

# Streaming Multiprocessors

GPU

NVIDIA GPUs contain functional units called **Streaming Multiprocessors, or SMs**

# GPU

SM

NVIDIA GPUs contain functional units called **Streaming Multiprocessors, or SMs**

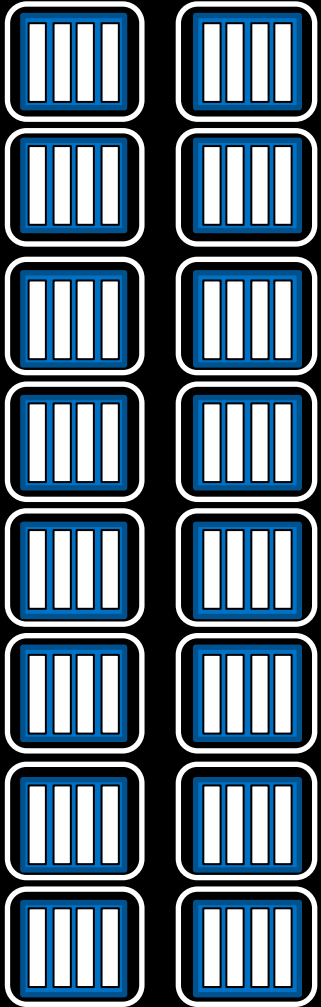
# GPU

SM

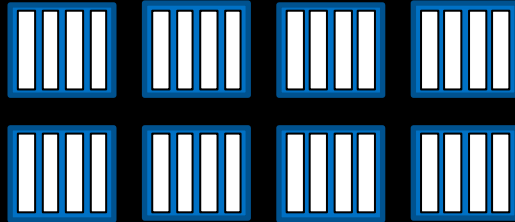
```
kernel<<<24, 4>>>()
```

Blocks of threads are scheduled to run on SMs

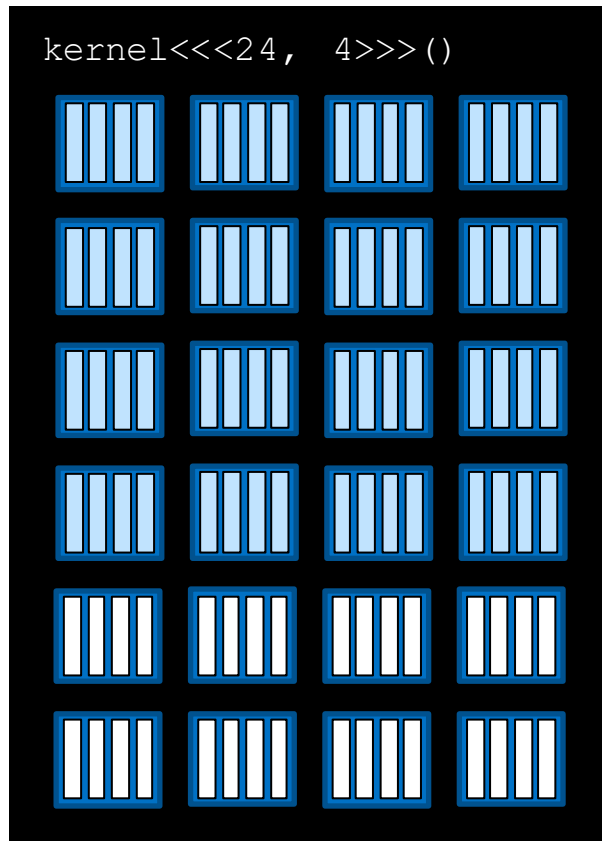
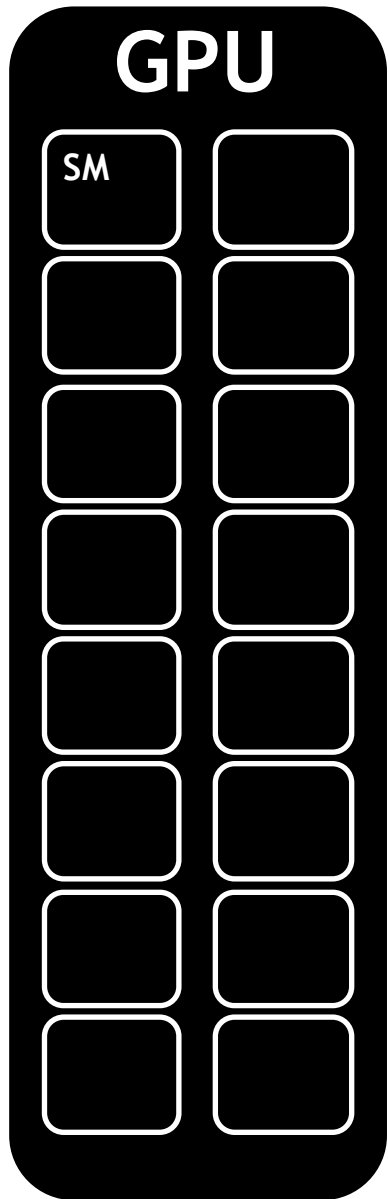
# GPU



```
kernel<<<24, 4>>>()
```

