

# Die neue Serenity Virtual Station

LEIF BRÄUER, SYSTEM- UND UNTERNEHMENSBERATER, UND OLIVER MARK, SENIOR IT ARCHITECT

Die Argumente für virtuelle Maschinen (VM) sind gerade im Linux-Bereich bekannt, nicht erst seit mit den beiden Großen dieser Branche, EMC<sup>2</sup> VMWare und Microsoft Virtual PC 2004, das Thema in den Mittelpunkt des Interesses gerückt ist:

Ihr aktueller PC verwendet ein modernes Betriebssystem, beispielsweise Linux, fast alle Anwendungen sind dafür verfügbar, doch eine ganz bestimmte »Killerapplikation« gibt es nur für DOS, Windows 95, Windows NT 4 oder gar für das ewig totgesagte IBM OS/2. Oder Sie sind Entwickler oder Tester oder arbeiten in der Hotline und müssen mit mehreren unterschiedlichen Konfigurationen und Betriebssystemen arbeiten. Vielleicht sind Sie auch Systemberater oder Techniker und haben für Präsentationen und Workshops am besten ein ganzes Netzwerk mit Servern und Arbeitsplätzen im Gepäck.

Die Lösungsansätze für alle diese Probleme waren in der Vergangenheit gezwungenermaßen kreativ, Multiboot-Konfigurationen oder Wechselplatten gehören zum Repertoire, bot ein gängiger PC doch gerade genug Leistung für den Betrieb eines einzelnen Systems. Praktisch war das selbstverständlich nicht, die Anwendung oder die Daten, die gerade gebraucht wurden, befanden sich gemäß Murphy's Law immer auf der Festplatte oder Partition, die gerade nicht im Zugriff war.

## Die Alternative

Mit modernen Prozessoren und Hauptspeichern im GByte-Bereich hat sich die Situation geändert: Virtualisierung ist zu einer praktikablen Lösung geworden, seit dessen

Vorreiter VMWare mit seinen Produkten ESX und GSX den Beweis erbracht hat, selbst Server virtuell über Monate bereitstellen zu können.

Windows NT, Windows 2000, Linux und IBM OS/2 sind die gängigen Betriebssysteme der Kunden, die die Systemberater der Autoren betreuen. Was tun, wenn man nur einen Laptop besitzt und so viele Umgebungen warten muß? Die Versuche mit VMWare und Virtual PC waren eine immense Erleichterung, jedoch nicht die Lösung: Bei dem einen fehlt der Support für OS/2, bei dem anderen der für Linux, mal machten Stabilitätsprobleme zu schaffen.

Auf der Suche nach einer Alternative stießen wir auf die Serenity Virtual Station, kurz SVISTA[1], der amerikanischen Firma Serenity Systems [2]. Die Featureliste der virtuellen Maschine liest sich wie die der Mitbewerber; das Produkt ist eine Besonderheit durch die konkurrenzlose Zahl unterstützter Host- und Gast-Betriebssysteme. Neben Windows 2000 und XP und den verschiedensten Linux-Distributionen stehen sogar FreeBSD und IBM OS/2 sowie die herstellereigene eComStation auf der Liste.

Die Anforderungen an den PC sind mit jedem aktuellen System leicht zu erreichen: ein Prozessor mit 800

