Alternativen zur Virtualisierung

Oliver Rath

GreenUnit

Eine subjektive Bestandsaufnahme

Früher im Büro

- Eigener normaler PC als Client
- Zentraler Server für Netzwerkdienste
- Firewall
- Standard-Netzwerk

Eine subjektive Bestandsaufnahme

Früher im Büro

- Betriebsystemlizenz für Client
- Betriebsystemlizenz für Server

Eine subjektive Bestandsaufnahme

Heute im Büro

- Eigener PC als Client mit (z.B.) VNC-Viewer
- Sehr leistungsfähige(r) Server(farm) mit virtuellen Maschinen
- High-Speed Netzwerk

Eine subjektive Bestandsaufnahme

Heute im Büro

- Betriebssystemlizenz für Client
- Betriebssystemlizenz für Server
- Betriebssystemlizenz für virtuelle Maschine
- VDA-Lizenz (Virtual Desktop Access, ab Win7) für die Nutzung von MS-Windows in virtueller Maschinen

Kosten

- Erhöhter Ressourcenbedarf
- Hardwarekosten steigen
- Lizenzkosten steigen (VDA)
- Softwarekosten steigen (Hypervisor)

Vorteile

- Zentrale Verwaltung der Images
- Zentrale Datensicherung
- Sicherheit durch zentrale Datenhaltung
- Jederzeit Zugriff auch von außerhalb

Nachteile

- Hardwarenutzen begrenzt
  - Keine Dongles möglich
  - Leistungsfähige Hardware (z.B. Grafikkarte, Spezialhardware) nur begrenzt oder gar nicht nutzbar
  - Reaktionszeit langsam beim Remotezugriff

Alternativen?

> Wie kann ich die Vorteile beider Welten verbinden ?

Netboot

- Jedes Bios besitzt heutzutage PXE-Boot (Pre Execution Environment)
- Standard von Intel 1999 im Alleingang definiert, nachdem Konsortium scheiterte
- Fähigkeiten sehr begrenzt, da nur Dateien per TFTP-Protokoll geladen werden können
- Benötigte PXE-Informationen werden vom DHCP-Server geliefert

gPXE

- PXE-Erweiterung seit 2008
- Entstanden aus dem Etherboot-Projekt (~1995)
- Erweiterung um HTTP(S), FTP, iSCSI, AoE
- Leider kaum Aktivität mehr seit 2009 (Ausnahme: IPv6-Erweiterung GoogleCode of Summer 2011)

iPXE

- Fork 2010 von gPXE-Mit-Maintainer Michael Brown
- Stabilisierung des Codes
- Menüerweiterung 2012
- IPv6-save
- UEFI-Boot
- Allerdings: Bisher keine „stable“, sondern ausschließlich git-Repository

Funktionsweise von iPXE

- Bios-PXE (LAN-Boot) bootet per TFTP ipxe-Binary
- IPXE übernimmt und bootet erneut (DHCP-Infos können gecached werden), nun aber mit iSCSI
- OS MBR wird geladen
- OS startet und holt Disk-Infos aus **iBFT** (iSCSI Base Firmware Table) und übernimmt mit eigenem Treiber
- Anmerkung: iPXE kann auch ins NIC-Bios geflashed werden (!)

iBFT

> Die iBFT (iSCSI Base Firmware Table) ist eine Daten-Struktur, durch die das PXE-Boot Informationen für nachfolgende Treiber bereithält

Womit funktioniert das?

- Windows 2000/XP/2003/2008/Win7/Win8
- Linux
- *BSD
- Unix
- Apple Leopard (ist aber nicht erlaubt, wenn Ziehlhardware non-Apple ist)

Für Leute mit weniger Zeit als Geld

<Werbung>

- Die GreenUnit UG bietet das Netboot aller Betriebsysteme als Desktop Managment Infrastruktur (DMI) out of the Box an.
- Updates der Plattenimages
- Verwaltung von Volumenlizenzen und Standardlizenzen parallel möglich
- http://www.greenun.it

</Werbung>

GreenUnit

iSCSI

- SCSI-Protokoll over IP
- Kommunikation auf Blockebene der Platten
- (schwache) Security durch Chan/Chap-Autentifizierung
- Starke Security durch IPsec möglich

