David Choffnes

Disco: Running Commodity Operating Systems on Scalable Multiprocessors
E. Bugnion, S. Devine, M. Rosenblum (Stanford)

One-liner: The authors present a ccNUMA-aware virtual machine monitor that can run multiple commodity operating systems while reducing emulation overhead, sharing memory and scaling to multiple processors to provide performance better than that of individual commodity operating systems operating directly on the physical hardware.

The authors note that the biggest problem is that kernels for commodity OSes were not designed to scale to large numbers of processors or to ccNUMA systems. Using VMs can reduce the cost of “porting” these OSes to multiprocessors and can allow multiple applications to execute on different specialized operating systems concurrently. VMs also offer a weak notion of isolation. Challenges for VMs include overhead for emulation, resource management with incomplete info and communication among VMs. Disco aggressively shares memory, attempts to minimize ushering between the OS, VMM and physical hardware, streamline “network” communication among VMs on the same physical machine and transparently migrates/replicates pages in the ccNUMA architecture to improve performance.

A big flaw in this paper is that the authors were forced to use many “tricks” to garner respectable performance from Disco. The authors claim that little source code was modified but I'm not convinced that such changes are not trivial. To drive home this point, it's worth noting that they only made one OS work well in Disco, and there were two pages detailing the changes to Irix to make it work. Perhaps the worst flaw about this work is that the performance was not evaluated on real hardware. I really can't trust the results that they present. Further, I'm not convinced that the experiments represented a real workload.

The ideas presented in this paper, however, are eminently relevant today. Although many of today's OSes provide limited support for ccNUMA architectures, it's still easier to use virtual machines to take advantage of large numbers of processing elements than to write a multi-purpose OS to do so efficiently. A certain Peter Dinda's work focuses on the potential advantages of an economic model allowing users to purchase arbitrary computing power, which relies on virtualization to provide resources to users. Despite the fact that VMs concende performance to a properly implemented scalable single operating system, the virtualization approach is not going away.