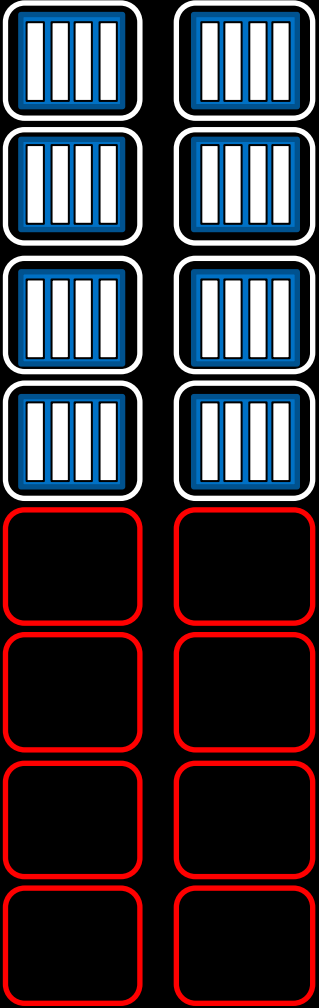


Depending on the number of SMs on a GPU, and the requirements of a block, more than one block can be scheduled on an SM

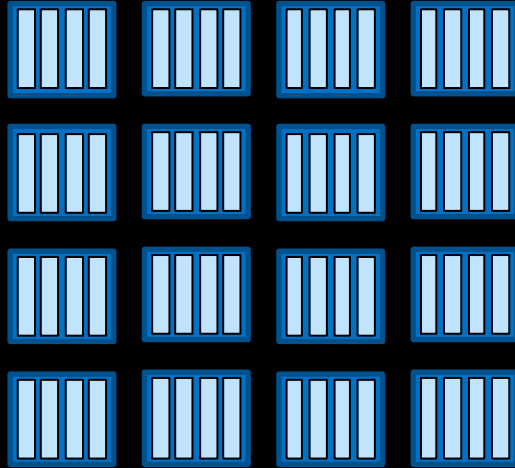


Depending on the number of SMs on a GPU, and the requirements of a block, more than one block can be scheduled on an SM

# GPU



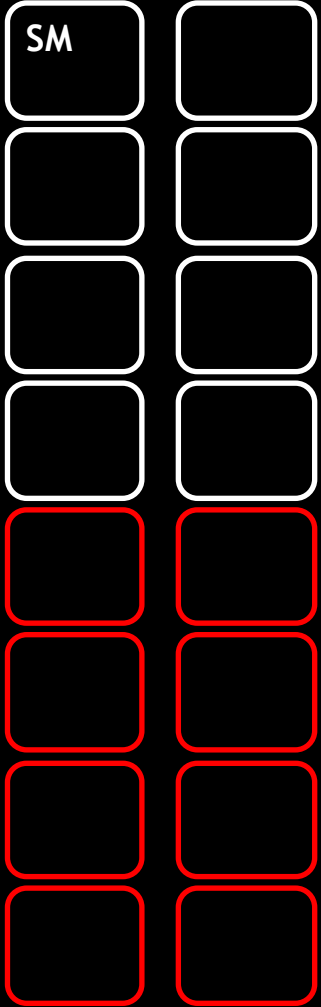
```
kernel<<<24, 4>>>()
```



Grid dimensions divisible by the number of SMs on a GPU can promote full SM utilization

