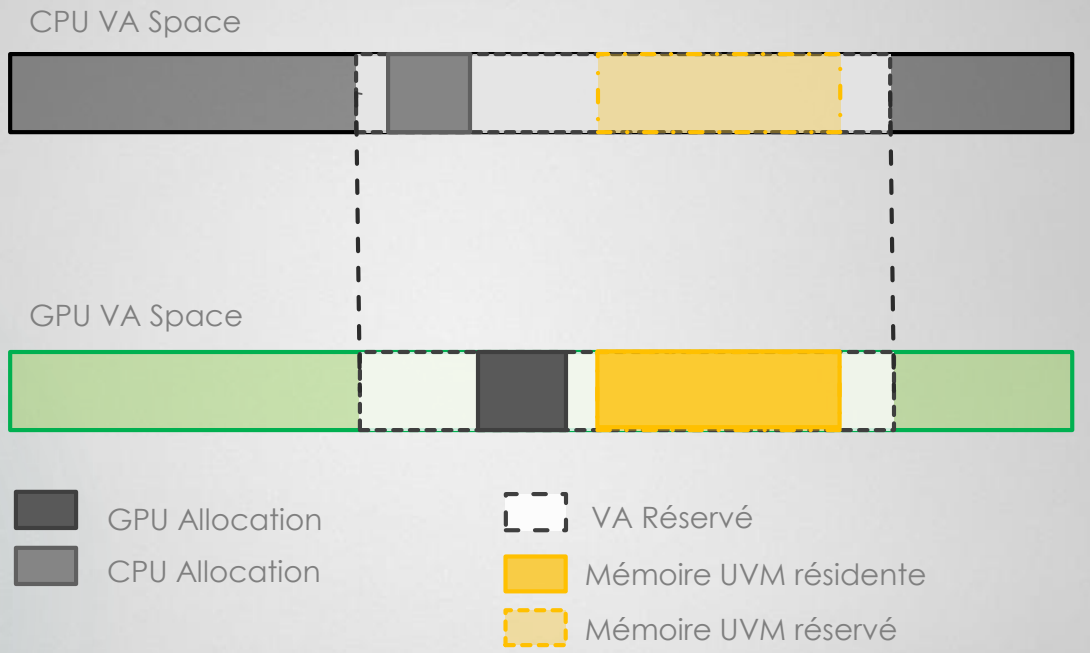


Description de UVA (Unified Virtual Address)

Qu’est ce que UVM?

Description de UVM (Unified Virtual Memory)

Qu’est ce que UVM?



Description de UVM (Unified Virtual Memory)

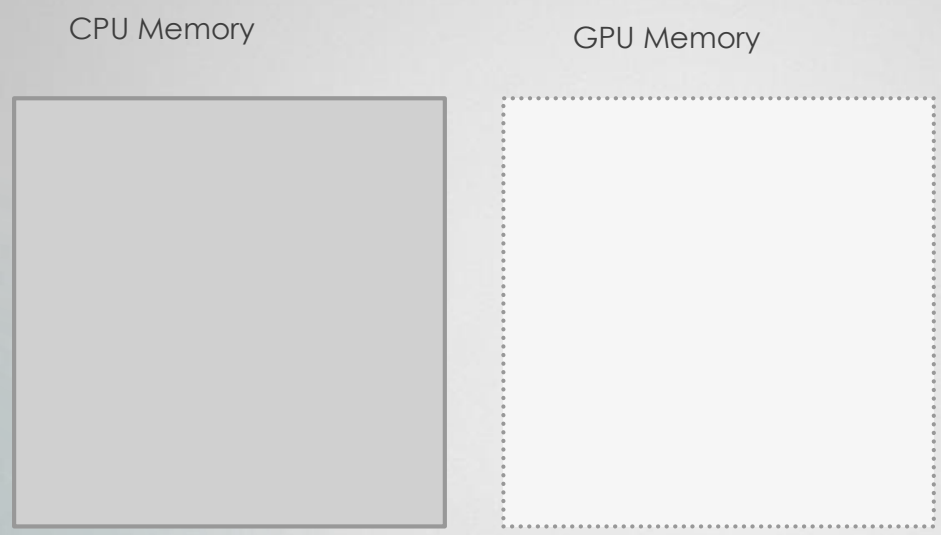
Qu’est ce que UVM?

