

Zhenyu Yan

## **Liedtke's u-kernel construction**

by

Jochen Liedtke

In this paper, Liedtke points out micro kernel is widely believed that a) micro kernel is a based system that are inherently inefficient and b) they are not sufficiently flexible, He shows evidences that inefficiency and inflexibility of 1995 micro kernel is not because of inherited from the basic idea. The majority reason is overloading the kernel and improper implementation. Also, Liedtke describe concepts and illustrate the flexibility of micro kernel, show the performance and illustrate why micro kernel are not inherently not portable. Micro kernel try to approach to have a clear interface, server can use mechanism like another user programs, and more flexible and tailorable, but these approach is not generally accepted. In order to approach, Liedtke felt a conceptual analysis is need which devices micro kernel concepts form pure functionality requirement, performance, flexibility, portability and some new developments.

For concepts, first is address spaces. It leaves memory management and paging outside the micro kernel; only grant, map and flush operation are inside the kernel. Second is threads and IPC, cross address space communication must be supported by the micro kernel. It is the class method to transferring message between threads. Third is the unique identifier(uid) for threads, tasks or communication. For flexibility, a small set of micro kernel concepts lead to abstractions which stress flexibility, provided they perform well enough. The only thing which cannot be implemented on top of these abstractions is the processor architecture, registers, first-level caches and first-level TLBs. For performance, Compared to the theoretical minimum, the kernel user mode switch is costly in some processes. However, they can improve 6 to 10 times structural investigations with the appropriate  $\mu$ -kernel compared to existing kernels. Also the micro kernel architectures inherently that cause of memory system degradation is not confirmed. Instead, the referenced measurements ensure that the correct assumptions are supported. For portability, micro kernel from the link between a minimal micro set of abstraction and the bare processor, the performance requirements that are comparable to previous performance requirements. Therefore, micro kernel are inherently not portable and their processor dependent basic for portable operating system.

For the new development, he tanks about spin, utah-mach, dp-mach, panda, cache kernel, exokernel. Because this paper is from 1995 and it is out of date, these developments is very old to us. And in modern time, most of the operating system are hybrid kernel, which contain good mechanism of micro kernel and monolithic kernel. I think in the future, programmer will continue improve the implementation and change algorithms and data structures for the kernel and makes it faster and more secure.

The presenter Mahdi K and Reid Mikul were good. Their slide were detail and highlighted, but it was a little bit too much, and made me feel bored. They also showed us a video about micro kernel and how it works. Their slides covered all the important information about the whole paper, and I learned a lot form them.