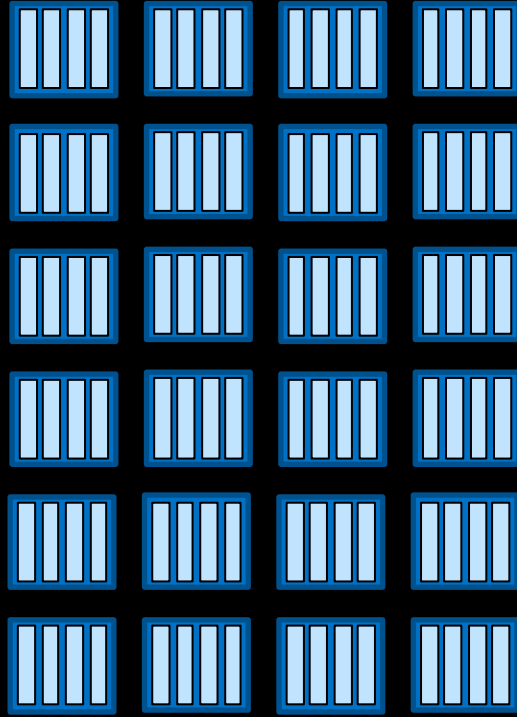
# GPU

SM



Here there are follow SMs

```
kernel<<<24, 4>>>()
```

