CS423 Summary: The structuring of systems using upcalls

Hongpeng Guo

February 2, 2019

Area:

This paper designed a system with the synchronous procedure calls between protected layers. The area lies in operating system design.

Problem:

At the time of this paper, a system of a number of layers abstractions require communication via asynchronous inter-process messages exchanging between layers. However, such asynchronous communication usually causes serious problems to the operating systems. This paper proposed a method enabling synchronous procedure calls between different layers bidirectionally, which improve the system performance significantly.

Methodology:

The paper's methodology contains two key concept:

- **Upcalls:** The flow of controls from lower layers to higher layer through procedure calls.
- **Multi-task module:** Subroutines in different tasks that make up the layer.

By modeling the layers as multi-task module permitting upcalls, this paper achieved the synchronous communication between layers and improved system performance.

Solution:

- A layer of the operating system is made up of subroutines in different tasks.
- These subroutines can be called bidirectionally from layers with the same task.
- Inter-task communication can only happens horizontally over the same layer through shared state.

Results:

Comparing the classical THE type operating system, the upcall design is more convenient to implement, and also provide the possibility for lower layer to ask advice from higher layer, which improve the system performance.

There are also some disadvantages such as infinite loop of upcalls, and lower layer undefined behavior upon upcall failure.

Takeaway:

(1) Upcalls and multi-task module ideas can greatly improve the operating system performance. But the implementer should pay attention to avoid shared memory bugs. (2) The idea of this paper is inspired from network protocols design. Sometimes, inter-disciplinary knowledge can benefit each other.