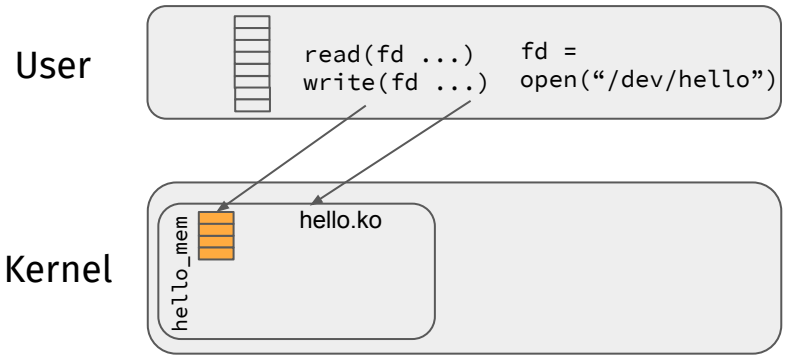
Kernel Space Programming (memory mapping)

Michio Honda

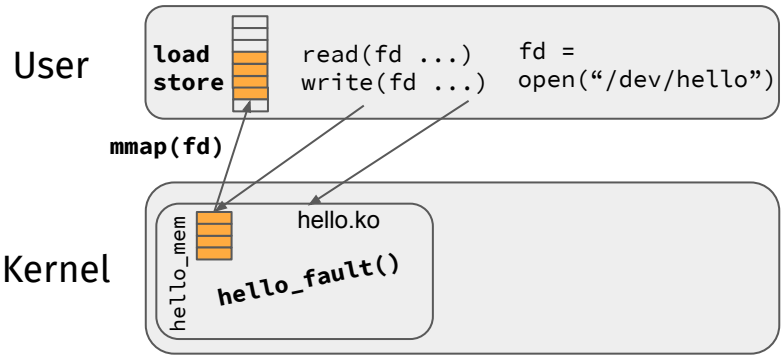
Recap

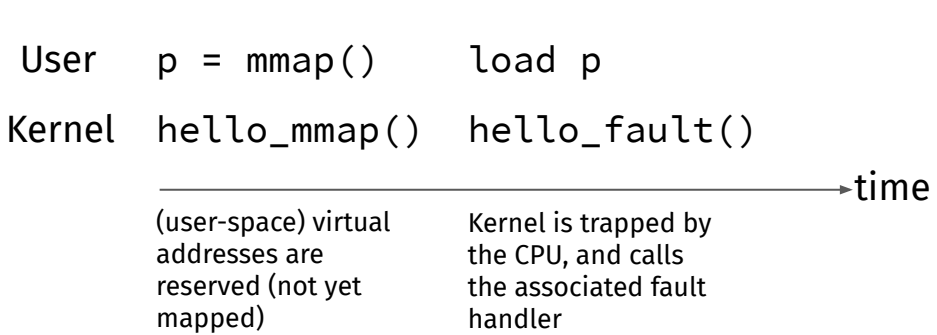
- **Read/write from/to the kernel as a file**



What we do today

- **Map the kernel memory into the user space**





- Why this is important?
  - Zero copy, system-call-less I/O
- How it works
  - User space accesses a virtual address not mapped to the physical memory
  - The CPU raises an interrupt to the kernel
  - Kernel load the physical page into the user's virtual address space

Practice

- Rearrange the key-value store in week3/3d2.c to work on an mmap()-ed kernel module