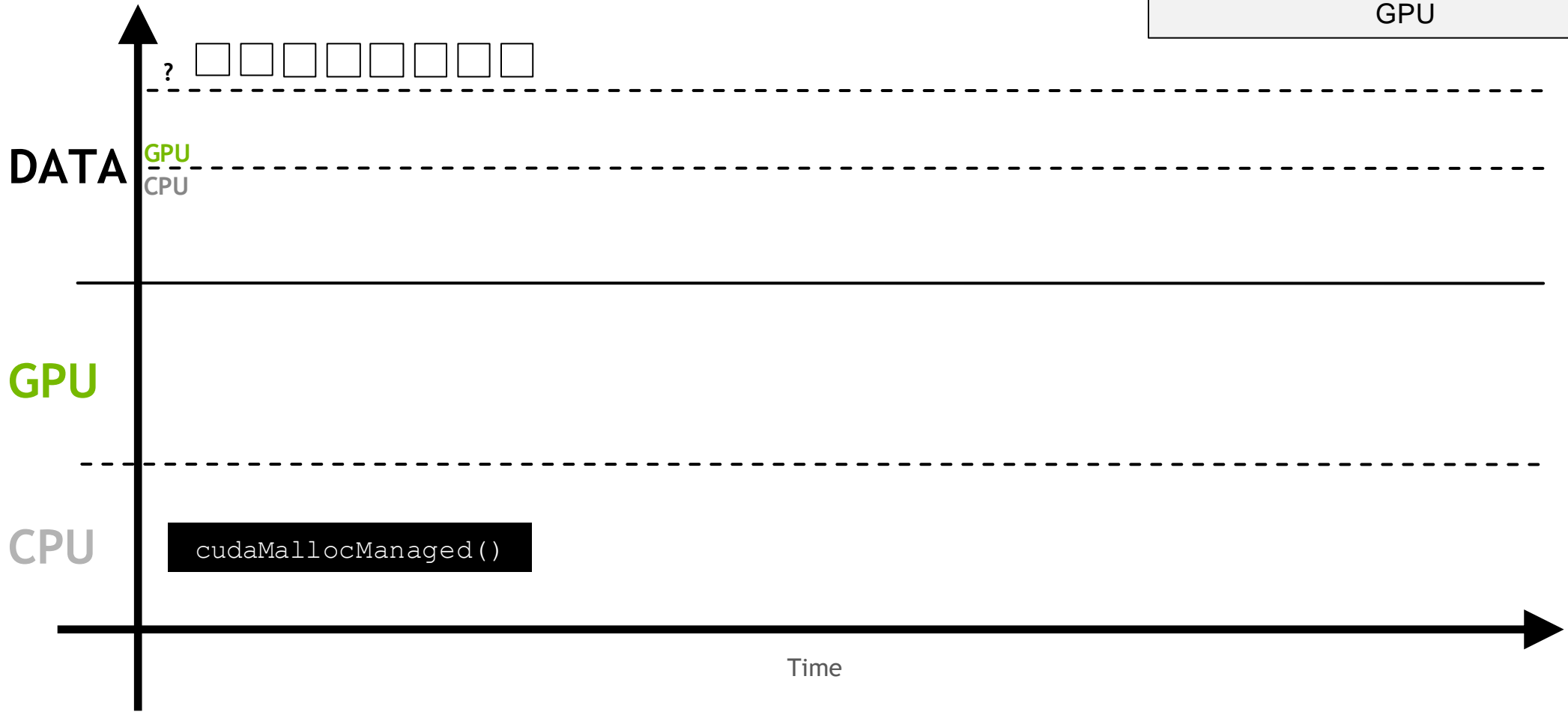




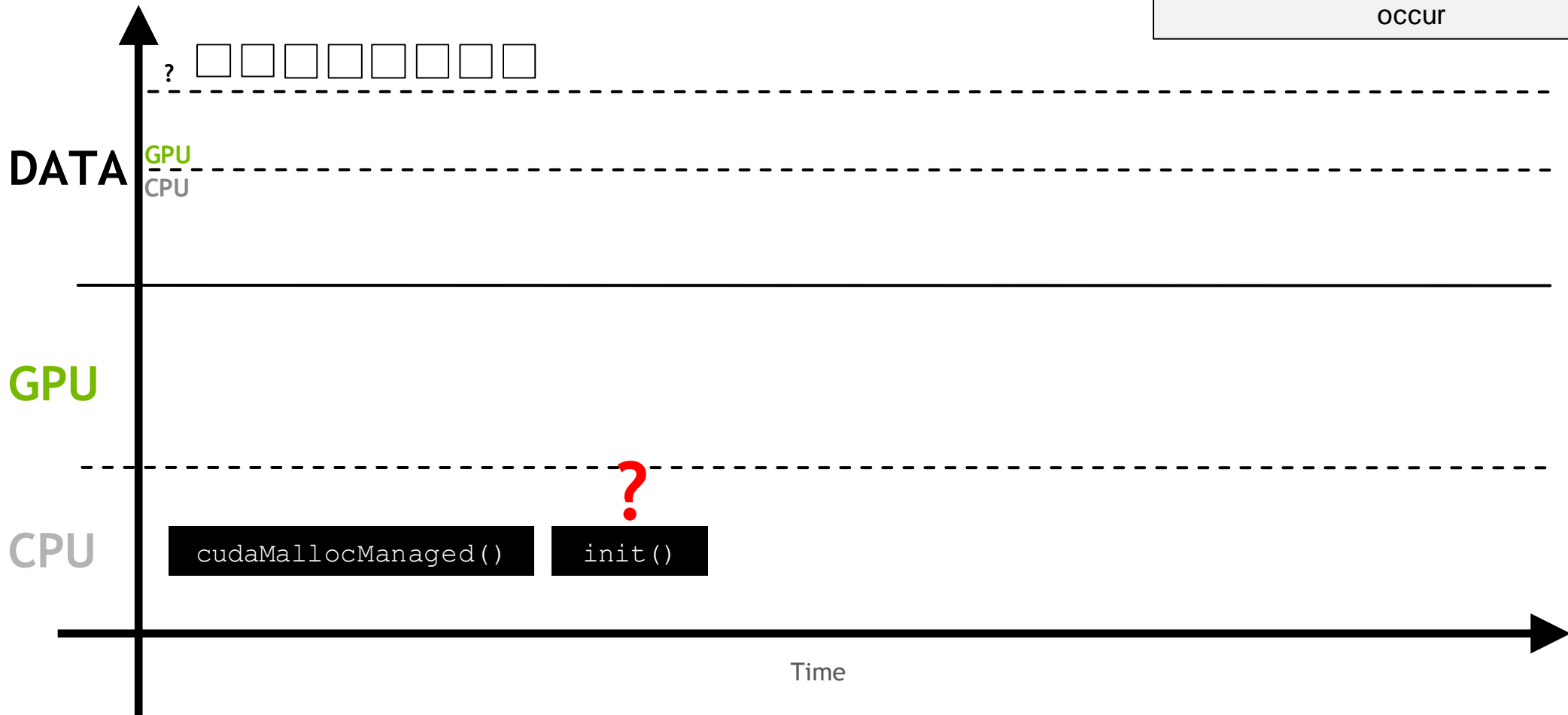
# Unified Memory Behavior



