

# Virtualisierung

Christian Voshage

11. Mai 2009

# Inhaltsverzeichnis

- 1 **Überlegungen**
  - Grundüberlegungen
  - Vorteile
  - Hardware-Emulation
  - Nachteile
- 2 **Einsatzgebiete**
  - Servervirtualisierung
  - Clientvirtualisierung
- 3 **Virtualisierungslösungen**
  - Applikationsvirtualisierung
  - Betriebssystemvirtualisierung
  - Virtual Machine Monitor
  - Paravirtualisierung
- 4 **Fazit**
- 5 **Hardwarevirtualisierung**

# Überlegungen

- Mehrere Instanzen (Partitionierung)
  - verhalten sich wie volle Maschinen (eigene CPU, RAM, Grafikkarte etc.)
    - virtualisierte Hardware sollte am Besten unabhängig von der echten Hardware einrichtbar sein
  - Virtualisierung soll nicht von Software erkannt werden
    - Erkennung der oder eine Anpassung für die Virtualisierungssoftware kann bei Entwicklungen stören
- Eine Instanz aus mehreren Rechnern (Konnotation)
  - Physikalisch Getrenntes zusammenfassen

# Überlegungen

- Zugriffsregelung
  - Direktzugriff: Direkter Aufruf der Hardware
  - Hardware virtualisieren: Hardwareanforderungen werden an virtualisierte Hardware gestellt oder werden abgefangen
- Ressourcenaufteilung
  - reelle Hardware muss nun auf mehrere virtuelle Systeme verteilt werden
- Einfluss zwischen den Systemen
  - Speicherzugriffe auf andere Systeme vermeiden

## Vorteile

- Änderungen am Server und Infrastrukturen kosten weniger Aufwand
  - Updates können zentral erledigt werden, statt auf einzelnen Systemen separat gemacht werden zu müssen
- Kosteneinsparung bei Entwicklungen, da benötigte Hardware virtualisiert werden kann (statt gekauft werden muss)
- System speicherbar (Snapshots) Gesamter Systemzustand wird in Daten gespeichert und kann kopiert, portiert und wieder aufgerufen werden
  - wiederherstellbar
  - vervielfältigbar

# Vorteile

- Sichere Umgebung
  - Abschottung gegenüber anderen Systemen
  - Wiederherstellbarkeit nach Fehlern/Abstürzen
  - Absturz eines Systems berührt andere Systeme nicht
- kostengünstige Testumgebung
  - Mehrere verschiedene Systeme auf einer Hardware
  - Zusätzliche oder überschüssige Hardware kann durch Simulation ausgeglichen werden
- Simulieren Emulieren alter Systeme
  - Kompatibilität zu älteren Programmen und Systemen

# Hardware-Emulation

- Gerät soll Hardware nachbilden
  - gleicher Eingabe
  - mit gleichen Programmen
  - sollen gleiches Ergebnis erzielen

## Nachteile

- Performanceaufteilung auf die einzelnen (Gast-)Systeme
  - Einzelne Hardware muss die Arbeit für mehrere Systeme leisten
- Emulationsaufwand (Virtualisierungsschwund) bis zu 25%
  - Hardware muss passend angesprochen werden (ggf. Zwischenschritte nötig)
  - Anfragen auf andere Systeme oder deren Ressourcen müssen abgefangen werden
- Kommunikation zwischen Gast- und Host-System vielleicht problematisch
  - Virtualisierungssoftware muss das Betriebssystem oder die Hardware unterstützen und passend ansprechen
- Wissen über Virtualisierung beim (betreuenden) Personal nötig
- Schlechte Virtualisierung kostet Zeit und Geld



# Servervirtualisierung

- Mehrere Server/Dienste auf einer Maschine
- Weniger Hardware spart
  - Anschaffungskosten
  - Stromkosten - Betrieb und Kühlung
  - Stellplätze
  - Administration