Weitere Vorteile von iPXE

- Durch die Protokolle FTP und HTTP(s) kann auch sicher übers Internet gebootet werden (z.B. rescue-Linux)
- Boot über Wlan (!) möglich
- NICs müssen iSCSI nicht unterstützen
- Generischer Boot via **UNDI**-Stack möglich (Universal Network Device Interface)

UNDI

- Jede PXE-fähige NIC kann auch UNDI
- UNDI ist ein Standard, um TFTP via NIC zu benutzen, ohne die Hardware näher zu kennen
- Historisch: UNDI-Nic-Drive-Floppy (Novell, DOS)

Risiken und Nebenwirkungen

- Pro
  - Viele OS'se booten schneller als von Platte (v.a. WinXP)
  - Platte liegt zentral vor, Leistung aber vor Ort
  - Vorteile von virtuellen Maschinen gepaart mit Vorteilen klassischer PCs
- Contra
  - Wenn das OS an der NIC-Config schraubt, kann das das System stehenbleiben
  - Installation genau wie klassisches Windows, sprich Installation wird auf Hardware geprägt (sysprep ..)
  - Betrieb steht und fällt mit dem Netzwerk

Risiken und Nebenwirkungen

- Pro
  - Viele OS'se booten schneller als von Platte (v.a. WinXP)
  - Platte liegt zentral vor, Leistung aber vor Ort
  - Vorteile von virtuellen Maschinen gepaart mit Vorteilen klassischer PCs
- Contra
  - Wenn das **OS an der NIC-Config schraubt, kann das das System stehenbleiben (kein Windows-Phänomen)**
  - Installation genau wie klassisches Windows, sprich Netzplatte wird auf Hardware gedongelt
  - Betrieb steht und fällt mit dem Netzwerk

Daher noch ein Weg: AoE

- AoE (**ATA** over Ethernet) benutzt Ethernet-Frames statt TCP/IP
- Erfunden von Brantley Coile (Cisco, Coraid etc., hat auch NAT miterfunden)
- MAC-Adressen als Quell- und Zielkoordinaten