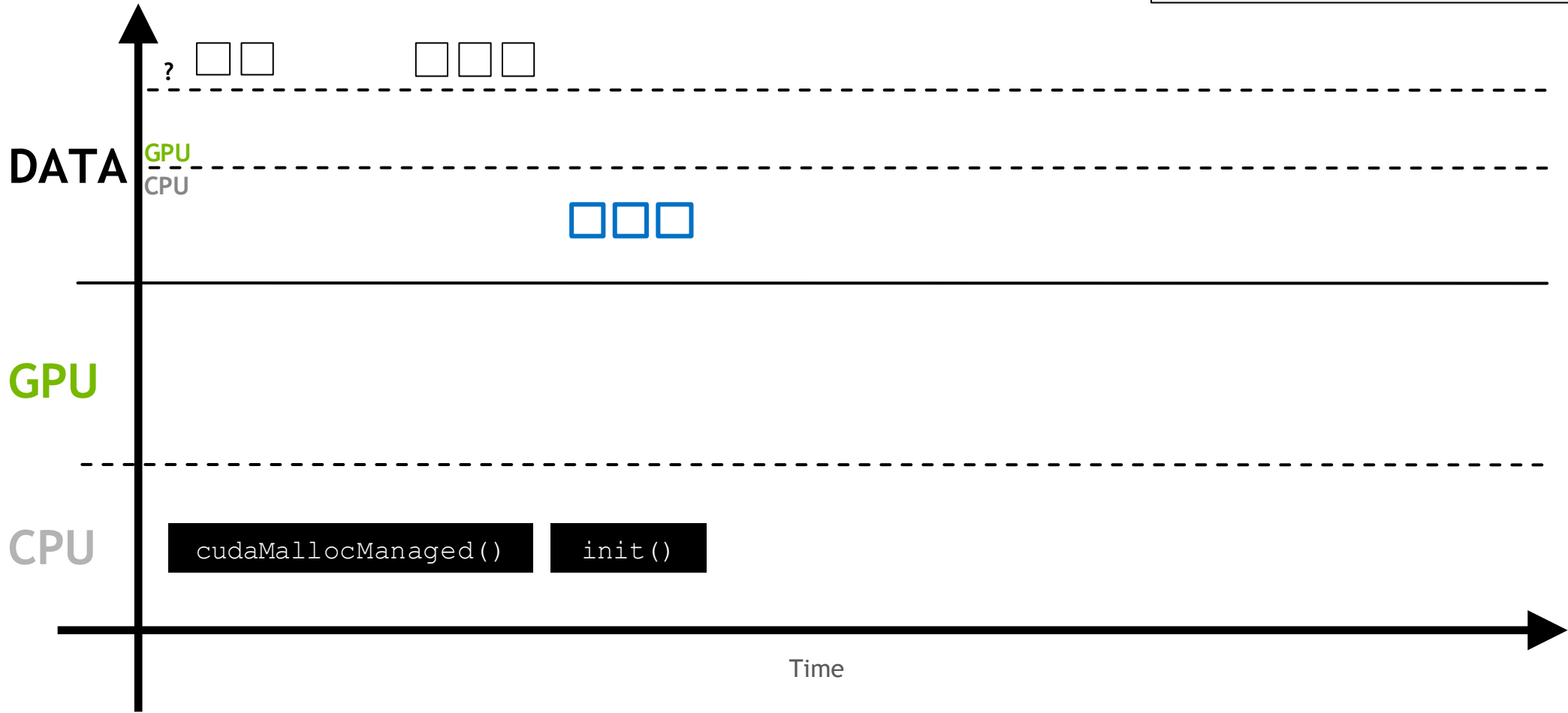
When **UM** is allocated, it may not be resident initially on the CPU or the GPU



