

## 第 12 章 关于内存虚拟化的对话

**学生：**那么，虚拟化讲完了吗？

**教授：**没有！

**学生：**嘿，没理由这么激动，我只是在问一个问题。学生就应该问问题，对吧？

**教授：**好吧，教授们总是这样说，但实际上他们的意思：提出问题，仅当它们是好问题，而且你实际上已经对这些问题进行了一些思考。

**学生：**好吧，那肯定会让我失去动力。

**教授：**我得逞了。不管怎么说，我们离讲完虚拟化还有一段时间！相反，你刚看到了如何虚拟化 CPU，但是真的有一个巨大的“怪物”——内存在壁橱里等着你。虚拟内存很复杂，需要我们理解关于硬件和操作系统交互方式的更多复杂细节。

**学生：**听起来很酷。为什么这很难？

**教授：**好吧，有很多细节，你必须牢记它们，才能真正对发生的事情建立一个思维模型。我们将从简单的开始，使用诸如基址/界限等非常基本的技术，并慢慢增加复杂性以应对新的挑战，包括有趣的主题，如 TLB 和多级页表。最终，我们将能够描述一个全功能的现代虚拟内存管理程序的工作原理。

**学生：**漂亮！对我这个可怜的学生有什么提示吗？会被这些信息淹没，并且一般都会睡眠不足？

**教授：**对于睡眠不足的人来说，这很简单：多睡一会儿（少一点派对）。对于理解虚拟内存，从这里开始：用户程序生成的每个地址都是虚拟地址（every address generated by a user program is a virtual address）。操作系统只是为每个进程提供一个假象，具体来说，就是它拥有自己的大量私有内存。在一些硬件帮助下，操作系统会将这些假的虚拟地址变成真实的物理地址，从而能够找到想要的信息。

**学生：**好的，我想我可以记住……（自言自语）用户程序中的每个地址都是虚拟的，用户程序中的每个地址都是虚拟的，每个地址都是……

**教授：**你在嘟囔什么？

**学生：**哦，没什么……（尴尬的停顿）……但是，操作系统为什么又要提供这种假象？

**教授：**主要是为了易于使用（ease of use）。操作系统会让每个程序觉得，它有一个很大的连续地址空间（address space）来放入其代码和数据。因此，作为一名程序员，您不必担心诸如“我应该在哪儿存储这个变量？”这样的事情，因为程序的虚拟地址空间很大，有很多空间可以存代码和数据。对于程序员来说，如果必须操心将所有的代码数据放入一个小而拥挤的内存，那么生活会变得痛苦得多。

**学生：**为什么呢？

**教授：**好吧，隔离（isolation）和保护（protection）也是大事。我们不希望一个错误的程序能够读取或者覆写其他程序的内存，对吗？

**学生：**可能不希望。除非它是由你不喜欢的人编写的程序。

**教授：**嗯……我想可能需要在下个学期为你安排一门道德与伦理课程。也许操作系统课程没有传递正确的信息。

**学生：**也许应该。但请记住，不是我对大家说，对于错误的进程行为，正确的操作系统反应是要“杀死”违规进程！