ECE5658: Operating Systems Design – week3 critique

2019710204 노대연

1. The Linux Scheduler: a Decade of Wasted Cores

(1) The Linux scheduler has some bugs. So, this paper study of four performance bugs: ① The group imbalance bug, ② The scheduling group construction bug ③ The overload-on-wakeup bug ④ The missing scheduling domains bug. These bugs induce some cores to idle state even though runnable threads are waiting for their turn to run. And these bugs are fatal because the Linux scheduler is work-conserving.

In this paper, the researchers designed two tools helping for checking the bugs. One is checker that periodically checks and catches the bugs on a live system, called online sanity checker. The other is visualization tool that visualizes trace of scheduling activity to expedite debugging, called scheduler visualization tool.

(2)

(+): They provide solutions about bugs which are hard to detect

(3) Are there other bugs that they do not cause the system crash, but eat away at performance?

2. Arachne: Core-Aware Thread Management

(1) Applications cannot tell the operating system how many physical cores they need, and they do not know which cores have been allocated for their use. As a result, they cannot adjust their internal parallelism to match the resources and optimize their use of resources. Arachne is core-aware thread management. So, Arachne can solve these problems by giving applications visibility into the physical resources they are using.

(2)

(+): Better throughput at low latency

(-): 1 thread combine with 1 core in managed cores is too wasteful

(3) I think 2:1 mapping is also good if not enough the number of managed cores.