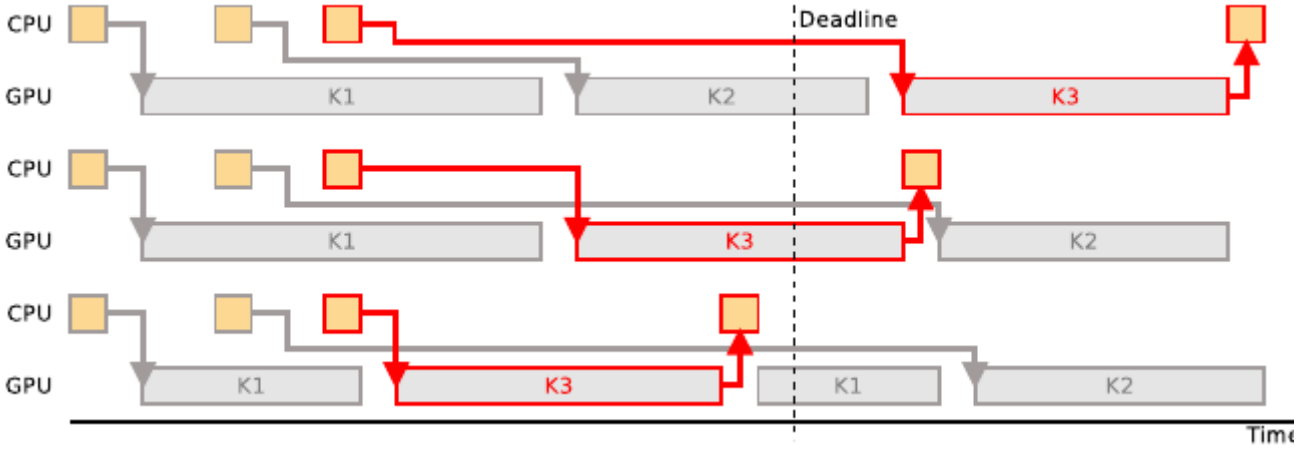
Preemptive multiprogramming on GPUs



Architectural support for better GPU integration with the operating system - we are developing a simulation infrastructure to enable us to move toward implement "preemptible faults" in Streaming Multiprocessors.

Summary

Implementing "preemptible faults" in SMs means being able to cancel instructions that cause a page fault and replay them at some point later. This enables the SM to do a CPU like context switch on faults. The challenge is that the SM doesn't have support for the precise exceptions used to achieve preemptable faults. Since introducing full blown precise exceptions support into GPU architectures is considered very costly, we are working on tweaking the SM architecture to find suitable alternatives.

Objectives

- Improve the performance in multiprogrammed / multi-tenant systems (e.g cloud) when UVM is used, since we are enabling faster and more predictable context switching.
- Running fault handlers on the GPU itself, which can improve the programmability and performance of future GPU systems (e.g. more flexible dynamic memory allocation support, memory mapped I/O...).
- Improving the system throughput in presence of faults causing long latency page migration (UVM).

Barcelona Supercomputing Center - Centro Nacional de Supercomputación