

CSE 60641: Operating Systems

Home work assignment #2

Open book, open notes, individual effort. Google searches are okay - always cite your source.
All questions carry equal points. Brevity is encouraged,

1. Adya et al. describe two axes: task management (serial, cooperative and preemptive) and stack management (manual and automatic). Event driven is “cooperative, manual”, multithreads is “preemptive, automatic” and their sweet spot was “cooperative, automatic”. Briefly describe why the other combinations: “serial, manual”, “serial, automatic”, “preemptive, manual” were not considered to be the “sweet spot”
2. In scheduler activations paper, the “kernel notifies the address space thread scheduler of every event affecting the address space. The kernel’s role is to vector events that influence user-level scheduling to the address space for the thread scheduler”. When a processor is preempted due to multiprogramming, the kernel notifies the application that two user-level threads have been stopped. Conversely, can a single scheduler notification inform the application that two user level threads can be scheduled because two virtual processors have just been assigned to the process? Explain.
3. In the continuations paper, in Table 1, discuss all the cases where continuations are not preferred? That is, under what conditions do you prefer to use the “Before continuations” code as opposed to “After continuations” code.
4. The resource container paper claims that (pg. 2) “In this paper, we address resource management in monolithic kernels. While microkernels and other novel systems offer interesting alternative approaches to this problem, monolithic kernels are still commercially significant, especially for Internet server applications”. Do you believe that microkernels and resource containers solve the same problem (using alternative approaches). If yes, describe the problem being solved by both the approaches. If no, describe whether resource containers are relevant for microkernels.