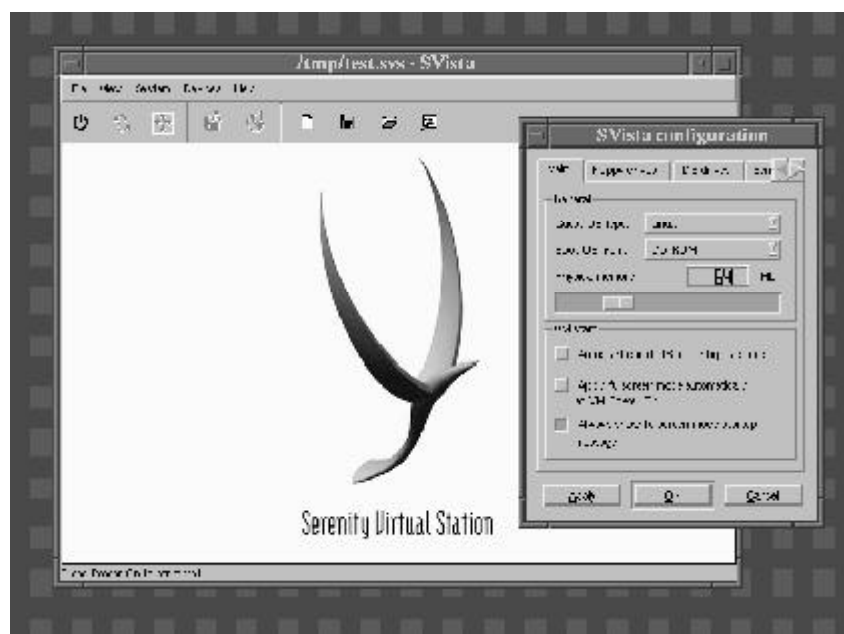


Bild 1: Die virtuelle Maschine

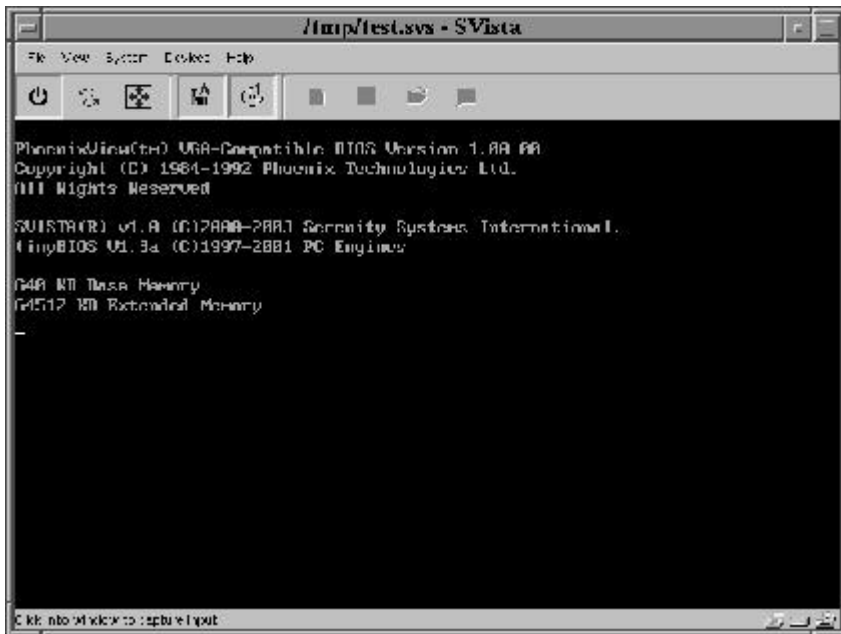


Bild 2: Das Booten der virtuellen Maschine für die Betriebssysteminstallation

MHz, genügend RAM für das Host-OS einschließlich SVISTA und das Gast-OS. In der Praxis ist es möglich, Linux mit installierter virtueller Maschine und beispielsweise Windows NT 4 als Gast mit 256 MByte RAM laufen zu lassen. Da der Speicher aus Performance-Gründen nicht ausgelagert werden kann, gilt hier im besonderen, daß Speicher nur durch mehr Speicher zu ersetzen ist. Mit 512 MByte ist ein vernünftiges Arbeiten mit mehr als einer laufenden VM möglich. Bei den virtuellen Festplatten ist leider eine Beschränkung auf eine virtuelle Festplatte mit maximal vier GByte pro VM gegeben. Die gute Nachricht dagegen ist, daß diese Images unabhängig vom Host-Betriebssystem sind. Zentral auf einem File-Server können so Windows, OS/2 oder Linux-Anwender gemeinsam die vorbereiteten virtuellen Systeme nutzen. Die Anforderungen an die Grafikkarte sind ebenfalls moderat: Sie muß vom Grafiksystem des Betriebssystems (wie zum Beispiel XFree86) mit mindestens 8 bpp Farbtiefe unterstützt werden. Jede von Linux unterstützte Ethernet-Netzwerkkarte und Token Ring (gut für OS/2 und seine Fans bei Banken und Versicherungen) vervollständigt die Liste.

