

Mainframe Internet Integration

**Prof. Dr. Martin Bogdan
Prof. Dr.-Ing. Wilhelm G. Spruth**

SS2013

Virtualisierung Teil 5

Weiterführende Information

Vielleicht interessiert Sie hier ein Video vom IBM Vertrieb

Virtualization with IBM System z

<http://www.youtube.com/watch?v=4xL8s8WZUxo&NR=1>

Einen Aufsatz “A Comparison of Software and Hardware Techniques for x86 Virtualization” können Sie herunterladen unter

<http://www.informatik.uni-leipzig.de/cs/Literature/esiiisup/part01.pdf>

Falls Sie dies interessiert, enthalten die folgenden Seiten zusätzliche Informationen zu den Virtualisierungsproblemen bei der x86 Architektur.

Backup Information zu VMware

VMware ist ein Produkt mit Virtual Machine Hypervisor Funktionalität für die x86 Architektur und ist in zwei Versionen verfügbar: *VMware Workstation* und *ESX Server*.

VMware Workstation ist die ursprüngliche Version. Es handelt sich um ein als Host-OS bezeichnetes Anwendungsprogramm, das entweder unter Windows oder unter Linux auf einer x86 Plattform im Benutzer-Modus läuft, und als Kombination die Funktionalität eines Host-Kernels bereitstellt. Eine zusätzliche, als *VMDriver* bezeichnete Komponente wird in Windows oder Linux integriert. Es bestehen Ähnlichkeiten mit der Art, mit der eine virtuelle Maschine in einem DOS-Fenster unter Windows abläuft. VMware Workstation nutzt die existierenden Ein-/Ausgabe Einrichtungen des Host-Betriebssystems. Wenn ein Gast-Betriebssystem eine Ein-/Ausgabe Operation ausführt, wird diese von VMDriver abgefangen und unter Benutzung geeigneter Systemaufrufe interpretiert. Ebenso werden Ein-/Ausgabe Unterbrechungen unter Beteiligung des VMDrivers bearbeitet.

Das Host-OS kann Seiten auslagern, die einer spezifischen virtuellen Maschine zugeordnet sind. Lediglich ein kleiner Satz an Seiten wird ständig im Hauptspeicher gehalten. Dies bedeutet, dass VMware Workstation vom Host-Betriebssystem wie ein normaler Benutzer-Prozess behandelt wird. Falls der Algorithmus des Hosts zur Seitenersetzung nicht optimal arbeitet, kann dies zu einer VMware Leistungsver schlechterung führen.

Die Nutzung von Host-Betriebssystem Komponenten verursacht einen Leistungsverlust, insbesondere bei Ein-/Ausgabe-Operationen. VMware stellt deshalb mit seinem ESX Server einen eigenen Host-Kernel zur Verfügung. Dieser benötigt keine Unterstützung durch das Host-Betriebssystem, läuft auf der realen Hardware und hat vergleichbare Funktionen wie der VM/370 Host-Kernel. Es werden bis zu 64 virtuelle Maschinen unterstützt. Gast-Betriebssysteme, die unter dem ESX Host-Kernel laufen, verfügen über einen mit der Adresse 0 beginnenden linearen virtuellen Adressenraum. Die Adressumsetzung erfolgt ähnlich wie bei VM/370 mit Hilfe von *shadow page tables* und einer als *pmap* bezeichneten Datenstruktur.

Gast-Maschinenbefehle, die Gast-Seitentabellen oder den TLB abändern, werden vom ESX Host-Kernel abgefangen und interpretiert. Der TLB enthält hierzu immer die in den shadow page tables enthaltenen Adressumsetzungen. Die wichtigsten Ein-/Ausgabe-Treiber sind in bezug auf maximale Leistung im Gastbetrieb-Modus optimiert. Die derzeitige Version ist in der Lage, einen symmetrischen Multiprozessor (SMP) mit bis zu 2 CPUs zu unterstützen.

