

master ▾

MIT6.S081 / lec01-introduction-and-examples /  
1.1-ke-cheng-jian-jie.md

Go to file

...



huihongxiao GitBook: [master...



Latest commit 2e42474 on Oct 13, 2020



History

1 contributor

27 lines (17 sloc) | 4.2 KB



Raw

Blame



## 1.1 课程内容简介

大家好，欢迎来到6.S081 --- 操作系统这门课程。我的名字叫Robert，我会与Frans一起讲这门课。David，Nicholas会作为这门课程的助教。在我们在线授课的过程中，请尽情提问。你可以直接语音打断我，或者通过聊天窗口提问，我们工作人员会关注聊天窗口。顺便说一下，我们也会通过视频录制这门课程，之后我们会通过网络发布视频，这样方便你复习，对于没能参加课程的同学也可以通过视频观看。

接下来我想从列出这门课程的目标来开始。这门课程的目标有：

1. 理解操作系统的设计和实现。设计是指整体的结构，实现是指具体的代码长什么样。对于这两者，我们都会花费大量时间讲解。
2. 为了深入了解具体的工作原理，你们可以通过一个小的叫做XV6的操作系统，获得实际动手经验。通过研究现有的操作系统，并结合课程配套的实验，你可以获得扩展操作系统，修改并提升操作系统的相关经验，并且能够通过操作系统接口，编写系统软件。

# 6.S081 - OPERATING SYSTEMS

## - O/S DESIGN.

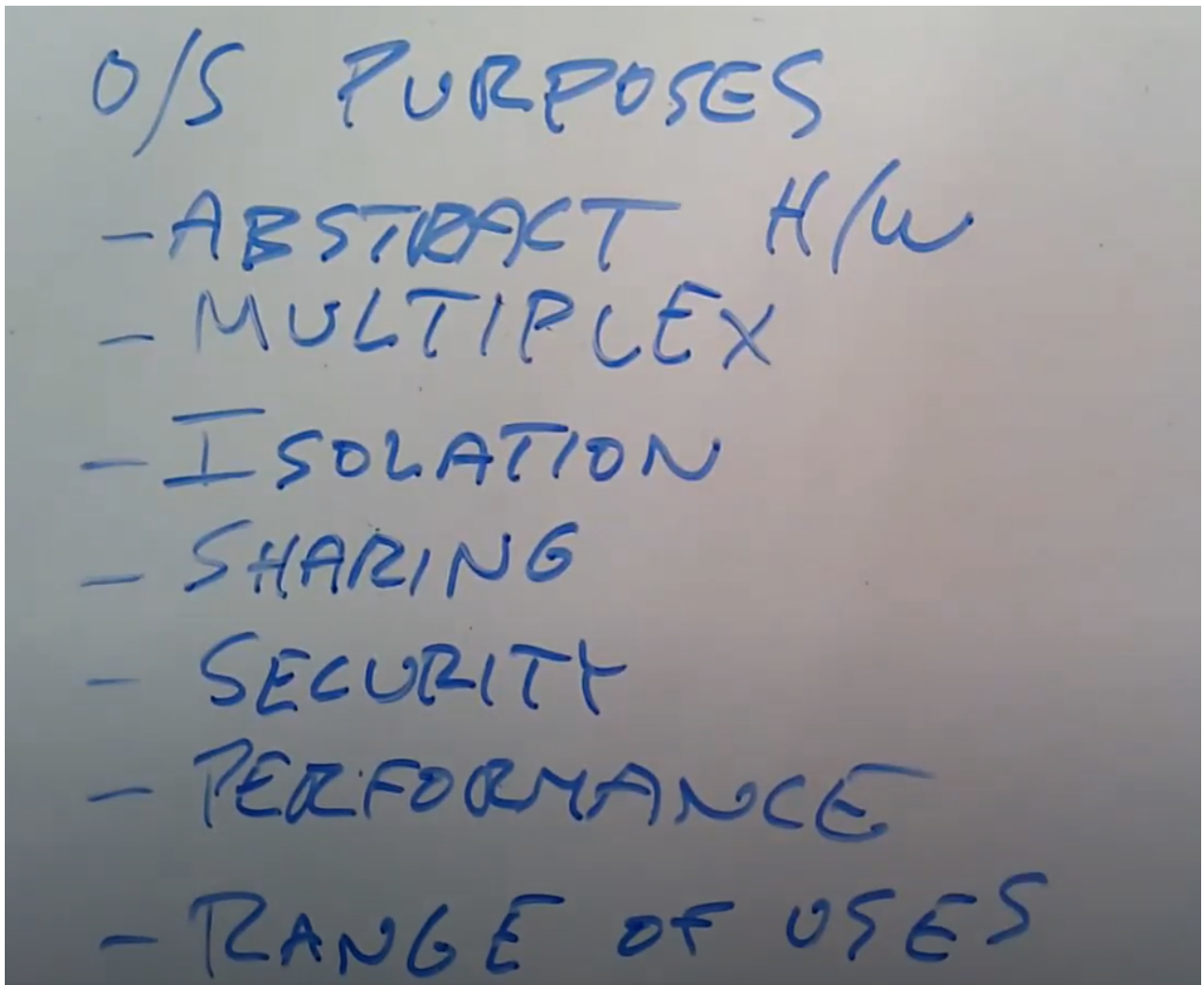
### - HANDSON EXP.