Description de UVM (Unified Virtual Memory)

Qu’est ce que UVA et UVM ?

Utiliser UVM dans une application

Utiliser UVM dans une application



Allocation d'un bloc de mémoire avec UVM et creation d'un stream pour le contenir.

Block mémoire réservé

Block mémoire résident

Utiliser UVM dans une application

CPU Memory

GPU Memory

Chargement de l'image dans le bloc mémoire.

Block mémoire réservé

Block mémoire résident

Utiliser UVM dans une application

CPU Memory

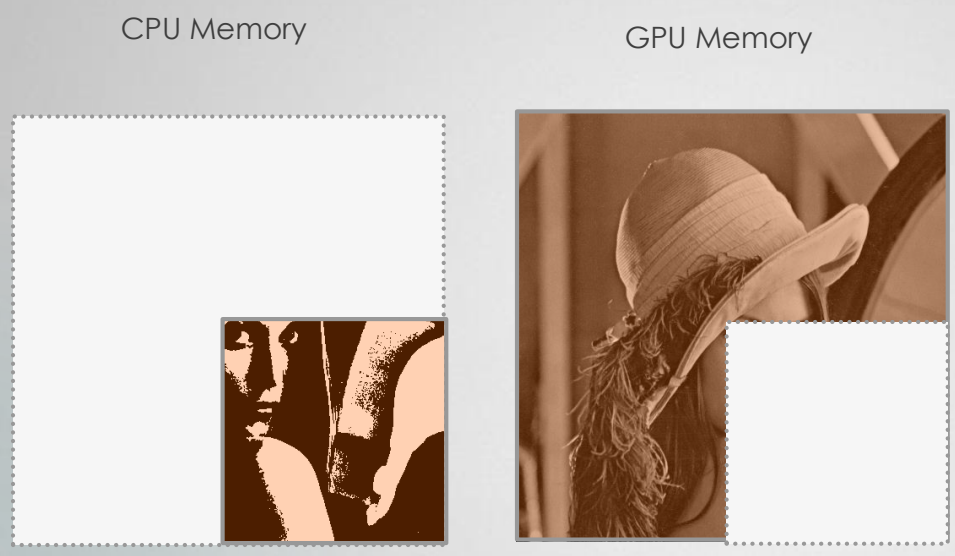
GPU Memory

Binarization d''un bloc de l'image dans UVM

Block mémoire réservé

Block mémoire résident

Utiliser UVM dans une application



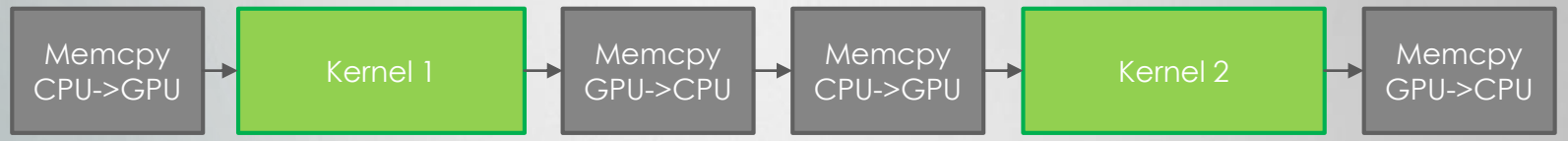
Migration GPU -> CPU lorsque l'utilisateur accède au données depuis le CPU.