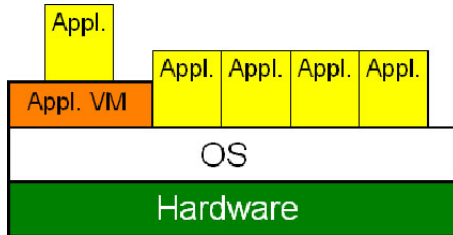
# Clientvirtualisierung

- Rechenzentrum-Infrastruktur
  - Daten(Einstellungen und Applikationen) werden via Netzwerk beim Starten geladen
- PC-Desktop virtualisiert
  - Betriebssystem
  - Anwendungen
  - Einstellungen
- Arbeitsplatz-PC geschützt

# Applikationsvirtualisierung - Überlegung

- Plattformunabhängig
  - nicht jede Applikation auf ein Betriebssystem ausrichten, sondern alle auf eine virtualisierte Maschine, die auf dem Betriebssystem läuft
- Systemschonend
  - Zwischenergebnisse und Daten können in der virtuellen Maschine gespeichert werden und beim Beenden gelöscht werden

# Applikationsvirtualisierung - Grafik



## Applikationsvirtualisierung - Vorteile

- Einfaches Einbinden der Applikation, da keine Verflechtung mit dem OS besteht
  - mehrere Lösungen & Versionen gleichzeitig lauffähig, auch wenn Ressourcen ungeteilt benötigt werden
  - Programm kann Admin-Rechte erhalten und der Virtualisierung nur eingeschränkte Rechte zugestanden werden
- Fehlerhafte programmierte Software ungefährlicher
  - Systemabstürze sind unwahrscheinlicher (da abgestürzte virtuelle Umgebung einfach terminiert werden kann)
  - Originalzustand des Programms kann jederzeit hergestellt werden
- Applikation ohne Installation verfügbar
- Zero Footprint
  - Registry von Windows geschont

## Applikationsvirtualisierung - Nachteile

- Weitere Komplexitätsebene
  - Mehr Zeit in der Planung nötig
- Rechenintensiv
  - Die virtualisierte Maschine muss jede Kommunikation mit dem Betriebssystem angemessen anpassen
- Fraglich, ob Virtualisierungssoftware auf anderen Systemen verfügbar ist (bei gängigen Betriebssystemen sicherlich der Fall)
- bis zu 100 € pro Client

## Applikationsvirtualisierung - Beispiele

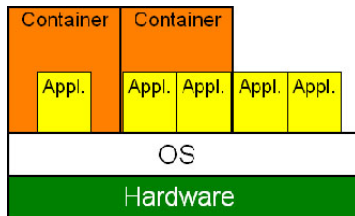
- Java Virtual Machine
- VMware ThinApp
- Xenocode Virtual Application Studio
- Microsoft App-V (Softgrid)

# Betriebssystemvirtualisierung - Überlegung

- Kein neues Betriebssystem, sondern neue Container(Jails) im System
  - Partitionierung des Betriebssystems
  - Mehrere Instanzen als Prozesse nebeneinander
- Unter einem Kernel laufen verschiedene Instanzen
  - Kernel kann allen Instanzen zur Verfügung gestellt werden
- Host-System kann wie gewohnt Ressourcen verteilen
  - Instanzen sind Prozesse, die im Betriebssystem laufen



# Betriebssystemvirtualisierung - Grafik



## Betriebssystemvirtualisierung - Vorteile

- Wenig Speicheraufwand
  - Änderungen und Einstellungen müssen nur gespeichert werden
- Hohe Geschwindigkeit im Gastsystem
- Geringer Ressourcenbedarf

## Betriebssystemvirtualisierung - Nachteile

- Nur ein Betriebssystem kann virtualisiert werden
  - durch gemeinsame Nutzung der Bibliothek kann nur das Host-Betriebssystem virtualisiert werden
- Hoständerungen (in geteilten Bereichen) betreffen Gäste

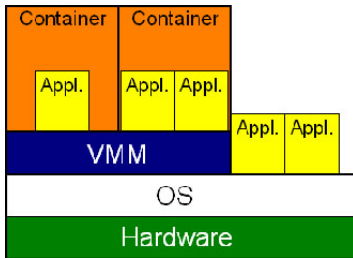
## Betriebssystemvirtualisierung - Beispiele

