## **Unified Memory Behavior**

When <b>UM</b> is allocated, it may not be
resident initially on the CPU or the
GPU

DATA	CPU	
GPU		
CPU	<pre>cudaMallocManaged()</pre>	
	Time	



		When some work asks for the memory for the first time, a <b>page fault</b> will occur
_	·	
DATA	GPU CPU	
GPU		
CPU	<pre>cudaMallocManaged() init()</pre>	
	Time	

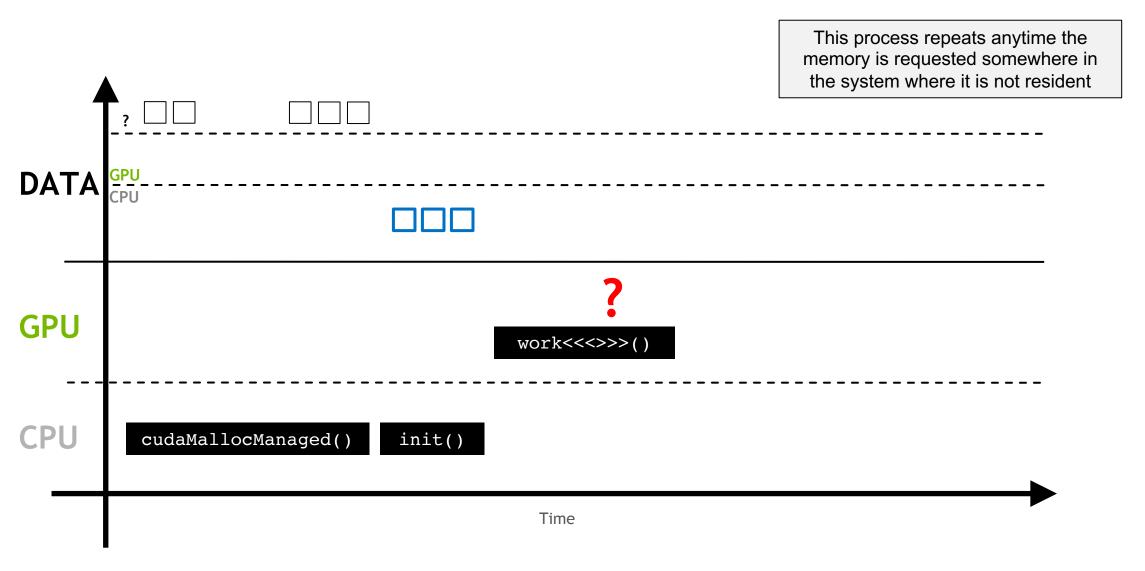
. . . . .



The page fault will trigger the migration of the demanded memory

	? 🗆 🔲	
DATA	GPU CPU	
GPU		
CPU	<pre>cudaMallocManaged() init()</pre>	••••••
	Time	







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

DATA	?			
GPU		work<<	<>>> ( )	
CPU	<pre>cudaMallocManaged()</pre>	<pre>init()</pre>		►
		Time		

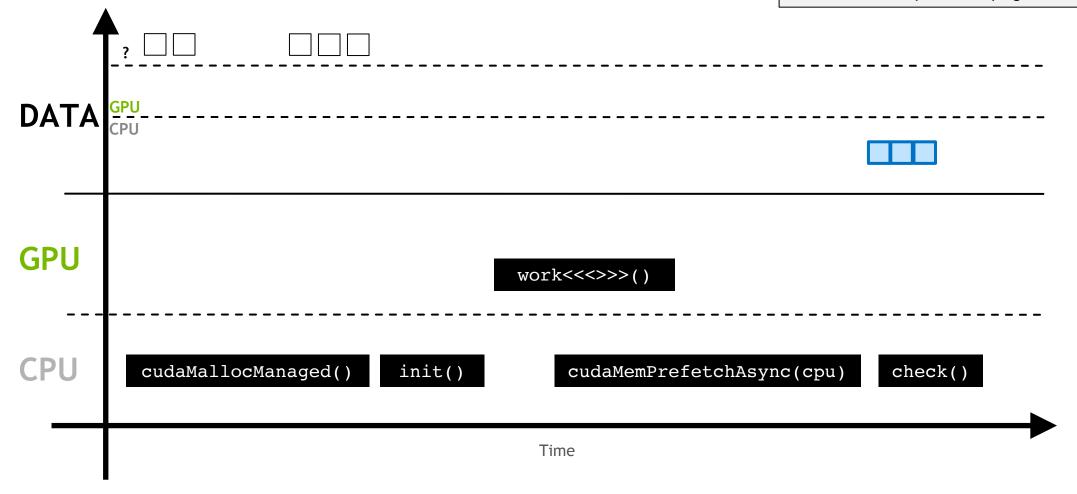


If it is known that the memory **will be** accessed somewhere it is not resident, asynchronous prefetching can be used

	? □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	
DATA	CPU	
GPU	work<<<>>>()	
CPU	<pre>cudaMallocManaged() init() cudaMemPrefetchAsyn</pre>	c(cpu)
	Time	



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







www.nvidia.com/dli