Bei Erscheinen der ersten Version von SVISTA sollen folgende Host-OS freigegeben sein: Windows NT 4 SP6, Windows 2000 SP2, Windows XP und Windows Server 2003, Linux mit Kernel 2.4 oder höher, IBM OS/2 und eComStation sowie FreeBSD. Auf der Seite der Gast-OS sind es IBM OS/2 2.1, Warp 3, Warp 4, Warp Server for eBusiness, Warp 4 Convenience Pack, eComStation 1.1, Windows NT 4 und DOS. Weitere Systeme, wie zum Beispiel Windows 2000 und XP sowie Linux laufen noch langsam, sind jedoch bereits mit genereller Verfügbarkeit des Produkts angekündigt. SVISTA bildet einen Standard-PC mit Intel Pentium CPU, VESA-Grafikkarte, leider zur Zeit nur maximal 256 MByte Speicher, vier seriellen und drei parallelen Schnittstellen und einer Netzwerkkarte (NE2000, Realtek 8029 oder Token Ring) ab. USB und Sound sollen in der kommenden Betaversion verfügbar sein. Einen ersten Blick darauf konnten wir darauf schon werfen, für einen Test reichte es dagegen noch nicht. Der Ansatz, Standardkomponenten nachzubilden, macht einem auch mit exotischen Betriebssystemen das Leben leicht: Selbst DOS oder Windows 3.11 bieten Treiber für NE2000 oder VESA.

Nun sollte sich die aktuelle Version 1.0 Build 42 unter SuSE 9.0 im Alltags-test auf einem IBM Thinkpad T40 mit Centrino Chipsatz und 512 MByte RAM beweisen. Zur Installation kann entweder ein RPM-Paket oder ein Archiv mit entsprechenden Installations- und Deinstallationskripten verwendet werden. Die Installation inklusive Kompilierung der Module ging problemlos, für einige Kernelversionen liegen die Module bei, ansonsten werden sie während der Installation kompiliert, folglich werden die entsprechenden Kernelquellen benötigt, zusätzlich das QTK.

## Die virtuelle Maschine

Es gab an anderer Stelle Berichte über Schwierigkeiten beim Kompilieren unter Red Hat, die mit den Vanillakernels von [ftp.kernel.org](http://ftp.kernel.org) umgangen werden konnten[3]. Nach weniger als fünf Minuten und ohne Neustart zeigt sich das Programm mit der Darstellung in Bild 1. Die Anwendung ist einfach, die Bedienung ohne Schnörkel und ohne Besonderheiten. Jeder, der mit VMWare oder Virtual PC vertraut ist, kommt schnell mit SVISTA klar. Uns gefielen besonders die ASCII-basierten Konfigurationsdateien, die einfach mit Skripten erstellt werden können und bis auf Pfadnamen sogar plattformübergreifend genutzt werden können. Schnell sind so einige Icons auf dem Desktop, mit denen man die jeweiligen Kunden- oder Präsentationsumgebungen im Überblick hat. Die ersten Tests absolvierte SVISTA mit MS-DOS sowie Windows NT 4. Die Anbindung ans Netzwerk, die Installation weiterer Software – alles lief so, wie man es auch von seinem echten PC kennt. Die Einbindung eines IBM OS/2-Warp-4-Gasts war der abschließende Härtestest. Die Installation und Hardwareerkennung lief erwartungsgemäß problemlos. Nach Abschluß der Installation fügten wir den Webexplorer (Mozilla)