Im Vergleich zu VM/370 sind der ESX Server und VMware benachteiligt, weil einige kritische Eigenschaften in der x86-Architektur fehlen. Für den Betrieb von Gast-Maschinen ist es erforderlich, dass alle Maschinenbefehle, welche den privilegierten Maschinenstatus abändern oder auch nur lesen, nur im Kernel-Modus ausgeführt werden können.

Dies sei an einem Beispiel erläutert. Wenn ein Gast ein Kontroll-Register schreibt, muss der Host-Kernel diesen Maschinenbefehl abfangen, damit nicht das reale Kontroll-Register des Hosts verändert wird. Der Host-Kernel wird jetzt nur die Effekte der Instruktion für diesen Gast simulieren. Liest der Gast anschließend diese Kontroll-Register wieder aus, so muss diese Instruktion ebenfalls abgefangen werden, damit der Gast wieder den Wert sieht, den er vorher in das Register geschrieben hat, und nicht etwa den realen Wert des Kontroll-Registers, der nur für den Host sichtbar ist.

Da die x86 Architektur diese Bedingung nicht erfüllt, ist es nicht möglich, wie unter VM/370 alle Maschinenbefehle einfach im Benutzer-Modus auszuführen, und auf Programmunterbrechungen zu vertrauen, wenn auf privilegierten Maschinenstatus Information zugegriffen wird. Beispielsweise:

Many models of Intel's machines allow user code to read registers and get the value that the privileged code put there instead of the value that the privileged code wishes the user code to see

VMware's ESX Server überschreibt hierzu dynamisch Teile des Gast-Kernels und schiebt Unterbrechungsbedingungen dort ein, wo eine Intervention des Host-Kernels erforderlich ist. Als Folge hiervon tritt ein deutlicher Leistungsverlust auf, besonders bei Ein-/Ausgabe-Operationen. Manche Funktionen sind nicht vorhanden oder können nicht genutzt werden. Kompatibilitätsprobleme treten auf; es kann sein, dass bestimmte Anwendungen nicht lauffähig sind.

Weitere Implementierungen

Ein anderes Produkt für die x86 Plattform ist VirtualPC von Microsoft. VirtualPC wurde ursprünglich von Connectix entwickelt.

VirtualBox ist eine von der Firma innotek (von Sun Microsystems übernommen, mittlerweile zu Oracle gehörend) entwickelte Virtualisierungslösung, die es dem Benutzer erlaubt, weitere Betriebssysteme (Gäste) unter einem laufenden System (Host) zu installieren und wie eine normale Anwendung zu nutzen. Als Hostsysteme werden Windows ab XP, Mac OS X, Linux (ab Kernel 2.4) und FreeBSD (ab 7.0), als Gastsysteme neben diesen zusätzlich noch Windows NT und 2000, OS/2, DOS-basierte Betriebssysteme, Linux (ab Kernel 2.2), L4, Solaris, NetWare sowie diverse BSD-Derivate unterstützt. In einer 2010 durchgeführten Survey durch LinuxJournal.com und LifeHacker.com, war VirtualBox das populärste Virtualisierungs Produkt mit über 50% der Stimmen.

Seit dem 15. Januar 2007 steht VirtualBox in zwei Versionen zur Verfügung: Eine Open-Source-Edition (kurz: OSE), die unter der GPL v2 veröffentlicht wird, und eine PUEL-Variante (Personal Use and Evaluation License), die unter bestimmten Bedingungen kostenlos verwendet werden darf. Diese Bedingungen sind recht weit gefasst, so dass der Gebrauch am heimischen PC bedenkenlos möglich ist.

Kernel-based Virtual Machine (KVM) ist eine Virtual Machine Implementierung, die den Operating System's Kernel benutzt. Dies ermöglicht eine bessere Performance als Lösungen, welche User-Space Driver einsetzen, In den meisten Fällen verwendet KVM eine Linux Kernel Virtualization Infrastruktur. KVM unterstützt native Virtualization für x86 Prozessoren, welche die Intel VT-x oder AMD-V Erweiterungen unterstützen. Es wurde auch auf die S390, PowerPC, und IA-64 (Itanium) Plattformen portiert. Ein ARM Port findet derzeit statt

KVM, unterstützt zahlreiche Gastbetriebssysteme, z.B. Linux, BSD, Solaris, Windows, Haiku, ReactOS and AROS Research Operating System, Eine modifizierte Version von Qemu ermöglicht die Benutzung von Mac OS X als Gast.