所以，这些就是你将要在课程中完成的内容。

我们同时也对操作系统本身的目标感兴趣。对于操作系统的目标，我也列出了几个点。你知道的，市面上有大量不同的操作系统，通常来说，他们都有一些共同的目标。

1. 第一个就是抽象硬件。通常来说，你会买一个计算机，里面包含了CPU，内存，但是这是一种非常低层级的资源。幸好我们有一些应用程序实现了高层级的接口和抽象，例如进程，文件系统。这些高层级的接口和抽象（Abstraction）方便了应用的开发，也提供了更好的移植性。
2. 操作系统的另一个重要的任务是：在多个应用程序之间共用硬件资源。你可以在一个操作系统同时运行文本编辑器，程序编译器，多个数据库等等。操作系统能非常神奇的在不相互干扰的前提下，同时运行这些程序。这里通常被称为multiplex。
3. 因为在操作系统中可能同时运行很多程序，即使程序出现了故障，多个程序之间互不干扰就变得非常重要。所以这里需要隔离性（Isolation），不同的活动之间不能相互干扰。
4. 但是另一方面，不同的活动之间有时又想要相互影响，比如说数据交互，协同完成任务等。举个例子，我通过文本编辑器创建了一个文件，并且我希望我的编译器能读取文件，我绝对想要数据能共享。所以，我们希望能在需要的时候实现共享（Sharing）。
5. 但是在很多场景下，用户并不想要共享，比如你登录到了一个公共的计算机，例如Athena，你不会想要其他人来读取你的文件。所以在共享的同时，我们也希望在没有必要的时候不共享。这里我们称为Security或者Permission System或者是Access Control System。
6. 另一个人认为操作系统应该具有的价值是：如果你在硬件上花费了大量的金钱，你会期望你的应用程序拥有硬件应该提供的完整性能，但是很多时候你只负责应用程序编程，你会期望操作系统也必须保证自身提供的服务不会阻止应用程序获得高性能。所以操作系统需要至少不阻止应用程序获得高性能，甚至需要帮助应用程序获得高性能（Performance）。
7. 最后，对于大部分操作系统，必须要支持大量不同类型的应用程序，或许这是一个笔记本，正在运行文本编辑器，正在运行游戏，或许你的操作系统需要支

持数据库服务器和云计算。通常来说，设计并构造一个操作系统代价是非常大的，所以人们总是希望在相同的操作系统上，例如Linux，运行大量的任务。我认为大部分人都已经跑过Linux，并使用了刚刚描述的所有的场景。所以，同一个操作系统需要能够支持大量不同的用户场景。



我们期望操作系统能够同时支持上面所有的目标。我们在这门课程中，会介绍更多有关这些目标的内容。