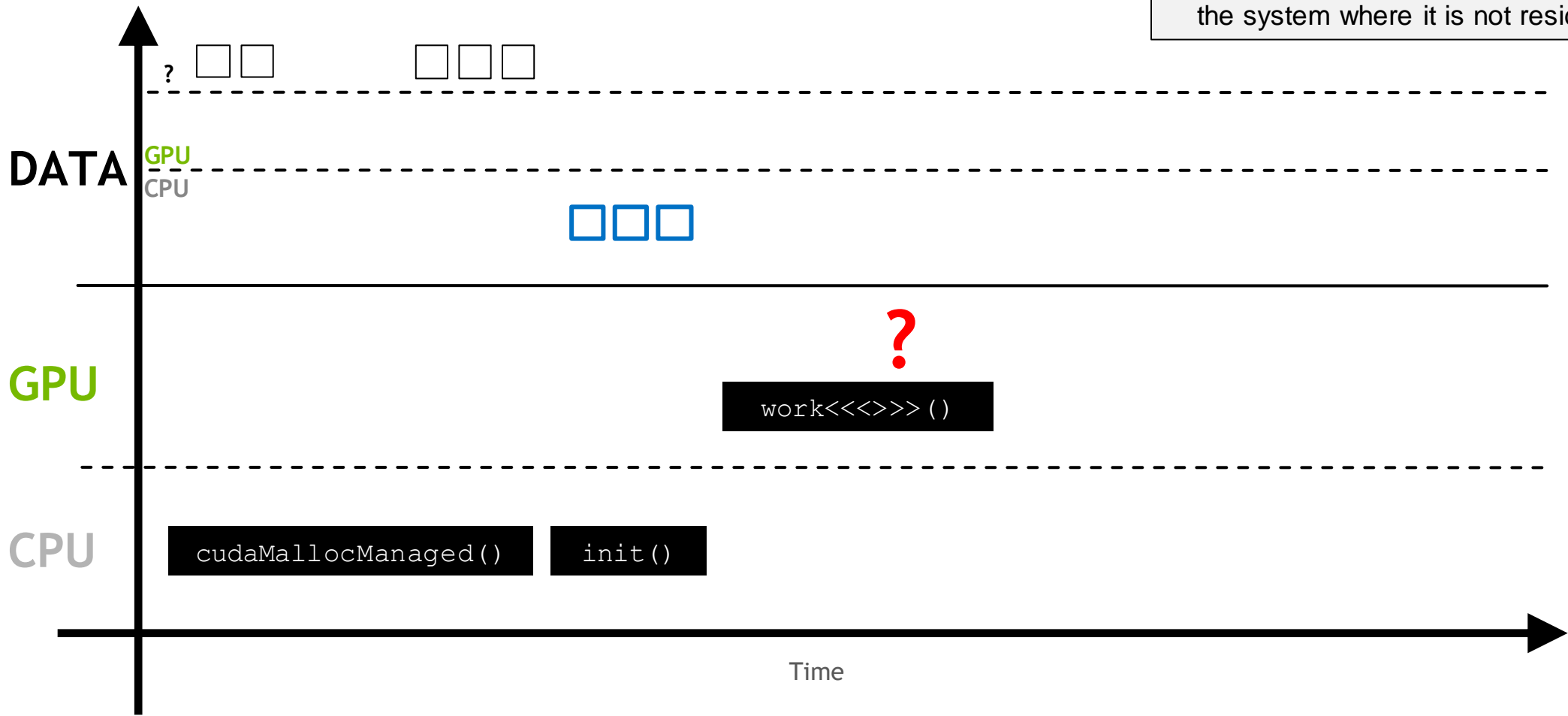
When some work asks for the memory for the first time, a **page fault** will occur



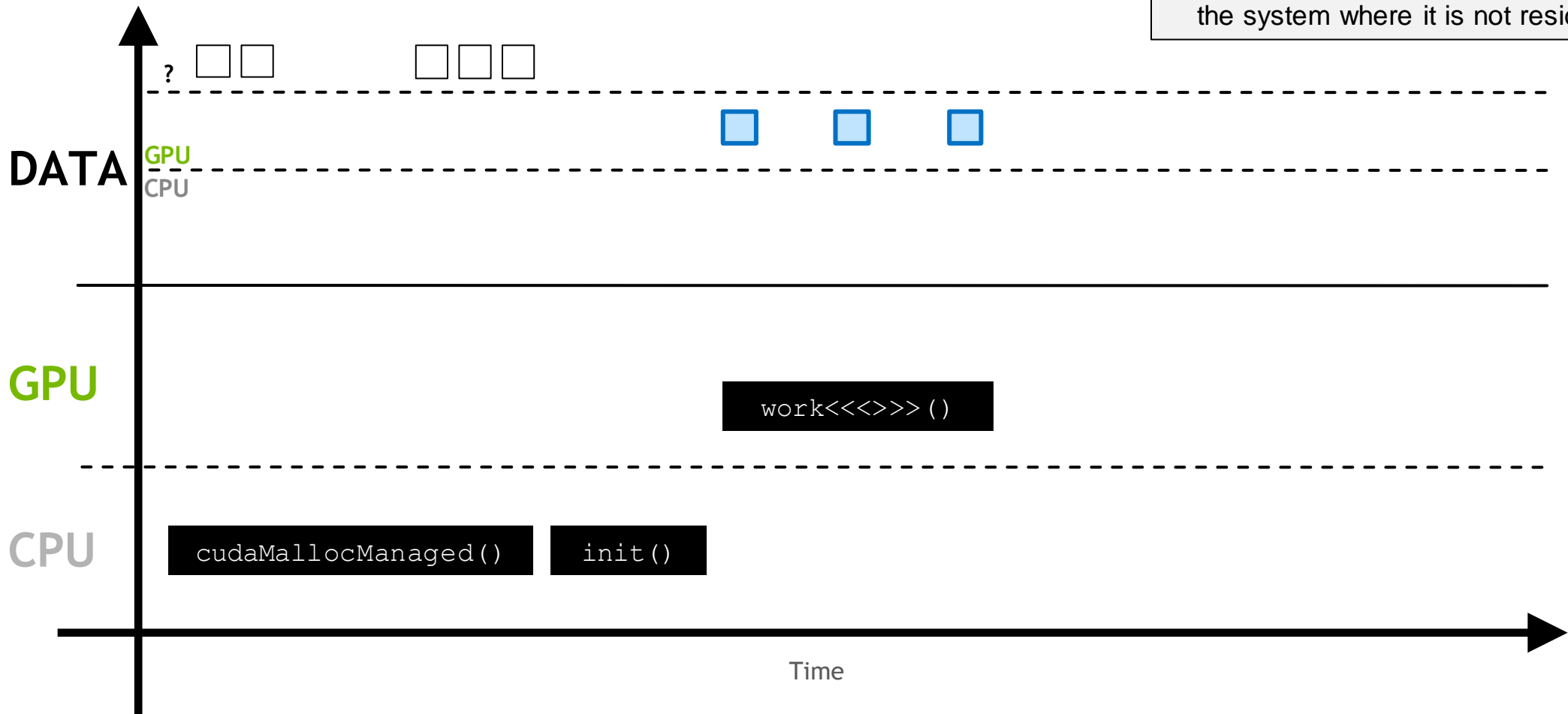
The page fault will trigger the migration of the demanded memory