Bild 3: Die Installation von OS/2 in der virtuellen Maschine

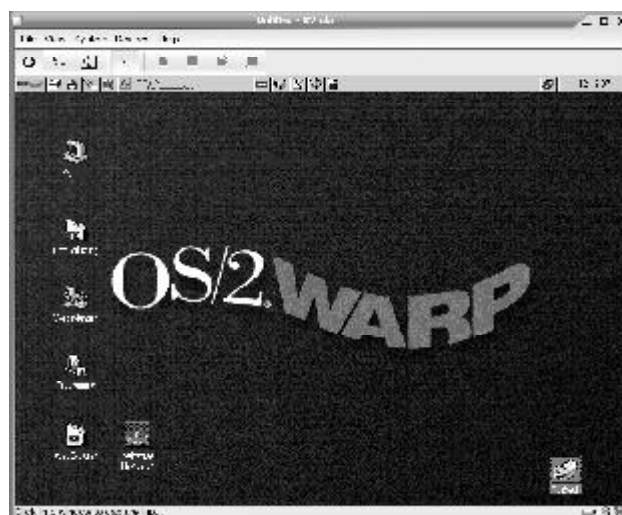


Bild 4: OS/2 im Fenster

und OpenOffice (beides für OS/2 verfügbar) für den abschließenden Praxistest hinzu. Ob das Erstellen von Präsentationen, Surfen oder E-Mails, Drucken, Verwenden von Netzwerkressourcen – keine Aufgabe war schwieriger zu bewältigen als auf einem realen System.

## Praxistest

Auch wenn auf Performancemessungen bei dieser Pre-Beta bewußt verzichtet wurde, kann man schon sagen, daß die Entwickler ein gutes Produkt abgeliefert haben und daß es schon jetzt sehr viel versprechend aussieht. Interessierte finden einige Performance-Vergleiche mit Virtual PC, der Betaversion von SVISTA und der realen Hardware unter [4]. Weitergehende Tests von David Haring unter anderem mit »Damn small Linux« und Vector Linux sind bei [3] zu finden. Liest man die Pläne und Konzepte auf der Produkt-homepage [1], ist noch einiges in der Zukunft zu erwarten: Interessant sind insbesondere serverbasierte Konzepte, bei denen SVISTA in Serverfarmen mit zentralisiertem Management und Load-Balancing betrieben wird.

Zu den Kritikpunkten zählen die jetzt noch vorhandenen Begrenzungen des Hauptspeichers und der Größe der virtuellen Festplatten sowie die Tatsache, daß die Tastenkombination, mit der die VM wieder verlas-

sen werden kann, nicht frei definierbar ist. Schmerzlich vermißt wurde auch ein Suspend-Mode, also die Möglichkeit, eine VM »schlafen« zu legen. Allerdings ist das alles auf der Todo-Liste der Entwickler. Seien wir gespannt, welche dieser Komfort-Features es in die erste Version schaffen werden.

Serenity hat für Mitte Mai ein erweitertes Betaprogramm angekün-

digt. Das Produkt soll am Anfang des dritten Quartals 2004 zu einem Preis von etwa 100 US-Dollar in den Handel kommen, der Preis für Deutschland stand noch nicht fest. Während der virtuelle Maschinenkern durch eine kommerzielle Lizenz geschützt wird, wird die Benutzerschnittstelle voraussichtlich unter eine freie Lizenz (vermutlich LGPL) fallen. ♦

### Links

- [1] <http://www.serenityvirtual.com>
- [2] <http://www.serenitysystems.com>
- [3] [http://www.osnews.com/story.php?news\\_id=6742](http://www.osnews.com/story.php?news_id=6742)
- [4] <http://xoomer.virgilio.it/acantato/articles/2004/vmtest.html>

### Übersicht:

Produkt:	Serenity Virtual Station Version 1.0
Hersteller:	Serenity Systems
Minimale Hardware-Voraussetzungen:	
CPU:	Pentium 800 MHz
RAM:	256 MB
Festplatte:	10 GB
Netzwerk:	Ethernet, Token Ring
Grafikkarte:	VESA oder besser (Support für X11, Windows, OS/2 PM)
Verfügbar für:	Linux (ab Kernel 2.4.19), FreeBSD, Windows, OS/2
Verfügbarkeit:	Drittes Quartal 2004
Preis:	Etwa 100 US-Dollar