Block mémoire réservé

Block mémoire résident

Avantages

Elimination des memcpy inutile automatique

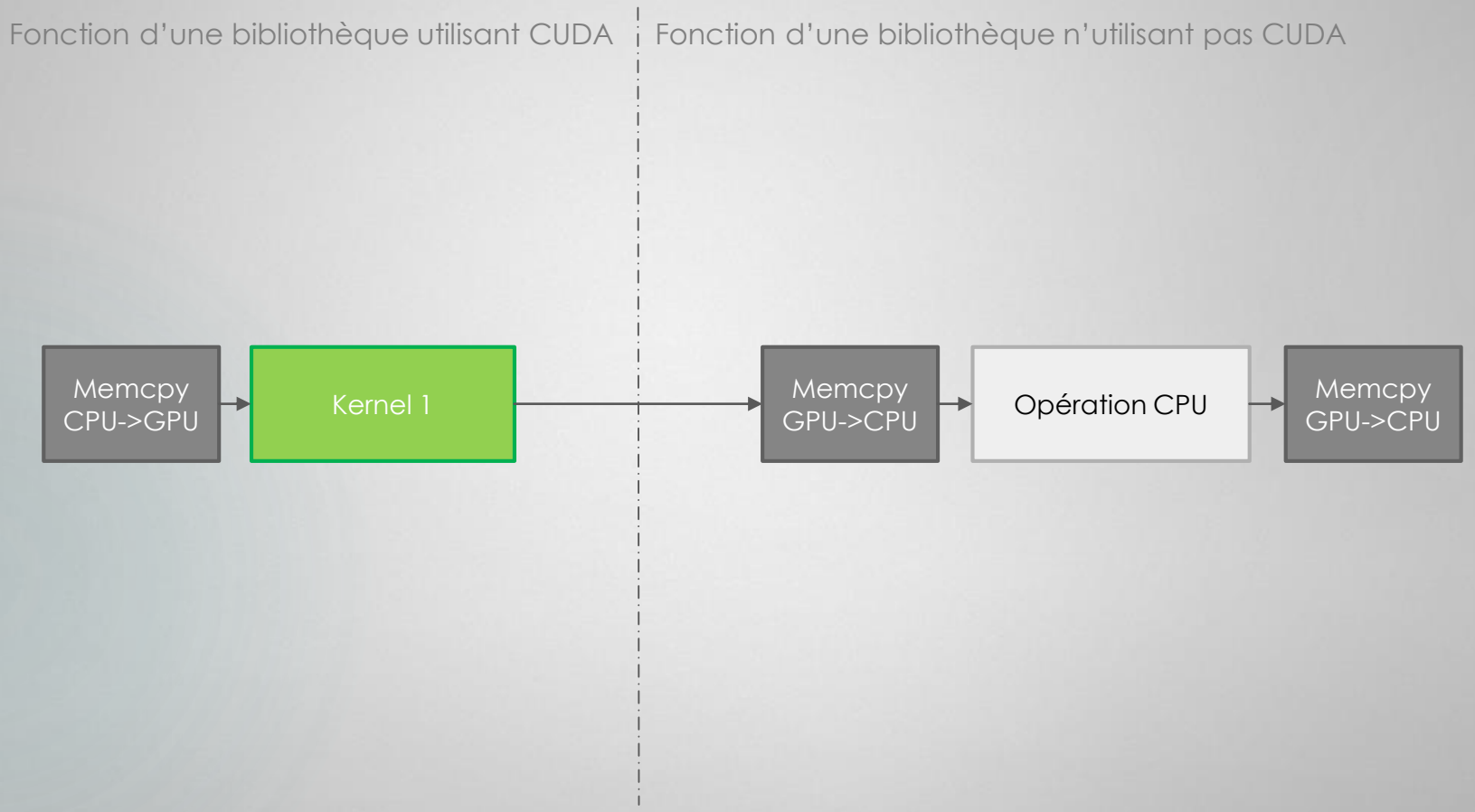


Avantages

Elimination des memcpy inutile automatique

Avantages

Cohabitation entre bibliothèques utilisant CUDA et bibliothèques conventionnelles



Utiliser UVM dans une application

Inconvénients

Plus lent qu'une optimisation manuelle des memcpy

Moins de contrôle sur les déplacement de mémoire

Utilisable sur des cartes Nvidia récentes uniquement

Conclusion

Questions ?