Die seit Herbst 2010 verfügbare z196 Mainframe Version mit ihrer zEnterprise Blade Extension (zBX) macht umfangreichen Gebrauch von KVM.

Paravirtualisierung

Mehrere Projekte adressieren das Problem der fehlenden Virtualisierungseinrichtungen der Host-Architektur, indem sie das Gast-Betriebssystem abändern.

Xen ist ein GNU Open Source Software Virtual Machine Monitor, der derzeit von der Systems Research Group des University of Cambridge Computer Laboratory entwickelt wird und ebenfalls auf der x86 Plattform läuft. Um die Probleme mit der ursprünglichen x86 Architektur zu umgehen, wird in einer Version ein als *Paravirtualization* bezeichneter Ansatz verwendet. Hierbei ist die Architektur der virtuellen Maschine nicht vollständig mit der Host Architektur identisch. Die Benutzerschnittstelle (Application Binary Interface, ABI) ist die gleiche. Der Gast-Kernel unterstellt jedoch Abweichungen zu der x86 Architektur. Dies verbessert das Leistungsverhalten, erfordert aber Änderungen des Gast-Kernels. Hiervon ist nur ein sehr kleiner Teil des Kernel-Codes betroffen. Derzeit existiert ein funktionsfähiger Linux-Port (XenoLinux), dessen ABI mit dem eines nicht-virtualisierten Linux 2.4 identisch ist. An Portierungen für Windows XP und BSD wird gearbeitet.

Die Firma SW-Soft setzt ihre Virtuozzo Virtual Private Servers (VPS) Software vor allem für Hosting Services bei Hosting Providern ein. Als Konkurrent tritt die Firma Ensim mit ihrer konzeptuell ähnlichen Extend Software auf. SW-Soft bezeichnet seine Produkte als *virtuelle Server*. Der als Virtual Environment (VE) bezeichnete Gast ist vom gleichen Typ wie das Host-Betriebssystem, normalerweise Linux. Er erscheint nach außen hin wie ein kompletter Server, verfügt aber über keinen eigenen Kernel, sondern benutzt stattdessen die Kernel Funktionen des Hosts. Zwischen dem Host-Betriebssystem und den virtuellen Environments befindet sich eine Zwischenschicht, die die Steuerung und Verwaltung der VEs übernimmt. Sie erweckt den Anschein, als ob eine Gruppe von Anwendungsprozessen auf getrennten Instanzen des Kernels läuft. Die VEs sind gegeneinander abgeschottet, können unabhängig voneinander gestartet und beendet werden und besitzen je ein eigenes Dateisystem.

Ein ähnlicher Ansatz wird von *Denali* verfolgt. Als Gast-Betriebssystem dient *llwaco*, eine speziell an den Denali Hypervisor angepasste BSD Version. Denali unterstützt nicht das vollständige x86 ABI. Es wurde für Netzwerk-Anwendungen entwickelt und unterstellt Einzelbenutzer-Anwendungen. Mehrfache virtuelle Adressräume sind unter llwaco nicht möglich.

Disco ist ein Hypervisor für einen ccNUMA Cluster mit MIPS R10000 Prozessoren und IRIX 5.3 als Gast-Betriebssystem. Auch die MIPS Architektur gestattet keine vollständige Virtualisierung des virtuellen Adressenraums des Kernels. Ähnlich Xen wurde der IRIX 5.3 Gast-Kernel für einen Betrieb unter Disco leicht abgeändert.

Plex86 ist ebenfalls ein Open Source Projekt. Als Gast Betriebssystem wird ein abgeändertes Linux verwendet, welches z.B. keine Ein-/Ausgabe Unterstützung enthält. Ein-/Ausgabe Operationen werden von einem Hardware Abstraction Layer direkt an den Plex86 Hypervisor weitergereicht.