Lessons Learned

Looking back, there are two main lessons that can be drawn from Xen: the importance of paravirtualization, and the benefits of open source development.

Paravirtualization

Foremost is the success of paravirtualization. A famous quote reminds us:

> **Any problem in computer science can be solved with another layer of indirection. *But that usually will create another problem.***
>
> —*David Wheeler*

Virtualization is simply a form of indirection, and even though modern computers have hardware support for virtualization, naïve reliance on this support leads to poor performance. The same problems arise when you make naïve use of any type of virtualization.

For example, virtual memory uses a hard disk to provide the illusion of a vast amount of available memory. However, if you write a program that tries to use all of it as if it were real, physical memory, the performance will be atrocious. In this case you could imagine "paravirtualizing" that program to make it aware of the physical limits, changing the algorithms and data structures used to make it run efficiently in combination with the virtual memory system.

In the context of operating systems, Xen has shown that paravirtualization—whether it be adding a virtual driver, changing the operating system wholesale, or adding enlightenments to improve performance in select areas—is an important technique for improving performance when running in a virtual environment.

Open Source Development

Perhaps the boldest decision taken during Xen's development was choosing to make it available as open source software when other hypervisors were available only as proprietary software.

This decision has definitely benefited Xen because of the sheer amount of software that it has been able to harness: from the Linux kernel and the QEMU machine emulator, down to the tiny program that draws the Xen logo at boot time.[†] Without this software, the Xen project would have involved a huge amount of reimplementation. By including software from these other projects, Xen benefits when the software is updated, and the other projects benefit from patches submitted by Xen developers.

Xen, which started out as the part-time project of a single research student at the University of Cambridge, has grown to include over 100 contributors from around the world. Some of

[†] Figlet: *http://www.figlet.org*