Operating System

05/10

1. 位于complex hardware和application之间; 每一个app都可能用到了很多硬件

人｜applications｜operating system｜hardware

1. Hide hardware complexity – 把底层硬件控制 抽象成命令
2. Resource management
3. Provide isolation and protection – 比如内存和storage读写 不会重叠
4. Abstraction vs arbitration
5. HDD (Hard disk drive 机械硬盘) / SSD (solid-state drive 固态硬盘) 区别

<http://zhidao.baidu.com/question/101021131.html>

1. 分为 Desktop， Embedded等

*Desktop*

Windows

Unix-based

- Mac OS X

- Linux (many versions)

*Embedded*

Android

iOS

1. OS elements: abstraction/mechanism/policy

Memory management example

1. User/kernel protection boundary – switch is supported by hardware (like some bit is set)

User level – applications

Kernel level – OS kernel, privileged direct hardware access

1. System call – to make a system call, an application need to
2. Write arguments
3. Save relevant data
4. Make system call

也就是User/kernel transitions (crossing the protection boundary)，有一定代价

1. Basic OS service:
2. CPU scheduler
3. Memory management
4. Block device driver
5. File system
6. *Monolithic OS:* 包含所有功能 大而全，缺点是hard to maintain, debug…

*Modular OS* (现在普遍使用的): 有一个基本的OS，之后根据应用，可以安装新的module interface

*Microkernel：*