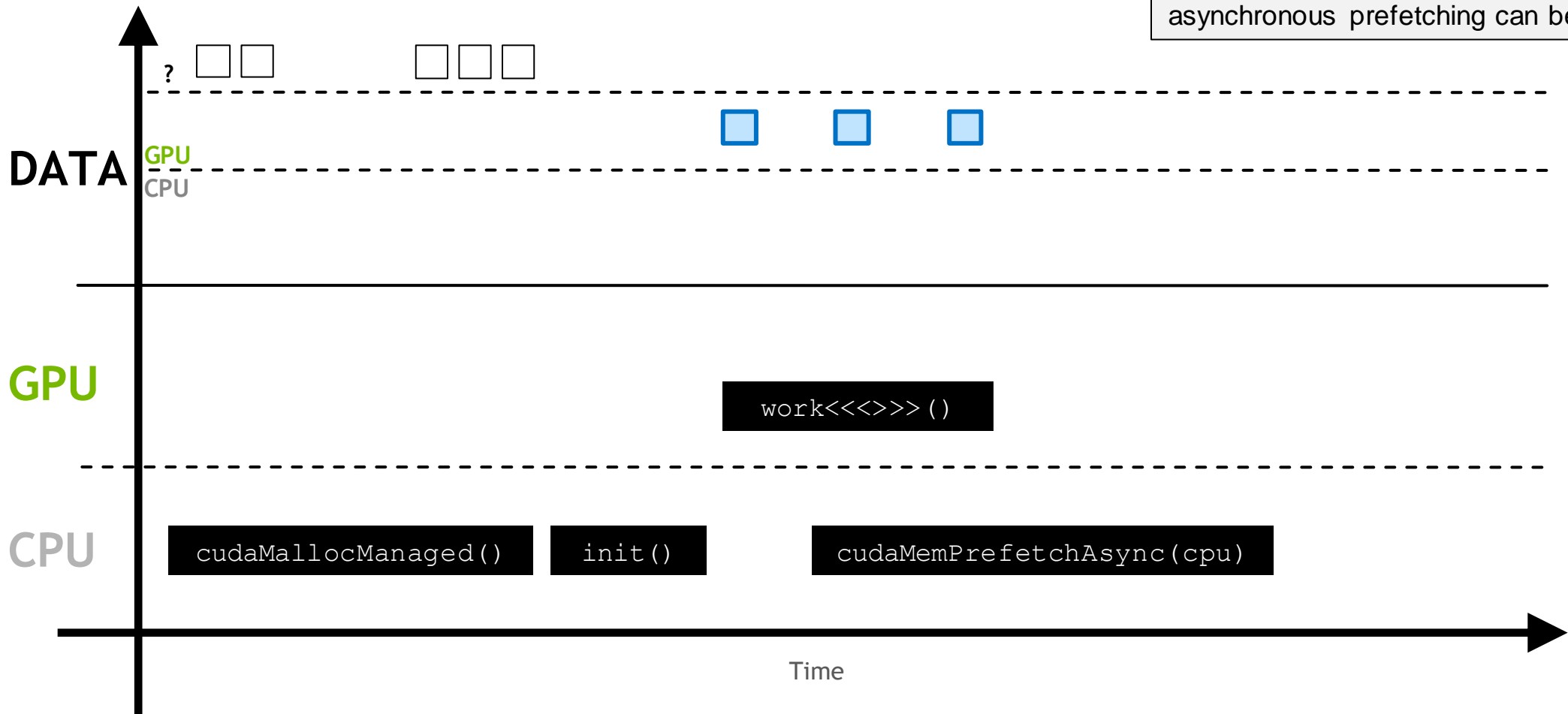
This process repeats anytime the memory is requested somewhere in the system where it is not resident