- FreeBSD Jails
- Solaris Zone/Container
- Open VZ
- Linux Vserver
- Virtuozzo

# Virtual Machine Monitor - Überlegung

- Auf einem Betriebssystem verschiedene Systeme simulieren
  - Gäste können unterschiedlich virtualisiert werden in
    - Hardware
    - Software
- Anpassung des Systems zur Laufzeit
  - nur teilweise möglich

# Virtual Machine Monitor - Grafik



## Virtual Machine Monitor - Vorteile

- Gast muss nicht angepasst werden
  - da Gäste immer den VMM ansprechen
- Gast-Hardware flexibel
  - ohne Abhängigkeit zum Betriebssystem kann die Hardware simuliert werden
- Host und Gäste (OS- und Versions-) unabhängig
  - Gäste nutzen eigene Bibliotheken im Gegensatz zu vorherig genannten Lösungen

## Virtual Machine Monitor - Nachteile

- Hardware muss unterstützt werden
- Hardware- und Prozessorvirtualisierung aufwendig
  - jede Virtuelle Maschine benötigt einen zusätzlichen Prozess, der die Hardwarevirtualisierung übernimmt
- Applikationen laufen optimal, wenn sie auf dem Gast installiert sind



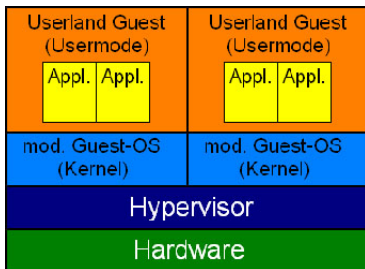
# Virtual Machine Monitor - Beispiele

- VM Ware Server
- Microsoft Virtual Server

# Paravirtualisierung - Überlegung

- Einmalige Virtualisierung der Hardware
  - VM weiß von der virtuellen Hardware
- Hypervisor
  - Abstrakte Verwaltungssicht
  - Verteilt Ressourcen wie Prozessor unter Applikationen
  - Keine Hardware, sondern eine API steht den Gastsystemen zur Verfügung

# Paravirtualisierung - Grafik



## Paravirtualisierung - Vorteile

- flexible Anpassung der Gasthardware
- Abbildung der virtuellen Hardware erfordert nur einen Prozess
  - verbesserter Zugriff auf die virtuelle Hardware durch das Gast-Betriebssystem
  - Geringerer Schwund

## Paravirtualisierung - Nachteile

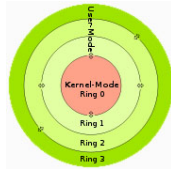
- Hardwarespezifischer als andere Lösungen
- Anpassung der Gast-Betriebssysteme notwendig
- stetige Anpassung bei Updates des Hosts/Gasts erforderlich

## Paravirtualisierung - Beispiele

- Citrix XenServer
- Virtual Iron
- Microsoft Hyper-V

- Die Wahl der Virtualisierungslösung ist stark abhängig von dem Umfang des zu virtualisierenden Bereichs
  - Reine Softwarevirtualisierung ist mit Applikationsvirtualisierung bereits möglich
  - Sollen mehrere Applikationen zusammen in einer virtuellen Umgebung laufen, so werden bereits Container nötig
  - Sollen auch andere Betriebssysteme und Hardware unterstützt werden muss ein Virtual Maschine Monitor eingesetzt werden
  - Wenn die Virtualisierung nicht über das Betriebssystem gehen soll erledigt dies die Hypervisor-Lösung, die direkt auf der Hardware sitzt

## Exkurs: CPU-Ringe



- Um das System vor Zugriffsverletzungen zu schützen nutzen CPUs verschiedene Rechtevergaben
- Ein Prozess darf nur auf Prozesse in weiter außen gelegenen Schichten Einfluss nehmen
- Heute ist es gängig nur Ring 0 für den Kernel des Betriebssystems zu nutzen und den Ring 3 für alle anderen Prozesse



# Hardwarevirtualisierung

- Neue CPUs haben Virtualisierungs-Erweiterung
  - CPU erkennt Ring0-Zugriffe von virtuellen Instanzen und verweigert Zugriffe auf Prozesse außerhalb der Instanz
- Privilegien neu geordnet
- Neue Hypervisorschicht beim Speicherzugriff