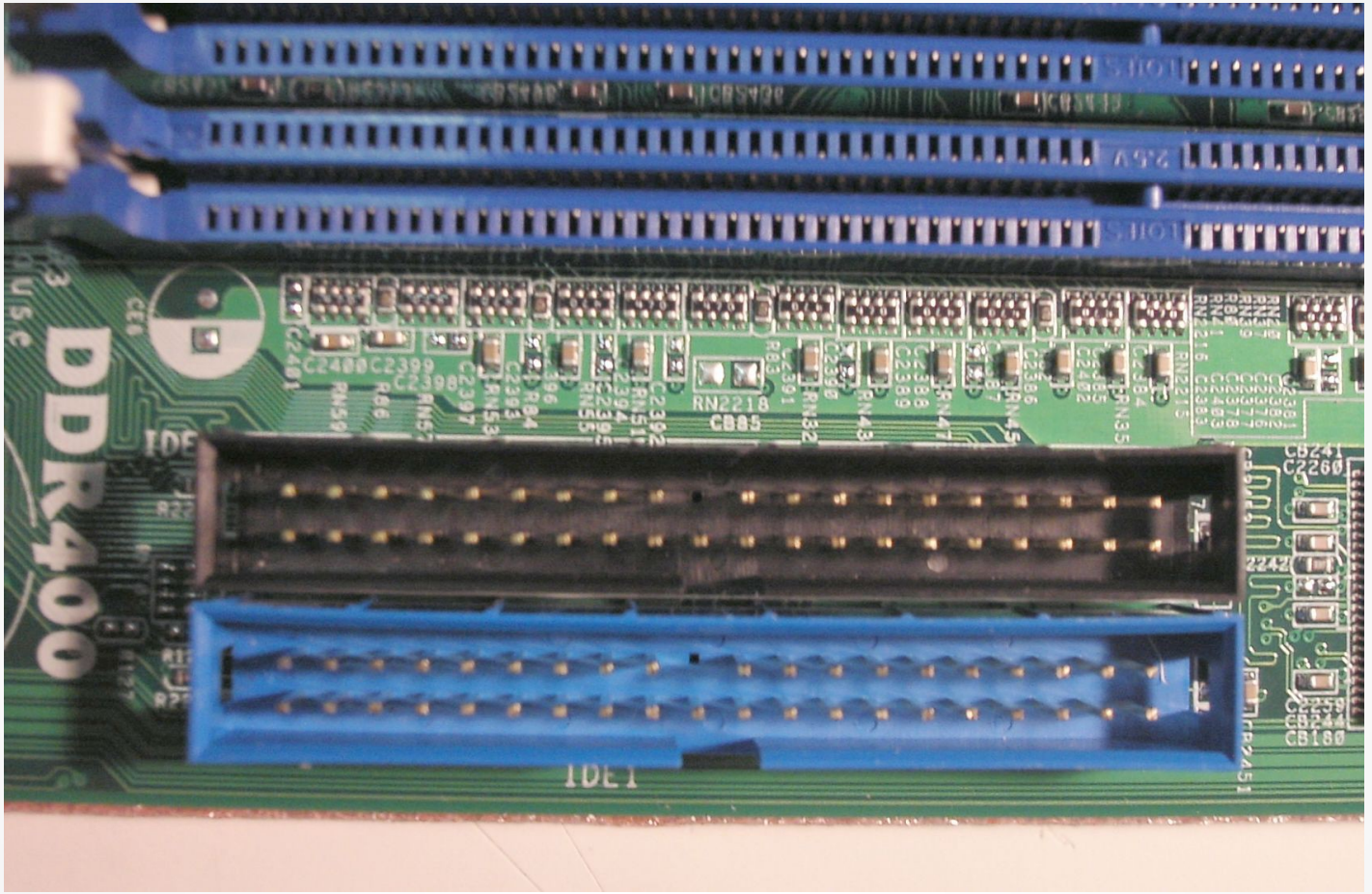
  => größere Frames

  => weniger Overhead
- Unempfindlich ggü. NIC-Rekonfiguration :-)

ATA

- ATA = Advanced Technologie Attachement
- Interface Standard für Storage-Systeme:
  - Parallel ATA (PATA)
  - Serial ATA (SATA)

PATA :-)



GreenUnit

AoE Topologie

- 2 Layer weniger
- TCP/IP-Konfiguration stört den Betrieb nicht mehr
- Einschränkung: Nicht routingfähig

GreenUnit

Implementierung iSCSI serverseitig

- ISCSI Enterprise Target (Ietd, RedHat)
- LIO-iSCSI (Kernel-Modul seit 3.2.0)
- Open-Iscsi Projekt
- Windows-Server 2003+
- Weitere Implementierungen in *BSD, Solaris

Implementierung iSCSI clientseitig

- Linux: Kernelmodul seit 2.6.11
- WinXP: iscsi-Initiator 2.08 seit 2005 (auch für Win2000)
- Seit Vista standardmäßig mitinstalliert

Unterstützung in MS-Windows

- Win2000 benötigt SP4 für iSCSI-Boot
- WinXP kann nicht ohne Zusatzsoftware via iSCSI booten (Ausnahme: iSCSI-capable NICs).
- Initiator-Version für XP steht seit 2005 auf 2.08 (läuft aber gut)
- Ab Vista Netboot standardmäßig eingebaut als Option
- In Win7/Win8 geht’s immer noch ;-)

Implementierung AoE Serverseitig

- vblade-Server (CoRaid, Userspace)
- ggaod (Linux-optimierter AoE-Server, Userspace)
- aoeserver (Kernelmodul, outdated seit 2007)

Implementierung AoE clientseitig

- Linux: Kernelmodul seit 2.6.11
- Windows: seit WinXP Treiber von Coraid

Fernzugriff?

- Netboot auch in VM (z.B. kvm) problemlos möglich
- Einfach weitere Instanz dafür kreieren
- Zugriff via rDesktop, VNC oder **Spice**

Spice

- KVM mit Spice integriert virtuelle Grafikkarte in die VM
- Windows-Treiber für Spice-Grafikkarte installieren
- Spice exportiert OpenGL zum Spice-Viewer
  - => 3D-Fähigkeit bleibt erhalten
  - => Hardware-Beschleunigung nutzbar
- Firefox-Plugin (spice.xpi) vorhanden
- http://www.spice-space.org (Redhat)

Resumee

- Die Vorteile zweier Welten zu verbinden ist möglich
- Der Teufel liegt manchmal im Detail (WinXP)
- AoE zwar schneller und stabiler, aber nicht so verbreitet

Geschafft!

Danke für die Aufmerksamkeit !

Fragen?

GreenUnit