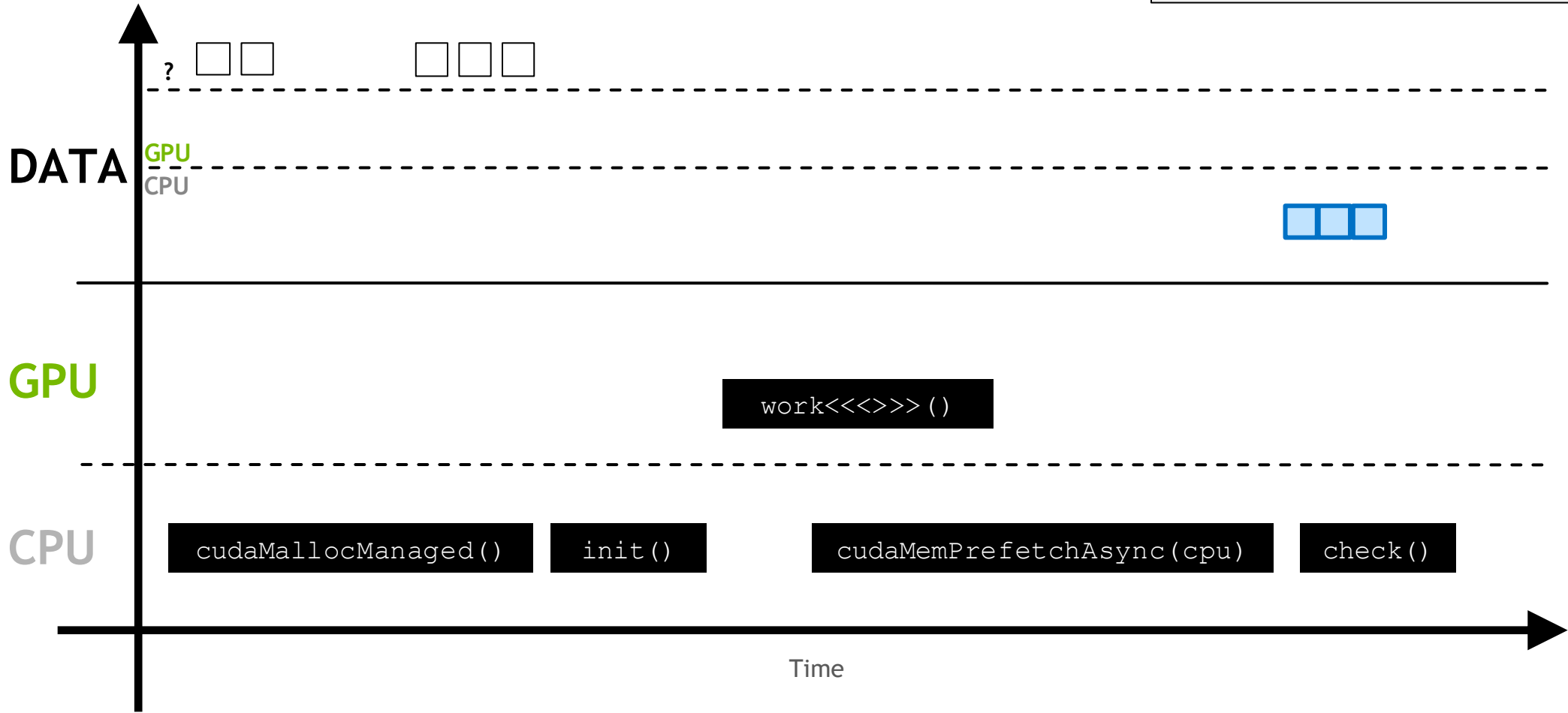
This process repeats anytime the memory is requested somewhere in the system where it is not resident



If it is known that the memory **will be** accessed somewhere it is not resident, asynchronous prefetching can be used



This moves the memory in larger batches, and prevents page faulting







DEEP  
LEARNING  
INSTITUTE

[www.nvidia.com/dli](http://www.nvidia.com/dli)